**A WEB-BASED DOCUMENT SUMMARIZING AND TOPIC PREDICTION SYSTEM USING NATURAL LANGUAGE PROCESSING.**

**BY:**

**EBEREONWU EINSTEIN MUNACHISO 19/0512**

**SHOYEMI OLASUBOMI TUTULORO 19/1167**

**UDO DAVID DAVID 19/1326**

**A PROJECT WORK SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE B.Sc. (HONS) IN SOFTWARE ENGINEERING**

**SUBMITTED TO:**

**THE DEPARTMENT OF SOFTWARE ENGINEERING**

**SCHOOL OF COMPUTING AND ENGINEERING SCIENCES**

**BABCOCK UNIVERSITY, ILISAN REMO**

**OGUN STATE, NIGERIA**

**MAY 2023**

**DECLARATION**

We declare that the project work, “A WEB-BASED DOCUMENT SUMMARIZING AND TOPIC PREDICTION SYSTEM USING NATURAL LANGUAGE PROCESSING” was carried out by the following people

**EBEREONWU EINSTEIN MUNACHISO 19/0512 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SIGNATURE/ DATE**

**SHOYEMI OLASUBOMI TUTULORO 19/1167 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SIGNATURE/ DATE**

**UDO DAVID DAVID 19/1326 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**SIGNATURE/ DATE**

**CERTIFICATION**

This is to certify that this project titled“A WEB-BASED DOCUMENT SUMMARIZING AND TOPIC PREDICTION SYSTEM USING NATURAL LANGUAGE PROCESSING” was carried out by the following students under the supervision of the Department of Software Engineering, Babcock University, Ilishan-Remo, Ogun State, Nigeria:

**EBEREONWU EINSTEIN MUNACHISO 19/0512** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**SHOYEMI OLASUBOMI TUTULORO 19/1167** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**UDO DAVID DAVID 19/1326** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DR. ADIGUN TAIWO DATE**

**PROJECT SUPERVISOR**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**EXTERNAL EXAMINER DATE**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DR. ADEKOLA OLUBUKOLA DATE**

**(HEAD OF SOFTWARE ENGINEERING DEPARTMENT)**

**DEDICATION**

This project is dedicated to God almighty, who has led us through our four years in his own university; Babcock and has granted us the grace to produce this project which is a proof of knowledge gained at our time in the university. We also dedicate this project to all our parents, family and friends who have added to our growth and knowledge in one way or the other throughout our 4 years of studies.

**ACKNOWLEDGEMENT**

We thank God for his guidance, protection and intervention in our lives throughout our journey in Babcock University and especially for the grace to overcome all challenges faced in our every endeavors. We sincerely appreciate our supervisor Dr. Adigun Taiwo for all his guidance and support throughout the creation of this project.

We are truly grateful to all our lecturers who have impacted in us, the knowledge and skills which we put into practice during the several stages of our project build. Lecturers such as Dr. Adigun Taiwo who is also our project supervisor, Dr. Adetunji Oluwatofunmi our able course advisor who taught us “Software Security Engineering, Introduction to Professional Ethics and Practice”, Professor Sunday Idowu who taught us “Introduction to Computer Science and Programming”, Dr. Maitanmi S. who taught us “Introduction to Web Technology and development”, Mr. Otuneme who taught us “Object Oriented Software Development”, Dr. Jet Akinsola who taught us “Algorithms and Data Structures”, Dr. Wunmi Ajayi who taught us “Software Requirements Engineering and Construction and Software Engineering Economics”, and all other lecturers whose names are not mentioned.

We also give a special thanks to our respective parents Dr. and Mrs. Ebereonwu, Mr. and Mrs. Udena, Mr. and Mrs. Shoyemi for their continual dedication to us and support throughout.

**ABSTRACT**

Students, lecturers, accountants, businessmen & women, politicians, researchers, and many more individuals of various backgrounds not listed above often have to read through large volumes of information or produce a summarized version of documents or other bodies of text. Such a task can easily be achieved with small volumes of text but becomes a hassle when individuals have to read through and summarize large volumes of text. For such a reason, the system “A WEB-BASED DOCUMENT SUMMARIZING AND TOPIC PREDICTION SYSTEM USING NATURAL LANGUAGE PROCESSING” was built. It provides a top-notch topic modeling feature where the system detects the topic of the body of text and displays it to the user, and a language detection feature where the system can identify the language of the text entered before providing an appropriate summary. In an event where a language is selected, yet the text entered into the system is of a different language, it displays an alert asking the user to choose the appropriate language.

It focuses on a particular field of AI called Natural Language Processing (NLP) for the summarization and language detection aspect of the system, the UI/UX was designed using Figma and implemented using HTML, CSS, and JavaScript, and the server side was built with Python using the Flask framework.

The project initially accepted only written texts, could only summarize English text and didn’t have the language detection feature. However, new feature recommendations were made and were added: document summarizing. This feature allows a user to select a word document

(.docx) or a text document (.txt) file and summarizes the content of such file. A download summary feature lets a user download the summary as a word document (.docx). A language detection/ language modeling feature allow user detect what language the body of text for summary is written in and finally, a text-to-speech feature which enables bling users to be able to hear their summary once it has been produced.

**Keywords: AI, NLP, Topic Prediction, Language Detection, Text-to-Speech.**

**Word count: 322.**

**TABLE OF CONTENT**

DECLARATION…………………………………………………………………...i

CERTIFICATION………………………………………………………………....ii

DEDICATION…………………………………………………………………….iii

ACKNOWLEDGEMENT…………………………………………………………iv

ABSTRACT……………………………………………………………………......v

LIST OF FIGURES……………………………………………………………….vi

1. CHAPTER ONE: INTRODUCTION..................................................................1
   1. BACKGROUND OF STUDY……………………………………………..1
   2. STATEMENT OF PROBLEM…………………………………………….2
   3. AIM AND OBJECTIVES OF THE STUDY…….………………………...3
   4. METHODOLOGY………………………………………………………....4
   5. SCOPE OF STUDY………………………………………………………..4
   6. SIGNIFICANCE OF STUDY……………………………………………...5
   7. RESEARCH JUSTIFICATION…………………………………………....5
   8. DEFINITION OF TERMS………………………………………………....6
   9. ORGANIZATION OF THE PROJECT…………………………………….7
2. CHAPTER TWO: LITERATURE…………………………………………….
   1. INTRODUCTION………………………………………………………
   2. HISTORICAL BACKGROUND OF THE RESARCH…………………..
      1. AI TEXT SUMMARIZATION IN THE 1950S AND 1960S …….
      2. AI TEXT SUMMARIZATION IN THE 1970S AND 1980S……
      3. AI TEXT SUMMARIZATION SINCE THE 1990S TILL PRESENT..
         1. FIRST PERIOD: SUMMARIZING BY EMPIRICAL METHODS……………………………………
         2. SECOND PERIOD: MACHINE LEARNING BASED APPROACH………………………………….
         3. THIRD PERIOD: DEEP LEARNING BASED APPROACH….
   3. REVIEW OF RELATED WORKS……………………………………
      1. A GRAPH BASED APPROACH FOR KEYWORD EXTRACTION FROM DOCUMENTS………………………………………..
      2. DOCUMENT SUMMARIZATION USING TEXTRANK AND SEMANTIC NETWORK……………………………………
      3. IMPROVING EXTRACTIVE DOCUMENT SUMMARIZATION WITH SENTENCE CENTRALITY……………………………
      4. ENHANCING BIOMEDICAL TEXT SUMMARIZATION USING SEMANTIC RELATION EXTRACTION………………………….
      5. DOCUMENT SUMMARIZATION BASED ON DATA RECONSTRUCTION
      6. EXTRACTION-BASED SINGLE-DOCUMENT SUMMARIZATION USING RANDOM INDEXING
      7. ARTIFICIAL INTELLIGENCE FOR AUTOMATIC TEXT SUMMARIZATION
      8. ABSTRACTIVE DOCUMENT SUMMARIZATION WITHOUT PARALLEL DATA
      9. A HYBRID MACHINE LEARNING MODEL FOR MULTI-DOCUMENT SUMMARIZATION
      10. USER-ORIENTED DOCUMENT SUMMARIZATION THROUGH VISION-BASED EYE-TRACKING
      11. SUMMARY OF REVIEWED LITERATURES…………
3. CHAPTER THREE: SYSTEM ANALYSIS AND DESIGN
   1. INTRODUCTION
   2. DESCRIPTION OF THE SYSTEM
   3. REQUIREMENT SPECIFICATIONS
      1. USER REQUIREMENTS AND SYSTEM REQUIREMENTS.
      2. FUNCTIONAL REQUIREMENTS AND NON-FUNCTIONAL REQUIREMENTS.
   4. DESIGN MODEL
   5. DESIGN AND DEVELOPMENT TOOLS
   6. DIAGRAMS
      1. USE CASE DIAGRAM
      2. ACTIVITY DIAGRAM
4. CHAPTER FOUR: IMPLEMENTATION
5. CHAPTER FIVE: SUMMARY, CONCLUSION AND
   1. RECOMMENDATION….
      1. REFERENCES
   2. APPENDICES

**CHAPTER ONE: INTRODUCTION**

1. **BACKGROUND OF THE STUDY**

In this modern world, technology is constantly increasing the efficiency, feasibility and ease of ways to carry out time consuming and brain tasking activities. With the increasing amount of information being generated through technology and the need for quick information processing, extracting the key information embedded in large body of text has become almost unachievable for individuals from various backgrounds such as education, medicine, tourism, law, and many more.

However, because of technological advancements and the existence of Artificial Intelligence (AI), with a focus on Natural Language Processing (NLP), the work of summarizing has become a very seamless task; systems are now very capable of churning large bodies of text while also withholding the semantics. Formerly, individuals had to read through multiple lines of text, absorb and digest the information it contained before proceeding to write a summary of essential points included in the original body of text or take any more actions, but due to the advancements of AI, this is no longer the case.

Artificial intelligence (AI) is the capacity of a machine controlled by a computer to perform jobs that generally necessitate human intelligence and judgement. (B.J. Copeland, 2022). AI is an advanced field which is more than capable of performing several tasks in areas such as medicine, education, finance, e.t.c

Natural Language Processing (NLP) is a subfield of Artificial Intelligence which came into existence in the middle of 20th century it enables computers to interpret spoken words or written texts in a way comparable to that of humans (IBM Cloud Education, 2022). NLP has been used to achieve several ground-breaking achievements including but not limited to text translation from one language to another, development of chat bots, text summarization as in this case, and personal-assistants such as Apple's Siri, Amazon's Alexa, and Google's Google Assistant, which are capable of having real-time non-human controlled conversations with individuals, carrying out specific tasks such as setting a reminder for 12:30, calling a friend, texting a friend, and much more can all be achieved simply by asking a home assistant which are able to understand humans due to their AI integration. Therefore, this study will apply NLP in order to determine the key content of any body of text which will be used to coin out a proper summary of the text entered and attempt to predict the text topic of the body of text which will help curb the issue of having to consume a lot of unnecessary information before getting the key points required of a body of text.

**1.2 STATEMENT OF PROBLEM**

The vastness of the information available online has brought about several pros and cons, one of which this research wishes to address. The issue of key point filtering in large bodies of text is one which is faced by researchers in areas such as finance, education, humanities, media and communication, business and management e.t.c. Researchers often times find themselves reading through lines, paragraphs, pages or chapters of documents before finally coming across the key points which will help them with their

research work. Most already existing systems are only capable of summarizing texts in languages if users specify and lack the Text-To-Speech feature, this system aims to bridge the gap between what exists (other systems i.e [Summarizer.org](https://www.summarizer.org/)) and what should exist (this system) by providing the following extra features; auto-language detection and reading the summary should the user want.

* 1. **AIM AND OBJECTIVES OF THE STUDY**

The aim of this study is to develop and implement a web-based document summarizing and topic prediction system using natural language processing which is able to provide summary, predict topic, convert text summary to speech. While the specific objectives of the system are:

1. To develop and integrate a text summarizing, language detection model.
2. To implement the text summarizing and language detection model.
3. To conduct a system evaluation.

**1.4 METHODOLOGY**

This study uses the methodologies below to achieve the afore mentioned objectives:

1. An NLP text summarizing model will be built using the python NLTK library. A language detection model will also be trained using already existing language data.
2. The built models will be integrated to the system including other features such as topic prediction and text-to speech using the python gTTS library.
3. The system will be tested by multiple users against a variety of test cases in order to determine the system’s efficiency, including the responsiveness of the web app interface.

**1.5 SCOPE OF STUDY**

This study focuses on developing a web-based document summarizing and topic prediction system using natural language processing. The system will also be able to suggest the topic of a body of text, detect text language, and a read aloud/text-to-speech feature. However, the project scope will be limited to the following:

1. Summarizing documents: It will summarize documents written in a few languages, not all.
2. Detecting languages: It will detect a select few numbers of languages; languages that were sampled in the training data.
3. Topic prediction: The accuracy of this feature depends on the size of text.
4. Only written text, .txt files and .docx file contents can be summarized directly by the system.
5. Summaries can only be downloaded as .docx files.

**1.6 SIGNIFICANCE OF STUDY**

The study focuses on developing an intelligent system that is able to quickly produce un-biased, un-ambiguous and relevant summaries (in a select few languages) for individuals at any point in time, show individuals the topic of discussion of any body of text in order to aid individuals decide if it is truly worth reading or not, provide individuals the ability to detect the language in which a given body of text is written in and finally speech-to-text to increase system usability especially for blind people.

Thanks to the advent of AI, the project which is of immerse importance to academics, research and other fields will significantly reduce the amount of time it takes to summarize any given body of text therefore providing individuals the key information alone and giving them more time to focus on the necessary information and be more efficient in carrying out the intended post-summary activities with the key information.

**1.7 RESEARCH JUSTIFICATION**

Advantages of the proposed system over the previous system are as follows:

1. The system provides text-to-speech feature
2. The system provides top 2 possible topics of the body of text.
3. The system features are totally free for all.
4. The system allows for language detection

**1.8 DEFINITION OF TERMS**

**AI:** Short for Artificial Intelligence is the ability of a machine to perform tasks thought to require human intelligence. (B.J. Copeland, 2022)

**NLP:** Short for Natural language processing refers to the branch of computer science and more specifically, the branch of artificial intelligence concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. (IBM Cloud Education, 2020).

**HTML:** Short for Hypertext Markup Language, is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. (Wikipedia, 1993)

**CSS:** Short for Cascading Style Sheets is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

**JavaScript: J**avaScript is a scripting language for creating dynamic web page content. It creates elements for improving site visitors’ interaction with web pages, such as dropdown menus, animated graphics, and dynamic background colors. (**Jordana A., 2022**)

**Python:** Python is a high-level programming language designed by Guido van Rossum which supports multiple programming paradigms such as Object-Oriented Programming (OOP), procedural programming. It can be used for several activities such as backend development, data analytics, machine learning, general scripting tasks and virtually anything one can think of.

**Flask:** Flask is what’s known as a WSGI framework. Mercifully pronounced “whiskey,” this stands for ​​Web Server Gateway Interface. Essentially, this is a way for web servers to pass requests to web applications or frameworks. Flask relies on the WSGI external library to function, as well as the Jinja2 template engine. (Derry, 2022)

**1.9 ORGANIZATION OF THE PROJECT**

This chapter has successfully introduced the web-based text/document summarizing and topic modeling system: an AI (NLP) approach to summarizing text including its inner workings, aims and objectives, methodology, scope of study and overall, the significance of the study.

**Chapter two** covers the literature review, it acknowledges previously existing work related to this study, their literature review, cited works and the shortcomings of the related studies including how this research can improve on such shortcoming.

**Chapter three** expatiates on the systems design methodology, its functional, non-functional, user and system requirements, logic flow which includes all necessary diagrams, design and finally, development tools used to build the system

**Chapter four** contains everything that has to do with system design implementation, testing strategies, minimum system requirements, software maintenance and evolution and a lot more

**Chapter five** discusses the summary, recommendations and conclusion aspect of the research work including how it can be applied in real world scenarios

**CHAPTER TWO: LITERATURE**

**2.1 INTRODUCTION**

The increasing number of available resources, documents online due to the advancement of technology is starting to become a burden to individuals as it leaves people overwhelmed and confused as to what resources to consume, considering the available time and true relevance of the resource.

A summary is defined by the Britannica dictionary as “a brief statement that gives the most important information about something” (Britannica) and by the Oxford Learner’s dictionary as “a short statement that gives only the main points of something, not the details

summary (of something)” (Oxfore Learners Dictionaries). The main purpose of an automated document summarizing system is to create an un-biased, un-ambiguous and relevant version of the original text while still fully capturing the notion of the original text.

Over the years, several strategies such as

* **Deletion:** Producing a summary simply by removing unnecessary information in the sentence of the source text.
* **Sentence Combination:** Producing a summary simply by combining multiple sentences of phrases from the original text together.
* **Generalization:** Producing a summary simply by replacing a general term from a list. Example, replacing; Nigeria, Ghana, America with the term countries.
* **Paraphrasing:** Producing a summary simply by replacing words from the original text with their synonyms.(Abdi et al., 2016b).

The two general text summarization techniques are; Extractive and Abstractive

The **Extractive text summarization technique** entails identifying various key areas of the text and reproducing them word for word, resulting in a subset of the original text's sentences. (Prasasthy, 2021).

The **Abstractive text summarization technique** however is a technique whereby upon interpreting and examining the original text, advanced natural language techniques are used to generate a new shorter text that conveys the most critical information from the original one. ([Prasasthy](https://medium.com/@prasasthy.sanal?source=post_page-----9d1b3787a707--------------------------------), 2021).

**2.2.** HISTORICAL BACKGROUND OF THE RESARCH

This section provides a brief history of automatic text summarization research, as well as some approaches used to automate text summarization over the years.

**2.2.1** AI TEXT SUMMARIZATION IN THE 1950S AND 1960S

Text summarizing research can be traced back to the onset of Artificial Intelligence (AI). Such times were referred to as the “early enthusiasm, great expectations” (Stuart Jonathan Russell & Peter Norvig, 2009). The very first attempt at automated text summarization is credited to Luhn (Luhn, 1958). He discovered that statistical data derived from word frequencies can be used to identify the significance of sentences in a text corpus. Another very significant attempt was made by Edmundson (Edmundson, 1969) who highlighted the fact that relying solely on word frequencies for identifying important sentences in texts is insufficient. He also suggested that a combination of other factors be considered, such as the presence of predefined cue words in a sentence that which can increase or decrease the sentence's

importance, whether the sentence contains words from the title, and the location of the sentence in the document or paragraph. These two papers are of high significant in the field because the features used by Luhn (Luhn, 1958) and Edmundson (Edmundson, 1969) to identify important sentences are still being used in some form or another by current summarisation methods. Furthermore, Edmundson was the first to assess how different features influence the resulting summaries, a process that is now routinely used in machine learning approaches. Despite the upbeat tone of both papers, progress in the field has been slower than expected. The limitations imposed by the hardware available at the time played a significant role in this: According to Luhn (Luhn, 1958), all texts had to be punched on cards before they could be processed while Due to limited computer storage, Edmundson was unable to process texts with more than 4,000 words.

**2.2.2**. AI TEXT SUMMARIZATION IN THE 1970S AND 1980S

The two publications mentioned above are both empirical approaches to producing summaries, whereas the 1970s and 1980s witnessed a shift in research toward rationalist approaches. During the 1970s and 1980s, the primary focus of artificial intelligence research was on the development of methods that relied heavily on information about the problem to be solved and, in some cases, attempted to solve problems as a human would (Stuart Jonathan Russell & Peter Norvig, 2009). Several times, the information was dependent on the domain and this limited the suitability of such methods. Despite a decline in the use of rationalist approaches in the early 1990s, researchers continued to develop methods inspired by this paradigm in order to

generate abstracts. Typically, these approaches include a Natural Language Generation (NLG) module which generates text from some form of internal representation (REITER & DALE, 1997). Typically, robust information extraction methods such as LaSIE (Gaizauskas and Humphreys, 1997) and InfoXtract (SRIHARI et al., 2008) are used to fill in templates rather than domain-specific extraction rules to gain an understanding of the source.

**2.2.3.** AI TEXT SUMMARIZATION SINCE THE 1990S TILL PRESENT

Since the 1990 till date, several approaches have been adopted in attempts to summarize texts. Such approaches can be broken down into 3 distinct periods namely;

* Period 1: Summarizing by Empirical Methods.
* Period 2: Machine Learning Based Approach.
* Period 3: Deep Learning Based Approach.

**2.2.3.1 SUMMARIZING BY EMPIRICAL METHODS:**

The empirical method makes use of observation about the properties of the input or by application of linguistic theories. To identify the most appropriate sentences for a summary, researchers used information retrieval methods that calculate links between texts and parts of texts (Salton, Singhal, Mitra, and Buckley, 1997) or relied on graph-based ranking models (Rada Mihalcea & Paul Tarau, 2004). Analysis of anaphoric and coreferential links in texts (ANDO et al., 2005) or lexical repetition in texts (Barzilay & Elhadad, 1999) was also used to calculate a score for all sentences in texts and extract only those with the highest score. Rhetorical Structure Theory (RST) (MANN & THOMPSON, 1988) a theory that organizes

text in primarily non-overlapping spans linked by rhetorical relations, has been successfully applied to the development of heuristics for selecting the most pertinent sentences for a summary (Alonso i Alemany & Fuentes Fort, 2003).

The main disadvantage of these methods is that they rely on researchers' intuitions about how to evaluate the importance of a sentence and employ approximations to implement complex linguistic theories such as RST. As a result, the summaries produced are not always of high quality. These methods were largely preferred during the early stages of the field's re-emergence, but they became less popular after the year 2000. However, successful applications of these approaches can be found later on, such as in (Lloret & Palomar, 2012). Furthermore, these methods are still used to generate features for machine learning-based summarization approaches and methods like TextRank. (Rada Mihalcea & Paul Tarau, 2004) are still used as baselines.

**2.2.3.2 MACHINE LEARNING BASED APPROACH**

The main feature of the second period is the use of machine learning approaches to generate summaries. The concept of training a classifier capable of identifying which sentences should be included in a summary was first used in (Kupiec et al., 1995), but it was not widely used until annotated corpora became available. The corpora developed in the DUC, as well as the automatic evaluation metrics proposed in the DUC, had the greatest impact on the community because they enabled direct comparison of methods. Prior to the availability of these corpora, researchers had to create their own annotated resources, which were often tailored to the research questions they were attempting to answer (Simone Teufel, 1997). Researchers tried nearly every machine learning algorithm available for years in an attempt to produce better summaries. Bayesian classifiers (Neto et al., 2002)

and decision trees (Neto et al., 2002) are among the methods tested, as are hidden Markov models (Conroy et al, 2001) and integer linear programming (Luo et al., 2018). (Knight & Marcu, 2002) modify the noisy channel used in Statistical Machine Translation to create a method for sentence compression, which is seen as a first step toward automatically producing summaries. In (NASERASADI et al., 2019), the summarization process was also viewed as an optimization problem in which weights are learned from data.

**2.2.3.3 DEEP LEARNING BASED APPROACH**

Traditional machine learning approaches are still used, but they are gradually being replaced by deep learning methods. The introduction of neural approaches for text summarization, which occurred around 2015, marked the beginning of the third period. This reflects changes in other fields of computational linguistics where the use of deep learning technologies has resulted in new and more accurate methods. There are numerous papers available now in which researchers attempt to improve on the state of the art by applying the most recent neural models for automatic text summarization. In some cases, the use of neural architectures is not entirely justified because the improvements are minor; however, when used correctly, the new methods allow researchers to produce better results.

Majority of the approaches developed in neural machine translation inspired the majority of the methods proposed in automatic summarization. Unlike NMT and machine translation (MT) in general, the output of a summarization system is much shorter than the source and does not always depend on its length. This presents some difficulties when adapting NMT methods. Furthermore, the summarization process results in information loss (ideally unimportant

information), whereas MT should produce an accurate representation of the source with no information loss. This complicates the processing even more, necessitating the incorporation of a method for determining the most important information in the source into the neural architecture.

Deep learning techniques are also used for extractive summarization. (Kobayashi et al., 2015) and (Yogatama et al., 2015) present two approaches that use the semantic information provided by word embeddings to propose unsupervised optimization algorithms for finding the best set of sentences given the search space created by the semantic representation of the sentences in the document.

This section has only presented a small sample of the new methods for producing summaries using neural networks. As with traditional ML-based methods, researchers try any new neural architecture that proves useful in other fields in the hope that it will lead to better summaries, which is why it is expected that research in this field will continue.

**2.3. REVIEW OF RELATED WORKS**

**2.3.1 A GRAPH BASED APPROACH FOR KEYWORD EXTRACTION FROM DOCUMENTS**

This study describes an unsupervised method for extracting keywords from a document. Summarizing a research paper is a time-consuming and difficult task. Keywords are used to make document summarization much easier. Keywords aid in the summarization of large amounts of textual data. Keyword extraction is the process of selecting a set of words that provide context for the entire document. The graph-based method combines the RAKE algorithm and Keyword Extraction using Collective Node Weight (KECNW), in which candidate keywords are extracted using the RAKE algorithm and keywords are selected using the KECNW model.

The final result of this method is a list of keywords extracted from a document, along with their rank. (Anjali et al., 2019).

**2.3.2. DOCUMENT SUMMARIZATION USING TEXTRANK AND SEMANTIC NETWORK**

The research used text rank algorithms, Semantic Networks, and Corpus Statistics to implement a document summarizing system. Using text rank, you can extract the main phrases of a document and use them as sentences in the summary output. text rank is made up of several processes, including sentence tokenization, graph formation, edge value calculation algorithms based on Semantic Networks and Corpus Statistics, vertex value calculation, vertex value sorting, and summary generation. To measure the quality of the system output, the recall, precision, and F-Score of the summary were calculated using ROUGE-N methods. The style of writing, the selection of words and symbols in the document, and the length all have an impact on the quality of the summaries produced. (Ashari & Riasetiawan, 2017).

**2.3.3. IMPROVING EXTRACTIVE DOCUMENT SUMMARIZATION WITH SENTENCE CENTRALITY**

Extractive document summarization (EDS) is commonly regarded as a sequence labeling task in which sentences from a document are extracted one by one to form a summary. Separating sentences, on the other hand, ignores the relationship between the sentences and the documents. One solution is to use sentence position information to improve sentence representation, but this will result in the sentence-leading bias problem, which is particularly noticeable in news

datasets. To address these two issues, novel sentence centrality has been proposed for the EDS task in this paper. The sentence centrality is based on directed graphs, and it reflects both the sentence-document relationship and the sentence position information in the document. By using sentence centrality to improve sentence representation, relevance of sentences and documents was implicitly been increased. Notably, swapping out the sentence position information combined with sentence centrality reduced sentence-leading bias without degrading model performance Experiments on the CNN/Daily Mail dataset revealed that EDS models with sentence centrality outperformed baseline models significantly. (Gong et al., 2022).

**2.3.4.** **ENHANCING BIOMEDICAL TEXT SUMMARIZATION USING SEMANTIC RELATION EXTRACTION**

Automatic text summarization for a biomedical concept can assist researchers in efficiently extracting the key points of a specific topic from a large amount of biomedical literature. Based on semantic relation extraction, this paper presents a method for generating text summaries for a given biomedical concept, such as H1N1 disease, from multiple documents. The strategy is divided into three stages: 1) Using the semantic knowledge representation tool SemRep, extract semantic relations in each sentence. 2) Creation of a relation-level retrieval method that selects the most relevant relations to each query concept and visualizes them graphically. 3) Extraction of informative sentences for relations in the relevant set-in order to interpret them from the document collection in order to generate text summaries using an information retrieval technique. The testing findings on summarization for a set of diseases demonstrate that addition of semantic knowledge improves performance, and our results outperform the MEAD system, a well-known text summarization tool. (Shang et al., 2011).

**2.3.5. DOCUMENT SUMMARIZATION BASED ON DATA RECONSTRUCTION**

Many real-world applications, such as snippet generation for search results and news headline generation, benefit greatly from document summarization. Document summarization has traditionally been accomplished by extracting sentences that cover the main topics of a document with the least amount of redundancy. In this paper, data reconstruction is taken from a different angle and a novel framework called Document Summarization Based on Data Reconstruction (DSDR) is being propose. In particular, this method generates a summary of the sentences that can best reconstruct the original document. It introduces two objective functions to model the relationship between sentences: (1) linear reconstruction, which uses linear combinations of selected sentences to approximate the document; (2) nonnegative linear reconstruction, which uses only additive, not subtractive, linear combinations. The reconstruction error in this framework becomes a natural criterion for measuring the summary quality. An effective solution was developed to solve each objective function. (He et al., n.d.) .

**2.3.6.** **EXTRACTION-BASED SINGLE-DOCUMENT SUMMARIZATION USING RANDOM INDEXING**

This paper presents a text document summarization technique that uses semantic similarity between sentences to remove redundancy from the text. Semantic similarity scores are calculated by randomly indexing sentences on a semantic space. In comparison to other semantic space algorithms, random indexing provides a computationally efficient method of implicit dimensionality reduction. It entails simple vector computations like addition. As a result, it provides an efficient method for computing similarities between words, sentences, and

documents. The semantic similarity scores of sentences were computed using random indexing, and an extract of the given text was produced using graph-based ranking algorithms. (Chatterjee & Mohan, 2007).

**2.3.7.** **ARTIFICIAL INTELLIGENCE FOR AUTOMATIC TEXT SUMMARIZATION**

With the advancement of technology, automatic text summarization has played a critical role in assisting people in obtaining key information from increasing massive data. Previously, few literatures have been related to using artificial intelligence to solve the problem of generating titles (short summaries) (AI). The goal of this research is to propose an AI approach for automatic text summarization. We created an AI text summarization system architecture using three models: a statistical model, a machine learning model, and a deep learning model, and we evaluated their performance. Essay titles and abstracts are used to train an artificial intelligence deep learning model to generate candidate titles, which are then evaluated by ROUGE for performance. (Day & Chen, 2018).

**2.3.8.** **ABSTRACTIVE DOCUMENT SUMMARIZATION WITHOUT PARALLEL DATA**

Typically, abstractive summarization is based on large collections of paired articles and summaries. However, parallel data is often scarce and expensive to obtain. A n abstractive summarization system that only uses large collections of example summaries and articles that do not match was developed. The method includes an unsupervised sentence extractor that selects salient sentences to include in the final summary, as well as a sentence abstractor trained on

pseudo-parallel and synthetic data that paraphrases each extracted sentence. (Nikolov & Hahnloser, 2019).

**2.3.9.** **A HYBRID MACHINE LEARNING MODEL FOR MULTI-DOCUMENT SUMMARIZATION**

This paper proposes a method for improving content selection in multi-document automatic text summarization using statistical tools. The method employs a trainable summarizer, which considers several factors, including word similarity among sentences, word similarity among paragraphs, text format, cue-words, a score related to the frequency of terms in the entire document, the title, sentence location, and the occurrence of non-essential information. Each of these sentence features' impact on the summarization task is investigated. These characteristics are then combined to build text summarizer models based on a maximum entropy model, a naive-Bayes classifier, and a support vector machine. The three models are combined to create a hybrid model that ranks the results in order of importance. (Fattah, 2014).

**2.3.10. USER-ORIENTED DOCUMENT SUMMARIZATION THROUGH VISION-BASED EYE-TRACKING**

A personalized document summarization algorithm whose   key idea is to use individual users' attention (reading) time spent on single words in a document as the essential clue. The prediction of user attention over each word in a document is based on the user's previous reads, which are acquired using a vision-based commodity eye-tracking mechanism. Through word semantics analysis, the algorithm can predict the user's attention over every word in the document once the user's attentions over a small collection of words are known. The document is then summarized

by the algorithm based on user attention to each individual word in the document. We created a document summarization prototype system using our algorithm**.** (Xu et al., 2009).

**2.3.8. SUMMARY OF REVIEWED LITERATURES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S/N** | **AUTHOR & YEAR** | **TITLE** | **METHODOLOGY** | **FEATURES** | **STRENGTHS** | **WEAKNESSES** |
| **1** | (Anjali et al., 2019) | A GRAPH BASED APPROACH FOR KEYWORD EXTRACTION FROM DOCUMENTS | Rapid Automatic Keyword Extraction (RAKE) algorithm, Keyword Extraction using Collective Node Weight (KECNW) | Use of keywords for summarizing. |  |  |
| **2** | (Ashari & Riasetiawan, 2017)**.** | DOCUMENT SUMMARIZATION USING TEXTRANK AND SEMANTIC NETWORK |  |  |  |  |
| **3** | (Gong et al., 2022) | IMPROVING EXTRACTIVE DOCUMENT SUMMARIZATION WITH SENTENCE CENTRALITY |  |  |  |  |
| **4** | (Shang et al., 2011) | ENHANCING BIOMEDICAL TEXT SUMMARIZATION USING SEMANTIC RELATION EXTRACTION |  |  |  |  |
| **5** | (He et al., n.d.) | DOCUMENT SUMMARIZATION BASED ON DATA RECONSTRUCTION |  |  |  |  |
| **6** | (Chatterjee & Mohan, 2007) | EXTRACTION-BASED SINGLE-DOCUMENT SUMMARIZATION USING RANDOM INDEXING |  |  |  |  |
| **7** | (Day & Chen, 2018) | ARTIFICIAL INTELLIGENCE FOR AUTOMATIC TEXT SUMMARIZATION |  |  |  |  |
| **8** | **.** (Nikolov & Hahnloser, 2019) | ABSTRACTIVE DOCUMENT SUMMARIZATION WITHOUT PARALLEL DATA |  |  |  |  |
| **9** | (Fattah, 2014)**.** | A HYBRID MACHINE LEARNING MODEL FOR MULTI-DOCUMENT SUMMARIZATION |  |  |  |  |
| **10** | (Xu et al., 2009)**.** | USER-ORIENTED DOCUMENT SUMMARIZATION THROUGH VISION-BASED EYE-TRACKING |  |  |  |  |

**CHAPTER THREE: SYSTEM ANALYSIS AND DESIGN**

**3.1. INTRODUCTION**

This chapter includes a thorough discussion of the project's system analysis and design. This chapter will detail the system that will be developed, as well as the process model that will be utilized in the design and implementation of the web-based document summarizing and topic prediction system, the numerous design and development tools that will be used, as well as the software requirements are described. Diagrams such as use case and activity diagrams related to the system's functions are also included.

**3.2. DESCRIPTION OF THE SYSTEM**

The system will provide an easy to use, interactive web-based document summarizing and topic prediction system. The system will also provide a few additional features such as language detection, summary download and text-to-speech in order to help blind users or speed the summary consumption process for users who can hear. The users will be able to type text into the text area, upload text (.txt) or word (.docx) documents only for content summarizing, the system has no limit to how much a user can summarize for the supported languages.

**3.3 REQUIREMENT SPECIFICATIONS**

Requirements are a list of the functionalities and/or services provided by a system and its operational constraints that reflect the needs of customers for a system that provides solution to an existing problem. Requirements are divided into types; **user and system requirements,** and categories; **functional and non-functional requirements.** The following are a list of all functionalities of the system grouped appropriately.

**3.3.1 USER REQUIREMENTS AND SYSTEM REQUIREMENTS.**

**User requirements**

A user should be able to do the following:

* Type in text to be summarized.
* Upload word (.docx) or text (.txt) file for summarizing.
* Specify summary length.
* See the length of the original text **S**
* See the length of the summarized text. **S**
* Hear the summary if they choose to. **S**
* View the topic of the provided body of text.
* Summarize text not only in English language. **S**
* Clear original text even after summary has been produced.
* Download the summarized text as a word document (.docx).

**System Requirements**

The system requires steady internet connection.

* A speaker for speech-to-text output.
* A JAVA Script supported browser.
* OS
* Hardware

**3.3.2 FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS**

**Functional Requirements**

* The system should be accessible over a network
* The system should be…………….

**Non-Functional Requirements**

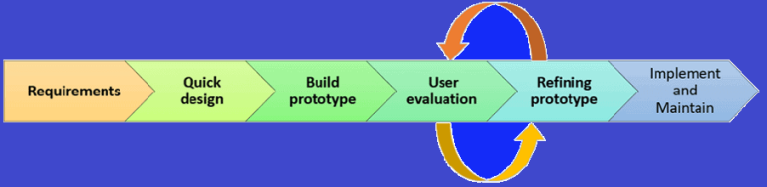
* The system must be fast.
* The system must be responsive
* The system should be scalable

**3.4 DESIGN MODEL**

A software design/development model is an outline of all processes involved in the designing, developing, deploying and eventually maintenance or any software product. Several design models such as the waterfall model, spiral model, v-model, iterative model, prototyping model, etcetera can be adopted for the development of any software project

however, for the purpose of developing this Web-Based Document Summarizing and Topic Prediction System Using Natural Language Processing, the prototyping model will be adopted. The model is a systems development method in which a prototype is built, tested, and then reconstructed until an acceptable outcome is reached from which the entire system

or product can be developed (Sarah Lewis, 2019) The prototyping model is adopted in this project due to the following beneficial reasons:

* Missing functionalities and errors are detected early: This will assist the development team to quickly identify and fix any issues.
* Fast user feedback: The feedback of users will help the development team to increase the software quality.
* Users are well involved in the development: The involvement of users in the development process will increase user satisfaction level as the system will fully represent their requirements.
* Provides better understanding of the system: Due to the fact that users are heavily involved in the building process, all the tests carried out by users will give them a better understanding of the system.

*Figure 3.1: Prototyping Process Model (Source: worldofitech, 2021*

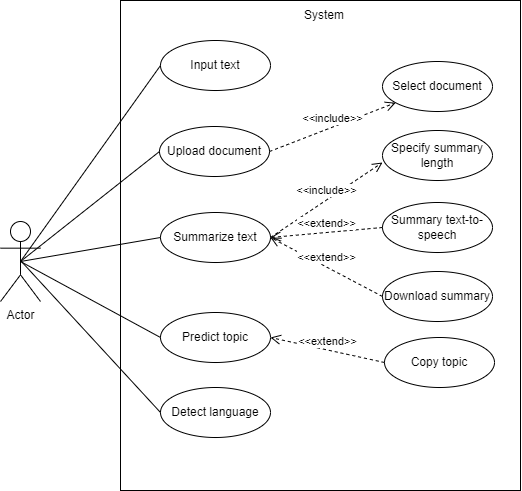
**3.5 DESIGN AND DEVELOPMENT TOOLS**

The Document Summarizing and Topic Prediction System Using Natural Language Processing is to be developed as a web-app. In order to fascilitate proper design and development of the system, the following development tools will be used:

* Microsoft Visual Studio Code: One of the most used IDEs (Integrated Development Environment) is the adopted development environment for the purpose of building this project as it supports all the programming languages to be used such as Python, HTML. It also provides a developer friendly syntax correction and easy code debugging feature which will aid in debugging quickly and finally, it is integrated with GIT, a version control system.
* Python: One of the top-rated high-level languages as of November, 2022. Python is being adopted due to the emphasis it lays on code readability, significant indentation, support of multiple code paradigms and above all, its versatility. Python will be used to train 2 NLP models and as the python Flask framework will be used for the server-side of the web-app due to its lightweight nature, low maintenance cost, flexibility and scalability.
* HTML: Short for Hypertext Markup Language is the standard language for the creation of web pages. It describes the structure of a web page by defining how elements appear on a web page. HTML will be used for the implementation of the front-end designs of the web pages.
* CSS: Cascading Style Sheets is a language for styling of HTML documents. CSS describes how the elements should be displayed as opposed to HTML which describes structure alone, CSS is used to style web page element colors, sizes, style, background colors, etcetera.
* JavaScript: The programming language most associated with web development. JavaScript is used to make changes and updates to HTML and CSS code dynamically at run time when prompted. It will be used to give the system some advanced adaptation features.

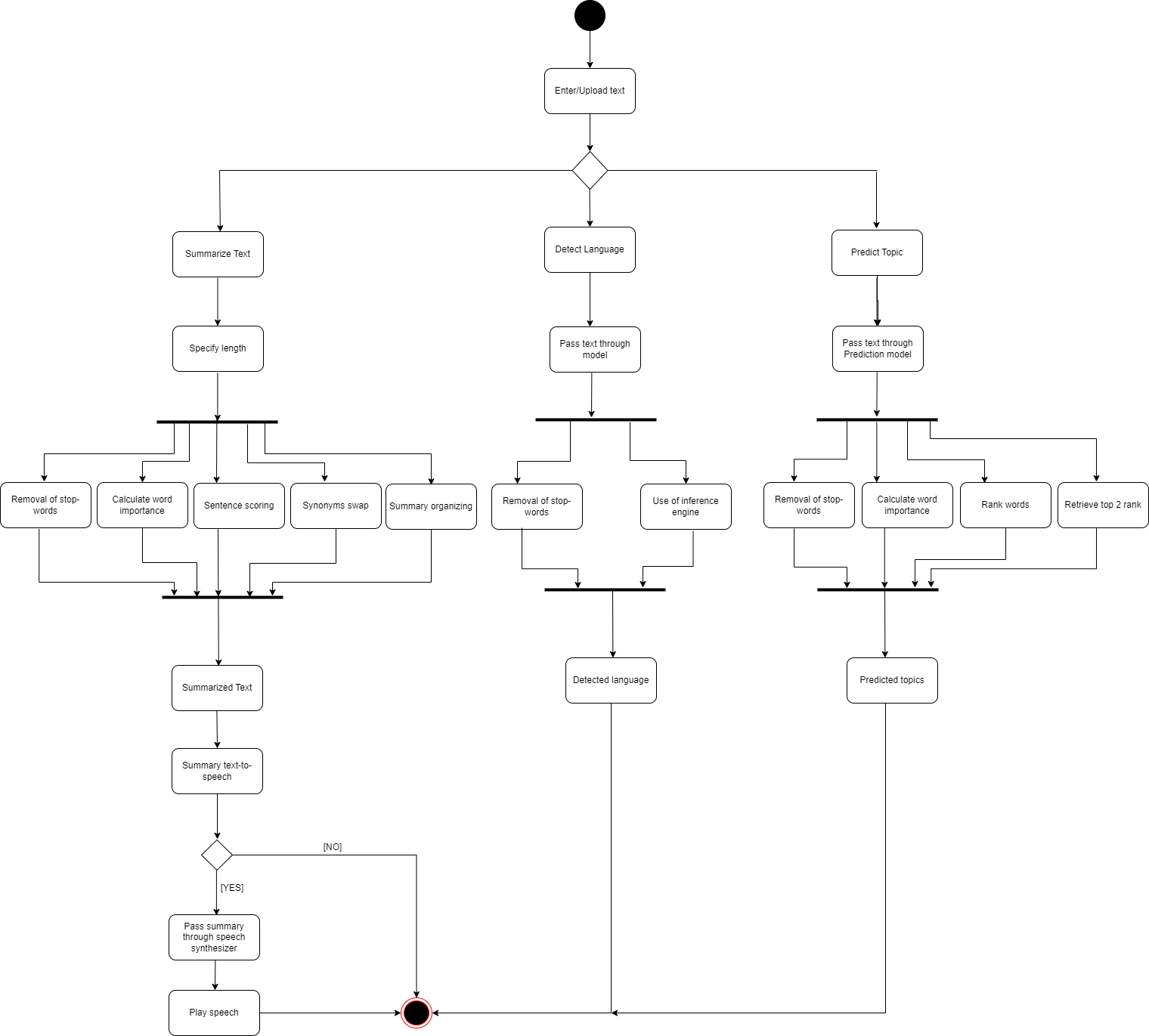
**3.6 DIAGRAMS**

**3.6.1 USE CASE DIAGRAM**

****

*Fig 3.2 Use case diagram*

**3.6.2 ACTIVITY DIAGRAM**

****

*Figure 3.3 Activity diagram.*

**REFERENCES**

Lewis. (2014, December 4). A Brief History of Artificial Intelligence | Live Science. livescience.com.

Retrieved September 30, 2022, from <https://www.livescience.com/49007-history-of-artificial-intelligence.html>

Education, I. C. (2020, July 2). What is Natural Language Processing? What Is Natural Language

Processing? | IBM. Retrieved September 30, 2022, from <https://www.ibm.com/cloud/learn/natural-language-processing#:~:text=Natural%20language%20processing%20(NLP)%20refers,same%20way%20human%20beings%20can>.

Menzli, A. (2021, July 28). 10 NLP Projects to Boost Your Resume - neptune.ai. neptune.ai. Retrieved

September 30, 2022, from <https://neptune.ai/blog/10-nlp-projects/amp>

(1993, January 1). HTML - Wikipedia. HTML - Wikipedia. Retrieved October 1, 2022, from

<https://en.wikipedia.org/wiki/HTML>

A., J. (2021, August 3). What Is JavaScript? A Basic Introduction to JS for Beginners. Hostinger

Tutorials. Retrieved October 2, 2022, from <https://www.hostinger.com/tutorials/what-is-javascript>

DEERY. (2022, August 1). The Flask Web Framework: A Beginner’s Guide. CareerFoundry. Retrieved

October 2, 2022, from <https://careerfoundry.com/en/blog/web-development/what-is-flask/>

(n.d.). *Summary Definition & Meaning | Britannica Dictionary*. Summary Definition & Meaning | Britannica Dictionary. Retrieved November 20, 2022, from

<https://www.britannica.com/dictionary/summary>

Learner’s. (n.d.). *summary1 noun - Definition, pictures, pronunciation and usage notes | Oxford Learner’s Dictionary of Academic English at OxfordLearnersDictionaries.com*. Summary1 Noun - Definition, Pictures, Pronunciation and Usage Notes | Oxford Learner’s Dictionary of Academic English at OxfordLearnersDictionaries.com. Retrieved November 20, 2022, from

<https://www.oxfordlearnersdictionaries.com/definition/academic/summary1>

B, P. K. (2021, July 17). *Brief history of Text Summarization*. Medium. Retrieved November 20, 2022, from

<https://medium.com/@prasasthy.sanal/brief-history-of-text-summarization-9d1b3787a707>

Salton, Singhal, Mitra, & Buckley. (1998, June 11). *Automatic text structuring and summarization*. Automatic Text Structuring and Summarization - ScienceDirect. Retrieved November 20, 2022, from

<https://www.sciencedirect.com/science/article/pii/S0306457396000623>

Abdi, A., Idris, N., Alguliyev, R. M., & Aliguliyev, R. M. (2016a). An Automated Summarization Assessment Algorithm for Identifying Summarizing Strategies. *PLOS ONE*, *11*(1), e0145809. https://doi.org/10.1371/journal.pone.0145809

Abdi, A., Idris, N., Alguliyev, R. M., & Aliguliyev, R. M. (2016b). An Automated Summarization Assessment Algorithm for Identifying Summarizing Strategies. *PLOS ONE*, *11*(1), e0145809. https://doi.org/10.1371/journal.pone.0145809

Alonso i Alemany, L., & Fuentes Fort, M. (2003). Integrating cohesion and coherence for automatic summarization. *Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics  - EACL ’03*, 1. https://doi.org/10.3115/1067737.1067739

ANDO, R., BOGURAEV, B., BYRD, R., & NEFF, M. (2005). Visualization-enabled multi-document summarization by Iterative Residual Rescaling. *Natural Language Engineering*, *11*(1), 67–86. https://doi.org/10.1017/S1351324904003389

Anjali, S., Meera, N. M., & Thushara, M. G. (2019). A Graph based Approach for Keyword Extraction from Documents. *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP)*, 1–4. https://doi.org/10.1109/ICACCP.2019.8882946

Ashari, A., & Riasetiawan, M. (2017). Document Summarization using TextRank and Semantic Network. *International Journal of Intelligent Systems and Applications*, *9*(11), 26–33. https://doi.org/10.5815/ijisa.2017.11.04

Barzilay, R., & Elhadad, M. (1999). Using Lexical Chains for Text Summarization. *The MIT Press, Cambridge, 111-121*.

B.J. Copeland. (2022). Artificial intelligence. *Encyclopedia Britannica*.

Chatterjee, N., & Mohan, S. (2007). Extraction-Based Single-Document Summarization Using Random Indexing. *19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007)*, 448–455. https://doi.org/10.1109/ICTAI.2007.28

Day, M.-Y., & Chen, C.-Y. (2018). Artificial Intelligence for Automatic Text Summarization. *2018 IEEE International Conference on Information Reuse and Integration (IRI)*, 478–484. https://doi.org/10.1109/IRI.2018.00076

Edmundson, H. P. (1969). New Methods in Automatic Extracting. *Journal of the ACM*, *16*(2), 264–285. https://doi.org/10.1145/321510.321519

Fattah, M. A. (2014). A hybrid machine learning model for multi-document summarization. *Applied Intelligence*, *40*(4), 592–600. https://doi.org/10.1007/s10489-013-0490-0

Gong, S., Zhu, Z., Qi, J., Tong, C., Lu, Q., & Wu, W. (2022). Improving extractive document summarization with sentence centrality. *PLOS ONE*, *17*(7), e0268278. https://doi.org/10.1371/journal.pone.0268278

He, Z., Chen, C., Bu, J., Wang, C., Zhang, L., Cai, D., & He, X. (n.d.). *Document Summarization Based on Data Reconstruction*. www.aaai.org

IBM Cloud Education. (2020). *Natural Language Processing (NLP)*.

Knight, K., & Marcu, D. (2002). Summarization beyond sentence extraction: A probabilistic approach to sentence compression. *Artificial Intelligence*, *139*(1), 91–107. https://doi.org/10.1016/S0004-3702(02)00222-9

Kobayashi, H., Noguchi, M., & Yatsuka, T. (2015). *Summarization based on embedding distributions*.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval  - SIGIR ’95*, 68–73. https://doi.org/10.1145/215206.215333

Lloret, E., & Palomar, M. (2012). Text summarisation in progress: a literature review. *Artificial Intelligence Review*, *37*(1), 1–41. https://doi.org/10.1007/s10462-011-9216-z

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, *2*(2), 159–165. https://doi.org/10.1147/rd.22.0159

Luo, W., Liu, F., Liu, Z., & Litman, D. (2018). *A Novel ILP Framework for Summarizing Content with High Lexical Variety*.

MANN, W. C., & THOMPSON, S. A. (1988). Rhetorical Structure Theory: Toward a functional theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse*, *8*(3). https://doi.org/10.1515/text.1.1988.8.3.243

NASERASADI, A., KHOSRAVI, H., & SADEGHI, F. (2019). Extractive multi-document summarization based on textual entailment and sentence compression via knapsack problem. *Natural Language Engineering*, *25*(1), 121–146. https://doi.org/10.1017/S1351324918000414

Neto, J. L., Freitas, A. A., & Kaestner, C. A. A. (2002). *Automatic Text Summarization Using a Machine Learning Approach* (pp. 205–215). https://doi.org/10.1007/3-540-36127-8\_20

Nikolov, N. I., & Hahnloser, R. H. R. (2019). *Abstractive Document Summarization without Parallel Data*.

Rada Mihalcea, & Paul Tarau. (2004). TextRank: Bringing Order into Text. *Association for Computational Linguistics*, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

REITER, E., & DALE, R. (1997). Building applied natural language generation systems. *Natural Language Engineering*, *3*(1), S1351324997001502. https://doi.org/10.1017/S1351324997001502

Sarah Lewis. (2019). Prototyping Model. *SearchCIO*.

Shang, Y., Li, Y., Lin, H., & Yang, Z. (2011). Enhancing Biomedical Text Summarization Using Semantic Relation Extraction. *PLoS ONE*, *6*(8), e23862. https://doi.org/10.1371/journal.pone.0023862

Simone Teufel. (1997). *Sentence extraction as a classification task*.

SRIHARI, R. K., LI, W., CORNELL, T., & NIU, C. (2008). InfoXtract: A customizable intermediate level information extraction engine. *Natural Language Engineering*, *14*(01). https://doi.org/10.1017/S1351324906004116

Stuart Jonathan Russell, & Peter Norvig. (2009). *Artificial Intelligence: A Modern Approach*.

Xu, S., Jiang, H., & Lau, F. C. M. (2009). User-oriented document summarization through vision-based eye-tracking. *Proceedings of the 14th International Conference on Intelligent User Interfaces*, 7–16. https://doi.org/10.1145/1502650.1502656

Yogatama, D., Liu, F., & Smith, N. A. (2015). Extractive Summarization by Maximizing Semantic Volume. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1961–1966. https://doi.org/10.18653/v1/D15-1228