e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:04/April-2023 Impact Factor- 7.868 www.irjmets.com

SURVEY PAPER ON YOUTUBE TRANSCRIPT SUMMARIZER

Eesha Inamdar*1, Varada Kalaskar*2, Vaidehi Zade*3

*1.2.3Dept. Of Elec. And Telecommunication Pune Institute Of Computer Technology.

DOI : https://www.doi.org/10.56726/IRJMETS37432

ABSTRACT

The goal of this project is to construct a chrome extension that will send a request to a backend REST API, conduct NLP, and return a summary version of a YouTube transcript to improve the surfing experience without distracting from it. This procedure combines both transcript productionand text summarising. To generate the transcript, first convert the video to audio using the PyTube package, which extracts audio in mp3 format. A toolkit called hugging sound is used for text creation from audio. Spacy, an NLP library, is used for text summarising. Moreover, an API based on Flask is being developed to allow users to communicate with the backend, as well as a Chrome extension to give a user-friendly browsingexperience.

Keywords: Youtube Videos, Text Summarization, ExtracTive, Hugging sound, Pytube, Flask, Spacy.

I. INTRODUCTION

In a day, huge amount of video recordings are producedand distributed online. YouTube had approximately 2.3 billionusers in 2020, and the number has been rapidly increasingeach year. Every minute, 300 hours of video watch time is uploaded to YouTube. According to a Google study, mobile users spend more than 48 hours per month on the website. As far as videos of educational field are concerned it has become extremely challenging to commit time to watching such movies, which could go on longer than anticipated. Our actions might be rendered ineffective if we are unable to extract relevant information from it. The searching canbe aggravating and time-consuming. Therefore, we have developed a Chrome Extension that will send requests to a backend REST API, conduct NLP, and provide a condensed transcript of a YouTube video as a response.

YouTube Transcript Summarizer is mainly based on the concept of Machine Learning. It is implemented using Natural Language Processing (NLP) and some amount of Deep Learning concepts as well. The programming language used is Python. The backend part (API creation) uses Flask.

II. LITERATURE REVIEW

It has become challenging to commit time to watching such movies, which could go on longer than we anticipate, and our efforts might be in vain if we are unable to learn anything useful from them. It is frustrating and time consuming tosearch for videos that contain the information we require. For example, there are many videos available online in which the speaker speaks for an extended period of time on a specific topic, but it is difficult to find the actual content the speaker

wishes to convey to the audience unless we watch the entire video. 8 Many-a-times the video content is irrelevant due to which it may be a wastage of time for the viewers.

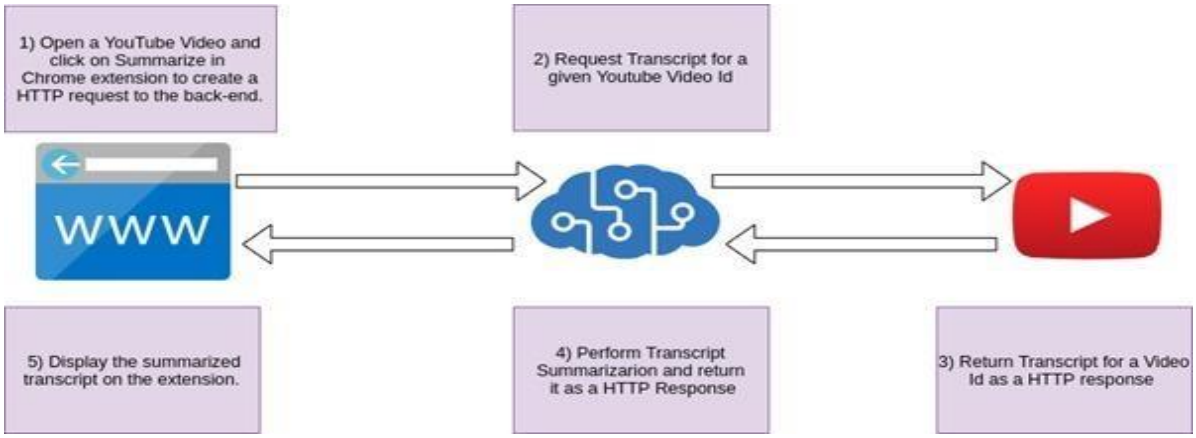

Fig. 1. Project Stages

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:04/April-2023 Impact Factor- 7.868 www.irjmets.com

Text Summarization	Extractive			Abstractive	
	Includes identifying important sections from the input text and generating an output using these important sections. Extractive text summary is a subset of the input text.			Abstractive text summarization uses natural language techniques, with the help of which, new words and phrases are generated to form the summarized output.	
	Method 1 Spacy: [7]	Method 2 Gensim: [8]			
	Spacy is an advanced natural language processing library, which can be used. Using Spacy, the input text is first filtered, cleaned, stop words, etc are removed after which normalization of the frequency of each word is done. Using this, the weightage of each sentence is calculated and thus, the sentences with highest weightage are included in the summary.	Gensim is another library which can be used. The approach called TextRank, includes cleaning the input and filtering followed by making graph with sentences that are vertices. The graph's edges denote the similarity between two sentences at the vertices. After this, PageRank algorithm is run on this weighted graph. Then, the highest scoring vertices are picked and appended to the summary.		For abstractive text summarization, one way is to use the transformers library by HuggingFace. This is an API which can be directly used to get the summary of a given input text.[9]	
Transcript Generation	Method 1	Method 2			
		Video to audio conversion[10]		Audio to text [11]	
	Converting YouTube video directly to text, using the you-get library. This python-based library uses terminal commands for the operations. The audio file can be retrieved in various outputs like .mp4, .webm, while the text output is presented in the SubRip subtitle output file(.srt).	MoviePy is a python library for video editing. This same library can be used to generate audio file from an input video file.	Pytube library can be used. It is a lightweight, Python-written library. To get the audio file from the input video file using the pytube library, the filter method is used.	AssemblyAI's speech-to-text API can be used to convert the obtained audio file into text.	HuggingSound is a toolkit for speech-related tasks based on the HuggingFace's tools. Using the HuggingSound API, the generated audio file can be converted into text.

Fig. 2. Literature Review

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:04/April-2023 Impact Factor- 7.868 www.irjmets.com

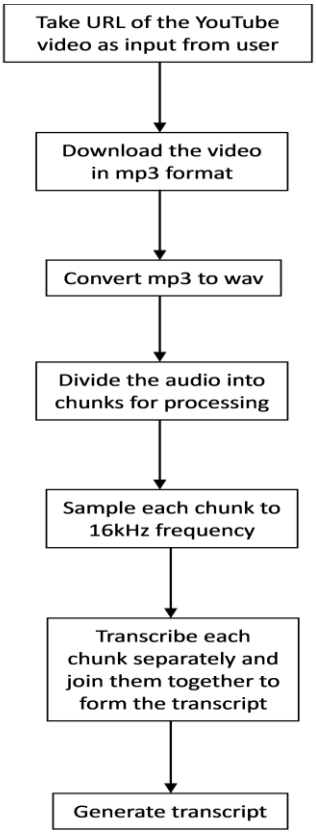

Fig. 3. Transaction Steps

III. PROPOSED METHODOLOGY

Exploring the research we figured out that summarization of YouTube videos would better work with NLP modules. After conducting the literature survey, we discovered that there are many libraries in NLP to summarize text using SpaCy, gensim. The method that is used for generating the transcript from YouTube videos includes libraries and transformers namely PyTube and Huggingsound. This is mainly because for the text summarization a depthwise parameter called cosine similarity is used which improves similarity by focusing on the importantkeywords from the text.

We propose a real-time chrome extension for giving summary of YouTube videos. Real-time audio extraction will be used to filter it from the video and then using Speech Recognition Model of huggingsound text gets generated, and dependingon the text the user will be provided summary using NLP library called SpaCy which is designed to build systems for information extraction.

IV. SYSTEM ARCHITECTURE

The diagram depicts the entire architecture of YouTubetranscript summarizer. The five major processes of the module are transcript creation from video, which includes video to audio and audio to text, text to text summarization, connecting to the backend, and creating a chrome extension. Initially, the method for generating transcript from video comprises extracting audio from video and converting it to mp3, for which the PyTube library is utilised. The user enters the active URL of a YouTube video, and the audio is extracted using the filter function of the stream class and saved to the target folder after downloading. Convert the audio file to.mp3.

V. CONCLUSION

A YouTube Transcript summarizer has been proposed for this project. The system uses the YouTube video the user has chosen as input and the Python API to acquire the video's transcripts when a user clicks the summarise button on the webpage for the Chrome extension. The transformers package is then used to summarise the transcripts that have been accessed. The user is then shown the summarised text in the chrome extension web page. This project greatly benefits users by saving them valuable time and resources.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:05/Issue:04/April-2023 Impact Factor- 7.868 www.irjmets.com

This allows us to get the gist of the video without having to watch the entire thing. Additionally, it helps the viewer spot unusual and objectionable material so that it does not ruin their experience.

VI. FUTURE SCOPE

A YouTube Transcript summarizer has been suggested forthis project. When the user clicks on the summary button onthe chrome extension web page, the system accepts the inputYouTube video from the Chrome extension of the GoogleChrome browser and accesses the transcripts of that movieusing the Python API. The transformers package is then usedto summarise the transcripts that have been retrieved. The useris then presented the summary content in the chrome extension web page. This initiative greatly benefits users by saving them precious time and resources. This allows us to grasp the idea of the video without having to watch the entire thing. It also assists the user in identifying strange and harmful information so that it does not interfere with their watching experience. This project also ensures great user interface experience in finding out the summarized text as chrome extensions have been used. This helps in getting the summarized text withoutcopying the URL and pasting at terminals or by any third-partyapplications.

ACKNOWLEDGMENT

We would like to start by sincerely thanking our project guide Prof. Mrs. Manjeeta Kale for her suggestions in choos- ing the right topic, constant and timely advices and useful inputs in making our project “Youtube Transcript Summarizer” a great success. Our appreciation also goes to our fellow members who with their moral support and timely contribution made the journey fruitful. Lastly, special appreciation for our team members for their complete coordination, honest efforts and willingness to complete the project. And we are also grateful to our H.O.D. Prof. Dr. M.V. Munot for her constant support.

1. Swaranjali Jugran; Ashish Kumar; Bhupendra Singh Tyagi, Vivek Anand, “Extractive Automatic Text Summarization using SpaCy in Python and NLP”, doi: 10.1109/ICACITE51222.2021.9404712

2. Q. Wang, Z. Mao, B. Wang, and L. Guo, “Knowledge graph embed- ding: A survey of approaches and applications,” IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 12, pp. 2724–2743, 2017.

3. H. Paulheim and P. Cimiano, “Knowledge graph refinement: A survey ofapproaches and evaluation methods,” Journal of Web Semantics: Science,Services and Agents on theWorldWideWeb, vol. 8, no. 3, pp. 489–508, 2016.

4. Article: Kamal Khumar, ”Text Summarization using Spacy”

5. Mofiz Mojib Haider, Md. Arman Hossin, Hasibur Rashid Mahi, Hossain Arif, “Automatic Text Summarization Using Gensim Word2Vec and K-

Means Clustering Algorithm” IEEE

6. Article: Sreyan Ghosh, ”Abstractive Summarization with Hugging FaceTransformers”

7. Article: Divesh Giri, ”Video To Audio Converter using moviepy inPython”

8. Article: Prashant Sharma, ”Speech to Text Conversion in Python – AStep-by-Step Tutorial”

VII. REFERENCES

[1] I. Awasthi, K. Gupta, P. S. Bhogal, S. S. Anand and P. K.Soni, ”Natural Language Processing (NLP) based Text Summariza- tion - A Survey,” 2021 6th International Conference on Inven- tive Computation Technologies (ICICT), 2021, pp. 1310-1317, doi: 10.1109/ICICT50816.2021.9358703

[2] Journal of King Saud University – Computer and Information Sciences: “ Review of automatic text summarization techniques methods ” By - Adhika Pramita Widyassari, Supriadi Rustad, Guruh Fajar Shidik. Edi Noersasongko Abdul Syukur, Affandy Affandy”, De Rosal Ignatius Moses Setiadi” Faculty of Computer Science, Dian Nuswantoro Univer- sity, Semarang, Indonesia Department of Electrical Engineering, STT Ronggolawe Cepu, Blora, Indonesia doi:

[3] P. R. Dedhia, H. P. Pachgade, A. P. Malani, N. Raul and M. Naik, ”Study on Abstractive Text Summarization Techniques,” 2020 International Conference on Emerging Trends in Informa-tion Technology and Engineering (icETITE), 2020, pp. 1-8, doi: 10.1109/icETITE47903.2020.087.

[4] A. Dilawari and M. U. G. Khan, ”ASoVS: Abstractive Summarization of Video Sequences,” in IEEE Access, vol. 7, pp. 29253-29263, 2019, doi: 10.1109/AC- CESS.2019.2902507