iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** IV **Month of publication:** April 2023

DOI: https://doi.org/10.22214/ijraset.2023.50001

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 11 Issue IV Apr 2023- Available at www.ijraset.com

Youtube Transcript Summarizer Using Flask

Surabhi Bandabe[1], Janhavi Zambre[2], Pooja Gosavi[3], Roshni Gupta[3], Prof. J. A. Gaikwad[5]
Computer Engineering Department, Datta Meghe College of Engineering, Navi Mumbai, Maharashtra, India

Abstract: ***Every day, countless videos are being created and shared on YouTube. and Such videos are the primary source of learning for college and school going students, people preparing for competitive exams and many more people use YouTube for productive outcomes. But longer than expected videos can be difficult to watch, and if we don't learn anything useful from them, our efforts might be in vain. Even sometimes while watching video people face lots of obstacles like network issues and all may lead to wastage of time. Automated summarization of video text allows us to quickly spot important trends and efficiently streamline the video's content, thus saving our time and efforts. In this YouTube transcript summarizer web application developed using flask, the transcript of the video is converted into text and thus summarizing that text and in case if there is no transcript available then the model will convert the audio directly into text using speech recognition followed by the summarization. Summary can be downloaded and translate in different language. Summarization of the video is done by using Python libraries and NLP (Natural Language Processing).***

Keywords: YouTube, Text Summarization, flask, Speech recognition, Translator, NLP.

I. INTRODUCTION

In modern times, there is a vast quantity of videos being produced and shared on youtube continuously. Globally, YouTube ranks as the second-highest frequented website.

YouTube offers a diverse selection of content, varying from short films and music videos to feature films, documentaries, corporate sponsored movie trailers, live streams, vlogs, and other material produced by famous YouTubers. Every day, video content on YouTube is being watched collectively by its users for more than a total of one billion hours. In 2020, there were approximately 2.3 billion people who used YouTube and the number of users has been quickly growing each year. At a rate of 300 hours of video uploaded per minute, YouTube constantly receives an immense amount of content.

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.

Searching for videos that contain the relevant content can be a tedious and exasperating process. Many videos posted online involve a speaker discussing a subject at length, yet it can prove challenging to locate the main message of the presentation without viewing the entire video. Python offers different packages that can be extremely useful. 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.

II. LITERATURE REVIEW

“Summary and Keyword Extraction from Youtube video Transcript” this research paper was published by Shraddha Yadav, Arun Kumar Behra, Chandra Shekhar Sahu, Nilmani Chandrakar that endeavors to utilize Natural Language Processing techniques in order to extract both summaries and significant keywords from video transcripts via extractive and abstractive summarization. Creating summaries of video transcripts can be a time-saving that results in efficiently acquiring significant and vital information. This technique eliminates the need to watch the entire video and can significantly reduce the effort invested.

©IJRASET: All Rights are Reserved

Aniqa Dilawari and Muhammad Usman Ghani Khan created "Abstractive Summarization of video Sequences." They made use of RCNN deep neural network model and multi-line video description. The flaw is that it just emphasises how succinct the summary is. Time restrictions and memory efficiency are not taken into account.

"Review of automatic text summarization techniques & methods" is developed by AdhikaPramita, SupriadiRustad, Abdul Shukur, Affandy. It was published in 2020. Text summary and systematic review techniques have been employed. The limitation of this model is that the Fuzzy based approach is weak in semantic problems. The approaches used in extractive industries need to close many gaps.

In 2021, "Natural Language Processing (NLP) based Text Summarization - A Survey" was published by Ishitva Awasthi, Kuntal Gupta, Prajbot Singh Bhojal, Anand, Piyush kumar. The techniques used for Summarisation of texts are through both Extractive and abstractive methods. The benefits involve computation of sentence implications through analyzing the linguistic and statistical features. Each summarizing method has its specific use, but there's a drawback to this variability. It is impossible to determine which technique shows more potential.

Parth Rajesh Dedhia, Hardik Pradeep, and Meghana Naik created "Research on Abstractive Text Summarization Methods". It was published in 2020. In this model seq2seq, Encoder-Decoder, and Pointer Mechanism is utilized. But the limitation of this model is that it cannot function effectively when more than one document is passed to the model.

The common factor in all the above text summarization models and in our text summarization model is that their model will give the similar output just like our model but with different methods like abstractive and Extractive methods. Our model not only converts the non transcript video to text but also tries to make that summarized text available in all the languages thus making the model more efficient and helpful.

III. REQUIREMENT ANALYSIS

A. Recommended Operating System

1) Window 7
2) Linux: Ubuntu 16.04-17.10

B. Hardware

1) *Processor:* Minimum 1 GHz; Recommended 2GHz or more
2) *Ethernet connection* (LAN) OR a wireless adapter (Wi-Fi)
3) *Hard Drive:* Minimum 32 GB; Recommended 64 GB or more
4) *Memory (RAM):* Minimum 1 GB; Recommended 4 GB or above

C. Software

1) Python
2) Visual Studio
3) Flask
4) Ffmpeg

IV. METHODOLOGY

This project will provide us the chance to put cutting-edge NLP techniques for Abstractive and Extractive text summarization into practise while also implementing an intriguing notion that is ideal for intermediates, as well as a reviving side endeavor for experts.

Steps for YouTube transcript Summarisation:-

1) Using a Python API, find the transcripts and subtitles for a particular YouTube video ID.
2) If transcripts are available then perform text summarization on obtained transcripts using HuggingFace transformers.
3) If transcript is not available then download then extract audio from the video then using speech recognition convert audio into text.
4) Summarize the converted text.
5) We can translate summarized text in hindi,english and marathi by just selecting a language to translate text.
6) If required we can download the summary in pdf format.

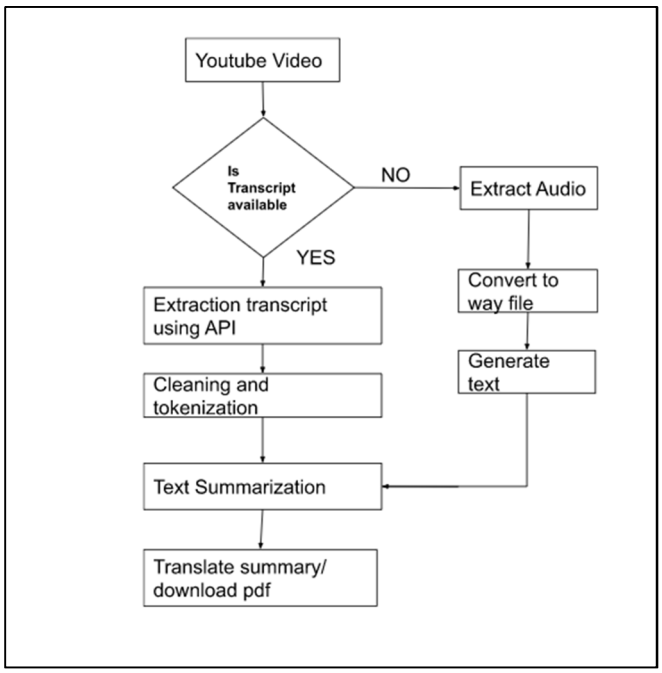


Fig -1: Flowchart

From the above System architecture, initially we open a YouTube video and click the button summarize. The subtitles will be downloaded using Youtube-Transcript-API After getting the transcripts in the text format the system performs Transcript Summarization.If transcripts are not available then extract audio from video convert it to text and summarize it Finally, it displays the summarized transcript.After that text translation is performed using google translation module in python. Also users can download a pdf of summary for their further references.

A. Backend

Main functioning of the system will be done in the python programming language. Python has various inbuilt modules like youtube-transcript-API used to get subtitles of videos. For summarization we will be using Hugging face transforms. To translate text in different languages, google translator api model will be useful.

B. Get Transcript

Using a python API called Youtube transcript api we can get the transcripts/subtitles for a given YouTube video. It also generates the transcript for youtube videos.

C. Convert Audio to text

If transcripts are not available then the system will download audio using pytube library in python.Now using ffmpeg convert the required audio file into .wav format.Using python speech recognition module system will convert audio into text.

D. Text Summarization

The process of condensing lengthier text into a concise summary while maintaining the main ideas and general meaning is known as text summarizing.

There are two methods that are frequently employed for text summarization:

1) *Extractive Summarization:* In this method, the model isolates the crucial phrases and sentences from the source text and only outputs them.
2) *Abstractive Summarization:* The model generates new sentences in a new format, resulting in an entirely distinct text that is shorter than the original. Transformers will be used in this project to implement this strategy.

In this system, abstractive text summarization will be done on the transcript received in the previous phase using the Python HuggingFace transformers module.

E. User Interface

User interface is needed to ensure that the user can interact with the system. User is done using languages like HTML, CSS and flask as a framework. It will be useful to provide users better interaction with the system.

V. ANALYSIS OF ALGORITHM

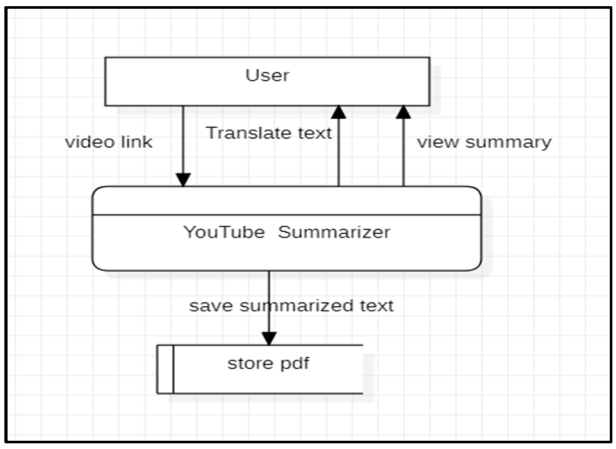


Fig -2: LEVEL 0 DFD

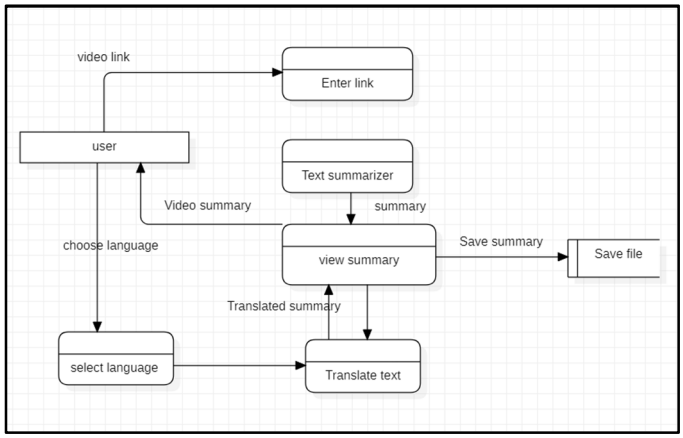


Fig -3: LEVEL 1 DFD

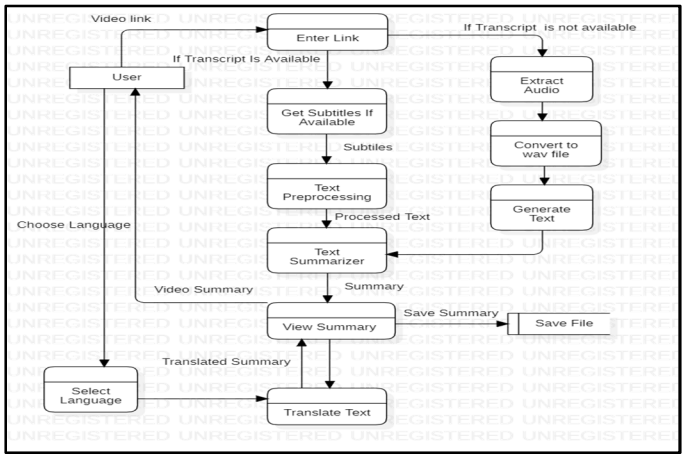


Fig -4: LEVEL 2 DFD

VI. TECHNOLOGIES USED

A. NLP

Thanks to NLP, computers can now understand natural language exactly like people do.Natural language processing employs artificial intelligence to take real-world data, interpret it, and make sense of it in a way that a computer can understand, regardless of whether the language is spoken or written.Similar to how humans use their brains to process different information, computers have programmes to do the same. During processing, the input is ultimately converted into computer-readable code.

B. Python

A high-level, all-purpose programming language is Python. Python uses garbage collection and has dynamic typing.It supports a number of programming paradigms, including structured programming, object-oriented programming, procedural programming, and functional programming (especially this).

.

C. Text Summarization

The process of creating a concise, fluid, and, most importantly, accurate summary of a lengthy text content is known as text summarization. The fundamental goal of automatic text summarization is to be able to extract the most important information from a large body of text and display it in a way that is human readable. Automatic text summarizing techniques could be particularly beneficial as online textual data increases since more informative material can be viewed quickly.

D. Google Translator

You have undoubtedly used Google Translate a lot in your life, unless you have been living under a rock. The Google Translate API is always working in the background to provide you with the appropriate translations whenever you attempt to translate a word or sentence from one language to another. Although anything may be translated by just visiting the Google Translate website, you can also include the Google Translate API into your desktop or web applications. The API's best feature is how simple it is to set up and utilize.

E. Hugging Face

Modern pretrained models can be simply downloaded and trained using the APIs and tools provided by Transformers. Pretrained models can save you the time and resources needed to train a model from scratch while lowering your compute expenses and carbon footprint. These models offer support for typical tasks across several modalities.
Using the Abstractive Summarization method, Hugging Face Transformer creates a complete, distinct text that is shorter than the original. The model creates new sentences in a new form, just like people do.

F. Speech Recognition

The ability of a machine or programme to recognise words spoken aloud and translate them into legible text is known as voice recognition, often known as speech-to-text. Voice recognition algorithms must adapt since human speech is highly contextualised and variable. The software algorithms that organise and transform audio into text are trained using a variety of speech patterns, speaking styles, languages, dialects, accents, and phrasings. The software also distinguishes speech sounds from the frequently present background noise.

G. Flask

Python-based Flask is a microweb framework. Because it doesn't need specific tools or libraries, it is categorized as a microframework. It lacks any components where pre-existing third-party libraries already provide common functions, such as a database abstraction layer, form validation, or other components. Flask allows extensions that can add features to applications as if they were built directly into the framework.

H. ffmpeg

A collection of libraries and tools for managing video, audio, and other multimedia files and streams make up this free and open-source software project. The command-line ffmpeg utility, which is designed to handle video and audio files, is its core part. It is frequently used for standard compliance, basic editing (cutting and joining), video scaling, and post-production video effects (SMPTE, ITU)

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

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue IV Apr 2023- Available at www.ijraset.com

VII. RESULT AND ANALYSIS

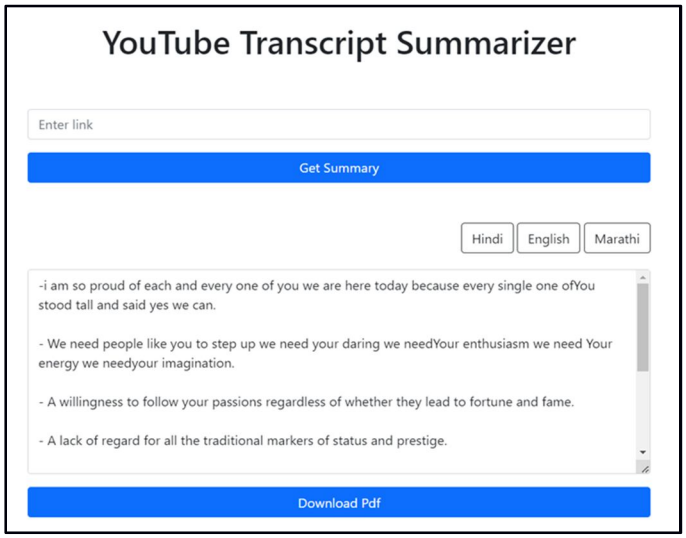


Fig -5: Screenshot of web page and summary of video

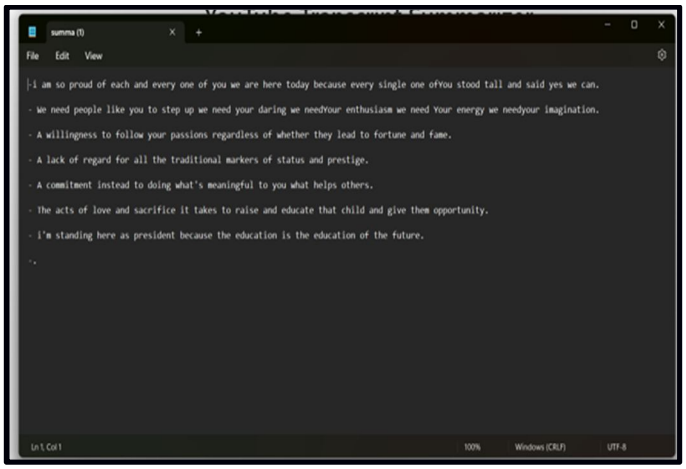


Fig -6: Download text file

VIII. CONCLUSION

This project has proposed a YouTube Transcript summarizer. The system takes the input YouTube video when the user clicks on the summarize button on the chrome extension web page, and access the transcripts of that video with the help of python API. The accessed transcripts are then summarized with the transformers package. Then the summarized text is shown to the user in the chrome extension web page.

The users of this initiative benefit greatly from the savings of their time and money. This enables us to comprehend the main points of the video without seeing the entire thing. Also, it assists the viewer in recognising strange and harmful content so that it won't interfere with their viewing experience.

IX. FUTURE SCOPE

1) This idea can be further extended to make a system that will automatically generate notes of a lecture.
2) Those who are deaf may find this useful.
3) For generating meeting notes(all important points that are covered in a virtual meeting).
4) By using this model it also arrange the important points discussed in parliament meeting and other government planning meeting

©IJRASET: All Rights are Reserved

REFERENCES

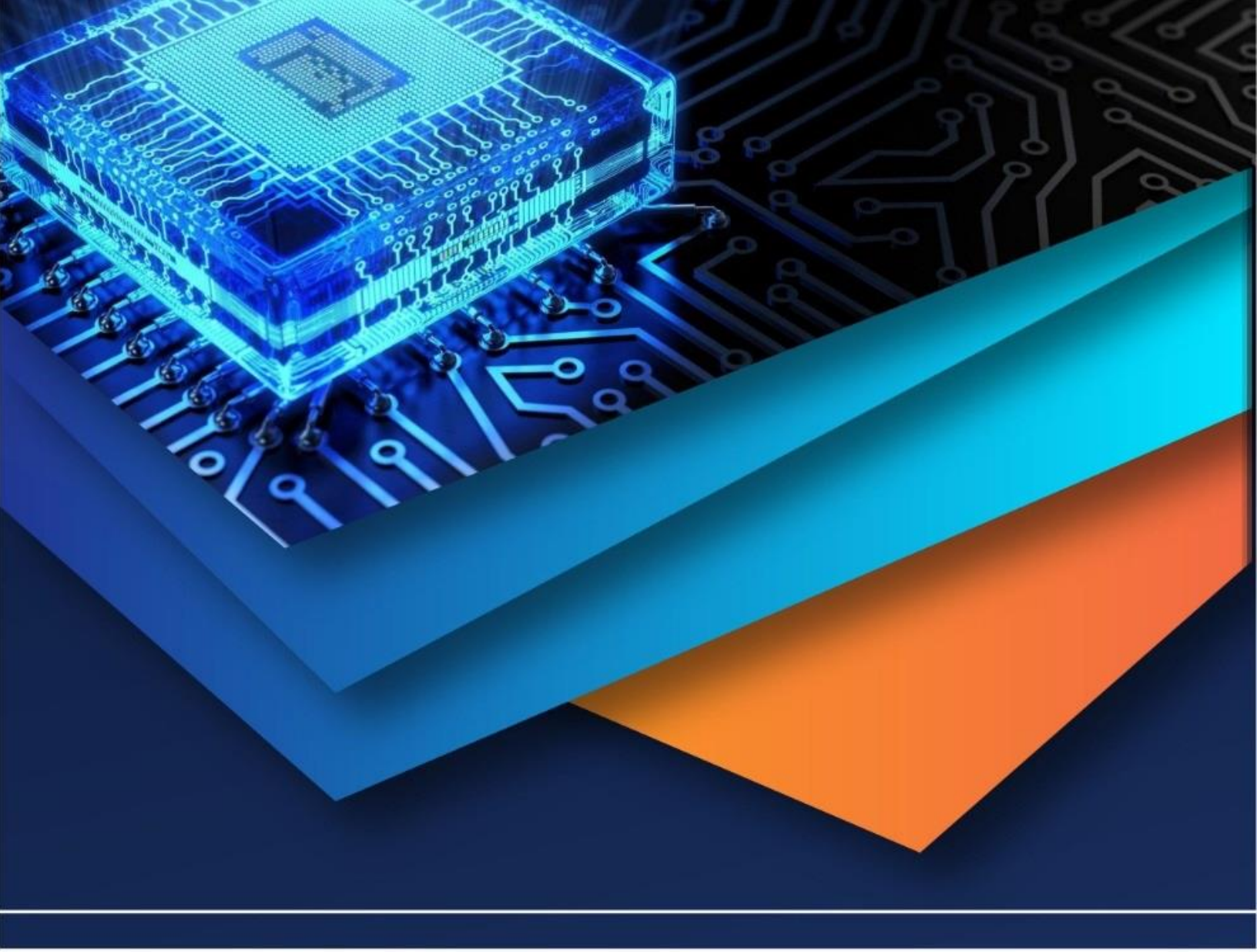
[1] IJCRT.ORG."YOUTUBE TRANSCRIPT SUMMARIZER." Ijcrt.org, Gousiya Begum , N. Musrat Sultana , Dharma Ashritha, 6 June 2022, https://ijcrt.org/papers/IJCRT22A6393.pdf. Accessed 30 March 2023.

[2] Analytic Vidya. "Creating a Youtube Summariser - Mini NLP Project." Analytics Vidhya, Basil Saji, 13 January 2022, https://www.analyticsvidhya.com/blog/2022/01/youtube-summariser-mini-nlp-project/. Accessed 30 March 2023.

[3] Rice, Damien, and Matt Galbraith. Video Transcript Summarizer, Atluri Naga Sai Sri Vybhavi, Laggisetti Valli Saroja, Jahnavi Duvvuru, JayanaBayana, 16 November 2008, https://ieeexplore.ieee.org/document/9751991. Accessed 30 March 2023.

[4] "YouTube Transcript Summarizer using Natural Language Processing." International Journal of Advanced Research in Science, Communication and Technology, https://ijarsct.co.in/Paper3034.pdf. Accessed 31 March 2023.

©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)