BIOMENTOR - PERSONALIZED E-LEARNING PLATFORM FOR ENGLISH MEDIUM A/L BIOLOGY SUBJECT STUDENTS IN SRILANKA

LLM BASED ABSTRACTIVE TEXT SUMMARIZATION TOOL WITH VOICE OUTPUT IMPLEMENTED IN DIFFERENT SOFTWARE ARCHITECTURES

24-25J-257

Project Proposal Report

Dharane.S - IT21068478

B.Sc. (Hons) in Information Technology Specializing in Software Engineering

Department of Computer Science & Software Engineering

Sri Lanka Institute of Information Technology Sri Lanka

BIOMENTOR - PERSONALIZED E-LEARNING PLATFORM FOR ENGLISH MEDIUM A/L BIOLOGY SUBJECT STUDENTS IN SRILANKA

LLM BASED ABSTRACTIVE TEXT SUMMARIZATION TOOL WITH VOICE OUTPUT IMPLEMENTED IN DIFFERENT SOFTWARE ARCHITECTURES

24-25J-257

Project Proposal Report

B.Sc. (Hons) in Information Technology Specializing in Software Engineering

Department of Computer Science & Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

Declaration

I declare that this is my own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously publish or written by another person expect where the acknowledgement is made in the text.

Name	Student ID	Signature
Dharane.S	IT21068478	

The above candidate is carrying out research for the undergraduate Dissertation under my supervision.

.....

Signature of the supervisor:

(Dr. Sanvitha Kasthuriarachchi) Date.....22/8/24'

.....

Signature of the Co-

supervisor:

(Ms. Karthiga Rajendran) Date.....22/08/2024

Abstract

Advanced text summarization tools play a crucial role in enhancing educational technology, particularly for complex subjects like biology. This research focuses on developing a tool integrated into an e-learning platform that offers two types of summaries: one generated directly from uploaded documents and another based on specific topics extracted from government-approved A/L Biology resources. The tool will allow users to customize the word count for both types of summaries, catering to diverse learning needs. While e-learning platforms provide flexible access to educational materials, they often encounter limitations in handling various document formats and delivering contextually relevant summaries. Additionally, existing tools may lack comprehensive voice output features, which are essential for auditory learners and those needing on-the-go review. This research addresses these challenges by developing a summarization tool that extracts and processes text from both uploaded documents and specified topics, ensuring accuracy and relevance by referencing authoritative educational resources. The tool will be implemented using different software architectures to evaluate and compare their performance. This comparative analysis will provide insights into the most effective architectural approaches for deploying advanced text summarization in e-learning environments.

Keywords: E-Learning, Text Summarization, Biology Education, Document Analysis, Voice Output, Software Architecture, Customizable Summaries

A/L	Advanced-Level
DB	Database
API	Application Framework Interface
VS CODE	Visual Studio Code
NLP	Natural Language Processing
OCR	Optical Character Recognition
BART	Bidirectional Auto-Regressive Transformers
T5	Text-to-Text Transfer Transformer
BERT	Bidirectional Encoder Representations from Transformers
TTS	Text-to-Speech
PDF	Portable Document Format

Table 1 : List of Abbreviation

Table of Contents

List of Figures

List of Tables

1.0 Introduction

1.1 Background and Literature

Text summarization is a critical component of information management, particularly in educational technology. It involves condensing large volumes of text into shorter, more digestible forms while retaining essential information and meaning. This process is especially valuable in e-learning platforms, where users need to access and understand extensive information quickly. Effective summarization tools help make complex subjects more accessible and manageable, aiding in the comprehension of detailed content, such as that found in biology.

Summarization techniques generally fall into two categories: extractive and abstractive summarization.

- Extractive Summarization involves selecting and compiling key sentences or phrases directly from the source text. This method identifies and pulls out the most relevant sections, which are then combined to form a summary. While effective in retaining exact wording from the original text, extractive summarization may lack coherence and the ability to rephrase content in a more concise manner.
- Abstractive Summarization, on the other hand, generates new sentences that capture the essence of the original text. It uses advanced models to understand the content and then rephrases and condenses the information into a coherent summary. Recent advancements in abstractive summarization have utilized models like BART or T5 to produce summaries that are both contextually relevant and fluidly articulated [1]. Abstractive methods are particularly valuable for generating summaries that are more readable and contextually appropriate.

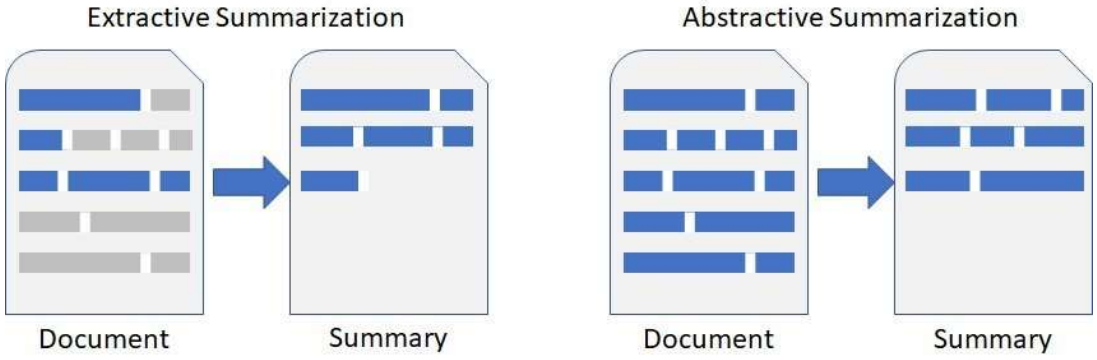

Figure 1: Summarization Techniques

Current Advancements and Challenges:

Existing research highlights the potential of these summarization techniques to enhance educational content delivery. For instance, NLP-enhanced summarization approaches have demonstrated the capability to condense information from extensive texts while preserving critical details [2]. However, the challenges remain in addressing the diverse needs of users, particularly in specialized fields like biology.

While abstractive summarization techniques like BART have shown promise in producing coherent summaries from long documents, the application to specialized educational content has room for improvement. Current summarization tools often focus on general content and may not adequately address the specific requirements of subjects like biology. Models like BERT (Bidirectional Encoder Representations from Transformers) have been explored for topic-level summaries, but there is a need for more tailored approaches that consider the unique complexities of academic subjects [3].

In the realm of e-learning, integrating voice output features into summarization tools has been limited. Although advancements in text-to-speech technologies have been made, providing high-quality and customizable voice outputs for educational content remains a challenge [4]. Such enhancements can significantly benefit auditory learners and those who need to review content on the go.

Additionally, the implementation of summarization tools across various software architectures has not been extensively studied. Evaluating different architectures for deploying summarization technologies can offer insights into optimizing performance and scalability [5].

Addressing these gaps, this research proposes the development of a summarization tool that not only generates summaries from uploaded documents but also extracts and summarizes information from government-approved resources based on specific topics. This approach aims to improve accuracy and relevance in summarization while integrating voice output features to support diverse learning styles.

1.2 Research Gap

Advanced text summarization tools have significant potential to enhance educational technology, particularly within e-learning platforms. However, several research gaps must be addressed to fully exploit their capabilities, especially for complex subjects like biology.

A primary research gap is the need for more effective audible summaries. Although text-to-speech technologies have advanced, many existing summarization tools do not incorporate high-quality, customizable voice output features. There is a need for research into developing voice output capabilities that are tailored to various learning styles, particularly for auditory learners who benefit from audible summaries in educational contexts.

Another challenge is the extraction of data from approved educational resources. Current tools often rely on general data sources, which may not always align with specific educational standards. Research is needed to develop methods for accurately extracting and summarizing content from government-approved resources, ensuring that summaries adhere to educational requirements and provide reliable information.

Customizable word count is another area requiring attention. Existing summarization tools may not offer sufficient flexibility for users to specify the length of summaries according to their needs. Research should focus on implementing features that allow users to adjust word counts dynamically while maintaining the quality and relevance of the summarized content.

Additionally, the handling of document uploads remains a challenge. While tools can process various document formats, there is a need for improved accuracy in text extraction, especially from diverse and complex document types such as scanned PDFs or Word files. Addressing this involves enhancing OCR (Optical Character Recognition) technologies and summarization algorithms to handle different document formats effectively.

Finally, there is a lack of comprehensive studies comparing different software architectures for implementing summarization tools. The performance, scalability, and effectiveness of these tools can vary based on the chosen architecture. Research is needed to evaluate and compare different architectural approaches to identify the most effective solutions for deploying advanced summarization technologies.

By addressing these research gaps, this project aims to develop a robust summarization tool that integrates advanced audible summary features, accurate data extraction from approved resources, customizable word counts, and efficient document upload processing. This will enhance the tool’s effectiveness in educational contexts and support diverse learning needs.

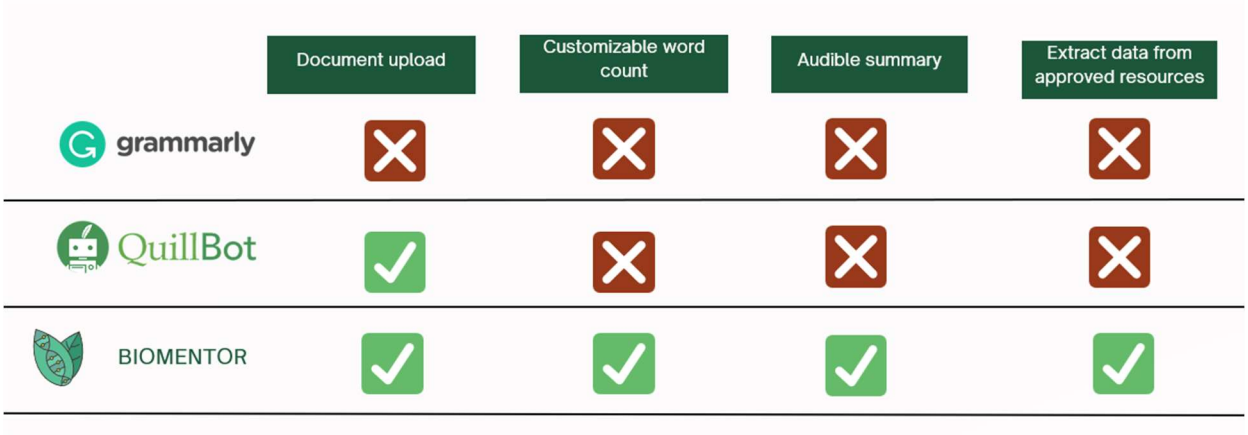

Figure 2: Competitive Analysis

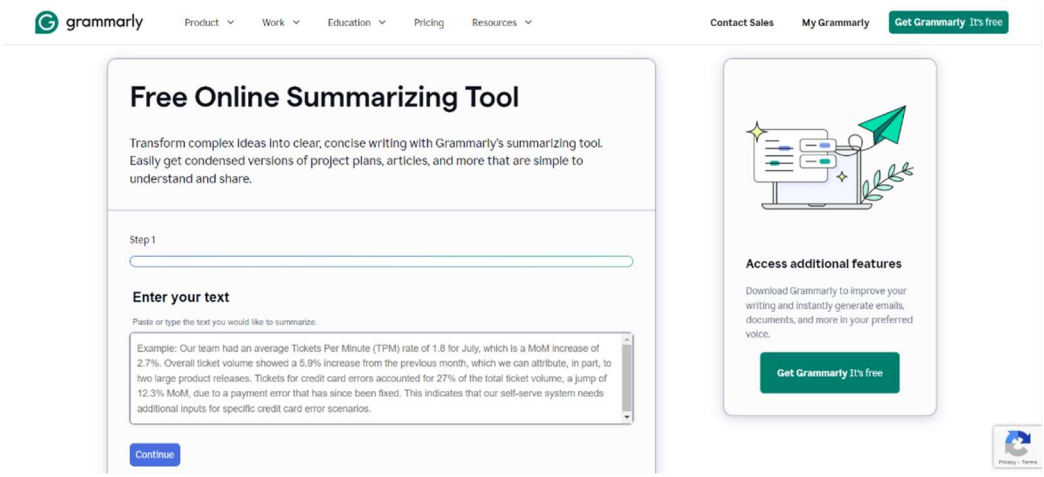

Figure 3: Grammarly – Summarization Tool

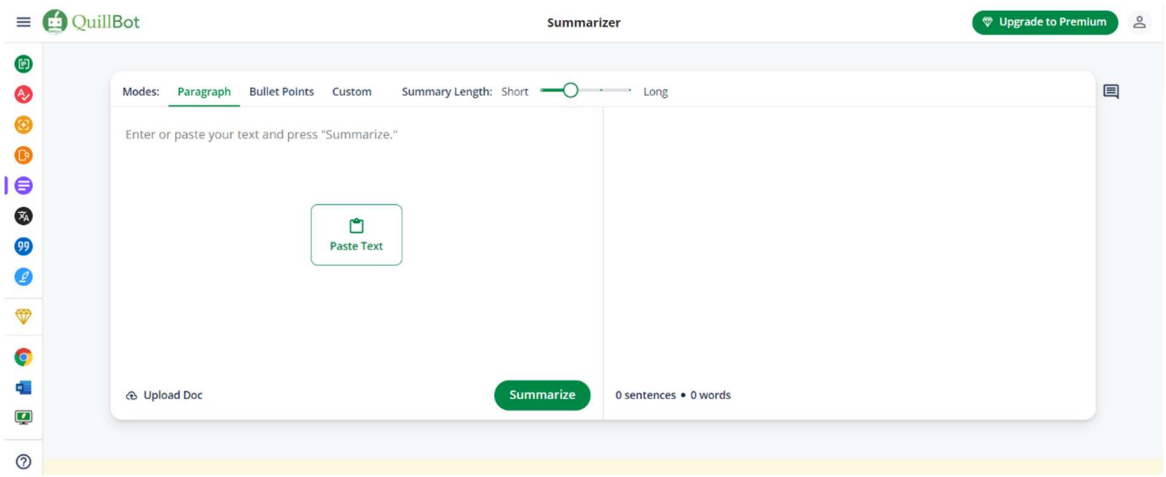

Figure 4: QuillBot – Summarization Tool

1.3 Research Problem

With the increasing adoption of e-learning platforms across educational institutions and organizations, there is a growing need for effective tools that enhance learning and address the challenges faced by students. While e-learning provides flexibility and access to a wealth of resources, it also presents several issues, including the need for effective summarization of complex educational content and the integration of supportive features to aid diverse learning styles.

One significant challenge is the accurate summarization of complex subjects such as biology from diverse document formats. Traditional summarization tools often fail to provide concise, relevant, and contextually appropriate summaries, which are crucial for students trying to grasp intricate concepts. Existing tools need to be enhanced to deliver high-quality summaries from both uploaded documents and government-approved educational resources, with the added capability of customizable word counts to meet varied user needs [1][2].

Moreover, there is a lack of integration between text summarization tools and voice output features. While text-to-speech technology has advanced, many tools do not effectively support auditory learners by providing customizable and high-quality audible summaries. This gap highlights the need for research into incorporating robust voice output features that improve accessibility and engagement [3][4].

Additionally, the process of extracting relevant information from government-approved educational resources is not well-researched. Ensuring that summaries generated from these resources adhere to educational standards and accurately reflect the content is a critical issue. Research is needed to develop methods for effective data extraction and summarization that align with curriculum requirements and provide reliable educational content [5].

The problem is further compounded by the need for comparative studies on different software architectures used to implement summarization tools. There is limited research on how various architectural approaches impact the performance, scalability, and effectiveness of summarization solutions. Addressing this gap involves evaluating different architectures to identify the most effective solutions for deploying advanced summarization technologies.

By addressing these research problems, this project aims to develop a comprehensive summarization tool that integrates high-quality audible summaries, accurate data extraction from approved resources, customizable word counts, and efficient document handling. This will enhance the effectiveness of e-learning platforms and better support students, educators, and professionals in managing and understanding educational content.

2.0 Objectives

2.1 Main Objective

The objective of this project is to develop a tool that can extract key concepts from A/L biology resources and the uploaded documents and prepare summaries that can aid students who prepare for their A/L exams.

2.2 Specific Objectives

- Ensure Accuracy: Government-approved resources for accurate content.
- Enhance Accessibility: Voice output is to be added to the summaries to be heard in audio.
- User Customization: Word count for the summary to be custom-made at the end-user's wish.
- Real-Time Solution: Immediate summarization for e-Learning.
- Sri Lankan Context: Features should be changed according to the Sri Lankan educational system.
- Better Learning: Make concise summaries to explain complex concepts.

3.0 Methodology

3.1 Requirement Gathering

Requirement gathering involves a comprehensive analysis of existing research, evaluation of current systems, and review of relevant online resources. Additionally, practical scenarios are analyzed to identify research problems and gather requirements. This process ensures that the developed tool addresses the needs of users effectively and aligns with current technological and educational trends.

3.2 Past Research Analysis

In past research, advanced text summarization technologies have been extensively studied, with a focus on models such as BART (Bidirectional and Auto-Regressive Transformers),T5 (Text-to-Text Transfer Transformer) and NLP-enhanced summarization approaches. These studies have demonstrated significant advancements in generating coherent and contextually relevant summaries from long documents. However, there is a notable gap in applying these technologies specifically to A/L biology content and integrating voice output features with customizable word counts .

Moreover, past research has not extensively explored the implementation of summarization solutions across different architectural approaches. Key architectural models include:

- **Monolithic Architecture**: Traditional monolithic systems consolidate all components into a single application. While this approach simplifies development and deployment, it can lead to scalability issues and challenges in maintaining and updating the system as it grows. Research has highlighted the limitations of monolithic systems in handling complex applications and large-scale data.

- **Layered Architecture:** In this architectural model, the system is divided into separate layers, each tasked with different functions such as user interface, business logic, and data management. This structure promotes a clear separation of responsibilities, which simplifies the management and maintenance of the system. Studies show that layered architecture can improve scalability and adaptability, particularly in modular development contexts, compared to more monolithic designs.

- **Microservices Architecture:** This architectural style breaks down the system into smaller, autonomous services that interact through APIs. Each microservice is responsible for a distinct function, allowing for independent development, deployment, and scaling. While microservices can significantly enhance flexibility, scalability, and ease of maintenance, they also add complexity to the implementation and management processes. Current research into the use of microservices for summarization systems is limited, indicating a need for more investigation into how this approach affects system performance and integration.

The lack of focus in past research on implementing summarization solutions across these different architectures underscores the need for this study. By evaluating how various architectures impact the performance and scalability of summarization tools, this research aims to provide valuable insights and recommendations for optimizing the tool's effectiveness in generating summaries from A/L biology resources and integrating voice output features.

3.3 Development Methodology

Effective management of the project requires careful consideration of time, scope, and cost, with flexibility to accommodate changes. The Agile methodology has been selected to allow for iterative development and continuous improvement. This approach supports the development of the text summarization tool through short, focused sprints, ensuring that feedback from the mentor can be incorporated regularly. By presenting functional prototypes at each stage, the methodology enables ongoing refinement and adjustment, resulting in a final product that meets the highest standards and aligns with both the author's and mentor's expectations.

3.4 Project Management Methodology

For managing the development of the advanced text summarization tool, we will utilize Jira. This tool will aid in planning, tracking, and overseeing the project's progress. Jira will enable us to structure tasks, assign roles, and keep track of milestones, ensuring effective project management and timely delivery of the tool.

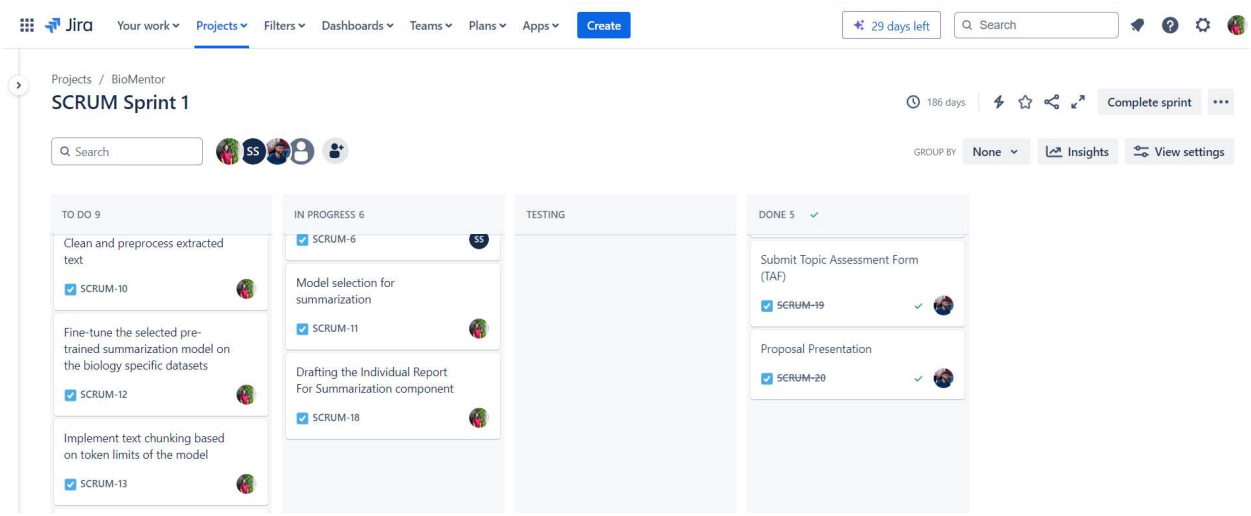

Figure 5: Jira – Project Management Tool

3.5 Feasibility Study

3.5.1 Technical Feasibility

3.5.1.1 Knowledge on Technologies

To develop the proposed text summarization tool, proficiency in the following technologies is required:

- Abstractive Summarization Models: Understanding and implementation of advanced models such as BART (Bidirectional and Auto-Regressive Transformers) or T5 (Text-to-Text Transfer Transformer) for generating coherent and contextually relevant summaries from long documents.
- Natural Language Processing (NLP): Knowledge of NLP techniques to enhance the summarization process and ensure accurate information condensation.
- Text-to-Speech (TTS) Technologies: Familiarity with TTS systems to integrate voice output features, allowing auditory learners to benefit from the summarized content.
- Machine Learning Frameworks: Experience with frameworks like TensorFlow or PyTorch for model training and fine-tuning.
- Data Preprocessing: Skills in data cleaning, transformation, and integration to prepare documents and text for effective summarization.

3.5.1.2 Knowledge on Tools

The development of the proposed tool requires a solid understanding of various development and project management tools. The team will use Jira for project management, which will help in tracking progress, managing tasks, and facilitating team collaboration.

3.5.1.3 Data Collection Knowledge

Effective data collection and preprocessing are crucial for the success of the summarization tool:

- Data Collection: Gathering high-quality A/L biology textbooks and government-approved resources for training and validation of the summarization model.
- Data Preprocessing: Includes data cleaning, transformation, reduction, and integration to prepare the text data for accurate summarization.

3.5.2 Schedule Feasibility

The project will be developed according to a defined timeline managed in Jira. The schedule will outline key milestones, development phases, and deadlines to ensure timely delivery of the tool.

3.5.3 Economic Feasibility

The development of the advanced text summarization tool aims to minimize costs while providing a cost-effective solution compared to alternative models. Although actual expenses may vary due to economic conditions, the preliminary cost estimation is detailed in Table 4 under the budget section. This approach ensures that the project remains economically viable while delivering valuable functionality. The draft estimation is in table 3 under budget section.

3.6 System Analysis

3.6.1 Software Solution Approach

The development of the text summarization tool involves several key stages, as outlined below.

1. Data Collection:

    - Gather Documents: Collect text documents, including A/L biology textbooks and government-approved resources. These documents will be stored in a vector database for efficient retrieval.

    - Topic-Based Extraction: Extract information based on specific topics from the approved resources.

    - Training Data: Use biology-specific datasets to train and validate the summarization model.

2. Data Preprocessing:

    - Text Extraction: For uploaded documents, extract textual content from any type of document. For image-based documents, employ Optical Character Recognition (OCR) to convert images into text.

    - Text Normalization: Clean and format the extracted text to prepare it for summarization. This includes removing irrelevant information and standardizing text format.

3. Summarization Process:

    - Abstractive Summarization: Apply advanced abstractive summarization techniques, such as BART (Bidirectional and Auto-Regressive Transformers), to generate concise and coherent summaries from the preprocessed text. This involves leveraging NLP models to understand and rephrase key concepts from the documents.

- Topic-Based Summarization: Extract relevant details from government-approved resources stored in the database and summarize them based on the given topic.

4. Voice Output Integration:

    - Text-to-Speech (TTS): Incorporate TTS technology to convert the generated summaries into audible formats. Ensure that the voice output is clear and customizable according to user preferences.

5. Customizable Word Count:

    - User Specification: Implement a feature allowing users to specify the desired length of the summaries. This involves adjusting the summarization process to meet different word count requirements.

6. System Architecture Implementation:

    - Monolithic Architecture: Implement a unified system where all components are integrated into a single application.

    - Layered Architecture: Develop the system in layers, including presentation, business logic, and data access layers, to improve modularity and maintainability.

    - Microservices Architecture: Design the system as a collection of loosely coupled services, each handling specific functionalities such as summarization, voice output, and data management. This approach enhances scalability and flexibility.

7. Evaluation and Testing:

    - Performance Metrics: Assess the performance of the summarization tool using metrics such as accuracy, relevance, and user satisfaction.

    - Testing: Conduct thorough testing to ensure the tool meets the requirements and performs effectively in various scenarios.

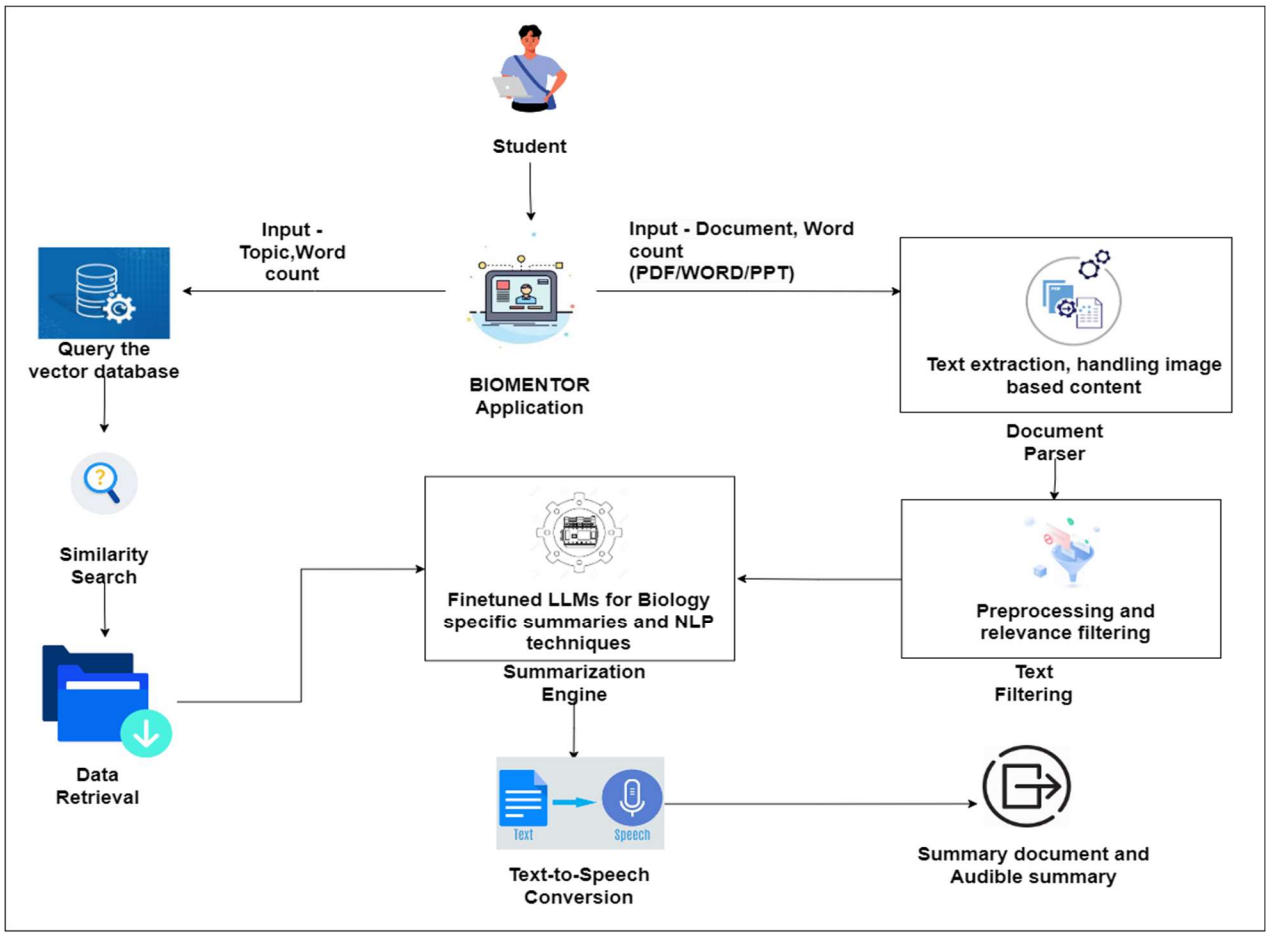

Figure 6: System Diagram

3.7 Tools and Technologies

Programming Languages – Python

Frameworks and Libraries:

- Transformers Library: For implementing abstractive summarization models like BART,BERT,T5.
- TensorFlow and PyTorch: For training and fine-tuning NLP models.
- NLTK (Natural Language Toolkit): For text preprocessing tasks such as tokenization, stemming, and part-of-speech tagging.
- Pyttsx3 or gtts : For integrating text-to-speech functionality.

Data Storage and Management:

- Vector Database (Faiss): For storing and retrieving A/L biology textbooks and government-approved resources efficiently.
- MongoDB

Text Extraction and OCR:

- OCR Libraries (e.g., Tesseract): For extracting text from image-based documents.
- Document Processing Libraries: For handling text extraction from various document formats.

Voice Output Integration:

- Text-to-Speech (TTS) Technologies: Such as Google Text-to-Speech for converting summaries into audible formats.

Software Architectures:

- Monolithic Architecture: For simpler implementations where all components are integrated into a single application.
- Layered Architecture: For organizing the system into presentation, business logic, and data access layers.
- Microservices Architecture: For designing the system as a collection of loosely coupled services, enhancing scalability and flexibility.

Development and Project Management Tools:

- Google Colab or VS Code: For development and testing of the summarization model.
- Jira: For project management, tracking progress, and managing tasks.

UI Design and Development:

- Figma: For designing user interfaces and user experience elements.
- React JS: For front-end development.
- Tailwind CSS: For designing and developing UI components.

Diagramming and Design Tools:

- Draw.io: For creating diagrams and flowcharts.

Version Control System:

- Git (GitHub or GitLab): For version control and collaborative development.

Collaboration Tools:

- Microsoft Teams or WhatsApp: For team communication and collaboration.

Testing Tools:

- Unit Test: For testing Python code.
- Postman: For API testing and validation.

Deployment Tools:

- Docker: For containerization of the application.
- Kubernetes: For orchestration and management of containerized applications.

Project Management:

- Jira: For managing project tasks, tracking progress, and ensuring effective project execution.

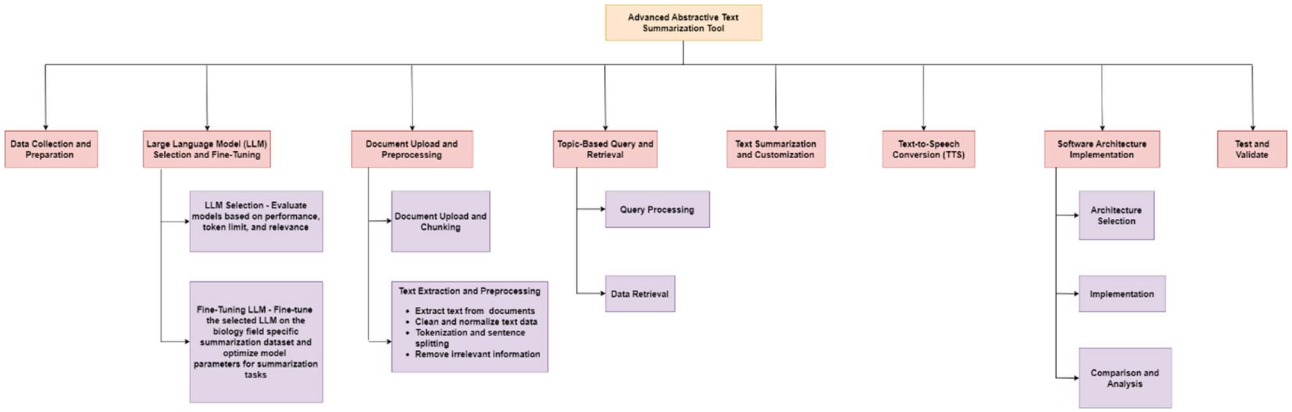

Figure 7: Work Breakdown Structure

3.8Project Requirements

3.8.1 Functional Requirements

The functional requirements of the proposed text summarization tool are as follows.

- Document Processing: Text documents of various types, such as PDFs and Word files, shall be accepted and processed. It should extract text from image-based documents using OCR.
- Data Collection: The tool should collect and make use of text data from government-approved A/L Biology resources for summarization.
- Text Extraction: Extract relevant text from the uploaded documents and government-approved resource documents.
- Summarization: Generate summaries from the extracted text, limited to a word count as defined by the user.
- Voice Output: The generated summaries are converted into voice output with the help of text-to-speech technology.

3.8.2 Non-Functional Requirements

The following non-functional requirements shall be prioritized in the development and implementation of the summarization tool:

- Accuracy
- Performance
- Availability
- Compliance
- Usability

3.9 Project Scope

3.9.1 In-Scope

- Development of a text summarization tool that extracts and summarizes key concepts from A/L biology textbooks and government-approved resources.
- Implementation of features to process various document types, including PDFs and Word documents, and handle image-based documents using OCR for text extraction.
- Creation of a user interface that allows users to specify the desired length of summaries and convert them into voice output using TTS technology.
- Utilization of advanced summarization models and techniques for both document-based and topic-based summaries.
- Evaluation of different software architectures (monolithic, layered, microservices) to determine the most effective approach for implementation.

3.9.2 Out of Scope

- The tool will not handle document types outside of academic textbooks and government-approved resources, such as personal or business documents.
- Summarization features are limited to text-based content; documents primarily containing images or non-textual data will not be processed for summarization.
- Real-time summarization or live document updates are not included; the tool focuses on pre-existing documents and batch processing.

3.10 Testing

Testing is a critical phase in ensuring the success of the proposed text summarization tool. Initially, the tool will undergo unit and system testing among team members to validate its functionality and identify any issues. This phase will involve evaluating the core features, including text extraction, summarization accuracy, and voice output.

In the subsequent phase, the tool will be tested with a selected group of A/L biology students in both local and deployed environments. This will encompass acceptance testing, including alpha and beta testing, to assess real-world performance and gather user feedback.

Testing will primarily be manual, supplemented by internal functions and assertions to verify specific aspects of the tool. This approach will help ensure that the tool meets functional requirements, performs reliably, and delivers accurate summaries and voice outputs.

3.11 Timeline

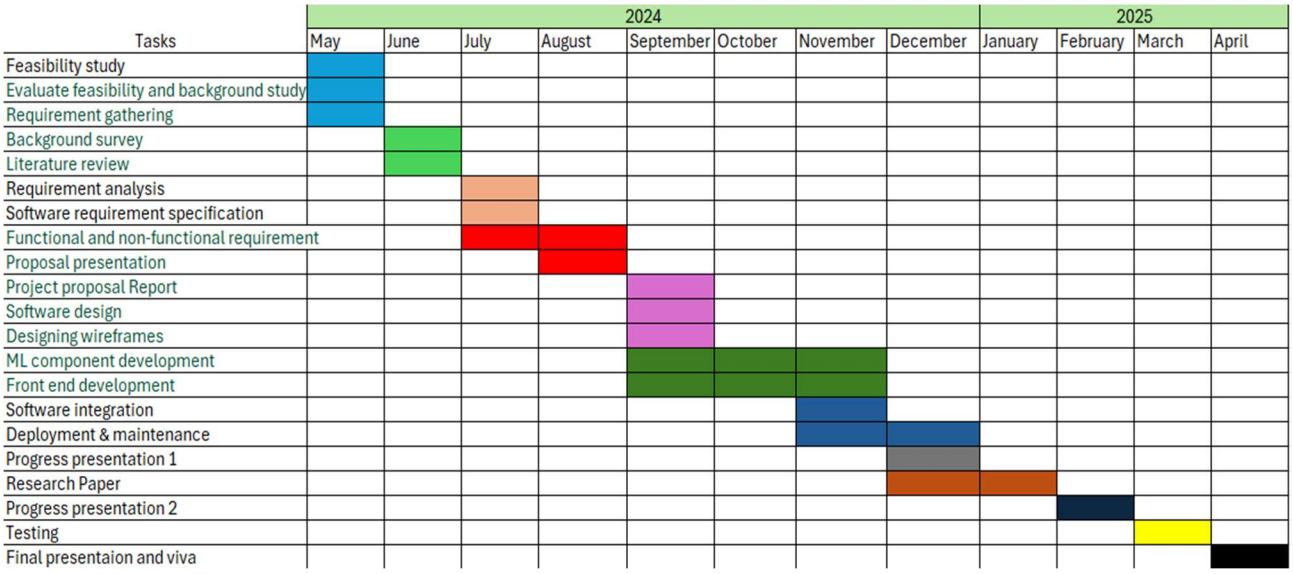

Figure 8: Gantt Chart

3.12 Risk Management Plan

The risk management plan describes the potential risks, their triggers, ownership, responses, and resources required to have a smooth development of the project. Identified below are the risks and the strategies to manage them:

Identified Risk	Risk level	Probability for occurrence of risk	Mitigation plan
Technical Failures or Bugs	High	High	Consult with experts or seek guidance from the supervisor and co-supervisor to resolve technical challenges.
Lack of Field Knowledge	High	Low	Gain foundational knowledge through online resources and courses to address any knowledge gaps.
Insufficient Technical Skills	High	Medium	Investigate existing projects and technologies to acquire necessary skills and expertise.
Changes in Project Scope	High	Medium	Review the scope in detail and determine feasible adjustments with the help of the supervisor and co-supervisor.
Difficulty in Specifying Requirements	High	Low	Use effective methods for requirement identification and consult with experts to clarify project needs.
Data Loss Due to System Failures	High	Medium	Regularly back up data and use cloud storage to safeguard against data loss.
Requested Modifications from Panel	High	Medium	Assess requested changes and make necessary adjustments within the project timeline, with support from the supervisor and co-supervisor.

Table 2: Risk Management Plan

3.13 Communication Management Plan

The Communication Management Plan ensures that all relevant parties, including team members, supervisor, and co-supervisor, are well-informed and able to perform their roles effectively throughout the project. Effective communication is key to project success and involves careful planning. This plan details how information will be shared, including the audience, content, format, frequency, and expected outcomes. It also defines stakeholder roles, how tasks will be allocated, and the communication methods suited to each stakeholder's impact and interests.

3.13.1 Communication Objectives

Effective communication should be:

- Appropriate: Ensuring the right format and content for the message.
- Targeted: Aimed at the specific audience.
- Comprehensive: Providing all necessary information.
- Concise: Clear and to the point, avoiding unnecessary details.
- Timely: Delivered at the right moments to keep everyone updated.

Following media will be used for the communication of the project:

- Email: Formal Communication, Updating and Documentation
- Documents: MS Word and PowerPoint for reports, presentations, and details
- Calls: Medium for immediate and urgent communication.
- Meetings: In-person or remote, via meeting rooms and conference phones or MS Teams.
- Discussion, Update, Decision Making
- Chats: WhatsApp for quick and informal communication and updates.

4.0 Commercialization

For the BIOMENTOR tool, commercialization will focus on two primary strategies: advertising and subscription models. The tool targets high school students and teachers, specifically for A/L Biology, and offers unique features tailored to their needs. Advertising will be utilized to promote the tool across educational platforms and social media, highlighting its benefits and unique features. Additionally, a subscription model will be implemented, providing access to premium features and updates. This approach ensures continuous revenue while offering a scalable solution to educational institutions and individual users. Both strategies aim to enhance the tool's reach and impact within the e-learning market.

5.0 Budget

As the result of the suggested model is a software-based solution, the implementation does not include any hardware. The membership fees to the cloud provider for the virtual machines' processing power will be the main source of expense.

However, additional expenses will be expected as indicated in the graph below.

Type	Cost
Internet use and web hosting	LKR.10,000.00
Training Cost	LKR.30,000.00
Publication Cost	LKR.70,000.00
Stationery	LKR.1,000.00
Total	LKR.110,000.00

Table 3: Cost Management Plan

6.0 Summary

The advanced text summarization tool for A/L biology is designed to generate concise, topic-based summaries from A/L biology textbooks and government-approved resources. This tool enhances students' understanding and exam preparation by providing clear and focused summaries. It integrates voice output technology to deliver audible summaries, benefiting auditory learners and enabling on-the-go reviews. The tool also offers customizable word count settings to accommodate individual needs. By implementing the tool in various software architectures—monolithic, layered, and microservices—the research aims to evaluate their effectiveness in delivering accurate and efficient summarization services. The goal is to improve learning experiences by providing timely, accessible content summaries and facilitating a personalized learning approach for students.

References

[1] Abstractive Text Summarization Using BART. [ONLINE] https://ieeexplore.ieee.org/document/9972639 [Accessed 15 May 2024]

[2] NLP-Enhanced Long Document Summarization: A Comprehensive Approach for Information Condensation. [ONLINE] https://ieeexplore.ieee.org/document/10551101 [Accessed 20 May 2024]

[3] Topic level summary generation using BERT induced Abstractive Summarization Model. [ONLINE] https://ieeexplore.ieee.org/document/9972639 [Accessed 01 June 2024]

[4] Speech-to-Text and Text-to-Speech Recognition using Deep Learning. [ONLINE] https://ieeexplore.ieee.org/document/9972639 [Accessed 05 June 2024]

[5] A Novel Text To Speech Conversion Using Hierarchical Neural Network. [ONLINE] https://ieeexplore.ieee.org/document/10533516 [Accessed 05 June 2024]