DIT247 Assignment 2 Report

Patrick Andersson
Andris Freimanis
Noa Holmén
Anton Nordkvist

March 11, 2024

1 Introduction

Latent Dirichlet Allocation (LDA) is a method used in natural language processing and text mining. It is a tool for identifying and categorizing underlying themes or 'topics' within a large corpus of text documents. This method is built on the concept that every document is a blend of multiple topics. Each topics is characterized by a specific distribution of words, allowing for a probabilistic understanding of topic composition in each document.

If we consider K as the number of latent topics in the corpus, and $\phi^{(k)}$ as the probability distribution over the corpus's fixed vocabulary for the kth topic, then each topic is defined as follows:

$$\phi^{(k)} \sim \text{Dirichlet}(\beta)$$

where β is a hyperparameter for the Dirichlet distribution.

For each document d in the corpus, the model assumes the following generative process:

1. Each document d is associated with a distribution over topics, denoted as θ_d, where:
$$\theta_d \sim \text{Dirichlet}(\alpha)$$
2. For each token t_{dj} in the document d:
   (a) A topic z_{dj} is chosen from the topic distribution θ_d:
$$z_{dj} \sim \text{Discrete}(\theta_d)$$

(b) A word w_i is chosen from the word distribution of topic z_i:

$$w_i \sim \text{Discrete}(\phi^{(z_i)})$$

where α is another hyperparameter for the Dirichlet distributions.

The joint distribution of the LDA model can be expressed as:

$$p(w, z, \theta, \phi|\alpha, \beta) = p(\phi|\beta)p(\theta|\alpha)p(z|\theta)p(w|\phi, z)$$

This formulation shows how LDA considers each document as a mixture of various topics and each topic as a mixture of various words, allowing it to learn topic associations in an unsupervised manner.

We test the LDA as a tool for topic modeling in the '20 newsgroups text dataset'. Our goal is to implement Gibbs Sampling, to extract meaningful topics from this dataset.

2 Method

2.1 Data Selection and Preprocessing

As a text corpus we used "The 20 newsgroups text dataset" from the scikit-learn library[1]. In total it contains 18,000 newsgroups posts, e.g. "They were attacking the Iraqis to drive them out of Kuwait, a country whose citizens have close blood and business ties to Saudi citizens...". Our train dataset contains approximately 11,000 newsposts, resulting in a corpus of around 750,000 words. This dataset has 20 ground truth labels, e.g., `comp.graphics`, `sport.hockey` and `politics.mideast`. While these are not needed for the task at hand, it can give us an idea of which topics we can expect to be generated. We preprocessed the text by removing stop words defined by the Spacy library[2] tokenizer, removing punctuation and a list of other irrelevant tokens, e.g., `"\", "<", ">"`. We also removed words with a frequency lower than 10. After this, we used the Spacy tokenizer to tokenize the text and create a vocabulary consisting of all the unique tokens.

2.2 Gibbs sampling

Gibbs Sampling is used to approximate the posterior distribution of the latent variables, i.e., the topics. Every word starts with a random topic assignment

[1] `https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html`

[2] https://spacy.io/

and then we iterate through each word in each document, reassigning the word to a topic. The reassignment is probabilistic and based on the the document's current distribution over topics (θ_d), the word's (w_{dj}) current distribution over topics ($m_k(w_{dj})$) and the total number of words currently assigned to topics, as well as the alpha and beta hyperparameters.

Gibbs sampling is used when it is difficult to sample from the joint distribution directly, instead, we break the problem down into a series of simpler sampling steps from conditional distributions.

We decided to use the collapsed Gibbs sampling in our implementation since it is more efficient.

2.3 Implementation

To initialize the LDA model we first create a sparse document-term matrix where each row corresponds to a document and each column corresponds to the words in the vocabulary. The values are the frequencies of the words in the document. Our matrix contained 11,314 documents and 11,303 unique tokens.

We then initialize the LDA parameters (θ, ϕ) as empty matrices of the corresponding shape, and then randomly assign a topic to each word in each document and update the parameters after each topic assignment. We then perform collapsed Gibbs sampling for 100 iterations. For each iteration we iterate over every word in every document and first decrement the parameter-matrices concerning the current word's topic assignment and then compute the conditional probability q for each topic k:

$$q_k := \frac{\left(\alpha + n_d^{-dj}(k)\right)\left(\beta + m_k^{-dj}(w_{dj})\right)}{V\beta + m_k^{-dj}}, \tag{1}$$

where n_d^{-dj} is the count of words assigned to topic k in the current document d excluding the current word, $m_k^{-dj}(w_{dj})$ is the count of times topic k is assigned to the current word w_{dj} excluding current word, and m_k^{-dj} is the total count of words assigned to topic k excluding the current word. V is the total number of unique words in the vocabulary.

This is used to sample the new topic from a multinomial distribution of the topics' conditional probabilities. We then increment the parameters with the new topic and word counts.

Table 1: The 20 most common words, based on relative frequency, in three groups that subjectively seem coherent. With our suggested labels as headers.

Medical Terms	Religious Terms	Baseball Terms
infections	inspired	pitching
causing	believers	pitchers
physician	sacred	bat
diagnosis	loving	inning
eating	salvation	hitter
doctor	passages	tonight
physicians	blessed	season
treatments	writings	winning
skin	biblical	playoffs
painful	interpretations	pitches
preventing	temple	cup
commonly	apostles	pens
bacteria	doctrines	walks
severe	gospel	innings
causes	glory	batter
treat	romans	teams
healthy	condemnation	nhl
risks	teachings	penalties
therapy	reject	playoff
surgery	scriptures	baseball

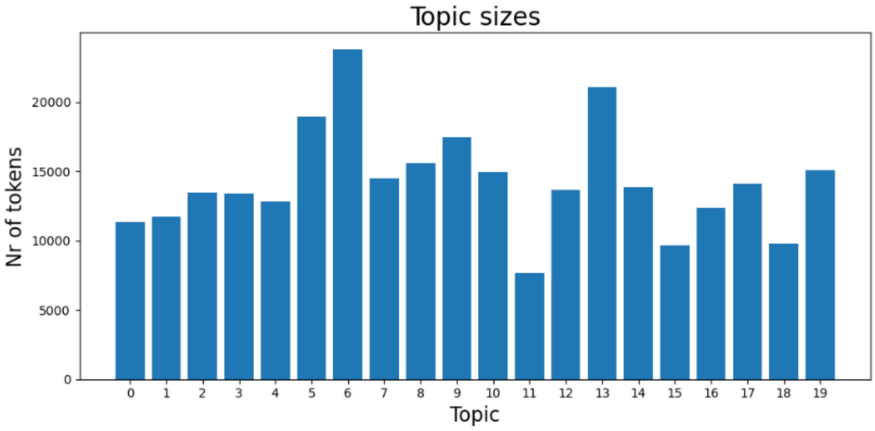
3 Results and analysis

We ran the LDA model with hyperparameters $\alpha = 0.1$, $\beta = 0.1$, and $k = 20$ for 100 iterations. To evaluate the performance we first performed a visual inspection of the words in a topic by extracting the top 20 most common words according to raw count, as well as the top 20 according to the relative frequency. The word groupings generally feel coherent, and the relative frequency tables seem to contain more specific words that are less likely to appear in other random topics. We chose the three groups that seem the most coherent according to the relative frequency and put our own semantic label on them, see table 1. The size of each topic are shown in figure 1.

We used the Umass coherence score as a measure of coherence within the topics:

$$C\left(t; V^{(t)}\right) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} \log \frac{D\left(v_m^{(t)}, v_l^{(t)}\right) + 1}{D\left(v_l^{(t)}\right)}, \tag{2}$$

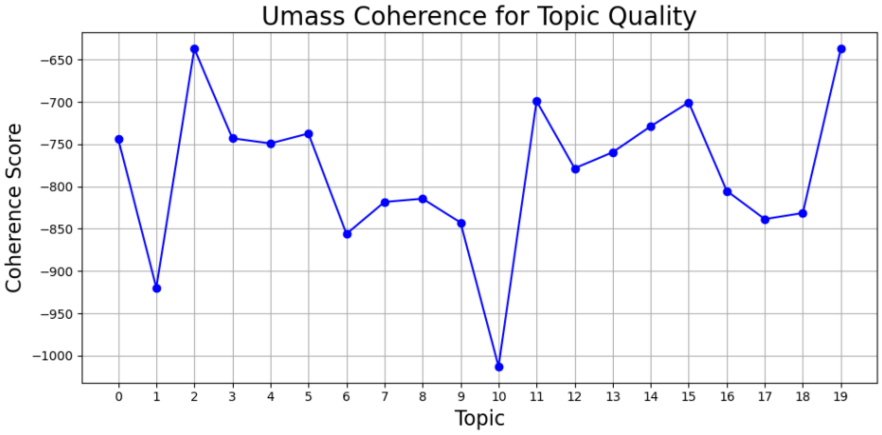
Figure 1: Topic sizes



where $V^{(t)}$ is a list of the M most probable words in topic t, and $D(v)$ is the document frequency of token v. $D(v_m, v_l)$ is the frequency of co-occurance of the tokens v_m and v_l in the same document.

The Umass coherence score assesses the quality of each topic, i.e., how coherent the topics are. A larger score indicates that the top words in the topic are more semantically related to each other, and if the top words do not frequently co-occur, the score will be lower which indicates a less coherent topic. Our scores are shown in figure 2, where our chosen topics are indexed 0, 5, and 7.

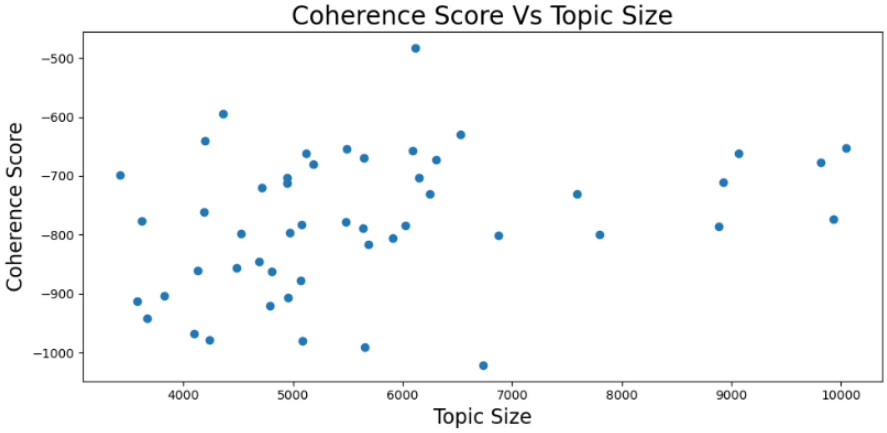
Figure 2: Umass coherence score for each topic.



4 Discussion

The paper “Optimizing Semantic Coherence in Topic Models” suggested that topic size is a good indicator of quality, therefore we decided to see if this was the case for our results as well. Since our corpus had 20 different ground truth labels, we wanted to see what happens with the same plot but running the LDA model with 50 topics. The result is plotted in figure 3.

Figure 3: Coherence score over topic size.



We can somewhat see that if a topic has a large size compared to the others, the coherence score will be higher. But a lower topic size did not necessarily mean a low coherence score.

The model extracted interpretable topics very well. These topics gave us insights about the content and structure of the dataset. Certain topics were concentrated around specific themes like medical, sport, and biblical, giving us an intuition that there is a diversity within the news documents. This does not only show the effectiveness of the model in uncovering underlying thematic structures in large text collections but also shows its potential to be used in real-world applications.

In some scenarios, such as content recommendation systems or document categorization, understanding the topics can greatly enhance the relevance in retrieval. For example, in academic research, this approach could help with quick categorization and review large volumes of literature. Also identifying gaps in the research world.

While the result of our LDA were promising, it comes with some limitations. The model’s performance depend on the choice of hyperparamters, such as the number of topics and the alpha and beta parameters which were set by intuition and never fine-tuned. It could also be useful to try with different datasets, with varying length and different styles of documents.