WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Intelligent Multi-Media Content Summarization Application

Sagar Birje [a], Meghana Anil Pawar [b], Asharani K. Nivadi [c], Shreya Basavaraj Karoshi [d], Sneha Shantinath Patil [e]

[a, b, c, d, e] Department of Artificial Intelligence and Data Science, Angadi Institute of Technology and Management, Belagavi-590009, India

ABSTRACT :

The project aims to address information overload by leveraging advanced language models. The primary objective is to develop an intelligent content summarization system using large language models, capable of extracting key information from lengthy texts while preserving contextual nuances. Through state-of-the-art natural language processing techniques, the system generates concise and coherent summaries across various content types, including news articles, research papers, documents, blogs, and YouTube videos. This aligns with the evolving landscape of information processing, offering a promising solution to enhance efficiency and accessibility in navigating the digital knowledge repository.

Keywords: Content Summarization, Large Language Models, Natural Language Processing. Generative AI.

1. Introduction :

In today's digital landscape, the relentless surge in data volume necessitates efficient condensation methods that retain both information and meaning. This paradigm shift requires innovative solutions to navigate and distil the wealth of available data. The escalating quantities of data underscore the need for sophisticated approaches to information processing, especially as we rely on diverse media formats, including text-based content, images, and videos. Intelligent systems must seamlessly integrate, comprehend, and summarize content from these various sources, handling not only the volume of data but also extracting meaningful insights from this diverse pool.

In response to this growing demand, our project aims to pioneer a dynamic application that harnesses the capabilities of large language models with advanced natural language processing (NLP) capabilities. These models are crucial for the analysis, interpretation, and summarization of content across domains such as news articles, research papers, blogs, and YouTube videos. Our versatile system will efficiently condense and present meaningful information from this wide spectrum of sources. Recognizing that traditional information consumption methods are becoming obsolete, our application adapts to evolving content trends and user preferences through continuous learning and refinement. Our goal is to address information overload by creating an intelligent, adaptive application, ushering in a new era of efficient content analysis and summarization, ultimately empowering users to navigate the vast digital information landscape with ease and insight.

2. Methodology

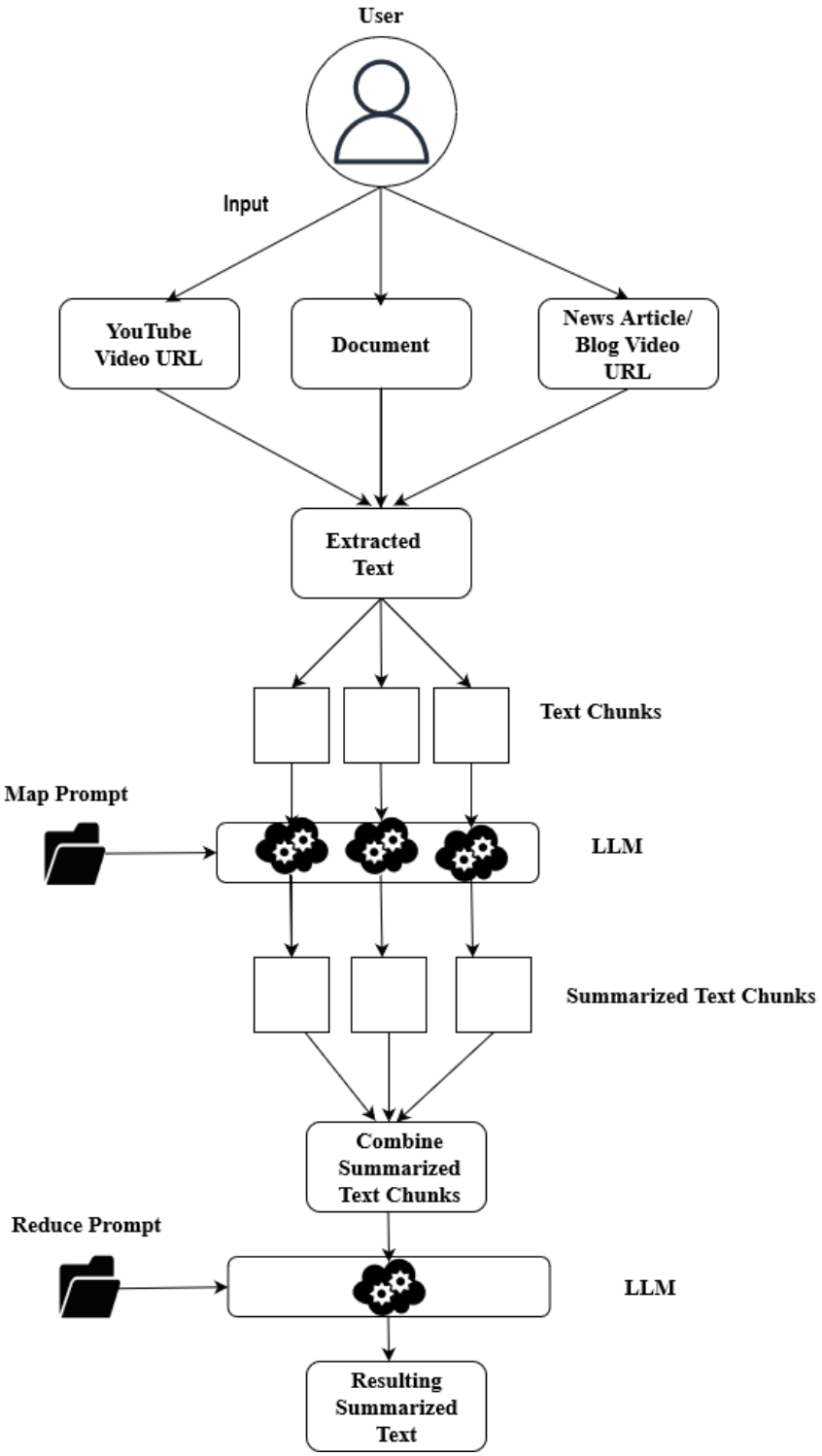

Fig. 1 - Block Schematic Diagram of the Application

The operational workflow of the summarization system, as depicted in Fig 1, follows a comprehensive and systematic approach to distilling essential information from input text. This structured process unfolds in distinct stages, each playing a crucial role in generating a coherent and concise summary. Below is an in-depth exploration of each stage, highlighting its significance and contribution to the overall summarization process:

a) **Input**: The system accepts various text formats, including YouTube video URLs, document URLs, news articles, and blog video URLs.

This versatility ensures that the system can accommodate a wide range of sources, enabling users to input text from different platforms and formats seamlessly. Whether summarizing a research paper, a news article, or a YouTube video transcript, the system processes the input effectively.

b) **Extracted Text**: Once the sources are selected, the system extracts the text from these sources. This extraction process prepares the text for the subsequent summarization stages. Regardless of the source format, whether it is a YouTube video, a document, a news article, or a blog post, the system effectively retrieves the text, readying it for the summarization process.

c) **Text Chunking**: To optimize processing efficiency, the extracted text is segmented into smaller, more manageable chunks. This segmentation process breaks down the text into smaller units, making it easier for the system to process and summarize large volumes of information effectively. By breaking the text into smaller chunks, the system can distribute the workload evenly, ensuring each part of the text receives adequate attention during the summarization process.

d) **Map Stage**: During the Map stage, each text chunk is assigned a "Map Prompt". This prompt guides a Large Language Model (LLM) on summarizing the text chunk effectively. The LLM generates a concise version of the text, creating summarized text chunks that capture the essence of the original content. Each text chunk is summarized independently, allowing for parallel processing, and ensuring the summarization process remains efficient even for large volumes of text.

e) **Combine Summarized Text Chunks**: The summarized text chunks generated in the Map stage are then merged into a single stream. This consolidation process brings together the individual summaries, preparing the data for further processing in the Reduce stage. By consolidating the summarized text chunks, the system ensures that the final summary reflects the entire input text comprehensively, capturing all relevant information.

f) **Reduce Stage**: In the Reduce stage, a "Reduce Prompt" is applied to the combined summarized text chunks. This prompt provides instructions to a Large Language Model (LLM) on how to best summarize the collection of summaries. The LLM then generates a final, comprehensive summary that encapsulates the essence of the original text, providing users with a concise yet informative overview of the input data. The Reduce stage is crucial for refining the individual summaries generated in the Map stage into a cohesive and comprehensive final summary.

This structured workflow seamlessly integrates text extraction, embedding conversion, knowledge base utilization, LLM processing, and summary generation. Each stage contributes a unique set of functionalities, collectively forming the core of the summarization system. The systematic progression from raw text extraction to generating a coherent summary demonstrates the sophistication and effectiveness of the approach in distilling pertinent information from diverse textual inputs. This comprehensive and methodical summarization process aligns with the evolving landscape of natural language processing, showcasing a strategic blend of linguistic understanding and computational analysis to achieve meaningful results.

3. System Architecture

Fig. 2 – System Architecture

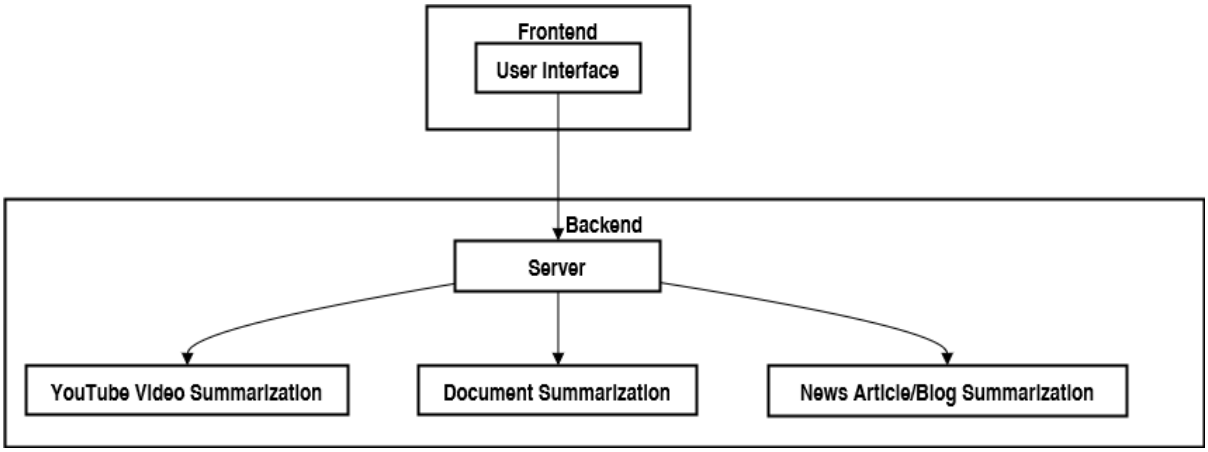

Fig. 2 illustrates the system architecture of a summarization application. The architecture is divided into two main components: Frontend and Backend. The Frontend consists of the User Interface (UI), which is the component through which users interact with the application. The Backend is responsible for the processing and summarization tasks. It includes a Server and three different summarization modules: YouTube Video Summarization, Document Summarization, and News Article/Blog Summarization.

When a user interacts with the UI, the request is forwarded to the Server. The Server then processes the request and routes it to the appropriate summarization module based on the type of content. For instance, if the request is for summarizing a YouTube video, it is directed to the YouTube Video Summarization module. Similarly, requests for document or news article summarization are routed to the Document Summarization and News Article/Blog Summarization modules, respectively.

This architecture ensures a clear separation of concerns between the frontend and backend components, making the system modular and easier to maintain. The use of a centralized Server allows for efficient handling and routing of user requests to the appropriate summarization modules. Additionally, this architecture allows for scalability, as new summarization modules can be added to the backend without significant changes to the overall system.

Overall, this system architecture provides a solid foundation for a summarization application, with clear delineation of responsibilities between the frontend and backend components, ensuring efficient and scalable operation.

4. Implementation

The application consists of three main modules:

- **Document Summarization:**
  This module allows users to summarize lengthy documents efficiently. It leverages the MapReduce summarization chain, a powerful technique for utilizing Large Language Models (LLMs) in effective document summarization, especially for large documents. The MapReduce model divides the summarization process into two distinct phases: the Map Phase, where the document is divided into smaller, more manageable chunks and fed into the LLM, and the Reduce Phase, where individual summaries are combined into a single, cohesive summary of the entire document. This module is highly beneficial for summarizing lengthy documents with improved coherence and efficiency.

- **YouTube Summarization:**
  This module enables users to summarize YouTube videos quickly and effectively. It leverages the MapReduce summarization chain, a powerful technique for utilizing Large Language Models (LLMs) to extract key information from video transcripts and generate concise summaries. By analysing the content of the video, identifying important topics, and summarizing the main points, this module provides users with an efficient way to grasp the essence of lengthy videos in a short amount of time.

- **News Article/Blog Summarization:**
  This module is designed to summarize news articles and blog posts accurately and efficiently. By leveraging the power of Large Language Models (LLMs), it analyses the content of articles and blogs, extracts important information, and generates concise summaries. This module is particularly useful for users who want to stay informed but have limited time to read lengthy articles. It provides them with a quick and efficient way to grasp the main points of an article or blog post without having to read the entire text.

4.1 Document Summarization

This module allows users to summarize lengthy documents efficiently by leveraging the MapReduce summarization chain. This technique is a powerful method for utilizing Large Language Models (LLMs) in effective document summarization, particularly for large documents. The MapReduce model divides the summarization process into two distinct phases:

- **Map Phase:**
  In this phase, the document is divided into smaller, more manageable chunks, which are then fed into the LLM. Each chunk is analysed separately, allowing the LLM to focus on a smaller piece of information within its context window limitations.

- **Reduce Phase:**
  In this phase, the individual summaries generated in the map phase are combined into a single, cohesive summary of the entire document. This step involves various strategies, such as merging the summaries directly or selecting the most salient points across summaries.

By breaking down the document into smaller chunks and analysing them individually, this module improves the coherence and efficiency of the summarization process. It is highly beneficial for summarizing lengthy documents, enabling users to obtain concise and informative summaries with ease.

4.2 YouTube Summarization

This module enables users to summarize YouTube videos quickly and effectively by leveraging the MapReduce summarization chain. This technique utilizes Large Language Models (LLMs) to extract key information from video transcripts and generate concise summaries. By analysing the content of the video, identifying important topics, and summarizing the main points, this module provides users with an efficient way to grasp the essence of lengthy videos in a short amount of time.

The MapReduce summarization chain divides the summarization process into two distinct phases:

- **Map Phase:**
  During this phase, the video transcript is divided into smaller, more manageable chunks, which are then analysed by the LLM. Each chunk is processed individually, allowing the LLM to focus on specific sections of the video.

- **Reduce Phase:**
  In this phase, the individual summaries generated in the map phase are combined into a single, cohesive summary of the entire video. The module identifies important topics and main points, ensuring that the summary provides users with a comprehensive overview of the video's content.

By utilizing the MapReduce summarization chain, this module improves the efficiency of summarizing YouTube videos. It enables users to quickly grasp the essence of lengthy videos, making it easier to consume and understand the content.

4.3 News Article/Blog Summarization

This module enables users to summarize news articles and blog posts accurately and efficiently. Leveraging the MapReduce summarization chain, it utilizes Large Language Models (LLMs) to extract key information from the text and generate concise summaries.

The MapReduce summarization chain divides the summarization process into two distinct phases:

- **Map Phase:**
  During this phase, the news article or blog post is divided into smaller, more manageable chunks, which are then analysed by the LLM. Each chunk is processed individually, allowing the LLM to focus on specific sections of the text.

- **Reduce Phase:**
  In this phase, the individual summaries generated in the map phase are combined into a single, cohesive summary of the entire article or blog post. The module identifies important topics, main points, and key information, ensuring that the summary provides users with a comprehensive overview of the text's content.

By utilizing the MapReduce summarization chain, this module improves the efficiency of summarizing news articles and blog posts. It enables users to quickly grasp the main points of lengthy texts, making it easier to consume and understand the content.

5. Results and Conclusion

The project tackles the pervasive issue of information overload by harnessing the power of advanced language models. The primary focus is on creating an intelligent content summarization system that utilizes large language models to extract key information from extensive texts, all while preserving essential contextual nuances. Through the application of state-of-the-art natural language processing techniques, the system demonstrates its ability to generate concise and coherent summaries spanning a variety of content types, including news articles, research papers, documents, blogs, and YouTube videos. This initiative is in direct alignment with the evolving landscape of information processing, presenting a promising solution to enhance efficiency and accessibility when navigating the vast digital knowledge repository. By addressing the challenges posed by the abundance of information, the project seeks to contribute significantly to streamlining content consumption, ultimately offering a valuable tool for users seeking efficient and meaningful engagement with diverse sources of information. Through the integration of cutting-edge technologies and a commitment to preserving contextual richness, this project endeavors to play a pivotal role in shaping the future of information retrieval and comprehension.

REFERENCES :

[1] Z. S. ,. B. I. ,. J. G. N. A. S. ,. P. A. E. ,. Z. X. Y. D. D. J. F. R. ,. C. W. a. Y. P. Liyan Tang, "Evaluating large language models on medical evidence," npj Digital Medicine, 2023.
[2] Z. F. S. Z. ,. F. ,. A. S. ,. A. N. a. L. F. N. Waseemullah, "A Novel Approach for Semantic Extractive Text Summarization," Applied Sciences, 2022.
[3] H. R. E. S. A. P. B. V. T. H. N. O. S. B. Min, "Recent advances in natural language processing via pre-trained language models: A survey," ACM Computing Surveys, 2021.
[4] M. S. a. S. Thakur, "Next word prediction using deep learning: A comparative survey," IEEE, 2022.
[5] R. X. H. S. Y. J. Y. S. a. T. Z. S. Diao, "Taming pre-trained language models with n-gram representations for low-resource domain adaptation.," in 59th Annual Meeting of the Association for Computational Linguistics, 2021.
[6] N. O. a. Q. Al-Tashi, "“Arabic nested noun compound extraction based on linguistic features and statistical measures," GEMA Online Journal of Language Studies, 2018.
[7] A. S. B. U. R. Q. A. a. Y. P. A. S. B. Rawat, "Recent deep learning based nlp techniques for chatbot development:An exhaustive survey," IEEE, 2022.

[8] M. K. M. K. M. a. I. M. M. Malik, "Automatic speech recognition: a survey," in Multimedia Tools and Applications, 2021.

[9] B. E. M. F. a. Y. V. J. R. R. V. P. Marcel, "A brief history of the artificial intelligence: chatgpt: The evolution of gpt," IEEE, 2023.

[10] U. A. G. N. a. V. J. H. F. F. Xu, "A systematic evaluation of large language models of code," in CM SIGPLAN International Symposium on Machine Programming, 2022.

[11] Q. A.-T. R. Q. A. S. A. M. M. I. A. Z. M. B. S. N. A. M. A. A.-G. J. W. a. S. M. Muhammad Usman Hadi, "Large Language Models: A Comprehensive Survey of Applications, Challenges, Limitations, and Future Prospects," Techrxiv, 2023.

[12] N. S. N. P. J. U. L. J. A. N. G. Ł. K. I. P. Ashish Vaswani, "Attention Is All You Need," Neural Information Processing Systems, 2017.