iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** II **Month of publication:** February 2024

DOI: https://doi.org/10.22214/ijraset.2024.58451

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue II Feb 2024- Available at www.ijraset.com

Clip Outliner

Nazia Sheikh[1], Shreoshi Roy[2], Prof. Dr. Sadhana Rana[3]
Computer Science and EngineeringSRMCEM Lucknow, India

Abstract: ***In the contemporary landscape, a substantial volume of video recordings floods the digital realm daily. However, sifting through these extensive recordings has become a challenging endeavor, especially when faced with time constraints. Extracting pertinent information from lengthy videos has proven arduous, often resulting in futile efforts. Clip Outliner strives to offer increased flexibility in downloading transcript summary files while streamlining the automation processes for WhatsApp and email functionalities. To mitigate this issue, the project implements an automated system for summarizing transcripts, enabling swift identification of critical patterns within the video content. By leveraging Python APIs for text transcription and subsequently employing natural language processing (NLP) techniques, the transcripts are succinctly summarized. User Interface is developed using a blend of HTML, CSS, JS, and Bootstrap, with Flask serving as the backend framework in Python. Users can conveniently download the summarized transcripts in formats like PDF and Word, facilitating easy sharing via email and WhatsApp.***

Keywords: Clip Outliner, Summarization, Natural Language Processing, Hugging Face Transformers.

I. INTRODUCTION

According to research conducted by Google, almost 33% of viewers on YouTube in India use their mobile devices to watch videos and spend more than 48 hours on the platform every month. YouTube is the primary source for each and every student where they can learn new concept and can do the self-study. But watching such lengthy videos has become challenging because it is possible to waste time without finding the desired information as our efforts may be unproductive if we fail to retrieve the relevant information we seek.

According to research conducted by Google, almost 33% of viewers on YouTube in India use their mobile devices to watch videos and spend more than 48 hours on the platform every month. YouTube is the primary source for each and every student where they can learn new concept and can do the self-study. But watching such lengthy videos has become challenging because it is possible to waste time without finding the desired information as our efforts may be unproductive if we fail to retrieve the relevant information we seek.

Accessing YouTube content, such as transcripts of videos, has now become more convenient with the assistance of the API in the Python library. We are able to view the video content directly and provide users with a summary by utilizing this benefit. One way to achieve this is through the application of Hugging Face transformer, a method for summarizing text. The generated summary is a result of using the hugging face transformer package. Typically, written descriptions are used to encapsulate the content of YouTube videos rather than automation. our model proposes the usage of a transformer package for summarizing the transcripts of the video, thereby providing a meaningful and important summary of the video. Our main concern is to summarize the data, by using the pre-trained summarization techniques.

Using the Flask framework, this backend takes API calls from the client and answers with a summary text response. This API can only be used with YouTube videos that have closed captions that have been properly prepared. The Summarizer is also available online, where users may make basic API calls and read the results on a webpage.

This backend accepts API calls from the client and responds with a summary text response using the Flask framework. This API can only be used with YouTube videos that have been correctly prepared closed captions. Users can also utilize the Summarizer online, where they can execute basic API calls and view the results on a webpage.

II. LITERATURE REVIEW

The realm of clip outliner has witnessed a transformative journeyfueled by the remarkable capabilities of deeplearning.

1) From [1], author proposed two different methods to generate summary and important keywords from the given YouTube video - extractive and abstractive. They have made a simple user interface through which users can easily get their summaries through these methods, and surely find it easy.

©IJRASET: All Rights are Reserved

2) In [2], authors propose a video summarizing system based on natural language processing (NLP) and Machine Learning to summarize the YouTube video transcripts without losing the key elements. The quantity of videos available on web platforms is steadily expanding. The content is made available globally, primarily for educational purposes. Additionally, educational content is available on YouTube, Facebook, Google, and Instagram. A significant issue of extracting information from videos is that unlike an image, where data can be collected from a single frame, a viewer must watch the entire video to grasp the context.
3) The proposed method in [3] is focuses on the recent advances in the area and provides a comprehensive survey of the existing deeplearning-based methods for generic video summarization.
4) According to [4], previous methods mainly take diversity and representativeness of generated summaries as prior knowledge in algorithm design. In this paper [4], they formulate video summarization as a content-based recommender problem, which should distill the most useful content from a long video for users who suffer from information overload.
5) From [5], we can conclude that, video summarization and skimming has become an indispensable tool of any practical video content management system. This paper [5] provides a tutorial on the existing abstraction work for generic videos and presents state-of-the-art techniques for feature film skimming.
6) As per [6], Automatic summarization techniques will give the user an easy way to look up important content of a collection of media and to browse media of their choice later. With the evolution of sophisticated capturing devices, cloud-based summarization solutions, which have a lot of turnaround time, are less preferred by end user.
7) In [7], Authors propose online video highlighting, a principled way of generating short video summarizing the most important and interesting contents of an unedited and unstructured video, costly both timewise and financially for manual processing.

III. PROPOSED METHODOLGY

Our methodology commences the following steps -

1) At first, we have to use the Abstractive Summarization technique to summarize the text. In which we have used Hugging Face Transformer *for the summarization of content.*
2) Hugging Face Transformer uses the DistilBERT model for summarization as DistilBERT is nothing more than a smaller version of the BERT technique developed by the hugging Face company.
3) In this procedure firstly extracting the captions or say subtitles through python API for a particular ID of a YouTube video.
4) In the very next step using the hugging face transformer for summarizing the obtained transcript of the YouTube video ID and for exposing the summarized version of the transcript is to create a Flask backend REST API. So that users can get the required summarized content of a YouTube video.
5) For general usage also infused this algorithm into the chrome extension so that everyone can have the access to it. Also developed chrome extension will use back-end API to display a summarized version of the text to the user.

IV. MODULE DESCRIPTION AND IMPLEMENTATION

1) *Server Side:* The first module, namely the server is a simple Flask app with an API /api/summarize?youtube video='URL' that can be used to get the summary of a YouTube video by making a simple GET XML HTTP request. The work in this module revolves around the acquiring the transcript file from the backend and making it available to process text summarization and then eventually is sent to the frontend to be displayed in the form of an API. The API is the JSON format.

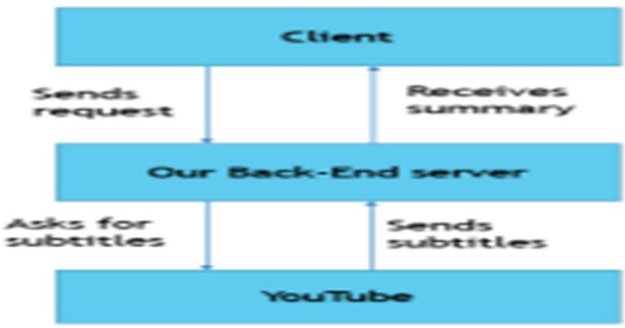


Fig 1. Workflow of the proposed project

2) *Input Module:* In a video summarization project, an input module is a crucial component responsible for collecting, processing, and preparing the raw video data for summarization. The input module plays a critical role in ensuring that the video data is well-prepared and organized before it is passed to the core summarization algorithms. It acts as the gateway for video content into the summarization pipeline, enabling the generation of meaningful video summaries or keyframes.
3) *Audio Analysis Module:* The audio analysis module is a component responsible for processing and analyzing the audio content within the video. It plays a crucial role in generating video summaries that consider not only visual but also auditory information. The key functionalities and components typically found in an audio analysis module for a video summarization project: o Audio Data Extraction o Audio Feature Extraction
4) *Natural Language Processing Module:* Integrating a Natural Language Processing (NLP) module into a video summarization project enhances content understanding. Begin by transcribing spoken words using ASR or models, extracting metadata like titles and subtitles. Preprocess text by tokenizing, cleaning, and performing tasks like stop-word removal. By integrating NLP into your video summarization project, you can provide users with more context and insights about the video content, making it easier for them to navigate and understand the material. Additionally, NLP can help automate the summarization process and improve the overall user experience.
5) *Summary Generation Module:* The Summary Generation Module in a Video Summarizer is a crucial component that condenses the content of a video into a concise and informative textual representation. Depending on the chosen approach, it uses either extractive or abstractive summarization techniques to generate a coherent textual summary of the video content. The module provides interfaces for users to interact with the summary.

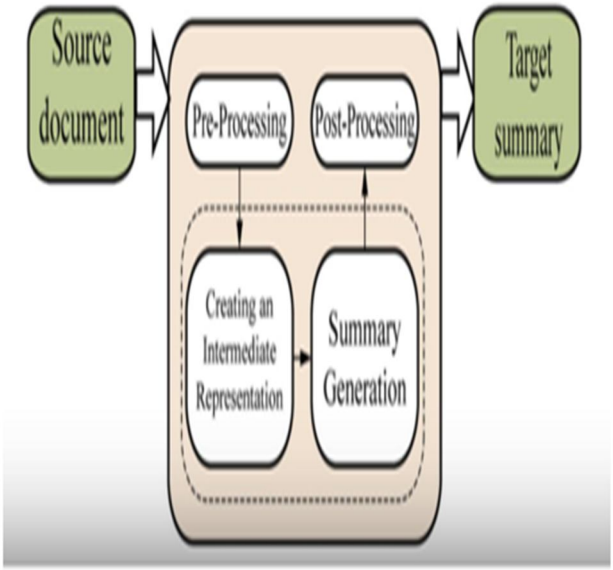


Fig.2: Outline of the proposed solution

6) *Output Module:* The Output Module in a Video Summarizer is responsible for delivering the summarized video content and associated information to users or other systems. The module may offer options for users to share the summarized video or export it in various formats for offline viewing or sharing with others. The Output Module acts as the interface through which users interact with the video summarization system, delivering the summarized content and enhancing the overall user experience
7) *Client Side:* It's a chrome addon that makes use of the API from the server module to render the summary of a YouTube video underneath the video player. Summarize button is clicked to see a synopsis of the YouTube video. The chapter of module description details about planning and structuring the effort required to implement the proposed system by dividing it into two modules. It lays out a description of each module and determines the effort required of each module. The total effort required of 100% is divided into two parts depending upon the weight of each module.

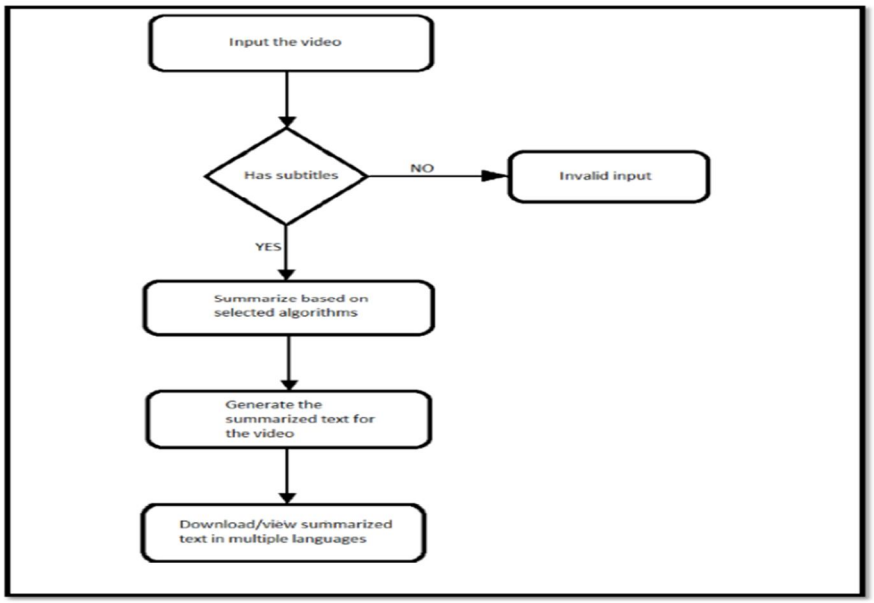


Fig 3. Flowchart of proposed project

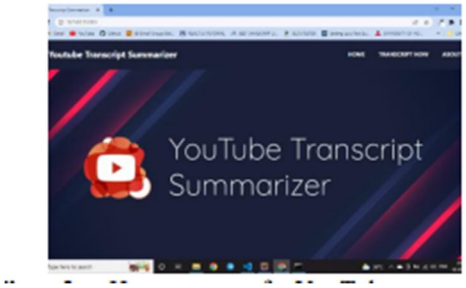


Fig 4. Homepage of YouTube transcript summarizer

V. RESULTS

YouTube Transcript Summarizer is designed for each and every person who needs to find the satisfied content they are looking for in short amount of time.

Enormous number of video recordings are being created and shared on the Internet throughout the day. It has become really difficult to spend time watching such videos which may have a longer duration than expected and sometimes our efforts may become futile if we couldn't find relevant information out of it. Summarizing transcripts of such videos automatically allows us to quickly lookout for the important patterns in the video and helps us to save time and effort to go through the whole content of the video.

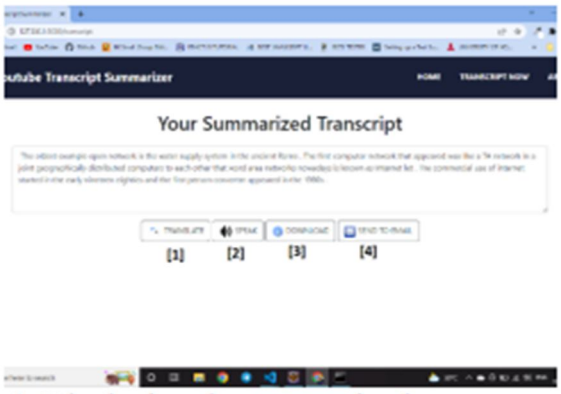


Fig 5.: Displaying the summarized text

©IJRASET: All Rights are Reserved

VI. CONCLUSION

We developed a system for transcribing YouTube videos, as well as a platform that summaries the transcript. We created a system with a simple user interface and a lot of features. We have made it possible for users to obtain their transcript files in many languages. Additionally, users can obtain a transcript file in a variety of formats. We created this system for folks who have trouble reading by including alternatives to speak and download as mp3 files. Using the send mail option, the user can send the transcript file to his or her own or any other email address. n total, we created a summarizing transcribing system with a user interface and numerous features.

A. Real Time Application

1) Transcripts the video from the given link abstractively.
2) Allows the user to translate the transcript file in different languages provided.
3) Allows user to download the transcript file in different file formats.
4) Provided the simple user interface for user convenience.
5) Decreases the efforts of user to know the contents of the YouTube video without watching

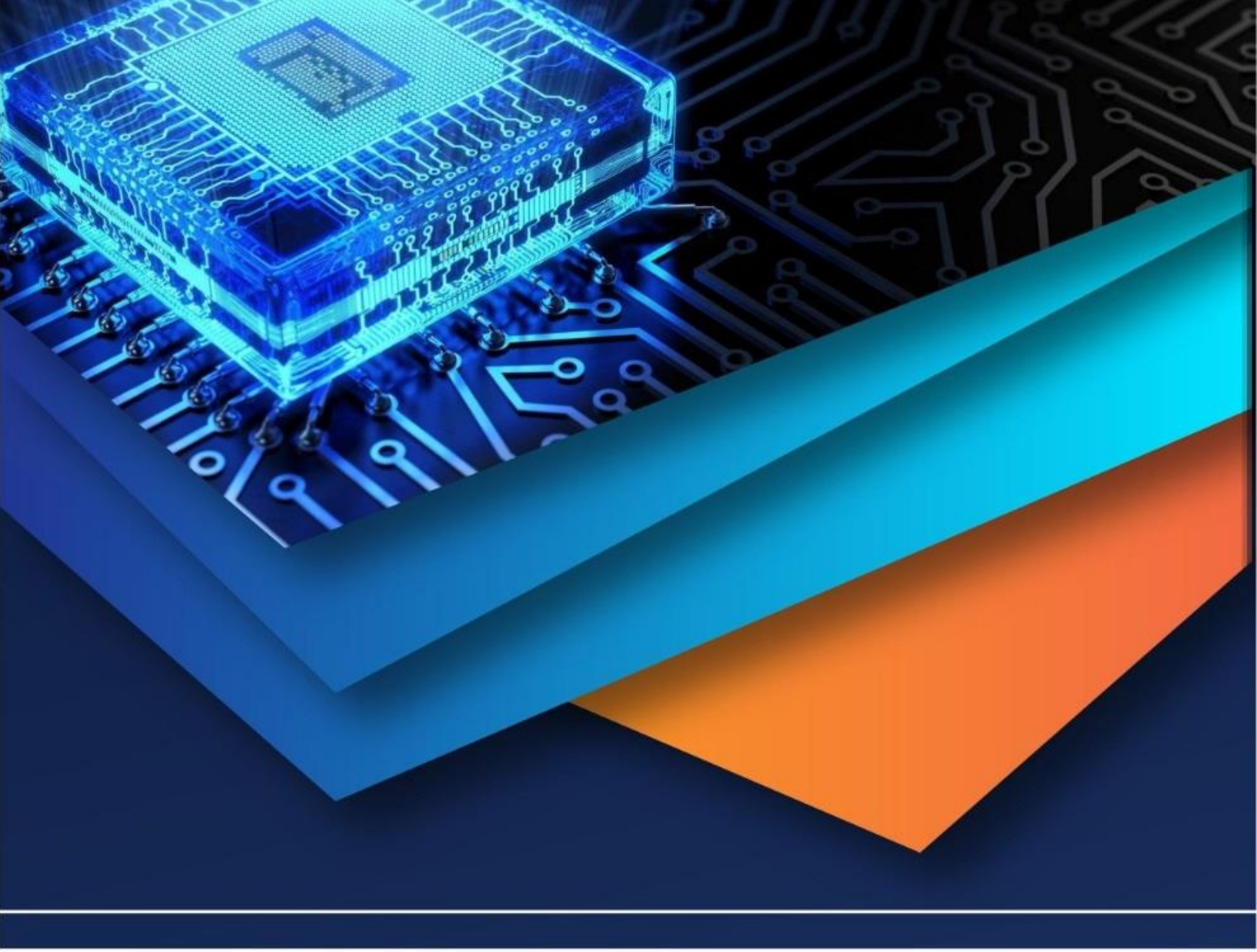
B. Limitations:

1) Transcript cannot get from the videos without subtitle.
2) Translated text other than English won't support text and pdf file formats because of encoding format

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.
[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.2008.9

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)