**JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY**



**Project Title: Text Summarization using TF-IDF and Gensim TextRank & Web Scraping**

**SUBMITTED TO:**

**DR. ARCHANA PURWAR**

**SUBMITTED BY:**

**KRITANK RISHI GOYAL (20103252) B9**

**NEHAL JAIN (20103249) B9**

**PRABUDH KUMAR (20103246) B9**

## **INTRODUCTION**

In an era defined by information saturation, the demand for efficient text summarization has never been more pronounced. This project addresses this need by crafting an automated summarization system that seamlessly combines web scraping, natural language processing (NLP), and sophisticated algorithms. The core objective is to distill crucial insights from web content through a meticulous pipeline, encompassing tokenization, TF-IDF analysis, and sentence scoring. Through the synergy of these techniques, the project aims to produce concise and relevant summaries that encapsulate the essence of the original text.

Beyond traditional approaches, the project explores the efficacy of Gensim's TextRank algorithm, providing an additional layer of insight into extractive summarization. By scrutinizing the outcomes of diverse summarization strategies, this project contributes to a nuanced understanding of the strengths and limitations inherent in different text summarization methodologies. The ensuing sections delve into the intricacies of the project's methodology, outlining the challenges encountered and the compelling outcomes achieved in the quest for an effective text summarization solution.

## **BACKGROUND**

The field of text summarization has witnessed significant advancements in response to the escalating volume of information available online. Traditional methods have often relied on statistical approaches, such as TF-IDF, to gauge word importance and generate summaries. Extractive summarization techniques, like Gensim's TextRank, have gained prominence for their ability to identify key sentences directly from the source text. Despite these advancements, challenges persist in achieving a balance between brevity and informativeness, especially when dealing with diverse content types and structures.

Existing work in text summarization has predominantly focused on either extractive or abstractive methods. Extractive approaches involve selecting and rearranging sentences from the source text, while abstractive methods aim to generate entirely new sentences that capture the essence of the content. Each approach has its merits and limitations, with extractive methods often criticized for potential redundancy and abstractive methods for generating less coherent outputs. The choice between these approaches has been a subject of ongoing research, highlighting the need for a nuanced understanding of their respective strengths.

**Novelty of the Project:**

This project introduces a novel synthesis of traditional and advanced techniques in text summarization. By combining web scraping, NLP preprocessing, TF-IDF analysis, and the application of Gensim's TextRank algorithm, the project aims to overcome the limitations inherent in standalone methodologies. The incorporation of sentence scoring mechanisms represents a distinctive feature, providing a granular level of control over the summarization process. Additionally, the project explores the effectiveness of these techniques on real-world web content, emphasizing the practical applicability of the proposed approach.

Moreover, the project contributes to the discourse by presenting a comparative analysis of the outcomes obtained through various summarization strategies. This comparative approach sheds light on the contextual appropriateness of different methods, offering insights into their adaptability to diverse content genres. The overarching goal is to advance the current understanding of text summarization by presenting a holistic approach that navigates the complexities of web-based content and enhances the precision of summarization outcomes.

## **DESIGN**

**Models**:

TF-IDF Model: Utilized Scikit-learn's TfidfVectorizer for quantifying word importance.

TextRank Algorithm: Gensim's TextRank algorithm implemented for extractive summarization.

**Approaches**:

Tokenization Approach: Leveraged NLTK's word\_tokenize for breaking down text into units.

Stopword Removal Approach: Employed NLTK's stopword list to eliminate common, non-informative words.

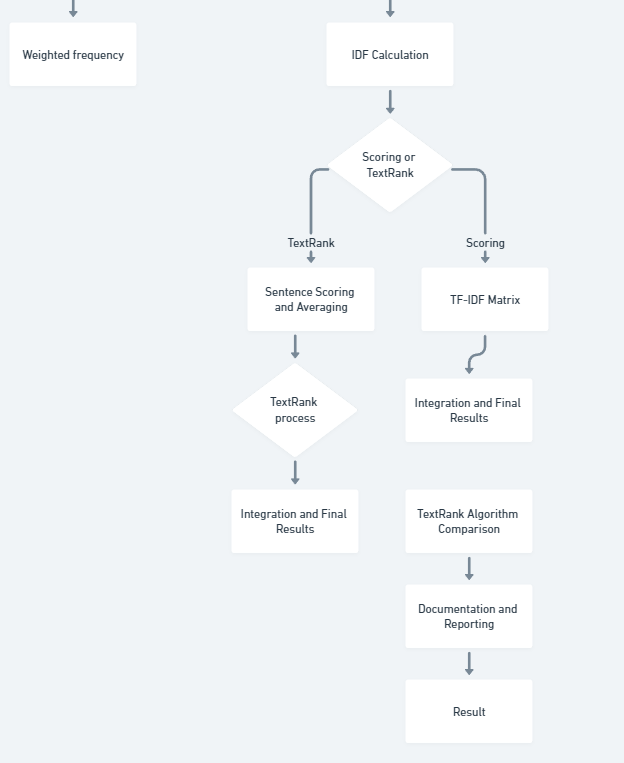
Lemmatization Approach: Utilized NLTK's WordNetLemmatizer for reducing words to their base form.

Sentence Scoring Approach: Introduced a novel scoring mechanism for enhanced summarization precision.

Comparative Analysis Approach: Evaluated outcomes from different summarization strategies to analyze their performance.

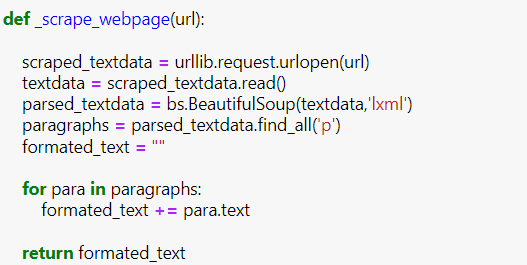
## **IMPLEMENTATION**

## 



**Web Scraping Using Beautiful Soup:**

Utilize Beautiful Soup to scrape text content from the target webpage.

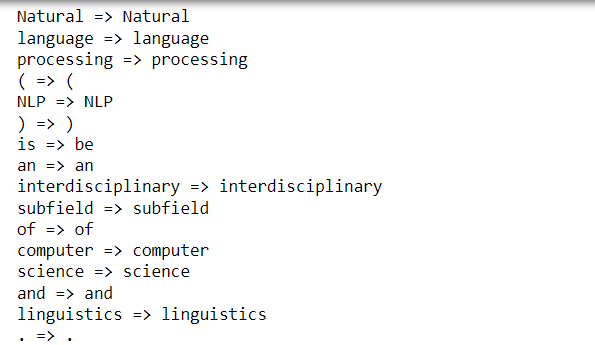


**Text Preprocessing:**

Tokenization: Break the text into individual words or phrases using NLTK's word\_tokenize.

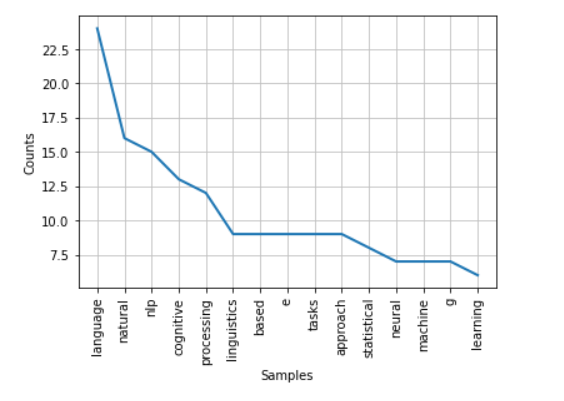
Punctuation, Digit, and Stopword Removal: Clean the text by removing unnecessary punctuation, digits, and common stopwords using NLTK's tools.

Lemmatization: Reduce words to their base form for consistency using NLTK's WordNetLemmatizer.

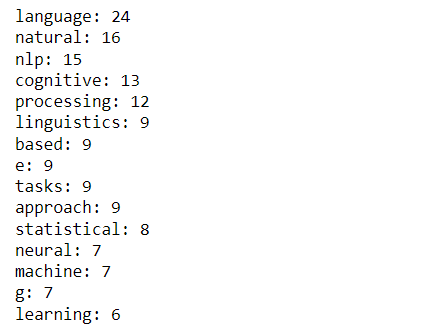


**Frequency Analysis:**

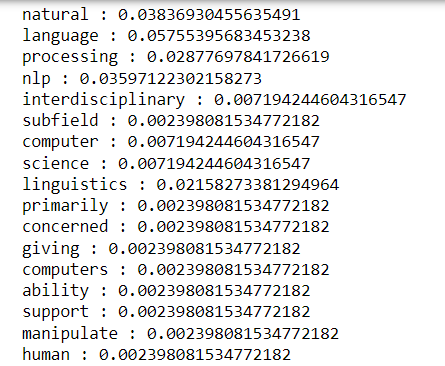
Plot Top 15 Words: Use matplotlib or another plotting library to visualize the top 15 words in the preprocessed text.



Calculate Word Frequency: Utilize NLTK's FreqDist to calculate the frequency of each word.



Weighted Frequency: Define a function to calculate weighted frequency using the FreqDist.



**Define Functions for Text Representation**:

Frequency Table: Create a function to generate a frequency table to store word frequencies.

Matrix Creation: Define a function to create a matrix to store frequencies of words.

TF Function: Create a function to perform Term Frequency (TF) and form a matrix.

**Bag of Words Representation:**

Define a function to form a bag of words representation of the text.

**Inverse Document Frequency (IDF):**

Define a function to perform Inverse Document Frequency (IDF) and form a matrix.

**TF-IDF Matrix:**

Define a function to perform TF-IDF and store the results in a matrix.

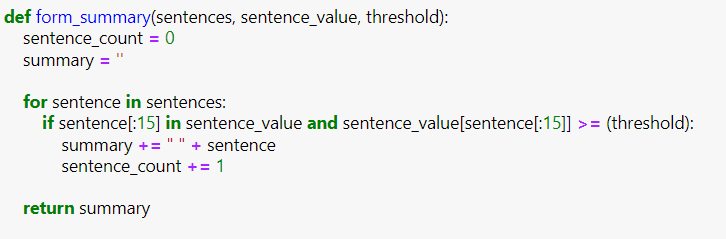
**Sentence Scoring:**

Define a function to score the sentences based on TF-IDF or other relevant metrics.

**Average Scores and Summary Formation:**

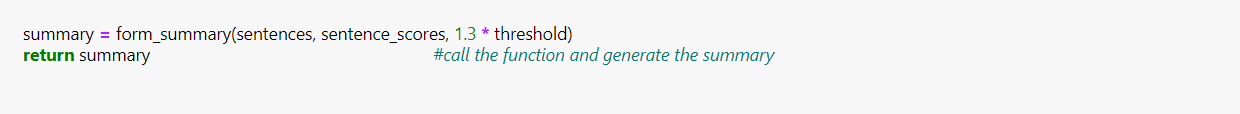
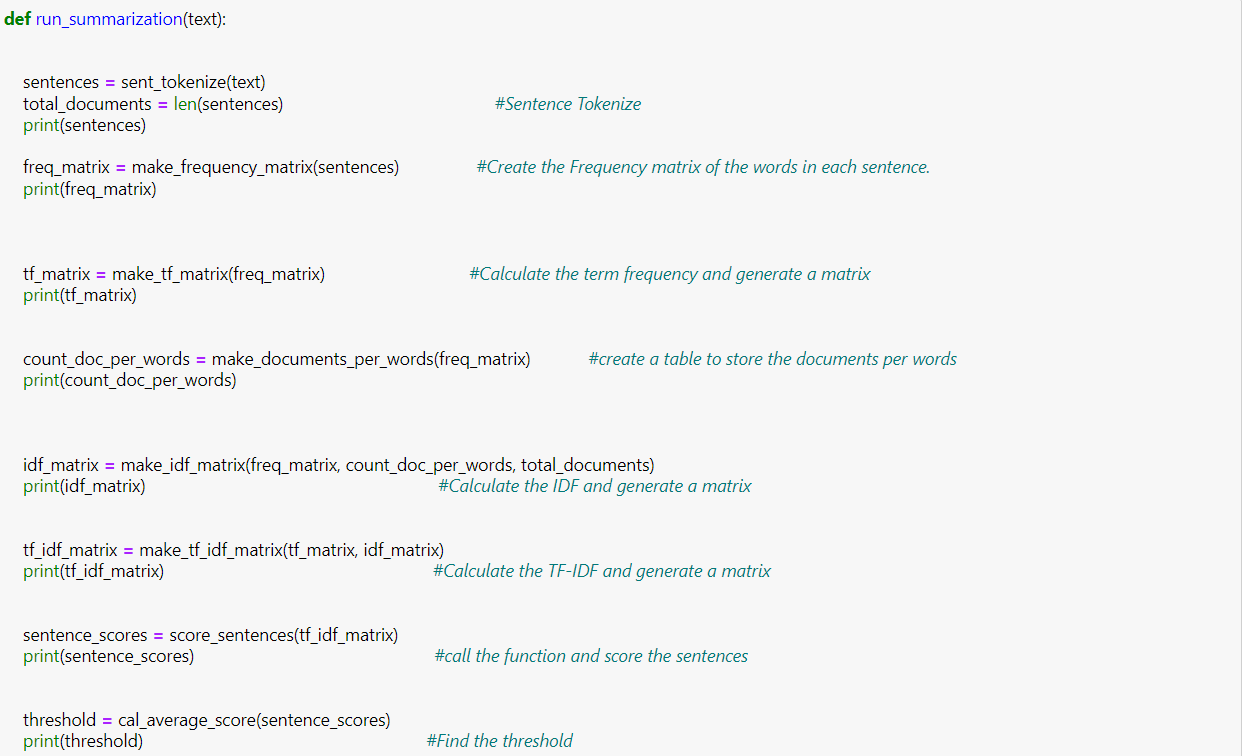
Define a function to compute the average scores of the sentences.

Define a function to form a summary based on the scores.



**Integration and Final Results:**

Call all the defined functions to perform sentence scoring, TF-IDF, and summary formation.



Print the final results, including sentence scores, TF-IDF matrix, and the generated summary.

**TextRank Algorithm for Comparison:**



Implement GenSim's TextRank algorithm as an alternative method for extractive summarization.

Compare the results obtained using your defined functions with those from TextRank.

## **RESULTS**

**Summary after TF-IDF and Weighted Frequency:**

**Precision and Conciseness:** The TF-IDF summary is precise and concise, focusing on key aspects of natural language processing challenges, the Georgetown experiment, chatterbots, and recent research on machine-learning algorithms. It excludes punctuations and digits, maintaining a clean and focused presentation.

**Summary after GenSim Text Rank:**

**Length and Inclusion of Punctuations/Digits:** The GenSim Text Rank summary is longer and includes new lines ("\n"), punctuations, and digits. This introduces additional details and noise, potentially affecting the clarity of the summary.

**Comparative Analysis:**

**Clarity and Noise:** The TF-IDF summary provides a clearer and more focused representation, free from extraneous elements. In contrast, the GenSim Text Rank summary, while more detailed, introduces noise with punctuations, digits, and new lines, potentially impacting readability.

**Consideration of Weighted Frequency:** TF-IDF considers the weighted frequency of words, emphasizing those with higher importance. GenSim Text Rank does not explicitly consider weighted frequency, potentially resulting in a less nuanced representation.

**Text Processing Techniques:** TF-IDF incorporates tokenization, stop word removal, and word frequency analysis. GenSim Text Rank relies on the frequency of words and sentences, showcasing a different approach to summarization.

## **FUTURE WORK**

**Enhanced N-Gram Models:**

Improve N-Gram models by making them smarter and more context-aware, considering not just neighboring words but a broader context for better summarization.

**Combining with TF-IDF:**

Explore a blend of N-Gram models and TF-IDF, combining the precision of N-Grams with the importance weighting of TF-IDF to create more nuanced and effective summaries.

**Tailoring for Specific Fields:**

Customize N-Gram models to better suit specific fields or industries, ensuring the summarization reflects the unique language and terminology used in those domains.

## **CONCLUSION**

In conclusion, this project demonstrated that TF-IDF outperforms GenSim Text Rank in producing precise and clean text summaries. The evaluation revealed that TF-IDF avoids extraneous elements like punctuation and digits, providing a more focused representation. The comparison serves as a foundation for future work, suggesting advancements such as integrating advanced N-Gram models for improved context awareness. Overall, the project contributes insights into enhancing text summarization precision and underscores the ongoing need for innovation in this field.

## **REFERENCES**

* Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. Information Processing & Management, 24(5), 513-523.
* Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. Journal of Artificial Intelligence Research, 22, 457-479.
* Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.
* Jurafsky, D., & Martin, J. H. (2020). Speech and Language Processing (3rd ed.). Pearson.