Mesgari, Nastaran

nmesgari@constructor.university

Analysis of Repetition in Teaching

Contents

[1 Executive Summery 2](#_Toc191646624)

[2 Introduction 3](#_Toc191646625)

[3 Methods of Preprocessing and finding Similarity 5](#_Toc191646626)

[3-1 TF-IDF (Term Frequency – Inverse Document Frequency) 6](#_Toc191646627)

[3-2 Latent Dirichlet Allocation (LDA) 7](#_Toc191646628)

[3-3 Process and Results of Similarity Evaluation 7](#_Toc191646629)

[3-3-1 Preprocessing: Basic Cleaning and Standardization 8](#_Toc191646630)

[3-3-2 Text Refinement: Tokenization and Bigram Extraction 9](#_Toc191646631)

[4.1.1 summarize: 9](#_Toc191646632)

[4.1.2 Reasons for High Similarities: 10](#_Toc191646633)

[4.2 Latent Dirichlet Allocation (LDA) 10](#_Toc191646634)

[4.2.1 Methodology 10](#_Toc191646635)

[4.5 Comparison Between LDA and Cosine Similarity Calculation 11](#_Toc191646636)

[4.5.1 Key Differences: 11](#_Toc191646637)

[4.5.2 Comparison of Results: 12](#_Toc191646638)

[4.5.3 Why the LDA Was Used After the Cosine Similarity Calculation 13](#_Toc191646639)

[4.6 Use Named Entity Recognition (NER): 13](#_Toc191646640)

[4.6.1 Key Features of the Code: 13](#_Toc191646641)

[4.6.2 Expected Results: 14](#_Toc191646642)

[4.15 Why Was This Update Introduced? 15](#_Toc191646643)

[4.16 Key Features of the Code: 15](#_Toc191646644)

[4.17 Function Breakdown: 16](#_Toc191646645)

[4.18 Expected Output: 16](#_Toc191646646)

[4.19 Why This Code is Useful: 16](#_Toc191646647)

[4.20 Potential Improvements/Modifications: 16](#_Toc191646648)

[4.21 Potential Improvements: 17](#_Toc191646649)

[4.22 Key Steps Taken: 19](#_Toc191646650)

[4.23 Results: 20](#_Toc191646651)

[4.24 How I Reached This: 20](#_Toc191646652)

[4.25 Explanation of the Code: 20](#_Toc191646653)

[4.26 How Similarity is Calculated: 21](#_Toc191646654)

[4.27 Result: 21](#_Toc191646655)

[4.28 How This Result is Reached: 21](#_Toc191646656)

[5 Conclusion: 21](#_Toc191646657)

[6 Suggestions for Next Steps: 22](#_Toc191646658)

[6.1 Summary: 22](#_Toc191646659)

# 1 Executive Summery

To find duplications and raise the standard of the curriculum overall, this project focuses on comparing the content of two courses: Artificial Intelligence in Business & Society and Visual Communication and Data Storytelling (*for simplicity, the course "Visual Communication and Data Storytelling" will be referred to as "Visualization," and "Artificial Intelligence in Business & Society" will be referred to as "AI" throughout the text*). Two courses were chosen: one on AI, which covered subjects like data models, algorithms, and machine learning," and the other on visualization, which emphasized graphical data representation, storytelling, and insight generation. To guarantee accurate analysis, the course content was prepared using several natural language processing (NLP) techniques. By removing stop words, digits, and special characters and applying lemmatization, the text becomes standardized, reducing noise and improving the accuracy of downstream models, and I completed my preprocessing.

After preprocessing, domain-specific keywords were identified using SEO techniques to discover related concepts on various websites*.* Depending on the reconstructed results of the course material and my understanding *(during the finding similarity, I checked out the keywords and changed them to find the best ones.)*, I selected the most relevant keywords for each subject.   
  
• AI focuses more on technology, algorithms, and models, with keywords like “ai”, “learning”, “machine”, “intelligence”, and “algorithm”.

• Visualization emphasizes conveying information and storytelling through data, with keywords such as “message”, “visualization”, “context”, “story”, “audience”, “insights”, and “experience”.

I tried to find similarities by using techniques of NLP, but there was a very high similarity. After cleaning the data, I found a minimum of about 47% similarity. This percentage is not acceptable to me because I do not think we have this similarity between these courses, so I decided to use another way to find the similarity. In this way, I calculated the similarity based on the frequency of keywords. Based on my keywords: **82 AI-related keywords and 27 visualization-related keywords** were found in the AI course, according to a keyword frequency analysis. There were **216 keywords related to visualization and 123 related to AI in the visualization course**. similarity score of **0.2433** was obtained from this overlaps, suggesting a moderate degree of redundancy between the two courses. To further show the distribution of keywords and give a visual depiction of the most common terms in the materials, word clouds were created for both courses.

The results indicate that although the courses retain their distinct, there are significant conceptual overlaps, especially in fields pertaining to data and model application. This overlap suggests possible duplications that, if resolved, could improve the uniqueness and efficacy of each course. In order to find more extensive redundancy patterns throughout the curriculum, the analysis could be extended to include more courses as a subsequent step. To learn more about content overlaps, advanced Natural Language Processing (NLP) methods like topic modeling and similarity analysis could be used. These results could guide focused content optimization tacti, guaranteeing a distinct distinction between courses and enhancing students’ overall educational experience. To improve course content, reduce duplication, and develop a more coherent curriculum, cooperation with instructors is advised.

This project lays the groundwork for future initiatives to maximize academic programs by highlighting the significance of assessing instructional materials for alignment and redundancy. University can improve learning outcomes, encourage innovation, and preserve student learning by making sure that courses are streamlined and complementary. The insights gained from this analysis are instrumental in achieving these goals and improving the quality of teaching and learning.

# 2 Introduction

In academic settings, the quality and structure of course content significantly impact students’ learning experiences and outcomes. With the growing importance of fields like Artificial Intelligence (AI) and Data Visualization, the need for well-structured and non-redundant educational programs has become increasingly critical. Redundant content, while sometimes unavoidable, can lead to inefficiencies in learning, reduce engagement, and limit students’ exposure to new concepts. Addressing such overlaps is essential for optimizing course offerings and ensuring students receive a comprehensive yet streamlined education.

This project aims to analyze and evaluate the extent of content similarities between two courses, one focusing on AI and the other on Visualization. These domains, while distinct, often intersect in their discussion of data-driven decision-making and the application of models to real-world problems. By systematically identifying overlaps, the project seeks to highlight areas of potential redundancy and offer solutions for improving course alignment.

The analysis is centered on identifying keywords representative of the core themes in each course. Preprocessing techniques are applied to refine the course materials for accurate assessment, followed by similarity metrics to quantify the degree of overlap. The results are visualized through tools such as Word Clouds, providing a clear picture of the shared and unique elements of the courses.

This initiative not only supports the academic goals of content optimization but also fosters collaboration among instructors. By aligning course objectives and minimizing redundancies, educators can offer more diverse and impactful learning experiences. This project serves as a steppingstone for broader efforts to evaluate and enhance curriculum design, ultimately contributing to higher standards in education and better preparing students for the challenges of an ever-evolving professional landscape.

To carry out this project, a step-by-step approach was followed to properly analyze the content similarity between two courses related to “Artificial Intelligence” and “Data Visualization.” These steps included data identification, text processing, feature extraction, and similarity analysis. Below is a breakdown of each step:

**1. Course Selection** Initially, two courses related to “AI” and “Visualization” were chosen as the subjects for the similarity analysis. The course materials were sourced from transcripts of lecture recordings, course slides, and supplementary reading materials provided as part of the curriculum.

For each course, the corresponding text files were extracted and prepared for processing:

**Course 1: AI2.docx:** This file contains content related to AI, including lecture transcripts, instructor notes, and key discussion points from 2 recorded sessions, each approximately 60–90 minutes long.

**Course 2: Vir1.docx:** This file consists of materials from the Data Visualization course, including slides, textual descriptions, and transcripts from 2 recorded sessions, with an average length of 60–90 minutes per session.

**2. Definition of Key Terms** Next, a set of key terms related to the topics of each course was defined. These terms were used to identify and compare the similarities between the courses:

For the AI course: {ai’, ‘machine’, ‘learning’, ‘intelligence’, ‘algorithm’, ‘data’, ‘model’}   
For the Visualization course: {‘story’, ‘narrative’, ‘data’, ‘visualization’, ‘insights’, ‘message’, ‘emotion’}

**3. Text Processing and Cleaning** The content of both text files was processed to remove any noise and irrelevant data. This step involved converting the text into a standard format, making it ready for analysis to ensure the results were accurate.

**4. Keyword Analysis** At this stage, the number of key terms found in each course was counted. This count served as a metric for comparing the similarities and differences:

In AI: 82 AI-related terms and 27 Visualization-related terms.

In Visualization: 123 AI-related terms and 216 Visualization-related terms.

**Similarity Formula Explanation:**

In this project, a simple formula is used to calculate the similarity between two courses, one related to "Artificial Intelligence" (AI) and the other to " Visualization." This formula computes similarity based on the number of shared keywords between the two courses. Specifically, it compares the keywords associated with each topic (AI and Visualization) in the course materials.

**Steps for Calculating Similarity:**

1. **Counting Keywords**: First, the number of keywords related to AI and Visualization is counted in both courses. These keywords are specific terms directly related to each topic. For example, for the "Artificial Intelligence" course, the keywords include terms like "AI," "machine learning," "data," and "algorithm." For the "Visualization" course, keywords such as "story," "narrative," "insights," "visualization," and "audience" are considered.
2. **Calculating Shared Keywords**: Next, the number of shared keywords between the two courses is calculated for each topic (AI and Visualization). The minimum count of the shared keywords for each topic is taken between the two courses.
3. **Similarity Formula**: The similarity is calculated using the following formula:

​

Where:

* are the counts of AI-related keywords in the two courses.
* are the counts of Data Visualization-related keywords in the two courses.
* represent the minimum counts of shared keywords for each topic (AI and Visualization) between the two courses.

This formula calculates the similarity based on the shared keywords for each topic, and the sum of these similarities is then divided by the total number of keywords for both courses.

**Conclusion:**

This formula provides an initial metric to assess the similarity between two courses. The resulting similarity score is a value between 0 and 1, indicating the degree of similarity or conceptual overlap between the two courses. A value closer to 1 suggests a high degree of similarity, while a value closer to 0 indicates significant differences.

# 3 Methods of Preprocessing and finding Similarity

In this project, I employed various stages of preprocessing and similarity computation to explore the relationship between documents. During the preprocessing phase, I applied different techniques to clean and prepare the text for further analysis, aiming to enhance the quality and accuracy of similarity measurement.

Initially, I performed basic text cleaning tasks such as converting the text to lowercase, removing punctuation, and eliminating irrelevant numbers. After that, I tokenized the text to split it into individual words and used n-grams (including bigrams) to capture multi-word patterns, enhancing the representation of the document.

In subsequent steps, I calculated the similarity between documents in different ways. First, I calculated simple similarity based on word overlap and frequency, and then I moved to more advanced methods, including **TF-IDF (Term Frequency-Inverse Document Frequency)** and **LDA (Latent Dirichlet Allocation)**, to model the textual data and extract meaningful features for comparison.

By adjusting the preprocessing techniques and similarity measures at each stage, I compared the results and analyzed how these changes affected the final similarity values. Through this iterative process, I aimed to provide a comprehensive understanding of the methods and their impact on similarity assessment

## 3-1 TF-IDF (Term Frequency – Inverse Document Frequency)

**TF-IDF (Term Frequency - Inverse Document Frequency)** is a statistical measure used in text mining and information retrieval to assess the importance of a word within a document relative to a collection or corpus of documents.

**Explanation of TF-IDF:**

1. **Term Frequency (TF)**:
   * **TF** measures how frequently a term (word) appears in a document.
   * It is calculated by the formula:
   * The more often a word appears in a document, the higher its term frequency. This helps capture the relevance of a word within that document.
2. **Inverse Document Frequency (IDF)**:
   * **IDF** measures the importance of a word across all documents in the corpus.
   * It is calculated by the formula:

* + Words that appear in many documents (e.g., common words like "the", “and” "is") will have a low IDF score, meaning they are less informative.
  + Words that appear in fewer documents will have a higher IDF score, indicating that they are more specific to certain documents.

**The TF-IDF Score:**

The **TF-IDF score** is the product of **TF** and **IDF**:

TF-IDF(t)=TF(t)×IDF(t)

* A high **TF-IDF** score indicates that a word is important in the given document but not common across all documents, making it more relevant or specific.

**Intuition Behind TF-IDF:**

* **TF** tells you how relevant a word is within a specific document.
* **IDF** tells you how rare or common a word is in the entire corpus.
* **TF-IDF** combines these two to identify words that are both frequent within a document but rare across all documents.

**TF-IDF** is an effective method for identifying important words in text documents by considering both their frequency within a specific document and their rarity across a corpus. This balance helps to highlight terms that contribute to the unique meaning of a document, making it a crucial tool in text analysis.

text processing and cleaning involved extracting content from .docx files, removing unnecessary numerical data from filenames, and preparing the content by structuring it into a usable format for analysis. This process is vital to ensure that only the relevant text data is considered in subsequent steps of the analysis.

## 3-2 Latent Dirichlet Allocation (LDA)

LDA is a generative probabilistic model used in natural language processing (NLP) and machine learning to discover the underlying topics in a collection of documents. LDA assumes that each document is a mixture of a small number of topics and that each word in the document is attributable to one of the document's topics. Here's a brief breakdown of how it works:

1. **Topic Modeling**: LDA aims to uncover hidden thematic structures in a large set of text data. It represents documents as a mixture of various topics, where each topic is a distribution over words.

To determine the **optimal number of topics in the LDA model**, we used several analytical methods to ensure that the chosen number was the most suitable for our dataset. The process of reaching this number consisted of three main steps: **Perplexity Score Analysis, Coherence Score Evaluation, and Topic Stability Assessment**.

**1-1. Perplexity Score Analysis**

In the first step, we calculated the **perplexity score** for different topic numbers, ranging from **2 to 50**. Perplexity indicates how well the model can predict unseen data; therefore, **the lower the perplexity score, the better the model's performance**. After evaluating the results, we found that the **minimum perplexity occurred at topic number 2**. This suggested that running the model with two topics would provide the best predictive accuracy.

**2-1. Coherence Score Evaluation**

Next, we examined the **coherence score**, which measures the semantic similarity between the top words in each topic. This score helps determine whether the keywords grouped within a topic are meaningfully related. Our analysis showed that **when the number of topics was set to 2, the coherence score reached its highest value**. This meant that splitting the data into two topics resulted in **logically structured topic groupings** with strong internal consistency.

**3-1. Topic Stability Assessment**

After the numerical evaluations, we also needed to **qualitatively assess the model’s output**. To do this, we reviewed **the most significant keywords in each topic**. If the number of topics was too high, the keywords became overly specific and fragmented. Conversely, if too few topics were chosen, unrelated concepts were merged together. We observed that **at topic number 2, the keywords in each topic were well-clustered and carried clear meaning**.

**Final Output and Keywords for Each Topic**

Ultimately, running the model with **two topics** showed that the analyzed documents could be divided into two **distinct categories**. The most **important keywords** in each topic were as follows:

* **AI:** Included words like *ai, decision, probably, say, yeah, like, maybe, kind, actually, human*
* **Topic 2:** yeah, think, right, data, income, like, maybe, bit, let, say.

1. **Document Representation**: Each document is considered a mixture of multiple topics. The assumption is that each document is generated by selecting a topic distribution and then selecting words based on that topic.
2. **Inference**: LDA tries to infer the topic distribution for each document and the word distribution for each topic, based on the observed words in the documents.

## 3-3 Process and Results of Similarity Evaluation

In this section, we detail the steps taken to evaluate the similarity between documents using different methods. The goal is to assess how well these methods capture the semantic similarity between text documents. We started with basic text preprocessing, moved through various techniques like tokenization and n-grams extraction, and then used different models such as TF-IDF and Latent Dirichlet Allocation (LDA) to measure similarity. After each step, we present the results and compare them to gain insights into how each method contributes to the overall similarity evaluation.

### 3-3-1 Similarity Calculation: Impact of Preprocessing: Basic Cleaning and Standardization

In this step, the text was preprocessed by converting it to lowercase, removing numbers, and eliminating punctuation marks. This basic cleaning step aimed to standardize the text and remove any unnecessary elements that could interfere with the analysis. In the next phase, **tokenization** and **bigram extraction** were added, which will expand the textual representation by breaking down the text into smaller units.

1. First Step: after loading the library, I read the text of each document and after that for preprocessing and cleaning the data I use these methods include the following steps:

* **Lowercasing:**  
  First, all texts were converted to lowercase to eliminate any potential discrepancies between uppercase and lowercase letters
* **Number Removal:**  
  Next, all numbers were removed from the text. Numbers are generally not useful for semantic analysis and can reduce the accuracy of the analysis, so removing them is necessary.
* **Punctuation Removal:**  
  After that, all punctuation marks like periods, commas, and question marks were removed from the text. Punctuation can introduce noise into the analysis and does not contribute to the meaning of the text.

1. Second Step: by using TF-IDF (Term Frequency - Inverse Document Frequency) Converting Text to Numerical Vectors.
2. Third Step: for finding Similarity by using Cosine Similarity Calculation

**Cosine Similarity Calculation**

Once the text was transformed into TF-IDF vectors, **Cosine Similarity** was used to quantify the similarity between two documents. The formula used for cosine similarity is:

Where:

A and B are the **TF-IDF vectors** of the two documents.

represents the **dot product** of the two vectors.

and are the **magnitudes (norms)** of the vectors.

The result is a similarity score between **0 and 1**, where a value closer to **1** indicates higher similarity.

***result TF-IDF: Similarity between AI2.docx and Vir1.docx: 0.8910***

***result LDA: Similarity between AI2.docx and Vir1.docx 0.4458***

This result shows high similarity between these courses. This result is not acceptable to me because it shows we high result, so I decided to improve that by using another method for cleaning and then finding the similarity between the two documents.

### 3-3-2 Similarity Calculation: Impact of Tokenization and Bigram Extraction

To enhance the accuracy and logical reasoning in measuring document similarity, I applied several Natural Language Processing (NLP) techniques. First, I performed **text preprocessing**, which involved cleaning and preparing the text by removing unnecessary elements such as punctuation, numbers, and converting the text to lowercase. This step helps standardize the text, ensuring that irrelevant details don't interfere with the analysis and makes the text cleaner and easier to process.

Next phase after Basic Cleaning and Standardization, I applied **tokenization**, which is the process of splitting the text into smaller units, typically words or phrases, known as tokens. Tokenization allows for the examination of individual words, making it a crucial step for tasks like frequency analysis or comparison, as it breaks down the text into manageable units for further processing.

Following that, I performed a **word frequency count**, where I counted the occurrences of each word in the document. Understanding word frequency is essential in text analysis because it helps determine which words are most significant in the context of a document, allowing for a more effective similarity calculation.

Additionally, I compared **common words** between documents. By identifying words that appear in both texts, I was able to measure the overlap between them, which serves as a key indicator of thematic similarity. Common words often reflect shared topics or themes, contributing to a more meaningful comparison.

Lastly, I extracted and compared **bigrams**, which are pairs of consecutive words. Bigrams capture more context than individual words, as they account for word pairs that frequently appear together, offering deeper insight into the relationships between words. This allowed me to improve the accuracy of the similarity calculation by incorporating context beyond individual word matches.

Using these methods, along with adjustments in preprocessing and the inclusion of bigrams, the similarity measure between the documents improved significantly. By applying **TF-IDF** and **Cosine Similarity** to the processed text, the calculated similarity between the documents reached ***49%***. This improvement highlights the effectiveness of the preprocessing steps and the addition of bigrams in capturing the deeper semantic relationship between the texts.

By applying for **Latent Dirichlet Allocation (LDA)** with an increased number of topics, I was able to improve the similarity measure between the two documents. The cosine similarity between **'AI2.docx' and 'Vir1.docx' was calculated to be *40%*,** showing a reasonable degree of similarity after adjusting the number of topics.

### 3-3-3 Similarity Calculation: Impact of Topic Modeling (LDA) and N-gram TF-IDF

The results of the similarity calculations reveal notable differences between the TF-IDF-based and LDA-based approaches. The TF-IDF similarity between *AI2.docx* and *Vir1.docx* is measured at **0.4787**, while the LDA-based similarity is slightly lower at **0.1501**. This difference suggests that while the documents share common terms and phrase structures, their underlying topics, as inferred by LDA, may not be entirely aligned.

Examining the extracted keywords provides further insight into these differences. The top TF-IDF keywords in *AI2.docx* include terms such as *ai*, *decision*, *probably*, *say*, and *human*, indicating a focus on artificial intelligence and decision-making concepts. On the other hand, *Vir1.docx* features words like *yeah*, *think*, *right*, *data*, and *income*, suggesting a more general discussion, potentially involving economic or data-driven aspects. These distinctions highlight how TF-IDF prioritizes frequently occurring and uniquely weighed terms within each document.

In contrast, the LDA model extracts topic-based keywords across multiple topics. While some redundancy appears in the topic-word distributions, certain topics introduce more specific terms. For instance, one of the topics includes words such as *ai*, *like*, *say*, *probably*, *maybe*, and *decision*, which align with the focus observed in *AI2.docx*. However, other topics predominantly feature words like *zoom*, *fast*, *explanation*, and *expressed*, which may not be particularly informative in distinguishing between documents. This redundancy suggests that the topic modeling process may require further fine-tuning, such as optimizing the number of topics to better capture the document themes.

The observed differences between TF-IDF and LDA demonstrate how each method approaches similarity measurement from different perspectives. While TF-IDF focuses on the statistical importance of individual words, LDA interprets documents based on topic distributions. The slight variation in similarity scores indicates that while the documents may share commonly used words, their broader contextual meaning and topic structures differ to some extent. By combining these two approaches, we gain a more comprehensive understanding of document relationships, balancing lexical and thematic similarities.

### 3-3-4 Keyword Optimization for Improved Similarity Measurement

Initially, I used a basic keyword matching approach to identify relevant terms within the text. However, the results obtained from this approach showed that the keywords extracted didn’t align well with the course materials, which was a limitation in achieving an accurate similarity assessment. The extracted keywords did not fully capture the subject matter or the essence of the lessons.

To overcome this issue, I decided to refine the keyword selection process by using a more advanced method. Specifically, I utilized a technique derived from **Google Search Optimization (SEO)**. This method is often employed to improve website ranking and involves selecting highly relevant and frequently searched keywords. The rationale for using this technique was to better align the extracted keywords with the course content, ensuring they were more contextually accurate and representative of the topics.

After applying this approach, I gathered the following refined set of keywords for each subject:

* **For the AI-related course**:  
  keywords\_ai = {'ai', 'machine', 'learning', 'intelligence', 'algorithm', 'data', 'model'}
* **For the VIR-related course**:  
  keywords\_vir = {'machine', 'learning', 'data', 'predict', 'regression', 'classification', 'algorithm'}

These keywords were then used to count the occurrence of each term in the respective documents, helping to better match the content of the files to the topics and lessons they were intended to represent.

By refining the keyword set using this method, I aimed to enhance the accuracy and relevance of the similarity measurement, ultimately improving the results of the comparison.

After the preprocessing step, I started by defining the specialized keywords for each topic.

Then I proceeded to read and preprocess all the text files. Once the text was cleaned, I counted how many of the defined keywords appeared in each file for both topics. This gave me the number of keywords related to AI and Virtualization in each document.

Next, I compared the keywords between the documents. I counted the occurrences of AI-related keywords and VIR-related keywords for each document and then compared these counts between pairs of documents. The similarity between each pair of documents was calculated by summing the shared keywords for both topics and dividing by the total number of keywords found in both documents. This provided a similarity score between the files based on the presence of the defined keywords.

In the end, this method allowed me to calculate the similarity between documents such as "AI2.docx" and "Vir1.docx", which resulted in a similarity score of 0.2433. This score indicated the degree of similarity between the two documents based on the presence of the specified AI and VIR keywords.  
he similarity between two files has been calculated using the sum of the keywords present in both files. For each pair of files, you used the following formula to compute the similarity:

Where:

* **shared\_AI\_keywords**: The number of common AI-related keywords between the two files.
* **shared\_Vir\_keywords**: The number of common ML-related keywords between the two files.
* **total\_keywords**: The total number of keywords found in both files.

This methodology effectively helped identify documents related to virtualization by analyzing the presence of key terms associated with this subject. By focusing on these keywords, we were able to quantify the relationship between documents and determine the degree of similarity based on their relevance to the topic of virtualization and AI. This approach can be extended further by refining the keyword sets or applying it to other subject areas to analyze document similarities more broadly.

# 4 Conclusion: