# Document Comparison and Topic Modeling Tool

Bohao Wu
bohaowu@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

Shusen Han
shusenh2@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

Wenjie Guo
wenjie6@illinois.edu
University of Illinois at Urbana-Champaign
Urbana, Illinois, USA

## ABSTRACT

This project develops an advanced text analysis tool that integrates Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and Large Language Model (LLM) technology, specifically the Gemini-1.5-pro model, to address the challenges of token limitations in LLMs and superficial document comparisons. Designed for a diverse audience including academic researchers and content creators, the tool preprocesses texts into topic-word distributions for deeper analysis. Its robust architecture features a user-friendly interface for interaction and visualization, and sophisticated back-end processes that provide detailed thematic analyses and concise LLM-generated summaries. By enhancing text processing efficiency and depth, the tool transforms extensive textual data into actionable insights, thereby facilitating improved decision-making and research across various sectors.

## KEYWORDS

Text Analysis, Topic Modeling, Large Language Models (LLMs), Document Comparison, Semantic Analysis

**ACM Reference Format:**
Bohao Wu, Shusen Han, and Wenjie Guo. 2024. Document Comparison and Topic Modeling Tool. In *Proceedings of Advanced Information Retrieval Course Project Workshop (CS510 Spring '24)*. CS510, Urbana, IL, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

## 1 INTRODUCTION

In the evolving field of text analysis, the ability to efficiently summarize, compare, and understand large volumes of text is increasingly critical across various domains, including academia, content creation, and other compliance. Traditional tools, however, often fall short when handling extensive documents due to inherent limitations in processing capacities and analysis depth. Specifically, the constraints imposed by token limits in large language models (LLMs) restrict their ability to process long texts comprehensively, which can lead to significant information loss and reduced analytical accuracy. Moreover, existing document comparison methods typically focus on surface-level text similarities, overlooking deeper thematic connections that could provide more meaningful insights.

This report details the development of a novel tool designed to address these critical pain points. By leveraging advanced topic modeling techniques such as Probabilistic Latent Semantic Analysis (PLSA)[2] and Latent Dirichlet Allocation (LDA)[1], our tool preprocesses texts into thematic abstracts that significantly reduce the input length required for further analysis. This approach not only circumvents the token limitations of current LLMs but also enriches the quality of textual analysis by focusing on thematic essence rather than mere word frequency.

The development track of this project emphasizes three core aspects: identifying the pain point, introducing the novelty of the solution, and defining the potential users. The ensuing sections will explore each of these aspects in detail, demonstrating how our tool fills a significant gap in the market, offers innovative functionalities, and caters to the specific needs of diverse user groups seeking deeper and more accurate text analysis.

## 2 MOTIVATION

The advancement of text analysis tools has greatly enhanced our ability to parse and interpret large volumes of data. However, as the demand for deeper, more meaningful insights from text grows, existing methodologies increasingly show their limitations. This section outlines the specific pain points that our project addresses, underscoring the necessity for a new approach in text analysis.

### 2.1 Limitation of LLM Input Tokens

The widespread adoption of large language models has transformed text analysis, offering unprecedented capabilities in generating human-like text and summarizing extensive materials. Despite these advancements, a significant limitation remains: token constraints. Most LLMs, such as OpenAI's GPT series, impose strict limits on the number of tokens they can process in a single request. This limitation is a substantial barrier when dealing with long documents where crucial information might span thousands of tokens, far exceeding the maximum token allowance. The inability to process entire documents in one go leads to segmenting texts, which can disrupt the narrative flow and omit critical context, resulting in summaries or analyses that lack coherence and comprehensiveness.

### 2.2 Inadequacy in Document Similarity Measurement

Another significant challenge lies in the conventional approaches to document similarity measurements. Current tools typically employ direct text comparison techniques, such as cosine similarity of TF-IDF vectors, which primarily focus on surface-level word occurrences. This method often overlooks the implicit thematic relationships between texts, which are crucial for understanding

*CS510 Spring '24, May 07, 2024, Urbana, IL*

ACM ISBN 978-1-4503-XXXX-X/18/06
https://doi.org/XXXXXXX.XXXXXXX

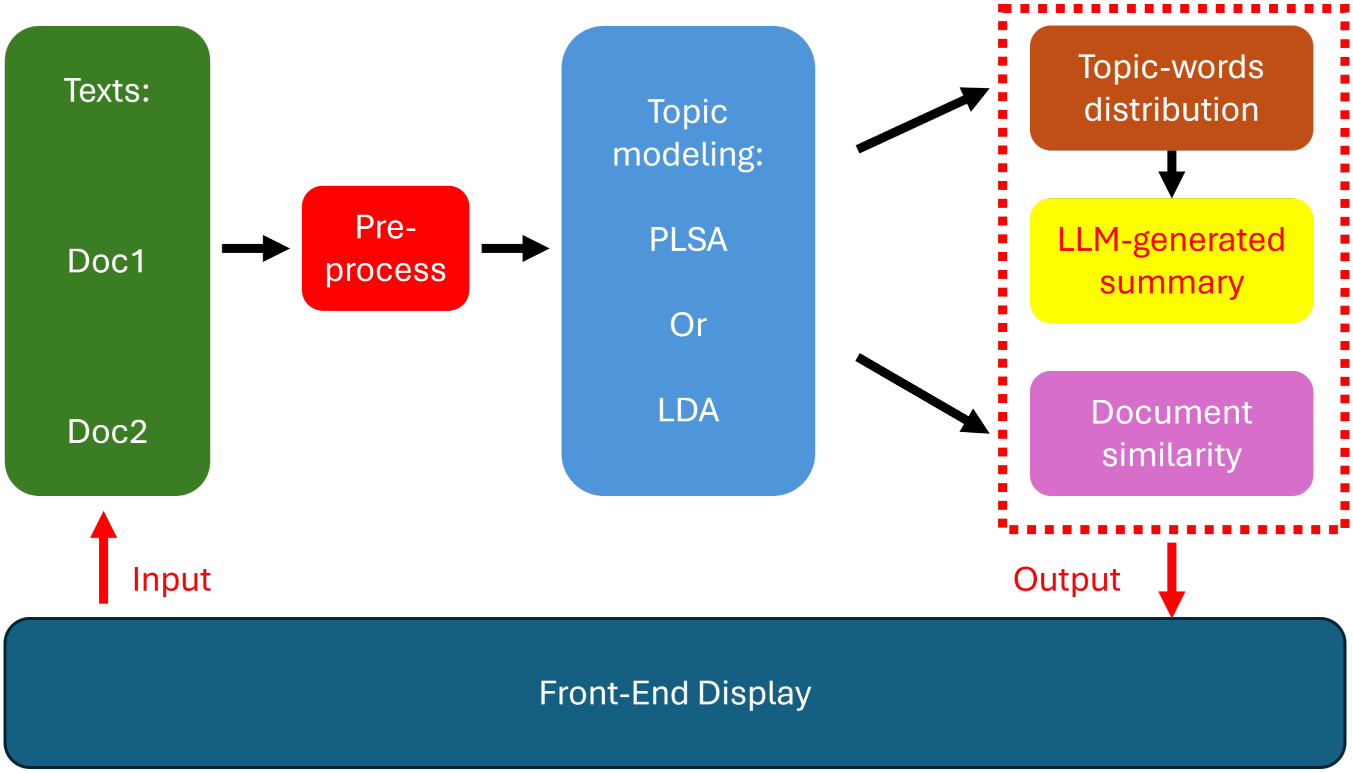

**Figure 1: Schematics of the software structure.**

nuanced similarities or differences. For instance, two documents might discuss similar themes like economic recession and financial markets using different vocabularies, leading traditional tools to underestimate their similarity.

### 2.3 Bridging Gaps in Text Analysis Across Sectors

The consequences of these limitations are felt across various sectors. Academic researchers may struggle to effectively synthesize literature, potentially overlooking vital connections or redundancies in their review processes. Content creators and marketers might fail to achieve the depth of content uniqueness and relevance required in highly competitive fields. Legal professionals and compliance officers require a high degree of accuracy and thoroughness when comparing documents to ensure adherence to standards and regulations, which current tools may not sufficiently provide.

In addressing these pain points, our tool introduces a method that not only manages the limitations of token counts in LLMs by preprocessing documents to extract their thematic essence but also enhances the document comparison process by focusing on thematic rather than lexical similarities. This approach not only meets but anticipates the needs of users in a landscape where depth and accuracy of text analysis are paramount.

## 3 STRUCTURE OF THE SOFTWARE TOOL

Our software tool is developed to address the pressing needs of comprehensive text analysis by integrating various interconnected components. As is shown in Fig 2, it spans from user interface design on the front-end to advanced processing capabilities on the back-end. This section details the functionality and integration of each component within the overall system architecture.

### 3.1 Front-end Platform

The front-end platform is developed using React combined with Django to establish a user-friendly interface on a local web server. Users interact with this interface by inputting two documents, which are immediately converted into plain text. This initial user interaction is crucial as it sets the stage for all subsequent analyses. Once processed, the results are conveyed back to the front-end where they are rendered using sophisticated visualization techniques. These visualizations are designed to make the analytical outcomes both accessible and engaging to the user.

### 3.2 Pre-processing

Pre-processing is a critical step where input documents undergo extensive cleaning and formatting to prepare them for deeper analysis. This stage is divided into two distinct methods depending on the text's linguistic features. The first method includes stopword removal using a predefined list to strip unnecessary words, followed by word tokenization and filtering with 'jieba', which is particularly

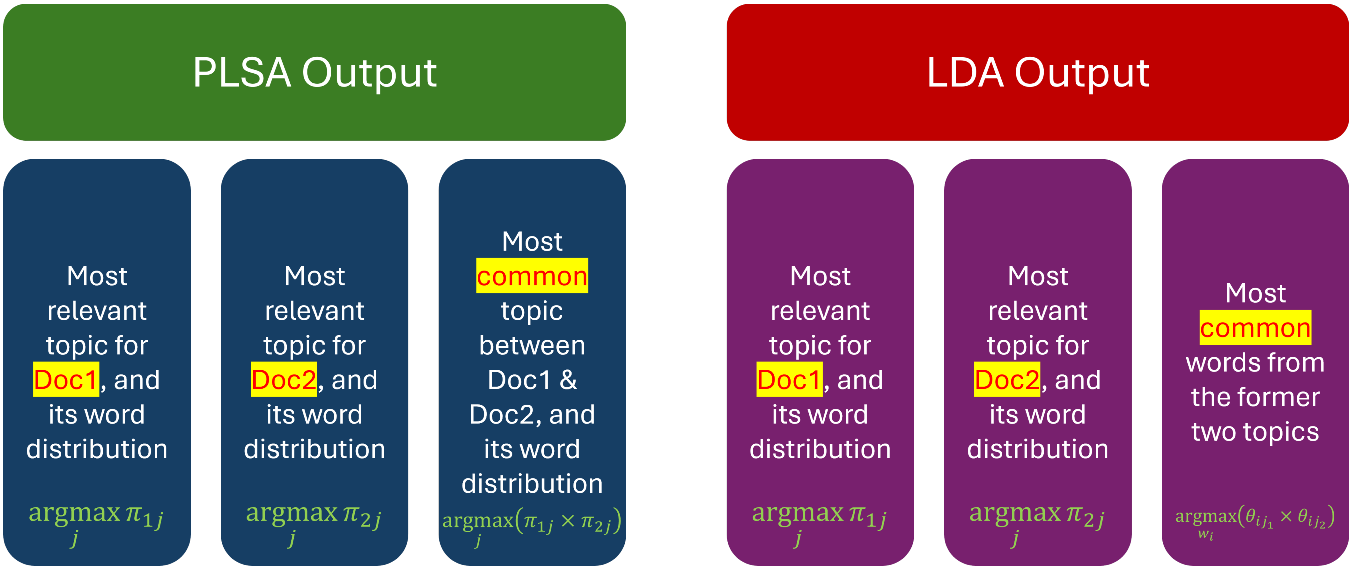
PLSA Output
LDA Output
Most relevant topic for Doc1, and its word distribution
$\underset{j}{\operatorname{argmax}} \pi_{1j}$
Most relevant topic for Doc2, and its word distribution
$\underset{j}{\operatorname{argmax}} \pi_{2j}$
Most common topic between Doc1 & Doc2, and its word distribution
$\underset{j}{\operatorname{argmax}}(\pi_{1j} \times \pi_{2j})$
Most relevant topic for Doc1, and its word distribution
$\underset{j}{\operatorname{argmax}} \pi_{1j}$
Most relevant topic for Doc2, and its word distribution
$\underset{j}{\operatorname{argmax}} \pi_{2j}$
Most common words from the former two topics
$\underset{w_i}{\operatorname{argmax}}(\theta_{ij_1} \times \theta_{ij_2})$

**Figure 2: Schematics of the software structure.**

effective for handling both English and Chinese text. This method also involves mapping each unique word to a unique identifier and counting its occurrences in the documents. The second method simplifies the process by employing the NLTK library for stopword removal and combines it with tokenization and lemmatization to reduce words to their base forms, thus standardizing different variations of the same word.

### 3.3 Topic Modeling

Following pre-processing, the documents are transformed into bags of tokens that are ready for topic modeling. Our software implements both Probabilistic Latent Semantic Analysis (PLSA)[3] and Latent Dirichlet Allocation (LDA) techniques. Users have the flexibility to select either method based on their specific needs. Both PLSA and LDA are utilized to derive the topic distribution of each document along with the distribution of words associated with each topic. This allows for a granular analysis of the documents' thematic structures.

### 3.4 LLM-generated Summary

Incorporating Large Language Models (LLMs), particularly the Gemini-1.5-pro model, we enhance our tool by generating summaries based on the most significant topics identified through topic modeling. By selecting and ranking topics, we feed the topic-word distributions into the LLM to create summaries that reflect the dominant themes using the most frequently occurring words within those topics.

### 3.5 Similarity Analysis

Moreover, our tool extends its analytical capability by incorporating a similarity analysis post-topic modeling. This feature uses the topics as an embedding model to map the documents into vector space, where the similarity between these vectors—reflective of the documents' thematic relevance—is computed. This vector-based similarity provides a quantitative measure of how the documents relate on a thematic level.

### 3.6 Feedback and Output

Finally, the outputs, including the selected topic-word distribution, the LLM-generated summaries, and the similarity metrics, are all consolidated and displayed to the user via the front-end. This comprehensive feedback mechanism allows users to gain valuable insights into the thematic content and similarities between the documents without the need to process the entire texts directly through the LLM, thus effectively managing the limitations associated with token restrictions.

## 4 OUTPUT CONTENTS

The output window of our text analysis tool provides users with intricate details about topic distributions, reflecting the most relevant information derived from the input documents. This section elucidates the technical specifics of the outputs, detailing how each output is generated and what it signifies.

### 4.1 Topics and Word Distribution

Each output is associated with the topic-word distributions from the documents processed. For the first output, the tool identifies the most relevant topic from input document 1. This is achieved by selecting the topic with the highest probability from the document's topic distribution, denoted mathematically as

$$\underset{j}{\operatorname{argmax}} \pi_{1j}$$

The output displays the word distribution for this topic, listing up to the top 20 most frequent words, although users can choose to view fewer or more words based on their preferences. A similar process is applied to document 2 for the second output, where the most relevant topic is

$$\underset{j}{\operatorname{argmax}} \pi_{2j}$$

and its word distribution is likewise ranked from the most to the less frequent words.

An important preprocessing step is the removal of English stopwords, which ensures that these outputs focus strictly on significant content words, thereby excluding any generic background topics from the topic modeling.

The third output varies depending on whether the user selects PLSA or LDA for their analysis, due to the intrinsic differences in how these models treat document relationships:

*4.1.1 PLSA.* The model treats the two documents together, allowing for a combined topic modeling. The third output, in this case, identifies the most common topic between document 1 and document 2, based on the maximum product of their respective topic distributions, calculated as

$$\underset{j}{\operatorname{argmax}} \left(\pi_{1j} \times \pi_{2j}\right)$$

This common topic's word distribution is then presented, ranking words from the most frequent to the less frequent ones.

*4.1.2 LDA.* Since LDA models each document separately, the third output does not represent a topic but rather a collection of words that are common to the most relevant topics of both documents. After determining the most relevant topics for each document, the tool calculates which words are most common across these topics, ranking them according to the product of their topic-word distributions from both topics in descending order, expressed as

$$\underset{w_i}{\operatorname{argmax}} \left(\theta_{ij_1} \times \theta_{ij_2}\right)$$

## 4.2 Document Similarity

To complement the detailed topic modeling outputs, our tool also quantifies the similarity between documents using a suite of advanced statistical and machine learning techniques. This subsection outlines the key concepts behind the similarity measures implemented in our software: BERT Similarity, Cosine Similarity, and the Pearson Correlation Coefficient.

*4.2.1 BERT Similarity.* Leveraging the capabilities of Bidirectional Encoder Representations from Transformers (BERT), this model captures deep contextual relationships between words in a text. BERT Similarity computes the semantic similarity between documents by encoding them into vector embeddings that represent their contextual meanings. This approach is particularly effective for understanding nuances in language that traditional models might overlook, such as synonyms or varied syntactic expressions that convey similar meanings. The similarity between two documents is quantified by the cosine of the angle between their BERT embeddings, calculated as follows:

$$\text{BERT Similarity } \left(D_1, D_2\right) = \frac{\text{BERT}\left(D_1\right) \cdot \text{BERT}\left(D_2\right)}{\left\|\text{BERT}\left(D_1\right)\right\| \left\|\text{BERT}\left(D_2\right)\right\|}$$

Here, *BERT* $(D_1)$ and *BERT* $(D_2)$ represent the BERT embeddings of documents $D_1$ and $D_2$ respectively, and the dot product of these embeddings is divided by the product of their norms. This formula calculates the cosine of the angle between the two embedding vectors, which corresponds to their semantic similarity.

*4.2.2 Cosine Similarity.* A mainstay in text analysis, Cosine Similarity measures the cosine of the angle between two non-zero vectors in a multi-dimensional space. In the context of our tool, it is used to calculate the similarity between the TF-IDF vectors of the documents. This metric quantifies how similar the documents are in terms of their word frequencies, providing a straightforward and effective measure of lexical similarity.

$$\text{Cosine Similarity } \left(D_1, D_2\right) = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\left\|\mathbf{v}_1\right\| \left\|\mathbf{v}_2\right\|}$$

Where $\mathbf{v}_1$ and $\mathbf{v}_2$ are the TF-IDF vectors of documents $D_1$ and $D_2$. The dot product of the vectors ( $\mathbf{v}_1 \cdot \mathbf{v}_2$ ) measures the vectors' alignment, and normalizing this by the product of the vectors' magnitudes ($\left\|\mathbf{v}_1\right\| \left\|\mathbf{v}_2\right\|$) scales the result to between -1 and 1 , representing the cosine of the angle between them.

*4.2.3 Pearson Correlation Coefficient.* While traditionally used in statistics to measure the linear correlation between two variables, in our tool, the Pearson Correlation Coefficient is applied to assess the strength and direction of a linear relationship between the representations of two documents. This coefficient ranges from -1 to 1, where 1 indicates a perfect positive linear relationship, -1 indicates a perfect negative linear relationship, and 0 signifies no linear relationship. This measure is useful for determining how changes in the word usage in one document are linearly related to changes in another, thus providing insight into their thematic alignment.

$$r_{xy} = \frac{\sum\left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sqrt{\sum\left(x_i - \bar{x}\right)^2 \sum\left(y_i - \bar{y}\right)^2}}$$

Here, $x_i$ and $y_i$ are the values of the $i$ th elements from the two sets of document data (e.g., word frequencies), $\bar{x}$ and $\bar{y}$ are the means of these datasets. The numerator calculates the covariance of the datasets, and the denominator the product of their standard deviations, resulting in a value that indicates the degree of linear correlation between the two documents.

## 4.3 LLM-Generated Summary

The last feature of our software tool is the integration of the Gemini-1.5-pro model, utilizing Google AI's API, to generate concise summaries from the topic-word distributions identified in the text analysis phase.

The summary generation process begins once the most relevant topics from the input documents have been identified and their associated word distributions have been established. These topic-word distributions, carefully selected based on their relevance and frequency, serve as the input for the Gemini-1.5-pro model. By constructing a pipeline that efficiently feeds these distributions as prompts into the LLM API, we ensure that the model has a focused context from which to generate meaningful content.

The prompts are designed to encapsulate the essence of each topic, allowing the LLM to leverage its extensive pre-trained knowledge and contextual understanding to produce summaries that are not only relevant but also rich in content. This approach harnesses the power of advanced neural network architectures inherent in the Gemini model, which is particularly adept at synthesizing large amounts of information into coherent, concise narratives.

The operation of this feature is straightforward from a user's perspective. Once the topic-word distributions are fed into the LLM:

- **Processing**: The Gemini-1.5-pro model processes the input, applying its neural network algorithms to interpret and summarize the key themes.
- **Generation**: The model generates a summary text, which is directly returned from the Google AI API.
- **Presentation**: This text is then presented to the user in the output window of our tool, offering an easily digestible summary that highlights the central themes and insights derived from the original documents.

The LLM-generated summary not only provides users with a quick understanding of the document's content but also adds value by highlighting significant information that might not be immediately apparent from a superficial reading. By integrating the LLM-generated summary into our tool, we offer a sophisticated solution that enhances user experience and provides substantial analytical depth, making complex document analysis more accessible and actionable.

## 5 POTENTIAL USERS

The versatility and depth of analysis provided by our software tool make it an invaluable resource for a broad spectrum of users. This section outlines the primary potential user groups who would benefit significantly from using our tool, ranging from academic sectors to professional environments.

### 5.1 Academic Researchers and Students

Our tool is particularly beneficial for those in academia, including researchers and students who engage with extensive document analyses, such as literature reviews, thesis research, and scholarly article comparisons. By providing quick and deep insights into document content and similarities, the tool aids in uncovering underlying themes and connections that might otherwise be overlooked, thereby enhancing the quality and efficiency of academic research.

### 5.2 Content Creators and Journalists

Writers, journalists, content marketers, and creators can use this tool to ensure the originality and depth of their content. It helps them identify common thematic elements across existing materials, facilitating the creation of unique and engaging content that stands out in a crowded media landscape. Additionally, the summarization feature allows quick understanding and reporting on complex topics, which is particularly useful in newsrooms and content-driven industries.

## 6 SUMMARY

This project develops an advanced text analysis tool designed to enhance the processing, understanding, and comparison of large textual datasets. Utilizing a combination of Probabilistic Latent Semantic Analysis (PLSA), Latent Dirichlet Allocation (LDA), and the latest Large Language Model (LLM) technology—specifically Gemini-1.5-pro provided by Google AI—our tool addresses critical limitations in current text analysis practices, particularly those related to the token limitations of LLMs and superficial document comparison methods.

At its core, the tool preprocesses input documents to distill them into concise topic-word distributions, thereby overcoming the token limit constraints of standard LLMs and ensuring more in-depth analysis. Users can choose between PLSA and LDA for topic modeling, each offering unique insights into the thematic structure of texts. The outputs include detailed topic-word distributions for individual documents, comparisons of thematic similarities, and LLM-generated summaries that highlight the most significant themes using sophisticated neural network technologies.

Targeted primarily at academic researchers, legal professionals, content creators, and business analysts, the tool is poised to revolutionize how users interact with and analyze textual data across various sectors. By providing a means to quickly and accurately extract, compare, and summarize large volumes of text, the tool not only increases efficiency but also enhances the depth and quality of textual analysis.

## REFERENCES

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.

[2] Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. 50–57.

[3] Laserwave. 2019. Probabilistic Latent Semantic Analysis Implementation. https://github.com/laserwave/plsa.

Received 07 May 2024; revised 09 May 2024