Video Transcription and Summarization using NLP

KhushiPorwal*, Harshit Srivastava, Ritik Gupta***, ShiveshPratap Mall****, Nidhi Gupta*****

* (Department of Computer Science and Engineering, Student, Raj Kumar Goel Institute Of Technology,Ghaziabad-201003
khushirokin@gmail.com)
** (Department of Computer Science and Engineering, Student, Raj Kumar Goel Institute Of Technology,Ghaziabad-201003
harshit.srivastava1608@gmail.com)
*** (Department of Computer Science and Engineering, Student, Raj Kumar Goel Institute Of Technology, Ghaziabad-201003
guptaritik519@gmail.com)
**** (Department of Computer Science and Engineering, Student, Raj Kumar Goel Institute Of Technology, Ghaziabad-201003
shiveshmall12499@gmail.com)
***** (Department of Computer Science and Engineering, Assistant Professor, Raj Kumar Goel Institute Of Technology, Ghaziabad-201003
nidhifcs@gmail.com)

ABSTRACT-This paper proposes a transcript summarization application that works on natural language processing techniques for extracting and summarizing content from audio and video files. The video transcription consists of mainly two parts: first divide the video into several frame-based audio chunks and then the audio chunks are further divided into tokens where each token is then extracted to text. The text obtained is then given to the summarization model. The technique used for summarization is extractive text summarization which extracts summary from top ranked coherent sentences. The efficiency of summarization is evaluated by using videos of different sizes.

Keywords -*Transcription, Text Summarization, Natural Language Processing, Speech Recognition, Extractive Text Summarization*

I. Introduction

The field of artificial intelligence and natural language processing are growing exponentially, as are their applications. This helped fabricate many tools for the purpose of expediting many complex calculations and data extraction and exploration. In this fast-paced world where time is invaluable, people don't have time to watch long videos on the internet. So to refrain from investing so much time, they instead watch them twice as fast to get the impression of what is explained in the video.

The internet is flooded with lots of audio and video content, and the list keeps growing every second. In addition to this content, there are daily online meetings and events that have become a daily routine since the pandemic. It's clear that people can miss one of these meetings, lectures because of busy work, time conflicts, or other time constraints. To solve this problem, we need a set of

tools that can not only convert this audio-video content into text but also qualitatively summarize it without changing its meaning. This helps save a lot of time and the extracted text can be used in different ways.[4]

Text or text summaries themselves will be very useful products that will probably be used every day in the near future, as you can find similar applications in many areas where only audio or video content summaries are needed.

II. Related Work

2.1 Amazon Transcribe

Amazon transcribe is an automatic speech recognition service that uses machine learning models to convert speech into text. With Amazon Transcribe, you can record voice input, create easy-to-read transcripts, improve accuracy with language adaptation, and filter content to ensure customer privacy. Practical use cases include transcription and analysis of customer agent calls, and creation of video captions.

2.2 Google Transcribe

Live Transcribe is a real-time captions smartphone application developed by Google. It takes speech and turns it into real–time captions with simply the usage of the phone's mic. It permits two-way verbal exchange through a type-back keyboard for users who cannot or do not need to speak.

III. Methodology

3.1 Transcript Generation

Transcription of a video is generated in series of steps involving conversion of video into audio then into text and this is made possible using the following Python modules and libraries.

3.1.1 Moviepy

The famous python module to edit videos. By making use of this library videos can be cut, concatenated, composted, and diluted with custom effects. Titles can also be inserted using it. It is compatible with all video formats including GIFs.ffmpeg is the framework of multimedia that is required by moviepy to function. ffmpeg consists of ffplay and ffprobe. ffplay is a simple media player whereas ffprobe is a command line to display. Moviepy is required here to convert the .mp4 format of video into the corresponding .mp3 audio format. To install the moviepy command used is: pip install moviepy.

3.1.2 Pydub

This python library is meant to be used exclusively for .wav audio files. It is used to play or edit audio files and similar operations on them. To install the pydub command used is: pip install pydubAudioSegment is a wrapper class for pydub.AudioSegment. It is being used here to read the ".mp3" audio and convert it into a corresponding ".wav" format audio file.

3.1.3 Transcription

Transcription of a video is extracting the texts from a video file or precisely can be said to be a textual form of audio in that particular video. The process of generating transcription starts from an audio file with .wav format which enables us to perform operations on them. Pydub helps us with a function split_on_silence() which splits the audio into small chunks depending on the silences found in the file. A folder to create the chunks is created so that they can be fetched easily when required to be converted to text. A loop statement is used over the chunk of audio obtained to get in touch with every part of the original audio. The loop would have the heart of the whole process or the most critical thing. The chunks obtained are converted and exported into their respective .wav file. An instance of speech recognition, Recognition is created to recognize the speech from an audio source. In the loop after every chunk is obtained it is sent to the record function that records the file to the AudioData instance. The next step being the most important is the final text from that AudioData which is input to the recognize_google() function which results in the extracted text. This process is repeated and finally, the text obtained is given as the transcript of the video provided.

3.2 Text Summarization

Text summarization is one of the applications of Natural Language Processing (NLP) which can have a huge impact on our ever growing virtual life. The Covid-19 pandemic has made virtual meetings and webinars an integral part of everyone's life. Text summarization can make it easy to get the summary of these long hour meetings and webinars. For sequences like webinars and instructional programs where the caption is not available, speech recognition may be performed on the audio to obtain the transcript. Once the text corresponding to the sequence is

available, one can perform text summarization techniques to obtain a summary.

The techniques used in text summarization can be divided into two groups:

1. Statistical analysis based on information-retrieval techniques: In this approach, a subset of existing words, phrases, or sentences in the original text is selected to form the summary. The sentences are ranked based on various features. The final summary includes a few top ranked sentences.

2. Natural Language Processing (NLP) analysis based on information extraction techniques: In this technique we make use of artificial intelligence, performing a detailed semantic analysis of the source text to build a source representation designed for a particular application.[2] Then a summary representation is formed using this source representation and the output summary text is synthesized. The generated summaries contain new phrases and sentences that may not appear in the source text.

3.2.1 Text Rank Algorithm

Text Rank is a text summarization technique which is used in Natural Language Processing to generate Document summaries. It uses an extractive approach and is an unsupervised graph-based text summarization technique.

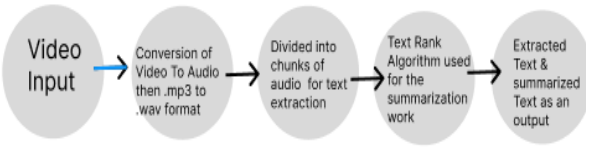

Fig 1: Working Flow of Video Summarization System

IV. Resultand Analysis

The system takes the video from either YouTube or local system and the system divides each video into several frame-based audio chunks and then the audio chunks are further divided into tokens where each token is then extracted to text using machine-learning Hugging Face Model.

Table 1 – Metric table of results

S.No.	Time Duration of video	Total size	Summary Time
1	0min 54sec	1.28Mb	30 sec
2	2min 13sec	3.81Mb	1Min
3	2min 17sec	5.34Mb	1.25 Min
4	7min 17sec	14.2Mb	1.34 Min
5	30min 21sec	43.23Mb	1.5 Min

Table 1 shows the result of the proposed algorithm used to obtain the video summarization. The algorithm gives less than 5 seconds of error video as output for the input given.

Table 2: Metric table of processing time and Memory usage

S.No.	Total Time of Video	Summary Time Requested by user	Memory Usage
1	0min 54sec	1min	22Mb
2	2min 13sec	3Min	72Mb
3	2min 17sec	3min	72Mb
4	7min 17sec	5min	102Mb
5	30min 21sec	7min	177Mb

Table 2 shows the metric table for results with memory usage and processing time. Here the memory usage and processing time results depend on the total time of the input video and the summary time requested by the user.

V. Conclusion

The number of video recordings is available on the Internet. It has become very difficult to spend time watching videos. The increase in video content on the internet requires an efficient way of representing the video. Summarizing transcripts of videos allows us to quickly lookout for the important content in the video and helps us to save time. Our video transcription model is ideal for indexing or subtitling video and/or multi-speaker content and uses machine learning technology.

We propose an algorithm to automatically summarize video programs. We use concepts from text summarization, applied to transcripts derived using automatic speech recognition. We also use temporal analysis of pauses between words to detect sentence boundaries. We have shown that the dominant word pair selection algorithm works well in identifying main topics in video speech transcripts. The problem of deriving good evaluation schemes for automatically generated video summaries is still a complex and open problem**.**

References

[1] Video Summarization using NLP Sanjana R1, Sai Gagana V2, Vedavathi K R3, Kiran K N4 *International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 08 Issue:* 08 | Aug 2021

[2] Kaiyang Zhou, Yu Qiao, Tao Xiang :*Deep Reinforcement Learning for Unsupervised Video Summarization* In: *Diversity-Representativeness Reward 2018, Association for the Advancement of Artificial Intelligence* (www.aaai.org).

[3] MrigankRochan, Linwei Ye, and Yang Wang: *Video Summarization Using Fully Convolutional Sequence Networks* In*: International Conference on Learning Representations (2018).*

[4] MayuOtani, Yuta Nakashima ,EsaRahtu , JanneHeikkil¨a , and NaokazuYokoya : Video *Summarization using Deep Semantic Features* In: *Proc. Advances in Neural Information Processing Systems (NIPS) 2016.*

[5] Wang F. and Ngo C.W. *Rushes video summarization by object and event understanding*. In TRECVID Workshop on Rushes Summarization in *ACM Multimedia Conference September 2007*.

[6] You J., Liu G., Sun L., and Li H. *A multiple visual models based perceptive analysis framework for multilevel video summarization. IEEE Trans. Circuits Syst. Video Tech., 17(3), 2007.*

[7] Ngo C.W., Ma Y.F., and Zhang H.J. *Video summarization and scene detection by graph*

modeling. IEEE Trans. Circuits Syst. Video Tech., 15(2):296–305, 2005.

[8] Xu C., Shao X., Maddage N.C., and Kankanhalli M.S. *Automatic music video summarization based on audiovisual-text analysis and alignment.*In Proc. *31st Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2005.

[9] Duan L.Y., Xu M., Chua T.S., Tian Q., and Xu C. *A MidLevel Representation Framework for Semantic Sports Video Analysis.*In Proc. *11th ACM Int. Conf. on Multimedia*, 2003.

[10] Ferman A.M. and Tekalp A.M. *Two-stage hierarchical video summary extraction to match low-level user browsing preferences. IEEE Trans. Multimedia, 5(2):244–256, 2003.*

[11] P.Sushma, Dr.S.Nagaprasad, Dr. V. Ajantha Devi. Youtube: *Big Data Analytics using Hadoop and Map Reduce in International Journal of Engineering Research* in Computer Science and Engineering *(IJERCSE), vol 5, Issue 4,* April 2018.

[12] J. Oh and K. A. Hua, "*An efficient technique for summarizing videos using visual contents*," Proceedings of the *IEEE International Conference on Multimedia and Expo (ICME 2000), July 30-August 2 2000, New York, NY*.

[13] S. Pfeiffer, R. Lienhart, S. Fischer, and W. Effelsberg, "*Abstracting digital movies automatically*," Journal of *Visual Communication and Image Processing, vol. 7, no. 4, pp. 345–353, December 1996*.

[14] R. Lienhart, "*Dynamic video summarization of home video*," Proceedings of *SPIE Conference on Storage and Retrieval for Media Databases 2000, vol. 3972, January 2000, San Jose, CA, pp. 378–389.*

[15] L. He, E. Sanocki, A. Gupta, and J. Grudin, "*Auto-summarization of audio-video presentations*," Proceedings of the 7th ACM *International Multimedia Conference, 30 October - 5 November 1999, Orlando, FL, pp. 489–498.*

[16] K. Ratakonda, I. M. Sezan, and R. J. Crinon, "*Hierarchical video summarization*," Proceedings of *SPIE Conference Visual Communications and Image Processing, vol. 3653, January 1999, San Jose, CA, pp. 1531–1541*.

[17] S. M. Iacob, R. L. Lagendijk, and M. E. Iacob, "*Video abstraction based on asymmetric similarity values*," Proceedings of *SPIE Conference on Multimedia Storage and Archiving Systems IV, vol. 3846, September 1999, Boston, MA, pp. 181–191.*

[18] L. Agnihotri, K. V. Devera, T. McGee, and N. Dimitrova, "*Summarization of video programs based on closed captions*," Proceedings of *SPIE Conference on Storage and Retrieval for Media Databases 2001, vol. 4315, January 2001, San Jose*, CA, *pp. 599–607*.

[19] M. A. Smith and T. Kanade, "*Video skimming and characterization through the combination of image and language understanding techniques*," Proceedings of the *IEEE Conference on Computer Vision and Pattern Recognition, June 1997*, San Juan, PR.

[20] M. Christel, A. G. Hauptman, A. S. Warmack, and S. A. Crosby, "*Adjustable filmstrips and skims as abstractions for a digital video library*," Proceedings of the *IEEE Conference on Advances in Digital Libraries, May 19-21-1999*, Baltimore, MD.