Topic-Specific Analysis of Sentences Using LDA

Bhargav
21dcs022@nith.ac.in

October 2024

Abstract

Topic-specific analysis of sentences involves classifying and examining text based on predefined themes or topics. This approach enhances content analysis by enabling researchers to identify relevant information, extract insights, and analyze the context of sentences within a larger corpus, facilitating a more focused understanding of specific subjects or issues.

Topic models have been widely used by researchers across disciplines to automatically analyze large textual data. However, they often fail to automate content analysis because the algorithms cannot accurately classify individual sentences into pre-defined topics. Aiming to make topic classification more theoretically grounded and content analysis more topic-specific, I incorporated Latent Dirichlet Allocation (LDA). Taking a large corpus of speeches delivered by delegates at the United Nations General Assembly as an example, I analyzed how it can classify sentences more accurately; how it accepts pre-defined topics in deductive or semi-deductive analysis; and how it enables topic-specific framing analysis in applied research. I took practical guidance on determining the optimal number of topics from this study and selecting seed words for the algorithm. Researchers across various fields have increasingly employed topic models to facilitate the automatic analysis of extensive textual datasets.

1 Introduction

Topic modeling serves as a powerful tool for identifying latent topics in large datasets. Accurate classification of individual sentences is crucial for effective content analysis, particularly in large corpora.

2 Methodology

2.1 Data Collection

We gathered a large corpus of textual data from the speeches delivered by delegates at the United Nations General Assembly. This diverse dataset provides a rich context for topic modeling.

Datasets Used:

- Dataset 1: UN General Debates (1989-2015)
- Dataset 2: Science Articles Published in News
- Dataset 3: Environment Articles
- Dataset 4: Sports Articles

Key Themes and Topics:

- Prominent Words: The size of each word in the word cloud represents its frequency in the dataset.
- Language Use: These word clouds reflect the specific language and terminology relevant to each dataset.
- Insights: These visualizations help rapidly understand overarching themes in the datasets, facilitating data-driven decision-making.

Figure 1: Word Cloud for Dataset 1 (UN General Debates)

2.2 Preprocessing

In order to preprocess the text data for further analysis, the following steps were applied to each dataset:

1. **Tokenization**: Each text document is tokenized using the `nltk.word_tokenize` function. The text is converted to lowercase and split into individual words (tokens).

2. **Stopword Removal**: Stopwords (common, non-informative words such as "the", "and", "of", etc.) are removed using the `nltk.corpus.stopwords` corpus. Only words that are alphabetic and not in the stopword list are retained.

3. **Collocation Extraction**: Bigram collocations (frequent word pairs) are extracted using the `nltk.BigramCollocation` and the chi-squared statistic from `nltk.metrics.BigramAssocMeasures`. These collocations are then combined with the tokenized unigrams.

4. **Infrequent Token Removal**: Any tokens that occur fewer than 10 times in the entire dataset are removed to reduce noise and ensure that only frequently occurring terms are retained.

5. **Final Tokens Creation**: The final set of tokens is created by combining the unigrams (tokens) and bigrams (collocations). These tokens are stored in a new column called `final_tokens`.

6. **JSON Output**: The processed datasets, including the original columns and the new `final_tokens` column, are saved in JSON format for further analysis. Each dataset is saved in a separate JSON file, with the naming convention based on the dataset name.

2.3 Algorithm Development

LDA Model Training and Analysis

The following steps describe the process of training an LDA model on a text dataset, optimizing hyperparameters, evaluating coherence scores, and analyzing the resulting topics by comparing them to predefined seed words.

1. **Data Loading**: Four datasets are loaded from preprocessed JSON files: *UN Speeches*, *Science Articles*, *Environment Articles*, and *Sports Articles*. The datasets are processed using the `pandas` library to ensure each contains a `final_tokens` column, which holds the preprocessed tokens (words and collocations) for each document.

2. **Dictionary and Corpus Creation**: A dictionary of unique tokens is created using the `corpora.Dictionary()` function from the `gensim` library. Each token is mapped to an integer ID. The corpus, a bag-of-words representation of the dataset, is created by converting each document's tokens into a list of tuples, where each tuple contains the token ID and its frequency in the document.

3. **LDA Model Training**: An LDA model is trained using `gensim.models.LdaModel()` with the following hyperparameters:

- `num_topics = 8`: The number of topics to extract.
- `id2word = dictionary`: The token-to-ID mapping created earlier.
- `passes = 15`: The number of passes through the corpus during training.

4. **Hyperparameter Optimization**: The model is trained with varying values of three hyperparameters:

   - `r (Residual Topics)`: Number of residual topics ($r = 2$).
   - `(Seed Weight)`: The weight given to seed words in the model ($\mu = 0.02$ and 0.04).
   - `(Preceding Sentence Influence)`: Influence of preceding sentences ($\sigma = 0$ and 0.5).

   For each combination of hyperparameters, the model's coherence score is calculated to evaluate how well the topics align with the data.

5. **Coherence Score Calculation**: The coherence score for each model is calculated using the `gensim.models.Coherence` function. The best model is selected based on the highest coherence score. The best model is saved for later use.

6. **Visualization**: A comparison plot of coherence scores across different hyperparameter combinations is created and saved as `coherence_scores_comparison.png`. The best LDA model's topics are also visualized interactively using `pyLDAvis`, and the visualization is saved as an HTML file (`lda_visualization_best_model.html`).

7. **Seed Word Matching**: The top 100 words for each topic generated by the best LDA model are compared to a predefined set of seed words. These seed words are categorized into several topics, including *Greeting*, *UN*, *Security*, *Human Rights*, *Democracy*, and *Development*. Each category contains specific words that are expected to be related to the topics. The matching process checks which top words from the LDA model correspond to these seed words.

3 Results

The results demonstrate that Seeded Sequential LDA significantly improves accuracy in sentence classification compared to traditional LDA methods. For optimal number of topics K=8 ,the best coherence score was 0.3415, achieved with the parameters: residual topics $r = 2$, seed weight $\mu = 0.02$, and preceding sentence influence $\sigma = 0.5$. Topics generated from the UN speeches corpus exhibit enhanced coherence and relevance, validating the effectiveness of seed word incorporation.

3.1 Coherence Scores for different γ

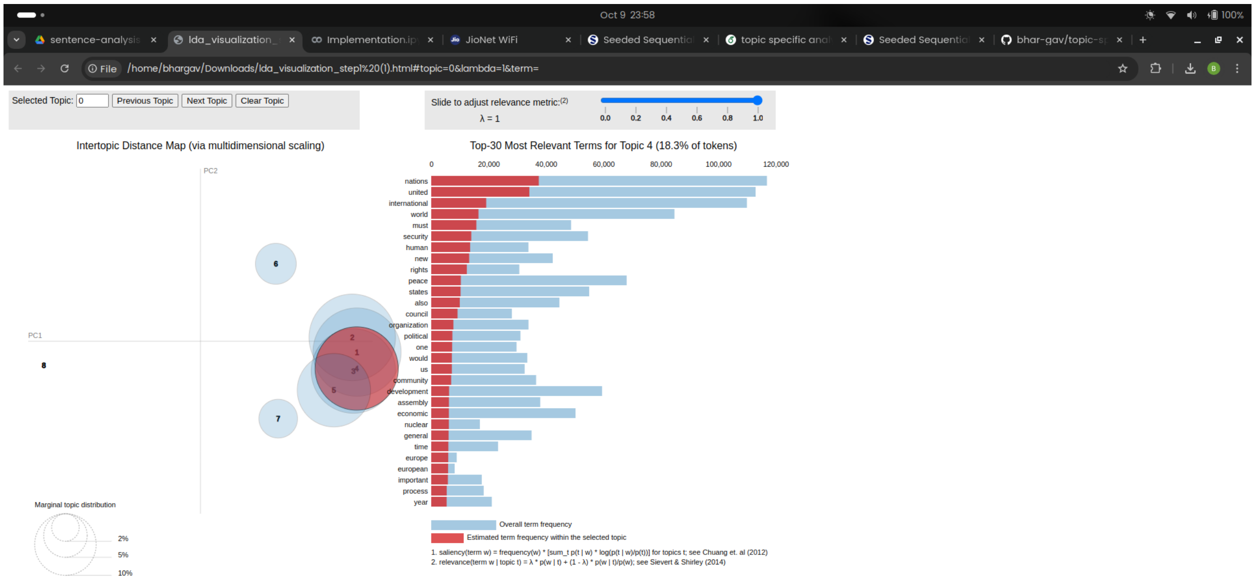

Figure 2:

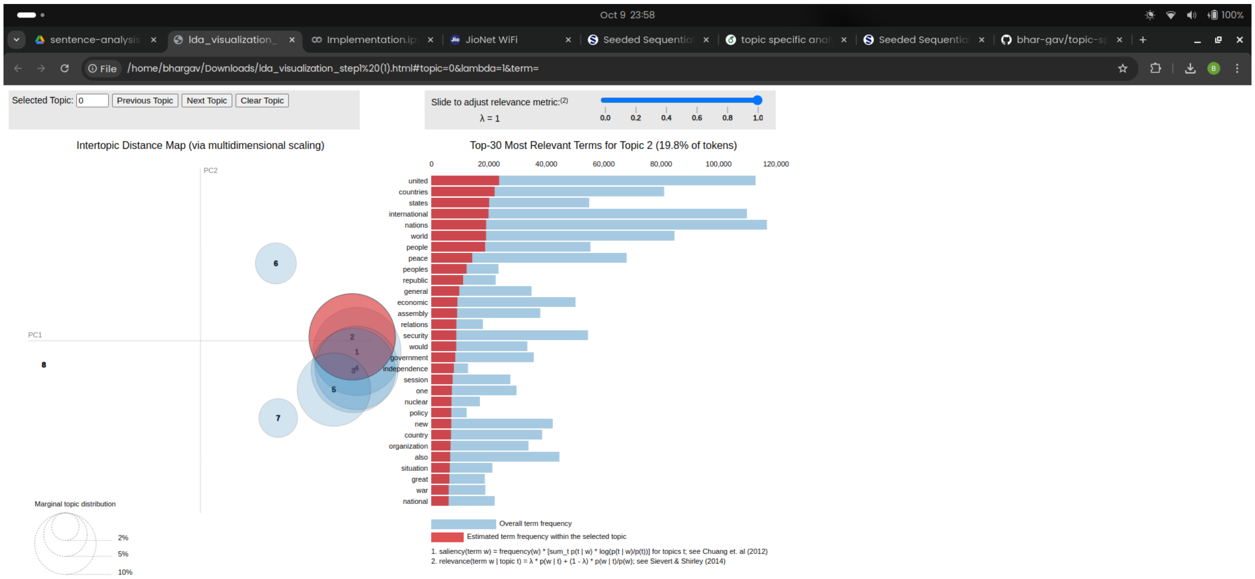

Figure 3:

3.2 Table 1: Top 100 Topic Words

Table 1: Top 100 Topic Words Identified By Non-Sequential Unseeded LDA

Index	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8
0	united	countries	world	international	nations	isis	world	international
1	states	international	nations	peace	united	kosova	us	united
2	international	nations	peace	security	international	georgians	must	world
..	...	...	....	...	...	...	...	...

3.3 Table 2: Seed Words Selected from Top 100 Words

Table 2: Seed Words Selected From the Top 100 Topic Words of A Non-Sequential Unseeded LDA Model

Index	Greeting	UN	Security	Human Rights	Democracy	Development
0	great	organization	security	community	democracy	development
1	hope	reform	peace	people	democratic	developing
2	respect	resolution	peaceful	respect	president	developed
..	...	...	....	...	...	...

4 Conclusion

LDA presents a substantial advancement in topic modeling, providing a more reliable framework for topic-specific content analysis. This method not only enhances classification accuracy but also offers practical implications.