AI-Powered Document Summarizer

Requirements and Specification Document

October 3, 2023

Version Major 1.0

Anvit Gupta
22114009
anvit_g@cs.iitr.ac.in

Utkarsh Lohiya
22114103
utkarsh_l@cs.iitr.ac.in

Krishna Aggarwal
22114048
krishna_a@cs.iitr.ac.in

Sarvasva Gupta
22114086
sarvasva_g@cs.iitr.ac.in

Vineet Kumar
22114107
vineet_k@cs.iitr.ac.in

Project Abstract

The proposed software is an AI-powered document summarization tool that accepts various input formats, including PDF, DOCx, and text pasting and generates extractive summaries in real-time. Users can choose from short, medium, or large summary options, and the tool supports multiple languages for English summarization. Additionally, it offers word count statistics for both input and output, enhancing the user's control and understanding of the summarization process.

Document Revision History

Version	*Date*	*Description*
Rev. 1.0	*September 29, 2023*	*Created the initial version of the document outlining the project's features and objectives.*

Customer

The dummy customer for this project is a technology-familiar professional named John, who works as a content strategist in a marketing agency and frequently deals with lengthy reports, articles, and research papers related to various industries. The challenge for John is to quickly assess the content of these documents and provide concise summaries for clients. John is looking for a user-friendly and efficient tool that can process different document formats, provide options for summary length, and support multiple languages. This tool will help John streamline the summarization process and timely deliver high-quality content summaries to clients.

Competitive Landscape

N/A

System Requirements

- **Functional Requirements**:

  This section outlines the use cases for each of the activities in the program.

  1. Document Upload:
     Category: A must-have.
     Use Case: The user uploads a PDF or DOCx document.
     Description: This use case describes the scenario where a user wants to input a document into the system for summarization. The software should handle the uploaded document and extract its text content.

  2. Paste Text:
     Category: A must-have.
     Use Case: The user pastes text into the provided textbox.
     Description: This use case covers the scenario where a user prefers to directly input text by pasting it into the system. The software should accept the pasted text and process it for summarization.

  3. Summary Length:
     Category: A must-have.
     Use Case: The user can choose the desired summary length (short, medium, or large).
     Description: Users can customize the summarization process level of detail required in the summary. This use case captures these choices.

  4. Summary Language:
     Category: Optional

Use Case: The user can select the language in which the summary is desired.
Description: Users can customize the language of the summary generated. This use case captures these choices.

5. Generate Summary:
   Category: A must-have.
   Use Case: The user initiates the summarization process.
   Description: When the user is satisfied with their document input and summarization preferences, they trigger the system to generate a summary. This use case represents the core functionality of the system.

6. Word Count Display:
   Category: A must-have.
   Use Case: The system displays word counts for the input document/text and the generated summary.
   Description: This use case ensures that the system provides transparency to the user by showing word counts for both the input and output. Users can check the summary's length against their requirements.

- **Non-Functional Requirements**:

1. Performance and Speed:
   Requirement: The system must provide timely responses.
   Description: Summarization results should be generated within a reasonable time frame, even for large documents. Users should not experience significant delays in obtaining summaries.

2. User Interface Responsiveness:
   Requirement: The user interface must be responsive and provide a smooth user experience.
   Description: User interactions with the web interface, such as selecting options and generating summaries, should be responsive (work smoothly on devices of various screen sizes).

3. Scalability:
   Requirement: The system should be designed for scalability.
   Description: As usage grows, the system should be able to handle an increasing number of users and documents without significant performance degradation. Scalability should be considered for both the web application and backend components.

4. Documentation:
   Requirement: Comprehensive documentation must be provided.
   Description: The system should be accompanied by clear and thorough documentation that guides users on using the tool effectively and provides insights into its functionality.

5. Error Handling:
   Requirement: Effective error handling and user-friendly error messages are required.
   Description: The system should provide informative error messages and gracefully handle unexpected situations to guide users in troubleshooting issues.