Extractive and Abstractive YouTube Transcript Summarizer

Dr. K. Rohit Kumar
Associate Professor, Dept of CSE (DS), Sreyas Institute of Engineering and Technology,
Telangana,India.

Akshitha Reddy Akkati
Dept of CSE, Sreyas Institute of Engineering and Technology,
Telangana,India.

Malkari Triveni
Dept of CSE, Sreyas Institute of Engineering and Technology,
Telangana,India.

C Rohit Bharadwaj
Dept of CSE, Sreyas Institute of Engineering and Technology,
Telangana,India.

Ram Charan Satla
Dept of CSE, Sreyas Institute of Engineering and Technology,
Telangana,India.

***Abstract*— The growing volume of video content on platforms such as YouTube poses a significant challenge for users seeking to efficiently extract and understand key information without viewing entire videos .Transcription plays a vital role for people who can't hear to understand the content of video time-time ,it helps to understand the different language content video by translating it to understandable language This project presents a YouTube Transcript Summarizer leveraging the Bidirectional Encoder Representations from Transformers (BERT) model and the Flask framework to address this challenge by providing concise and accurate summaries of video transcripts. The system retrieves transcripts from user-provided video links and applies advanced Natural Language Processing (NLP) techniques, specifically utilizing BERT-based models for both extractive and abstractive summarization. The extractive summarization model, BERTSUM, identifies and selects the most important sentences from the transcript, while the abstractive summarization models, BART and T5, generate coherent and concise summaries that capture the essence of the video content. The summarized transcripts enable users to quickly grasp the main points and critical information of lengthy videos, thereby enhancing accessibility and information retrieval.**

Keywords*—*YouTube Transcript Summarizer, Video Content, Transcription, BERT , Flask Framework, Natural Language Processing (NLP), Extractive Summarization, Abstractive Summarization, BERTSUM, BART, T5, Video Transcripts, Accessibility, Information Retrieval

I. INTRODUCTION

Using the Flask framework, this project integrates backend services to handle API requests from clients and return summarized text as a response. The summarization feature is tailored for YouTube videos with properly prepared closed captions. Users can access the summarizer online,

where basic API calls provide quick results via a user-friendly webpage.

YouTube, a leading online video-sharing platform, hosts a vast array of videos generating billions of viewing hours daily. Closed captions extracted from these videos facilitate an understanding of content without requiring auditory attention. Summarizing these captions is particularly valuable for lengthy videos, allowing viewers to focus solely on relevant content and boosting productivity. By leveraging video subtitles, this summarization tool helps distill the most significant information, enhancing the user experience.

The backend system is designed to receive requests with parameters like video ID, summary algorithm choice, and desired response ratio. Summarized content is generated using various NLP techniques, making the system adaptable to future algorithmic enhancements without requiring user-side updates.

Natural language processing (NLP), a branch of artificial intelligence, enables machines to interpret human language and produce human-readable outputs. This system applies NLP to efficiently summarize lengthy video transcripts, presenting the information in a concise, fluent, and coherent manner.

YouTube: Development and Integration

The development of the "Extractive and Abstractive YouTube Transcript Summarizer" is centered around leveraging YouTube's closed captioning system and integrating NLP-based summarization techniques to deliver a seamless and efficient user experience. The following steps outline the core development process:

1. Caption Retrieval via YouTube API

- The system interacts with the YouTube API to fetch subtitles of videos provided by users. Subtitles serve as the primary input for both extractive and abstractive summarization techniques.
- This dependency ensures that the summarizer can process only videos with pre-existing, properly formatted captions.

2. Summarization Models

- **Extractive Summarization:** Key sentences or phrases are extracted directly from the captions to generate concise summaries. This approach prioritizes preserving original phrasing and key information.
- **Abstractive Summarization:** Advanced NLP models such as BERT-based or Hugging Face Transformers rephrase and restructure content to create human-like summaries. This approach allows the system to present information in a more coherent and user-friendly manner.

3. Backend Implementation with Flask

- Flask, a lightweight Python web framework, is used to handle API requests and responses. It processes user inputs, manages the summarization logic, and delivers outputs in real time.

- The backend integrates NLP models and algorithms to perform summarization tasks efficiently.

4. User Interaction and Features

- A user-friendly interface was developed using HTML, CSS, and Bootstrap. Users can easily input video links, select their preferred summarization method, and obtain results.
- Features like text-to-speech, translation, and file downloads enhance the accessibility and usability of the application.

5. Testing and Optimization

- Rigorous testing was conducted to ensure accuracy, responsiveness, and reliability of both extractive and abstractive summarization methods.
- Limitations, such as dependency on YouTube's caption quality and computational resource constraints, were identified and addressed in the system design.

II. LITERATURE SURVEY

In [1], the authors introduced two distinct methods to generate summaries and extract key keywords from YouTube videos: extractive and abstractive. They developed a user-friendly interface that enables users to easily obtain summaries using these methods. This approach allows users to save time and effort by providing concise, relevant information, eliminating the need to watch lengthy videos. This not only meets the user's needs but also offers more time for acquiring additional knowledge.

In [2], the authors propose a system for video summarization that utilizes Natural Language Processing (NLP) and Machine Learning. Their approach focuses on summarizing the transcripts of YouTube videos while maintaining key elements. With the growing volume of educational content across platforms like YouTube, Facebook, and Google, extracting useful information from videos is a challenge. Unlike images, where data can be derived from a single frame, videos require full viewing to understand the context. The authors' method involves retrieving video transcripts from user-provided links and summarizing the text using Hugging Face Transformers. The model generates summarized transcripts based on user-specified video durations.

In [3], the authors examine recent advancements in deep learning-based video summarization methods, offering a thorough review of existing technologies. They provide a framework for understanding the video summarization task and the deep learning pipelines used for this purpose. Additionally, the authors categorize the existing algorithms and track the evolution of deep learning in this field, offering recommendations for future research.

According to [4], prior methods for video summarization primarily emphasize diversity and representativeness in the generated summaries. In this study, the authors propose viewing video summarization as a content-based recommendation problem, aiming to extract the most relevant content for users overwhelmed by information. They suggest a deep neural network model that predicts

whether a video segment is valuable by considering both the segment and the video as a whole. Additionally, the paper discusses the impact of audio-visual features in summarization tasks.

From [5], we conclude that video summarization and skimming are now essential tools for managing video content. This paper reviews various abstraction techniques and introduces advanced methods for feature film skimming, including audiovisual tempo analysis and specific cinematic rules. With advancements in genre classification and video understanding, an automatic system for analyzing and navigating movie content is becoming increasingly feasible.

As stated in [6], automatic summarization techniques offer users an efficient way to access key content in a media collection. With the rise of advanced capturing devices, cloud-based summarization solutions have become less favored due to longer turnaround times. In this study, the authors propose a real-time video summarization technique for mobile platforms that generates summaries during live camera recording. The method analyzes intrinsic video data and extrinsic metadata, achieving an F-measure of 0.66 on the SumMe dataset and 0.84 on the SumLive dataset, with low power consumption on embedded systems.

In [7], the authors present an online video highlighting method for generating concise summaries of unedited videos. This approach uses group sparse coding to learn a dictionary from the video and updates the dictionary dynamically. The generated summary consists of video segments that cannot be sparsely reconstructed. The online nature of the method enables it to process long videos in real-time, with processing time close to the video's duration.

In [8], the authors propose a user attention model to estimate which parts of a video attract viewers' attention. This model uses both visual and auditory stimuli, along with some semantic understanding, to rank video content by importance. The study demonstrates that the user attention model is an effective alternative for video summarization, without relying on full semantic understanding or complex rules.

According to [9], video summarization could be enhanced by incorporating external, user-based information to overcome challenges such as the semantic gap. The goal is to create video summaries that are more relevant to individual users.

In [10], the authors survey the literature on video classification, noting that features are derived from three modalities: text, audio, and visual. The study explores various feature combinations and classification methods, offering a summary of research trends and suggesting future research directions.

III.PROBLEM STATEMENT

With the growing volume of video content shared online, it has become increasingly difficult to find the time to watch lengthy videos. Many times, videos can be longer than anticipated, and without extracting useful insights, the effort spent watching them may feel unproductive. Automatically summarizing transcripts from such videos enables quick identification of key points, helping users save time by avoiding the need to watch the entire video.

IV.WORK FLOW

The process to obtain and summarize transcript information is as follows:

1. The client sends a request to the backend server built with Flask.
2. The Flask server requests subtitles from YouTube using the YouTube API.
3. YouTube responds by providing the subtitles to the server, where text summarization is performed.
4. If subtitles are not available for the provided YouTube video link, the system will display a message indicating that no subtitles are available for the video.
5. The client receives the summarized transcript.

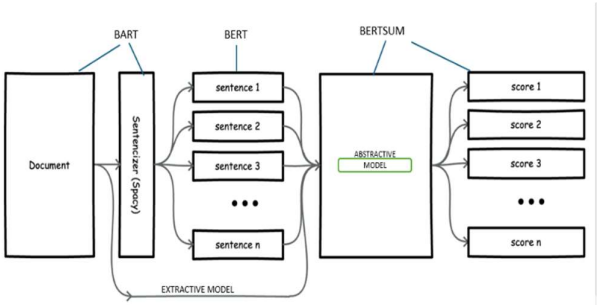

Fig 1: Architecture of the Project

1. Document Preprocessing

The process begins with a YouTube transcript or document being input into the system. This raw text is first processed to divide it into individual sentences. This step ensures that the transcript is segmented correctly, allowing downstream models to work with manageable units of data.

2. Extractive Summarization Using BERT

The segmented sentences are passed to a pre-trained BERT (Bidirectional Encoder Representations from Transformers) model. BERT is utilized as an extractive model, where it analyzes and scores each sentence based on its importance within the context of the entire document. This scoring process helps identify the most relevant sentences for summarization.

3. Abstractive Summarization Using BART

The system integrates an abstractive summarization approach with the BART (Bidirectional and Auto-Regressive Transformers) model. Instead of merely selecting sentences, the BART model rephrases and generates new sentences based on the input, capturing the essence of the document while adding a level of creativity. This enables the creation of human-like summaries that go beyond copy-pasting.

4. Hybrid Summarization with BERTSUM

The final component, BERTSUM, combines the extractive outputs (the most relevant sentences) with the abstractive model's generated summaries. BERTSUM assigns scores to both extractive and abstractive outputs, ensuring that the final summary effectively balances relevance, coherence, and fluency. This hybrid approach combines the precision of extractive summarization with the flexibility and depth of abstractive summarization.

5. Output Generation

The summarizer produces a concise and meaningful summary of the YouTube transcript. By leveraging both extractive and abstractive methodologies, the system captures key points while retaining a natural and readable flow.

This architecture ensures that large and complex transcripts are summarized in a way that is both accurate and user-friendly.

V.RESULT ANALYSIS

In this process, the transcript is generated using Natural Language Processing (NLP) techniques and Flask

.

Figure 2 illustrates the user interface where the user inputs a YouTube video link and receives the corresponding video transcript.

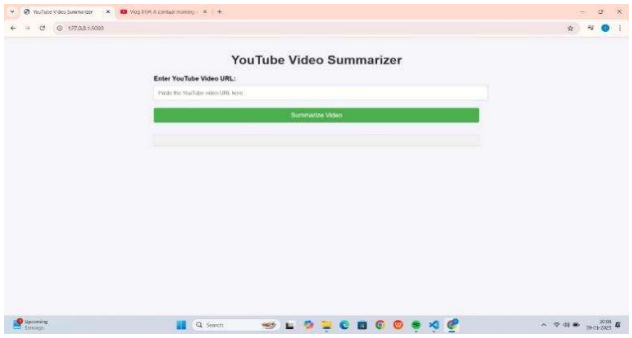

Fig 2: User Interface of Extractive and Abstractive YouTube Transcript Summarizer

Figure 3 shows the summarized text generated from a valid YouTube URL that contains subtitles.

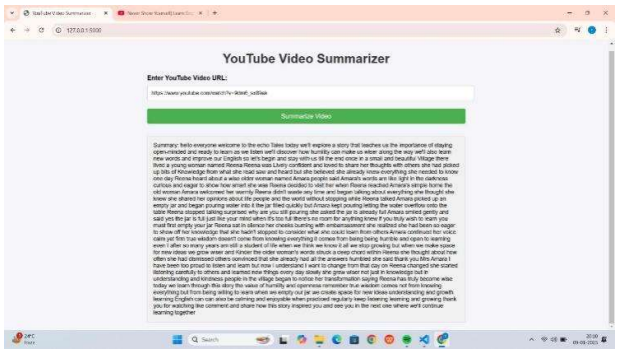

Fig 3: User Interface for accepting the url link and displays the summarized text

VI.LIMITATIONS AND FUTURE SCOPE

The YouTube transcript summarizer faces challenges such as reduced accuracy on complex or noisy transcripts and reliance on the YouTube API, which may face disruptions or quota limits. Its high computational demands, especially for abstractive summarization, make it less suitable for low-end systems. Additionally, it primarily supports English transcripts, lacking multilingual capabilities, and is not optimized for real-time summarization, limiting its use in dynamic scenarios.

Enhancements to the YouTube summarizer could include multilingual support, real-time summarization for live streams, and improved accuracy through domain-specific fine-tuning. Adding voice-to-text integration would enable summarization of videos without transcripts, and mobile or cross-platform apps would increase user accessibility and convenience.

VII.CONCLUSION

This project introduces an efficient system for summarizing YouTube video transcripts, combining a user-friendly interface with robust backend processing. By leveraging NLP techniques, this solution simplifies the process of extracting valuable insights from lengthy video content, significantly reducing user effort and improving productivity

REFERENCES

[1] Shraddha Yadav, Arun Kumar Behra , Chandra Shekhar Sahu, Nilmani Chandrakar, " SUMMARY AND KEYWORD EXTRACTION FROM YOUTUBE VIDEO TRANSCRIPT", International Research Journal of Modernization in Engineering Technology and Science Volume:03/Issue:06/June-2021 Impact Factor- 5.354 .

[2] A. N. S. S. Vybhavi, L. V. Saroja, J. Duvvuru and J. Bayana, "Video Transcript Summarizer," 2022 International Mobile and Embedded Technology Conference (MECON), 2022, pp. 461-465, doi: 10.1109/MECON53876.2022.9751991.

[3] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris and I. Patras, "Video Summarization Using Deep Neural Networks: A Survey," in Proceedings of the IEEE, vol. 109, no. 11, pp. 1838-1863, Nov. 2021, doi:10.1109/JPROC.2021.3117472. 1209 Journal of Positive School Psychology

[4] Yudong Jiang, Kaixu Cui, Bo Peng, Changliang Xu; "Comprehensive Video Understanding: Video Summarization with Content-Based Video Recommender Design"; Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 0-0.

[5] Ying Li, Shih-Hung Lee, Chia-Hung Yeh and C. . -C. J. Kuo, "Techniques for movie content analysis and skimming: tutorial and overview on video abstraction techniques," in IEEE Signal Processing Magazine, vol. 23, no. 2, pp. 79-89, March 2006, doi: 10.1109/MSP.2006.1621451.

[6] P. Choudhary, S. P. Munukutla, K. S. Rajesh and A. S. Shukla, "Real time video summarization on mobile platform," 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017, pp. 1045-1050, doi: 10.1109/ICME.2017.8019530.

[7] Bin Zhao, Eric P. Xing; Quasi Real-Time Summarization for Consumer Videos; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 2513-2520.

[8] Yu-Fei Ma, Xian-Sheng Hua, Lie Lu and Hong-Jiang Zhang, "A generic framework of user attention model and its application in video summarization," in IEEE Transactions on Multimedia, vol. 7, no. 5, pp. 907-919, Oct. 2005, doi: 10.1109/TMM.2005.854410.

[9] Video summarization: A conceptual framework and survey of the state of the art, Journal of Visual Communication and Image Representation, Volume 19, Issue 2,2008, Pages 121Arthur G. Money, Harry Agios, - 143, ISSN 1047-3203.

[10] D. Brezeale and D. J. Cook, "Automatic Video Classification: A Survey of the Literature," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 38, no. 3, pp. 416-430, May 2008, doi: 10.1109/TSMCC.2008.91.