TECHNOLOGY REVIEW

CS 410 Text Information Systems Fall 2020

Latent Dirichlet Allocation

Introduction

Topic modeling is a method for unsupervised classification of documents. Topic modeling provides methods for automatically organizing, understanding, searching, and summarizing large electronic archives. It can help with discovering the hidden themes in the collection, classifying the documents into the discovered themes and using the classification to organize/summarize/search the documents.

Latent Dirichlet Allocation(**LDA**) is one of the most popular topic modeling methods. It helps in creating a type of statistical model for discovering the abstract topics that occur in a collection of documents. This was originally proposed by Blei et al. in 2003.

Body:

Assumptions and Terminology:

- Documents with similar topics will use similar groups of words
- Document definitions/modelling:
  - Documents are probability distributions over latent topics
  - Topics are probability distributions over words

This means according to LDA every document contains several topics. Each topic has a distribution of words associated with it. LDA works with probability distributions rather than strict word frequencies.

Plate notations:

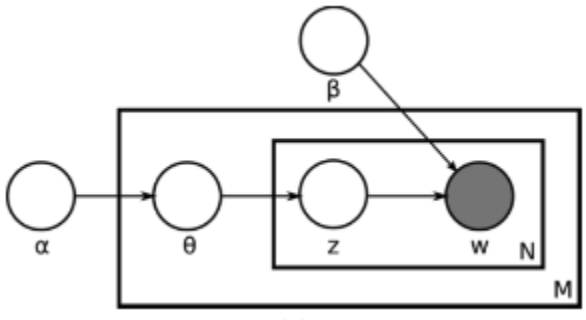

Figure 1: Plate notation representing the LDA model

Large rectangle denoted by **M** indicates the total number of documents within the corpus and the smaller rectangle **N** denotes the number of words in a document. Two parameters **α** and **β** exist outside of the rectangles are called **Dirichlet priors**.

α is Dirichlet prior on per document topic distribution. A high **α** implies each document is likely to contain a mixture of most of the topics and not just one or two. Conversely, low **α** implies each document will likely contain just a few of the topics.

β is Dirichlet prior on per topic word distribution .A high **β** implies each topic will contain a mixture of most of the words. A low **β** implies each topic may contain a mixture of just a few of the words.

$\boldsymbol{\theta}$ is topic distribution for document **M** . **z** is used to notate each topic which is assigned to each word therefore making each document a mixture of these topics. **w** is the observed word.

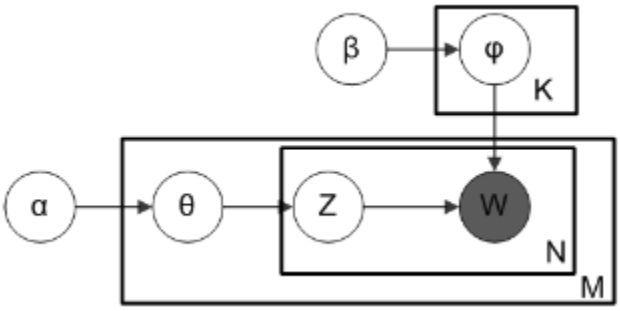

Figure 2 : Plate notation for LDA with Dirichlet-distributed topic-word distributions.

where **K** denotes the number of topics and $\boldsymbol{\varphi_1 \ldots \varphi_K}$ are **V** dimensional vectors storing the parameters of the Dirichlet-distributed topic-word distributions (**V** is the number of words in the vocabulary). It is helpful to think of the entities represented by $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$ as matrices created by decomposing the original document-word matrix that represents the corpus of documents being modeled. In this view, $\boldsymbol{\theta}$ consists of rows defined by documents and columns defined by topics,

while $\boldsymbol{\varphi}$ consists of rows defined by topics and columns defined by words. Thus, $\boldsymbol{\varphi_1 \ldots \varphi_K}$ refers to a set of rows, or vectors, each of which is a distribution over words, and $\boldsymbol{\theta_1 \ldots \theta_M}$ refers to a set of rows, each of which is a distribution over topics.

Generative process :

In order to understand how LDA assumes the documents are created we need to look at the generative process . LDA assumes new documents are created in the following way:

- Determine number of words in a document.

- Choose a topic mixture for the document over a fixed set of topics (i.e. 20% topic A , 30% topic B , 50% topic C)
- Generate the words in the document by:
    - For each of Nj words in document j
    - First pick a topic based on the document's multinomial distribution ***Mult(θj)***
    - Next, pick a word based on the topic's multinomial distribution. ***Mult(φzij)***
    - Where the parameters of the multinomials for topics in a document $\boldsymbol{\theta}_j$ and words in a topic $\boldsymbol{\varphi}_K$ have **Dirichlet priors**.
    - Iterate through this until we reach that number of words that we had specified.

Generative process example :

Consider we have a group of articles and we assume that all of those articles can be characterized by 3 topics : Animals , Cooking , Politics

Each of those topics can be described by the following words:

- Animals: dog, chicken, cat , nature ,zoo
- Cooking : oven, food ,restaurant, plates, taste ,delicious
- Politics : Republican, Democrats , Congress , ineffective , divisive

Say we want to generate a new document that is 80% about animals and 20% about cooking

1. We choose the length of the article (say 1000 words).

2. We choose a topic based on our specified mixture (800 from animals 200 from cooking).

3. We choose a word based on the word distribution for each topic.

Working backwards :

Suppose you have a corpus of documents .You want LDA to learn the topic representation of K topics in each document and the word distribution of each topic. LDA backtracks from the document level to identify topics that are likely to have generated the corpus .

1. Randomly assign each word in each document to one of the K topics .
2. For each document d:
    - Assume that all the topic assignments except for the current one are correct.
    - Calculate two proportions :
        - Proportion of words in document d that are currently assigned to topic **t = P (topic t | document d)**
        - Proportion of assignments to topic t over all documents that come from this word **w = P(word w | topic t)**

    - Multiply those two proportions and assign w a new topic based on that probability. **P(topic t | document d) * P (word w | topic t)**
3. Eventually we'll reach a steady state where assignments make sense.

After getting the topic mixtures of each of our documents in the underlying word distributions as well .We can manually add human readable labels to these various topics, but it isn't necessarily the point of LDA nor is it necessary. We can calculate similarities between articles without ever assigning labels to our topics .

Applications:

- Typically used to detect underlying topics in text documents with applications in Role discovery: Social Network Analysis (SNA) , Automatic essay grading , Anti-Phishing etc.
- Other use cases include: Sentiment analysis , Object localization for images , Automatic harmonic analysis for music , Bioinformatics etc.

Advantages :

- Effective tool for topic modeling.
- Easy to understand conceptually.
- Has shown to produce good results over many domains.
- New applications and use cases every day.

Limitations :

- Must know the number of topics K in advance.
- Dirichlet topic distribution cannot capture correlations among topics may have multiple correlated topics.

Conclusions:

- Documents are probability distributions over latent topics.
- Topics are probability distributions over words.
- LDA takes several documents .It assumes that the words in each document are related. It then tries to figure out the **"recipe"** for how each document could have been created. We just need to tell the model how many topics to construct and it uses that **"recipe"** to generate topic and word distributions over a corpus. Based on that output, we can identify similar documents within the corpus .

References:

Blei, D.M., Ng, A.Y., and Jordan, M.I., ―Latent Dirichlet Allocation ,*Journal of Machine Learning Research*, 3, 2003, 993-1022.

Rubayyi Alghamdi and Khalid Alfalqi, "A Survey of Topic Modeling in Text Mining" International Journal of Advanced Computer Science and Applications(IJACSA), 6(1), 2015. http://dx.doi.org/10.14569/IJACSA.2015.060121

https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation