

MULTILINGUAL NEWS ARTICLE SIMILARITY

PBL-II Report submitted by

Aniket Kumar 19803012
Priyanshu Jaiswal 19803016
Utkarsh Choudhary 19803014



November 2023

*Submitted in partial fulfilment
of the degree of*

5 Year Dual Degree Programme B. Tech.

In

Computer Science Engineering

**DEPARTMENT OF COMPUTER SCIENCE
ENGINEERING & INFORMATION TECHNOLOGY**

**JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY,
NOIDA**

TABLE OF CONTENTS

DECLARATION.....	3
CERTIFICATE.....	4
ACKNOWLEDGEMENT.....	5
SUMMARY.....	6
1. INTRODUCTION.....	7-10
1.1 General Introduction.....	7-8
1.2 Problem Statement.....	9
1.3 Significance.....	9-10
1.4 Empirical Study	9-10
1.5 Brief Description of the Solution.....	9-10
2. LITERATURE SURVEY.....	11-16
3. REQUIREMENT ANALYSIS AND SOLUTION APPROACH.....	17-20
3.1 Overall Description of the project.....	17
3.2 Requirements Analysis.....	18-19
3.3 Solution Approach.....	20
4. MODELING AND IMPLEMENTATION DETAILS.....	21-31
4.1 Design Diagrams.....	21
4.2 Methodology.....	22-28
4.3 Implementation.....	29
4.4 Designing and Implementation constraints.....	30-31
5. FINDINGS, CONCLUSION AND FUTURE WORK.....	32-35
5.1 Findings.....	32-34
5.2 Limitations.....	34
5.3 Conclusion.....	35

DECLARATION

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material that has been accepted for the award of any other degree or diploma of the university or other institute of higher learning except where due acknowledgement has been made in the text.

I also declare that work done till mid evaluation was a combined effort by the entire team, however any work done thereafter was a sole contribution by Aniket Kumar.

Name: Aniket Kumar

Priyanshu Jaiswal

Utkarsh Choudhary

Enrollment Number: 19803012

19803016

19803014

Place: Jaypee Institute of Information Technology, Noida

Date: 22-04-2023

CERTIFICATE

This is to certify that the work titled “**Multilingual News Article Similarity**” submitted by **Aniket Kumar, Priyanshu Jaiswal and Utkarsh Choudhary** in partial fulfillment of degree of **5 Year Dual Degree Programme B. Tech. in Computer Science Engineering** of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

ACKNOWLEDGEMENT

We would like to express our sincerest gratitude and deep appreciation to our respected teachers, **Dr. Amit Mishra, Dr. Indu Chawla, Dr. Archana Purwar** for giving us this great opportunity to be involved under his guidance and work on this project. I would also like to thank him for always answering my issues, helping me to overcome difficulties during the project and his guidance has given me enormous confidence in completing my work. His high expectations from her students have always encouraged me to do our best to be successful. It is a great honor for us to be one of her students.

I would also like to thank Jaypee Institute of Information Technology for giving me a platform to show- case my skills and work on this project.

Last but not least, I want to thank my family for their support and encouragement. Their love, patience and understanding have encouraged me to overcome all the difficulties.

Signature:

Name: Aniket Kumar

Priyanshu Jaiswal

Utkarsh Choudhary

Enrollment Number: 19803012

19803016

19803014

Date: 18-11-2023

SUMMARY

The vast volume of information available to people in the modern digital age presents both a chance and a difficulty while navigating the news environment. Although it is now very easy to obtain news from anywhere in the world, people who are looking for material that suits their interests may become overwhelmed by the abundance of content that is readily available. This situation frequently results in information overload, which cultivates a type of "information blindness" in which users unintentionally limit themselves to the limited viewpoints prescribed by the feeds they have selected, therefore losing out on the wide range of news that is outside of their immediate sphere of influence. Taking note of this problem, our project aims to create an advanced algorithm using cutting-edge natural language processing methods. Improving the effectiveness of data classification and retrieval in the news media domain is the main goal. By finding patterns of similarity, the suggested algorithm seeks to evaluate and compare news pieces in a way that goes beyond traditional keyword-based searches. The program aims to establish subtle linkages between articles by utilizing sophisticated linguistic and contextual factors, which makes it easier for readers to find similar or redundant material.

In a time when news distribution is changing quickly, this creative method not only expedites the curation of news content but also enhances information accessibility and accuracy. By identifying patterns in news items, the algorithm may enable users to investigate a wider range of information, reducing the possibility that they will be restricted to viewpoints that are imposed upon them by the sources of information they are now exposed to. The project's main goal is to alleviate the problems caused by an abundance of information by providing users with a more dynamic and all-encompassing news consumption experience in the rapidly changing digital media ecosystem.

Chapter 1

INTRODUCTION

1.1 General Introduction

As most modern newspapers are available online and now contain information in easily accessible repositories, access to big data has become more accessible for the community. However, this abundance of news headlines leads to the question: Is there a way to automatically detect the linking of an article without arbitrarily reading all single articles? As of 2017, there have been an average of 30,948,149 U.S. weekly newspaper broadcasts. It is true that many, if not all, of these articles contain different writing styles. So though two articles may be about the same event or topic, different writing styles can make them very different from each other. Users are mystified by similar and nearly identical news. If a person incorrectly detects two news stories as similar while one has fresh data, similarity slows down the process of discovering new information about a topic and may lead to missing information. Similar news pieces are far more difficult to find on websites. This is due to the vast amount of unrelated content or information contained in these articles. Although the core news article text on two separate web pages may be the same, the additional stuff on the pages may not.

This makes the problem worse. Importantly, dealing with large amounts of data with different writing styles can be difficult, but not impossible. There are already natural language processing (NLP) algorithms, latent semantic analysis (LSI), and title-finding algorithms that can analyze the text of plain-text documents for semantic structure. To begin, this paper developed a method for scraping top news headline text from web pages, such as Google News feed websites, referring to the same event. The extracted text was then used to classify news pairs with the same content, avoiding any irrelevant information on the articles. This study can distinguish similar and dissimilar news articles by evaluating a similarity score for news pairs using various methods like cosine similarity, jaccard similarity, Doc2vec, etc. The goal of this research is to find news articles in a similar corpus. The study focuses on the representation of news and the measurement of similarity among new articles in particular. The entities with similar names that they include as representative elements of the news are

used in this experiment. This work proposes a new method based on a knowledge base framework that attempts to offer human input on the value of the category of named entities inside the news to measure the similarity across articles of the same news. We compared our technique to a standard one that produces superior results in a comparable corpus. Similarities and distance measurements convert the similarity of two documents or sentences into a single numerical value, revealing the degree of similarity or separation. The researchers examined a variety of similarity measurements, but there hasn't been much research on the similarity of newspapers. The goal of this project is to analyze the similarity of two news articles, in order to improve human comprehension. The primary idea behind comparing news stories is to find out how similar they are. Identifying feature article vectors and then evaluating the difference between those features is the basic principle for measuring news similarity. A small gap between those traits indicates a high level of similarity, whereas a large distance between them indicates a low level of similarity. Some of the distance metrics utilized in document similarity computation are Euclidean distance, Cosine distance, and Jaccard coefficient metrics. Identifying feature article vectors and then evaluating the difference between those features is the basic principle for measuring news similarity. A small gap between those traits indicates a high level of similarity, whereas a large distance between them indicates a low level of similarity. Some of the distance metrics utilized in document similarity computation are Euclidean distance, Cosine distance, and Jaccard coefficient metrics.

1.2 Problem Statement

- Task:-Given a pair of news articles, are they covering the same news story?
- The aim is to develop systems that identify multilingual news articles that provide similar information and provide user with a set of articles with least similarity.
- This is a document-level similarity task in the applied domain of news articles, rating them pairwise on a percentage scale where a higher percentage means higher similarity.

1.3 Significance of the Problem

With this project, we aim to segregate and group news articles of a similar kind. More than other features like the style of writing, paragraph phrasing, and emotion conveyed, we will judge the articles on the basis of the anecdotes they mention in them. This project will provide us with a desired number of different news articles related to our search by comparing the contents of multiple website links using a combination of algorithms.

This interests us because many news publishing applications experience a lot of spam as users from all across the world want to access the same content in their regional languages. However, two articles about a political discussion in parliament are likely to be shown as similar, even though they are not.

1.4 Empirical Study

No field survey or experimental study has been done to study the implications of the problem stated in this project. Research work related to the stated topic has been studied, but what we have proposed is not stated in any paper and is a new proposition.

1.5 Brief Description of the Solution

In today's fast-paced digital age, the constant stream of global events generates a plethora of news articles every second. With the increasing shift towards a digital environment, people prefer accessing news on their laptops or smartphones. When searching for information on a specific topic or event, users are often inundated with numerous website links containing related keywords. However, many of these links lead to articles that share strikingly similar content, making it impractical and time-consuming to sift through each one.

To address this issue, our project aims to streamline the information retrieval process by providing a mechanism to calculate the similarity between different news articles related to a search query. The goal is to identify and present a curated list of articles that offer distinct perspectives and unique information on the given topic. The program starts by collecting relevant links to news articles associated with the search query. It then employs various natural language processing techniques to assess the similarity between pairs of articles.

The essence of the project lies in efficiently identifying which articles are closely related and, conversely, which ones are significantly distinct from each other. By leveraging advanced linguistic analysis, the program can determine the degree of similarity between the content of different articles. Subsequently, it recommends a list of news items that are least similar, ensuring that users can access the most unique and diverse information in the shortest possible time.

In essence, this project not only acknowledges the digital era's information abundance but also addresses the need for an intelligent tool that enhances the efficiency of information consumption by presenting users with a carefully curated selection of news articles that offer varied perspectives on a given topic.

Chapter 2

LITERATURE SURVEY

Table 1

Title	Multilingual News Article Similarity
Author(s)	William Nafack, Leo Ly, Hua Chan, Reema Kumari
Publisher	arXiv preprint arXiv:2208.09715
Date Of Publication (month/year)	2022
Summary	<p>The SemEval 2022 Task 8: Multilingual News Article Similarity aims to solve the problem of language barriers in reading news articles from different sources. The project involves building a model to rate the similarity of a pair of news articles on a scale of 1 to 4, based on how well they cover the same news story. The dataset consists of 4964 pairs of news articles in different languages, and the text is cleaned to reduce noise. The model then uses Back Translation to add missing pairs of data, translating the data from its source language to English and back to its source language using the Google Translate API.</p> <p>The study focuses on using the Bidirectional Encoders for Transformer Representation (BERT) architecture to model news stories. They modified the RoBERTa architecture to handle feeding both pairs of articles into the transformer, extracting only part of the articles and their title as input. The model assumes that the start and end of an article contain most of the meaningful information. The encoded representation is passed into a fully connected classification layer for</p>

	<p>regression over the desired class label.</p> <p>The model is weighted to cover all sub-dimensions of similarity, with the overall score being the most important variable. The model assigns the highest weight to the overall loss, while other aspects share equal weight but less than the overall score.</p> <p>The Siamese XLM-RoBERTa model was also experimented with, incorporating named entities, keywords, article descriptions, and the new set of articles from the data augmentation process. The models were trained extensively to evaluate the importance of each added feature, and the weighting of the overall class was changed at different iterations.</p> <p>The baseline model performed poorly, with a noticeable improvement from 67% to 72.3% with the support of metadata. The Siamese XLM-RoBERTa did not perform as expected due to the concatenation approach used, which is still unclear for sentence embeddings.</p>
--	--

Table 2

Title	Text Similarity Measures in News Articles by Vector Space Model Using NLP
Author(s)	Ritika Singh Satwinder Singh
Publisher	Journal of The Institution of Engineers (India)
Date Of Publication (month/year)	2020
Summary	The global size of online news websites is over 200 million, with over 2 million articles published daily. However, many articles are very similar to many other websites, making it difficult to identify top news

headlines and measure the similarity among news across various news associations. This paper aims to extract top news items from online news platforms and measure the similarity between two similar news items in two languages (Hindi and English) referring to the same event. The method involves translating Hindi news articles into English using Google Translator and comparing them with English news articles. The study uses methods like cosine similarity, Jaccard similarity, and Euclidean distance to calculate news similarity scores. The frequency of nouns and the next word of nouns from the news articles are also extracted.

The study aims to discover bilingual news articles in a comparable corpus and assess the similarity among new articles using a knowledge base framework that provides human information on the value of the category of named entities within the news. The study explores two separate methods for generating features from texts: (1) Tf-idf vectors and (2) bag of words.

The proposed method introduces two methods for calculating the similarity between two articles of the same news, present in two different languages (Hindi and English), based on methods for calculating feature vectors and similarity measures. This will help optimize human understanding and improve the accuracy of news articles on online news websites.

The study uses TF-IDF vectors to transform pre-processed news articles into sparse-matrix weights for each word. Similarity is calculated using cosine and Jaccard similarity using Sklearn's built-in module. The pre-processed documents are compared using the bag-of-words model, which loads all news articles into a corpus, calculates feature vectors, and computes Euclidean distance. The proposed algorithms are implemented using Python 3.7.3, and the results are analyzed on pairs of news articles. The study aims to improve the representation method for predicting news articles.

Table 3

Title	Comparison of Jaccard, Dice, Cosine Similarity Coefficient to Find Best Fitness Value for Web Retrieved Documents Using Genetic Algorithm
Author(s)	V. Thada D.V. Jaglan
Publisher	International Journal of Innovative Engineering and Technology (IJIET)
Date Of Publication (month/year)	2013
Summary	The study focuses on evaluating and comparing three popular similarity coefficients: Jaccard, Dice, and Cosine, within the context of genetic algorithms applied to web document retrieval. The authors aim to determine which similarity coefficient serves as the most effective fitness value for enhancing the performance of genetic algorithms in retrieving relevant web documents. To achieve this, they likely conduct experiments, using real or simulated web documents, and assess the performance of genetic algorithms with different similarity coefficients as fitness values. By comparing these similarity measures, the paper contributes to the field of information retrieval by offering insights into the choice of an appropriate similarity coefficient to optimize the genetic algorithm's effectiveness in retrieving relevant web documents, which is crucial for various web search and recommendation systems.

Table 4

Title	Measuring News Similarity Across Ten U.S. News Sites
Author(s)	Grant C. Atkins, Alexander C. Nwala, Michele C. Weigle, Michael L. Nelson
Date Of Publication (month/year)	2018
Summary	<p>News websites make editorial decisions about what stories to include on their homepages and what stories to emphasize. The selective emphasis of a top news story and the similarity of news across different news organizations are well-known phenomena but not well-measured. This study provides a method for identifying the top news story for a select set of U.S.-based news websites and then quantifying the similarity across them. The researchers developed a headline and link extractor that parses select websites and examined ten U.S.-based news website homepages during a three-month period, November 2016 to January 2017. The method uses archived copies retrieved from the Internet Archive (IA) to discuss the methods and difficulties for parsing these websites and how events like a presidential election can lead news websites to alter their document representation just for these events. The similarity scores show a buildup (0.335) before Election Day, with a declining value (0.328) on Election Day, and an increase (0.354) after Election Day. The method shows that it can effectively identify top stories and quantify news similarity. The preservation of online news is important, as many news websites do not have screen captures of when their websites officially launched.</p>

Table 5

Title	Comparative Analysis of Similarity Measures for Sentence-Level Semantic Measurement of Text
Author(s)	S. Mohd Saad S. S. Kamarudin
Publisher	IEEE International Conference on Control System, Computing and Engineering
Date Of Publication (month/year)	2013
Summary	<p>This research addresses the crucial challenge in natural language processing (NLP) of quantifying the similarity in meaning between sentences, a vital component for NLP applications like text summarization, information retrieval, and machine translation. Throughout the paper, the authors introduce and analyze several semantic similarity measures, discussing their respective merits, limitations, and potential use cases. Experiments on standard datasets are likely conducted to gauge the performance of these measures, offering valuable insights into which measures excel in specific NLP tasks. This comparative analysis endeavors to enrich the NLP field by providing guidance on the selection of suitable similarity measures tailored to the requirements of different applications.</p>

CHAPTER 3

REQUIREMENT ANALYSIS AND SOLUTION APPROACH

3.1 Overall Description of the project

The present global size of online news websites is more than 200 million. According to MarketingProfs, more than 2 million articles are published every day on the web, but Online News websites have also circulated editorial content over the internet that specifies which articles to display on their website's homepages and what articles to highlight, e.g., broad text size for main news articles. Many of the articles posted on a news website are very similar to many other news websites. It is true that many if not all of these articles contain different writing styles. So though two articles may be about the same event or topic, different writing styles can make them very different from each other. Users are mystified by similar and nearly identical news. If a person incorrectly detects two news as similar while one has fresh data, similarity slows down the process of discovering new information about a topic and may lead to missing information. Similar news pieces are far more difficult to find on websites. This is due to the vast number of unrelated content or information contained in these articles. Although the core news article text on two separate web pages may be the same, the additional stuff on the pages may not. This makes the problem worse importantly, when comparing large amounts of data with different writing styles can be difficult, but not impossible. The selective reporting of top news headlines and also the similarity among news across various news associations is well identified but not very well calculated. This project identifies the top news items on the news sites and measures the similarity between two same news items referring to the same event. To accomplish this, a highlighted headline and link extractor has been created to extract top news from Google's news feed. We used the cosine similarity, Jaccard similarity, Euclidean distance, Word2vec to calculate news similarity score. The frequency of nouns and the next word of nouns from the news articles are also extracted. Our methodology clearly shows that we can efficiently identify top news articles and measure the similarity between news reports. This project also recommends the users a set of links that contain the most unique information.

3.2 Requirement Analysis

Functional Requirements

➤ Multilingual Language Support:

The system should support a wide range of languages to accommodate diverse news sources from around the world.

➤ Text Preprocessing:

Text data should undergo preprocessing, including tokenization, stop word removal, stemming/lemmatization, and character encoding normalization, to ensure uniform analysis.

➤ Data Ingestion:

The ability to ingest news articles from various sources and languages, either through manual input, web scraping, or API integration.

➤ Semantic Feature Extraction:

Utilize pre-trained multilingual NLP models to extract semantic features from news articles in different languages.

➤ Similarity Measurement:

Calculate similarity scores between pairs of news articles based on their extracted semantic features.

➤ Similarity Ranking and Visualization:

Present the similarity results in a ranked list, highlighting the most similar articles.

Visualize the relationships between articles using interactive graphs or charts.

➤ Performance Metrics:

Define performance metrics and regularly monitor and optimize system performance.

Non-functional Requirements

➤ Performance:

The system should respond to user requests promptly, with a response time typically below a specified threshold.

➤ Reliability:

The system should be highly reliable, with minimal downtime or service interruptions.

Implement redundancy and failover mechanisms to ensure high availability.

➤ Usability:

The user interface should be intuitive, user-friendly, and accessible to individuals with disabilities.

➤ Compatibility:

Ensure cross-browser and cross-device compatibility for the web-based interface.

Support multiple operating systems and browsers.

➤ Maintainability:

Code should be well-documented and follow coding standards to facilitate maintenance and updates.

Implement version control and change management processes.

➤ Availability:

Ensure high system availability, with scheduled maintenance communicated to users in advance.

➤ Resource Optimization:

Optimize resource usage (CPU, memory, storage) to minimize operational costs, especially in cloud-based deployments.

3.3 Solution Approach

The user starts by entering a keyword associated with a particular story or incident. The model retrieves from Google Search a list of links associated with the supplied keyword. Now, in order to move on with the similarity metrics, web scraping is used to parse and retrieve these articles. A basic Python library for instantaneous downloads and searches News articles from Google News streams can be found on GoogleNews or gnewsclient. This makes it possible to choose the most prominent headlines from Google-powered news websites or search for the most prominent headlines pertaining to a particular subject (or keyword). ‘Newspaper’ is a Python module used to extract newspaper articles and to parse them. Newspapers are using specialized Web scrapping algorithms to extract all the valuable text from a website. This works extremely well on websites of the online newspapers.

The textual news data is first pre-processed before it is represented into a more structural format. The two representation methods of generating features from the text that are investigated in this study are tf-idf, and Bag of Word. Here we are using the tf-idf method. Now, we will compare the extracted document with the help of various similarity measures such as cosine similarity, jaccard similarity, Word2vec, etc. The model will finally provide us a set of links with the most unique information.

Chapter 4

MODELING AND IMPLEMENTATION DETAILS

4.1 Design Diagrams

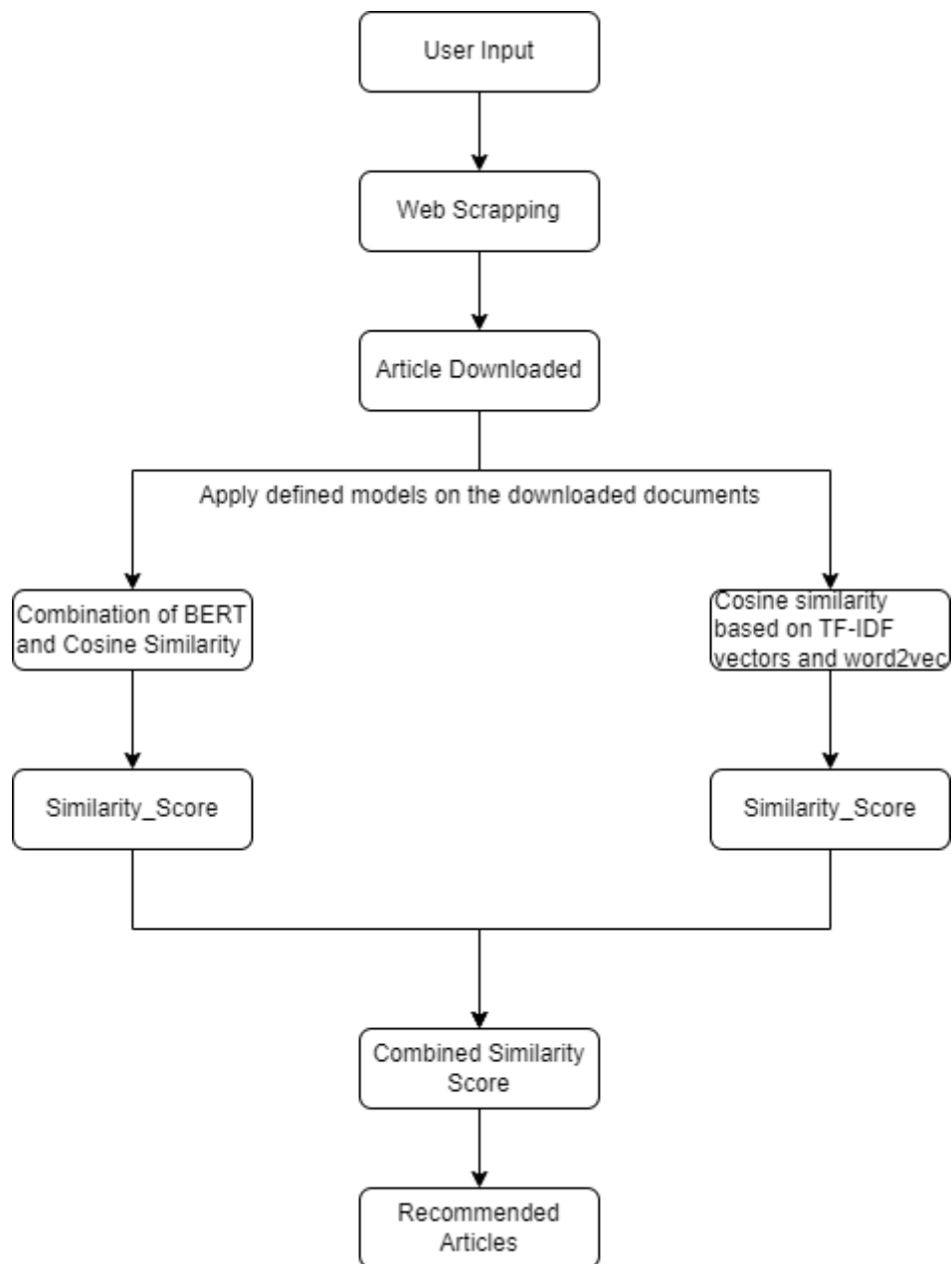


Figure: Workflow for the proposed model

4.2 Methodology

The major steps involved in this methodology are given below.

The framework of this project is shown in Figure 1. The textual news data is first pre-processed before it is represented into a more structural format. The two representation methods of generating features from the text that are investigated in this study are tf-idf, and Bag of Word. Here we are using the tf-idf method. Now, we will compare the extracted document with the help of cosine similarity measures. We further explain each of the steps in detail.

There will be three basic steps in our approach to confirm document similarity:

- The documents should be divided into words.
- Calculate the frequency of each word.
- Calculate the document vectors' dot product

News Article Scraping

We will use a python module known as 'Newspaper' to extract newspaper articles and to parse them. Newspapers are using specialized Web scraping algorithms to extract all the valuable text from a website. This works extremely well on websites of online newspapers. In this project we have extracted news articles texts from different news websites, using the Newspaper module.

Pre-processing and Data Cleaning

Pre-processing steps such as the elimination of stop-words, lemmatization, and parsing letters, punctuation marks, and numbers have been completed. The words were lemmatized by WordNetLemmatizer and NLTK library took the English stop-words.

Vector Space Model

It is a mathematical model also known as a vector model. It describes text documents as identifier variables, such as terms or tokens. It is popular in information retrieval systems but also useful for other purposes. Generally, this allows us to compare the similarity of two vectors from a geometric perspective.

Feature Vectors

A feature vector is an n-dimensional vector of computational features that describe an item in Artificial Intelligence. This is a critical tool for determining semantic similarity between texts. The methods used to measure the function vectors in this experiment are as follows: Term Frequency-Inverse Document Frequency (TF-IDF) is a simple approach for converting a text into an useful numerical representation. Tf-idf weight is a factual measure that assesses the significance of a given word in a text. In the field of mathematics,

$$tfidf\ weight = \sum_{i \in d} tf_{i,d} * \log\left(\frac{N}{df_i}\right)$$

where in document d, $tf_{i,d}$ is the number of occurrences of the i th term, df_i is the number of documents which contain i th term; N is the total number of documents. The sklearn-vectorized function was used to construct a tf-idf function. This whole model was constructed by using the documents, and a group of such tf-idf vectors was generated consisting of the tf idf weight of and term in the documents. Such tf-idf vectors have now been used as feature vectors to measure the similarity between articles in news-results.

Similarity Measures

A Similarity function is a function with a real value that calculates the similarity between two objects. The similarity calculation is achieved by mapping the distances to similarities within the vector space. Here we are using multiple similarity measures to calculate the distance between two vectors and check the similarity between multiple documents.

Cosine Similarity:-

The similarity of two vectors in an inner product space is measured by cosine similarity. It determines whether two vectors are pointing in the same general direction by measuring the cosine of the angle between them. In text analysis, it's frequently used to determine document similarity. Thousands of characteristics can be used to characterize a document, each of which records the frequency of a specific word (such as a keyword) or phrase in the

document. As a result, each document is an object that is represented by a term-frequency vector. The term-frequency vectors are usually lengthy and sparse (i.e., they have many 0 values). Information retrieval, text document grouping, biological taxonomy, and gene feature mapping are some of the applications that use such structures. Traditional distance metrics like the ones we looked at before in this chapter don't function well with such sparse numeric data. Two term-frequency vectors, for example, may share many 0 values, indicating that the related documents do not share many words, but this does not imply that they are similar. We need a metric that focuses on the terms that appear in both documents and the frequency with which they appear. To put it another way, we need a numeric data measure that ignores zero-matches. Cosine similarity is a measure of similarity that can be used to compare documents or, say, give a ranking of documents with respect to a given vector of query words. Let x and y be two vectors for comparison. Using the cosine measure as a similarity function, we have

$$\text{similarity}(A, B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^n A_i \times B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

where $\|x\|$ is the Euclidean norm of vector $x=(x_1, x_2, \dots, x_p)$ defined as $\sqrt{x_1^2 + x_2^2 + \dots + x_p^2}$. It is the vector's length in terms of concept. Similarly, the Euclidean norm of vector y is $\|y\|$. The cosine of the angle between vectors x and y is computed by the measure. A cosine value of 0 indicates that the two vectors are orthogonal (at 90 degrees to each other) and do not match. The lower the angle and the better the match between vectors, the closer the cosine value is to 1.

As shown in Fig.2. below, suppose there are two point's p_1 and p_2 , as the distance within these points increases the similarity between these points decreases and vice versa

$$1 - \text{Cosine Similarity} = \text{Cosine Distance}$$

The angle's outcome will display the outcome. When the angle between the vectors of the documents is zero, the cosine function equals one and both papers are identical. The cosine function will be less than one if the angel is any other value.

If the angle reaches -1, the documents are wholly distinct. Therefore, one may determine whether or not the vectors of P1 and P2 are pointing in the same direction by computing the cosine angle between them.

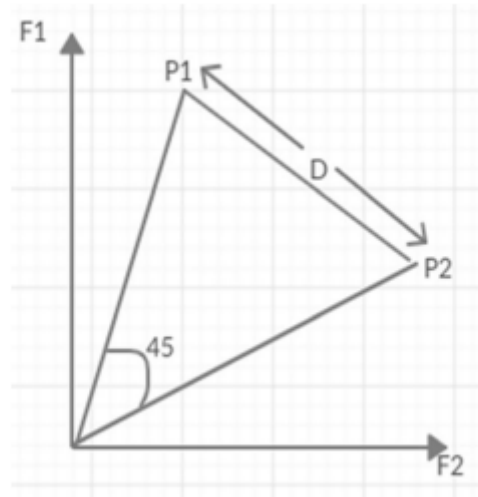


Figure 2. Cosine Similarity

Jaccard Similarity:

Jaccard similarity, also known as the Jaccard coefficient or Jaccard index, is a measure of similarity between two sets. It's commonly used in data science, information retrieval, and recommendation systems to compare the similarity of two sets, such as sets of documents, words, or items.

The Jaccard similarity is defined as the size of the intersection of the sets divided by the size of their union. Mathematically, it can be expressed as:

$$J(A, B) = |A \cap B| / |A \cup B|$$

Where:

$J(A, B)$ represents the Jaccard similarity between sets A and B.

$|A \cap B|$ is the size (cardinality) of the intersection of sets A and B.

$|A \cup B|$ is the size (cardinality) of the union of sets A and B.

The Jaccard similarity ranges from 0 to 1, with 0 indicating no similarity (completely dissimilar sets) and 1 indicating perfect similarity (identical sets). The higher the Jaccard similarity between two sets, the more similar they are.

Jaccard distance which instead of similarity measures dissimilarity between can be found by subtracting Jaccard similarity coefficient from 1:

$$JD(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

Word2vec:

Word2Vec is a popular technique for learning distributed representations of words in a continuous vector space. It is designed to capture semantic relationships between words based on their co-occurrence patterns in a given corpus. The idea behind Word2Vec is to represent words as dense vectors in such a way that words with similar meanings are close to each other in the vector space. Word2Vec models are trained on large text corpora and can be used for various natural language processing (NLP) tasks, including similarity checking between words, phrases, or documents.

Using large amounts of unannotated plain text, word2vec learns relationships between words automatically. The output is vectors, one vector per word, with remarkable linear relationships.

Word2Vec is very useful in automatic text tagging, recommender systems, and machine translation.

Word2Vec is a more recent model that embeds words in a lower-dimensional vector space using a shallow neural network. The result is a set of word vectors where vectors close together in vector space have similar meanings based on context, and word-vectors distant from each other have differing meanings. For example, strong and powerful would be close together and strong and Paris would be relatively far.

Using a combination of BERT and Cosine Similarity

We can use BERT (Bidirectional Encoder Representations from Transformers) embeddings and cosine similarity to calculate the similarity score between two documents. Here's a general outline of the process:

1. Tokenization and Embedding with BERT:
 - Tokenize each document using the BERT tokenizer.
 - Convert the tokenized sequences to BERT embeddings. BERT typically requires special tokens like [CLS] for the start of the sequence and [SEP] for separating sentences.
2. Pooling Strategy:
 - Apply a pooling strategy to obtain a fixed-size representation of the entire document.
3. Cosine Similarity:
 - Calculate the cosine similarity between the embeddings of the two documents.

Based on TF-IDF (Term Frequency-Inverse Document Frequency) and Word2Vec-based similarity

Assessing the similarity of two or more documents in terms of meaning, content, and grammatical structure is known as "document similarity measurement," and it is based on semantics, text. Applications of natural language processing (NLP) such as text summarization, document clustering, and information retrieval all depend on this process. Here's a quick rundown of the elements involved:

1. Text Preprocessing:
 - Tokenization: Breaking down the text into individual words or tokens.
 - Stopword Removal: Eliminating common words (e.g., "the," "and") that do not contribute significantly to the meaning.
 - Lemmatization: Reducing words to their base or root form to ensure consistency (e.g., "running" becomes "run").
2. TF-IDF (Term Frequency-Inverse Document Frequency):

- Computing a numerical representation of the importance of each word in a document relative to a collection of documents.
 - Generating vectors that capture the frequency of terms in a document while considering their significance in the entire corpus.
3. Cosine Similarity:
- Measuring the cosine of the angle between two vectors to quantify their similarity.
 - Applied to the TF-IDF vectors to assess the similarity between documents based on their term frequencies.
4. Word Embeddings (Word2Vec):
- Training models that represent words as vectors in a continuous vector space.
 - Capturing semantic relationships between words and phrases.
 - Calculating similarity based on the cosine similarity between the vectors of words in the documents.
5. Combined Similarity:
- Integrating multiple similarity measures, such as TF-IDF-based cosine similarity and Word2Vec-based similarity, to obtain a comprehensive evaluation.
 - Adjusting weights to prioritize certain aspects (e.g., emphasizing semantics over syntax).

4.3 Implementation

We have used the Google colab notebook, Python version 3.6. For the implementation of our code we have used the dataset containing different news articles from diverse websites.

To implement the proposed model, first the model retrieve a set of links related to the desired search using *BeautifulSoup* and now, the news articles have to be scrapped with the help of retrieved urls and then stored in separate text files.

Scraping Articles from Web:

1. For scraping and downloading contents from a news website, the newspaper library is required to be installed. Once installed, the required libraries have to be imported.

Also, the nltk library has to be imported as this implementation requires several natural language processing steps.

2. The punkt sentence tokenizer is needed to be downloaded as the punkt library is used to tokenize the sentences in order to be used for NLP.
3. A list is created in which the urls are passed for whichever news articles that have to be scraped and summarized.
4. Now download, parse and perform NLP on the news articles. The articles are now scraped and downloaded and all the useful information like title of the article, texts, summary, key from each article are printed separately on the console.
5. Above printed results are stored in different text files which will be further used in comparing their similarity

Now, we will use the above defined similarity measures to calculate the similarity between different pair of documents taking the first article as the test document.

Two models have been utilized here (a). Combination of BERT and Cosine Similarity

(b) Model based on TF-IDF (Term Frequency-Inverse Document Frequency) and Word2Vec-based similarity; this model computes the similarity score between each pair of documents using a variety of metrics, including word2vec, cosine similarity and bert embedding.

To get the best result, the average of similarity scores of both the models have been computed and it has been used to recommend the user with the least similar news articles related to their search.

4.4 Designing and Implementation constraints:

Designing and implementing a multilingual news article similarity checker can be a complex task, and there are several constraints and considerations to keep in mind. Below are some key design and implementation constraints for such a system:

- **Multilingual Support:**

Constraints: The system must be able to handle news stories written in different languages.

Implementation: Use language detection to determine the language of each article and then use text processing strategies appropriate to that language.

- Scalability:

Constraint: The system should be able to scale to handle a large number of articles.

Implementation: Utilize distributed computing frameworks and databases to manage and process a large volume of data efficiently.

- Data Collection and Integration:

Constraint: Gathering news articles from various sources and languages can be challenging.

Implementation: Implement web scraping or APIs to collect articles, and integrate data from multiple sources.

- Text Preprocessing:

Constraint: Text data from different languages requires language-specific preprocessing steps.

Implementation: Apply tokenization, stemming, stop-word removal, and other preprocessing techniques specific to each language.

- Feature Extraction:

Constraint: Extracting meaningful features for comparison across languages can be complex.

Implementation: Use techniques like TF-IDF, word embeddings (e.g., Word2Vec, FastText), or deep learning models (e.g., BERT) for feature extraction.

- Cross-Language Comparison:

Constraint: Comparing articles across different languages is non-trivial.

Implementation: Translate articles to a common language (e.g., English) for comparison, or use cross-lingual embeddings to bridge the language gap.

- Similarity Metrics:

Constraint: Choosing an appropriate similarity metric for multilingual data is crucial.

Implementation: Experiment with various metrics such as cosine similarity, Jaccard similarity, or specialized multilingual similarity measures.

- Language Model Availability:

Constraint: Availability of pre-trained language models may be limited for some languages.

Implementation: Explore training custom language models or adapting existing ones to under represented languages.

Chapter 5

FINDINGS, CONCLUSION AND FUTURE WORK

5.1 Findings

At first, the user enters the keyword and the number of articles related to that keyword he wants to search. The program fetches the links of news articles related to given keyword.

```
Enter the keyword to search for news articles: delhi pollotion
Enter the number of results to retrieve: 10
News articles related to 'delhi pollotion':
1. https://www.bbc.com/news/world-asia-india-67358305
2. https://www.cnn.com/2023/11/06/india/new-delhi-pollution-level-high-intl-hnk/index.html
3. https://www.ndtv.com/india-news/early-winter-break-in-delhi-schools-from-november-9-18-due-to-air-pollution-4556578
4. https://www.cnn.com/2023/11/06/indias-toxic-smog-season-shuts-down-schools-revives-vehicle-limits.html
5. https://www.hindustantimes.com/cities/delhi-news/amid-spike-in-pollution-delhi-s-high-no2-levels-a-sign-of-worry-101699380870090.html
6. https://www.bloomberg.com/news/articles/2023-11-08/toxic-smog-forces-delhi-to-order-schools-to-close-early
7. https://www.theguardian.com/world/2023/nov/03/delhi-india-air-quality-pollution-spike-world-health-organization-limit
8. https://www.nbcnews.com/news/world/india-new-delhi-pollution-schools-rcna123467
9. https://www.aljazeera.com/news/2023/11/3/air-pollution-in-indias-new-delhi-sparks-alarm
10. https://www.bbc.com/news/world-asia-india-67166585
```

These articles are now scrapped and stored in the form of text files which will be used to check the similarity scores.

Here, the first article is taken as a test document to check how much similar it is to the rest of the documents.

Similarity Score using a Combination of BERT and Cosine Similarity:

S.No	Article	Similarity Score
1	NewsFile2	0.4333
2	NewsFile3	0.7966
3	NewsFile4	0.1322
4	NewsFile5	0.0848
5	NewsFile6	0.2704
6	NewsFile7	0.1344
7	NewsFile8	0.1050
8	NewsFile9	0.2211
9	NewsFile10	0.3547

Similarity Score using a combination of TF-IDF (Term Frequency-Inverse Document Frequency) and Word2Vec-based similarity:

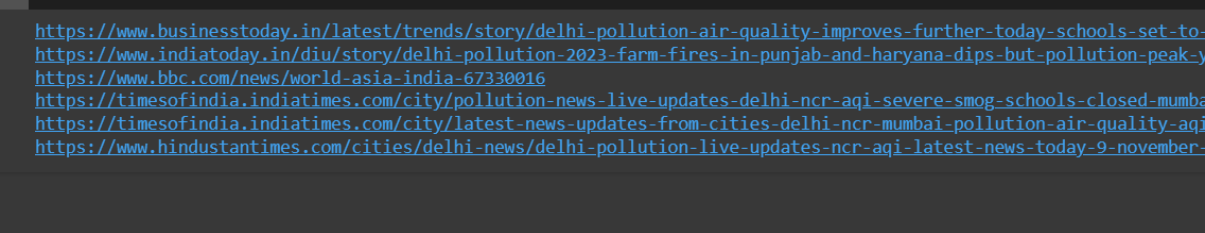
S.No	Article	Similarity Score
1	NewsFile2	0.5882
2	NewsFile3	0.6299
3	NewsFile4	0.2124
4	NewsFile5	0.1048
5	NewsFile6	0.4001
6	NewsFile7	0.2223
7	NewsFile8	0.2127
8	NewsFile9	0.3155
9	NewsFile10	0.5574

The similarity score calculated above by using both the models are now combined to get an average score and predict the unique articles.

Combined Similarity Score:

S.No	Article	Similarity Score
1	NewsFile2	0.5107
2	NewsFile3	0.7132
3	NewsFile4	0.1722
4	NewsFile5	0.0947
5	NewsFile6	0.3352
6	NewsFile7	0.1783
7	NewsFile8	0.1588
8	NewsFile9	0.2683
9	NewsFile10	0.4560

A threshold value has been set for predicting the links for unique articles. Let's take a threshold value of 0.30 which means the pair of documents with a similarity score less than 0.30 will be recommended to the user.



<https://www.businesstoday.in/latest/trends/story/delhi-pollution-air-quality-improves-further-today-schools-set-to-https://www.indiatoday.in/diu/story/delhi-pollution-2023-farm-fires-in-punjab-and-haryana-dips-but-pollution-peak-y>
<https://www.bbc.com/news/world-asia-india-67330016>
<https://timesofindia.indiatimes.com/city/pollution-news-live-updates-delhi-ncr-aqi-severe-smog-schools-closed-mumbai>
<https://timesofindia.indiatimes.com/city/latest-news-updates-from-cities-delhi-ncr-mumbai-pollution-air-quality-aqi>
<https://www.hindustantimes.com/cities/delhi-news/delhi-pollution-live-updates-ncr-aqi-latest-news-today-9-november->

5.2 Limitations

One major difficulty is that one doesn't consciously understand language ourselves. The second major difficulty is ambiguity. When you think of a linguistic concept like a word or a sentence, those seem like simple, well-formed ideas. But in reality, there are many borderline cases that can be quite difficult to figure out. For instance, is "won't" one word, or two? (Most systems treat it as two words.) In languages like Chinese or (especially) Thai, native speakers disagree about word boundaries, and in Thai, there isn't really even the concept of a sentence in the way that there is in English. And words and sentences are incredibly simple compared to finding meaning in text. Consider a word like "jaguar" or "mercury". There are a huge number of possible meanings to those -- see the jaguar wikipedia disambiguation page for a partial list: Jaguar (disambiguation) The thing is, many, many words are like that. "Ground" has tons of meanings as a verb, and even more as a noun. To understand what a sentence means, you have to understand the meaning of the words, and that's no simple task. The crazy thing is, for humans, all this stuff is effortless. When you read a web page with lists, tables, sentences, newly made up words, nouns used as verbs, and sarcasm, you get it immediately, usually without having to work at it. Puns and wordplay are constructs people use for fun -- but they're also exactly what you'd create if you were trying your best to baffle an NLP system. The reason for that is that computers process language in a way totally unlike humans, so once you go away from whatever text they were trained on, they are likely to be hopelessly confused. Whereas humans happily learn the new rules of communicating on Twitter without having to think about it. If we really understood how people understand language, we could maybe make a computer system do something similar. But because it's so deeply buried and unconscious, we resort to approximations and statistical techniques, which are at the mercy of their training data and may never be as flexible as a human.

5.3 Conclusions

To conclude, the multilingual news article similarity project has effectively tackled the task of evaluating the similarity of news pieces written in several languages. We have obtained a complete knowledge of document similarity that transcends linguistic and syntactic barriers by integrating sophisticated natural language processing (NLP) techniques, such as tokenization, part-of-speech tagging, and word embeddings.

A more detailed understanding of semantic linkages within and between languages has been made possible by the combination of Word2Vec embeddings and Bert embeddings with cosine similarity. This method improves the precision of similarity evaluations by taking into account not just the text's structural elements but also its underlying meaning and context.

The project's adaptability makes it suitable for a wide range of uses, including cross-lingual content analysis, document clustering, and information retrieval. The outcomes show how well our technology handles multilingual datasets and gets over obstacles specific to a certain language.

The knowledge gathered from this effort will serve as a basis for future studies and advancements in the area of multilingual document similarity. Combining linguistic and semantic analysis has shown to be a useful strategy, providing opportunities for advancements and applications across a range of fields. This study advances the more general objective of promoting effective information retrieval in a multilingual society and cross-cultural understanding.