Latent Dirichlet Analysis for Document Topic Modelling

Xiang Li
xngli@umich.edu

Zheng Luo
zluo@umich.edu

Yan Chen
yanchenm@umich.edu

Sajan Patel
sajanptl@umich.edu

Abstract

In this project, we studied the topic latent Dirichlet allocation (LDA), a nonparametric Bayesian model for collections of discrete data such as text corpora, which was first proposed by Blei, Ng, and Jordan in 2003 [1]. We followed the original paper to implement LDA using variance inference techniques and an EM algorithm for estimating the Bayes parameters and extracting subtopic from a text corpus. In addition, we also implemented LDA using collapsed Gibbs Sampling following Griffiths and Steyvers (2004). We recreated the document modeling experiment in [1] to extract subtopics on AP dataset. It was found variational inference and Gibbs sampling generated similar perplexities and subtopics. We also compared LDA with simple unigram model (and find that ...).

1 Introduction

Here, describe the problem statement: Given a text document, model the topics of the document. (expand on that more).

1.1 Related Works

Based on Blei, briefly talk about unigram, mixture of unigram, and plsi.

2 Notation

We used similar notation as those denoted in the paper:
K = number of topics
D = number of documents
V = vocabulary size
N = number of words in total.
z_j = the j-th topic
d_i = the i-th document
w_i = the i-th word in the document

3 LDA

The latent Dirichlet allocation (LDA) model explains the generation of text documents, which can be viewed as samples from a mixture of multinomial distributions over a vocabulary of words. Each

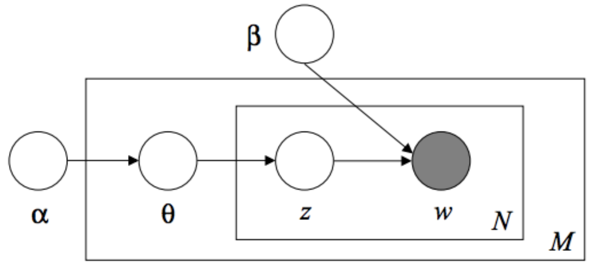

Figure 1: Graphic model of LDA. The boxes are "plates" representing replicates. The outer plate represents documents, while the inner plate represents the repeated choice of topics and words within a document.

multinomial mixture component is called a topic. The general process of the model to write a document is the following:

- The number of words N in document $\sim$ Poisson (ζ)
- The topic mixture θ for the document $\sim$ Dir (α) (with a fixed set of K topics)
- Then for each word w_i in the document:
    1. Choose a topic t_i based on multinomial distribution with parameter θ from step (2)
    2. Use this topic z_i to generate word itself by using the existing probability for each word in this topic (i.e. $p(w_i|z_i, \beta)$

This is the generative model for a collection of documents. LDA then tries to backtrack from the (training) documents to find a set of topics that are likely to have generated the collection. Now the question is how does LDA backtrack to find the parameters in this model. Suppose we have a set of documents D, and we set the number of topics to be K. What we want is to use LDA to learn two things 1) the topic representation of each document and 2) the words associated to each topic.

There are two methods to learn these two things: collapsed Gibbs sampling [1] and variational inference [2, 3]. We first discuss Gibbs sampling method here:

3.1 Gibbs Sampling Method

- For each document, randomly assign each word in the document to one of the K topics.
    1. this step gives topic representation of all the documents
    2. this step gives word distributions of all the topics
    3. since randomly assign topics to each word is very native, so we need to improve it
- For each word w_k in document d_i
    1. For each topic t_j that this word belongs to, compute:
        (a) $p(z_j|d_i) = \frac{number of words assigned to z_j in d_i}{total number of words in d_i}$
        (b) $p(w_k|z_j) = \frac{number of words assigned to z_j in d_i}{number of words assigned to z_j for all docs}$
- we compute the product of i) and ii) above which gives the new topics to assign to this word.
- repeating step 2 over and over until it reaches a steady state where the assignments make good sense.
- Use this model to estimate the topic mixtures of each document and words associated to each topic, which are the two things we want to learn.

3.2 Inference Method

With the same Dirichlet distribution model assumption shown in the previous section, variational parameters (γ and ϕ) are introduced. The way to do this is to place a distribution q over hidden

Estimation
Initializatize: $\phi_{ni}^0 = 1/K$ for all i and n
Initializatize: $\gamma_i^0 = \alpha_i + N/K$ for all i
repeat
for $n = 1$ to N **do**
  for $i = 1$ to K **do**
    $\phi_{ni}^{t+1} := \beta_{iw_n} \exp\{\varphi(\gamma_i^t) - \varphi(\sum_{j=1}^K \gamma_j^t)\}$
  end forend
end forend
$\gamma^{t+1} := \alpha + \sum_{n=1}^N \phi_n^{t+1}$
$t \leftarrow t+1$
until convergence

(a) E-step

Maximization
Update β
for $i = 1$ to K **do**
  for $j = 1$ to V **do**
    $\beta_{ij} \propto \sum_{d=1}^D \sum_{n=1}^{N_d} \phi_{dni} w_{dn}^j$
  end forend
end forend
repeat
Caculate gradient $g(\alpha^t)$ and Hessian $H(\alpha^t)$
$\alpha^{t+1} = \alpha^t - H^{-1}(\alpha^t) g(\alpha^t)$
$t \leftarrow t+1$
until convergence

(b) M-step

Alg. 1: Two algorithms

variables (θ and Z) with free parameters which are the so called variational parameters. Then, an optimization process can be performed to make the placed distribution close to the posterior in KL divergence:

$$D(q(\theta, Z|\gamma, \phi)||p(\theta, Z|W, \alpha, \beta)) = \log p(W|\alpha, \beta) - L(\gamma, \phi; \alpha, \beta)$$

where

$$L(\gamma, \phi; \alpha, \beta) = E_q[\log p(\theta, Z, W|\alpha, \beta)] - E_q[\log q(\theta, Z)]$$

However, since the KL divergence is intractable, Jensen's inequality is applied to yield an evidence lower bound. Minimizing the KL divergence is equivalent to maximizing the evidence lower bound [5]. The posterior distribution can be obtained with those variational parameters using EM algorithm.

In the maximization step, the gradient $g(\alpha^t)$ and Hessian $H(\alpha^t)$ are calculated using the evidence lower bound

$$L(\gamma, \phi; \alpha, \beta) : g(\alpha^t) \tag{1}$$

When applying LDA to training documents, the estimation and maximization steps are iteratively applied until the convergence of the evidence lower bound (or equivalently, until the convergence of perplexity $\exp\{-L/N\}$).

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

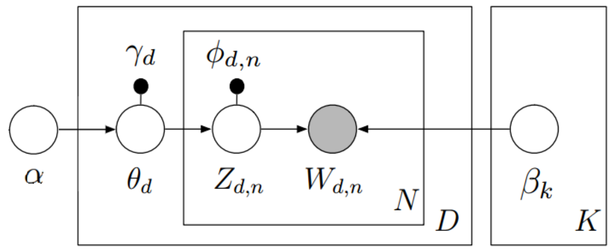

Figure 2: Graphical model representation of the variational distribution used to approximate the posterior in LDA.

4 Our Implementation

5 Evaluations and Empirical Results

5.1 Dataset and Preprocessing

To evaluate our algorithm, we replicated one of the experiments implemented by Blei et al. on document modeling. We used AP dataset which contains 2246 (verify this number) news articles from the associated press [6]. The dataset was partitioned into 80 percent training set and 20 percent test data. In preprocessing of the data, we removed a list of stop words from the text. We further removed words that appear less than 20 times (rare words), as well as words that appear in more than 90 percent of the documents (common words). We ended up with a vocabulary size of 3564.

5.2 Convergence of Algorithms

In AP dataset, each document is unlabeled. Therefore, here we are doing unsupervised learning with a purpose to estimate the likelihood of the test dataset. In natural language processing, the likelihood of a document is usually called perplexity, which is the inverse of the average per-word log likelihood. The perplexity of on the text corpus is defined by

$$\begin{aligned} perplexity(D_{test}) &= \exp\left\{-\frac{\sum_{d=1}^{M}\log p(w_d)}{\sum_{d=1}^{M}N_d}\right\} \\ &= \exp\left\{-\frac{1}{N}\sum_{d=1}^{D}\sum_{i=1}^{N_d}N_{di}\log\sum_{k=1}^{K}\theta_{w_{di},k}\phi_{d,k}\right\} \end{aligned}$$

$$\text{where, } \theta_{v,k} = \frac{C_{vk}+\beta}{\sum_{v'}C_{v'k}+\beta}, \phi_{d,k} = \frac{C_{dk+\alpha}}{\sum_{k'}C_{dk'}+K\alpha}$$

While implementing our algorithms, we first monitors the convergence of the algorithm. This was done by monitoring the perplexity (defined below) of training data, as shown Figure 3. We can see that in both cases the algorithm converges after enough number of iterations, although the numbers of iterations required to reach convergence are very different. Variational inference LDA usually converges after about 10 iteration, while Gibbs sampling LDA needs several hundred. This is a primary reason that the variational inference algorithm is much faster and more commonly used in practice.

5.3 Extracted Subtopics

We trained a 10-topic LDA model on the training dataset using both Gibbs sampling and variational inference, and 5 of the generated subtopics in shown in Figure 4. For each topic, the 10 words that are most likely to occur is listed. As we expected, LDA automatically generated meaning full subtopics from the AP corpus, with similar words more likely to appear in the same topic. We also

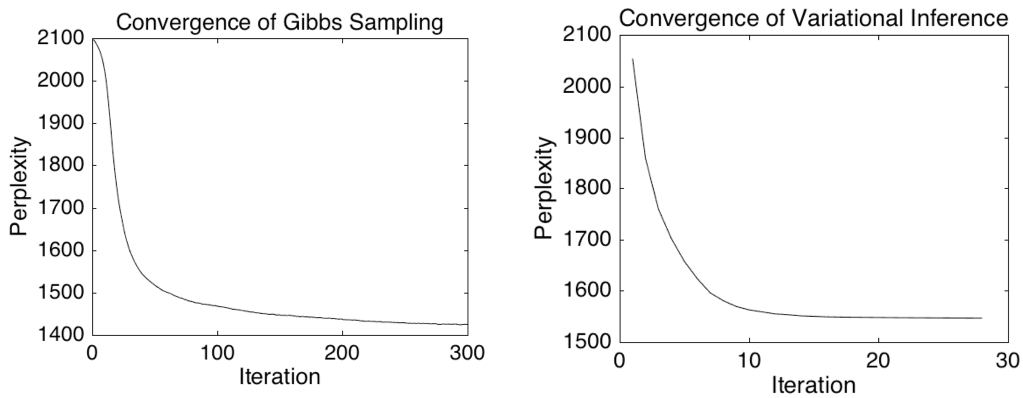

Figure 3: Convergence of LDA using Gibbs Sampling (*left*) and Variational Inference (*right*)

"Law"	**"War"**	**"Trade"**	**"Politics"**	**"Company"**
court	united	percent	government	million
state	states	market	soviet	year
federal	military	prices	party	billion
case	war	stock	union	company
department	president	dollar	south	new
law	american	trade	gorbachev	workers
attorney	iraq	year	political	based
judge	officials	late	west	last
office	aid	oil	country	corp
former	israel	higher	president	co

(a) List of topics generated by Gibbs sampling.

"Law"	**"War"**	**"Trade"**	**"Politics"**	**"Company"**
court	president	dollar	dukakis	percent
year	bush	late	bush	million
years	united	new	new	year
one	states	one	year	billion
two	government	yen	campaign	market
new	new	air	people	new
state	year	london	president	stock
people	soviet	two	state	prices
case	military	york	one	company
last	house	bid	democratic	last

(b) List of topics generated by variational inference.

Figure 4: List of topics generated by LDA using two difference algorithms.

note that the choice and order of words in the topic generated by Gibbs sampling and variational inference are different, although the topics they represented are similar. We think this discrepancy is because the two algorithms we used might converge into different local opitma, and each algorithm used (different) random initialization.

6 Conclusion

Discuss the subsequent conclusions we gained from this reimplementation of LDA. Summarize advantages and disadvantages as well.

Author Contributions

All authors contributed to the overall study of LDA and its implementation. Xiang preprocessed AP dataset in Python to generate input matrices for LDA, and implemented Gibbs Sampling LDA in MATLAB. Zheng implemented variational inference LDA in MATLAB. Sajan implemented the Unigram model in MATLAB. Yan wrote MATLAB functions for calculating perplexity and extracting subtopics from LDA outputs. All authors participated in final data analysis and interpretation, as well as writing of the manuscript.

References

[1] Griffiths, TL., Steyvers, M. "Finding scientific topics." Proceedings of the National Academy of Sciences. 101(suppl 1) (2004): 5228-35.

[2]. Jordan, Michael I., et al. "An introduction to variational methods for graphical models." Machine Learning 37.2 (1999): 183-233.

[3]. Teh, Yee Whye, et al. "Hierarchical Dirichlet Processes." Journal of the American Statistical Association 101.476 (2006).

[4]. Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent Dirichlet Allocation." the Journal of Machine Learning Research 3 (2003): 993-1022.

[5]. D. Blei and J. Lafferty. "Topic Models." In A. Srivastava and M. Sahami, editors, Text Mining: Classification, Clustering, and Applications. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, 2009.

[6]. `http://www.cs.columbia.edu/~blei/lda-c/`. It should be noted that the size of the dataset here is about one eighth of the dataset used by Blei et al. So there might be some discrepancy between our result and those presented in Blei et al.