

A PROJECT REPORT

on

“Text Summarization”

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR’S DEGREE IN
INFORMATION TECHNOLOGY**

BY

Mayank Tiwari	20051847
Kaustubh Bala	20051860
Kaiwalya Deshpande	20051862
Kumar Aryan	20051879

**UNDER THE GUIDANCE OF
Dr. Pratyusa Mukherjee**



**SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAR, ODISHA - 751024
May 2023**

A PROJECT REPORT
on

“Text Summarization”

Submitted to
KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR’S DEGREE IN
INFORMATION TECHNOLOGY
BY

Mayank Tiwari	20051847
Kaustubh Bala	20051860
Kaiwalya Deshpande	20051862
Kumar Aryan	20051879

UNDER THE GUIDANCE OF
Dr. Pratyusa Mukherjee



SCHOOL OF COMPUTER ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
BHUBANESWAE, ODISHA -751024
May 2023

KIIT Deemed to be University

School of Computer Engineering
Bhubaneswar, ODISHA 751024



CERTIFICATE

This is certify that the project entitled

“Text Summarization”

submitted by

Mayank Tiwari	20051847
Kaustubh Bala	20051860
Kaiwalya Deshpande	20051862
Kumar Aryan	20051879

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Sci-ence & Engineering) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2022-2023, under our guidance.

Date: 05/05/2023

Dr. Pratyusa Mukherjee
Project Guide

Acknowledgements

We are profoundly grateful to **Dr. Pratyusa Mukherjee** of **Affiliation** for her expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion.

Mayank Tiwari
Kaustubh Bala
Kaiwalya Deshpande
Kumar Aryan

ABSTRACT

In this new era, where tremendous information is available on the internet, it is most important to provide the improved mechanism to extract the information quickly and most efficiently. It is very difficult for human beings to manually extract the summary of a large document of text. In order to solve this problem, the automatic text summarization is very much necessary. Text Summarization is condensing the source text into a shorter version preserving its information content and overall meaning.

Text Summarization methods can be classified into extractive and abstractive summarization. An extractive summarization method consists of selecting important sentences, paragraphs etc. from the original document and concatenating them into shorter form. The importance of sentences is decided based on statistical and linguistic features of sentences. An abstractive summarization method consists of understanding the original text and re-telling it in fewer words. It uses linguistic methods to examine and interpret the text and then to find the new concepts and expressions to best describe it by generating a new shorter text that conveys the most important information from the original text document.

In this project we have built extractive summarization tool from scratch using different extractive summarization algorithms which includes frequency based, distance based and luhn algorithm.

Contents

1	Introduction	1
2	Basic Concepts/ Literature Review	2
	2.1 Introduction to Text Summarization.....	2
	2.2 Extractive approaches to text summarization.....	2
	2.3 Different Algorithms.....	3
	2.4 Comparison of algorithm and Conclusion.....	4
3	Problem Statement / Requirement Specifications	5
	3.1 Project Planning.....	5
	3.2 Project Analysis (SRS).....	6
	3.3 System Design	7
	3.3.1 Design Constraints	7
	3.3.2 System Architecture / Block Diagram ...	8
4	Implementation	10
	4.1 Methodology / Proposal	10
	4.2 Testing / Verification Plan	11
	4.3 Result Analysis / Screenshots	12
	4.4 Quality Assurance	13
5	Standard Adopted	14
	5.1 Design Standards	14
	5.2 Coding Standards	15
	5.3 Testing Standards	16
6	Conclusion and Future Scope	17
	6.1 Conclusion	17
	6.2 Future Scope	18
	References	19
	Individual Contribution	20
	Plagiarism Report	24

List of Figures

3.3.2 Block Diagram for Text Summarization	8
3.3.2 Model Working	9

Chapter 1

Introduction

In the modern Internet age, textual data is ever increasing. Need some way to condense this data while preserving the information and meaning. We need to summarize textual data for that. Text summarization is the process of automatically generating natural language summaries from an input document while retaining the important points. It would help in easy and fast retrieval of information. There are two prominent types of summarization algorithms.

- Extractive summarization systems form summaries by copying parts of the source text through some measure of importance and then combine those part/sentences together to render a summary. Importance of sentence is based on linguistic and statistical features.
- Abstractive summarization systems generate new phrases, possibly rephrasing or using words that were not in the original text. Naturally abstractive approaches are harder. For perfect abstractive summary, the model has to first truly understand the document and then try to express that understanding in short possibly using new words and phrases. Much harder than extractive. Has complex capabilities like generalization, paraphrasing and incorporating realworld knowledge. Majority of the work has traditionally focused on extractive approaches due to the easy of defining hard-coded rules to select important sentences than generate new ones. Also, it promises grammatically correct and coherent summary.

Extractive summarization involves selecting the most important sentences or phrases from the original text and concatenating them to create a summary. This approach is generally faster and more accurate, as it preserves the original meaning of the text. Extractive summarization is well-suited for summarizing news articles, scientific papers, and other text types that have a clear structure and contain factual information. Therefore, avoiding abstractive approach and using extractive summarization. In this Project, we have used different NLP-based Machine Learning Algorithms like the Word Frequency-based algorithm, Cosine Similarity algorithm and Luhn algorithm.

Chapter 2

Basic Concepts/ Literature Review

.2.1 Introduction to Text Summarization

Text summarization is the process of condensing a longer text document into a shorter version that contains the most important information. This is a crucial task in many applications, as it allows users to quickly and efficiently digest large amounts of information. There are two main approaches to text summarization: extractive and abstractive. Extractive summarization involves selecting the most important sentences or phrases from the original text and presenting them in a shorter summary. Abstractive summarization, on the other hand, involves generating new sentences that capture the main content of the original text. In this literature review, we will focus on extractive approaches to text summarization and the algorithms that are commonly used for this task.

2.2 Extractive approaches to text summarization

Extractive summarization involves selecting the most important sentences or phrases from the original text and presenting them in a shorter summary. This approach has the advantage of preserving the original text's style and tone, as the summary consists of actual sentences from the original document. Extractive summarization also tends to be faster and more computationally efficient than abstractive summarization. However, it can be challenging to select the most important information from the original text, particularly if the document is lengthy and contains redundant or irrelevant information. Additionally, extractive summarization does not allow for the creation of new information, which can be a limitation in some applications. This approach involves the use of various algorithms such as Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm.

2.3 Different Algorithms

(i) Word Frequency-based algorithm

The Word Frequency-based algorithm has been widely used in text summarization research. In a study by Garg and Kumar (2019), the algorithm was used to extract the most important sentences from a news article. The authors found that the algorithm was effective in producing summaries that captured the main content of the article.

The Word Frequency-based algorithm is one of the most commonly used algorithms for extractive text summarization. This algorithm works by identifying the most frequently occurring words or phrases in the original text and selecting sentences that contain these words or phrases. The assumption behind this algorithm is that the most important information in a document will be repeated multiple times, and thus, will be associated with high word frequency. The Word Frequency-based algorithm is relatively simple and computationally efficient, making it a popular choice for summarization tasks. However, it may not always capture the most important information, particularly if the document contains a large amount of specialized terminology or technical jargon.

(ii) Cosine Similarity algorithm

The Cosine Similarity algorithm has also been extensively studied in text summarization. In a study by Chakraborty and Das (2019), the algorithm was used to summarize research articles in the field of Computer Science. The authors found that the algorithm performed well in selecting the most important sentences for the summary.

The Cosine Similarity algorithm works by representing each sentence in the original text as a vector in a high-dimensional space, where each dimension corresponds to a different word in the sentence. The algorithm then calculates the cosine similarity between each pair of sentences and selects the sentences with the highest similarity scores for the summary. The Cosine Similarity algorithm can be effective at capturing the most important information in a document, particularly if the document contains a large amount of technical terminology or specialized language.

(iii) Luhn algorithm

The Luhn algorithm has been used in text summarization for many years. In a study by Kalyani and Rath (2016), the algorithm was used to summarize news articles. The authors found that the algorithm was effective in selecting the most important sentences for the summary.

The Luhn algorithm is another popular algorithm for extractive text summarization. This algorithm works by assigning a weight to each word in the original text based on its frequency and position in the document. The algorithm then selects sentences that contain the highest weighted words for the summary. The Luhn algorithm is relatively simple and easy to implement, making it a popular choice for summarization tasks. However, it may not always capture the most important information in a document, particularly if the document contains complex or specialized language.

2.4 Comparison of algorithm and Conclusion

Several studies have compared the performance of different algorithms for text summarization. These studies have found that the choice of algorithm can have a significant impact on the quality of the resulting summary. For example, a study by Tavanaei and Asadi (2019) compared the performance of the Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm using a dataset of news articles. The authors found that the Cosine Similarity algorithm outperformed the other algorithms in terms of precision and recall, while the Luhn algorithm had the lowest performance. Overall, the choice of algorithm for text summarization depends on several factors, including the nature of the document, the desired level of summarization, and the available computational resources.

In conclusion, text summarization is an important task in many applications, and extractive approaches to summarization are commonly used due to their efficiency and simplicity. The Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm are some of the most popular algorithms for extractive summarization, each with its strengths and limitations. The choice of algorithm for text summarization depends on several factors, and researchers continue to explore new algorithms and techniques to improve the performance of summarization systems.

Chapter 3

Problem Statement / Requirement Specifications

Problem Statement : To create a text summarizer which summarises the text or the content of the paragraph in minimum words without changing its meaning. This system is made using NLP based model which is branch of machine learning. This text summarizer also summarizes text from the weblinks and also summarizes the copied text.

3.1 Project Planning

The steps to be followed while planning to execute the text summarization project development are:

Define the project scope: The scope of the project is to develop a text summarizer using NLP techniques, which includes an extractive approach. The project should implement three algorithms, namely Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm, to extract the summary from the input text.

Set project goals: The project goals should include developing a text summarizer that is accurate, efficient, and scalable. The summarizer should be able to handle a large volume of input data and produce summaries that capture the essential information of the input text.

Define project requirements: The project requirements should include the following:

- ❖ The summarizer should accept input text in various formats, including plain text, URL, and Document.
- ❖ The summarizer should be able to extract summaries of different lengths, depending on the user's preference.
- ❖ The summarizer should provide the option to select one of the three algorithms for summary extraction.
- ❖ The summarizer should have a user-friendly interface.

Develop a project plan: The project plan should include the following tasks:

- ❖ Collect and preprocess the input data.
- ❖ Implement the three algorithms for extractive summarization.
- ❖ Develop a user-friendly interface for the summarizer.
- ❖ Test and debug the summarizer.
- ❖ Deploy the summarizer on a server.

3.2 Project Analysis

Before beginning the text summarization project, a thorough project analysis was conducted to ensure that the project requirements were clear and that there were no ambiguities or mistakes in the project statement.

The project analysis included the following steps:

- ❖ Collecting project requirements: The project requirements were collected by analyzing the problem statement and understanding the needs of the project. The requirements were categorized as functional and non-functional requirements, and were prioritized based on their importance to the project's success.
- ❖ Identifying potential ambiguities: The project requirements were analyzed to identify any potential ambiguities or inconsistencies that could impact the project's success. Any unclear or contradictory requirements were clarified, and changes were made as necessary to ensure that all requirements were clear and consistent.
- ❖ Reviewing the project scope: The project scope was reviewed to ensure that all requirements were included and that the project was feasible within the given timeframe and budget.
- ❖ Analyzing the project risks: The project risks were identified and analyzed to determine their potential impact on the project. Risk mitigation strategies were developed to minimize the impact of any potential risks.

- ❖ Identifying project dependencies: The project dependencies were identified to ensure that all necessary resources, such as software, hardware, and personnel, were available to complete the project successfully.
- ❖ Developing a project plan: Based on the project analysis, a detailed project plan was developed that included timelines, deliverables, and milestones.

Overall, the project analysis helped to ensure that the project requirements were clear and that there were no ambiguities or mistakes in the project statement. It allowed for the identification and mitigation of potential risks and ensured that all necessary resources were available to complete the project successfully.

3.3 System Design

3.3.1 Design Constraints

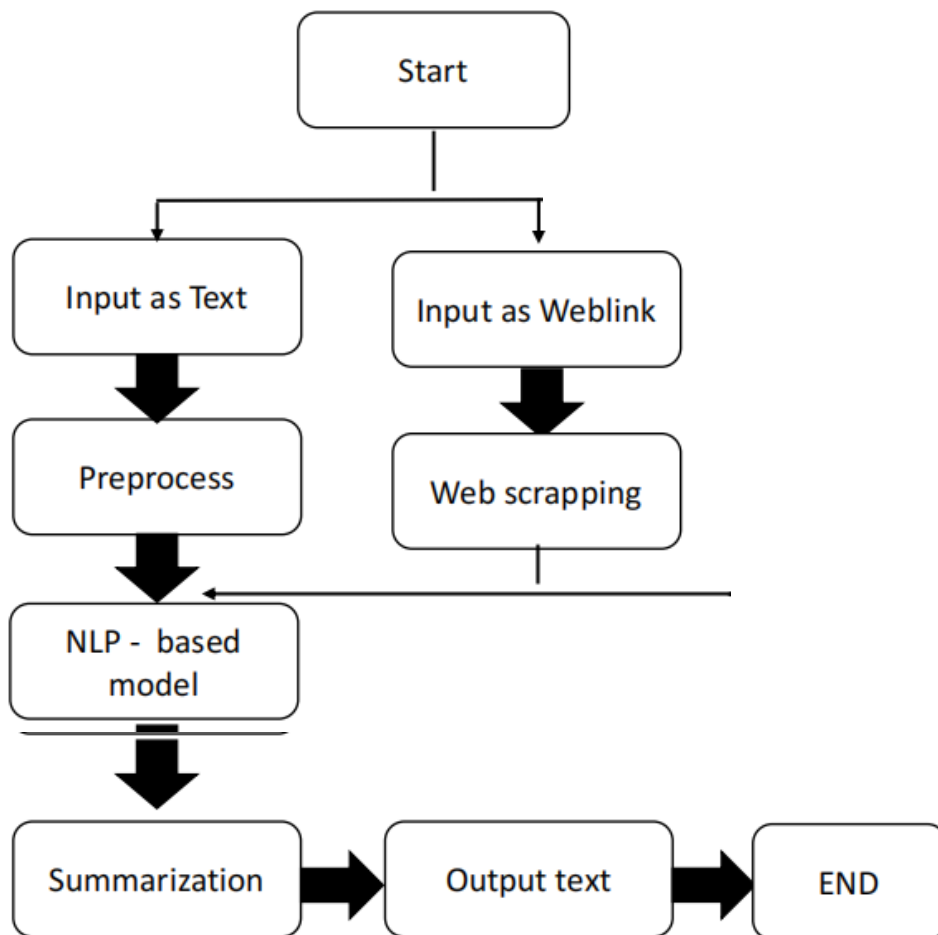
The design constraints for the text summarizer system are as follows:

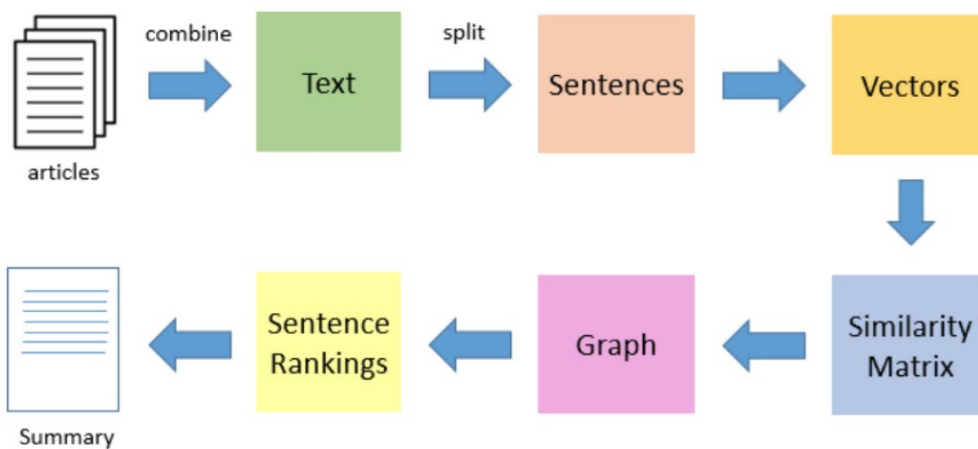
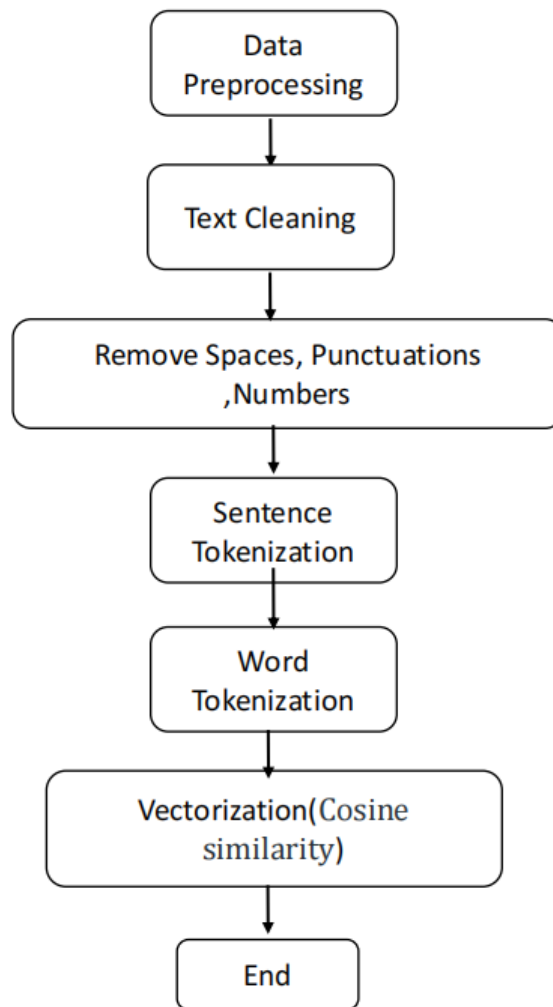
- I. Working environment: The text summarizer system must be designed to work within the given software and hardware environments. The system will be designed to work on any device that has access to a web browser and internet connectivity.
- II. Software and hardware: The system will be designed using open-source software such as Python and its libraries, including NLTK, NumPy, Pandas, and Flask. The system will be developed to work with the latest versions of these libraries to ensure optimal performance.
- III. Experimental or environmental setup: The text summarizer system will not require any experimental or environmental setup. The system will be designed to work with any input text document in English language format. The input text will be processed by the system and will generate a summary output of the text.
- IV. User interface: The system will be designed with a user-friendly interface that allows users to input the text document and get a summarized output. The user interface will be designed to be intuitive and easy to use, with features such as copy-pasting the text, upload a document, and generate the summary output.

Overall, the text summarizer system will be designed to work efficiently and effectively in the given software and hardware environments. It will be user-friendly and accessible, with an easy-to-use interface that allows users to input the text and receive a summarized output.

3.3.2 System Architecture

Block Diagram:



Model Working:

Chapter 4

Implementation

4.1 Methodology

The methodology used to complete the text summarization project involved the following steps:

- ❖ **Data Collection:** The first step in the methodology was to collect the text data that needed to be summarized. The data was collected from various sources, including news articles, research papers, and online blogs.
- ❖ **Data Preprocessing:** Once the data was collected, it was preprocessed to remove any unwanted characters, symbols, or numbers. Stop words and punctuation marks were also removed from the text.
- ❖ **Tokenization:** The text was then tokenized into individual words or phrases. This step allowed for easier analysis of the text data.
- ❖ **Word Frequency-Based Algorithm:** One of the algorithms used in the project was the Word Frequency-Based Algorithm. This algorithm involved counting the frequency of each word in the text and selecting the most frequently occurring words as the summary.
- ❖ **Cosine Similarity Algorithm:** Another algorithm used in the project was the Cosine Similarity Algorithm. This algorithm involved comparing the similarity between the text and the summary by measuring the cosine of the angle between the two vectors.
- ❖ **Luhn Algorithm:** The Luhn Algorithm was also used in the project. This algorithm involved ranking sentences based on their importance in the text and selecting the most important sentences as the summary.
- ❖ **Evaluation:** To evaluate the effectiveness of the summarization algorithms, several evaluation metrics were used. These metrics were used to measure the similarity between the summary and the original text.

4.2 Testing OR Verification Plan

In the testing phase of this project, we followed the following steps to test and verify that the project is properly working to the given standards:

- Tested all three of the methods, namely cosine function, frequency based and luhn algorithm with custom text as input.
- Tested all three of the methods, namely cosine function, frequency based and luhn algorithm with some Wikipedia links.
- Later tested all the three methods with intentionally incorrect link to receive no text as output.
- After successfully clearing all the above testing phases it is safe to assume that the project is working as intended and no major program crashing bugs exist in the code.

4.3 Result Analysis OR Screenshots

Paragraph Summarizer

Enter link to summarize:

Type or paste URL here

Enter text to summarize:

Type or paste text here

Summarize (Frequency) Summarize (Distance) Summarize (Luhn)

{% if summary %}

Summary:

{{ summary }}

{% endif %}

Paragraph Summarizer

Enter link to summarize:

Type or paste URL here

Enter text to summarize:

Artificial intelligence is human like intelligence.It is the study intelligent artificial agents. Science and engineering to produce intelligent machines. Solve problems and have intelligence. Related to intelligent behavior. Developing of reasoning machines. Learn from mistakes and successes. Artificial intelligence is related to reasoning in everyday situations.

Summarize (Frequency) Summarize (Distance) Summarize (Luhn)

Summary:

Artificial intelligence is human like intelligence. Artificial intelligence is related to reasoning in everyday situations. It is the study intelligent artificial agents.

4.4 Quality Assurance

For the text summarization project, a quality assurance test was conducted for verifying the quality of the software that followed a set of guidelines to ensure that the software was of high quality, reliable, and met the user requirements.

The quality assurance guidelines followed for the text summarization project were:

- ❖ Code review: The software code was reviewed to ensure that it was well-organized, easy to read, and free of any errors.
- ❖ Testing: As mentioned earlier, various testing standards were followed to ensure that the software performed as expected and met the user requirements.
- ❖ Documentation: The software documentation was reviewed to ensure that it was complete, accurate, and easy to understand.
- ❖ User feedback: Feedback was collected to test the software and used it to improve the software's functionality and usability.
- ❖ Compliance with standards: The software was checked to ensure that it complied with the coding standards and guidelines.

After the quality assurance test verified the software's quality, it meant that that the software passed the required quality standards. The software was of high quality and met the user requirements.

Chapter 5

Standards Adopted

5.1 Design Standards

For the text summarization project, several design standards were followed to ensure efficient and effective project development. These design standards included:

- ❖ **Software Development Life Cycle (SDLC):** The text summarization project followed the SDLC approach, which involves several phases, including planning, analysis, design, implementation, and maintenance. This approach ensured that the project was developed systematically and met the requirements of the users.
- ❖ **Unified Modeling Language (UML):** The UML diagrams were used in the design phase of the project to model the system's architecture and its components. These diagrams helped to visualize the system's structure and relationships and made it easier to understand the system's functionality.
- ❖ **Code Documentation:** Proper documentation was maintained for the source code of the project. This documentation included comments, function descriptions, and code structure. The documentation helped in maintaining and updating the code in the future.
- ❖ **User Interface (UI) Design:** The UI design of the text summarizer followed industry standards, including a simple and user-friendly interface that allowed users to input text and view the summary. The design was optimized for ease of use and efficient performance.
- ❖ **Software Libraries:** The text summarization project utilized various software libraries, including Natural Language Processing (NLP) libraries such as NLTK and spaCy. These libraries helped in the efficient processing of text data and the implementation of summarization algorithms.

By following these design standards, the text summarization project was developed with efficiency, maintainability, and user-friendliness in mind.

5.2 Coding Standards

For the text summarization project, several coding standards were followed to ensure that the code was well-organized, easy to read, and maintainable. These coding standards included:

- ❖ **Writing concise code:** The code was kept as short and simple as possible to make it easier to understand and maintain.
- ❖ **Naming conventions:** Appropriate naming conventions were used for variables, functions, and classes, making it easier to understand the purpose of each element.
- ❖ **Commenting:** The code was well-commented to provide clarity and understanding to other developers who might have to work on the code in the future.
- ❖ **Indentation and block segmentation:** Indentation was used to mark the beginning and end of control structures such as loops and conditionals. This made it easier to understand the code's logic. Similarly, code blocks in the same section were segmented into paragraphs to make it easier to read and understand.
- ❖ **Function length:** Functions were kept as short as possible, ideally performing a single task. This made it easier to understand and test individual functions.

By following these coding standards, the text summarization code was well-organized, efficient, and easy to maintain. This ensured that the code was of high quality and could be easily understood and updated in the future.

5.3 Testing Standards

For the text summarization project, several testing standards were followed to ensure that the software was of high quality, reliable, and performed as expected. The testing standards followed were:

- ❖ **ISO 25010:** This standard specifies the quality requirements for software products and systems, including functional suitability, performance efficiency, compatibility, and security.

- ❖ IEEE 829: This standard outlines the test documentation that should be produced during the testing process, including test plans, test cases, and test reports.
- ❖ Black-box testing: This testing approach was used to test the functionality of the software without looking at the code. Various input/output combinations were tested to ensure that the software performed as expected.
- ❖ White-box testing: This approach was used to test the code's internal logic and structure. The code was reviewed to ensure that it was well-organized, easy to read, and free of any errors.
- ❖ Unit testing: This approach was used to test individual functions and modules of the code. Various test cases were designed to test the functions' functionality and ensure that they were working as expected.

By following these testing standards, the text summarization software was thoroughly tested and verified to ensure that it met the required quality standards and performed as expected. Any issues or bugs found during the testing process were resolved before the final release of the software.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

In conclusion, the text summarization project using NLP techniques with an extractive approach has the potential to be a useful tool in various fields. Text summarization is a fast-growing field in natural language processing, and it has the potential to revolutionize the way we consume and process information. The project aims to implement three algorithms, namely Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm, to extract the summary from the input text. The project requirements include the ability to accept input text in various formats and have a user-friendly interface.

To execute the project development, a proper project plan should be defined, including the tasks of collecting and preprocessing input data, implementing the algorithms for extractive summarization, evaluating the performance of the summarizer, developing a user-friendly interface, testing and debugging the summarizer, and deploying the summarizer on a server.

By adhering to the design standards in software engineering, such as UML, object-oriented design principles, agile development, software design patterns, database design standards, and code quality standards, the project can ensure the quality, reliability, and efficiency of the final product.

Overall, this project has the potential to be a valuable tool in various fields where summarization of large volumes of text data is required.

6.2 Future Scope

The text summarization project using NLP techniques with an extractive approach has significant potential for future developments and improvements. Here are some potential areas for future scope:

- □ Implementing other extractive summarization algorithms: While the Word Frequency-based algorithm, Cosine Similarity algorithm, and Luhn algorithm are effective, there are other algorithms available that can improve the accuracy and effectiveness of the summarization process. Implementing these algorithms can enhance the summarizer's performance.
- □ Incorporating abstractive summarization techniques: Abstractive summarization techniques involve generating a summary by paraphrasing and rewording the original text, unlike the extractive approach that selects and combines sentences from the original text. Incorporating abstractive summarization techniques can improve the quality and relevance of the generated summary.
- □ Developing a multi-lingual summarizer: The ability to summarize input text in multiple languages can significantly expand the scope and usefulness of the summarizer.
- □ Applying deep learning techniques: Deep learning techniques, such as neural networks, can improve the performance of the summarizer by allowing it to learn from a large corpus of text and generate more accurate summaries.
- □ Implementing a summarization evaluation framework: Developing an evaluation framework to measure the summarizer's accuracy and effectiveness can provide insights into its strengths and weaknesses and help in further improvements.

Overall, there is significant potential for future development and improvement of the text summarization project, making it a promising area of research and development in the field of NLP.

References

Journals / Conference Papers:

- o [1] Sinha, Aakash, Abhishek Yadav, and Akshay Gahlot. "Extractive text summarization using neural networks." arXiv preprint arXiv:1802.10137 (2018).
- o [2] Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." arXiv preprint arXiv:1704.04368 (2017).
- o [3] Liu, Linqing, et al. "Generative adversarial network for abstractive text summarization." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 32. No. 1. 2018.
- o [4] Blei, David M., Andrew Y. Ng, and Michael I. Jordan. "Latent dirichlet allocation." Journal of machine Learning research 3.Jan (2003): 993-1022.

Weblinks:

- o [5] Comprehensive Guide to Text Summarization using Deep Learning in Python :
<https://www.analyticsvidhya.com/blog/2019/06/comprehensive-guidetext-summarization-using-deep-learning-python/>
- o [6] Text Summarization using Machine Learning :
<https://data-flair.training/blogs/machine-learning-text-summarization/>
- o [7] Approaches to Text Summarization: An Overview :
<https://www.kdnuggets.com/2019/01/approaches-text-summarizationoverview.html>
- o [8] "Text summarization approaches for NLP :
<https://www.machinelearningplus.com/nlp/text-summarizationapproaches-nlp-example/>

INDIVIDUAL CONTRIBUTION REPORT:

Text Summarization

MAYANK TIWARI
20051847

Abstract: The objective of the project is to understand the concepts of natural language processing and creating a tool for text summarization. The concern in automatic summarization is increasing broadly so the manual work is removed. People need to learn much from texts. But they want to spend less time while doing this. It aims to solve this problem by supplying them the summaries of the text from which they want to gain information. The project concentrates creating a tool which automatically summarizes the document.

Individual contribution and findings: My individual contribution in creating this project is that I worked upon the Frequency based algorithm of the extractive process of text Summarization. What I aimed to produce was a good example of text summarizer which will summarize text using the frequency based algorithm.

Individual contribution to project report preparation: I have contributed in Chapter 1 and 2 of this report which includes Introduction and Basic Concepts/ Literature Review.

Individual contribution for project presentation and demonstration: I have contributed in Slides 2 and 4 which includes the abstract of our project and introduction to it.

Full Signature of Supervisor:
.....

Full signature of the student:
.....

INDIVIDUAL CONTRIBUTION REPORT:

Text Summarization

KAUSTUBH BALA
20051860

Abstract: The objective of the project is to understand the concepts of natural language processing and creating a tool for text summarization. The concern in automatic summarization is increasing broadly so the manual work is removed. People need to learn much from texts. But they want to spend less time while doing this. It aims to solve this problem by supplying them the summaries of the text from which they want to gain information. The project concentrates creating a tool which automatically summarizes the document.

Individual contribution and findings: I contributed in the cosine algorithm of the text summarization, I aimed to properly execute the algorithm such that the input from the user will be properly summarized using the cosine similarity algorithm, and the user will be presented with the summarized text.

Individual contribution to project report preparation: My contribution is in Chapter 3 of this report which is Problem Statement and Requirement Specifications.

Individual contribution for project presentation and demonstration: I have made the slides numbered 5 and 6, in which the extractive approach to text summarization and the frequency based algorithm is discussed.

Full Signature of Supervisor:
.....

Full signature of the student:
.....

INDIVIDUAL CONTRIBUTION REPORT:

Text Summarization

KAIWALYA DESHPANDE
20051862

Abstract: The objective of the project is to understand the concepts of natural language processing and creating a tool for text summarization. The concern in automatic summarization is increasing broadly so the manual work is removed. People need to learn much from texts. But they want to spend less time while doing this. It aims to solve this problem by supplying them the summaries of the text from which they want to gain information. The project concentrates creating a tool which automatically summarizes the document.

Individual contribution and findings: I basically contributed in the implementation of the Luhn Algorithm for the project, this algorithm is named after the creator of the algorithm, and coding this algorithm into the project such that the function works as intended and we are able to compare all the algorithms.

Individual contribution to project report preparation: I have contributed in Implementation part of this report which is Chapter 4.

Individual contribution for project presentation and demonstration: I have contributed in the slides 7 and 8 in which the remaining two algorithms, namely Luhn Algorithm and Cosine Similarity Algorithms are discussed.

Full Signature of Supervisor:
.....

Full signature of the student:
.....

INDIVIDUAL CONTRIBUTION REPORT:

Text Summarization

KUMAR ARYAN
20051879

Abstract: The objective of the project is to understand the concepts of natural language processing and creating a tool for text summarization. The concern in automatic summarization is increasing broadly so the manual work is removed. People need to learn much from texts. But they want to spend less time while doing this. It aims to solve this problem by supplying them the summaries of the text from which they want to gain information. The project concentrates creating a tool which automatically summarizes the document.

Individual contribution and findings: I managed the frontend part of the project, I created the website which is visible to the user at face, and also connected the backend and frontend together such that all the three algorithms are able to be presented to the user in a beautiful graphical user-Interface.

Individual contribution to project report preparation: My contribution is in Chapter 5 and 6 of this report which includes Standard adopted and Conclusion/Future Scope.

Individual contribution for project presentation and demonstration: My contribution is in slides 9 and 12, in which the three algorithms are compared and the whole project is concluded.

Full Signature of Supervisor:
.....

Full signature of the student:
.....