A Hyperparameter-Optimized Latent Topic Model

Siddharth Nand

Introduction

Topic modeling uncovers hidden thematic structures within a set of documents. It represents documents as mixtures of topics, where each topic is a probability distribution over words. This project introduces a novel topic modeling approach with a custom likelihood function and flexible parameterization. Users can adjust how word frequencies influence topics and the model's robustness to rare words.

The GitHub repository for this project can be found at https://github.com/sidnand/Hyperparameter-Optimized-Latent-Topic-Model.

Literature Review

The origins of topic models lie in Latent Dirichlet Allocation (LDA) (D. M. Blei, Ng, and Jordan 2003), a seminal work that introduced the probabilistic framework for unsupervised discovery of latent thematic structures. LDA represents documents as mixtures of probability distributions over words, where these distributions correspond to topics. Through Bayesian inference techniques, LDA estimates both the topic-word distributions (defining the vocabulary associated with each topic) and the topic proportions within each document (indicating how strongly a document relates to different topics). The Bayesian approach provides a natural way to incorporate prior beliefs, handle uncertainty, and prevent overfitting.

Building upon LDA's foundation, subsequent research has greatly enhanced the flexibility and capabilities of Bayesian topic modeling. Non-parametric models like the Hierarchical Dirichlet Process (HDP) (Teh et al. 2004) address the constraint of pre-specifying the number of topics. HDP infers an optimal number of topics directly from the data, allowing models to adapt to the inherent complexity of the corpus. Further advancements have focused on capturing the relationships between topics, such as the Correlated Topic Model (CTM) (D. Blei and Lafferty 2006), which relaxes LDA's assumption of topic independence and allows for richer representations. For corpora where the content evolves over time, Dynamic Topic Models (DTM) (D. M. Blei and Lafferty 2006) introduce a temporal dimension, modeling the evolution of topics and their changing word associations.

The power and adaptability of Bayesian topic models have led to their widespread adoption across a multitude of domains. They are invaluable tools in information retrieval and text classification (D. M. Blei, Ng, and Jordan 2003), sentiment analysis and opinion mining (Titov and McDonald 2008), social network analysis (Chang and Blei 2009), and the exploration of scientific literature (Hall, Jurafsky, and Manning 2008). Bayesian topic modeling remains an active and vibrant research area, with ongoing developments in areas such as deep generative models, topic model interpretability, and applications to new data modalities.

Problem Formulation

Data Definitions

Our analysis begins with a word-document matrix X which represents the observed word frequencies. Each entry $X_{i,j}$ denotes the number of occurrences for word i within each document j. For clarity, we'll use the following notations: m represents the number of topics, n the size of the vocabulary (number of words), and d the total number of documents within the corpus.

Model

Prior Distribution

The topic distribution for each word i is modeled using a Dirichlet distribution, $\theta_i \sim \textit{Dirichlet}(\mathbf{a})$. This distribution is ideal for topic modeling because it represents probability distributions over categories (like our topics). It ensures that topic probabilities for each word sum to 1 and allows words to have varying probabilities of belonging to multiple topics. The Dirichlet concentration parameter, $\mathbf{a}$ (a vector with one value per topic), controls the sparsity of topic distributions. Higher values of $\mathbf{a}$ lead to more even probability distributions across topics, resulting in broader topics. Conversely, lower values of $\mathbf{a}$ encourage words to concentrate their probability mass on a smaller number of topics, yielding more distinct and focused thematic groups.

Likelihood Function

The likelihood function models how likely it is to observe our word-document matrix X, given a particular set of topic distributions θ. θ is a matrix where each row corresponds to a word and each column corresponds to a topic. The core assumption is that words that appear frequently within a topic are more likely to be strongly associated with that topic. We define the likelihood function as follows:

$$\mathbb{P}(X|\theta) = \prod_{i=1}^{n} \left(\frac{\theta_{i,k} + \beta}{\sum_{k'=1}^{m} \theta_{i,k'} + \beta \cdot m} \right)^{\alpha \cdot \sum_{j=1}^{d} X_{i,j}}$$

The log-likelihood function is used for computational efficiency and numerical stability:

$$\log \mathbb{P}(X|\theta) = \sum_{i=1}^{n} \left(\alpha \cdot \sum_{j=1}^{d} X_{i,j} \right) \log \left(\frac{\theta_{i,k} + \beta}{\sum_{k'=1}^{m} \theta_{i,k'} + \beta \cdot m} \right)$$

The model includes two key parameters that control how word frequencies and topic assignments are related. The influence parameter (α) controls the influence of word frequencies on the calculation of the likelihood. The smoothing parameter (β) adds a small amount to word counts, preventing zero probabilities and making the model more robust, especially for less frequent words.

Case Study

We'll apply our hyperparameter-optimized latent topic model to a real-world dataset, uncovering hidden thematic structures in a collection of financial news headlines. We'll begin with a simple document classifier, then we'll compare our model to LDA and examine the differences. Finally we will do inference diagnostics to see the validity of our inference. Our goal is to organize the news headlines into coherent topics reflecting underlying themes. The `Stan` code for the model and data preprocessing details are provided in the appendix.

Data Preprocessing

After preprocessing our data, we create a word-document matrix, with rows representing documents, columns representing unique words, and cells indicating word frequencies in each document. Due to computational constraints, we randomly select a subset of 25 documents for analysis in our case study.

Categorizing Documents

An application to topic modeling is to identify similar documents based on their thematic contents. This can be used for document clustering or recommendation systems. Due to θ_i being a probability distribution over topics for each word, one way we can categorize a document j is if topic k has the most number of highly probable words in a document, then document j is most likely to be about topic k. The below table is a subset of the categorizations of the documents into topics. Due to the unsupervised nature of topic modeling, there is no "true" interpretation of the topics. From the below table, there does not seem to be a clear interpretation of the topics that we can infer. The below table is just a subset of the categorizations, so it could be that the full table would provide a clearer interpretation.

topic	document
1	hong kong risks global finance status covid isolation
1	australia spend hydrogen carbon capture projects
1	equity futures trading higher morning notched another record high despite surge consumer prices may
2	coronavirus vaccines weakened link infections death says scientist
2	government top fuel supplier work secure pipelines closure enters fourth day
2	stock investors celebrate red-hot five-quarter run
3	chinas banking regulator warns global asset bubble risks
3	economy expanded first quarter pointing fastest growth years
3	french daily covid infections rise

Comparison with LDA

Now we will compare our model to Latent Dirichlet Allocation (LDA) by plotting the word-topic distribution of the top 10 words which are the most probable for each topic. For our model, the hyperparameters α and β are set to 0.5 and the Dirichlet concentration parameter $\mathbf{a}$ are set to 0.5 for each topic. For LDA, we used the Gibbs sampling method. We will use 3 topics for this comparision. Figure 1 shows the word-topic distribution for our model, while Figure 2 shows the word-topic distributions for LDA. The y-axis is the probability of a word being in a specific topic. Also note that the words on the x-axis are after the stemming process (were we reduce words to their root form, e.g. "captures, capturing, captured" all become "captur"). The most noticeable difference is that the probability of a word belonging to a topic is more evenly distributed in our model compared to LDA and the value of those probabilities are higher. This is most likely due to the fact that our model only looks at the word frequencies within documents labeled with a topic, while LDA considers the entire corpus. Another difference between our models is the choice of words in each topic. Due to the unsupervised nature of topics modelling, there is no objective way to compare the quality of the categorization because both models use different liklihood functions and thus would have different interpretations of the topics.

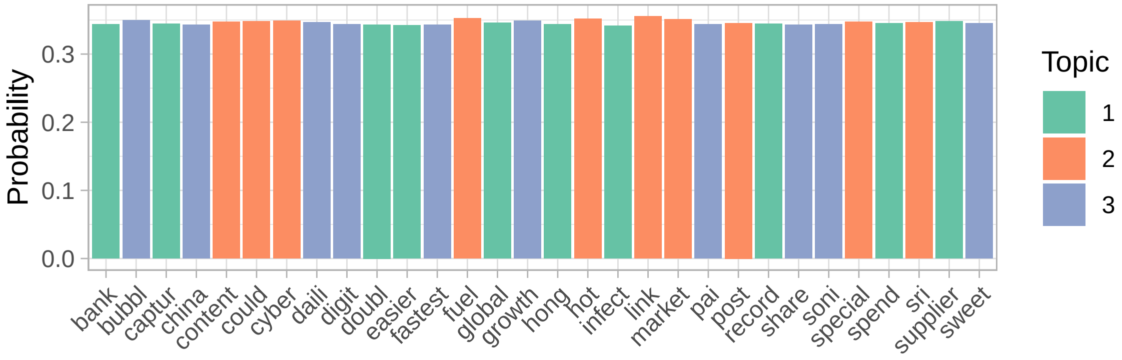

Figure 1: Hyperparameter-Optimized Word-Topic Distribution

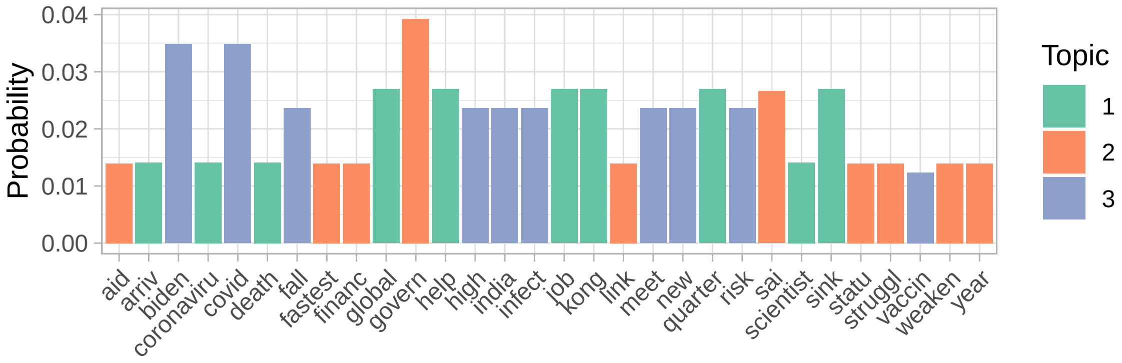

Figure 2: LDA Word-Topic Distribution

Effective Sample Size

For our model, the parameter of interest, θ is a matrix. So for each chain, we will have an effective sample size (ESS) for each parameter in the matrix. We can make a boxplot of the ESS for each parameter in θ per chain. In Figure 3, we can see that the ESS values are quite high, indicating that the chains have mixed well and that the samples are independent. This tells us that our inference is valid.

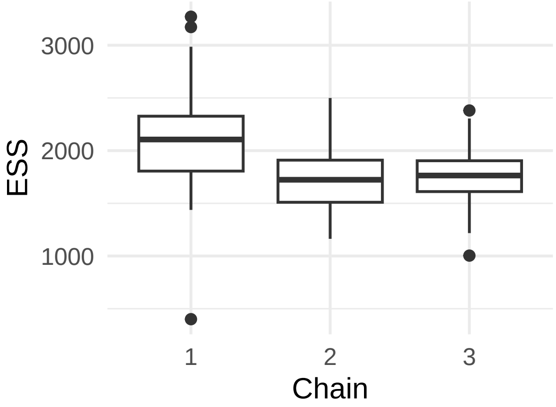

Figure 3: Effective Sample Size per Chain

Trace Plots

We can also plot the trace plots for each parameter in θ per chain. However, due to the large number of parameters, we will only plot the trace plots for the first 3 parameters in θ for each topic per chain. In Figure 4, we can see that as the number of iterations increases, the parameter values do not show any clear trend, indicating that the chains have mixed well and that the samples are independent.

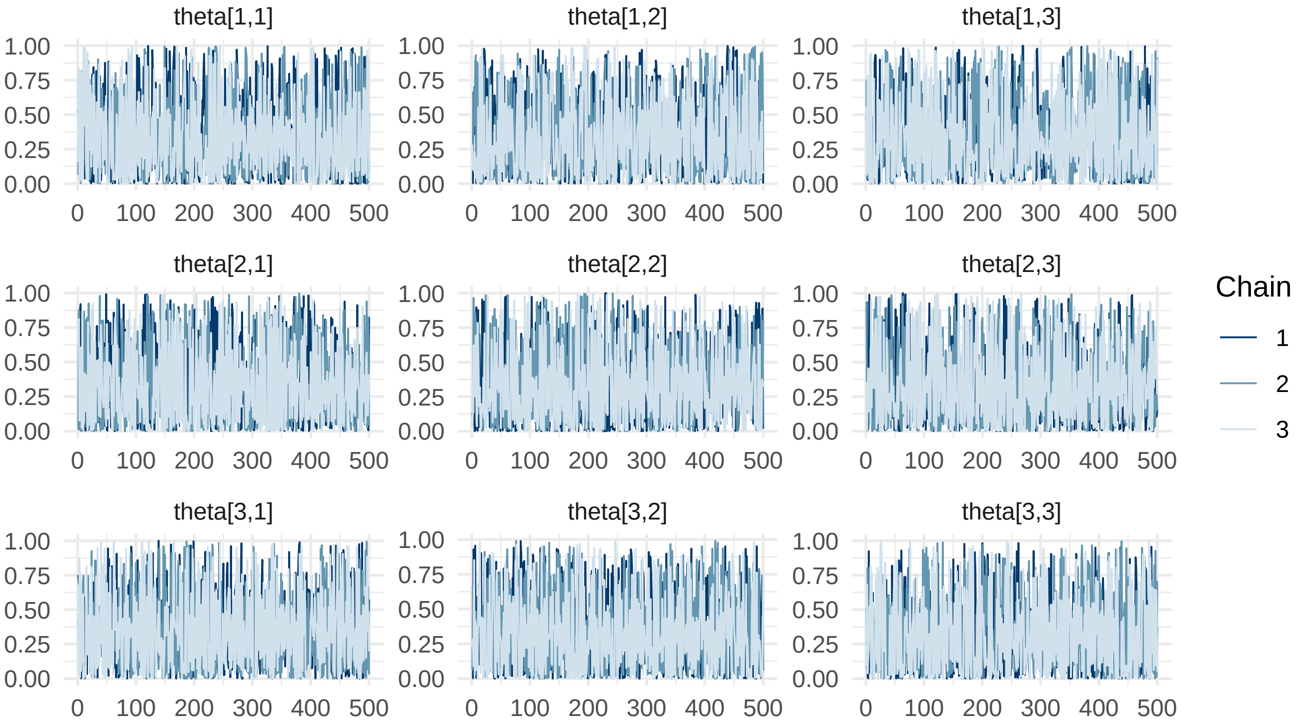

Figure 4: Trace Plots per Chain

Discussion & Conclusion

From our case study, we can see that the inference behavior of our model is good, with Figure 3 and 4 showing high effective sample sizes and well-mixed chains. The word-topic distributions in Figure 1 show that our model clusters words into topics differently than LDA, with more evenly distributed probabilities. The categorization of documents into topics in Table 1 does not provide a clear interpretation of the topics, which is expected due to the unsupervised nature of topic modeling.

Our model is not without limitations. The choice of hyperparameters α and β can significantly impact the model's performance. The Dirichlet concentration parameter **a** can also affect the sparsity of the topic distributions because it controls how words are distributed across topics. Because our parameter of interest, θ, is a matrix, the number of parameters can be quite large, which can impact the computational efficiency of the model. Fitting our data matrix of 25 documents took about 5mins as-opposed to LDA which took a few seconds. Thus our model will be difficult to scale to larger datasets. Another large limitation to our model is our liklihood function. It is quite simple and does not take into account the order of words in a document or the context in which they appear. This can lead to topics that are not coherent or meaningful.

Despite these limitations, our model offers a flexible and interpretable approach to topic modeling. By allowing users to adjust the influence of word frequencies and the robustness to rare words, our model can work on any type of textual data. Future work could focus on improving the likelihood function to capture more complex relationships between words and topics, as well as exploring more efficient computational methods to scale the model to larger datasets.

Appendix

Data Preprocessing

```r
set.seed(123)

# Load the data
data <- read_csv("data/sentiment/sentiment.csv", show_col_types = FALSE) %>%
  select(text)

# Convert data to utf-8
data$text <- iconv(data$text, to = "UTF-8")

# Clean the data
clean_data <- data %>%
  mutate(text = removeNumbers(text)) %>%
  mutate(text = removePunctuation(text,
    reserve_intra_word_contractions = TRUE,
    preserve_intra_word_dashes = TRUE,
    preserve_intra_word_underscore = TRUE
  )) %>%
  mutate(text = removeWords(text, stopwords("en"))) %>%
  mutate(text = str_replace_all(text, " ", "")) %>%
  mutate(text = str_replace_all(text, "\\s+", " ")) %>%
  mutate(text = str_replace_all(text, "’", "")) %>%
  mutate(text = str_replace_all(text, "\\b\\w{1,2}\\b", "")) %>%
  filter(text != "" | !is.na(text) | text != "NA") %>%
  mutate(text = tolower(text)) %>%
  mutate(text = str_trim(text)) %>%
  sample_n(25) %>%
  mutate(document_id = row_number())

# Tokenization and stemming
token_data <- clean_data %>%
  unnest_tokens(word, text) %>%
  mutate(word = wordStem(word))

# Create the word-document matrix with words as columns and documents as rows
X <- token_data %>%
  count(document_id, word) %>%
  cast_dtm(document_id, word, n) %>%
  as.matrix()
```

Model Code

```
data {
  int<lower=1> n;  // Number of words
  int<lower=1> d;  // Number of documents
  int<lower=0> X[d, n]; // Word-document matrix

  int<lower=1> m;  // Number of topics

  vector<lower=0>[m] a; // Dirichlet concentration parameter
  real<lower=0> alpha;  // Word frequency influence
  real<lower=0> beta;   // Smoothing parameter
}

parameters {
  simplex[m] theta[n];
}

model {
  for (i in 1:n) {
    theta[i] ~ dirichlet(a);
  }

  for (i in 1:n) {
    for (j in 1:d) {
      for (k in 1:m) {

        real w = (alpha * X[j, i]);
        real u = log((theta[i, k] + beta) / (sum(theta[i]) + beta * m));
        target += w * u;

      }
    }
  }
}
```

References

Blei, David M, and John D Lafferty. 2006. “Dynamic Topic Models.” In *Proceedings of the 23rd International Conference on Machine Learning*, 113–20.

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. “Latent Dirichlet Allocation.” *Journal of Machine Learning Research* 3 (Jan): 993–1022.

Blei, David, and John Lafferty. 2006. “Correlated Topic Models.” *Advances in Neural Information Processing Systems* 18: 147.

Chang, Jonathan, and David Blei. 2009. “Relational Topic Models for Document Networks.” In *Artificial Intelligence and Statistics*, 81–88. PMLR.

Hall, David, Dan Jurafsky, and Christopher D Manning. 2008. “Studying the History of Ideas Using Topic Models.” In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 363–71.

Teh, Yee, Michael Jordan, Matthew Beal, and David Blei. 2004. “Sharing Clusters Among Related Groups: Hierarchical Dirichlet Processes.” *Advances in Neural Information Processing Systems* 17.

Titov, Ivan, and Ryan McDonald. 2008. “Modeling Online Reviews with Multi-Grain Topic Models.” In *Proceedings of the 17th International Conference on World Wide Web*, 111–20.