YouTube Transcriber and Summarizer: Simplifying Youtube Videos with Langchain and Streamlit

Aaron Mathew Shaji[1], Dr. Chandra J[2]
[1]PG Scholar, [2]Associate Professor
Department of Computer Science
CHRIST (Deemed to be University)
Bangalore, India

Abstract—**The YouTube Transcriber and Summarizer App is a dynamic web-based solution created through the integration of the Langchain library and Streamlit framework. Leveraging this application, users can effortlessly transcribe and summarize YouTube videos by simply inputting the video's URL. The interface is thoughtfully designed for seamless user interaction, catering to content creators, researchers, and a diverse range of users. This innovation marks a significant advancement in enhancing the accessibility and comprehensibility of YouTube video content.**

Index Terms—**Text Analytics, Streamlit, Langchain, YouTube, Transcription, Summarization**

I. INTRODUCTION

The YouTube Transcriber and Summarizer app is a Python-based transcription and summarizer application that allows users to transcribe and summarize YouTube videos by simply providing the video's URL. Additionally, it fetches the video's thumbnail from the YouTube host to enhance the user experience. The app utilizes the Langchain library for transcription, the HuggingFace BART (large-sized model) for summarization and Streamlit for building the user interface. The objective of this project was to develop a convenient and user-friendly solution for transcribing and summarizing YouTube videos. By leveraging the Langchain library, HuggingFace models and Streamlit framework, the app enables users to obtain accurate transcriptions and summaries of YouTube videos effortlessly. This report provides an overview of the project, including its implementation details and functionalities. The hosted link for the app is: https://aaron-youtube-transcribe-ta.streamlit.app/.

II. TECHNOLOGIES USED

- **Python**: The primary programming language for building the app.
- **Streamlit**: The web application framework used for creating the user interface..
- **Langchain**: The library used for video transcription.
- **YouTube Transcript API**: An API for retrieving YouTube video transcripts.
- **Pytube**: A library for interacting with YouTube videos.
- **Unstructured**: A library for analyzing and processing unstructured text data.
- **Sentencepiece**: A library for tokenizing text into sentences.
- **Hugging Face API**: The API used for the text summarization aspect of the application, using the BART large-sized model.

III. METHODOLOGY

The YouTube Transcription and Summarizer App is deployed using the Streamlit framework, which allows for rapid development and deployment of web applications. The code is written in Python and follows a modular structure for improved maintainability and readability. The code can be accessed through Github.

A. Code Overview

1) **User Input**:
   The user is prompted to enter the URL of the YouTube video they want to transcribe/summarize using the `st.text_input()` function. The input URL is validated and processed to extract the YouTube video ID using the `extract_youtubeID()` function.

2) **Thumbnail Display**:
   The `get_thumbnail()` function fetches the thumbnail image from the YouTube host (`http://i.ytimg.com`) based on the extracted video ID. The thumbnail is displayed using the `st.image()` function.

3) **Transcription Retrieval**:
   The `getTranscription()` function retrieves the transcription of the YouTube video using the Langchain library. The YouTube video URL is passed to the YoutubeLoader class, which loads the video's content. The resulting transcription is displayed using the `st.write()` function.

4) **Summarization Integration**:
   The `query(payload)` function is responsible for sending the POST request (with an input) to the Hugging Face API for text summarization. From the response obtained, `summarized_output[0]['summary_text']` is used to extract the summarized text. The resulting

summarization is displayed using the `st.write()` function.

5) **User Interface**:
The Streamlit framework is used to create the user interface. If no YouTube URL is provided, a warning message is displayed, prompting the user to enter a valid URL. An expandable section provides an example URL for reference. The thumbnail, transcription and summarization are displayed in separate expandable sections for easy access and readability.

B. Deployment

The app is deployed using Streamlit's deployment platform, allowing users to access it through the hosted link.

IV. Results and Discussion

To ensure the functionality and accuracy of the YouTube Transcription and Summarizer App, the results of a sample test case conducted using a valid YouTube video URL is shown in **Fig 1** and **Fig 2**. The test case involved providing a valid URL of a YouTube video and observing the app's behavior before and after the transcription/summarization process.
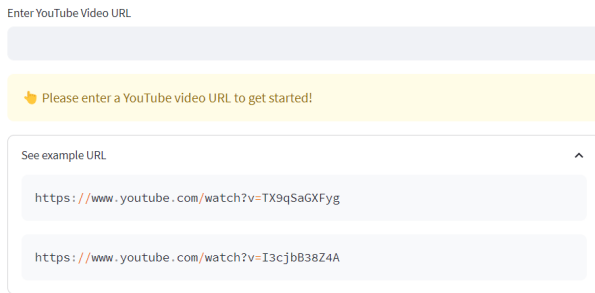

Fig. 1. Before Transcription/Summarization

As seen in **Fig 2**, the YouTube Transcription and Summarization App successfully transcribed and summarized the YouTube video associated with the provided valid URL. The app effectively processed the URL, displayed the thumbnail image, retrieved the transcription using the Langchain library, and summarized the transcription as well. The transcribed and summarized text accurately represented the content of the video, demonstrating the app's reliable capabilities.
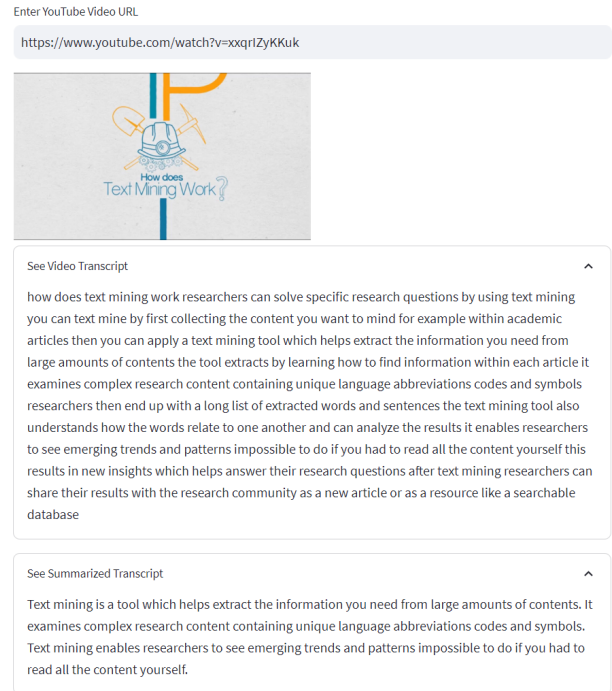

Fig. 2. After Transcription/Summarization

V. Conclusion

The YouTube Transcription and Summarization App provides a seamless solution for transcribing and summarizing YouTube videos. By leveraging the Langchain library, the HuggingFace API and Streamlit framework, the app simplifies the transcription and summarization processes and enhances the user experience. The app's user-friendly interface and efficient transcription/summarization capabilities make it a valuable tool for content creators, researchers, and anyone who needs accurate comprehensibility of YouTube videos.

References

[1] "YouTube transcripts — Langchain," YouTube transcripts — Langchain. https://python.langchain.com/docs/modules/data_connection/document_loaders/integrations/youtube_transcript

[2] "API Reference - Streamlit Docs," API Reference - Streamlit Docs. https://docs.streamlit.io/

[3] "youtube-transcript-api," PyPI, Jun. 16, 2023. https://pypi.org/project/youtube-transcript-api/

[4] "facebook/bart-large-cnn", Hugging Face, Nov. 18, 2022. https://huggingface.co/facebook/bart-large-cnn/