STA663: Statistical Programming

Gibbs Sampling for Topic Link Block LDA (May 1, 2017)

Author: Derek Owens-Oas

Abstract

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. The model is verified on a data set of political blog posts from 2012. Inference uses parallelizable Gibbs sampling to simulate from the posterior conditionals for model parameters. Top words from selected topics are displayed, discovered communities are discussed, and results are compared with multiple strong baselines from the topic and network modeling literature.

1 Introduction

Many data sets exist that involve networked entities with associated text documents. Potential applications include 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.

There is a limited existing literature of probabilistic models which combine topic modeling and network modeling. One highly cited example is the Relational Topic Model of Chang and Blei (2009). This paper extends the foundational Latent Dirichlet Allocation topic modeling paper Blei et al. (2003) 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 paper is Balasubramanyan and Cohen (2011), which introduces the Block LDA model, generating connections between pairs of nodes with associated text and combining aspects of LDA from Blei et al. (2003) and the Mixed Membership Stochastic Blockmodels from Airoldi et al. (2008).

2 Topic Link Block LDA

We extend Latent Dirichlet Allocation with a novel network topic model. Rather than assigning each document it's 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 Model 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 1 below.

Table 1: Notation

$\boldsymbol{\theta}_b$	:	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

2.1 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. $w_{w,n} \sim Cat(\boldsymbol{\phi}_{z_{w,n}})$
   (c) For each link l:
      i. $y_{l,n} \sim Cat(\boldsymbol{\pi}_{b_n})$

3 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:

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

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 $\{\boldsymbol{\pi}_b\}_{b=1}^{B}$.

3.1 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:

$z_{wn}|- \sim Cat(\hat{\boldsymbol{p}}_{z_{wn}})$, with $\hat{p}_{z_{wn},k} = p(z_{wn} = k) = \frac{\phi_{k,x_{wn}} \theta_{b_n,k}}{\sum_{k=1}^{K} \phi_{k,x_{wn}} \theta_{b_n,k}}$

$b_n|- \sim Cat(\hat{\boldsymbol{p}}_{b_n})$, with $\hat{p}_{b_n,b} = p(b_n = b) = \frac{\frac{1}{B} \prod_{l=1}^{L_n} \pi_{y_{ln},b} \prod_{w=1}^{W_n} \theta_{z_{wn},b}}{\sum_{b=1}^{B} \frac{1}{B} \prod_{l=1}^{L_n} \pi_{y_{ln},b} \prod_{w=1}^{W_n} \theta_{z_{wn},b}}$

$\boldsymbol{\phi}_k|- \sim Dir_V(\hat{\boldsymbol{\alpha}}_{\phi_k})$, with $\hat{\alpha}_{\phi_k,v} = \alpha_{2,v} + \sum_{n=1}^{N} \sum_{w=1}^{W_n} 1(x_{wn} = v, z_{wn} = k)$

$\boldsymbol{\theta}_b|- \sim Dir_K(\hat{\boldsymbol{\alpha}}_{\theta_b})$, with $\hat{\alpha}_{\theta_b,k} = \alpha_{3,k} + \sum_{n=1}^{N} \sum_{w=1}^{W_n} 1(b_n = b, z_{wn} = k)$

$\boldsymbol{\pi}_b|- \sim Dir_D(\hat{\boldsymbol{\alpha}}_{\pi_b})$, with $\hat{\alpha}_{\pi_b,d} = \alpha_{1,d} + \sum_{n=1}^{N} \sum_{l=1}^{L_n} 1(b_n = b, y_{ln} = d)$

3.2 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 the Topic Link Block model

4 Application

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

4.1 Political Blog Posts

We experiment with a data set of political blog posts from 2012. Technorati provided a list of top political blogs and Max Point Interactive scraped and formatted the posts. We removed punctuation and spacing, symbols, digits, stop words from a list provided by University of Glascow, 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.

Some relevant information about the size of the data is given in Table 3 below.

Table 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`

4.2 Details of Analysis

We experimented with the first week worth 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 assigning blogs to common blocks parameters gradually update given the data. The specifications for this analysis are given in Table 3 below.

Table 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

4.3 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.

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

4.4 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. The top documents for the climate change topic and we find they are 1581, 1757, and 362. Similarly, for education and republican presidential candidates, they are 1125, 1044, and 1431, and 1708, 299, and 753, respectively.

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

To validate the network model, one could compare with the MMSB. At each iteration, the MMSB outputs the number of times each node is assigned a particular block. Similarly, given the block assignments of every observation from the Topic Link Block Model, we can aggregate over all observations on a particular node, collecting a node block count matrix. To compare the models one could compute counts $C_{d,b}^{(MMSB)}$ and $C_{d,b}^{(TLBM)}$, use this to estimate the node specific block membership $\hat{\pi}_{d,b}^{(MMSB)}$ and $\hat{\lambda}_{d,b}^{(TLBM)}$.

4.5 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
- Sampling z (vectorized probability computation and large single loop through concatenated numpy array, sample multinomial, and reconstruct): 1.93
- Collecting c (medium double loop through array of arrays and obtain sufficient statistics): 0.61
- Sampling ϕ (small single loop of sampling updated dirichlets): 0.014
- Sampling π (medium single loop of sampling updated dirichlets): 0.031
- Sampling θ (medium single loop of sampling updated dirichlets): 0.016
- Sampling b (expensive probability computation and medium single loop through observations): 10.15
- Total Time Ellapsed: 13.73

5 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, top documents are noted, and discovered communities are discussed.

References

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

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*, pages 450–461. SIAM.

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

Chang, J. and Blei, D. M. (2009). Relational topic models for document networks. In *AIStats*, volume 9, pages 81–88.

Max Point Interactive.

Technorati.

University of Glascow. Stop word list. *Computer Science and Information Retrieval.*