Abstractive Summarizer for YouTube Videos

Sulochana Devi(✉), Rahul Nadar, Tejas Nichat, and Alfredprem Lucas

Department of Information Technology, Xavier Institute of Engineering, Mumbai, India
sulochana.d@xavier.ac.in

Abstract. The paper goal is to design a user interface where the user can get the summary of the requested YouTube video using Natural Language Processing (NLP) and Machine Learning. Enormous number of videos are uploaded to YouTube daily. It has become difficult to find the relevant content that we are looking for, sometimes it may take longer than expected, and our efforts be- come futile if we are unable to extract meaningful information from it. This is where our project becomes handy. It will summarize the video and display the summary of it. Abstractive summarization model extracts YouTube video transcripts and generates a summarized version. It saves our time by shortening the content to read while maintaining the important content of the actual document. The implementation process is still ongoing. However, the entire structure and studies are presented.

Keywords: Natural Language Processing · Machine Learning · Abstractive summarization

1 Introduction

In this era of digitalization, the importance of technology is the most fundamental basis of improvement in today's society. With a massive number of users accessing the internet throughout the day and growing at an alarming rate for a variety of rea- sons, it has become really frustrating and time-consuming to search for relevant in- formation on the internet. The ease with which content producers may reach a huge audience via YouTube is excessive. As a result, there is a wide selection of content available. Around 3.7 million videos are uploaded on YouTube every day. So it has become a task to find the perfect video with the required information contained within it. The disadvantage of this is that it encourages the creation of endless click bait videos, which waste the user's time. For a given search query, there are so many results in which there are some click bait videos and some that contain very little required or relevant information for the user, which obviously they don't know about and watch the whole video till the last minute and end up with very little to zero information.

By summarizing, you can learn how to identify the key concepts in a text or video, how to filter out unimportant material, and how to connect the main ideas in an engaging way. This model will also be able to convert the summarized text's language from some given options made available in the extension which will be beneficial. A short glance at the synopsis will tell the user whether the video is worth their time

S. Tamane et al. (Eds.): ICAMIDA 2022, ACSR 105, pp. 431–438, 2023.
https://doi.org/10.2991/978-94-6463-136-4_36

and whether it covers the themes they are looking for. It can also be used to briefly summarize important information from the video. This can be achieved by two types of summarization extractive and abstractive.

Extractive and abstractive are the two most common ways to summarize text. Extractive approaches depend only on extracting significant phrases from a document and later combining those significant phrases into a coherent summary. Abstractive method analyses the document of text, which generates a coherent summary with new words and phrases just like how humans create a syntactic and well-formed representation of the documents in their brain. This paper focuses on video summarization using abstractive summarization method while some papers are based on extractive summarization.

2 Literature Survey

From [1], to create a summary using the provided YouTube video, the author suggested a model. If the transcript of the video is not available then it will convert the audio of the video into transcript and then apply an Abstractive method for text summarization in the transcript and obtain the summary of the given YouTube link video. By giving the user only the knowledge, they need to solve their problems, their paper helps them avoid having to watch those lengthy videos and makes better use of their time.

In [2], the project had been created using an embedding layer that converted words into some vector representation so that the model could generalize the words to do any kind of prediction or summary generation. The encoder records the context of the input sequence as a hidden state vector. Finding important terms in the corpus is done using the TFIDF technique

An Automatic NLP based LSA summarization algorithm has been performed on the subtitle of the videos to generate the summary. In [3] research paper they have used the LSA Natural Language Processing algorithm, which requires less processing power and no training data required to train the algorithm.

They have used Latent Semantic Analysis (LSA) technique to extract the features of the sentences that cannot be directly mentioned. This summarization technique works in such a way that the top most ranked subtitles are taken into consideration for the final video. They found the average duration of each subtitle by dividing the Total duration of the video with the Number of subtitles. This summarization technique works in such a way that the top most ranked subtitles are taken into consideration for the final summarization.

In this paper [4], authors surveyed on Abstractive Transcript Summarization of YouTube Videos. The difference between the Extractive and Abstractive summarization methods are stated they researched on many Abstractive Summarization methods and models. To make the summary effective in the least amount of time by choosing the best summarization methods and models. They also mentioned the pros and cons of each method.

In [5], the paper “YouTube Transcript Summarizer” in which they have used Hugging Face Transformer to perform abstractive summarization on the transcripts and have used RESTAPI in the backend. The YouTube transcript is used as an input parameter for the model, which outputs a condensed summary.

From [6], It is safe to say that video summarization and skimming have evolved into essential tools for any real world video management system. This article offers a guide to the current abstraction work for typical videos. The article also discusses the authors' current work on movie skimming, which makes use of rhythmic analysis of audio and visual content as well as some cinematic conventions.

They examine a number of topics that the AMIS project has looked into in this paper. By summarizing the original video, a system for comprehending a foreign video has been created. [7] They wanted to extract the key idea from a video and translate it into English. Several subsystems have been implemented to create a larger system, with each of them posing a significant scientific challenge.

From [8] they have provided a technical background for document summarizing in this study. This essay has also covered a number of difficulties with the current summarizing techniques. They have learned from these conversations that many methodologies face numerous difficulties. For instance, the clustering-based method needs accurate information about the number of clusters it is building, and the MMR uses various strategies for the coverage and non-redundancy elements of the summary. Pre-processing and textual unit assessment are the fundamental processing, components, and resources needed to complete these steps.

The extractive strategy is the main emphasis of this study work. As a result, extractive summarization's literature is far superior to abstractive summarizations. This study looked at several strategies and techniques. [9] They come to the conclusion that combining two ways is more likely to achieve ideal outcomes and improve the quality of the summarization than utilizing only one approach.

In this paper [10] the author suggests a transcript summarizer that uses NLP methods to extract and summarize material from video files. The video transcripts are split into two sections: It will first split the video into many frame-based audio chunks, then the audio chunks will be further split into tokens, and finally each token will be extracted to text. The summarizing model is then provided with the resulting text. Extractive text summarization is a method for summarizing where it extracts summary from topranking sentences that make sense. Videos of various sizes can be used to test the effectiveness of summarization.

The paper goal is to design a chrome extension for YouTube Transcript Summarizer which has the following features over other researches:

- You can use our research to block out background noise or keep track of your to-do list to obtain a clear picture of all you need to finish.
- Because it has access to its own set of APIs, our extension is able to per- form tasks that a website cannot.
- Our extension provides many extra capabilities, such the ability for users to share via email, obtain a summary transcript, translate, or use the speak feature.

3 Proposed Work

The proposed work of YouTube Transcript Summarizer. In this model a HTTP request to the back end is created from YouTube transcript and perform transcript summarization

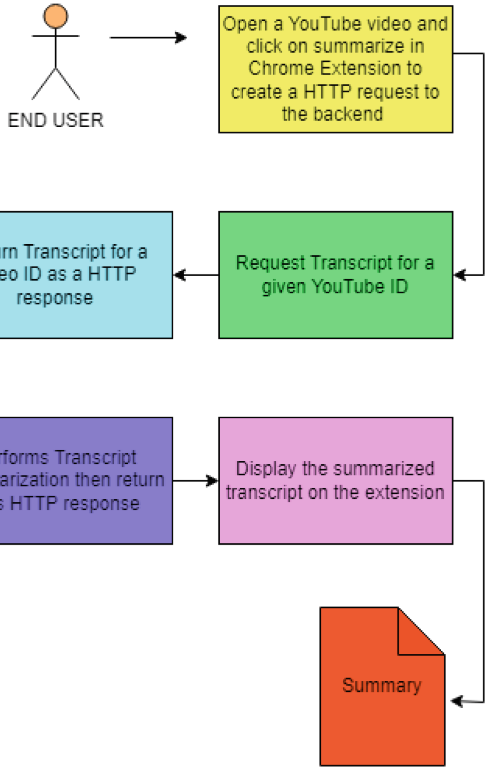
END USER
Open a YouTube video and click on summarize in Chrome Extension to create a HTTP request to the backend
Request Transcript for a given YouTube ID
Return Transcript for a video ID as a HTTP response
Performs Transcript Summarization then return it as HTTP response
Display the summarized transcript on the extension
Summary

Fig. 1. Project Stages

and return it as HTTP response then the summarized transcript is shown on the Extension. Below is a detailed explanation of the features and method- ology of the summary utilized for this purpose (Fig. 1).

The processes taken to complete the transcript information are as follows: (Fig. 2).

1. Open a YouTube video and click on summarize in Chrome Extension to create a HTTP request to the back end.
2. Request Transcript for the given YouTube ID.
3. Perform transcript Summarization and return it as HTTP response.
4. The summarized transcript is shown on the extension.

3.1 Methodology

Our transcript summarization method is divided into following parts:

3.1.1 Text Cleaning

We'll make use of the Spacy library. Spacy is made to be built for systems that extract information and understand natural language. The Spacy library has the ability

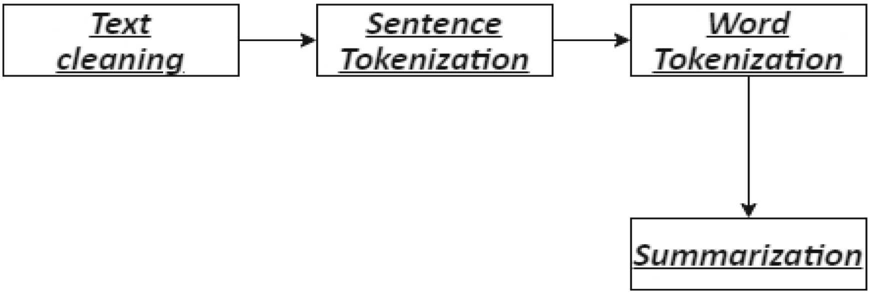

Fig. 2. Process of Summarization

to segment text into words, punctuation, and assign word roots. Additionally, Spacy is capable of serialization and text classification. Using spacy any text may be turned into a processed Doc object using this technique, and properties can be inferred.

3.1.2 Sentence Tokenization

The sent tokenize function makes use of an instance of the Punkt Sentence Tokenizer from the nltk tokenize module, which has already been trained and is quite knowledgeable about where to mark a sentence's beginning and conclusion with respect to characters and punctuation. The advantages of word tokenization using NLTK include White Space Tokenization, Dictionary Based Tokenization, Rule-Based Tokenization, Regular Expression Tokenization, Penn Treebank Tokenization, Spacy Tokenization, Moses Tokenization, and Subword Tokenization. The text normalization procedure includes all kinds of word tokenization. The accuracy of the language understanding algorithms is increased by stemming and lemmatizing the text to normalize it.

3.1.3 Word Tokenization

A sequence of strings is tokenized when it is divided up into numerous components, such as words, phrases, symbols, and other so-called tokens. A wrapper function called WordTokenize() calls tokenize() on an instance of the Treebank. Class of words in Word Tokenizer Table. Splitting a big sample of text into words is called word tokenization. This is necessary for jobs involving natural language processing, where each word must be recorded and submitted to additional analysis, such as clas- sification and counting for a specific sentiment, etc.

3.1.4 Summarization

We will be calculating the frequency of each word in our text data and store the frequency together with the text data in a dictionary. Then we tokenize the text data. We will include the sentences in our final summary data that contain more high frequency sentences which will be calculated using each word frequency.

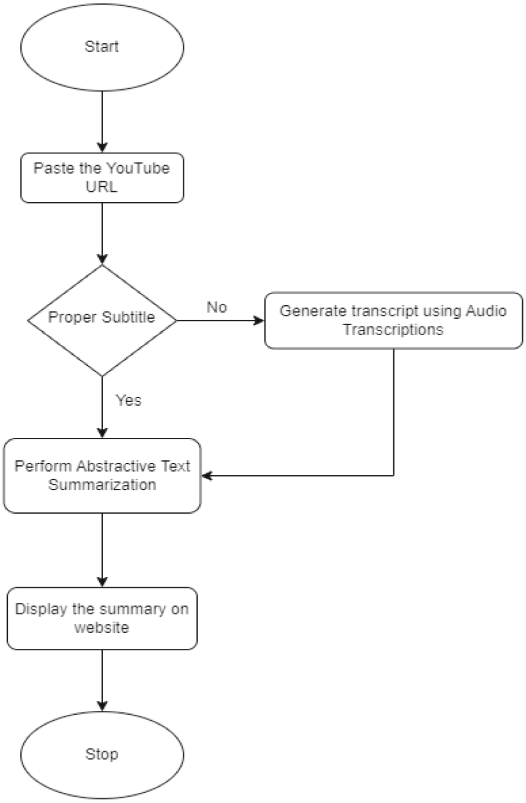
Start
Paste the YouTube URL
Proper Subtitle
No
Generate transcript using Audio Transcriptions
Yes
Perform Abstractive Text Summarization
Display the summary on website
Stop

Fig. 3. Flow Chart of the system

3.1.5 Checking Grammar

Grammar and spelling checks are performed using Language Tool, an open-source programme also used as OpenOffice's spellchecker. This package enables programmers to find grammatical and spelling errors using a Command-line interface or a Python code snippet (CLI).

3.2 Flowchart

As shown in Fig. 3

- Insert the YouTube URL into the summary extension.
- A transcript is generated if the video includes subtitles; otherwise, audio transcription is used to create the transcript.
- The transcript will be summarized using abstractive-based summarization when it is generated.
- Finally, the extension will display the summery.

3.3 Features

The user is given four options after the content has been summarized (buttons).

- Translation: The user has the option of translating their text summary into another language, like Hindi or Marathi
- Speak: The user can hear the condensed transcript read aloud by pressing the speak button.
- Download: The output for downloading the condensed text into several file formats allows the user to obtain their condensed text in a variety of file formats.
- Send to mail: The user can utilize this if they wish to send the transcript file to their email.

4 Conclusion and Future Scope

In this project, the summary of the transcript is done by Abstractive summarization method. It is a technique which does not make use of sentences from the original content to make the summary rather it uses paraphrasing of the original text. Existing video summarization systems require a good hold of technical knowledge. Summarizing videos based on its subtitle is the fastest way of generating summary, because dealing with text is much easier and faster compared to training various videos using machine learning models.

This project may benefit hearing-impaired people who have trouble understanding videos without subtitles or transcripts because it will be accessible to them. For such videos they can understand it with the help of a generated summary. In future work we can manage to change the language of the summarized text to different available languages from the extension. The idea of Transcript Summarizer can be implemented in other streaming services as well in future.

References

1. Lavish Mangal, Dhruv Aggarwal, Gulshan Kumar, Jai Kumar, Meenakshi Ag- garwal, “YouTube Transcript Summarizer.”, 2022.
2. Sanjana R, Sai Gagana V, Vedhavathi K R, Kiran K N, “Video Summarization using NLP.”, 2021.
3. Shraddha Yadav, Arun Kumar Behra, Chandra Shekhar Sahu, Nilmani Chan- drakar, “Summary and Keyword Extraction from YouTube Video Tran- script.”, 2021.
4. S. Tharun, R. Kranthi Kumar, P. Sai Sravanth, G. Srujan Reddy, B. Akshay, “Survey on Abstractive Transcript Summarization of YouTube Videos.”, 2022.
5. Gousiya Begum, N. Musrat Sultana, Dharma Ashritha, “YouTube Transcript Summarizer.”, 2022.
6. Li, Y., Lee, S. H., Yeh, C. H., Kuo, C. C, “Techniques for movie content anal- ysis and skimming.”, 2006.
7. Smaïli, K., Fohr, D., González-Gallardo, C. E., Grega, M., Janowski, L., Jouvet, D., Garcia-Zapirain, B, “A first summarization system of a video in a target language.” 2018.

8. Verma, P., Verma, A, “A review on text summarization techniques.”, 2020.
9. S. Alhojely., J. Kalita, “Recent Progress on Text Summarization.”, 2020.
10. Porwal, K., Srivastava, H., Gupta, R., Pratap Mall, S., Gupta, N, “Video Tran- scription and Summarization using NLP.”, 2022.