



# INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

## YOUTUBE TRANSCRIPT ESSENCE AI

Mrs K.Vanisree<sup>1</sup>, Ms. Konda Sravya Sri Reddy<sup>2</sup>, Ms.Muppidi Ramyakapardini<sup>3</sup>

Assistant Professor , Department Of CSE , UCETW, Kakatiya University

B.Tech Final Year Student, Department Of CSE , UCETW Kakatiya University

B.Tech Final Year Student, Department Of CSE , UCETW Kakatiya University

### ABSTRACT

The AI-powered YouTube Transcript Summarizer provides a new way to consume video content more efficiently through summarized, condensed versions of YouTube videos that are really long with just a simple Chrome Extension. It takes advantage of advanced NLP techniques together with the Hugging Face Transformers library to extract meaningful insights from the transcripts of videos, whereby users will be able to get the vital pieces of information without having to actually watch complete videos. It combines three basic key components: a Text Summarizer summarizing important information, Hugging Face Transformers to ensure coherence and preciseness of summaries, and a Python-based Web API that communicates the extension with the back-end services seamlessly. A comparative study highlights that such a tool serves in enhancing the user experience of decreasing exposure to irrelevant or misleading information and provides substantive improvements to educational It is designed for professional and personal uses of videos. This tool will be going to change the way users consume online video content. The present fast-moving scenario of today's world faces problems of video consumption, which is a time-consuming and inefficient process.

**Index Terms** - Component, formatting ,style, styling ,insert.

### 1.Introduction

The YouTube Transcript Summarizer is a tool powered by AI, which changes how users can view video content by providing the summarized version of lengthy video content. Since digital content doubled, YouTube users upload almost 5 million videos daily, making the quest and concentration of users on the relevant material a difficult task. People often spend hours watching videos, only to realize that they contain little or no valuable information, or they fall victim to clickbait that misleads them. This lack of efficiency has created a very strong desire for a tool with intelligence that can capture the essence of a video shorter and yet provide enough details to inform the user of the main ideas without the need to watch the whole video.

The AI-powered summarizer solves this problem using advanced Natural Language Processing (NLP) techniques, leveraging the Hugging Face Transformers library to generate concise and meaningful summaries from the video transcripts of the YouTube channel. The summarizer gets the main points and the critical information that the user then gets the gist of what is in the video, so she/he can decide whether she/he wants to spend the time to watch the whole thing or not.

The YouTube Transcript Summarizer is Chrome Extension, which connects to the Python-based Web API, that is the server with which video transcript is processed and the summary is retrieved. The summarization is a combination of a feature extraction method and a method of creating a summary known as abstractive summarization. The primary feature is abstractive summarization. Extractive summarization becomes complete sentences that are another part of the text whereas abstractive summarization refashions the summaries and creates a more human-like form of the text, which makes it understandable and helps keep the original content consistent and precise.

This tool is very customizable, allowing users to change the length of the summaries and the level of detail according to their needs. This adaptability thus enables the users to write more focussed summaries on the specific basis of their own wishes and the specific content requirements. The first of which is to help the summation process watching the video, take advantage of it to the maximum extent and at the same time avoid falling into the traps of misinformation and redundant information.

This tool's influence manifests itself especially in the teaching sector where easy access to stable and well-organized knowledge is a must. This summarizer isn't restricted to a certain group of people, so students, professionals, and people who just appreciate knowledge can use it. It's a perfect tool that assists students, professionals and other people to quickly get key points out of educative lectures, tutorials, and industry-related content. Through this, the audience is able to decide whether a clip is worth watching in its entirety, for they will not have to spend the time-consuming process of watching a whole video.

In times of information overload in which knowledge gathering is a struggle due to video content, the YouTube Transcript Summarizer is for sure the AI-driven approach of the century. It will make the customers more satisfied with the services offered, save time and make the whole process of information consumption more effective and logical—hence it is the most powerful digital media engagement tool in the entire market.

## **2.REVIEW OF RELATED LITERATURE**

YouTube, the technology platform which is acknowledged for being the most popular for various online videos, has an incredible amount of content. The fact that so much content is available to its users that it could be difficult to locate the required information is because of the enormous Content uploaded. YouTube transcript summarization, a subfield of Natural Language Processing (NLP), addresses this issue by providing concise summaries of video transcripts. Through this information, the reader is able to comprehend the details and concepts of the related topic. Accordingly, these easy-to-understand summaries beep the same information and help users quickly distinguish whether the video is very useful or not. The present study provides further evidence that this task is well suited to readers and that it is

ready for wider applications. Hence the user experience on the one hand is improved, and on the other, time is saved because valuable insights are delivered instantly.

### Key Research Studies

"Video Summarization Using Fully Convolutional Sequence Networks" (2018)Rochan, Ye, and Wang came up with the idea to carry out their work using fully convolutional sequence networks for video summarization. Their work is actually based on the ability to process large and unstructured data which lead them to the idea that video summarization tasks could be efficiently done by the NLP model.

"Learning to Summarize Videos with Attention" (2019)Fajtl and his colleagues' innovative idea was to draw on attention mechanisms for the video summarization process. They categorized it as a method that dives deep into the context of the original text and inevitably the relevance between them. The study reports that using this procedure, a tone-deaf model is prone, even if results may be more accurate, an outcome still quite far from what a person would grasp.

"Deep Reinforcement Learning for Unsupervised Video Summarization" (2018)The paper by Zhou, Qiao, and Xiang names the deep reinforcement learning-based unsupervised summarization approach as one of the trusted methods. They argue that instructing a model to do this simple task aids in the scaling and automatization of the process, thus the need for annotated data is removed.

"Two-Stage Hierarchical Video Summary Extraction" (2003)Ferman and Tekalp designed a two-stage hierarchical video summation system, which was the first of its kind in the project. Consequently, users will enjoy personalized solutions and user satisfaction with video summary experiences, as it is demonstrated in this case on the subsequent slide.

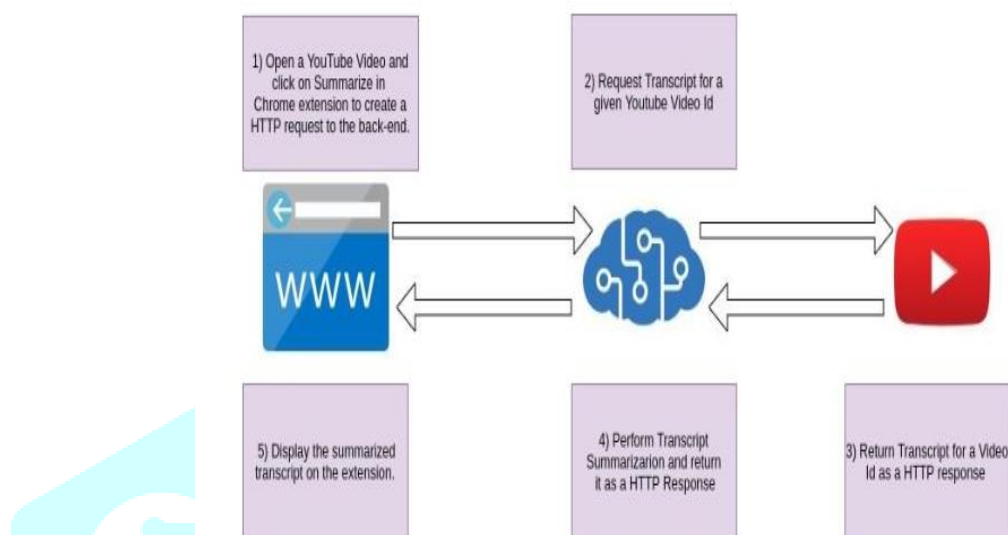
In 2021, "Natural Language Processing (NLP) based Text Summarization - A Survey" was published by Ishitva Awasthi, Kuntal Gupta, Prajbot Singh Bhojal, Anand, Piyush kumar. The techniques used are Extractive and abstract methods for summarizing texts. The advantages are: Based on linguistic and statistical characteristics, the implications of sentences are calculated. The disadvantage is each type of summarization technique is useful in different situations. One cannot say which technique is more promising .

"Review of automatic text summarization techniques & methods" is developed by Adhika Pramita, Supriadi Rustad, Abdul Shukur, Affandy. It was published in 2020. They have used Text summarization, Systematic review techniques. The drawback is that the Fuzzy based approach is weak in semantic problems. Many loopholes have to be improved in extractive methods. "Study on Abstractive Text Summarization Techniques" is developed by Parth Rajesh Dedhia, Hardik Pradeep, Meghana Naik. It was published in 2020. They have used Seq2Seq, Encoder-Decoder, and Pointer Mechanism. The drawback is that the current model does not work if multiple documents are passed to the model.

"Abstractive Summarization of video sequences" is developed by Aniqadilawari, Muhammad Usman Ghani Khan. They have used multi-line video description, RCNN deep neural network model. The drawback is It focuses only on the conciseness of the summary. Memory efficiency and time constraints are not under consideration.

### 3. TECHNICAL ARCHITECTURE AND WORKFLOW

The AI YouTube Transcript Summarizer's diagram of system architecture represents the harmonization of PHP backend with a Python Flask application and a Google Chrome extension. This step-by-step guide illustrates the functional zones' interaction that provides for the content summarization from YouTube video transcripts, and at the same time, it supplements the satisfaction of user needs as well as information storage.



Dig-1. Project Stages

#### Framework

##### **Input:**

*YouTube Video ID - It is a code consisting of letters and numbers specific to each YouTube video.*

##### **Step 1: Retrieve Your Transcript**

*Library Used: youtube\_transcript\_api (Python).*

##### **Process:**

*Fetch the subtitles or transcripts for the given YouTube video id.*

*Transcripts are categorized as :*

- *Manually generated*
- *Automatically generated,*
- *Videos without transcripts (excluded from the process).*

##### **Step 2: Restore Punctuation**

• *Library Used: punctuator (Python).*

• *Purpose: Correct the absence of punctuation signs (such as periods and commas) that brings the clarity and coherence back to a text.*

##### **Step 3: Analyzing Text**

*Steps Involved:*

- *Tokenization: Taking text and breaking it up into its constituent sentences and words that are capitalized or are added to the list of vocabulary.*
- *Lowercasing: For the sake of consistency, the text is changed into the lowercase.*
- *Stop Word Removal: Notices the appearance of quite common words such as “a” or “the” and disregards those in the text .*
- *Punctuation Removal: No dots, commas, or exclamations are attacked by the algorithm.*
- *Stemming/Lemmatization: Use words in their root form (e.g., “running” to “run”).*

#### **Step 4: Text Summarization**

*Technique: Text Rank model.*

*Algorithm:*

1. *TF-IDF and Text Rank model is a text summarization technique to create a summary of a given text using keywords which match the meaning of the document.*
2. *Term Frequency (TF): Defined as the ratio of the number of times a term is shown among other terms in a given sentence to the total number of words in that sentence.*
3. *Inverse Document Frequency (IDF): A logarithmic and statistical function of frequency of words in a given sentence.*
4. *Vectorization: A mathematical technique that employs factorization methods to compute low-dimensional data representations using vector and matrix operations.*

#### **Step 5: Generate Summary**

*Process:*

*Rank sentences or phrases on the basis of their TF-IDF scores.*

*Select the highest items ensuring the summary is faithful to the context and main idea of the original transcription content.*

## **Components**

### **1. Chrome Extension**

- **Manifest File (manifest.json):** Gives permission to the extension and defines its icons and settings.
- **Popup HTML (popup.html):** Its objective is to let the user stimulate the summarization, as well as to show the results through it.
- **Popup JavaScript (popup.js):** The script deals with user interactions, communication with the backend, and updating the UI.

### **2. Backend Flask Application**

**Core Application (app.py):**

This is where the API handles requests.

It also controls the successful retrieval and summarization of the video transcript.

Endpoints:

summary: A tool that functions efficiently only with YouTube links, it extracts video IDs, downloads transcripts, and forms summaries automatically.

## Workflow Diagram

### User Interaction

**Action:** The user needs to click the extension icon.

**Popup Interface:** Before proceeding, the “Summarize” button must be clicked and then output done in the text area.

### Request Initiation

Action: User clicks the “Summarize” button.

Process: popup.js requests the backend by using the GET method and becomes the message listener.

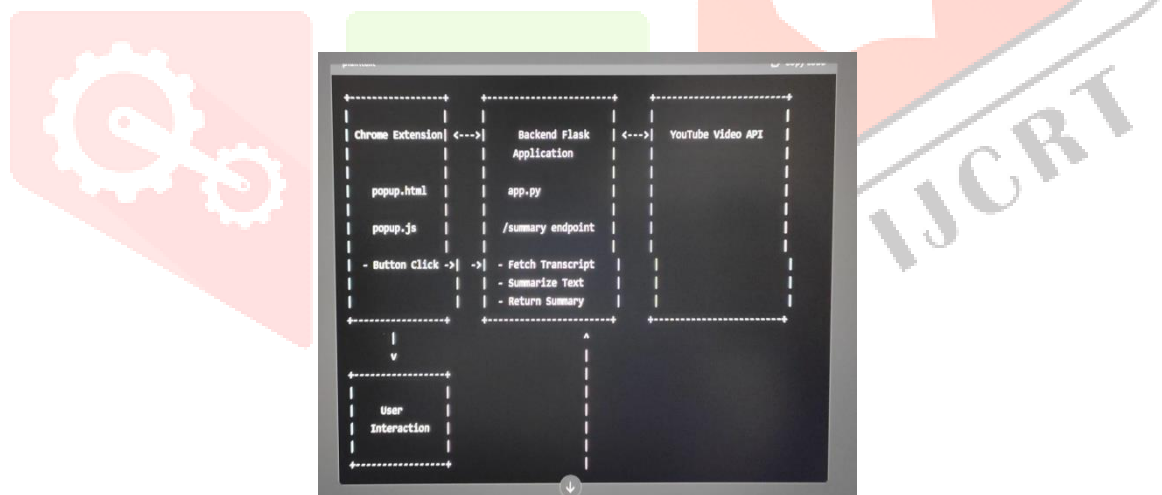
## 3. Backend Processing

**Endpoint:** /createSummary in the Flask framework gets the request.

Extracts the video ID from the URL.

Calls `get_transcript(video_id)` to get the transcript of the video.

Calls `get_summary(transcript)` which performs the summarizing task using Hugging Face Transformers.

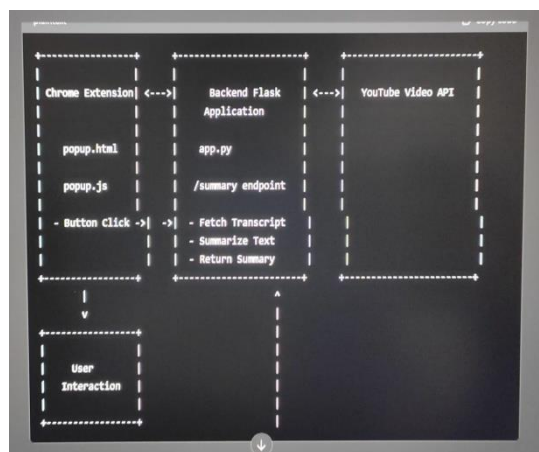


## 4. Transcript Retrieval

**Function:** `get_transcript(video_id)`

Retrieves video transcript data and the video ID to the video ID.

Joins the transcript segments into one understandable item.



## 5. Text Summarization

**Function:** `get_summary(transcript)`

Splits the transcripts and processes them by using summarization pipelines of the transformer to generate brief results.

Segments the transcripts and prepares them for summaries by using summarization pipelines.

Summarization pipelines are utilized to make more brief versions of the text.

## 6. Response Handling

**Backend:** The summary message is returned as an HTTP response to the extension.

**Frontend:**

- `popup.js` updates the UI with the summary.
- Re-enables the "Summarize" button for reuse.

### Component Interaction

#### 1.Chrome Extension to Backend

The `popup.js` is used to GET the backend API with YouTube video URLs through the extension.

#### 2.Backend Processing

Flask is in charge of the transcript getting and summarization by using:

YouTubeTranscriptApi for text extraction.

Hugging Face Transformers for abstractive summarization.

#### 3.Backend to Chrome Extension

Summarized text is sent to `popup.js` that in turn, user interface is updated.

### Schematic Diagram:

The schematic diagram is the graphical representation of the communication between the Chrome extension and the backend Flask application, with specific emphasis on the request flow, processing steps, and response handling.

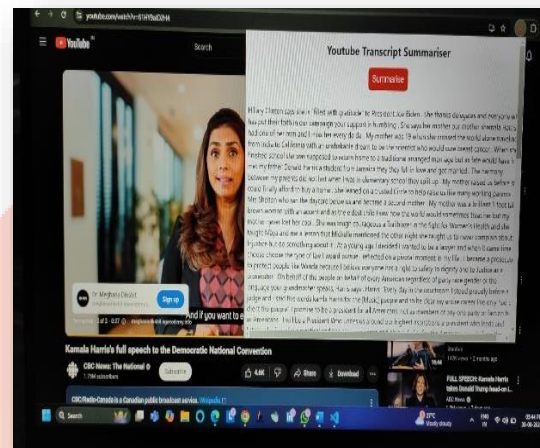
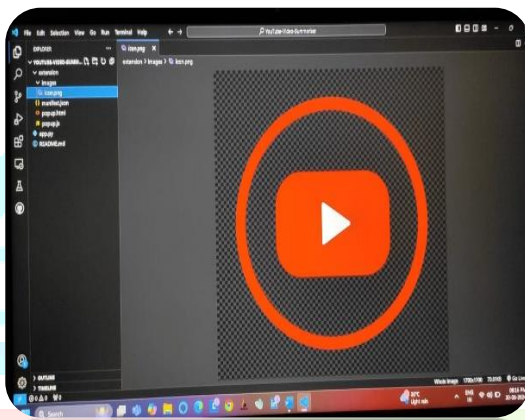
## Directory Structure :

YOUTUBE-VIDEO-SUMMARISER/

```

|
|— extension/
| |— images/
| | |— icon.png
| |— manifest.json
| |— popup.html
| |— popup.js
|— app.py
|— README.md
  
```

## 4. RESULT SCREEN



## 5. CONCLUSION

The AI YouTube Transcript Summarizer Chrome Extension short for Artificial Intelligence YouTube Transcript Summarizer Chrome Extension aims to achieve concise and accurate video content consumption by offering direct and efficient summaries on the browser while. Moreover, by integrating the Chrome browser with natural language processing and creating a link between the tool and the browser, accessibility and usability are affected lots of users are able to use long video content that is informative and enjoyable to help them.

## 6. FUTURE ENHANCEMENTS

Exploitation of the Non-English Video Support: Make integrations of translation systems as a means to allowing English summaries for non-English videos, thereby succeeding for the videos to be seen and well communicated around the world.

Custom summary Lengths: Give users the ability to select the length of the summary that is needed to make the information more personal.

Summarizing Without Subtitles: Apply a voice-to-text live translation technology to summarize videos without subtitles and also, include any content that cannot be summarized otherwise.

These developments are expected to make the summarizer more adaptable, user-friendly and therefore, more efficient for the digital video content management sector.

## REFERENCES

- [1] Lavish Mangal, Dhruv Aggarwal, Gulshan Kumar, Jai Kumar, Meenakshi Aggarwal, “YouTube Transcript Summarizer.”, 2022.
- [2] Sanjana R, Sai Gagana V, Vedhavathi K R, Kiran K N, “Video Summarization using NLP.”, 2021. Shradha Yadav, Arun Kumar Behra, Chandra Shekhar Sahu, NilmaniChandrakar, “Summary and Keyword Extraction from YouTube Video Transcript.”, 2021.
- [3] S. Tharun, R. Kranthi Kumar, P. Sai Sravanth, G. Srujan Reddy, B. Akshay, “Survey on Abstractive Transcript Summarization of YouTube Videos.”, 2022.
- [4] Gousiya Begum, N. Musrat Sultana, Dharma Ashritha, “YouTube Transcript Summarizer.”, 2022.
- Li, Y., Lee, S. H., Yeh, C. H., Kuo, C. C, “Techniques for movie content analysis and skimming.”, 2006.
- [5] Smaïli, K., Fohr, D., González-Gallardo, C. E., Grega, M., Janowski, L., Jouvét, D., Garcia Zapirain, B, “A first summarization system of a video in a target language.”, 2018.
- [6] Verma, P., Verma, A, “A review on text summarization techniques.”, 2020.
- S. Alhojely, J. Kalita, “Recent Progress on Text Summarization.”, 2020.
- [7] Porwal, K., Srivastava, H., Gupta, R., Pratap Mall, S., Gupta, N, “Video Transcription and Summarization using NLP.”, 2022.
- [8] P. Sushma, Dr. S. Nagaprasad, Dr. V. Ajantha Devi, “**Youtube: Bigdata Analytics using Hadoop and Map Reduce.**”, *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)*, Vol 5, Issue 4, April 2018.

## Online Resources:

1. [Crio Projects - Python YouTube Transcript](#)
2. [PyPI: youtube-transcript-api](#)

## [The Python Code: Text Summarization using Hugging Face Transformers](#)