
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:

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



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.

- The number of words N in document $\sim \text{Poisson}(\zeta)$
- The topic mixture θ for the document $\sim \text{Dir}(\alpha)$ (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 naive, 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) $p(z_j|d_i) = \frac{\text{number of words assigned to } z_j \text{ in } d_i}{\text{total number of words in } d_i}$
 - (b) $p(w_k|z_j) = \frac{\text{number of words assigned to } z_j \text{ in } d_i}{\text{number of words assigned to } z_j \text{ for all docs}}$
 - 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

Show derivation and algorithm

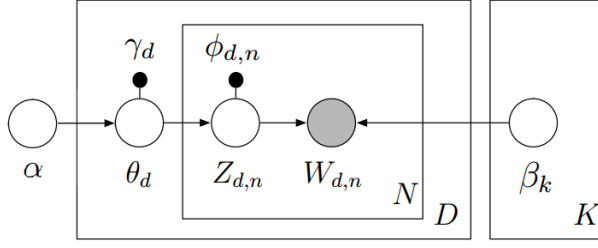


Figure 2: Graphical model representation of the variational distribution used to approximate the posterior in LDA.

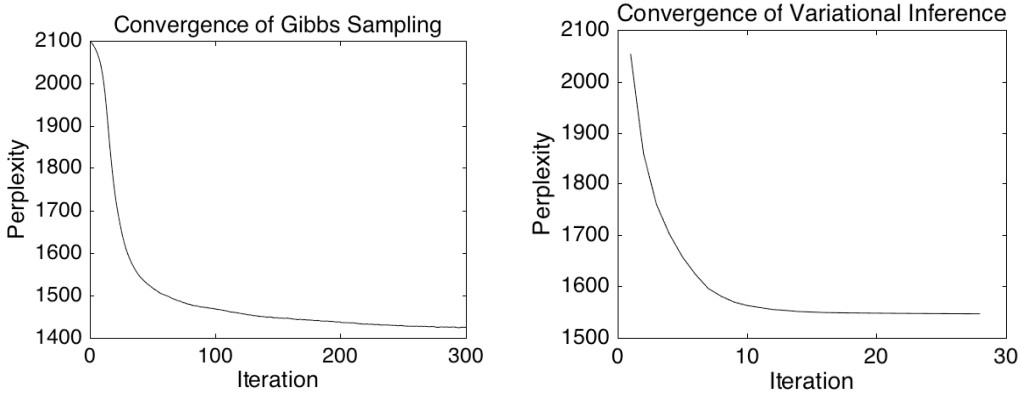


Figure 3: Convergence of LDA using Gibbs Sampling (*left*) and Variational Inference (*right*)

4 Our Implementation

5 Evaluations and Empirical Results

5.1 Dataset

To evaluate our algorithm, we replicated one of the experiments implemented by Blei et al. on document modeling. We used AP dataset which contains 2346 (verify this number) news articles from the associated press [6]. The dataset was partitioned into 80 percent training set and 20 percent test data.

5.2 Metrics and Results

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

Describe perplexity here (need help with Yan).

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.

| “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 |

Figure 4: List of topics generated by LDA.

6 Conclusion

Discuss the subsequent conclusions we gained from this reimplementaion 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] <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.