Probabilistic Models for Text in Social Networks

by

Derek Owens-Oas

Department of Statistical Science
Duke University

Date: _____

Approved:

_____
David Banks, Supervisor

_____
Katherine Heller

_____
Alexander Volfovsky

_____
James Moody

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Statistical Science
in the Graduate School of Duke University
2018

Abstract

Probabilistic Models for Text in Social Networks

by

Derek Owens-Oas

Department of Statistical Science
Duke University

Date: _____

Approved:

_____
David Banks, Supervisor

_____
Katherine Heller

_____
Alexander Volfovsky

_____
James Moody

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Statistical Science in the Graduate School of Duke University
2018

Abstract

Text in social networks is a common form of data. Common examples include emails between coworkers, text messages in a group chat, or comments on Facebook. There is value in developing models for such data. Examples of related services include archiving emails by topic and recommending job prospects for those seeking employment. However, due to privacy concerns, these data are relatively hard to obtain. We therefore work with similar data of the same structure which are publically available to design and experiment.

Motivated primarily by topic discovery, this thesis begins with a thorough survey of models which extend the foundational probabilistic topic model, latent Dirichlet allocation. My focus is on those which endow documents with meta data, like a time stamp, the author, or a set of links to other authors. Each approach is given common notation, described in terms of a structural innovation to LDA, and presented in a graphical model. The review reveals, to our knowledge, there was previously no model which combines dynamic topic modeling and community detection.

The first data set studied in this thesis is a corpus of political blog posts. Our motivation is to learn communities, guided by the presence of links and dynamic topic interests. This formulation enables new link recommendation. We therefore develop an appropriate Bayesian probabilistic model to learn these parameters jointly. Experiments reveal the model successfully identifies a groups of blogs which discuss sensational crime, despite having very few links between these blogs. It also enables

presentation of top blogs, according to various criteria, for a specified topic interest community.

In a second analysis of the blog post data I develop a similar model. The motivation is to partition documents into groups. The groups are defined by shared topic interest proportions and shared linking patterns. Documents in the same group are reasonable recommendations to a reader. The model is designed to extend the foundational LDA. This enables easy comparison to a strong baseline. Also, it offers an alternative to LDA for situations where a hard clustering of documents is desired, and documents with similar enough topic proportions are clustered together. It simultaneously learns the linking tendency for each of these groups.

We show a different application of a probabilistic model for text data in social networks to related text event sequence data. Here we analyze a transcription of group conversation data from the movie 12 Angry Men. A main contribution is an algorithm based on marked multivariate Hawkes processes to recover latent structure, learning the root source of an event. The algorithm is tested on synthetic data and a Reddit data set where structure is observed. The algorithm enables partial credit attribution, distributing the credit over likely people who start each new conversation thread.

The above models and applications demonstrate the value of text network data. Generalized software for such data enables visualization and summarization of model outputs for text data in social networks.

This thesis is dedicated to my parents, brother, and other family members who support and encourage me. It's also dedicated to my girlfriend and friends who bring me joy and inspire me and for whom I work so hard. It's for academics who have contributed to related work and those in industry who turn the work into services.

Contents

List of Tables

List of Figures

List of Abbreviations and Symbols

The following symbols and abbreviations are used throughout this dissertation:

Symbols

$\mathbf{X}$	all observed data
$\mathbf{\Theta}$	all parameters of the model

Abbreviations

DTN	dynamic text network
LDA	latent Dirichlet allocation
MCMC	Markov chain Monte Carlo
EM	expectation maximization
PFA	Poisson factor analysis
RTM	relational topic model
MMSB	mixed membership stochastic block model
TDM	term document matrix
iid	independent and identically distributed
LSA	latent semantic analysis
SVD	singular value decomposition
AT	author topic model
ART	author recipient topic model
RART	role author recipient topic model

DTM	dynamic topic model
TOT	topics over time model
ELBO	evidence lower bound
PLSA	probabilistic latent semantic analysis
TF-IDF	term-frequency inverse-document-frequency
TF-ITF	term-frequency inverse-time-frequency
ERGM	exponential random graph model
TERGM	temporal exponential random graph model
GSDMM	Gibbs sampler for the Dirichlet mixture model

Acknowledgements

I want to thank my adviser, David Banks, for the original research idea of combining probabilistic models for text and networks and for recommending I write a review of topic models. This is the foundation for my dissertation and the main ideas for chapters 2 and 3. Also thanks for outlining the blogs paper, writing the abstract, and writing the intro. Thank you also for providing me with funding and finding external funding for me. I also appreciate your feedback on numerous presentations and willingness to serve on my committee. Teague Henry is a very important part of this work. Teague, thank you for deriving the inference algorithm, writing code, and presenting the experimental results in chapter 2. Christine Chai was present for some of this work.

Thanks also to Katherine Heller for inviting me to join you and your collaborators with the Hawkes process research. This provided me with an additional application for text and network modeling and got me started on chapter 4. I also appreciate your funding for research and conferences. Your edits on our paper submission are appreciated, as are your feedback on presentations and willingness to serve on my committee. Thanks also to collaborator Wei Zhang. Your work writing the model specification, deriving the inference algorithm and root probability algorithm, writing the code for the estimation, and performing the simulated data analysis were crucial. Also Fan Bu contributed significantly, writing the literature review, consolidating the appendix, and creating figures for the tree plots. Thank you for your work. Jerry

Zhu was a coauthor. Peter Hase helped write code to construct a comment tree.

Thanks to James Moody with the Duke Network Analysis Center, for funding and feedback on a presentation. Alexander Volfovsky also provided feedback on presentations and a foundational knowledge of statistical network literature. Thank you both for serving on my committee and bolstering my knowledge of network modeling.

1

Introduction

Since the emergence of the internet, social networks have become a popular venue for human interaction. Social networks accumulate enormous amounts of data, which often contain text. The text of hyperlinked blog posts, group conversation transcriptions, and Reddit comment threads among users can be considered social network data with associated text. In this thesis I focus primarily on instances where the network and text content are evolving over time. I will refer to this form of data as dynamic text networks.

First I focus on describing the similarities between the above three data sources. In each case, we can separate the data into observational units which have associated text. The observational unit in these cases is a blog post, a spoken comment, and a Reddit comment, respectively. In the group conversation transcript, we define an observation as all words spoken by a person between adjacent comments from other people, split into multiple comments when the time between words exceeds a threshold. Along with the text of each observation are up to three covariates. The first is a time stamp, giving time information regarding the text occurrence. These may be discrete or continuous. They are a date of 2012 for the blog posts and the

amount of time since the conversation started or since the thread opened, for verbal and Reddit comments, respectively. The second covariate is the source which emitted the observed text or comment. For the blog posts, this is the web domain on which the post is written, for example, *businessinsider.com*; for conversation data, this is the person who spoke the words; for Reddit, this is the user who wrote the comment. Third, each observation may have a set of links to other sources. This only is present in the blog posts and is comprised of hyperlinks which occurred in the text.

I describe the data in this way to emphasize a common format. Specifically, this thesis focuses on applications in which the data can be described as follows. Consider an observation $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$ of N data points. Each data point $\mathbf{x}_n = (t_n, s_n, \boldsymbol{\ell}_n, \mathbf{w}_n)$ can be characterized by the time at which it occurred t_n, its source s_n, its set of links $\boldsymbol{\ell}_n$ to other sources, and an ordered sequence $\mathbf{w}_n$ of words. Having this shared notation is particularly relevant in the development of software or methodology for data. Analyses designed for this format of data can be applied to other data.

Despite the fact that all three applications fall into this framework, the way the data sets originate are different. We consider two cases. In the first case we have blog posts, and as above, each can be written $\mathbf{x}_n = (t_n, s_n, \boldsymbol{\ell}_n, \mathbf{w}_n)$, where we observe time discretely. Therefore I introduce a date list $\mathbf{t}^* = \{t_1^*, ..., t_T^*\}$, which enumerates chronologically the dates of the year (ie. t_1^* = January 1, 2012). Both the the verbal and Reddit comments are instances of a second case, in which each observation is written $\mathbf{x}_n = (t_n, s_n, \mathbf{w}_n)$. Here links are unobserved, and time is continuous, so $t_n \in \mathbb{R}_+$. In words, this dissertation focuses on probabilistic models for text data in dynamic social networks.

In Chapter 2, I review existing methodology for learning topics in a corpus of documents, when additional information is available. This review focuses on probabilistic models which extend the foundational topic model, latent Dirichlet allocation (LDA), described in Blei et al. (2003b). The primary contribution of the review is to

provide a catalog from which an appropriate topic model can be selected for various applications. In doing so, I provide a common notation, so that shared parameters and data in each model are clearly identified. Specifically, the document topic distributions, topic word distributions, word topic indicators, and words which comprise the LDA model act as a framework on which the other models build. In this notation, I present the graphical model for each approach as a plate diagram. This further emphasizes the common architecture of each LDA extension. Next the method of parameter estimation is described for each model. All approaches considered in this thesis use some version of the Markov chain Monte Carlo (MCMC) method, often Gibbs sampling, or an optimization method, usually variational expectation maximization (EM). To clarify for which setting each model is designed, I list which data sets are used in the applications of each model. For example, for each method, I mention whether the authors apply it to scientific abstracts, which tends to be a common application of topic models which include link information. Also, even when applying models to the same data, the goals of analysis can differ. Therefore, I also present the goal of analysis for each method. Common goals of analysis include learning topics and discovering communities among document sources. Information about the models, including which attributes of the data and parameters are explicitly modeled, is summarized in Table 2.2. Next, I mention other LDA based extensions which are not discussed in detail. The review conveys that different corpora have subtle distinctions which motivate use of specialized models and inference algorithms.

In Chapter 3 I focus on analyzing a corpus of political blog posts. Work in this chapter is drawn from Henry et al. (2016), which is submitted to the Journal of Classification and under revision. The work is included in this dissertation, because it exemplifies an application for a probabilistic model for text data in a social network. Our approach is application motivated and develops a method which combines topic

modeling with community detection. My contribution was designing a model which can answer questions of interest from a blogger, politician, or news reporter. The primary question of interest regarding blog posts is: what are they about? Topic modeling addresses this question, by learning a set of topics which occur throughout the text. I'll use topic modeling terminology henceforth. Topic modeling can provide:

- topics which occur throughout the corpus,
- topics which occur in each document,
- and the most probable words in each topic.

Also of importance in the blog post data is answering the question: what are the groups of blogs which tend to link to one another? This is the goal of community discovery. In more detail, community discovery can provide:

- communities of nodes which tend to link to one another,
- the assignment of nodes into communities,
- a way to predict future links.

Last, an analyst may wish to identify events. In the data source itself, there is no information available about when, for example, the presidential primaries began. Event identification finds:

- the time stamps of significant events.

Because the blog posts data set contains words, nodes and links, and time stamps, various combinations of the above contributions are available. First, for each community, we reveal which topics are of interest, and second, for each day, we reveal the most probable words of each topic.

To achieve these goals, we utilize Bayesian statistics to reason about the population of political blogs given our sample. The first step is defining a generative model which is parameterized to address the points mentioned above. Specifically, we propose a novel Bayesian probabilistic generative model which includes three primary components. First are dynamic topics, each of which allows word probabilities to vary over time. Second are communities, which are defined by a set of interest topics and which guide linking behavior. Third are latent event indicators which identify when the post rates for a particular topic elevate. Parameter inference uses Gibbs sampling to iteratively approximate each parameter's posterior distribution. When an analytic conditional posterior is not available, Metropolis Hastings provides a means of sampling from the posterior.

With these posterior samples we analyze a set of blog posts from top political blogs of 2012. We find an election topic, with word usage that varies in time to accommodate evolving political discussion as the election cycle progresses through primaries, nominations, debates, voting, and election. In terms of communities, we discover one community of 21 blogs which have primary interest in only the sensational crime topic. These blogs don't link with one another vary often, nor do they tend to receive many links. They do however link to external blogs frequently. Therefore they can be considered a community of commentator blogs that react to content from other blogs. This type of community doesn't follow the assumptions of traditional community detection algorithms and is presumably discovered by its shared topical interests.

A second analysis of the blog posts is given in Chapter 4. This approach is methodologically motivated. The first goal is to reduce the number of parameters by letting documents with similar topic distributions share a parameter. The second goal is to get a hard partition of documents, where group membership depends on both topic interests and linking tendency. The topic link block model achieves both

these goals with a model which extends LDA. Parts of the inference are parallelizable which theoretically helps the method scale to larger data sets, because conditional independence allows for document specific parameters to be inferred in blocks. The model identifies topics similar to those found by the relational topic model (RTM), though without an additional pre-processing step. Furthermore, I show how document cluster assignments can be used to learn communities of nodes and compare a visualization of learned communities with those of the mixed membership stochastic block model (MMSB). Model outputs are helpful for visualizing a network with associated documents.

In Chapter 5, I present work involving a model for dynamic text network data which arose out of related text event sequences literature. Specifically, Zhang et al. (2018a) use a marked multivariate Hawkes process to specify a model for text events where node and time information are also available. This work is included to show that there are applications other than topic modeling for probabilistic models for text in social networks. The goal of this analysis is a task we call *root source identification*, which is to identify, for each text event, the node initially responsible for starting the current conversation. The primary contribution is a linear time algorithm to compute root source probability distributions over nodes. The task can be useful for attributing credit or blame for initiating a sequence of text events and for learning a quantification of innovation and influence for each node. Experiments on simulated data show successful root source recovery. I include the generative model, parameter estimation, and root source identification algorithm in Chapter 5 for completeness, and acknowledge this work was done primarily by coauthors in Zhang et al. (2018a). My primary contributions in this work are threefold. First, I apply of the approach to real world data. On group conversation data from a transcription of the subtitles from the film *12 Angry Men* (Richard Guo, 2012), our model confirms human intuition about which nodes are most innovative and correctly identifies root source in cases in

which word inheritance occurs. In a Reddit comment tree (Pushshift, 2017), the root source identification algorithm outperforms all baselines we compare with in terms of accuracy and conditional log probability. Second, I develop empirical Bayes priors for two model parameters and hyperparameter specifications to improve accuracy. Third, I modified existing code from the *igraph* package in *R* to generate tree and network plots (Csardi and Nepusz, 2006).

After discussing existing topic modeling literature for text data in social networks and discussing application of three probabilistic models, I conclude with a summary of results found and the state of the field. Models discussed in the review of Chapter 2 and the blog post applications of 3, and 4 are capable of identifing observations with similar topics and differing communities for friendship recommendation and similar community but differing topics for content recommendations. The model of Chapter 5 can help for credit attribution in related text event sequences.

2

A Literature Review of LDA-Based Topic Models

2.1 Topic Models for Social Network Data

Since the emergence of the internet, social networks have become an increasingly large source of data. Often each node emits text data, which can be grouped into observations of documents. It can be helpful to summarize the topical contents of documents, so browsing and archiving are more efficient. To extend analysis of a sample corpus to a larger population, scientists often address uncertainty with a probabilistic model. In early models, topical composition of one document is modeled as conditionally independent of others. However, network data about the source of a document and its relations with other document sources can inform topic discovery. We review the evolution of topic and network models as separate disciplines and discuss recent attempts to marry them. We call the intersection network topic models. This is a technical comparison of latent Dirichlet allocation (LDA) (Blei et al., 2003b) based topic model architectures. The motivations are to provide a catalog of top topic models for various applications and to clarify the state of the literature, exposing open areas.

2.2 Topic Modeling Applications

Topic models group items across observations into latent topics. The most common application is the categorization of words across documents in a corpus. Documents are things like tweets, emails, or conference abstracts. In this setting it can reveal semantic meaning which is helpful when summarizing, searching, classifying, and browsing documents. Outside of the text domain, recommender systems for music, movies, and products can be described in a topic modeling framework with a user by item matrix. Additionally, topic models can help doctors with health records to group patients into latent categories (ie. by their symptom counts). For clarity, we maintain document modeling as an ongoing example.

Latent Semantic Analysis

The foundational paper on latent semantic analysis (LSA) (Deerwester et al., 1990) proposes an early and effective method for dimension reduction in large document collections. The approach makes a "bag of words" assumption, approximating the meaning of a document by the sum of the meanings of its words. This strategy addmittedly ignores word order to reduce compuational cost substantially. The counts of each word in each document are then stored in an $N \times V$ term document matrix (TDM). By using a singular value decomposition (SVD) (De Lathauwer et al., 2000) of the term document matrix, one can represent each document and each word in terms of its position in a K-dimensional space. For cases where the number of documents or number of terms is large, one can choose a smaller K to significantly improve the computational efficiency for learning algorithms.

Probabilistic Latent Semantic Analysis

Probabilistic latent semantic analysis of Hofmann (1999) is an extension which allows for uncertainty in the topic assignment of each document. Specifically, it allows for

documents to be comprised of a mixture of topics.

Assumptions

Like the previous methods, the papers described in this review adopt the "bag of words" assumption, modeling document specific word counts rather than sequences and reducing computational cost substantially. The models considered here assume each document's words are generated first by choosing a topic and then by choosing a word from that topic, which is a distribution over unique words.

2.3 LDA-Based Topic Models

The foundational LDA paper extends probabilistic latent semantic indexing by modeling the document topic proportions as a random variable with a Dirichlet prior. This theoretically allows some sharing of information across documents, centering estimates of a document's topic proportions around some corpus level proportion, potentially improving estimation in settings with short documents and preventing over fitting.

Notation for LDA-Based Topic Modeling

One of our contributions is translating all models to have a shared notation. This enables clear comparison of model architectures. The notation for indices, hyperparameters, parameters, latent variables, and data is given in Table 4.1 below.

Table 2.1: Notation for LDA-Based Topic Modeling

d	:	document index
n	:	word index
k	:	topic index
$\boldsymbol{\alpha}$	:	document topic proportions concentration
$\boldsymbol{\beta}$	:	topic specific word proportions concentration
$\boldsymbol{\theta}_n$	:	topic proportions for document d
$\boldsymbol{\phi}_k$	:	word proportions for topic k
$z_{n,d}$	:	topic assignment for word n of document d
$x_{n,d}$	:	term assignment for word n of document d

2.3.1 The LDA Model

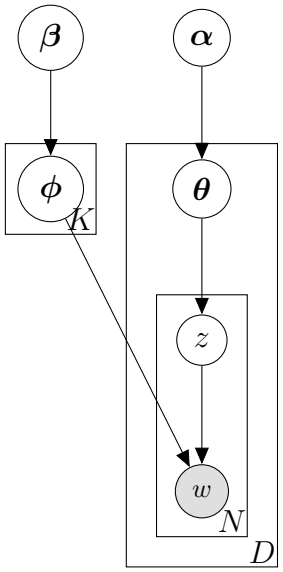
The LDA model is a framework on which many other topic models originate. There are colloquial uses of the LDA framework which differ slightly, so we define a specific LDA model below for clarity.

Smoothed LDA

We begin by recalling that each topic is defined as a distribution over words in a dictionary. The LDA model assumes each document has an underlying topic distribution, summarizing its contents at a high level. The topic distributions are independent and identically distributed (i.i.d.) from a K-dimensional Dirichlet distribution ($Dir_K(\boldsymbol{\alpha})$), with $\boldsymbol{\alpha}$ the concentration parameter vector. Each word is assigned a topic label from the document's topic distribution which ultimately accounts for uncertainty in the topic discussed. These are i.i.d. from a Categorical distribution ($Cat(\boldsymbol{\theta}_d)$) with probabilitiy parameter vector $\boldsymbol{p} = \boldsymbol{\theta}_d$. The highest probability words of each topic can be used to summarize a corpus, and topic proportions can be used to identify similar documents.

Generative Model The well known data generating model for smoothed LDA from Blei et al. (2003b) is:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$



Inference The authors use variational inference to approximate the joint posterior for each document's topic distribution and every word's topic assignment. Other parameters are then estimated using empirical Bayes followed by an expectation maximization (EM) algorithm.

2.3.2 Including Author Information

Multiple topic models exist which extend LDA by including author information. In this subsection we discuss two of them.

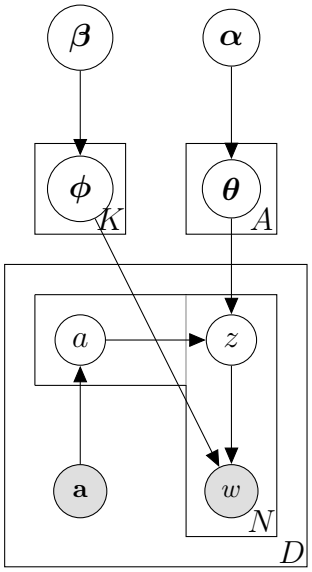
Author Topic Model

In many situations there are authors associated with each document. The author topic model (AT) of Rosen-Zvi et al. (2004) extends LDA by endowing each author with a distribution over topics $\boldsymbol{\theta}_a$. When multiple authors contribute to the document, the topic proportions are expressed as a mixture of those specific to each author. A latent variable indicating the author responsible for writing each word is sampled from the known authors of the paper with uniform probability distribution. This is acheived by sampling a Categorical distribution over authors with probability

parameters $\{p_1, ..., p_{A_d}\} = \frac{1}{A_d} 1(\mathbf{a} \in \mathbf{a}_d)$. The model can be used to assess similarity of authors in terms of their topical interests. Relative to LDA, the model tends to perform better when the number of training documents and number of topics are smaller.

Generative Model The data are generated as follows:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each author a:
   (a) $\boldsymbol{\theta}_a \sim Dir_K(\boldsymbol{\alpha})$
3. For each document d:
   (a) For each word n:
      i. $a_{n,d} \sim Cat(\frac{1}{A_d} 1(\mathbf{a} \in \mathbf{a}_d))$
      ii. $z_{n,d} \sim Cat(\boldsymbol{\theta}_{a_{n,d}})$
      iii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$



Inference Parameters of interest are the topic specific word distributions $\boldsymbol{\phi}_k$, author specific topic proportions $\boldsymbol{\theta}_a$, and latent word specific topic assignments $z_{n,d}$ and author assignments $x_{n,d}$. To infer these parameters, the authors marginalize out ϕ and θ with Dirichlet integrals, and use Gibbs sampling to sample the posterior of $z_{n,d}$, $x_{n,d}$ as a block pair. The latent counts can be used to estimate $\boldsymbol{\phi}_k$ and $\boldsymbol{\theta}_a$. Hyperparameters for the author specific topic distributions and topic specific word distributions are not estimated and are set to $\boldsymbol{\alpha} = 50/K$ and $\boldsymbol{\beta} = 0.01$.

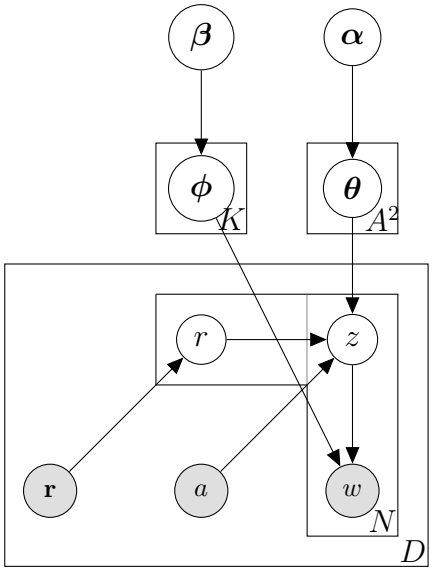
Author Recipient Topic Model

The author recipient topic (ART) model of McCallum et al. (2005) is a social network analysis model for language content which associates topics with directed relations

between nodes of a network. While the AT model endows each document with author information, this work differs by considering recipients. This is accomplished by giving each sender-recipient pair a distribution over topics. In the paper, authors demonstrate the model's ability to learn topics and predict roles in two email data sets. The first contains ≈ 0.5 million emails from 150 people (Shetty and Adibi, 2004) associated with the Enron energy company. The second is comprised of personal emails from a professor, Andrew McCallum. The model also can be used to infer person specific topic composition for both sent and received messages. This provides a means of clustering people based on similarity of the topic distributions of their received emails, identifying roles regardless of shared connections. An extension, the role author recipient topic (RART) model, endows each person group membership probabilities and conditions topic assignments on pairs of these groups.

Generative Model The data generating procedure is below:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each author, recipient pair a, r:
   (a) $\boldsymbol{\theta}_{a,r} \sim Dir_K(\boldsymbol{\alpha})$
3. For each document d:
   (a) For each word n:
      i. $r_{n,d} \sim Unif(\mathbf{r}_d)$
      ii. $z_{n,d} \sim Cat(\boldsymbol{\theta}_{a_d, r_{n,d}})$
      iii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$



Inference The authors first specify the joint distribution of the words, topic and recipient assignments, topic proportions, and topic specific word proportions. Marginal-

izing our the latent variables gives an expression for the marginal likelihood of the documents' words. The exact posterior of the latent variables of interest, the word specific topic and recipient assignments, is not a known distribution. The authors therefore resort to collapsed block Gibbs sampling for posterior inference. First they integrate out topics to find the probability of words given latent word specific topic and author assignments. The technique is similar to obtain the joint distribution of these latents. Combining, Bayes theorem enables writing the posterior for the latents given the words, though the denominator can't be calculated. Instead, solving for $p(z_i, x_i, w_i|\mathbf{z}_{-i}, \mathbf{x}_{-i}, \mathbf{w}_{-i})$ facilitates Gibbs sampling the latent variables (x_i, z_i) as a block-pair from their posterior conditionals $p(z_i|\mathbf{z}_{-i}, \mathbf{x}, \mathbf{w}$ and $p(x_i|\mathbf{z}, \mathbf{x}_{-i}, \mathbf{w})$, respectively.

2.3.3 Including Time Information

Other topic models extend LDA by including information about a time associated with each document. In this subsection we discuss two of them.

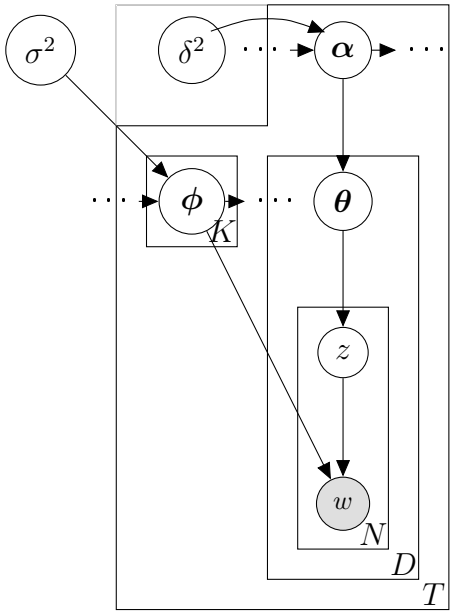
Dynamic Topic Model

The dynamic topic model (DTM) (Blei and Lafferty, 2006) uses a state space model on the natural parameters of the Dirichlet distribution from which topic specific word distributions are drawn. One contribution is a model capable of predicting topical composition of future documents. The model also provides a simplified qualitative analysis of the textual content of a corpus, by learning a number of time evolving topics, each with smoothly varying word probabilities. The concentration parameter for the distribution of document topic proportions also evolves auto-regressively.

Generative Model The generative model is:

For each time t:

1. $\boldsymbol{\alpha}_t \sim Norm_K(\boldsymbol{\alpha}_{t-1}, \delta^2 \mathbf{I})$
2. For each topic k:
   (a) $\boldsymbol{\phi}_{k,t} \sim Norm_V(\boldsymbol{\phi}_{k,t-1}, \sigma^2 \mathbf{I})$
3. For each document d:
   (a) $\boldsymbol{\theta}_{d,t} \sim Norm_K(\boldsymbol{\alpha}_t)$
   (b) For each word n:
      i. $z_{n,d,t} \sim Cat(\pi(\boldsymbol{\theta}_{d,t}))$
      ii. $w_{n,d,t} \sim Cat(\pi(\boldsymbol{\phi}_{z_{n,d,t},t}))$
      where
      $\pi(\boldsymbol{\phi}_{k,t})_v = \frac{\exp(\phi_{v,k,t})}{\sum_v \exp(\phi_{v,k,t})}$



Inference Parameters of interest include time specific topics $\boldsymbol{\phi}_{k,t}$, document specific topic proportions $\boldsymbol{\theta}_{d,t}$, and latent word specific topic indicators $z_{n,d,t}$. To distinguish from mean field approximation (independence), the authors ensure sequential structure remains by using Gaussian variational observations. Free document level variational parameters are optimized with coordinate ascent and have a closed form, while topic level variational observations use a conjugate gradient method. The time evolving topic parameters are approximated with a variational Kalman filter or a wavelet regression.

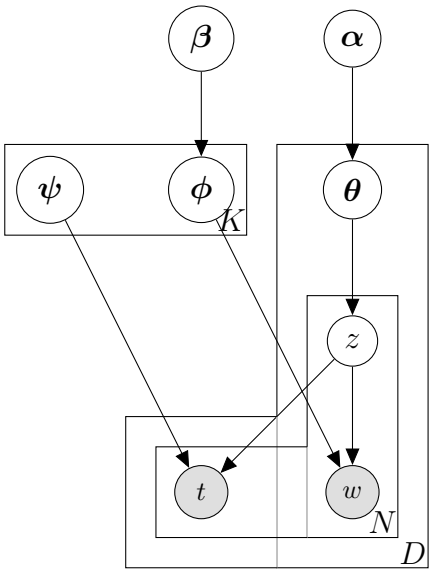
Topics over Time

Topics over time (TOT) (Wang and McCallum, 2006) uses both word co-occurrence and the time stamps to estimate each document's topic distribution. It differs from the Dynamic Topic Model, because it is non-Markov and is continuous time. Each document only has one time stamp, but the authors assign this time to each word of the document. The generative model for a document is then the same as LDA,

with the addition of these time stamps, which are drawn from a distribution with parameter dependent on that word's topic. The time stamp corresponds to the proportion of the way the word is through the total time interval of the corpus. TOT is capable of modeling long term topic dependencies in time, predicting the time stamp for an unstamped document, and predicting the topic distribution for a document with a specified time stamp.

Generative Model The data generating model is:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$
      iii. $t_{n,d} \sim Beta(\boldsymbol{\psi}_{z_{n,d}})$



Inference Primary parameters of interest are $z_{n,d}$ and $\boldsymbol{\psi}_k$. Multiple categorical variables are multinomial distributed, and Dirichlet-multinomial conjugacy is used to integrate out $\boldsymbol{\phi}_k$ and $\boldsymbol{\theta}_d$. The authors use Gibbs sampling to infer $z_{n,d}$ given the data and other parameters. After each Gibbs sample, the topic specific $\boldsymbol{\psi}_k$ beta distribution parameters for $t_{n,d}$ are updated using the method of moments. Hyperparameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can be estimated with Gibbs EM. Posterior estimates for $\boldsymbol{\phi}_k$ and $\boldsymbol{\theta}_d$ can be collected after Gibbs sampling, presumably via $z_{n,d}$.

2.3.4 Modeling Links Between Documents

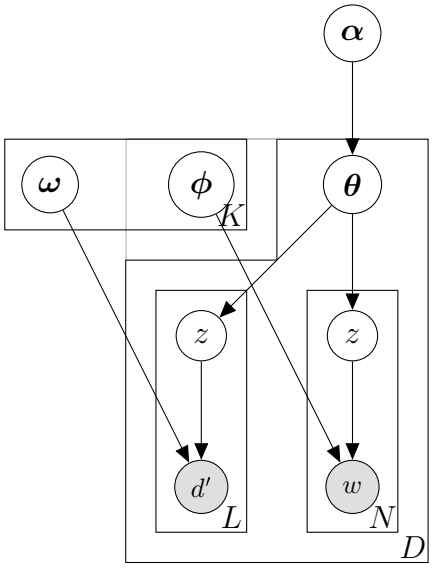
An interesting group of models which extend LDA are those which model links between document. In this subsection we discuss five of them.

Link LDA

In the situation where documents have links, the model of Erosheva et al. (2004), which eventually became known as Link-LDA, is one of the first to simultaneously model links and words. Like LDA, this hierarchical Bayesian model posits a set of K latent topics; however, each is characterized by an additional distribution over citations. That is, at the document level, each citation is drawn from a mixture of topic specific distributions over citations. The model is designed to soft classify each scientific article d, by assigning it partial membership proportions in a variety of topics $\boldsymbol{\theta}_d = \{\theta_{1,d}, ..., \theta_{K,d}\}$, given the words $\{x_{n,d}\}_{n=1}^{N_d}$ in their abstract and their citations $\{a_{l,d}\}_{l=1}^{L_d}$.

Generative Model The model generates data as follows:

1. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$
   (a) For each link l:
      i. $z_{l,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $d'_{l,d} \sim Cat(\boldsymbol{\omega}_{z_{l,d}})$



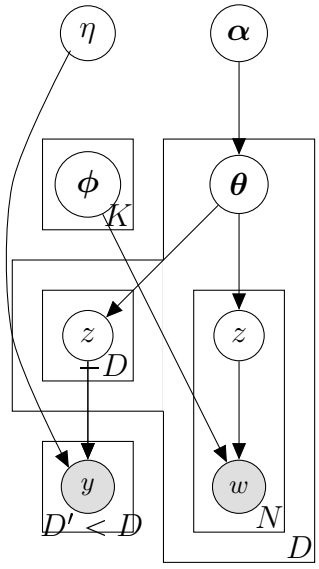
Inference The integral in the marginal likelihood of a document given topics cannot be explicitly evaluated. One approach uses a variational approximation which bounds the mixture of topics below, enabling integrability, and then maximizes the bound. A second approach uses expectation-propagation to approximate the topics and topic proportions with moment matching. With the marginal likelihood approximated, EM can provide parameter estimates.

Pairwise Link LDA

In the Pairwise Link LDA model of Nallapati et al. (2008), the authors combine aspects of LDA and the mixed membership stochastic block model (MMSB) of Airoldi et al. (2008). Conceptually, for each directed document pair d and d', the sender and recipient each draw a link specific topic assignment $z_{d,d'}$ and $z_{d',d}$ in proportion to their topic proportions θ_d and $\theta_{d'}$, respectively. Links are then generated between documents with the corresponding topic pair specific probability $\eta_{z_{d,d'},z_{d',d}}$. The asymmetric matrix $\boldsymbol{\eta}$ allows the probability of a link from a large scale inference publication to a social networks publication to differ from the probability of a link in the opposite direction. The computational complexity scales with the square of the number of documents and therefore is not amenable to large corpora. The experiments involve outgoing citation prediction among blog posts and academic publications.

Generative Model Data are generated as shown below:

1. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$
2. For each pair of documents d, d':
   (a) $z_{d,d'} \sim Cat(\boldsymbol{\theta}_d)$
   (b) $z_{d',d} \sim Cat(\boldsymbol{\theta}_{d'})$
   (c) $y_{d',d} \sim Bern(\eta_{z_{d,d'},z_{d',d}})$



Inference The approach is to integrate out the word topic assignments and document topic proportions to write the marginal likelihood of words and links as a function of topic proportions, prior concentration parameter, topic pair specific link probabilities, and topic word distributions. The exact posterior for document specific topic proportions and word topic assignments is intractable, so the inference procedure uses a mean field variational approximation. The resulting algorithm uses the digamma function to evaluate Dirichlet integrals and involves iterative updates for the variational parameters, the topics, and the topic pair specific link probability matrix. Within this routine, the update for the probability matrix is one document pair at a time, and proceeds by iterating between the two documents until convergence. The rest of the inference then continues until the evidence lower bound (ELBO) converges. Recall the ELBO is the negative Kullback-Leibler (KL) divergence from the posterior to its approximation plus the natural log of the marginal likelihood. For citing documents, updates for topics and the probability matrix are excluded. Furthermore, only the variational parameter for the distribution of topic

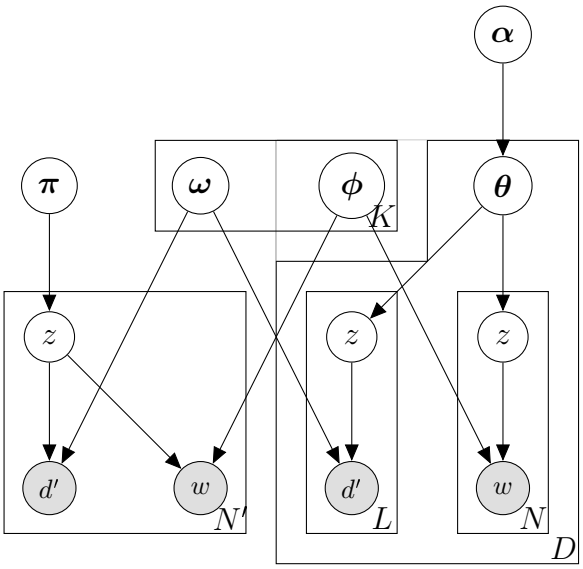
proportions and for word topic assignments are updated, while holding the values for cited documents constant. The experiments involve outgoing citation prediction among blog posts and academic publications.

Link-PLSA-LDA

The Link-PLSA-LDA model (probabilistic latent semantic analysis) of Nallapati and Cohen (2008) incorporates positive features of the Link-LDA model and the Pairwise Link LDA model. Like the first, it models each link as a categorical sample, allowing for computational scalability. Like the second, it explicitly models topic content of the cited and citing document. To achieve this the authors assume every document is either a cited or citing document but not both. For citing documents, data generation is the same as Link-LDA. For cited documents, word allocation is generated as in PLSA (Hofmann, 1999), using the same matrix of topic specific citation proportions as the citing documents.

Generative Model Data are generated as follows:

1. For word n' of cited documents:
   (a) $z_{n'} \sim Cat(\boldsymbol{\pi})$
   (b) $d'_{n'} \sim Cat(\boldsymbol{\Omega}_{z_{n'}})$
   (c) $w_{n'} \sim Cat(\boldsymbol{\phi}_{z_{n'}})$
2. For citing documents d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$
   (c) For each citation position l:
      i. $z_{l,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $d'_{l,d} \sim Cat(\boldsymbol{\Omega}_{z_{l,d}})$



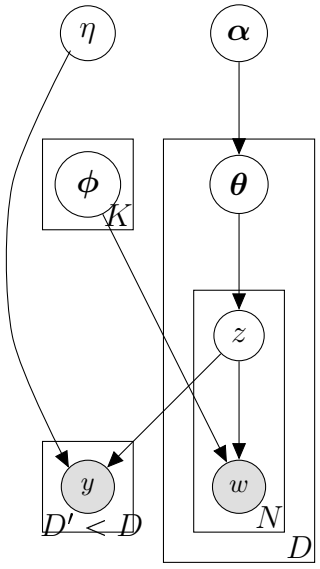
Inference The authors write the marginal likelihood of the words and citations of cited and citing documents as a function of cited document topic proportions, citing documents' topic proportions' prior concentration parameter vector, topic citation probabilities, and topic word distributions. To derive a posterior distribution for the hidden variable topic proportions and topic assignments, the paper includes a mean field variational approximation. Parameters are updated iteratively. To encourage reasonable values of the variational citing document link topic proportions and word proportions and the document topic proportions probability parameter, there is a nested iteration through these parameters until convergence within each outer iteration.

Relational Topic Model

In some situations, links exist between documents. Here, the relational topic model (RTM) of Chang and Blei (2009a) can summarize topical contents, predict between-document links given contents, and predict words within documents given links. Graphically, it is the same as LDA except links between each pair of documents are conditioned on those documents' word specific topic assignments. Multiple link functions (logistic and exponential) are proposed. The model's predicting ability is evaluated on large data sets of networked scientific abstracts and web pages.

Generative Model The data generation is:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\beta}_{z_{n,d}})$
3. For each pair of documents d, d':
   (a) $y_{d,d'} \sim Bern(\sigma(\boldsymbol{\eta}^\top(\bar{\mathbf{z}}_d \circ \bar{\mathbf{z}}_{d'}) + \nu))$, with $\bar{\mathbf{z}}_d = \frac{1}{N_d}\sum_n z_{n,d}$ and $\sigma(x) = \frac{1}{1+e^{-x}}$



Inference Parameters of interest are topic specific word distributions $\boldsymbol{\phi}_k$, document specific topic proportions $\boldsymbol{\theta}_d$, and latent word specific topic assignments $z_{n,d}$. For inference, a variational approximation to the posterior distribution, for topic probabilities and topic assignments, minimizes the relative entropy by coordinate ascent. This is the same as original LDA, with an additional term in the evidence lower

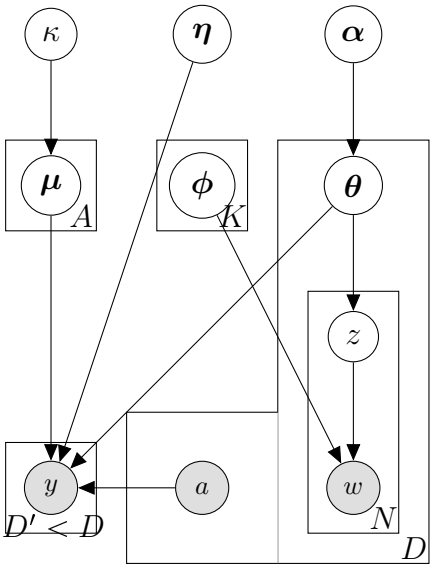
bound (ELBO) for the log probability of observed links $y_{d,d'}$ between documents. This requires an additional parameter ρ which estimates the proportion of unobserved non-links in the network. Topics $\boldsymbol{\phi}_k$ and the link regression coefficients $\boldsymbol{\eta}$ and intercept ν are estimated with maximum likelihood approximations, using variational EM.

Topic Link LDA

In the situation where documents have authors and are linked, the Topic-Link LDA model of Liu et al. (2009) enables an analysis. With a hierarchical Bayesian model, the paper simultaneously achieves two common goals of topic discovery and community detection. The authors believe that two documents with similar topics may not be linked because the authors might not be in the same community and therefore wouldn't likely know each other. Compared with the relational topic model, this model differs because link generation depends not only on topic similarity but also on community similarity and a random term. The experiments involve outgoing citation prediction.

Generative Model The data generating procedure is below:

1. For each author p:
   (a) $\mu_p \sim Dir_C(\boldsymbol{\kappa})$
2. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\phi}_{z_{n,d}})$
3. For each pair of documents d, d':
   (a) $y_{d,d'} \sim Bern(\sigma(\eta_1 \mu_{a_d}^T \mu_{a_{d'}} + \eta_2 \theta_d^T \theta_{d'} + \eta_3))$, with $\sigma(x) = \frac{1}{1+e^{-x}}$



Inference The authors use mean field variational inference to approximate the true posterior of latent variables with a factorized distribution. Computing the objective function involves computing Dirichlet integrals with the digamma function and an additional variational approximation for the expectation of the logistic function. Minimizing the KL divergence gives a closed form update for the topic variational parameter. For the latent communities and topic proportions, constrained iterative active set optimization and line searching algorithms are applied.

2.3.5 Supervised Learning of Topics

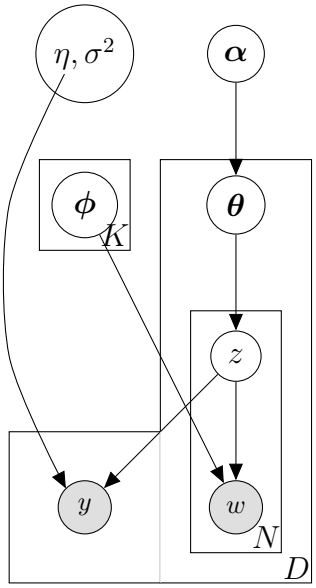
To make use of additional data, some models extend LDA by allowing each document an associated response or label. We describe two LDA-based models with associated response data.

Supervised LDA

The Supervised LDA model of Mcauliffe and Blei (2008) is the same as LDA, except with a generalized response variable for each document. The response can be real (a rating for a book), count (number of likes for a comment), or category valued (the label of a news article). A regression model predicts the response given the empirical topic proportions for that document and global coefficient and variance parameters. The joint model parameters are optimized for predicting out-of-sample responses given documents. Experiments in the paper involve predicting movie ratings from reviews and tone of US senatorial amendments from their text.

Generative Model Data are generated as follows:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each document d:
   (a) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha})$
   (b) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\beta}_{z_{n,d}})$
   (c) $y_d \sim Norm(\boldsymbol{\eta}^\top \bar{\mathbf{z}}_d, \sigma^2)$, with $\bar{\mathbf{z}}_d = \frac{1}{N_d} \sum_n z_{n,d}$



Inference The method uses posterior inference on the document topic proportions θ_d and word topic indicators z_{wn}. This posterior is analytically intractable, so a variational approximation is used. Other parameters of interest are the topic proportions' concentration parameter $\boldsymbol{\alpha}$, topics $\boldsymbol{\phi}_k$, normal linear model coefficients $\boldsymbol{\eta}$ and variance σ^2. Following LDA, these parameters are treated as unknown constants and

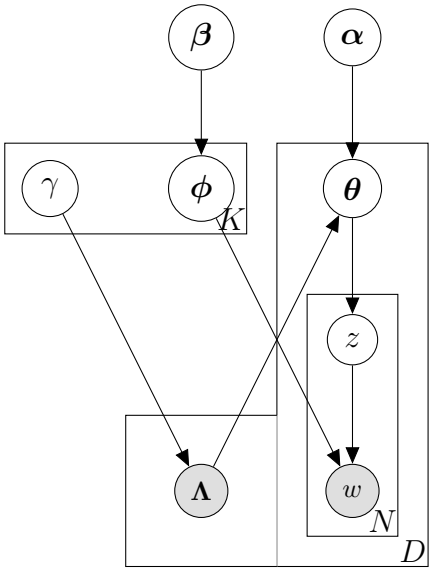
are estimated using variational EM to approximate the maximum likelihood. The objective function differs from that of LDA by the expectation of the log probability of the response given the topic assignments.

Labeled LDA

The Labeled LDA model of Ramage et al. (2009) is an approach to credit attribution in which each word of a document is associated with a label. Here, $\boldsymbol{\tau}_d$ is the matrix indicating for document d whether the lth label $\lambda_{l,d}$ is equal to topic k. This extends the usual LDA model by defining a one-to-one mapping between topics and labels. Labeled LDA contributes a method for label specific snippet extraction and multi-label classification. When compared with a support vector machine, labeled LDA performed better with multiple labels per document and worse with a single label per document. L-LDA is believed to perform better when labels are more semantically diverse.

Generative Model The generative model is:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each document d:
   (a) For each topic k:
      i. $\Lambda_{k,d} \sim Bern(\gamma_k)$
      ii. $\boldsymbol{\lambda}_d = \{k : \Lambda_{k,d} = 1\}$
      iii. For each label l:
         A. $\tau_{l,k,d} = 1(\lambda_{l,d} = k)$
   (a) $\boldsymbol{\alpha}_d = \boldsymbol{\tau}_d \times \mathbf{1}_K \boldsymbol{\alpha}$
   (b) $\boldsymbol{\theta}_d \sim Dir_K(\boldsymbol{\alpha}_d)$
   (c) For each word n:
      i. $z_{n,d} \sim Cat(\boldsymbol{\theta}_d)$
      ii. $w_{n,d} \sim Cat(\boldsymbol{\beta}_{z_{n,d}})$



Inference The inference for each document's word specific topic assignments z_{wn} is similar to that of LDA with collapsed Gibbs sampling, but with the topics confined to belong to a subset of labels. In particular, each document's word topic assignments are drawn from among its own labels. The strategy is to learn topics $\boldsymbol{\beta}$ on a training set, where labels and words are available. Gibbs sampling can then generate topic assignments z_{wn} for a new document given only labels $\boldsymbol{\lambda}_n$ and the learned parameters.

2.3.6 Other LDA-Based Topic Models

Some topic models relax the assumption of independent topics by modeling topic correlation. These models include the correlated topic model (CTM) of Lafferty and Blei (2006) and Pachinko Allocation of Li and McCallum (2006).

Other topic models can use the data to learn an appropriate number of topics. The hierarchical Dirichlet process (HDP) of Teh et al. (2005) is an example which uses a Bayesian non-parametric prior on the number of topics.

Other topic models learn hierarchies of topics. The nested Chinese restaurant process (nCRP) of Blei et al. (2010) defines a Bayesian non-parametric prior over trees of topics.

Other topic models relax the bag of words assumption by modeling word ordering. Beyond Bag of Words (Wallach, 2006) uses n-grams, Integrating Topics and Syntax of Griffiths et al. (2005) combines use of HMMs for syntax with LDA for modeling semantics.

Some other popular LDA-based topic models include the citation influence model (CIM) of Dietz et al. (2007), the author conference topic (ACT) model of Tang et al. (2008), multi-grain LDA (MGLDA) of Titov and McDonald (2008), the sparse additive generative (SAGE) model for text of Eisenstein et al. (2011), the structural topic model (STM) of Roberts et al. (2013), the group topic (GT) and groups over time (GOT) models of McCallum et al. (2007), and the multi-grain clustering topic

model (MGCTM) of Xie and Xing (2013).

2.4 A Catalog of Topic Models

We provide a summary of the above models in Table 2.2

Table 2.2: Comparing LDA-Based Topic Models

Statistical Models for Documents and Networks							
Method (Authors)	**Words**	**Topics**	**Time**	**Links**	**Communities**	**Data Sets**	**Applications**
Smoothed LDA (Blei, et. al.)	✓	✓	✗	✗	✗	TREC AP (16K news stories), C Elegans (5K scientific abstracts), Reuters-21578 (22K news stories), EachMovie (4K users' movie ratings)	doc. modeling, text class., collab. filtering
Dynamic Topic Model (Blei, et. al.)	✓	✓	✓	✗	✗	Science (30K articles, 120 years)	Predictive model in sequential corpus, qualitative summary

Method (Authors)	**Words**	**Topics**	**Time**	**Links**	**Communities**	**Data Sets**	**Applications**
Topics over Time (Wang and McCallum)	✓	✓	✓	✗	✗	McCallum's emails (13K messages, 9 months), NIPS papers (2K papers, 17 years), state-of-the-union addresses (208 transcripts (split into 6K documents), 200+ years)	Time stamp prediction, interpretable topic trends
Author-Topic Model (Rosen-Zvi, et al.)	✓	✓	✗	✗	✗	NIPS papers (2K papers, 2K authors), CiteSeer (162K abstracts, 85K authors)	author-author topic similarities

Method (Authors)	**Words**	**Topics**	**Time**	**Links**	**Communities**	**Data Sets**	**Applications**
Relational Topic Model (Chang et al.)	✓	✓	✗	✓	✗	Cora research paper search engine (3K abstracts and 5K citations), WebKB CompSci (877 web pages and 1K hyperlinks), PNAS (2K abstracts and 2K citations)	test document link prediction given words and vice versa
Supervised LDA (Blei, et al.)	✓	✓	✗	✗	✗	movie reviews and ratings (5K), Digg web pages (4K descriptions and likes)	predicting a rating or label given text
Labeled LDA (Ramage et al.)	✓	✓	✗	✗	✗	del.icio.us (4K web pages and tags) and Yahoo (8 data sets, each with multiple categories)	credit attribution, label specific snippet generation, multi-label classification

Method (Authors)	**Words**	**Topics**	**Time**	**Links**	**Communities**	**Data Sets**	**Applications**
Link LDA (Erosheva et al.)	✓	✓	✗	✓	✓	PNAS biology (12K abstracts and 77K unique citations)	divides words and references into topics, soft classifies documents into topics, predicts outgoing links for new documents
Topic Link LDA (Liu et al.)	✓	✓	✗	✓	✓	web 2.0 blogs (2K technology posts, 75 blogs), political blogs (5K posts, 101 blogs), CORA (3K papers, 423 authors, citations)	topic and community inference, predicts links between document pairs
Pairwise Link LDA (Nallapati et al.)	✓	✓	✗	✓	✗	CiteSeer (1K abstracts, citations), Nielson Buzzmetrics blogs (4K post)	predicts incoming and outgoing links for a new document, visualizations of topic pair specific link probability

Method (Authors)	Words	Topics	Time	Links	Communities	Data Sets	Applications
Link PLSA LDA (Nallapati et al.)	✓	✓	✗	✓	✓	CiteSeer (1K abstracts, citations), Nielson Buzzmetrics blogs (4K post)	predicts outgoing links for a new document, infers top documents for a topic, topic popularity
ART (McCallum et al.)	✓	✓	✗	✓	✓ (implicit)	Enron emails (23K messages, 147 users) and McCallum emails (14K messages, 825 users)	predicts outgoing links for a new document, infers prominent relations for each topic, identifies roles as distributions over authors

2.5 Future Directions

With the prevalence of text based social network data (ie. tweets, blog posts, or emails), a natural extension is a dynamic model which jointly models text and a network. One paper (Wang et al., 2011a) describes a combined models for text, links, and time stamps.

2.5.1 Dynamic Network Topic Models

There are a few reasonable characteristics which can be included in such a model. First, community detection among authors can be guided by topical content along with linking behavior. Second, documents can have increased probability of sharing topical content when they share community membership or there is a link between them. This time the topics themselves are conditioned on link information, perhaps through sampling a portion of the document topic scores from a distribution centered on the linking document's topic scores. Last, the occurrence of events can trigger new topics to come into existence. The idea is that, while topics like health or education may be talked about consistently through time, others like the Trayvon Martin shooting may be introduced suddenly after the occurrence of an event. To our knowledge, this is a gap in LDA-based topic modeling literature.

2.6 Conclusion

This review includes a catalog of LDA-based topic models, intended to help an analyst choose an appropriate model for his or her application. I provide a common notation and graphical model for 12 methods, to clearly emphasize their similarities and differences. Discussion includes proposed method of inference for each model, whether a variational approximation, MCMC, or an other algorithm. I mention the data sets each method is applied to, along with a description of the size of the data and the variables which are modeled. Recommended doable tasks, like learning topics and communities, are stated.

3

A Dynamic Text Network Model for Political Blog Posts

3.1 Comments About the Dynamic Blog Post Network Paper

Work presented in this chapter is drawn from Henry et al. (2016) which is submitted to the Journal of Classification. The work is included in this dissertation, because it exemplifies an application for a probabilistic model for text data in a social network. My contributions were pre-processing the data, developing the formalization of the generative model, deriving full conditional posterior distributions for Gibbs sampling, and re-wording and simplifying the introduction, literature review, discussion and presentation of results, and conclusion. Coauthor contributions are detailed in the acknowledgements.

In the paper we develop a novel Bayesian probabilistic model for text data, in which each observation is associated with a node of a network, a time stamp, and, in some cases, links to other nodes. We make joint inference about dynamic topics, communities of blogs, and the presence of events. To our knowledge there is no other existing literature which simultaneously models these three variables. Topic

assignment is guided by the text of the document and whether a related event has occurred, and community assignment is guided by the links between blogs and topics in which the blog is interested. We analyze a corpus of networked political blog posts from 2012. We discover 22 appropriate topics, which vary in time and reveal that token usage increases following a related event. We also find communities of blogs with reasonable sets of topic interests and one community interested in sensational crime, which forms a community despite rarely linking with each other.

3.2 Motivating a Dynamic Text Network Analysis

Since the emergence of the Internet, dynamic text networks have become a common form of data. Examples include articles and their links on Wikipedia (Hoffman et al., 2010), academic articles and their citations (Moody, 2004), and hyperlinked blog posts (Latouche et al., 2011). Idiosyncrasies in the data generating procedure of each application prompt modifications to existing literature which lead to specialized models. Therefore, despite the existence of methodology for dynamic text networks (Wang et al., 2011b), it is reasonable to develop new models for specific applications of dynamic text networks.

We develop a model to analyze a data set of the words from documents and associated hyperlinks from 467 U.S. political blogs in 2012. Also available in the data are a time stamp giving the date of the blog post and a URL for the blog website. Our main motivations are to summarize evolving topical contents of blog posts, discover communities of blogs with similar text and links, and identify political events. Methodologically, our approach is to make use of text and event information to guide the learned topic summary and to then use blogs' links and topic interests to inform community discovery.

Our approach borrows vocabulary and techniques primarily from *topic modeling* and *network modeling*. We elaborate now on each of these classes of models.

In topic modeling, we begin with a *corpus*, or collection, of *documents*, here, the blog posts. In *bag-of-words* models (Harris, 1954), each document is defined as a set of *tokens* and the number of occurrences of each. A token is either a word or an *n-gram*, which is a phrase consisting of n words which occur in order relatively often. One common 3-gram is "President of the United States". This is not considered a 5-gram, because in practice the words "of" and "the" are both commonly considered *stop-words* and are removed prior to analysis, so only "President United States" remains. Stop-words occur frequently in many topics, and therefore don't distinguish topics as well as other tokens. The unobserved groups to which a document is assigned are called *topics*. Examples in the political blog posts are "election", "economy", and "sensational crime". Topics guide the occurrence of tokens in a document, and a topic is mathematically defined as a probability distribution over the set of unique tokens occurring in the corpus. We clarify, topics don't usually come with a name. Rather, an analyst names them based on, for example, the high probability tokens. An "election" topic, then, may assign relatively high probability to tokens like "vote" and "candidate", and an "economy" topic to words like "income" and "revenue".

Much of probabilistic topic modeling is built around a foundational model called latent Dirichlet allocation (LDA) from the paper Blei et al. (2003a). The approach is to posit the data are theoretically generated by the following process. First the topics are drawn from a Dirichlet distribution. Each topic is a distribution over unique tokens. Next, from another Dirichlet distribution, each document gets a distribution over topics, which summarizes its contents. For example, a document might be 70% about the "election" topic, 25% about the "economy" topic, and 5% about the remaining topics. Next, is the token generation. First a topic assignment is drawn from a multinomial distribution with probability parameter equal to the document specific topic proportions. For example, we'll assume we sample the "election" topic. Then a word is drawn from the "election" topic. Perhaps it is "campaign". This

process repeats until all of the tokens of the document are generated. The strategy for deriving estimates for model parameters can be understood as reversing the data generating process. A primary goal of inference is estimating the latent topic assignments for each token of each document. Computing summary statistics of these assignments enables estimation of the two major model parameters, document specific topic proportions and topic specific token proportions. Together these two parameters summarize the corpus. One assumption which simplifies inference is that each document is about a single topic. An example of this approach is described in Yin and Wang (2014), which we follow in this paper.

Some topic models are dynamic, since they allow topic specific token distributions to vary over time. One paper uses same generative model as LDA, though it assumes the concentration parameters for the topic proportions Dirichlet distribution are time specific (Blei and Lafferty, 2006). In more detail, one day's parameter is related to those of the past day through a normal, linear autoregressive process with lag 1 ($AR(1)$). Similarly, the topic specific word distributions are time specific, and are related to those from the previous time through an $AR(1)$ process. We use a similar idea in our paper to extend Yin and Wang (2014) to have dynamic topics. However, rather than using an $AR(1)$ process with a Markov dependence, we allow dependence on multiple time units from the past. A further innovation to the topic dynamics is inspired by a characteristic of the political blog data. That is, the presence of major news events like the announcement of Paul Ryan as Mitt Romney's vice presidential candidate. To accommodate events like this, we introduce daily topic specific event indicators which boost the rate of posting about a topic at the indicated times. These effectively increase the probability of documents being assigned to a topic during an active related event. In this example, there likely was a temporary increase in post rate about topics related to the election or the republican party. Furthermore, this clarifies the importance of allowing for time evolving topics, because while the

previous probability of using a token like "Paul Ryan" may have been relatively small, it likely increased after his vice presidential candidacy was announced.

Next I describe network modeling. Typically there is a set of *nodes*, which in our case are blogs. When two nodes interact in a specified way, there is a *link* between them. Links can be either *directed* or *undirected*. In our case, they are directed, and a link forms when blog i includes a hyperlink to blog j in a post. Links are often stored in an *adjacency matrix*, where often entry i, j is 1 if there is a link from i to j and zero otherwise. Some approaches use a *weighted* adjacency matrix, with entries taking on values besides 0 and 1, though we choose only to model the presence or absence of a link. One common approach for network modeling is exponential random graph models (ERGM) which are described in (Holland and Leinhardt, 1981; Wasserman and Pattison, 1996). The strategy is to model the presence of each link with a logistic regression, with predictors including node and link covariates. Important in our application to the political blog posts are node covariates. For example, the popularity of the blog is likely to impact link probabilities. The model is extended first in Robins et al. (2001) to allow for dependence in graphs across adjacent time steps. Later, Hanneke et al. (2010) extends the ERGM by providing an exponential family distribution for the evolution of graphs in discrete time, also calling it a discrete temporal ERGM, or TERGM. Another time dependent extension is in Krivitsky and Handcock (2014), which contributes a separable temporal ERGM, or STERGM. The key innovation is to model link formation and duration separately. In the blogs, this allows us to account for things like a blog's popularity and whether the blogs have previously linked with each other. This framework is flexible and can be used in combination with clustering methods on nodes to perform *community detection*.

There are a variety of recent approaches to community detection. One way of defining a community is a group of nodes which is more likely to link to each other

than to other nodes. In fact, one quantification of the communities discovered is called *modularity*. Generally speaking, modularity is the difference between the observed proportion of edges falling within community and the expected proportion under random edge formation. Larger modularity means more distinguishable communities. Approaches can be categorized as non-parametric and parametric. For non-parametric options, a number of papers use modularity optimization algorithms, notably Newman and Girvan (2004) which uses a spectral technique. Parametric options include the foundational stochastic block model (SBM) of Snijders and Nowicki (1997). This approach assigns nodes into latent blocks (or communities) of nodes, which have block-pair specific linking probabilities. One extension is to incorporate other nodal covariates as in Faust and Wasserman (1992). Another group of network models called latent space models are described in Hoff et al. (2002). These models assign each node a position in a latent space, based on node covariates, and groups nodes into communities based according to their proximity in the latent space.

Our network model combines aspects of the ERGM framework, SBM, and a latent space model, because the log odds of a link is regressed on some node specific predictors, gets an additive effect for blogs being in the same block, and depends on a quantification of the similarity of the blogs' latent features, which are topic interests.

Other work combines topic modeling with network modeling. Extending LDA is the relational topic model of Chang and Blei (2009b), which models the probability of a link between documents as a function of element-wise products of their respective topic proportions. Our model is similar, though modeling links between nodes, rather than between documents, and offering an additional boost in link probability when the two nodes are in the same community. Another approach in Ho et al. (2012a) claims to find hierarchical groups of documents with "similar word distributions and dense network connections". Third, the model in Wang et al. (2011b) brings joint text and network modeling to the setting of a dynamic social networks. The model

is most similar to ours in that it has dynamic topics and a regression for links. Our model differs from all of the above models, because it simultaneously learns communities of nodes and dynamic topics in the documents. We emphasize our model is for time stamped links, so the network model too is dynamic.

To summarize, we develop a novel Bayesian model for dynamic text network data. The model achieves two primary goals. First, it discovers topics whose word probabilities evolve in time and whose prevalence accommodates for the presence of related events. Second, it learns communities of blogs based on time stamped links, similarity of topical interests, and node covariates. We achieve these goals by combining a novel dynamic extension of Dirichlet mixture modeling (Yin and Wang, 2014) with a stochastic block model (Snijders and Nowicki, 1997) with partial co-block membership defined by topic interests. We use the model to jointly analyze the dynamic network of interactions and associated documents among 467 top U.S. political blogs from 2012.

The remainder of this chapter is structured as follows. First I describe our data set, from where we got it to pre-processing it for analysis. I then describe the observed data, including the text and the links, and the parameters we estimate, such as the topics and communities. Next I propose a generative Bayesian model which relates parameters to observed data. After, I describe the probabilistic inversion of the generative model, through Markov chain Monte Carlo, which is used to infer model parameters. Last I present several findings from our analysis of the political blog data, finishing with a summary discussion and possible extensions.

3.3 Political Blogs of 2012

In the following sections, we discuss the data to which we apply our model.

3.3.1 Acquiring the Political Blog Posts

We apply our model to a data set of blog posts during 2012 from 467 top political blogs. The blogs were ranked by Technorati (2012), a publisher advertising platform. The data were scraped (following `robots.txt` protocols) from the blogs by Andrew Cron of MaxPoint Interactive, now Valassis Digital (Valassis Digital, 2012). The scraped text was stemmed, using a modified version of Snowball (McNamee and Mayfield, 2003) at MaxPoint. This replaces words like "writing" and "writes" with their root word "write". We wanted to retain three-letter acronyms like EPA and NSA which required a modification. Thanks to people at both companies for their work in providing the data.

3.3.2 Suitability for Dynamic Topic Modeling and Community Detection

The corpus is reasonably considered a dynamic text network, because along with its text, each blog post has a time stamp, a node of a network, and sometimes links to other nodes. We recall that the time stamps are dates in 2012, nodes are blog websites, and links are hyperlinks to other blogs. The data are suitable for an event guided dynamic topic model. For example, in a topic related to sensational crime, following the shooting of Trayvon Martin, a news event in 2012, the token "Zimmerman" which names the shooter, increases rapidly. It's also reasonable to hypothesize the existence of blog communities in this data, at least divided along political affiliations, and possibly with liberals more interested in a topic like gay marriage and conservatives in religion.

3.3.3 Pre-Processing the Data

The data set as I received it consisted of 111615 blog posts from 366 days of 2012. There was a very large vocabulary including many words which appeared to be typos and only occurred once. To reduce the vocabulary size to expedite computation, we

took steps to filter unwanted tokens.

Vocabulary Filtering

One approach was to use unweighted TF-IDF variance thresholding (Ramos, 2003). The TF-IDF score for token w in blog post d is

$$\text{TF-IDF}_{wd} = f_{wd}/n_w \tag{3.1}$$

,

where f_{wd} is the number of occurrences of token w in blog post d, and n_w is the number of posts that use token w. Examples of words with low variance TF-IDF are "therefore" and "because", which are relatively common across all posts and therefore all topics. The tokens "basketball" and "soccer" are more likely high variance TF-IDF, because they occur a lot in posts about a "sports" topic but rarely in other posts. We set the threshold high enough to retain tokens like "Obama", which surprisingly is a relatively low variance TF-IDF token, because it occurs in many posts. We note that variations of TF-IDF exist which emphasize keywords with slightly differing properties. For example, term-frequency inverse-time-frequency (TF-ITF) is the same as TF-IDF, but the denominator counts the number of days on which the word was used. This is useful to identify terms that are associated with events, because a token like "Benghazi" (a location in Libya that gained news attention following an attack) might occur on a very small number of days and therefore have a high TF-ITF. Others variations natural log transform either of the numerator or the denominator of the equation to de-emphasize either the term frequency or inverse document frequency, respectively.

Next, we removed rare tokens that were mentioned in less than 0.02% of the posts. These tokens are unlikely to have large impact on the topic token distribution across all posts. This reduced the number of unique tokens and therefore

the computation time for topic word distributions. Some of these tokens were misspellings (e.g. "Merkle" for "Merkel", the Chancellor of Germany), and some were incomprehensible to us.

Finding n-grams

One approach to improve interpretability of topic models is to find significant n-grams. We begun by identifying bi-grams. The strategy was to compare the observed bi-gram count to that expected to occur at random. To compute the expected count of a bi-gram we used the marginal uni-gram counts and an independence model. For example, the approximation we used for the expected count of the phrase "white house" was $Np_{white}p_{house}$ where $p_{word} = \frac{\text{count of word}}{\text{total number of words}}$. It is possible to then obtain an empirical distribution of differences. We initially selected as significant bi-grams with differences in the largest 5%. This left a large number of bi-grams, many of which we didn't consider meaningful phrases, so we used frequency thresholding to remove any which occurred fewer than 500 times. This retained about 70% of the more common bi-grams. Next, for the selected bi-grams, we replaced all occurrences in the corpus with a single token which encodes the bi-gram (e.g. "white" "house" is replaced with "white.house"). We then repeated the bi-gram identification procedure, this time the results included tri-gram and 4-gram candidates. We selected those which occurred at least 100 times in the corpus. The threshold was chosen to include phrases like "voter registration form". We didn't think of any 5-grams or more and stopped the general n-gramming procedure here. However, we did identify significant 20+-grams to remove phrases which occurred many times and diluted the results. For example, one blogger included the second amendment at the end of many posts. I believe this over emphasized the topic and detracted from other findings, so I removed all but the first occurrence. There is a sophisticated literature on n-gramming, and more detail on strategies can be found in Brown et al. (1992).

After pre-processing, the total vocabulary consisted of 7987 unique tokens. This vocabulary filtering left some posts with very little content remaining. After removing short posts, the resulting data set which we analyzed has 109,055 posts.

3.3.4 Identifying Quantities of Interest

With preprocessing complete, we now describe the relevant observed variables of the dataset and the theoretical quantities of interest which become model parameters. Thanks to the work done in Henry et al. (2016), submitted for publication in the Journal of Classification and under revision, there is an existing description of these quantities. We rely on the previous description, and the remainder of this subsection is verbatim from the paper:

- **Observed Data**
  - **Posts** are individual blog posts, along with relevant covariates. In particular, each post consists of the document text, along with the date the post was published, the blog the post was published on, and links to other blogs in the network. For modeling, the text from each document is reduced to counts of each unique token, using the well known bag-of-words model. Our model assumes that each post is about a single topic.
  - **Blog** is the web domain on which a post is written. Examples include *thinkprogress.org*, *politicalticker.blogs.cnn.com*, and *jonathanturley.org*. A blog is comprised of posts written by one or more authors. The data set doesn't have author information, so we model posts as occurring on blogs.
  - **Links** are hyperlinks found in the text of a post. Roughly half of all posts contain links to other blogs in the network. To clarify, the links in the data are to blogs (i.e. the web domains like thinkprogress.org), not to specific blog posts. We choose to aggregate links into day specific

adjacency matrices, each of which encodes whether blog i links to blog j at time t.

  - **Tokens** are the items which comprise the textual contents of a post. These include single words like "economy" as well as discovered n-grams like "health care". If a word exceeds inclusion thresholds as both a uni-gram and a bi-gram (i.e. "white" and "white house"), then both are included as tokens. For each post, we consider the text as a "bag-of-words" count vector, where the wth entry gives the number of occurrences of token w in the post.

- **Inferred Parameters**

  - **Topics** are summaries of the content in the text data, probabilistically described as time evolving distributions over the selected vocabulary. That is, at each time, a particular topic assigns probabilities (which sum to 1) to each unique, allowable token. For example, we might call a topic the "Sensational Crime" topic if it assigns high probabilities to tokens like "Trayvon Martin", "George Zimmerman", and "shoot".

  - **Post Topic Assignment** is the group into which a post is assigned, based on its textual content and linkage information about the blog on which it was written.

  - **Events** are things like the Iowa caucus or the state of the union address, which help describe differences in posting rates that occur over time. Here, we conceptualize events as being topic specific, and when an event is occurring, there will be an increased rate of posting on the associated topic. For example, if an event tied to the general election occurs, posts on the general election will increase. It is important to note that these

"events" are unobserved in our dataset, and we attempt to estimate their occurrence.

- **Event Caused Post Rate Boosts** are a quantification of the increases in post rate due to an event, which are specific to a particular topic.
- **Communities** are groups of blogs, defined by unique sets of topic interests. Each blog is assigned a single community. One community may have interest, for example, exclusively in the "presidential election" topic, another community in the "sensational crime" and "defense" topics, and a third community in all topics. Each community also has its own linking probabilities, such that blogs within the same community tend to link to one another more than to blogs in different communities. A blog's probability of posting on a specific topic depends on the blog's community membership. For example, if a community is interested in taxation, sensational crime, and the Senate, blogs assigned to this community will tend to post mainly on those three topics. Communities link our textual information with our network information by relating topic interests with link probability.
- **Blog Community Membership** is a blog's assignment into a particular community, based on linkage pattern and the topic assignments of each of the blog's posts.
- **Blog Interest Proportions** are the relative weights (summing to 1) of interest in each topic for a blog. These tend to be highest for topics which characterize the community to which the blog is assigned, though they allow for interest in all topics.
- **Blog Post Rate** is the baseline rate at which posts are written on a blog when there are no events happening.

– **Link Formation Parameters** are the effect sizes for various factors believed to govern the probability of linkage. These include community membership overlap, lagged reciprocity, in-degree, out-degree, and a baseline link rate.

With our quantities of interest described at a conceptual level, we recall that the model is motivated by two main goals. The first is to discover a set of topics that change over time. The second is to classify blogs into interest communities, such that each community is characterized by a set of topics and specific linking tendency. With these goals in mind, we propose a generative model which links observed and estimated quantities together in a substantively feasible and probabilistically tractable way.

3.4 Model

I note the contents of this section are taken, almost verbatim from Henry et al. (2016). To achieve the aforementioned goals, we utilize Bayesian modeling and inference methods. As is routine in Bayesian statistics, we first specify a hypothetical model from which the data may feasibly have been generated. This hypothetical model makes no real distinction between observed and to-be-estimated quantities, and therefore operates in a position of perfect knowledge, where, for example, topic and community distributions are known. We parameterize this model such that quantities of interest, specifically the topics and communities, govern data generation. With a probabilistic model $p(\mathbf{X}|\mathbf{\Theta})$, for data $\mathbf{X}$ given parameters $\mathbf{\Theta}$, specified, we then mathematically reverse the process using Bayes theorem to infer the posterior distribution $p(\mathbf{\Theta}|\mathbf{X})$ of model parameters given the observed data.

3.4.1 Generative Model Overview

We rely on the terminology from Section 2 to describe the generative model in a few related subsections. Here in the subsection 1 we briefly summarize the involved steps: subsection 2 is for generating K topics, which are distributions over tokens and allowed to change over time; subsection 3 is for the generation of events relating to each topic; in subsection 4, B topic-interest communities are created, and each blog is assigned to one; in subsection 5, for each blog, on each day, the number of posts on each topic is generated; in subsection 6, the words of each post are generated from the appropriate day specific topic; and finally, links between blogs are generated according to their communities and other covariates. To any non-statistician reader, while this is termed a *generative* model, it is purely a theoretical tool, and no data are being created. Instead, the model is described to posit the hypothesized structure of our empirical data set. This structure is mirrored in the inference section. We now describe mathematical and probabilistic details of each step in the generative model.

3.4.2 Dynamic Topic Generation

To begin, we propose a generative model for each topic, which is a probability distribution over unique tokens. As topics here are thought to be dynamic, we allow for the probabilities of each token to change over time. For each topic k, on a specified day t, we assume the token probabilities $\mathbf{V}_{kt}$ are drawn from a Dirichlet distribution prior. This allows for some day specific idiosyncrasies in topic specific word probabilities. However, to encourage some continuity across days, we calculate the average of topic k's topic distribution $\mathbf{V}_{k(t-1):(t-\ell)}$ from the previous ℓ days and use it as the concentration parameter for the Dirichlet distribution from which the present day's topic $\mathbf{V}_{kt}$ is drawn. On the first day, each topic $\mathbf{V}_{k1}$ for k in $1:K$ are initialized from a $Dir_W(\mathbf{a}_{k0} = \mathbf{1})$. The sampling then proceeds in sequence, first calculating each

topic's concentration parameter as in equation 3.2 and then sampling each topic as in 3.3 and then moving to the next day. This procedure repeats for times $t = 1 : T$. The topics are then distributed:

$$\boldsymbol{a}_{kt} = \frac{1}{\ell} \sum_{t'=1}^{\ell} \mathbf{V}_{k(t-t')}, \tag{3.2}$$

$$\mathbf{V}_{kt} \sim Dir_W(\mathbf{a}_{kt}). \tag{3.3}$$

3.4.3 Topic Specific Event Generation

To reflect the event driven nature of blogs posting on various topics, we generate events, each of which then boosts the post rate of a specific topic. For each topic k, at each time t there is some probability η_k of an event occurring. One can choose $\eta_k = .01$ for all k, which suggests each topic has on average 1 event every 100 days. Alternatively, different topics can have different daily event probabilities or one can specify a prior distribution for η_k. Given η_k, the daily, topic-specific event indicators are sampled as:

$$E_{kt} \sim Bern(\eta_k). \tag{3.4}$$

Here, $Bern(p)$ is a Bernoulli distribution with probability parameter p, giving the probability of an event. The expectation is p and variance is $1 - p$. When an event happens on topic k, blogs with interest in topic k have their posting rates increase by a factor determined by ψ_k. Speculating that some topics have events which are much more influential than others, we let this multiplier be topic specific:

$$\psi_k \sim Gam(a_\psi, b_\psi). \tag{3.5}$$

Similarly, $Gam(a, b)$ is a gamma distribution with shape parameter a an rate parameter b, having expectation a/b and variance a/b^2.

3.4.4 Community and Blog Specific Topic Interests Generation

With our topic distributions and topic specific events generated, we can now assign blogs to topic interest communities. We begin by defining the community-specific topic-interests matrix $\mathbf{I}$, where each column b indicates which of the k topics are of interest to community b. The first $\binom{K}{1}$ columns correspond to the singleton communities, which are interested only in topic 1, topic 2, up through topic K, respectively. The next $\binom{K}{2}$ columns define doublet communities, which have interest in all of the possible topic pairs. The next $\binom{K}{3}$ columns correspond to communities which have interest in exactly 3 topics, and the final column is for the community which has interest in all K topics:

$$I_{kb} = \left[\begin{array}{ccccc:ccccc:ccccc:c} 1 & 0 & 0 & \dots & 0 & 1 & 1 & 1 & \dots & 0 & 1 & 1 & 1 & \dots & 0 & 1 \\ 0 & 1 & 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 & 1 & 1 & 1 & \dots & 0 & 1 \\ 0 & 0 & 1 & \dots & 0 & 0 & 1 & 0 & \dots & 0 & 1 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & \dots & 0 & 0 & 1 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 & \dots & 1 & 1 \end{array}\right] \tag{3.6}$$

To assign blogs to communities, we sample their membership with a single draw from a multinomial distribution. This means each blog is a member of only a single community, characterized by the topic interests in the above matrix. Each community assignment is then drawn from a multinomial distribution:

$$b_i \sim Mult(1, \mathbf{p}_B). \tag{3.7}$$

Here, $Mult(\mathbf{p})$ is a multinomial distribution with probability parameter vector $\mathbf{p}$, giving the probabilities for each category. One can choose the probabilities of belonging to each community uniformly, by setting each element of $p_b = 1/B$ where B is the number of communities. Another approach is to partition the probabilities vector into the singlets, doublets, triplets, and all-topics communities, and allocate probability uniformly to each of these categories, and then uniformly divide up the probability among communities within each category:

$$\mathbf{p}_B = \begin{pmatrix} \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \\ p_K \end{pmatrix}, \text{ with } \mathbf{p}_1 = \begin{pmatrix} p_{1,1} \\ p_{1,2} \\ \vdots \\ p_{1,\binom{K}{1}} \end{pmatrix}, \mathbf{p}_2 = \begin{pmatrix} p_{2,1} \\ p_{2,2} \\ \vdots \\ p_{2,\binom{K}{2}} \end{pmatrix}, \mathbf{p}_3 = \begin{pmatrix} p_{3,1} \\ p_{3,2} \\ \vdots \\ p_{3,\binom{K}{3}} \end{pmatrix}, p_K = p_{K,\binom{K}{K}}. \tag{3.8}$$

For notational convenience throughout the rest of the paper, we define $\mathbf{B}_i$ to be the set of topics which are of interest to blog i:

$$\mathbf{B}_i = \{k : I_{kb_i} = 1\}. \tag{3.9}$$

With each blog's topic interest indicators known, we can generate blog-specific topic-interest proportions. For example, two blogs may be in the community with interest in topic 1 and topic 2, but one may have interest proportions $(.9, .1)$ while the other has $(.5, .5)$. As is conventional in topic modeling, topic (interest) proportions are drawn from a Dirichlet distribution, though we make the distinction that each blog has a specific set of hyperparameters $\boldsymbol{\alpha}_i$. An individual topic interest vector $\boldsymbol{\pi}_i$ is then a draw from a Dirichlet distribution:

$$\boldsymbol{\pi}_i \sim Dir_K(\boldsymbol{\alpha}_i). \tag{3.10}$$

The hyperparameters are chosen such that a blog with interest in topics 1 and 2 is likely to have most of its interest in those topics, though it allows for interest in other topics to occur with small probabilities:

$$\boldsymbol{\alpha}_i = \begin{pmatrix} \alpha_{i1} \\ \alpha_{i2} \\ \vdots \\ \alpha_{iK} \end{pmatrix}, \text{ with } \alpha_{ik} = P1(k \in \mathbf{B}_i) + 1(k \notin \mathbf{B}_i). \tag{3.11}$$

3.4.5 Blog Specific Post Generation

Given a blog's topic interests and community membership, along with event indicators and magnitudes, we can now generate the number of posts it produces on a particular topic. Each blog may post on multiple topics, but each post is associated with a single topic. Every blog has a baseline posting rate which characterizes how active it generally is on days without events. For blog i the baseline post rate ρ_i is sampled from the following distribution:

$$\rho_i \sim Gam(a_\rho, b_\rho). \tag{3.12}$$

With the blog specific baseline post rate ρ_i, the blog specific topic interest proportions π_{ik}, the topic specific daily event indicators E_{kt}, and topic specific post rate multipliers ψ_k accounted for, we construct the expected post rate for each topic, on each blog, each day:

$$\lambda_{tki} = \rho_i \pi_{ik} + \rho_i E_{tk} \psi_k. \tag{3.13}$$

Given this post rate, the count D_{tki} of posts about topic k, on blog i, on day t are generated:

$$D_{tki} \sim Pois(\lambda_{tki}). \tag{3.14}$$

In the observed data, we don't know the post counts D_{tki} on each topic, but instead we know the marginal counts D_{ti}. These are referenced throughout the inference procedure described in section 3.5 and are calculated:

$$D_{ti} = \sum_{k=1}^{K} D_{tki}. \tag{3.15}$$

3.4.6 Text Generation

With daily topic specific post counts and token probabilities available, the posts can be populated with tokens. We first sample a total number of tokens for each post. In particular, on day t, the token count W_{tkid} for post d about topic k on blog i is sampled:

$$W_{tkid} \sim Pois(\lambda_D). \tag{3.16}$$

Where λ_D is the average number of tokens over all posts. The W_{tkid} tokens can then be sampled from the appropriate day and topic specific multinomial distribution with probability vector $\mathbf{V}_{kt}$. This is done for all of the posts in the corpus like so:

$$N^{w}_{tkid} \sim Mult(W_{tkid}, \mathbf{V}_{kt}). \tag{3.17}$$

3.4.7 Network Generation

Finally, we generate the network of links between blogs. Rather than modeling link generation at a post level, we model it at a daily blog to blog level. Specifically, we

model a directed adjacency matrix A_t of links, with entry $a_{ii't}$ indicating whether any posts from blog i have links to blog i' on day t. The binary logistic regression is suitable for this scenario. We assume the link probability $p_{ii't} = p(A_{ii't} = 1)$ depends on the following factors:

- $B(i, i') = 1(b_i = b_{i'}) + \boldsymbol{\pi}_i^T \boldsymbol{\pi}_{i'} 1(b_i \neq b_{i'})$ is the similarity (in topic interests) for nodes i and i', and is in the interval [0,1], taking value 1 if and only if blogs i and i' are in the same community.

- $L_{i'it} = 1((\sum_{t'=t-7}^{t-1} a_{i'it'}) > 0)$ indicates if blog i' has linked to blog i within the last week (previous to the current time t).

- $I_{i't} = \frac{1}{t-1} \sum_{t'=1}^{t-1} \sum_i a_{ii't'}$ is the average indegree (through time t-1) of the receiving node i'.

- $O_{it} = \frac{1}{t-1} \sum_{t'=1}^{t-1} \sum_{i'} a_{ii't'}$ is the average outdegree (through time t-1) of the sending node i.

The first covariate is sampled and constructed in equations 3.6-3.11, and the other three covariates are defined and calculated as statistics of the past data $\{A_{t'}\}_{t'=1}^{t-1}$. Together with an intercept, they comprise the regressors in a logistic regression for links, which can be written as in Equation 3.18,

$$\log(\frac{p_{ii't}}{1 - p_{ii't}}) = \theta_0 + \theta_1 B(i, i') + \theta_2 L_{i'it} + \theta_3 I_{i't} + \theta_4 O_{it}. \tag{3.18}$$

We specify a normal prior for the intercept and the regression coefficients:

$$\theta_p \sim Norm(\mu_\theta, \sigma_\theta^2). \tag{3.19}$$

We can use the logistic function to write the probability of a link as:

$$p_{ii't} = p(A_{ii't} = 1) = \frac{\exp(\boldsymbol{\theta}^T \mathbf{S}_{ii't})}{1 + \exp(\boldsymbol{\theta}^T \mathbf{S}_{ii't})}, \tag{3.20}$$

with coefficients and covariates written as:

$$\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3, \theta_4)^T \text{ and } \mathbf{S}_{ii't} = (1, B(i, i'), L_{i'it}, I_{i't}, O_{it})^T. \tag{3.21}$$

This form makes it more clear how the model could be cast within the exponential random graph model framework (Holland and Leinhardt, 1981), (Frank and Strauss, 1986). However, some of the covariates are time dependent. Similar models can be expressed in a temporal exponential random graph model (TERGM) (Krivitsky and Handcock, 2014). We note, however, that ERGMs generally include structural graph components, while Equation 3.18 does not and may be more simply understood as a log linear model. We also note that covariates depend only on past linking data, which makes this a predictive model of links. Finally, we sample each link as a single Bernoulli trial with the appropriate probability as defined in Equation 3.20:

$$A_{ii't} \sim Bern(p_{ii't}). \tag{3.22}$$

This generative model for the links can be thought of as a variant of stochastic block modeling (Snijders and Nowicki, 1997), where community membership permits partial co-block memberships. In our model, while members of the same community will have the highest probability of linking with other members of the same community, individuals who share similar topic interests, but are not in the same community are more likely to link than individuals who share no topic interests. This allows for linkage patterns that more accurately reflect the empirical phenomena of topic based blog hyperlinks.

At the end of data generation we have $W \times K \times T$ array $\mathbf{V}$ of daily topic specific token probabilities; the topic-assignment z_d for each post d; $K \times T$ matrix $\mathbf{E}$ with

daily topic specific event indicators; $\boldsymbol{\psi}$ of K topic specific event causted post rate increases; $K \times B$ community specific interests matrix $\mathbf{I}$; b_i, $\mathbf{B}_i$, $\boldsymbol{\pi}_i$, and ρ_i giving the community assignment, topic interest set, topic interest proportions, and baseline posting rate, respectively, for each blog i; $K \times I \times T$ array $\mathbf{D}$ with entry D_{kit} giving the number of posts about topic k on blog i at time t; multidimensional object $\mathbf{N}$ containing the count N^w_{tkid} for each token w in the dth post about topic k on blog i at time t; the link regression parameters $\boldsymbol{\theta}$, and the $I \times I$ daily link matrices $\{\mathbf{A}\}_{t=1}^{T}$.

With a theoretically justified data generating mechanism in place, we proceed to section 4 to "reverse the generative model" and derive posterior inference for the parameters of interest.

3.5 Parameter Inference and Estimation

The observed dataset of blog posts consists of the text as bags of words count vectors, the time stamps, the labels of which blog is posting, and the posts' links to other blogs. Therefore, as suggested in Section 3.4, we want to estimate the parameters which relate to the observed data through the generative model. This section details the specifics of how we estimate these quantities of interest from given the empirical dataset.

3.5.1 Joint Distribution of Data and Parameters

As is typical in Bayesian inference, to derive the estimation procedure, we begin by specifying the joint distribution for data and parameters according to the generative model.

$$
\begin{aligned}
P(\mathbf{E}, \boldsymbol{\psi}, \mathbf{B}, \boldsymbol{\pi}, \boldsymbol{\rho}, \mathbf{D}, \mathbf{V}, \mathbf{W}, \mathbf{N}, \boldsymbol{\theta}, \mathbf{A}) = & \prod_{k=1}^{K}[\prod_{t=1}^{T}[Bern(E_{kt}|\eta_k)]Gam(\psi_k|a_\psi, b_\psi)] \\
& \prod_{i=1}^{I}[Mult(\mathbf{B}_i|1, \mathbf{p}_B)Dir_K(\boldsymbol{\pi}_i|\boldsymbol{\alpha}_\pi(\mathbf{B}_i))Gam(\rho_i|a_\rho, b_\rho) \\
& \prod_{k=1}^{K}[\prod_{t=1}^{T}[Pois(D_{kit}|E_{kt}, \psi_k, \boldsymbol{\pi}_i, \rho_i)]]] \\
& \prod_{t=1}^{T}[\prod_{k=1}^{K}[Dir_V(\mathbf{V}_{kt}|\mathbf{a}_{kt}) \\
& \prod_{i=1}^{I}[\prod_{d=1}^{D_{kit}}[Pois(W_{dkit}|\lambda_D)Mult(\mathbf{N}_{dkit}|W_{dkit}, \mathbf{V}_{kt})]]]] \\
& \prod_{p=1}^{P}[Norm(\theta_p|\mu_\theta, \sigma_\theta^2)] \\
& \prod_{t=1}^{T}[\prod_{i=1}^{I}[\prod_{i' \neq i}[Bern(A_{ii't}|\boldsymbol{\theta}, B(i,i'), L_{i'it}, I_{i't}, O_{it})]]]
\end{aligned}
\tag{3.23}
$$

3.5.2 A Data Augmentation

While the generative model assumes a Poisson distribution on post counts D_{kit}, we rely on a data augmentation for the inference procedure. Because counts D_{it} of posts on each blog each day are already known, we augment the generative model with latent variables $\{z_{d_{it}}\}_{d_{it}=1}^{D_{it}}$ which instead tell the latent topic assignment of post d_{it}. We can then re-write the Poisson likelihood $\prod_{k=1}^{K} Pois(D_{kit}|\lambda_{kit})$ as a multinomial likelihood $\prod_{d_{it}=1}^{D_{it}} Mult(z_{d_{it}}|1, \boldsymbol{\xi}_{it})$ with $\xi_{kit} = \frac{\lambda_{kit}}{\sum_{k=1}^{K}\lambda_{kit}}$. This reformulation enables use of the topic assignment inference algorithm from GSDMM.

After augmenting the data, we re-write a revised joint distribution:

$$
\begin{aligned}
P(\mathbf{E}, \boldsymbol{\psi}, \mathbf{B}, \boldsymbol{\pi}, \boldsymbol{\rho}, \mathbf{D}, \mathbf{V}, \mathbf{W}, \mathbf{N}, \boldsymbol{\theta}, \mathbf{A}) = & \prod_{k=1}^{K}[\prod_{t=1}^{T}[Bern(E_{kt}|\eta_k)]Gam(\psi_k|a_\psi, b_\psi)] \\
& \prod_{i=1}^{I}[Mult(\mathbf{B}_i|1, \mathbf{p}_B)Dir_K(\boldsymbol{\pi}_i|\boldsymbol{\alpha}_\pi(\mathbf{B}_i))Gam(\rho_i|a_\rho, b_\rho) \\
& \prod_{t=1}^{T}[\prod_{d=1}^{D_{it}}[Mult(z_{kit}|1, E_{kt}, \psi_k, \boldsymbol{\pi}_i, \rho_i)]]] \\
& \prod_{t=1}^{T}[\prod_{k=1}^{K}[Dir_V(\mathbf{V}_{kt}|\mathbf{a}_{kt}) \\
& \prod_{i=1}^{I}[\prod_{d=1}^{D_{kit}}[Mult(\mathbf{N}_{dkit}|W_{dit}, \mathbf{V}_{z_{dit}t})]]]] \\
& \prod_{p=1}^{P}[Norm(\theta_p|\mu_\theta, \sigma_\theta^2)] \\
& \prod_{t=1}^{T}[\prod_{i=1}^{I}[\prod_{i' \neq i}[Bern(A_{ii't}|\boldsymbol{\theta}, B(i,i'), L_{i'it}, I_{i't}, O_{it})]]]
\end{aligned}
\tag{3.24}
$$

3.5.3 Posterior Inference with Gibbs Sampling and Metropolis Hastings

To sample from the joint posterior for model parameters, we use Gibbs sampling of Geman and Geman (1987). When a parameter's posterior distribution can't be sampled from a known distribution, we use Metropolis Hastings within Gibbs sampling, following Gilks et al. (1995). The overall inference procedure consists of four stages, which we do in this order:

1. Each day t, for each blog i, sample a topic assignment $z_{d_{it}}$ for each post d_{it} and update the matrix of daily topic specific token-distributions $\mathbf{V}_t$.

2. For blogs, update topic interest proportions (π_{ik}), and base rate for posting (ρ_i). For events, update the event matrix $\mathbf{E}$, and activation level parameters

(ψ_k).

3. Update the network parameters, i.e., $\theta_0,\ \theta_1,\ \theta_2,\ \theta_3$ and θ_4.

4. Update each blog's block assignment b_i and corresponding topic interest indicators $\mathbf{B}_i$.

3.5.4 Full Conditional Posterior Distributions for Parameters

From the full joint distribution we can read off an expression proportional to the full conditional posterior distribution for each parameter given all others by using Bayes theorem. The strategy is to retain the terms which include the parameter of interest and absorb others into a proportionality constant.

Update Document Topic Assignments

For the document specific topic assignments $p(z_{dit} = k|-)$ given all other values, the full conditional posterior probability is:

$$p(z_{dit} = k|-) \propto \lambda_{kit} \prod_{w=1}^{W_{dit}} V_{ktw_{dit}}. \tag{3.25}$$

A reasonable option is to use Metropolis Hastings to sample from this distribution. In the existing inference algorithm, however, we follow Yin and Wang (2014) and marginalize out dependence on other parameters, sampling from a distribution dependent only on summary statistics of the other z_{dit} and model hyperparameters. This enables sampling each z_{dit} from a closed form multinomial as defined by equations 3.26, 3.6, 3.28, and 3.29.

Details of the derivation are given in Henry et al. (2016), and an expression for the final posterior probability is:

$$\mathbb{P}[Z_d = k \,|\, \mathbf{V}_{k,t}, d, \lambda_{ikt}] = \frac{\mathbb{P}[Z_d = k \,|\, \mathbf{V}_{k,t}, d]\mathbb{P}[Z_d = k \,|\, \lambda_{ikt}]}{\sum_{q=1}^{K} \mathbb{P}[Z_d = q \,|\, \mathbf{V}_{q,t}, d]\mathbb{P}[Z_d = q \,|\, \lambda_{iqt}]}. \tag{3.26}$$

The first term in the numerator is a version of the update given in Yin and Wang (2014) and modified to be time dependent. It can be written:

$$\mathbb{P}[z_d = k \mid \mathbf{V}_{kt}, d] = \frac{m^*_{k,t,-d} + \alpha}{|D_{t-\ell:t}| - 1 + K\alpha} \frac{\prod_{w \in d} \prod_{s=1}^{N_d^w} (n^{*w}_{k,t,-d} + \beta + s - 1)}{\prod_{i=1}^{N_d} (n^*_{k,t,-d} + W\beta + i - 1)}, \tag{3.27}$$

where to reduce computation, we approximate $\mathbb{P}[Z_d = k \mid \lambda_{ikt}]$ with $\lambda_{ikt} / \sum_{j=1}^{K} \lambda_{ijt}$, as clarified in the data augmentation above. Also

$$m^*_{k,t,-d} = (\sum_{t'=t-\ell}^{t} D_{t'k}) - 1 \text{ with } D_{tk} = \sum_i D_{tki}, \tag{3.28}$$

to be the number of posts assigned to topic k in the interval from $t - \ell$ to t, not including post d by blog i at time t. Also

$$n^{*w}_{k,t,-d} = \sum_{t'=t-\ell}^{t} n^w_{k,t'}, \tag{3.29}$$

is the number of times that token w occurs in topic k in the interval from $t - \ell$ to t, excluding post d.

The α controls the prior probability that a post is assigned to a topic; increasing α implies that all topics grow equally likely. The β relates to the prior probability for each token in a topic; increasing β results in broader, and therefore fewer topics found by the sampler. Finally, $|D|$ is the number of posts in total, and W is the size of the vocabulary. Here $|D_{t-\ell:t}|$ is the number of posts within the lag window.

Update Daily Topic Specific Word Distributions

It's possible to read off the posterior distributions for other parameters using the same approach. Next we derive the update for the topic word distributions:

$$p(\mathbf{V}_{kt}|-) \sim Dir(\mathbf{a}_{kt} + \sum_{i=1}^{I} \sum_{d=1}^{D_{it}} \mathbf{w}_{kdit}). \tag{3.30}$$

The existing implementation using summary statistics to get a point estimate for each $\mathbf{V}_{kt}$ from 3.29. Therefore instead of sampling we compute 3.31:

$$\mathbf{V}_{kt} = n^{*w}_{k,t,-d}. \tag{3.31}$$

Currently the estimation is coded to first update the document specific topic assignment, then to update the corresponding topic distribution using only that post, and then to proceed to the next document. This procedure is repeated for all documents written at that time. To allow the topic specific word distributions to stabilize, this procedure is repeated for 10 iterations within a day. Moving to the next day $t+1$, we then update summary statistics $\mathbf{m}_t$ which estimates the topic specific token distribution $\mathbf{V}_{kt}$ from the previous day t. This continues from $t = 1 : T$.

Update Blog Specific Topic Interest Proportions

The next update is for blog specific topic interests $\boldsymbol{\pi}_i$. This posterior distribution is given by:

$$\begin{aligned} p(\boldsymbol{\pi}_i|-) \propto & \prod_{k=1}^{K} [\pi_{ik}^{P1(k\in\mathbf{B}(b))+1(k\notin\mathbf{B}(b))-1} \\ & \prod_{t=1}^{T} [(\rho_i\pi_{ik} + \rho_i E_{tk}\psi_k)^{D_{kit}} \exp(-(\rho_i\pi_{ik} + \rho_i E_{tk}\psi_k)^{D_{kit}})]] \\ & \prod_{t=1}^{T} [\prod_{\{s,r:(s=i)\oplus(r=i)\}} [p_{srt}^{A_{srt}}(1-p_{srt})^{1-A_{srt}}]] \end{aligned} \tag{3.32}$$

Here there is additional dependence on $\boldsymbol{\pi}_i$ through the linking probability $p_{ii't}$ defined in Equations 3.20 and 3.21 and more specifically through the regressor $B(i,i') = 1(b_i = b_{i'}) + \boldsymbol{\pi}_i^T\boldsymbol{\pi}_{i'}1(b_i \neq b_{i'})$ for the node similarity.

Because this isn't a closed-form distribution from which we can directly sample, we use Metropolis-Hastings to generate samples. The proposal distribution utilizes the previous π_i and is:

$$\boldsymbol{\pi}_i^* \sim Dir(\boldsymbol{\pi}_i D_i), \tag{3.33}$$

where D_i is the total number of posts generated by node i.

Update Blog Specific Baseline Post Rates

We find the posterior distribution $p(\rho_i|-)$ for each blog's baseline posting rate is:

$$p(\rho_i|-) \sim Gam(a_\rho + \sum_{t=1}^{T}\sum_{k=1}^{K} D_{kit}, b_\rho + \sum_{t=1}^{T}\sum_{k=1}^{K} \pi_{ik} + E_{tk}\psi_k). \tag{3.34}$$

We use Metropolis-Hastings to generate samples. The proposal distribution utilizes the previous ρ_i and is:

$$\rho_i^* \sim Nor_{0+}(\rho_i, \sigma_\rho^2), \tag{3.35}$$

where Nor_{0+} is a zero-truncated normal distribution, chosen to uncouple the mean and variance.

Update Daily Topic Specific Event Indicators

Intuitively it makes sense that baseline post rates can guide estimation of events. The posterior distribution for event indicators is:

$$p(E_{kt} = 1|-) \propto \eta_k \prod_{i=1}^{I} [(\pi_{ik} + \psi)^{D_{kit}}] \exp(\psi_k \sum_{i=1}^{I} [\rho_i]). \tag{3.36}$$

We use Metropolis-Hastings to generate samples. The proposal distribution is:

$$p(E_{kt}^*|E_{kt}) = 1(E_{kt} = 1) \tag{3.37}$$

Update Topic Specific Event Magnitudes

The posterior distribution for topic specific event magnitudes can then be written:

$$p(\psi_k|-) \propto \psi^{a_\psi - 1} \prod_{i=1}^{I} [\prod_{t=1}^{T} [(\pi_{ik} + E_{tk}\psi_k)^{D_{kit}}]] \exp(-(b_\psi + \sum_{t=1}^{T} [E_{tk}] \sum_{i=1}^{I} [\rho_i])\psi_k). \tag{3.38}$$

We use Metropolis-Hastings to generate samples. The proposal distribution is:

$$\psi_k^* \sim Nor_{0+}(\psi_k, \sigma_\psi^2). \tag{3.39}$$

Update Link Regression Coefficients

The update for the coefficients for each variable in the loistic regression for links is:

$$p(\theta_p|-) \propto \exp(\frac{-(\theta_p - \mu_\theta)^2}{2\sigma_\theta^2}) \prod_{t=1}^{T} [\prod_{\{s,r:(s=i)\oplus(r=i)\}} [p_{srt}^{A_{srt}}(1-p_{srt})^{1-A_{srt}}]]. \tag{3.40}$$

Here there is additional dependence on θ_p through the linking probability $p_{ii't}$ defined in Equations 3.20 and 3.21.

We use Metropolis-Hastings to generate samples. The proposal distribution is:

$$\theta_p^* \sim Nor_(\theta_p, \sigma_{\theta_p}^2). \tag{3.41}$$

Update Blog Specific Community Assignments

Perhaps the most difficult posterior distributions to derive is that of the blog specific community assignments $\{\mathbf{B}_i\}_{i=1}^I$. Use of the full joint distribution of data and parameters allows us to derive a general expression for $P(\mathbf{B}_i = \mathbf{B}(b))$ up to proportionality which can be used to sample each blog's community assignment given the rest. As in equation (3.9), we let $\mathbf{B}(b) = \{k : I_{kb} = 1\}$ denote the set of topic interests for community b. The posterior is then

$$
\begin{aligned}
P(\mathbf{B}_i = \mathbf{B}(b)|\mathbf{B}_{-i}, \boldsymbol{\pi}, \boldsymbol{\theta}, \mathbf{A}) \propto & Mult(\mathbf{B}_i|1, \mathbf{p}_B) Dir_K(\boldsymbol{\pi}_i|\boldsymbol{\alpha}_\pi(\mathbf{B}(b))) \\
& \prod_{t=1}^{T}[\prod_{\{s,r:(s=i)\oplus(r=i)\}}[Bern(A_{srt}|\boldsymbol{\theta}, B(s,r), L_{rst}, I_{rt}, O_{st})]] \\
& \propto p_b \frac{\Gamma(\sum_{k=1}^{K}[P1(k \in \mathbf{B}(b)) + 1(k \notin \mathbf{B}(b))])}{\sum_{k=1}^{K}\Gamma(P1(k \in \mathbf{B}(b)) + 1(k \in \mathbf{B}(b)))} \\
& \prod_{k=1}^{K}[\pi_{ik}^{P1(k\in\mathbf{B}(b))+1(k\notin\mathbf{B}(b))-1}] \\
& \prod_{t=1}^{T}[\prod_{\{s,r:(s=i)\oplus(r=i)\}}[p_{srt}^{A_{srt}}(1-p_{srt})^{1-A_{srt}}]]
\end{aligned}
\tag{3.42}
$$

Moving from the first expression to the second, the multinomial portion simply becomes the prior probability p_b of being assigned to community b. The Dirichlet portion has concentration parameter vector $\boldsymbol{\alpha}_{\pi i}(\mathbf{B}(b))$ which we write explicitly as a function of the community $\mathbf{B}(b)$ as defined in equation (3.11). In the second product of the final term, $\oplus$ is the "XOR"-symbol, or the exclusive disjunction, and in this context it means we consider all blog pairs in which blog i is either the sender ($s = i$) or the receiver ($r = i$) but not both. We recall p_{srt} is defined as in equation (3.20) and equation (3.21), along with the network covariates in the bulleted portion of the modeling section. the expression for community similarity between nodes $i*$ and i' depends whether i is sender ($i* = i$) or receiver ($i' = i$), and can be written:

$$
\begin{aligned}
B(i*, i') = & [1(\mathbf{B}(b) = \mathbf{B}_{i'}) + \boldsymbol{\pi}_{i*}^T \boldsymbol{\pi}_{i'} 1(\mathbf{B}(b) \neq \mathbf{B}_{i'})] 1(i* = i) + \\
& [1(\mathbf{B}_{i*} = \mathbf{B}(b)) + \boldsymbol{\pi}_{i*}^T \boldsymbol{\pi}_{i'} 1(\mathbf{B}_i \neq \mathbf{B}(b))] 1(i* \neq i)
\end{aligned}
\tag{3.43}
$$

In the existing implementation we made some simplifications.

Details are given in Henry et al. (2016), and the final expression used in the code is:

$$\mathbb{P}[b_i = b \mid \mathbf{A}, \boldsymbol{\theta}, \boldsymbol{\pi}_i, \mathbf{B}_{-i}] \propto$$

$$\frac{N_{b,-i} + \alpha_B}{\alpha_B |B| + N - 1} P(A \mid \boldsymbol{\theta}, B_i = b) P(\pi_i \mid B_i = b) P(|B| \mid \lambda_B), \tag{3.44}$$

where $N_{b,-i}$ is the number of nodes assigned to community b, excluding node i, α_B relates to the prior probability of being assigned to any community, $|B|$ is the number of non-empty communities when sampling node i's assignment to community b, λ_B is the prior expected number of communities, and $\mathbf{B}_{-i}$ is the set of community assignments with the ith blog's community assignment removed.

Last, we restricted the number of eligible communities in a way that can be practically achieved as follows. First, multiply each expressiion proportional to the posterior probability by an indicator $1([\sum_{d_i=1}^{D_i} 1(z_{d_i} = k)] > 0)$ that is 1 only if at least one document on blog i has been assigned to topic k. This reduces computational complexity of the sampling.

3.6 Results

Hyperparameter Specification

The model described in Section 3.4 has several hyperparameters to specify. One hyperparameter to specify is the number of topics. We discuss this in the following section after giving other hyperparameter specifications. Another hyperparameter is the time lag ℓ for temporal topic dependency, which specifies the maximum number of days ago which topical content is remembered by the model. We expect that in the blog setting, content in the education topic from over two months ago is unlikely to significantly influence today's discussion of education content, so we choose $\ell = 62$ days.

The hyperparameter P guides the number of times as likely a blog is to post about a community specific interest rather than a topic which is not of interest to that community. We want the communities to have large inpact on topical discussion content so we choose a large $P = 50$. We recall that for a typical community there

are many more topics which are not of interest that that are, so we expect a blog posts about a non-community interest more often than 1/50 of the time.

The number B of communities is equal to the total number of allowable unique combinations from the set of K topics. Allowing combinations of any size, yields a total of 2^K communities. For even a small number of topics $K = 20$ this is over one million communities, which is computationally difficult to work with. Our restriction of considering only communities of size one, two, three, or all topics results in a more tractable number of communities. The above hyperparameters can be easily varied depending on the specific characteristics of the application.

We wanted priors $N(\mu_{\theta_p}, \sigma^2_{\theta_p})$ for each of the network model parameters to be non-informative. Therefore we chose prior mean $\mu_{\theta_p} = 0$ and standard deviation $\sigma^2_{\theta_p} = 1000$ for each. In the Metropolis Hastings, the proposal standard deviation was set to 1 for the edge parameter, and to 0.25 for each of the other parameters in the network model. For the topic model, regarding the update in , we use $\alpha = \beta = 0.1$. These are the concentration parameters for the priors on the document topic proportions and topic word proportions. The prior for the baseline post rates ρ_i in Equation 3.12, has mean $\mu_\rho = 4$ and standard deviation $\sigma_\rho = 1000$. The prior for topic specific event magnitude parameters ψ_k in Equation 3.5 has $\mu_\psi = 0$ and standard deviation $\sigma_\psi = 1000$, and a proposal standard deviation of $\sigma^*_\psi = 0.5$. Additionally, $\lambda_B = 25$ was the prior mean number of communities, and the prior tendency for community membership $\alpha_B = 1$. The prior probability of topic specific events was set to $\eta_k = 0.2$.

3.6.1 Specifications for the Sampling Algorithm

The analysis was acquired from an outer loop of 1000 samples. For the burn, the first 100 samples were discarded. We thinned the remaining 900 samples to reduce autocorrelation, keeping every 10th sample, retaining 90 samples. In each sample, the document assignments and topic word distributions were updated 10 times, the network parameters were updated 10 times, and the community assignments were

updated 10 times.

3.6.2 Choosing the Number of Topics

Here we discuss how to choose an appropriate value for K, the number of topics. Choosing a small K around 20 usually leads to broader topics (e.g., education, entertainment, and health) with a large number of relatively high probability tokens, and choosing a big K around 200 usually leads to more specific topics (e.g. statistics, probability, and machine learning) which focus on a small number of tokens. We are interested in broader topics. Following Arun et al. (2010), we fit models with values of K ranging from 10 to 30 and show the resulting criteria in Figure 3.1. We select the value $K = 22$ which minimizes the entropy based KL criterion. This results in $B = 1794$ possible topic interest communities. We note that fully Bayesian and non-parametric approaches which put a prior on K and let the data choose are an attractive alternative Teh et al. (2005).

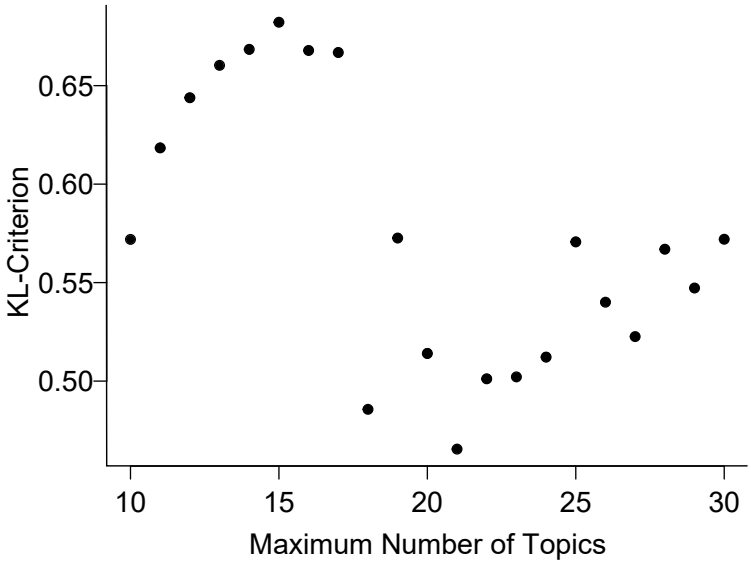


FIGURE 3.1: The criterion curve, as in Arun et al. (2010), for determining the number of topics.

3.6.3 Assessing Convergence and Autocorrelation

Overall the parameters converged. To assess the mixing of the document-specific topic assignments and of the blog-specific community assignments, we calculated Adjusted Rand Indices (ARI) (Hubert and Arabie, 1985), (Steinley, 2004) for each iteration i compared to iteration $i-1$. The former was very stable, with a mean ARI of 0.806 and ARI standard deviation of 0.047. The community assignment was less stable, with a mean ARI of 0.471 and ARI standard deviation of 0.031. We believe this variability is due to an observation that many bloggers posted about a variety of news event topics but not all of them. Therefore the sampler didn't always assign them to a community with interest in three or fewer topics or the community that is interested in all topics.

Topic Results

All 22 topics had distinct subject matter. Topics were named by the authors on the basis of the highest probability and most distinctive tokens over all days. Distinctiveness was calculated using Bayes' theorem as in Equation 3.45. The idea is for each topic, to sort words in order according to which word's usage corresponds to the highest probability of that topic.

$$P(Z_d = k \mid w \in d) = \frac{P(w \in d \mid Z_d = k)P(Z_d = k)}{P(w \in d)} \approx \frac{[V_{kw.}][D_{k..}/D]}{[W_w/W]}. \tag{3.45}$$

Here, $D_{k..}$ gives the number of documents assigned to topic k, D the total number of documents, W_w the count for word w, and W the total number of words. However, sometimes the empirical estimate is 0, so we give each topic and each word a small weight and renormalize so that there is always a non-zero minimum probability.

Table 3.1 contains the five highest probability tokens in each topic over all days.

Table 3.1: The most frequent words in each topic.

Topic Names	Most Frequent Words				
	1	2	3	4	5
Feminism	women	peopl	dont	person	life
Keystone Pipeline	energi	oil	price	compani	industri
Birth Control	right	state	law	marriag	women
Election	obama	romney	peopl	presid	polit
Mortgages	case	court	bank	judg	attorney
Entertainment	peopl	dont	good	work	game
Middle East	israel	islam	american	peopl	countri
LGBT Rights	gay	peopl	marriag	homosexu	support
Sensational Crime	gun	polic	report	peopl	zimmerman
Technology	compani	googl	facebook	appl	user
Supreme Court	law	court	state	case	constitut
Bank Regulation	bank	market	money	price	compani
National Defense	iran	militari	israel	nuclear	obama
Republican Primary	romney	republican	obama	poll	vote
Voting Laws	state	vote	elect	voter	counti
Political Theory	libertarian	peopl	right	state	govern
Eurozone	bank	debt	economi	rate	govern
Taxation	tax	state	govern	cut	obama
Diet and Nutrition	peopl	dont	govern	polit	work
Education	school	student	teacher	educ	state
Global Warming	climat	climat.chang	temperatur	scienc	scientist
Terrorism	report	govern	attack	inform	case

Table 3.2 contains the chosen topic names, total number of posts in each topic, and the three tokens that have the highest distinctiveness for each topic over all days.

Table 3.2: Topic names and their most specific tokens.

Topic Name	# of posts	Highest Specificity Tokens		
		1	2	3
Feminism	3971	russel.saunder.juli	circumcis	femin
Keystone Pipeline	4422	loan.guarante.program	product.tax.credit	tar.sand.pipelin
Birth Control	2703	contracept.coverag	birth.control.coverag	religi.organ
Election	14713	soptic	cheroke	eastwood
Mortgages	2130	estat	probat	fiduciari
Entertainment	10555	email.read.add	olivia	free.van
Middle East	6068	mursi	morsi	fatah
LGBT Rights	5425	anti.gay.right	support.equal.marriag	equal.marriag
Sensational Crime	6423	zimmerman	lanza	mass.shoot
Technology	3230	mail.feel.free	pipa	ret
Supreme Court	1767	commerc.claus	bork	chief.justic.robert
Bank Regulation	5222	volcker	dimon	libor
National Defense	1977	iaea	iranian.nuclear.weapon	warhead
Republican Primary	9351	poll.mitt.romney	nation.popular.vote	romney.lead
Voting Laws	7865	ohio.secretari.state	voter.registr.form	hust
Political Theory	1448	bylaw	rawl	sweatshop
Eurozone	1832	standalon	troika	ecb
Taxation	8435	tax.polici.center	health.care.spend	top.tax.rate
Diet and Nutrition	3057	spielberg	harlan	calori
Education	2909	chicago.teacher.union	chicago.public.school	charter.school
Global Warming	2205	arctic.sea.ice	sea.ice	sea.level.rise
Terrorism	3347	kimberlin	broadwel	assang

We note the popularity of three topics: "Election", "Entertainment", and "Republican Primary". The highest specificity tokens are those whose usage is most

distinctive to the topic. Therefore the use of the word "Soptic", presumably referring to Joe Soptic, a man from a controversial ad campaign during the election, is most likely to indicate the topic is "Election". While some of the tokens may seem obscure, they are generally pertinent to the identified topics.

We don't detail the dynamics of all 22 topics; rather, we focus on one topic, "Sensational Crime", as an example of what our analysis can reveal. These posts largely pertain to four tragic events: the shooting of Trayvon Martin in February, the Aurora movie theater massacre in July, the Sikh Temple shooting in August, and the Sandy Hook massacre in December.

To illustrate how the salience of a token changes over time, we use a weighted frequency proportion which is equal to:

$$WF_{w \in k} = \frac{P(Z_{d_t} = k_t \mid w \in d) F(w \in k_t)}{\sum_{w^* \in V} P(Z_{d_t} = k_t \mid w^* \in d) F(w^* \in k_t)}. \tag{3.46}$$

Here, in the numerator, the first term is a day specific version of the probability of a topic given a word from 3.45, and the second term is the count of the token w in posts assigned to the kth topic at time t. It can be interpreted as the proportion of tokens assigned to topic k at time t that are token w, weighted by the distinctivity of token w in topic k at time t. It is useful in this context since it upweights tokens that are very distinctive to the topic at the time, boosting them relative to tokens which have high frequency in multiple topics (such as "people") but may not be especially distinctive. Recall that the topic-specific token-distributions, which are used in the distinctivity computations, are computed over the past 62 days, which accounts for the smoothness of the curves. The shading around a curve is a 95% Bayesian credible interval.

Figure 3.2 presents the weighted frequency curves for chosen tokens related to "Sensational Crime" topic through 2012. Notably, the highest scoring token is "Zimmerman" specifically, rather than the usual top terms over the course of the year which are "police" and "gun". This demonstrates our criteria is capable of finding distinctive tokens for a topic over time. Furthermore, the figure demonstrates the

potential of this quantity to aid in topic specific event detection by conditioning on token weighted proportion, which increases sharply after events.

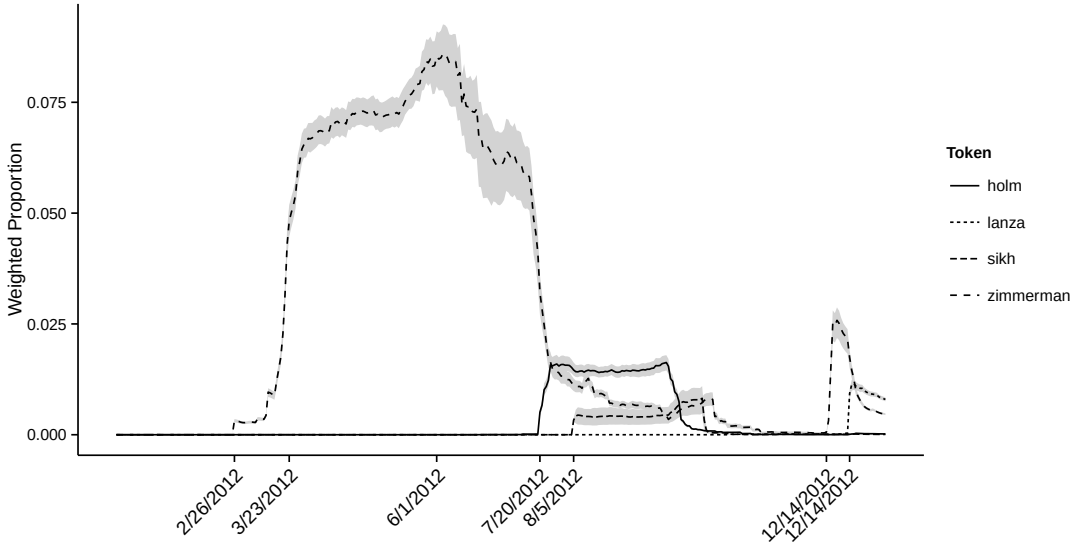


FIGURE 3.2: Token specific weighted frequencies over time during major events in the Sensational Crime topic. 2/26: Trayvon Martin shot, 3/23: Barack Obama comments, 7/20: Aurora shooting by James Holmes, 8/5: Sikh Temple shooting by Michael Page, 12/3: Zimmerman's injuries released, 12/14: Sandy Hook massacre by Adam Lanza.

3.6.4 Blog Specific Baseline Post Rates

Blog specific baseline post rates had posterior mean of 0.632 and standard deviation of 1.67. The distribution was very right skewed, with most blogs having a low post rate and few having a very high post rate. The largest post rate was 22.69.

3.6.5 Topic Specific Event Magnitudes

Posterior means and standard deviations for topic specific event magnitudes are in Table 3.3 and were calculated after the topics had been named.

Table 3.3: Topic Specific Activation Parameters ψ_k

Topic	Posterior Mean	Standard Deviation
Feminism	0.0034	0.0029
Keystone Pipeline	0.0037	0.0017
Birth Control	.00001	.00003
Election	0.3917	0.0249
Mortgages	.00001	.00005
Entertainment	0.0018	0.0014
Middle East	0.0378	0.0129
LGBT Rights	0.0028	0.0026
Sensational Crime	0.0208	0.0085
Technology	0.0012	0.001
Supreme Court	0.0022	0.0032
Bank Regulation	0.0021	0.0021
National Defense	0.0014	0.0013
Republican Primary	0.2267	0.0188
Voting Laws	0.0375	0.0095
Political Theory	0.0012	.00001
Eurozone	0.0001	.00002
Taxation	0.0135	0.0084
Diet and Nutrition	0.0804	0.0139
Education	0.0011	0.0012
Global Warming	0.0019	0.0011
Terrorism	0.0022	0.0019

The "Election" and "Republican Primary" topics have the largest posterior means, which suggests that these topics were relatively more event driven than other topics.

3.6.6 Link Regression Results

Table 3.4 shows the posterior means of the network parameters with 95% credible intervals.

Table 3.4: Posterior means and 95% credible intervals for network parameters.

Parameter	Posterior Mean	95% CI
Intercept	-8.524	[-8.539, -8.513]
Community	1.058	[0.638, 1.485]
7 day lag	-0.163	[-0.198, -0.131]
Indegree of receiver	0.497	[0.496, 0.499]
Outdegree of receiver	0.330	[0.329, 0.332]

The small posterior mean for the intercept parameter indicates that the network is generally sparse. The community parameter is relatively large and positive, suggesting that blogs with shared interests are more likely to link to each other. This result directly links the network model to the topic model, and hustifies claims about topics guiding the discovered communities. Contrary to our intuition, the 7-day lag parameter was negative, suggesting that blogs which were recently linked are less likely to link. There are a few possible explanations, likely involving confounding variables. One, links may not be prompted by recent links but by events taking place. Increased links following event would be followed by decreased links as the event expires. Second, if bloggers are conflict averse and linking is argumentative, then once the counter-point is made, the blogger might be less likely to reply. Third, perhaps bloggers are offput by reactive links. They may like to link to other posts which have not been linked yet. The in-degree and out-degree of a blog both positively associate with the probability that the blog receives links. These parameters control for the influence of highly outgoing and popular blogs such as *The Blaze* and *The Huffington Post* (Henry et al., 2016).

3.6.7 Community Detection Results

A main contribution of this analysis is an ability to inspect link patterns for a specified topic interest community. To demonstrate, we focus on a community of 21 blogs whose posterior mode community assignment has interest primarily in "sensational crime". First we inspect links received. Only two of these blogs received links in 2012, and only one received links within the community (*legalinsurrection.com*). Despite minimal within community link data, our inference algorithm recognizes this community, presumably because its constituent blogs write about the same topic. This is an improvement over what a modularity based assortative community detection method would likely find, because communities need not have high within block linking probability.

Next we consider sent links. There are a relatively enormous 62 blogs, to which members of this community link. Of these, 15 receive $\approx 90\%$ of the links. We can

therefore reasonably describe "sensational crime" blogs as a community of "commenter" blogs that react to posts on larger blogs.

It's also possible with our model to describe linking patterns in a specific community around a related event. Figures 3.3 and 3.4 depict links from the "sensational crime" community, before and after the Aurora, Colorado movie theater shooting on 7/20/2012. In each plot, the links are aggregated over fifteen days.

For clearer presentation, only blogs satisfying at least one of the following two criteria are plotted: the blog is in the "sensational crime" community and posted during the specified time interval, or the blog is in the highly linked 15 block subset and is linked during the interval. Only links sent by members of the "sensational crime" community are plotted. Link width is proportional to square root of link count. Circular nodes are in the "sensational crime" community, and square nodes are generally in multi-topic communities, where one of the topics is "sensational crime".

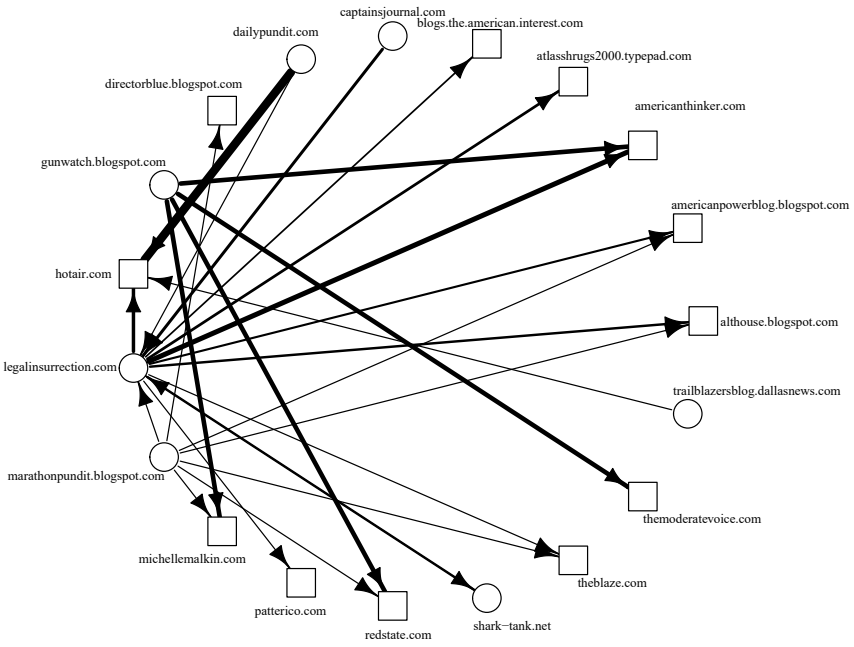


FIGURE 3.3: Links from the "sensational crime" community shortly *before* the Aurora, Colorado shooting.

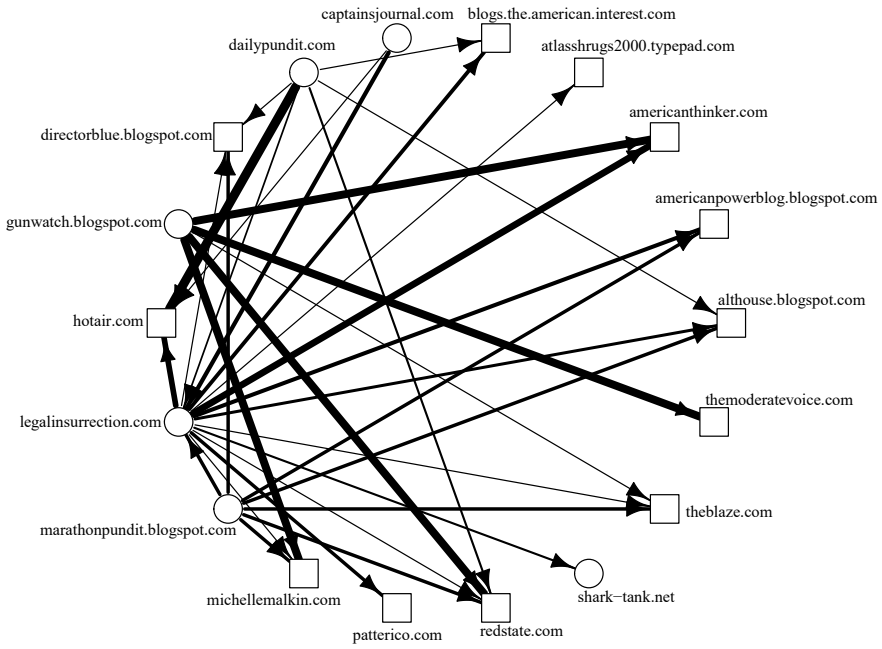


FIGURE 3.4: Links from the "sensational crime" community shortly *after* the Aurora, Colorado shooting.

The linkage patterns from the community are generally similar, with two noteworthy changes. First, some existing links increase in count, for example, the four links emerging from `gunwatch.blogspot.com` in the "10 o'clock" position increase in thickness. Second, a few new links occur, for example, sent by `dailypundit.com` at "11 o'clock" or received by `redstate.com` at "6 o'clock". The number of links in the 15 days before the shooting was 197, but afterwards it was 427. We conclude that the methodology can detect persisting communities with external, topic-guided links, whose linkage rates are increase with relevant news events.

A similar analysis of the network before and after Barack Obama commented about Trayvon Martin revealed apparently small change in network structure. An increase in post rate following this event resulted in more posts, those these were generally assigned to the "election" topic, rather than the "sensational crime" topic.

3.7 Conclusion

We developed a Bayesian model to simultaneously learn communities and dynamic topics in a dynamic text network. The topic assignments depend on word usage and related event indicators, and the Community assignment depends on both the observed links and the topics in which the bloggers are interested.

In a network of political blog posts, our model learns interpretable communities with realistic topic co-occurrences. Two examples are a community interested in the "election" and the "republican primaries", which are related through shared mention of republican candidate, Mitt Romney, and a second community interested only in "sensational crime", which links more following violent news events. We note that the model for link formation is flexible and could be extended to include additional data about the blogs such as political affiliation.

Topic word distributions are day specific and inferred from summary statistics of the document topic assignments. The learned topics for the political blog are consistent with familiar topics in politics including "education", "national defense", and "taxation". We present topics with tokens sorted by probability and introduce a method based on Bayes theorem for quantifying words' distinctiveness to each topic. Using a daily, topic-specific token frequency, weighted by a distinctiveness, we show that key token prevalence in a topic increases sharply following events relevant to that topic.

Finally, the generative model is general enough that this model and inference algorithm can be reasonably applied to other dynamic text networks. I am curious what topics, communities, and events the model might discover in a corpus of Twitter data, with tweets, re-tweets, and discretized time stamps.

4

Partitioning Documents by Topics and Node Level Links

4.1 Overview of the Topic Link Block Model

In this chapter, we introduce a novel Bayesian statistical model for simultaneously discovering topics and clustering documents which have a network structure. In much of existing literature for network topic models, links occur at a document-to-document level or a node-to-node level. Here, we model links at a document-to-node level, because they occur this way in the data. Specifically, we apply the model to political blog posts from 2012. Inference uses parallelizable Gibbs sampling to simulate from the posterior conditional distributions for model parameters. Most probable words from selected topics are displayed, discovered blocks are visualized, and results are compared with a strong baseline from the joint network topic modeling literature.

4.2 Probabilistic Topic Models on Networks

Many data sets exist that involve networked entities with associated text documents. Potential applications include analyzing web pages, text messages, chat forums, blogs, and emails. In such settings it can be informative to group entities into blocks with common text usage or link sending patterns and to summarize textual content with a set of topics.

Here we discuss the existing literature of probabilistic models which combine topic modeling and network modeling. One example is the relational topic model (RTM) of Chang and Blei (2009a). This paper extends the foundational latent Dirichlet allocation (LDA) topic modeling paper Blei et al. (2003b) by conditioning the probability of a link between documents on an inner product of latent topic counts for each document. This paper does not take into account author information for the documents, which is sometimes available.

Another relevant model is Block LDA of Balasubramanyan and Cohen (2011), which combines aspects of LDA from Blei et al. (2003b) and the mixed membership stochastic block model (MMSB) from Airoldi et al. (2008). In this model, links happen at the entity to entity level, and a topic pair is sampled for each existing link between entities.

The supervised LDA model of Mcauliffe and Blei (2008) is similar to our model in the sense that it has an LDA framework with a response associated with each document. The difference is that their model can accommodate a single (categorical) response, and the regression coefficients are the same for every document, while our model handles multiple responses.

A similar approach is described in Ramage et al. (2009) and called Labeled LDA. This model associates with each document a label, giving the presence or absence of each topic. It then uses a `num_present_labels` $\times$ `num_topics` matrix which restricts

document specific topic proportions to be non-zero only for topics which are present in the label.

A relatively early paper (Erosheva et al., 2004) provides a foundation for the following two papers and is later called Link LDA. In addition to the LDA model, it introduces a second matrix of topics which instead consists of distributions over links. For each link in the document, links are assigned a topic and then drawn from the corresponding topic-link multinomial.

Pairwise Link LDA is also described in Nallapati et al. (2008). It is the same as LDA, but for each document pair, the sender and receiver each draw a topic assignment from their corresponding topic proportions vector, and a link is sampled with a topic-to-topic specific linking probability. This is the same as the MMSB with topics playing the role of blocks and documents the role of nodes.

The final model described in Nallapati et al. (2008) is Link PLSA LDA, which distinguishes between cited and citing documents. In this model cited documents are assigned a single topic. The words of the document and the documents which cite it are then drawn from their corresponding topic specific distributions. For citing documents, data generation is the same as Link-LDA described above. The model is applicable when each document is either cited or citing, but not both.

Extending Link LDA, in Liu et al. (2009) the authors describe Topic Link LDA. Each document has an author, and the probability of a link between documents is a log-linear combination of the two documents' topical similarities and authors' community similarity.

In the Author Recipient Topic model of McCallum et al. (2005), the generative model is for documents with a single author and multiple recipients. For each word of the document, a recipient is sampled and then a topic sampled from the author-recipient pair specific topic distribution.

Another approach described in Ho et al. (2012b) is called TopicBlock. The strat-

egy is to use a hierarchical latent space model in which each document is represented as a complete path through the hierarchy, and text drawn from a mixture of distributions specific to the nodes at each level of the hierarchy for that document. The links are drawn from non-terminal node specific linking probabilities which govern documents who share that node as their deepest common ancestor.

4.3 Topic Link Block LDA

We extend LDA with a novel network topic model. Rather than assigning each document its own distribution over topics, we group documents into blocks such that each block has a common distribution over topics and distribution over links. Given its block membership each document then draws topic indicators for each word using the appropriate block specific topic proportions and then words from the appropriate topic as in LDA. The links from each document are simply drawn from the appropriate block specific link proportions multinomial.

One strategy for fitting the topic link block LDA (TLB-LDA) is to begin with every observation in its own block, that is, setting $B = N$, and then letting the inference algorithm "merge blocks" when observations are close enough in link and topic distribution.

The notation for data, latent variables, and parameters is given in Table 4.1 below.

Table 4.1: Notation for TLB-LDA

$\boldsymbol{\theta}_b$	:	topic proportions for block b
$\boldsymbol{\phi}_k$	:	word proportions for topic k
$\boldsymbol{\pi}_b$	:	link proportions for block b
b_n	:	block assignment for document n
z_{wn}	:	topic assignment for word w of document n
x_{wn}	:	term assignment for word w of document n
y_{ln}	:	domain assignment for link l of document n

4.3.1 Data Generation

As is common practice in much of the Bayesian literature, the data generating procedure is discussed in terms of a probabilistic graphical model, which can be written as follows:

1. For each topic k:
   (a) $\boldsymbol{\phi}_k \sim Dir_V(\boldsymbol{\beta})$
2. For each block b:
   (a) $\boldsymbol{\theta}_b \sim Dir_K(\boldsymbol{\alpha})$
   (b) $\boldsymbol{\pi}_b \sim Dir_D(\boldsymbol{\gamma})$
3. For each document n:
   (a) $b_n \sim Cat(1/B)$
   (b) For each word w:
      i. $z_{w,n} \sim Cat(\boldsymbol{\theta}_{b_n})$
      ii. $x_{w,n} \sim Cat(\boldsymbol{\phi}_{z_{w,n}})$
   (c) For each link l:
      i. $y_{l,n} \sim Cat(\boldsymbol{\pi}_{b_n})$

Here, $Dir_K(\boldsymbol{\alpha})$ is a K-dimensional Dirichlet distribution, with $\boldsymbol{\alpha}$ the concentration parameter vector, and $Cat(\boldsymbol{p})$ is a categorical distribution (equivalently, a single draw from a multinomial distribution), with $\boldsymbol{p}$ the vector of probabilities for each category.

4.4 Inference

To derive the inference algorithm, we begin by writing the joint distribution of the data, latent variables, and model parameters. As the model is a probabilistic graphical model, the joint distribution can be written as a composition of likelihood and priors as follows:

$$
\begin{aligned}
p(X, Y, Z, b, \phi, \pi, \theta | B, K, \alpha_1, \alpha_2, \alpha_3) = & \{\prod_{n=1}^{N} [\prod_{l=1}^{L_n} Cat(y_{ln} | \boldsymbol{\pi}_{b_n})] \\
& [\prod_{w=1}^{W_n} Cat(x_{wn} | \boldsymbol{\phi}_{z_{w,n}}) Cat(z_{wn} | \boldsymbol{\theta}_{b_n})] \\
& Cat(b_n | 1/B)\} \\
& \{\prod_{b=1}^{B} Dir_K(\boldsymbol{\theta}_b | \boldsymbol{\alpha}) Dir_D(\boldsymbol{\pi}_b | \boldsymbol{\gamma})\} \\
& \{\prod_{k=1}^{K} Dir_V(\boldsymbol{\phi}_k | \boldsymbol{\beta})\}
\end{aligned} \tag{4.1}
$$

We wish to sample from the posterior distribution of the latent variables and parameters given the data. The latent variables and parameters of interest are $\{\{Z_{wn}\}_{w=1}^{W_n}\}_{n=1}^{N}$, $\{b_n\}_{n=1}^{N}$, $\{\boldsymbol{\phi}_k\}_{k=1}^{K}$, $\{\boldsymbol{\theta}_b\}_{b=1}^{B}$, and $\{\boldsymbol{\pi}_b\}_{b=1}^{B}$.

4.4.1 Gibbs Sampling

For Gibbs sampling, one can "read off" the full conditional distributions for each quantity by noting which portion of the joint distribution involves it. The resulting conditional posteriors are given in Equations 4.2, 4.3, 4.4, 4.5, and 4.6. Inference proceeds as follows:

First sample each word's topic assignment $z_{wn}|- \sim Cat(\hat{\boldsymbol{p}}_{z_{wn}})$, with

$$
\hat{p}_{z_{wn},k} = p(z_{wn} = k) = \frac{\phi_{k,x_{wn}} \theta_{b_n,k}}{\sum_{k=1}^{K} \phi_{k,x_{wn}} \theta_{b_n,k}}. \tag{4.2}
$$

Next sample each document's block assignment $b_n|- \sim Cat(\hat{\boldsymbol{p}}_{b_n})$, with

$$
\hat{p}_{b_n,b} = p(b_n = b) = \frac{\frac{1}{B} \prod_{l=1}^{L_n} \pi_{y_{ln},b} \prod_{w=1}^{W_n} \theta_{z_{wn},b}}{\sum_{b=1}^{B} \frac{1}{B} \prod_{l=1}^{L_n} \pi_{y_{ln},b} \prod_{w=1}^{W_n} \theta_{z_{wn},b}}. \tag{4.3}
$$

Next sample each topic's word proportions $\boldsymbol{\phi}_k|- \sim Dir_V(\hat{\boldsymbol{\alpha}}_{\phi_k})$, with

$$\hat{\alpha}_{\phi_k,v} = \alpha_{2,v} + \sum_{n=1}^{N} \sum_{w=1}^{W_n} 1(x_{wn} = v, z_{wn} = k). \tag{4.4}$$

Next sample each block's topic proportions $\boldsymbol{\theta}_b|- \sim Dir_K(\hat{\boldsymbol{\alpha}}_{\theta_b})$, with

$$\hat{\alpha}_{\theta_b,k} = \alpha_{3,k} + \sum_{n=1}^{N} \sum_{w=1}^{W_n} 1(b_n = b, z_{wn} = k). \tag{4.5}$$

Next sample each block's link proportions $\boldsymbol{\pi}_b|- \sim Dir_D(\hat{\boldsymbol{\alpha}}_{\pi_b})$, with

$$\hat{\alpha}_{\pi_b,d} = \alpha_{1,d} + \sum_{n=1}^{N} \sum_{l=1}^{L_n} 1(b_n = b, y_{ln} = d). \tag{4.6}$$

4.4.2 Pseudo Code

Simple pseudo code for inference in this model can be found in Algorithm 1:

Data: X,Y = words, links
Result: Posterior samples from z, b, ϕ, θ, π | X,Y
Randomly initialize z, b, ϕ, θ, π from the model;
while *iter* < *num_iters* **do**
    Gibbs sample z, b, ϕ, θ, π;
    if *iter* > *burn* **then**
        Collect samples of z, b, ϕ, θ, π;
    end
end

Algorithm 1: Gibbs sampling for TLB-LDA

4.5 Application

In general, the model can be used whenever each observation consists of a sequence of discrete random variables (e.g., words of a document) coupled with a (possibly empty) set of discrete random variables (e.g., links to nodes of a network).

4.5.1 Political Blog Posts

We experiment with a data set of political blog posts from 2012. Technorati (2012) provided a list of top political blogs and Valassis Digital (2012) scraped and formatted the posts. We removed punctuation and spacing, symbols, digits, stop words from a list provided by University of Glasgow (2017), individual letters, days, months, numbers, uncommon, and non-informative words. Each post consists of a date and web domain (which are not modeled), a set of links to other web domains, and a sequence of words. In Figure 4.1, we show ten example blog posts.

	dates	domains	links	words
1	02/06/2012	articles.businessinsider.com	businessinsider.com	tom brady super todays kim date football player entire n...
2	02/25/2012	ordinary-gentlemen.com		taxes welfare james death taxes economics finance post t...
3	03/12/2012	pinknews.co.uk		pm australia leads uk support equal marriage australia le...
4	03/27/2012	azizonomics.com	antiwar.com crookedtimber.org zerohedge.com	chart economic history economics economics generation...
5	04/05/2012	therightplanet.com	americanthinker.com atlasshrugs2000.typepad.com book...	washington senator john today said senate agriculture co...
6	04/09/2012	blogs.reuters.com		fear ben walsh email print welcome email page just matt...
7	04/17/2012	laobserved.com		dow points mark lacter pm days common weeks significa...
8	04/18/2012	dennismansfield.com		james bond beer beer just daniel comments new bond fil...
9	05/07/2012	marathonpundit.blogspot.com	althouse.blogspot.com americanpowerblog.blogspot.com...	greater mistake did little burke occupy movement lincoln...
10	10/01/2012	paradigmsanddemographics.blogspot.com		public alliance environmental media politicians interests ...

FIGURE 4.1: Ten example blog post observations

Some relevant information about the size of the data set is given in Table 4.3 below.

Table 4.2: Size of the Political Blog Posts Data Set

$N = 111,615$	:	Number of blog posts
$T = 366$	:	Total number of days
$W = 24,363,882$	:	Total number of words
$V = 5,213$	:	Number of terms in the vocabulary
$D = 467$	:	Number of blogs (web domains)
$L = 81$	:	Number of blogs linked
$E = 274,041$	:	Total number of links

This data set is rich for text or network analysis and is hosted publicly at:

https://www.dropbox.com/s/20egdva07p7p4wf/textNetwork.csv?dl=0

4.5.2 Details of Analysis

We experimented with the first week of data. We initialized the block assignments vector $\{b_n\}_{n=1}^N$ such that each blog is assigned to its own block. This allows every document to have its own topic distribution. The Gibbs sampling algorithm them proceeds by assigning blogs to common blocks, and parameter estimates gradually update given the data. The specifications for this analysis are given in Table 4.3 below.

Table 4.3: Specifications for the Blog Posts Analysis

$B = N = 1,781$	:	Number of blocks
$K = 50$	:	Number of topics
$\alpha = 0.1$	:	Hyperparameter for block topic prior
$\beta = 0.01$	:	Hyperparameter for topic word prior
$\gamma = 0.1$	:	Hyperparameter for block link prior
$num\_burn = 900$	:	Number of samples to burn
$num\_samps = 1000$	:	Number of Gibbs iterations
$num\_cores = 4$	:	Number of CPU cores for parallelization

4.5.3 Results

As is often done in topic modeling literature, we present top words from each of the learned topics. The topics were named subjectively by a human, and top words were chosen by ranking words within each topic by their posterior probabilities in the last collected Gibbs sample.

Table 4.4: Topics Learned by TLB-LDA

	Education	Court	Energy	Employment	Marriage	Climate
1	school	court	oil	jobs	marriage	climate
2	students	supreme	energy	job	gay	global
3	education	texas	gas	workers	sex	change
4	schools	courts	environmental	unemployment	state	warming
5	college	state	industry	people	men	extreme
6	student	case	prices	work	man	countries
7	class	district	natural	economy	children	winter
8	teacher	cases	pipeline	labor	sexual	new
9	law	decision	coal	economic	couples	north
10	teachers	maps	development	growth	rights	weather

Given the block assignments of every observation from the TLB-LDA, we can aggregate over all observations on a particular node, collecting a node block count matrix. More specifically, we compute counts $C_{d,b}^{(TLB)}$ of each node being assigned to a particular block, as either sender or receiver, and we divide each row by the the row sum to estimate the node specific block membership probabilities $\hat{\lambda}_{d,b}^{(TLB)}$. To avoid the potential issue of label switching, we count assignments based on the last iteration of Gibbs sampling.

One goal of this analysis is community visualization. It is difficult to visualize multi-community membership, so we opt to use k-means clustering on the blocks matrix to sort each node into a single cluster. Figure 4.2 shows the network of nodes, re-ordered and colored, according to the clustering partition found by TLB-LDA followed by k-means.

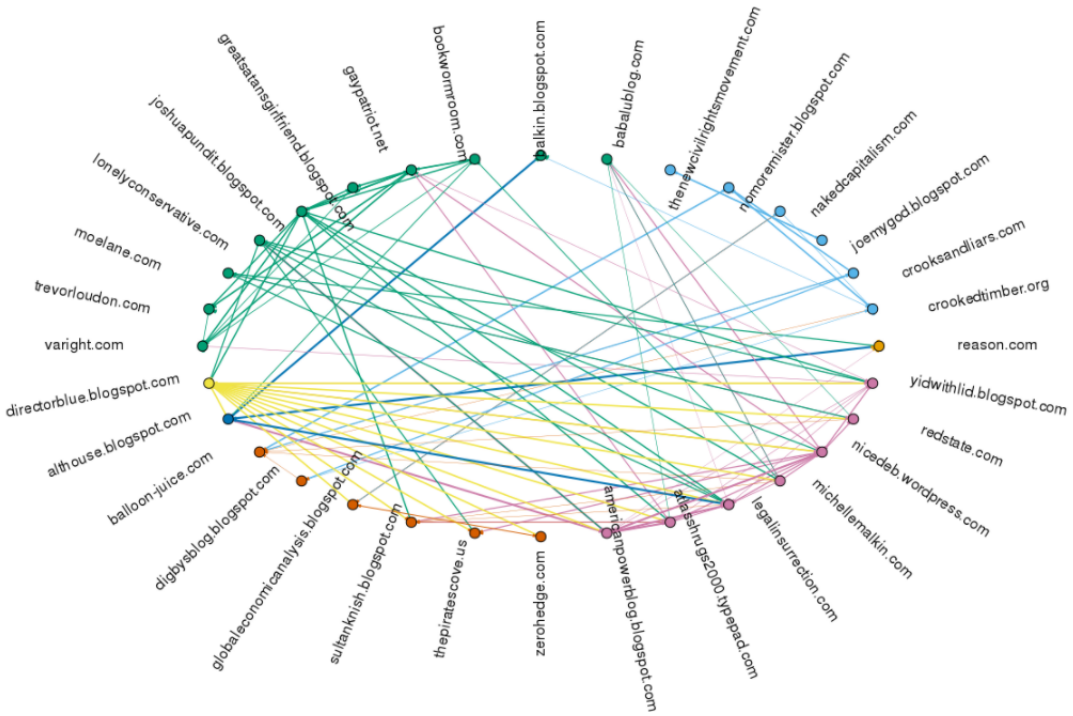


FIGURE 4.2: Cluster Assignments with TLB-LDA and k-means

Two largest clusters are colored in turquoise and purple, respectively. Five blogs in the first are: *gaypatriot*, *greatsatansgirlfriend*, *joshuapundit*, *trevorloudon*, and *variright*. Three blogs in the second cluster are: *nicedeb*, *americanpowerblog*, and *legalinsurrection*. It turns out four of the five blogs in the first cluster and all three in the second are identifiable as conservative. The last is NA. I compare these clusters with those found by MMSB later.

4.5.4 Model Validation

To validate the topic model, we compare with the RTM. The RTM was run for 1000 iterations using 50 topics. Below we display top words for a selection of topics and notice the similarity to the topics discovered in our analysis. The similarity is reasonable, as the topics in both models arise from an LDA based foundation.

Table 4.5: Topics Learned by the RTM

	Education	Court	Energy	Employment	Marriage	Climate
1	school	court	oil	jobs	marriage	climate
2	students	state	gas	unemployment	women	global
3	education	law	energy	rate	gay	warming
4	schools	supreme	environmental	numbers	state	change
5	college	case	pipeline	labor	sex	year
6	class	states	prices	number	rights	extreme
7	student	courts	coal	job	children	asia
8	teacher	texas	natural	people	couples	countries
9	law	district	gasoline	million	law	world
10	teachers	judge	price	force	men	emissions

To validate the network model, we compare the node clusters partition to that found using the node block assignments matrix from the MMSB. To fit the MMSB we use an R implementation of collapsed Gibbs sampling given by Chang and Chang (2010). We find the asymmetric adjacency matrix by aggregating over all posts and assigning $A_{ij} = 1$ when node i links to node j at least once in the corpus. The relevant output of the Gibbs sampling is a matrix of node block counts from the last iteration. Figure 4.3 shows the network of nodes, re-ordered and colored, according to the clustering partition found by the MMSB model.

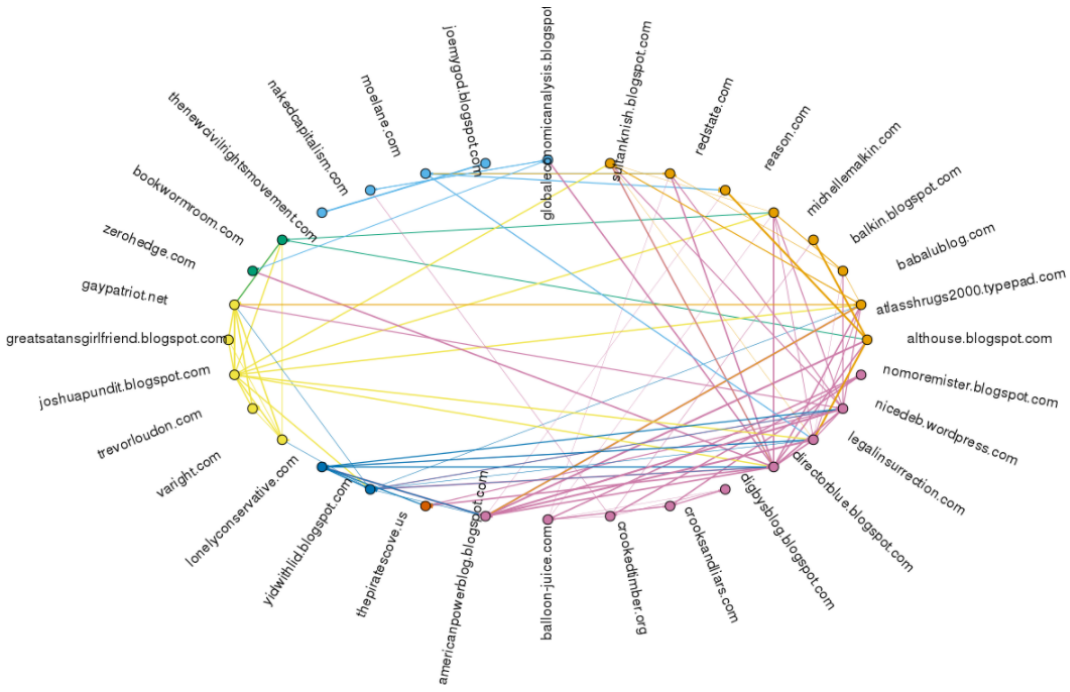


FIGURE 4.3: Cluster Assignments with MMSB

The results are similar to those found by TLB-LDA and k-means as follows: the five blogs from cluster one are in a cluster, and the three blogs from cluster two are in a cluster. However, while TLB-LDA and k-means also put *redstate*, *michellemalkin*, and *atlasshrugs2000* in cluster two, MMSB split them into their own cluster. We emphasize a major difference between the two approaches is that the blocks found by TLB-LDA include word and topic information as well as links, whereas MMSB uses only links.

4.5.5 Implementation and Computation

The model is implemented in Python. The approximate amount of time spent in (seconds per iteration) for various computations can be described as follows:

- Initializing time: 0.98 seconds
- Sampling z (vectorized probability computation and large single loop through

concatenated numpy array, sample multinomial, and reconstruct): 1.93 seconds

- Collecting c (medium double loop through array of arrays and obtain sufficient statistics): 0.61 seconds
- Sampling ϕ (small single loop of sampling updated dirichlets): 0.014 seconds
- Sampling π (medium single loop of sampling updated dirichlets): 0.031 seconds
- Sampling θ (medium single loop of sampling updated dirichlets): 0.016 seconds
- Sampling b (expensive probability computation and medium single loop through observations): 10.15 seconds
- Total Time Ellapsed: 13.73 seconds

4.6 Conclusions

The model can discover topics and clusters in a corpus of political blog posts with links. Results are comparable to those provided by LDA, RTM, and MMSB. A Gibbs sampler is derived to generate posterior samples of model parameters and latent variables. The inference algorithm is amenable to vectorization and parallelization which can yield significant speed-ups. Top words of selected topics are displayed, and discovered communities are visualized.

5

Learning Root Source with Marked Multivariate Hawkes Processes

5.1 Comments About the Root Source Identification Paper

This chapter is included to exemplify another application of a probabilistic model for text in social networks. Here, the goal is to learn the root source in related text-event sequences. I present work from Zhang et al. (2018a) and its supplementary material (Zhang et al., 2018b). The model specification, parameter estimation, root source probability learning algorithm, and experiments on synthetic data are developed by coauthors or in related work. Further details of their contributions are given in the Acknowledgements. My contributions are summarized in the following paragraph.

I chose and preprocessed two real world data sets for parameter estimation and performance evaluation for the method. The first data are comments from a thread about the 2016 USA presidential election returns (Pushshift, 2017). The second data are comments from a transcription (Richard Guo, 2012) of the movie *12 Angry Men* (Lumet, S. and Rose, R., 1957). I proposed and implemented baselines and compared our method with them in root source identification, presenting accuracy

and conditional log probability assessments. I implemented empirical Bayes priors on two model parameters and modified the parameter estimation algorithm and implementation accordingly. Additionally, I reasoned through and grid searched over hyperparameter specifications to choose values appropriate for the application. Combined, these contributions significantly improved model performance. I created a plot to visualize a source-to-source influence network. I recognized the root source probabilities can be summed to give a quantification of innovation of a source. I also contributed significantly to the abstract, introduction, related works section, and conclusion, and I proposed additional application ideas for the model. In this chapter I include rewrites of work done by coauthors, along with my work, for completeness.

5.2 Applying a Model to Networked Text-Event Sequence Data

In this section, I introduce the data structure as an instance of a dynamic text network, define the task we seek to perform, and mention some existing work.

5.2.1 Networked Text Event Sequences

Another goal in document modeling is to describe the process of document creation in time. In many cases there are multiple nodes, each associated with the creation of observational units of text data over time. Sometimes the nodes interact, and a text event on one node can be associated with an increase in the probability of future text events. Events are self exciting when they affect their own node's event rate and mutually exciting when they affect another node's event rate. As time elapses, the sequences of related events comprise a forest-like structure of multiple trees, which is sometimes called a *branching process*.

Examples of related text event sequences include comments by people in a group conversation setting, messages from multiple authors on a public web forum, and crimes committed by rival gangs in a city.

5.2.2 Learning the Root Source

One task of interest, which we call *root source identification*, is to identify the source of an event which is responsible for a cluster of subsequent events. In group conversation settings, people may wish to know who is the first to start a new topic of discussion, so this person can be acknowledged as am innovator. In online forums, people have developed a term for the person who starts a new discussion topic, calling him or her the OP, for *original poster*. Knowing the OP is useful for credit attribution and inferring power relations Danescu-Niculescu-Mizil et al. (2012). In a sequence of associated, gang-related crimes, police may want to know which gang is initially responsible.

5.2.3 Existing Work on Related Event Sequences

Early work designing and applying models for pairwise influence networks in point process data uses multivariate Hawkes processes and is in Hawkes (1971a) and Hawkes (1971b). Subsequent work has focused on dependencies in edge formation (Blundell et al., 2012; Tan et al., 2016), latent network structure (Linderman and Adams, 2014), and social influence (Zhou et al., 2013).

There is also work on causal inference in Hawkes processes. Some of this work is parametric (Xu et al., 2016; Etesami et al., 2016), and other work is non-parametric (Lewis and Mohler, 2011; Bacry and Muzy, 2014). Two papers, Guo et al. (2015) and He et al. (2015), model event sequences with text as a covariate. The first demonstrates capability of inferring influence via modeling linguistic accommodation, and the second generalizes to modeling topic inheritance in related text events.

There is a lot of work on information diffusion. This includes studying viral diffusion in Yang and Zha (2013), learning most probable diffusion pathway in Farajtabar et al. (2015a) and Rong et al. (2015), and predicting future event pathways in Du et al. (2013) and Farajtabar et al. (2015b).

Our work is distinguished because it infers influence *at the event level*, aggregates over *multiple possible pathways*, and it traces *multiple levels, back to the root.*

5.3 Multivariate Hawkes Process Model with Text Marks

We consider a sequence of text events $e_i = \{t_i, s_i, \mathbf{x}_i\}$, for $i = 1 : N$, where each event has a time-stamp t_i, node s_i, and text mark $\mathbf{x}_i$. This data can be viewed as a *branching process*, where each text event happens on a node and is either an original event or it is a response to a previous event. Figure 5.1 represents events happening on three nodes. Solid links point to original events, and dashed links point to response events. Each event's shape represents its root node. We find an event's root by tracing its links back in time until reaching an original event which we call its root. The node associated with this event is then the root node. Events rooted on node one are triangles, on node two are squares, and on node three are pentagons.

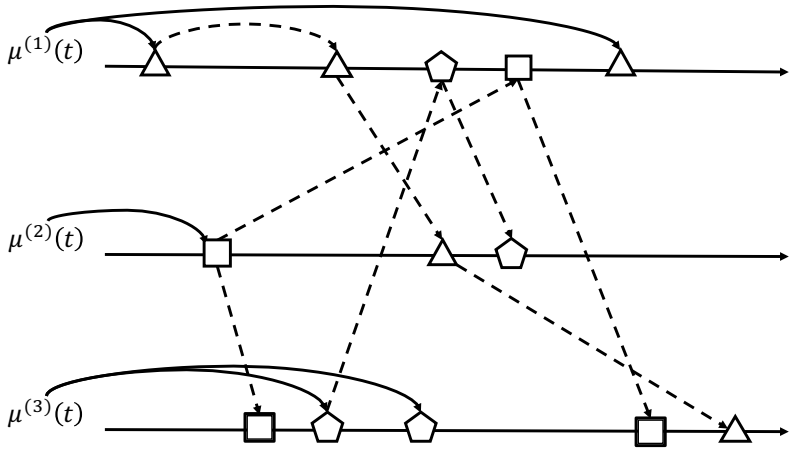


FIGURE 5.1: Structure of a Multivariate Branching Process

5.3.1 A Generative Model for Related Text Event Sequences

This section includes a probabilistic generative model for related text event sequence data as a branching process. The data generating procedure is:

Sample the Node Specific Baseline Event Rate

For each source s, sample a baseline event rate parameter:

$$\rho_s \sim \text{Gam}(a_\rho^{(s)}, b_\rho) \tag{5.1}$$

Sample Node-Pair Influences

.

For each source-pair s,s', sample an influence parameter:

$$\alpha_{s,s'} \sim \text{Gam}(a_\alpha^{(s)}, b_\alpha). \tag{5.2}$$

Specify and Compute the Source Specific Event Rate Function

For each source s:

Specify a baseline intensity shape function $\bar{\mu}^{(s)}(t)$, and compute baseline event rate function:

$$\mu^{(s)}(t) = \rho_s \bar{\mu}^{(s)}(t). \tag{5.3}$$

Specify an event specific impact $\beta(\mathbf{x}_i)$, mutual excitation shape function $\kappa^{(s)}(t_i, t)$, and compute the event response rate function:

$$\lambda_i^{(s)}(t) = \alpha_{s,s_i} \beta(\mathbf{x}_i) \kappa^{(s)}(t_i, t). \tag{5.4}$$

With a baseline rate function $\mu^{(s)}(t)$ and additional rate component $\lambda_i^{(s)}(t)$ due to each event i specified, compute the overall rate function for source s at time t:

$$\lambda^{(s)}(t|\mathcal{H}_{t-}) = \mu^{(s)}(t) + \sum_{t_i<t} \lambda_i^{(s)}(t). \tag{5.5}$$

Together the S rate functions can be written:

$$\boldsymbol{\lambda}(t) = \left(\lambda^{(1)}(t), \ldots, \lambda^{(S)}(t)\right). \tag{5.6}$$

Generate Event Sources and Time-Stamps

For each event index $i = 1, 2, \ldots$, the data are assumed to be generated as follows:

Draw event time-stamps and sources:

$$(t_i, s_i) \sim \texttt{Inhomogenous-Poisson-Process}(\boldsymbol{\lambda}(t)). \tag{5.7}$$

Sampling Event Specific Parent Indicators

Define a parent variable $\mathbf{z}_i = (z_{ij})_{j=0,\ldots,n}$ for event e_i has $z_{i0} = 1$ when e_i is an original event and $z_{ij} = 1$ when it is a response event. Sample the parent variable $\mathbf{z}_i$ from a Multinomial distribution with probabilities given by:

$$P(\mathbf{z}_i | t_i, s_i, \mathcal{H}_{t_i-}) = \begin{cases} \mu^{(s_i)}(t_i)/\lambda^{(s_i)}(t_i|\mathcal{H}_{t_i-}) & z_{i0} = 1 \\ \lambda_j^{(s_i)}(t_i)/\lambda^{(s_i)}(t_i|\mathcal{H}_{t_i-}) & z_{ij} = 1 \\ 0 & else. \end{cases} \tag{5.8}$$

Sampling the Text Mark

First draw the number of words $L_i \sim Pois(\cdot|d^{(s_i)})$. Next draw the mark $\mathbf{x}_i$ as:

$$\mathbf{x}_i \sim P(\cdot|t_i, s_i, \mathbf{z}_i, \mathcal{H}_{t_i-}) = \begin{cases} Mult(\cdot|L_i, \boldsymbol{\theta}^{(s_i)}) & z_{i0} = 1 \\ Mult(\cdot|L_i, (1-\gamma)\boldsymbol{\theta}^{(s_i)} + \gamma\tilde{\mathbf{x}}_j) & z_{ij} = 1. \end{cases} \tag{5.9}$$

5.3.2 Specifying Empirical Bayes Priors

We specify prior hyperparameters as a function of the data, letting $a_\rho^{(s)} = N_s, b_\rho = ST/(1-c), a_\alpha^{(s)} = N_s, b_\alpha = T/c$. Here N_s is the total number of events on source s, and c is the expected proportion of baseline events (original comments/posts). This could theoretically be done before seeing a data set to maintain status as a legitimate *prior* distribution. Alternatively, this can be provided by an expert in the area of

research, or one can set $a_*^{(s)} = 1$ and $b_* = 0$ to obtain updates equivalent to those without any priors.

5.4 Estimating Model Parameters

We don't know how to compute maximum likelihood estimates exactly. Therefore we use a variational approximation for the probabilities of the parent variables, assuming an independent distribution for each. Therefore, $Q(\mathbf{z}_{1:n}) \sim \prod_{i=1}^{N} Mult(\boldsymbol{\eta}_i)$.

5.4.1 The Evidence Lower Bound

Derivation of the the modified evidence lower bound (ELBO), after using the gamma priors, is:

$$
\begin{aligned}
&\tilde{\mathcal{L}}(\Theta, Q) \\
&= -\sum_s \rho_s \int_0^T \bar{\mu}^{(s)}(t)dt - \sum_s \sum_{i=1}^{n} \alpha_{s,s_i}\beta(\mathbf{x}_i) \int_{t_i}^{T} \kappa(t_i, t)dt \\
&\quad + \sum_{i=1}^{n} \eta_{i0} \log \left[\rho_{s_i} \bar{\mu}^{(s_i)}(t_i) f(\mathbf{x}_i | t_i, s_i)\right] \\
&\quad + \sum_{i=1}^{n} \sum_{j<i} \eta_{ij} \log \left[\alpha_{s_i,s_j} \kappa(t_j, t_i) f(\mathbf{x}_i | t_i, s_i, e_j)\right] \\
&\quad - \sum_{i=1}^{n} \left(\eta_{i0} \log \eta_{i0} + \sum_{j<i} \eta_{ij} \log \eta_{ij}\right) \\
&\quad + \sum_s \left[(a_\rho^{(s)} - 1) \log \rho_s - b_\rho \rho_s\right] + \sum_s \sum_{s'} \left[(a_\alpha^{(s)} - 1) \log \alpha_{s,s'} - b_\alpha \alpha_{s,s'}\right].
\end{aligned}
$$

5.4.2 Variational Expectation Maximization

The parameter estimation procedure is variational expectation maximization (EM). This involves iteratively updating model parameters in sequence until the ELBO

converges. More details about how the parameter updates were derived are given in Zhang et al. (2018a) and Zhang et al. (2018b).

1. Update $\boldsymbol{\rho}$ and $\mathbf{A}$:

$$\rho_s = \frac{a_\rho^{(s)} - 1 + \sum_{i=1}^{n} \delta_{s_i,s}\eta_{i0}}{b_\rho + \int_0^T \bar{\mu}^{(s)}(t)dt} \tag{5.10}$$

$$\alpha_{s,s'} = \frac{a_\alpha^{(s)} - 1 + \sum_{i=1}^{n}\sum_{j<i} \delta_{s_i,s}\delta_{s_j,s'}\eta_{ij}}{b_\alpha + \sum_{i=1}^{n} \delta_{s_i,s'}\beta(\mathbf{x}_i)\int_{t_i}^{T} \kappa^{(s_i)}(t_i,t)dt}. \tag{5.11}$$

2. Update $\boldsymbol{\eta}$

$$\eta_{i0} \propto \mu^{(s_i)}(t_i) f(\mathbf{x}_i | t_i, s_i) \tag{5.12}$$

$$\eta_{ij} \propto \lambda_j^{(s_i)}(t_i) f(\mathbf{x}_i | t_i, s_i, e_j). \tag{5.13}$$

3. Update $\boldsymbol{\theta}$ and γ:

$$\theta_v^{(s)} \propto \sum_i \delta_{s_i,s}[\eta_{i0}x_{i,v} + \sum_{j<i} \eta_{ij}(1 - \xi_{e_j,s})x_{i,v}] \tag{5.14}$$

$$\gamma = \frac{\sum_{i=1}^{n}\sum_{j<i}\sum_{v=1}^{V} \eta_{ij}x_{i,v}\xi_{j,v}^{(s_i)}}{\sum_{i=1}^{n}\sum_{j<i}\sum_{v=1}^{V} \eta_{ij}x_{i,v}} \tag{5.15}$$

where $\xi_{j,v}^{(s)}$ depends on previous values $\hat{\boldsymbol{\theta}}^{(s)}$ and $\hat{\gamma}$:

$$\xi_{j,v}^{(s)} = \frac{\hat{\gamma}\tilde{x}_{j,v}}{(1-\gamma)\hat{\theta}_v^{(s)} + \hat{\gamma}\tilde{x}_{j,v}}. \tag{5.16}$$

5.5 Root Probability Computation

Once model parameters are estimated, it's possible to use a dynamic-programming-like algorithm to compute root probabilities from those estimates.

Theorem: Given an S-dimensional marked MHP with sample $\mathcal{H}_T$ of size n, the rooted probability $\mathbf{r}_i$ for all $i \in [n]$ satisfy the recursive condition as follows:

$$r_i^{(s)} \propto \delta_{s_i,s} \mu^{(s)}(t_i) f(\mathbf{x}_i | t_i, s_i) + \sum_{j<i} r_j^{(s)} \lambda_j^{(s_i)}(t_i) f(\mathbf{x}_i | t_i, s_i, e_j). \tag{5.17}$$

The proof of this result is a primary contribution of my coauthors on the paper Zhang et al. (2018a), and is omited for brevity.

5.6 Real Data Experiments with MMHP and Root Source Probabilities

In this section I discuss my primary contribution to this work, applying the MMHP model, inference procedure, and root-source probability compting algorithm to two real world data sets. Given a root source probabilities $\mathbf{r}_i$, our best guess of the root source for the comment is $s_i^* = argmax_s(\mathbf{r}_{is})$, the source assigned the highest root probability. We use this method to assess root source identification performance.

5.6.1 Reddit Data

We apply the model to comment data from *reddit.com* (Pushshift, 2017), a website for social news and web content discussion. Comments are from a very popular article, on the *politics* subreddit, titled, "2016 Election Day Returns Megathread". Each comment is either a root comment, responding directly to the original article, or it is a response to a previous comment.

We begin by extracting all comments occurring on the thread from November 8 through November 15 of 2016. This interval begins when the thread opened, on election Tuesday. We consider comments from authors who comment at least 5 times. To ensure the data have a relatively constant event rate, we analyze the first 30 minutes of data before comment rates decay drastically. Comments which

respond to deleted comments are discarded. After pre-processing, there are about 750 comments from about 100 sources.

Each comment's OP (original poster), or root source, is defined as the author of its first-level parent comment. That is, in this setting the ground truth tree structure is known. We ignore the tree structure during training, but use it for model evaluation. To learn the parameters of the model and estimate root probabilities, we consider only a comment's time-stamp, author, and text.

5.6.2 Hyperparameter Specification for Reddit Data

For the baseline intensity shape function $\bar{\mu}^{(s)}(t)$ we choose a constant function defined by hyperparameter $k = 1$. Similarly, we set the mark importance $\beta(\mathbf{x}_i) = 1$. For the decay kernels $\kappa^{(s)}(t, t')$ we choose an exponential kernel $\frac{1}{\nu_s} e^{-\frac{1}{\nu_s}(t'-t)}$ and set hyperparameters $\nu_s = \nu$. To choose ν, we make a symmetry argument. Assuming comments are uniformly distributed from 0 to T_{max}, a typical comment waits approximately $T_{max}/4$ seconds before replying. We therefore set $\nu = T_{max}/4 = 450$.

5.6.3 Word Inheritance in Reddit Comments

In Figure 5.2, we show word inheritance in Reddit comments. We present comments whose root source was correctly identified by our model.

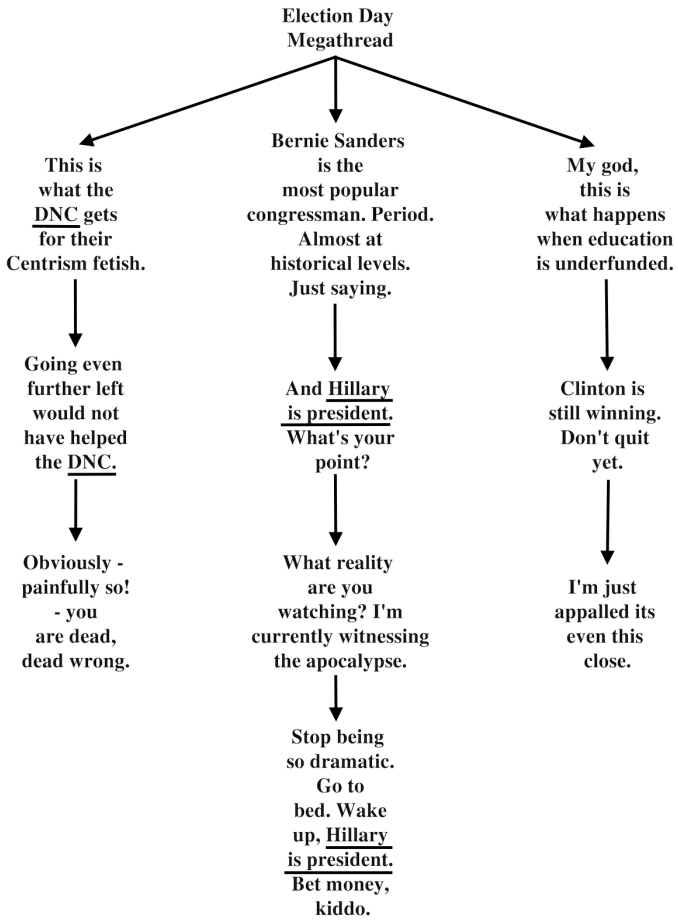


FIGURE 5.2: Word Inheritance in a Election Day Megathread Subtree of Reddit Comments

5.6.4 Assessing Performance

To our knowledge, there is no existing literature providing root source predictions in settings where tree structure is unobserved. Therefore, for model comparison, we compute the performance of a number of baseline models. RW stands for running-window, and describes a family of baselines which use previous comments' sources to form an estimate. `RW_10` assigns root probabilities proportionate to the empirical source distribution for the past 10 comments, `RW_Inf` uses observed global source proportions to estimate root probabilities, and `RW_0` assigns all probability to the

comment's own source. The `Uniform` method gives model performance under uniform random guessing. Our model is called `RP_FIT`.

We evaluate root predictions with three different scoring methods. In table 5.1 we give the multinomial log root probability, the accuracy, and the top-5 accuracy for each model.

Table 5.1: Performance of various methods.

Method	Log Prob.	Acc.	Top-10 Acc.
RP_FIT	**-852.15**	**0.74**	**0.79**
RW_Inf	-1807.84	0.04	0.30
RW_10	-2807.71	0.11	0.77
RW_0	-2300.28	0.74	0.77
Uniform	-2111.18	0.00	0.07

5.6.5 Root Source Probabilities to Quantify Innovation

Root source probabilities can be summed over events, $\sum_{i=1}^{N} r_{si}$, to quantify the *innovation* of each source s. Innovation quantifies an author's potential of exciting future comments, measuring the *indirect* impact of that author on the social network. Table 5.3 presents the five most influential Redditors ranked by innovation and compares it with the rankings using total Reddit Gold. We have replaced user names with code names. The most influential users by innovation also receive relatively high rankings by total Reddit Gold. This shows innovation can act as a proxy for popularity of an author in an online social network.

Table 5.2: Five most influential sources for Reddit data.

Rank	Username	Innovation	Gold	Gold Rank
1	User E	15.94	385	2
2	User S1	11.66	72	17
3	User S2	10.38	44	28
4	User S3	10.32	73	16
5	User R	10.20	73	15

5.6.6 *12 Angry Men Data*

In this section I describe experiments on the *12 Angry Men* data set. We consider the first approximately 600 utterances in the movie and train the model only on the timestamps, textual contents, and speaker identities. We also manually labeled the replying structure of the dataset as a human intuition reference against model outputs.

Figure 5.3 shows word inheritance in comments whose root source was correctly identified. We notice inheritance of the word "witnessess", the phrase "could they be wrong", and the word "people".

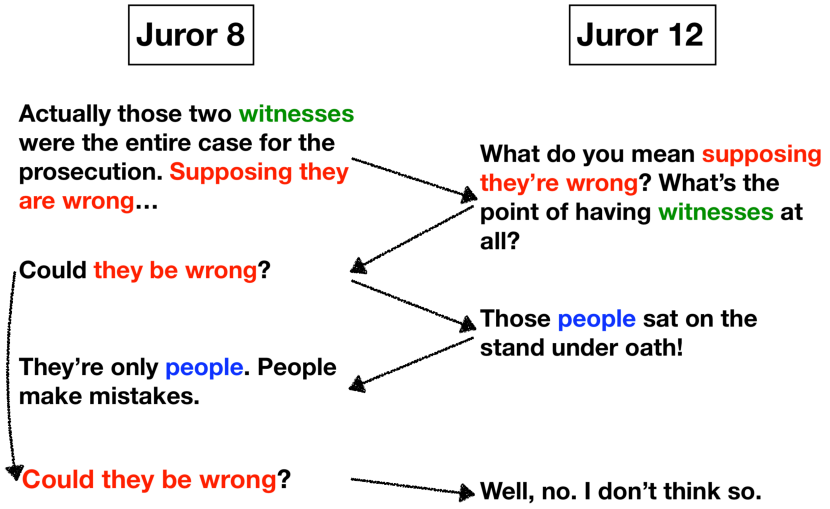


FIGURE 5.3: Word inheritance in the *12 Angry Men* conversations.

5.6.7 *Plotting the Influence Network*

One output of the analysis is the influence matrix $A = \{\alpha_{ss'}\}_{ss'}$, quantifying the impact strength of a comment by source s' on the comment rate of source s. We threshold and plot the resulting network of interactions between sources. Edge widths are proportional to influence, and vertex sizes are proportional to baseline comment

rate. Influential authors have multiple, thick outgoing edges. Figure 5.4 shows the influence network for the 12 Angry Men data.

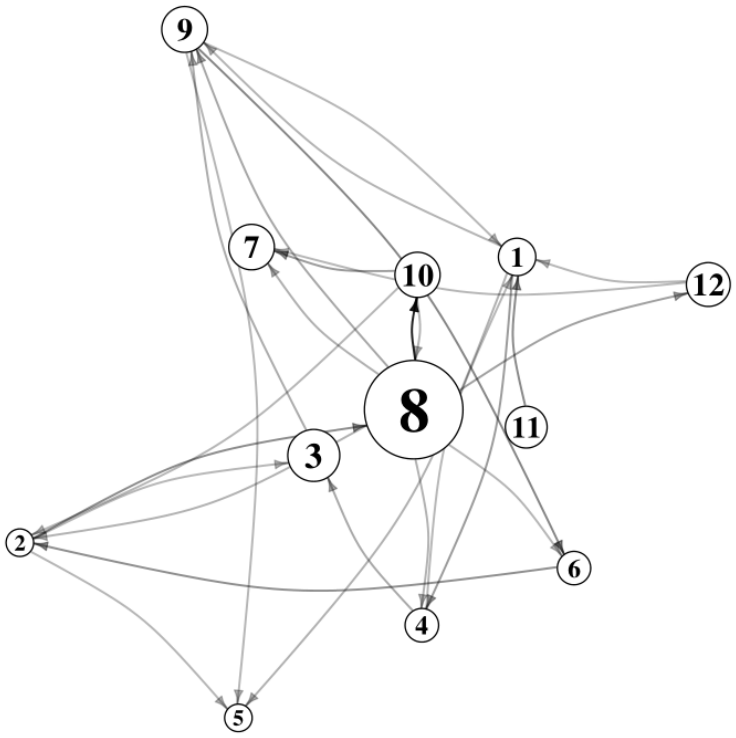


FIGURE 5.4: Influence Network of *12 Angry Men*

The model successfully identified Juror 8, the movie's protagonist, as an influential person. In the movie, he is the only juror initially voting "not guilty", and by the end, all other jurors change their votes. In Table 5.3, we sort sources by their influence.

Table 5.3: Five most influential sources for *12 Angry Men* data.

Rank	Source	Influence	Remark
1	Juror 8	269.89	Who insists "not guilty"
2	Juror 3	53.34	Who insists "guilty"
3	Juror 7	40.29	
4	Juror 1	36.99	Who serves as Foreman
5	Juror 10	35.78	

The model also is able to identify arguably the second most influential character. Juror 3 is the antagonist who is last to change his vote to "not guilty".

5.7 Conclusion

The contributions in Zhang et al. (2018a) are threefold. First it provides a model specification for simultaneous related text streams, based on a Marked Multivariate Hawkes Process. Second, it proposes a dynamic-programming-like algorithm to estimate root-source (OP) probabilities for each event. Third, it learns a source-pair influence matrix which can be used to infer power relations.

Here I discuss the main results found in the paper. On synthetic data, the model outperforms baseline models: only time-stamps, only text marks, a naive Bayes classifier, and some moving average models, in terms of root source prediction accuracy. On real data, potential value identifying influential sources who make novel contributions, when time-stamped conversation transcripts are available but structure is unobserved.

Possible extensions include studying a distribution over root-posts, rather than root-sources; learning structural information (i.e., branching structure) and train a model in a supervised manner to predict missing structural data; and modeling the changing character of a conversation as it evolves, to account for a decaying post rate as a conversation becomes outdated.

6

Discussion and Conclusion

Having dedicated much of five years to researching, designing, and applying dynamic text network models, I offer my conclusions.

6.1 Finding a Niche in the Literature Tree

My research at Duke began with a data set and a methods driven goal. We wanted to learn discussion topics on political blogs while simultaneously learning communities in which the blogs are members. A review of the vast topic modeling literature revealed some existing methods which claimed to simultaneously model links and documents. It sometimes seems there's nothing new under the sun. However, a closer look reveals the settings discussed in other papers are not *exactly* the same. Ultimately the paper in Chapter 3 has a niche combining dynamic topic modeling with community detection, and the paper in Chatper 4 has a niche of modeling documents with links to multiple nodes.

6.2 Preprocessing Matters

In a bag of words topic model, the quality of the input is related very directly to the quality of the output. Specifically, a topic can only be comprised of constituent tokens of the chosen dictionary. For this reason, eliminating characters we don't want to see in the results is an important part of the procedure. This involves removing stopwords, other uninformative words, and rare words. Additionally, it can be improve the quality of the resulting topics by n-gramming the corpus to identify significant multiword phrases. This helps with ambiguity and allows for the token "white" to be a high probability word in a "color" or "race" topic, but not in a "president" topic where instead "white house" is a high probability token. In short, garbage in leads to garbage out.

6.3 Common Notation and Precise Language

Common notation helps clarify the similarities of two models. For example, some topic modeling literature uses k to index topics and other uses t to index topics. This can lead to confusion when referring to a derivation or inference algorithm, for example, to obtain the form of a posterior distribution. Having read a large amount of topic modeling literature, I recommend using k for topics and t for time.

Second, referring to topics can be ambiguous, because in LDA topics have two attributes: word probabilities and proportions in each document. It is more clear to specify topic scores and topic loadings to distinguish between document topic proportions and topic word proportions.

6.4 Extending a Foundational Model

Identifying a strong foundation like LDA to build off of is helpful. In the case of the model in Chapter 4 when there are no links and every document is treated as its own

block, the model reduces to LDA. This model is known to recover recognizable topics. Therefore it is possible to adjust the parameter governing the number of blocks in our model to merge document topic proportion parameters as much as is possible. The model is designed with network visualization in mind, so that each document cluster may be defined by a shared set of topic interests and linking proportions. If a document is one of a kind, it simply gets its own cluster.

6.5 What's the Finished Product?

With the models discussed in this thesis being largely methodologically motivated, it can be a challenge to identify what the finished product will be used for. After much conversation with political scientists and commentators, it has become more clear. In the case of the DTN model for blog posts in Chapter 3, we can contribute a topic-specific list of top blogs, sorted by links receieved. Regarding the outputs of Chapter 4, the contribution is a way of recommending similar blog posts to an interested reader. For the root source identification project of Chapter 5, it required a more thorough review of the literature to expose the primary application. With this algorithm it is possible to identify and quantify innovation in a group conversation setting.

6.6 Methods Driven Experiments are Hard to Find

The goal of the root source learning project was to apply a model, estimation procedure, and novel algorithm to a real world data set. The model was initially applied to a pair of event streams with a goal of learning which Facebook posts reply to which news articles, and perhaps which news articles refer to which Facebook posts. Later we applied it to Reddit comments. This application was artificial, because the sought structure was available, though it helped for model testing. Finally, a potential application of identifying innovators in group conversation transcriptions

emerged. A potential application is to identify the initial source of viral stories. For fake news stories, this can reveal who initially is responsible for publishing the fraudulent information, so their intentions can be questioned.

Bibliography

Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008), "Mixed membership stochastic blockmodels," *Journal of Machine Learning Research*, 9, 1981–2014.

Arun, R., Suresh, V., Madhavan, C. V., and Murthy, M. N. (2010), "On finding the natural number of topics with latent dirichlet allocation: Some observations," in *Advances in Knowledge Discovery and Data Mining*, pp. 391–402, Springer.

Bacry, E. and Muzy, J. (2014), "Second order statistics characterization of Hawkes processes and non-parametric estimation," *arXiv preprint arXiv:1401.0903*, pp. 1–19.

Balasubramanyan, R. and Cohen, W. W. (2011), "Block-LDA: Jointly modeling entity-annotated text and entity-entity links," in *Proceedings of the 2011 SIAM International Conference on Data Mining*, pp. 450–461, SIAM.

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

Blei, D. M. and Lafferty, J. D. (2006), "Dynamic topic models," in *Proceedings of the 23rd international conference on Machine learning*, pp. 113–120, ACM.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003b), "Latent dirichlet allocation," *Journal of machine Learning research*, 3, 993–1022.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. (2010), "The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies," *Journal of the ACM (JACM)*, 57, 7.

Blundell, C., Beck, J., and Heller, K. A. (2012), "Modelling Reciprocating Relationships with Hawkes Processes," in *Advances in Neural Information Processing Systems*, pp. 2600–2608.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992), "Class-based n-gram models of natural language," *Computational linguistics*, 18, 467–479.

Chang, J. and Blei, D. M. (2009a), "Relational Topic Models for Document Networks." in *AIStats*, vol. 9, pp. 81–88.

Chang, J. and Blei, D. M. (2009b), "Relational topic models for document networks," in *International conference on artificial intelligence and statistics*, pp. 81–88.

Chang, J. and Chang, M. J. (2010), "Package lda," .

Csardi, G. and Nepusz, T. (2006), "The igraph software package for complex network research," *InterJournal*, Complex Systems, 1695.

Danescu-Niculescu-Mizil, C., Lee, L., Pang, B., and Kleinberg, J. (2012), "Echoes of power: Language effects and power differences in social interaction," in *Proceedings of the 21st international conference on World Wide Web*, pp. 699–708, ACM.

De Lathauwer, L., De Moor, B., and Vandewalle, J. (2000), "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, 21, 1253–1278.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990), "Indexing by latent semantic analysis," *Journal of the American society for information science*, 41, 391.

Dietz, L., Bickel, S., and Scheffer, T. (2007), "Unsupervised prediction of citation influences," in *Proceedings of the 24th international conference on Machine learning*, pp. 233–240, ACM.

Du, N., Song, L., Gomez-Rodriguez, M., and Zha, H. (2013), "Scalable Influence Estimation in Continuous-Time Diffusion Networks," *Advances in Neural Information Processing Systems*, pp. 3147–3155.

Eisenstein, J., Ahmed, A., and Xing, E. P. (2011), "Sparse additive generative models of text," .

Erosheva, E., Fienberg, S., and Lafferty, J. (2004), "Mixed-membership models of scientific publications," *Proceedings of the National Academy of Sciences*, 101, 5220–5227.

Etesami, J., Kiyavash, N., Zhang, K., and Singhal, K. (2016), "Learning network of multivariate Hawkes processes: a time series approach," in *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, pp. 1–14, AUAI Press.

Farajtabar, M., Gomez-Rodriguez, M., Zamani, M., Du, N., Zha, H., and Song, L. (2015a), "Back to the Past: Source Identification in Diffusion Networks from Partially Observed Cascades." in *AISTATS*, vol. 38.

Farajtabar, M., Wang, Y., Rodriguez, M. G., Li, S., Zha, H., and Song, L. (2015b), "Coevolve: A joint point process model for information diffusion and network co-evolution," in *Advances in Neural Information Processing Systems*, pp. 1954–1962.

Faust, K. and Wasserman, S. (1992), "Blockmodels: Interpretation and evaluation," *Social networks*, 14, 5–61.

Frank, O. and Strauss, D. (1986), "Markov graphs," *Journal of the american Statistical association*, 81, 832–842.

Geman, S. and Geman, D. (1987), "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," in *Readings in Computer Vision*, pp. 564–584, Elsevier.

Gilks, W. R., Best, N., and Tan, K. (1995), "Adaptive rejection Metropolis sampling within Gibbs sampling," *Applied Statistics*, pp. 455–472.

Griffiths, T. L., Steyvers, M., Blei, D. M., and Tenenbaum, J. B. (2005), "Integrating topics and syntax," in *Advances in neural information processing systems*, pp. 537–544.

Guo, F., Blundell, C., Wallach, H., and Heller, K. (2015), "The bayesian echo chamber: Modeling social influence via linguistic accommodation," in *Artificial Intelligence and Statistics*, pp. 315–323.

Hanneke, S., Fu, W., Xing, E. P., et al. (2010), "Discrete temporal models of social networks," *Electronic Journal of Statistics*, 4, 585–605.

Harris, Z. S. (1954), "Distributional structure," *Word*, 10, 146–162.

Hawkes, A. G. (1971a), "Point spectra of some mutually exciting point processes," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 438–443.

Hawkes, A. G. (1971b), "Spectra of some self-exciting and mutually exciting point processes," *Biometrika*, 58, 83–90.

He, X., Rekatsinas, T., Foulds, J., Getoor, L., and Liu, Y. (2015), "HawkesTopic: A Joint Model for Network Inference and Topic Modeling from Text-Based Cascades," *Proceedings of the 32 nd International Conference on Machine Learning*, 37.

Henry, T., Banks, D., Chai, C., and Owens-Oas, D. (2016), "Modeling community structure and topics in dynamic text networks," *arXiv preprint arXiv:1610.05756*.

Ho, Q., Eisenstein, J., and Xing, E. P. (2012a), "Document hierarchies from text and links," in *Proceedings of the 21st international conference on World Wide Web*, pp. 739–748, ACM.

Ho, Q., Eisenstein, J., and Xing, E. P. (2012b), "Document hierarchies from text and links," in *Proceedings of the 21st international conference on World Wide Web*, pp. 739–748, ACM.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002), "Latent Space Approaches to Social Network Analysis," *Journal of the American Statistical Association*, 97, 1090–1098.

Hoffman, M., Bach, F. R., and Blei, D. M. (2010), "Online learning for latent dirichlet allocation," in *advances in neural information processing systems*, pp. 856–864.

Hofmann, T. (1999), "Probabilistic latent semantic analysis," in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pp. 289–296, Morgan Kaufmann Publishers Inc.

Holland, P. W. and Leinhardt, S. (1981), "An exponential family of probability distributions for directed graphs," *Journal of the american Statistical association*, 76, 33–50.

Hubert, L. and Arabie, P. (1985), "Comparing partitions," *Journal of Classification*, 2, 193–218.

Krivitsky, P. N. and Handcock, M. S. (2014), "A separable model for dynamic networks," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76, 29–46.

Lafferty, J. D. and Blei, D. M. (2006), "Correlated topic models," in *Advances in neural information processing systems*, pp. 147–154.

Latouche, P., Birmel'e, E., and Ambroise, C. (2011), "Overlapping Stochastic Block Models with Application to the French Political Blogosphere," *Annals of Applied Statistics*, 5, 309–336.

Lewis, E. and Mohler, G. (2011), "A Nonparametric EM algorithm for Multiscale Hawkes Processes," *Journal of nonparametric statistics*, pp. 1–20.

Li, W. and McCallum, A. (2006), "Pachinko allocation: DAG-structured mixture models of topic correlations," in *Proceedings of the 23rd international conference on Machine learning*, pp. 577–584, ACM.

Linderman, S. W. and Adams, R. P. (2014), "Discovering Latent Network Structure in Point Process Data," *ICML*, 32, 1413–1421.

Liu, Y., Niculescu-Mizil, A., and Gryc, W. (2009), "Topic-link LDA: joint models of topic and author community," in *proceedings of the 26th annual international conference on machine learning*, pp. 665–672, ACM.

Lumet, S. and Rose, R. (1957), "Twelve Angry Men," .

Mcauliffe, J. D. and Blei, D. M. (2008), "Supervised topic models," in *Advances in neural information processing systems*, pp. 121–128.

McCallum, A., Corrada-Emmanuel, A., and Wang, X. (2005), "The author-recipient-topic model for topic and role discovery in social networks: Experiments with enron and academic email," .

McCallum, A., Wang, X., and Mohanty, N. (2007), "Joint group and topic discovery from relations and text," in *Statistical network analysis: Models, issues, and new directions*, pp. 28–44, Springer.

McNamee, P. and Mayfield, J. (2003), "JHU/APL experiments in tokenization and non-word translation," in *Comparative Evaluation of Multilingual Information Access Systems*, pp. 85–97, Springer.

Moody, J. (2004), "The structure of a social science collaboration network: Disciplinary cohesion from 1963 to 1999," *American sociological review*, 69, 213–238.

Nallapati, R. and Cohen, W. W. (2008), "Link-PLSA-LDA: A New Unsupervised Model for Topics and Influence of Blogs." in *ICWSM*.

Nallapati, R. M., Ahmed, A., Xing, E. P., and Cohen, W. W. (2008), "Joint latent topic models for text and citations," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 542–550, ACM.

Newman, M. E. J. and Girvan, M. (2004), "Finding and evaluating community structure in networks," *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 69, 026113.

Pushshift (2017), "Reddit comment data," .

Ramage, D., Hall, D., Nallapati, R., and Manning, C. D. (2009), "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pp. 248–256, Association for Computational Linguistics.

Ramos, J. (2003), "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*.

Richard Guo (2012), "12 Angry Men Data," .

Roberts, M. E., Stewart, B. M., Tingley, D., Airoldi, E. M., et al. (2013), "The structural topic model and applied social science," in *Advances in Neural Information Processing Systems Workshop on Topic Models: Computation, Application, and Evaluation*.

Robins, G., Elliott, P., and Pattison, P. (2001), "Network models for social selection processes," *Social Networks*, 23, 1–30.

Rong, Y., Cheng, H., and Mo, Z. (2015), "Why It Happened: Identifying and Modeling the Reasons of the Happening of Social Events," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15*, pp. 1015–1024.

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

Shetty, J. and Adibi, J. (2004), "Enron email data," .

Snijders, T. A. and Nowicki, K. (1997), "Estimation and prediction for stochastic blockmodels for graphs with latent block structure," *Journal of classification*, 14, 75–100.

Steinley, D. (2004), "Properties of the Hubert-Arable Adjusted Rand Index." *Psychological methods*, 9, 386.

Tan, X., Naqvi, S. A. Z., Qi, A. Y. Y., Heller, K. A., and Rao, V. (2016), "Content-based Modeling of Reciprocal Relationships using Hawkes and Gaussian Processes," in *UAI*, p. 2016.

Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. (2008), "Arnetminer: extraction and mining of academic social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 990–998, ACM.

Technorati (2012).

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2005), "Sharing clusters among related groups: Hierarchical Dirichlet processes," in *Advances in neural information processing systems*, pp. 1385–1392.

Titov, I. and McDonald, R. (2008), "Modeling online reviews with multi-grain topic models," in *Proceedings of the 17th international conference on World Wide Web*, pp. 111–120, ACM.

University of Glasgow (2017), "Stop Word List," *Computer Science and Information Retrieval*.

Valassis Digital (2012), "2012 political blog posts data," .

Wallach, H. M. (2006), "Topic modeling: beyond bag-of-words," in *Proceedings of the 23rd international conference on Machine learning*, pp. 977–984, ACM.

Wang, E., Silva, J., Willett, R., and Carin, L. (2011a), "Dynamic relational topic model for social network analysis with noisy links," in *2011 IEEE Statistical Signal Processing Workshop (SSP)*, pp. 497–500, IEEE.

Wang, E., Silva, J., Willett, R., and Carin, L. (2011b), "Dynamic relational topic model for social network analysis with noisy links," in *Statistical Signal Processing Workshop (SSP), 2011 IEEE*, pp. 497–500, IEEE.

Wang, X. and McCallum, A. (2006), "Topics over time: a non-Markov continuous-time model of topical trends," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 424–433, ACM.

Wasserman, S. and Pattison, P. (1996), "Logit models and logistic regressions for social networks: I. An introduction to Markov graphs andp," *Psychometrika*, 61, 401–425.

Xie, P. and Xing, E. P. (2013), "Integrating document clustering and topic modeling," *arXiv preprint arXiv:1309.6874*.

Xu, H., Farajtabar, M., and Zha, H. (2016), "Learning Granger Causality for Hawkes Processes," *International Conference on Machine Learning*, 48, 1717–1726.

Yang, S. and Zha, H. (2013), "Mixture of Mutually Exciting Processes for Viral Diffusion," in *Journal of Machine Learning Research*, vol. 28, pp. 1–9.

Yin, J. and Wang, J. (2014), "A dirichlet multinomial mixture model-based approach for short text clustering," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 233–242, ACM.

Zhang, W., Bu, F., Owens-Oas, D., Zhu, J., and Heller, K. (2018a), "Learning Root Source with Marked Multivariate Hawkes Processes," in *Submitted to International Joint Conference on Artificial Intelligence*, pp. 1–7.

Zhang, W., Bu, F., Owens-Oas, D., Zhu, J., and Heller, K. (2018b), "Learning Root Source with Marked Multivariate Hawkes Processes Supplementary Material," in *Submitted to International Joint Conference on Artificial Intelligence as a hyperlink*, pp. 1–4.

Zhou, K., Zha, H., and Song, L. (2013), "Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes," in *Artificial Intelligence and Statistics*, pp. 641–649.

Biography

Derek Owens-Oas was born on February 3, 1991 in Ashland, Oregon. He has earned the Bachelor of Arts in Mathematics from Pomona College, the Master of Science in Statistical Science from Duke University, and he will earn the Doctor of Philosophy in Statistical Science from Duke University.

Derek was awarded a teaching assistant of the year award in 2017 for the Bayesian Methods and Modern Statistics class. This is a graduate and undergraduate level statistical science course, numbered STA 360-602.

In his first year in the Statistical Science Ph.D. program at Duke, Derek was supported by a first year fellowship. He has also been supported by his adviser, Dr. David Banks, working as a teaching assistant, the Duke network analysis center (DNAC), a summer mentor Dr. Katherine Heller, and a pair of data expedition grants.

One project he has contributed to is currently submitted for publication, and there is a plan to submit a second. The first, Henry et al. (2016) is under revision by the Journal of Classification. The second, Zhang et al. (2018a) will likely be submitted to the conference on Empirical Methods in Natural Language Processing.

Derek has not yet determined where he will be working next. He is currently interviewing for data scientist and integration project manager roles at a pair of tech startups, Infinia ML and Sageworks.