

Video Transcript Summarizer

Atluri Naga Sai Sri Vybhavi
Department of Computer
Science and Engineering
VR Siddhartha Engineering
College
Vijayawada, India
vybhavinaga@gmail.com

Laggiseti Valli Saroja
Department of Computer
Science and Engineering
VR Siddhartha Engineering
College
Vijayawada, India
valli.laggiseti@gmail.com

Jahnavi Duvvuru
Department of Computer
Science and Engineering
VR Siddhartha Engineering
College
Vijayawada, India
jahnaviduvvuru9@gmail.com

Jayanag Bayana
Department of Computer
Science and Engineering
VR Siddhartha Engineering
College
Vijayawada, India
jayanagb@vrsiddhartha.ac.in

Abstract— This project proposes 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. This study aims to shorten the length of the transcript text of the given video. The suggested method involves retrieving transcripts from the video link provided by the user and then summarizing the text by using Hugging Face Transformers and Pipelining. The built model accepts video links and the required summary duration as input from the user and generates a summarized transcript as output. According to the results, the final translated text was obtained in less time when compared with other proposed techniques. Furthermore, the video's central concept is accurately present in the final text without any deviations.

Keywords— *Key words: Video summarizer, NLP, YouTube, Summarizing algorithms, Transcript.*

I. INTRODUCTION

The number of YouTube users for the most part was estimated to essentially be over 2.3 billion in 2020, and it basically has been growing every year, definitely contrary to popular belief. 300 hours of YouTube videos really are generally posted per minute, demonstrating that the number of YouTube users specifically was estimated to be over 2.3 billion in 2020, and it definitely has been growing every year in a very big way. For example, there actually are pretty many Ted really Talk videos available online in which the speaker speaks for an extended period of time about a for all intents and purposes specific topic, but finding the content the speaker really is most focused on requires watching the entire video in a pretty major way. We particularly propose in this study to for all intents and purposes employ the LSA generally Natural Language Computing algorithm, which requires sort of less processing resources and requires no training data, showing how 300 hours of YouTube videos kind of are essentially posted per minute, demonstrating that the number of YouTube user kind of was estimated to

literally be over 2.3 billion in 2020, and it mostly has been growing every year in a pretty major way.

A. Motivation

Summarizing the transcript of YouTube videos to help the user have a quick glance of what is present in the video and also to increase user readability and make it comfortable. During Sem-end exams we faced this problem mainly when we want to revise the content in a short time.

B. Objectives

1. The goals of this We would like to present a method for obtaining a summary transcript from a large stream of YouTube videos, as well as an algorithm to convert the YouTube videos voice feed to text and summarize key elements.

II. LITERATURE SURVEY

[1] This paper uses Deep-learning algorithms to perform video summarization. It analyses the critical parts of the video using techniques like pipelining. As a result, it also compares the similarity of the input video and the modified video. Finally, it also determines the accuracy of the summarized text. [2] They propose a model to do the job using Natural Language Techniques like Latent Semantic Analysis. They use the Algebraic Statistical method and MoviePy Library to attach video strings based on subtitles to obtain the summarized text. One significant advantage of the model is that it has less processing power, and no prior training data is required. [3] The paper presents a method to solve traffic on internetwork by reducing the audio-visual content. It starts by selecting only essential elements from the original videos and then producing the final video, sort of a movie. Mainly useful for real-time businesses like the Entertainment field. [4] This study discusses how to summarize the video sequences with the help of deep neural networks and abstractive summarization. A joint model is proposed allowing the users to distinguish between useful and unnecessary information and also achieve better results compared with other methods in the context.[5] By explicitly modelling both segment and video, a scalable deep neural network is suggested for predicting whether one video segment is a desirable segment for the consumers. Furthermore, the study focused on perform scene and action identification in uncut videos in order to

discover more relationships between different parts of video comprehension tasks. In addition, the impact of audio and visual characteristics on the summarising task and how this model is different from the previous methods that obtained summaries based on prior knowledge is discussed. [6] The main difference in the method proposed here compared with other papers is that first, it classifies the videos as static and dynamic, then does the necessary translation of the transcript. The algorithm used here is Short boundary Detection. [7] This study focuses on developing a prototype to index videos mainly concerned with lectures employing Syntactic Similarity Measures. Based on dynamic programming techniques, captions are made available along with the video with the help of an auto-caption generator feature. It also provides a survey of existing video summarization methods. [8] A real-time video summarising technique for mobile platforms is proposed in this paper, which analyses the video during live camera recording and creates a summary in real time. The mentioned method analyses intrinsic video data, such as the video stream's contents, as well as associated external metadata, such as the video stream's external camera information. [9] The study discusses how a normal summarization system with basic features is not suitable for the user and the need for a unique customized system. The suggested method creates a video summary based on the user's preferences and the top-ranked pictures that are semantically relevant. [10] For a similar category of videos, the algorithm uses supervised learning to perform summarization. First, a set of videos is considered in that based on the summary of one video, the summary of other videos present in the same subset is generated. Each transition of a frame of video is considered a state and loss function; cross-validation scores are used. [11] This study researches how video summarizing is the key to meaningful browsing and video entity activities. It also shows how automated video summarization is possible based on accurate predictions of the transition of the video sequence. [12] The main focus is to summarize a very long video simply and understandably. For this, a hierarchal-based video transition graph and time-based constraints are used. [13] For compressed wave files, a method is given that uses a random carrier to embed the watermark in the audio signal sequence. After adaptive differential pulse code modulation and before compression, the watermark is embedded lucently in the audio stream. The proposed approach has been built, and its characteristics have been compared to the best known method of auditory watermarking. [14] A system is presented for generating elements for feature selection using support vector machines that includes the augmentation of relational notions using a classification-type method and a variety of feature generation strategies. By incorporating new techniques, classification is utilised to boost the productivity of feature space. Despite creating features in advance, feature generation in run-time resulted in the construction of models with higher accuracy.

III. PROPOSED SYSTEM

The process flow diagram for the proposed methodology describes the process of data preprocessing, and all the actions taking place in the proposed methodology. The process flow diagram is as shown in Fig 1 describes various modules like retrieving video, extracting emotion and obtaining transcripts and subtitles, and summarizing video transcript.

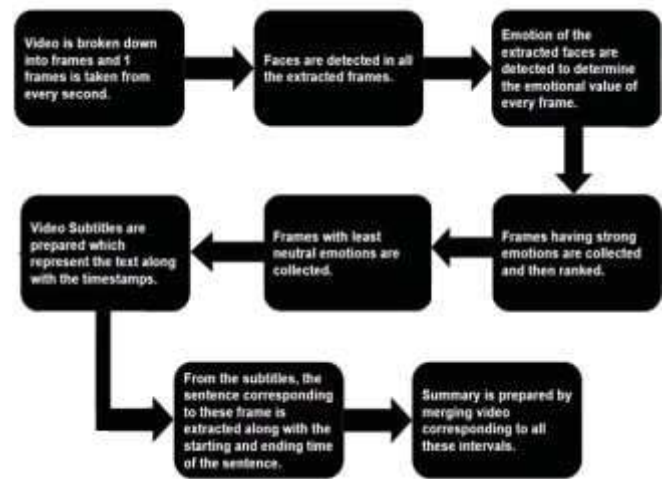


Fig 1. Process Flow Diagram

The given input video transcripts are generated. After that the data is trained and the transcripts are processed using pipelining which created a model using python Hugging Face transformers. Finally, the text derived from transcripts is summarized and displayed as output.

Firstly, from the input video transcripts respective to it are obtained in data classification phase. Next in the training phase it identifies the tone of the text received from transcripts followed by data pipelining techniques to generate the final summarised text. The architecture of the system is as shown in the figure.

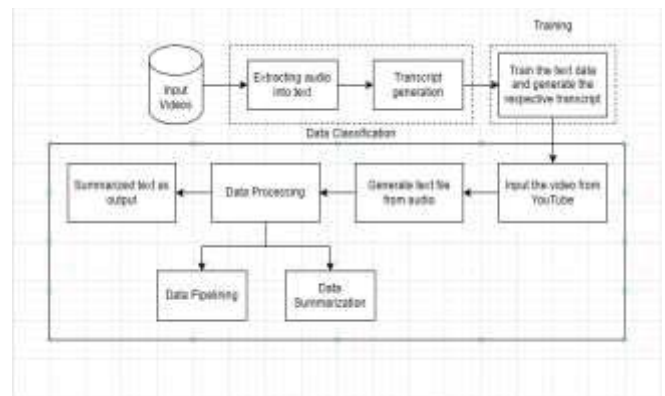


Fig 2. Architecture Diagram

A. Methodology

We noticed that many of the approaches suggested summarizing videos take considerable training and execution time after examining them. As a result, we evaluated resolving the issue. Instead of directly creating the text from the video, we used the transcripts of YouTube videos to summarize the text.

- 1) To begin, we'll use a Python API to retrieve the transcripts/subtitles for a certain YouTube video.
- 2) Obtain the transcripts using a custom function that will later be used as a feed input for the NLP engine.
- 3) Perform Extractive and Abstractive Summarization. Also, perform abstractive text summarization on the transcript produced in the previous module using Hugging Face's transformers package in Python.
- 4) Lastly, make the user interface easy so that they can interact with and examine the summarized content.

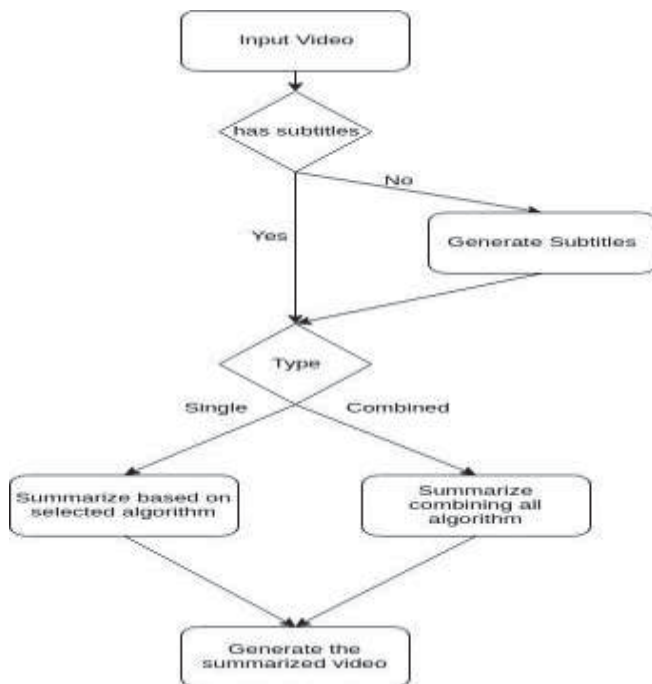


Fig 3. Applied Methodology

B. Algorithm

Module 1: Preparing the input video and obtaining required transcripts.

Algorithm

Step 1: Firstly, install transformers as they will help in data preparation

Step 2: Next, install youtube_transcript_api so that we can get the transcript of provided video

Step 3: Give the video link of the video to be summarized

Step 4: Now obtain the video id with the help of split function as follows

```
video_id = youtube_video.split("=")[1]
```

Step 5: Then display the video id

Module 2: Fetch the transcripts of the input video into a function so that they can be processed for summarization

Step 1: Display the input video for confirmation using IPython.display module

Step 2: Get the transcript of the specified video as follows.

```
transcript = YouTubeTranscriptApi.get_transcript(video_id)
```

Step 3: Observe the transcript and its contents by transcript[0:5]

```

transcript[0:5]
[{'duration': 4.96, 'start': 1.52, 'text': 'for germany it's the end of an era'},
 {'duration': 5.279, 'start': 4.4, 'text': 'and as europe's biggest economy there'},
 {'duration': 4.72, 'start': 6.48, 'text': 'are some huge challenges ahead'},
 {'duration': 3.681, 'start': 9.679, 'text': 'from its increasingly complex'},
 {'duration': 3.12, 'start': 11.2, 'text': 'relationship with china to climate'}]
  
```

Fig 4. Screenshot of the resulted output

Step 4: Now, obtain the word count of the initial video text before summarization using the following code.

```

result = ""
for i in transcript:
    result += ' ' + i['text']
  
```

Step 5: Finally, print the length of the result so we can see the word length of the video.

Module 3: Perform summarization methods with the help of pipelining on transcripts.

The model detects and outputs the relevant phrases and sentences from the actual text in Extractive Summarization.

In Abstractive Summarization, the model generates a completely new text that is significantly shorter than the original. It, like humans, develops new phrases in a different format. This method will be implemented using transformers in this project.

Step 1: Use pipeline function that will create a model with the help of Hugging Face transformers

Step 2: Now compare the initial text and its summarized version for every iteration. The related code is

```

num_iters = int(len(result)/1000)
summarized_text = []
for i in range(0, num_iters + 1):
    start = 0
    start = i * 1000
    end = (i + 1) * 1000
    print("input text \n" + result[start:end])
    out = summarizer(result[start:end])
    out = out[0]
    out = out['summary_text']
    print("Summarized text\n"+out)
    summarized_text.append(out)
  
```

Module 4: Output the summarized version of the text of the video.

Step 1: Generate the word count of summariz2ed text

Step 2: Print the summarized text by converting it into the string using str() for easy reading.

C. Dataset Used

The following dataset is used to experiment the summarization of text. The dataset is classified into six columns that contain necessary information to perform the text summarization.

A author	A date	A headline	A url, news	A text	A photo
Shivani Tyagi	01 Aug 2017, Thursday	Women & this week's most mandatory lockdowns in office order	http://www.hindustantimes.com/india/news/indianwomen/indian-women-empower-9...	The administration of Haryana Territory issued an order that made it compulsory for...	The Women and the administration on Wednesday withdrew a circular that asked women staff to the task...
Deepa Mehta	01 Aug 2017, Thursday	Malaya Akula reel was pulled bar for 'showing riot bar'	http://www.hindustantimes.com/entertainment/hollywood/malaya-akula-reel-pulled-for-showing-riots-10...	Malaya Akula claimed as Instagram user who pulled her for 'showing a riot' and 'hating the w...	From her special numbers in TV appearances, Bollywood actor Malaya Akula has managed to survive...
Ashish Chopra	01 Aug 2017, Thursday	'Virgin' now attracted to 'married' in SDRS' task	http://www.hindustantimes.com/entertainment/celebrity/ashish-chopra-virgin-now-attracted-to-married-in-sdrs-task-10...	The India's Gandhi Institute of Medical Sciences (IGIMS) in Patna on Thursday made a declaration to...	The India's Gandhi Institute of Medical Sciences (IGIMS) in Patna announced its medical declaration for...
Ramsha Sharma	01 Aug 2017, Thursday	AKI, some school boys, let men before killed	http://indianexpress.com/story/india/2017/08/01/akali-some-school-boys-let-men-before-killed-4568861.html	Ladakhers (Tibetan in Kashmir) commander Ali Durrani, who was killed by an army helicopter, said "I felt that a ...	Ladakhers (Tibetan in Kashmir) commander Ali Durrani was killed in an encounter in a village in Indian di...

Fig 5. Dataset

D. Requirements

The functional requirements of a software system define what this should be able to do. It defines the function of a software system or module. A set of inputs to the system under test is compared to the system's output to determine its functionality. The functional requirements for this project are as follows: Transcript generation, Transcript summarization, and Text analysis.

Software requirements are

1. Programming Language: Python
2. Operating System: Windows 7 (minimum)
3. Development Environment: Google Colab

IV. IMPLEMENTATION

When comparing the length of the text received from the original video to the summary text, the results clearly illustrate how the transcripts of video are summarized and the length of the text gained from the original video is reduced by more than 70%. The most difficult step is condensing the transcript without losing crucial points or distorting the sense of the original content. It is certain that no significant information is removed from the input text and that only frequently used and useless terms are eliminated from the output text by observing the input text and output text.



Fig. 6a. Video Used

```
print(len(result))
```

15880

Fig. 6b. Word count before summarization

No model was supplied, defaulted to sshleifer/distilbart-cnn-12-6 (<https://huggingface.co/sshleifer/distilbart-cnn-12-6>)






Downloading: 100%		1.76k/1.76k [00:00<00:00, 41.1kB/s]
Downloading: 100%		1.14G/1.14G [00:27<00:00, 44.3MB/s]
Downloading: 100%		26.0/26.0 [00:00<00:00, 610B/s]
Downloading: 100%		878k/878k [00:00<00:00, 2.86MB/s]
Downloading: 100%		448k/448k [00:00<00:00, 1.12MB/s]

Fig. 6c. Model supplied for execution

```
s=len(str(summarized_text))
print(s)
```

6341

Fig. 6d. Word count after summarization

''' Data economy is one of the hottest topics the emergence of AI is, refers to how much data has grown over the past few years and how much more it has got in the coming years. The explosion of data has given rise to a new economy and there is a constant b.p. $\frac{1}{2}$ ''' Data has grown at a rapid pace in the past few years and is going to continue to grow. Big data has given rise to big data which helps manage huge amounts of data. Data science is going towards a paradigm where one can teach machines to learn from data and draw a variety of useful insights giving rise to artificial intelligence. $\frac{1}{2}$ ''' Artificial intelligence is revolutionizing industries by providing greater personalization to users and automating processes using examples of artificial intelligence is given in the well-driving news. Google's algorithm is a computer program that plays the board game go it is the first computer program to defeat a world champion the ancient Chinese game of go. $\frac{1}{2}$ ''' AI recommendations.

Fig. 6e. Summarized text

V. CONCLUSIONS AND FUTURE WORK

We presented a solution to summarize the transcript of YouTube videos as it would be very useful for the user to examine the material in this project, and we introduced techniques to accurately minimize the size of text. In its approach to the problems, the suggested solution is effective and easy. The proposed strategies have the potential to reduce the length of a transcript while also maintaining its original meaning. These approaches are also responsible for deleting unwanted phrases. Only English-language YouTube videos were evaluated in this study. This study could be expanded by looking at a huge number of videos from other industries and languages.

REFERENCES

- [1] Apostolidis, Evlampios, et al. "Video Summarization Using Deep Neural Networks: A Survey." arXiv preprint arXiv:2101.06072 (2021).
- [2] Sanjana R, et al. "Video Summarization using NLP" International Research Journal of Engineering and Technology (IRJET). 2021
- [3] PRIYANKA, G., and M. PRASHA MEENA. "Survey and Evaluation on Video Summarization Techniques." Journal of Critical Reviews 7.8 (2020).
- [4] Aniqi Dilawari And Muhammad Usman Ghani Khan1 "Abstractive Summarization Of Video Sequences" 2019 IEEE
- [5] Yudong Jiang, Kaixu Cui, Bo Peng, Changliang Xu "Comprehensive Video Understanding: Video Summarization with Content-Based Video Recommender Design" International Conference on Computer Vision Workshop (ICCVW), 2019 IEEE

- [6] Holly, Smaili, Kamel, et al. "A first summarization system of a video in a target language." International Conference on Multimedia and Network Information System. Springer, Cham, 2018.
- [7] Jaiswal, Shubhangi, and Manoj Misra. "Automatic indexing of lecture videos using syntactic similarity measures." 2018 5th International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, 2018.
- [8] Pradeep Choudhary, Sowmya P. Munukutla, K. S. Rajesh, Alok S. Shukla "Real time video summarization on mobile platform" International Conference on Multimedia and Expo (ICME), 2017 IEEE
- [9] Rajkumar Kannan, Gheorghita Ghinea, Sridhar Swaminathan, Suresh Kannaiyan "Improving video summarization based on user preferences" 2013 Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)
- [10] Jayanta Basak, Varun Luthra and Santanu Chaudhury "Video Summarization with Supervised Learning" 2008 IEEE.
- [11] Wei REN Yuesheng ZHU "A Video Summarization Approach based on Machine Learning" International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2008 IEEE
- [12] Taskiran, Cuneyt M., et al. "Automated video summarization using speech transcripts." Storage and Retrieval for Media Databases 2002. Vol. 4676. International Society for Optics and Photonics, 2001.
- [13] Rohit Anand, Gulshan Shrivastava, Sachin Gupta, Sheng-Lung Peng, Nidhi Sindhvani "Audio Watermarking With Reduced Number of Random Samples" In Handbook of Research on Network Forensics and Analysis Techniques (pp. 372-394). IGI Global.
- [14] Garima Bakshi, Rati Shukla, Vikash Yadav, Aman Dahiya, Rohit Anand, Nidhi Sindhvani and Harinder Singh "An Optimized Approach for Feature Extraction in Multi-Relational Statistical Learning" Journal of Scientific and Industrial Research (JSIR).