YOUTUBE TRANSCRIPT SUMMARIZATION USING TEXTRANK ALGORITHM

A PROJECT REPORT

Submitted by

KESAVAN PR - 312319104072

KIRUBA SHANKAR AP - 312319104075

in partial fulfilment for the award of the Degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

St. JOSEPH'S COLLEGE OF ENGINEERING

(An Autonomous Institution)

St. Joseph's Group of Institutions

Jeppiaar Educational Trust

OMR, Chennai 600 119

ANNA UNIVERSITY:: CHENNAI 600 025

JUNE 2022

ANNA UNIVERSITY

BONAFIDE CERTIFICATE

Certified that this project report on **YOUTUBE TRANSCRIPT SUMMARIZATION USING TEXTRANK ALGORITHM** *is the bonafide work of* ***KESAVAN PR (312319104072)*** *and* ***KIRUBA SHANKAR AP(312319104075)*** *who carried out the project under my supervision during the academic year 2021 - 2022.*

Signature of the
Head of the Department

Dr. R. Pugalendhi M.E., Ph. D HOD – Lab Affairs
Department of Computer Science and Engineering
St. Joseph's College of Engineering
OMR, Chennai- 600119.

Signature of the Supervisor

Dr. J. Ramya M.E., Ph. D
SUPERVISOR
Associate Professor
Department of Computer Science and Engineering
St. Joseph's College of Engineering
OMR, Chennai- 600119

CERTIFICATE OF EVALUATION

COLLEGE NAME **:** St. Joseph's College of Engineering,

BRANCH **:** B.E. - Computer Science and Engineering

SEMESTER **:** VI

S. No	**Name of the Student**	**Title of The Project**	**Name of the Supervisor with Designation**
1.	KESAVAN.PR (312319104072)	YOUTUBE TRANSCRIPT SUMMARIZATION USING TEXTRANK ALGORITHM	**Dr.J.Ramya, M.E.,Ph.D** Associate Professor
2.	KIRUBA SHANKAR AP. (312319104075)		

The report of the project work submitted by the above students in partial fulfilment for the award of the Degree of Bachelor of Engineering in Computer Science and Engineering at Anna University is confirmed to be a report of the work done by the above students and then evaluated.

Submitted to Project and Viva Examination held on ____________________.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

At the outset, we would like to express our sincere gratitude to the **ALMIGHTY** and our beloved ***Chairman, Dr. B. Babu Manoharan M.A., M.B.A., Ph.D St. Joseph's Group of Institutions*** for his constant guidance and support to the student community and the Society.

We would like to express our hearty thanks to our respected ***Managing Director, Mrs. S. Jessie Priya M.Com. St. Joseph's Group of Institutions*** for her kind encouragement and blessings. We wish to express our sincere thanks to our ***Executive Director Mr. B. Shashi Sekar, M.Sc. St. Joseph's Group of Institutions*** for providing ample facilities in the institution.

We would like to express sincere gratitude to our beloved and highly respected ***Principal Dr. Vaddi Seshagiri Rao M.E., M.B.A., Ph.D., F.I.E.*** for his inspirational ideas during the course of the project. We would like to express sincere gratitude and our utmost respect to our beloved ***Dr. B. Parvathavarthini M.E., M.B.A., Ph.D., Dean (Research)*** for her inspirational ideas during the course of the project.

I also express my sincere thanks and most heartfelt sense of gratitude to **Dr. A.Chandrasekar, M.E., Ph.D., Head of the Department of Computer Science and Engineering** for his dedication, commendable support and encouragement for the completion of project work with perfection

We would like to acknowledge our gratitude to our supervisor **Dr. J.Ramya M.E., Ph.D** for her excellent guidance and connoisseur's suggestion throughout the study carried out successfully.

Finally, we thank the **Faculty Members** and **our Family**, who helped and encouraged us constantly to complete the project successfully.

TABLE OF CONTENTS

ABSTRACT

Enormous No. of video recordings are being created and shared on the Internet through out the day. It has become really difficult to spend time in watching such videos which may have a longer duration than expected and sometimes our efforts may become futile if we couldn’t find relevant information out of it. Summarizing transcripts of such videos automatically allows us to quickly look out for the important patterns in the video and helps us to save time and efforts to go through the whole content of the video . In this project ,We will be a creating an Application which will make a request transcript for a video where it will perform NLP and respond with a summarized version of a YouTube transcript which we are going to do is extracting the summaries from a video transcripts and also generating important keywords from it which will use Natural Language Processing (NLP) methods for extractive and abstractive summarization.

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACRONYM	DESCRIPTION
NLP	Natural Language Processing
GUI	Graphical User Interface
PFM	Position Frequence Matrix
LSA	Latent Semantic Analysis
TF	Term Frequency
IDF	Inverse Document Frequency
API	Application Programming Interface

CHAPTER 1

1.INTRODUCTION

Natural Language Processing (NLP) is a field of Artificial Intelligence that focuses mainly on the study of the interaction between human languages and machines. Generating summaries of video transcripts is the process of generating short, fluent, and most importantly accurate summaries of longer videos. The main idea behind it is to be able to find a short subset of the most essential information from the entire set and present it in a human-readable format. As online textual data grows, automatic summarization of text methods has the potential to be very helpful because more useful information can be read in a short time. Generating summaries of transcripts is in the field of NLP because machines are required to understand what humans have written and produce human-readable outputs. Nowadays, it has become very easy to watch videos on Youtube for anything, be it educational or entertainment, there are lots of videos of those kinds of genres and because of the No., it is very important for us to find out the exact content that we want to consume. And sometimes because of network issues, we aren't able to watch text-based YouTube videos in high resolution.

since it makes the text blurry. There are also those long and click bait videos that are only made to earn profit that contain not a single bit of useful information which wastes our time. So, by removing the useless part of the videos, skipping ads, and getting summaries, we can directly jump to the information that interests us and save lots of time and effort. The main objective of our project is to save the time of a user and increase work efficiency. There are countless situations when a user lures into the catchy title and thumbnail of the video and wastes all its time in consuming useless information.

Students browse youtube videos before their exams so that they can get the most of the information in less time because of that they usually watch videos at double speed and because of this, it doubles the confusion about totally new topics. Sometimes people want to know what is happened in a long meeting in short, if the meeting is recorded and has a transcript we can find out its summary which will save lots of time for the people of the organization. This is where our project is pointing out, it will create a summary of the transcript or the captions of whatever video you are watching. The most important part of our project is to collect the most necessary information and concentrate it into a small paragraph. Getting all the necessary information about the topic which interests you in the form of that small paragraph will save lots of time and effort for other things, which was wasted when you were lured into click bait videos and wasted your whole half or one hour or more.

1.1 INTRODUCTION TO YOUTUBE TRANSCRIPT SUMMARIZATION

On YouTube, lots of videos are uploaded every single day around the world which are very long, mostly educational Related Videos. And if We want to find some useful information from those videos, it is an almost impossible task to achieve since most of the videos contain so much useless content and less useful content, because of this you have to watch the whole video which probably wastes Our lots of time and efforts or watch multiple videos to extract that useful information,

So generating the summaries of those video transcripts will save Our lots of time and We can quickly gather more useful and important information from it, which will surely save our efforts and time to watch the whole video. And Even For Meetings and video-conferencing A system that could turn voice to text and then generate summaries from your team meetings will save our time and This will be very useful . Mostly a summarizer is used to extract the most salient claims across patents .

For this project we are going to use python API for scrapping transcript or subtitles for a Youtube video and then after obtaining the transcripts we will perform text summarization using HuggingFace Transformers and build a flask backend REST API to expose the summarization service to the client amd finally display summarized text to the user .

1.2 INTRODUCTION TO TRANSCRIPTION

Transcription assists the conversion of human speech into a text transcript. Audio or video files can be transcribed manually or automatically.Transcriptionists can replay a recording several times in a transcription editor and type what they hear. By using transcription hot keys, the manual transcription can be accelerated, the sound filtered, equalized or have the tempo adjusted when the clarity is not great.

With speech recognition technology, transcriptionists can automatically convert recordings to text transcripts by opening recordings in a PC and uploading them to a cloud for automatic transcription, or transcribe recordings in real-time by using digital dictation. Depending on quality of recordings, machine generated transcripts may still need to be manually verified. The accuracy rate of the automatic transcription depends on several factors such as background noises, speakers' distance to the microphone, and accents.

Transcription software, as with transcription services, is often provided for business, legal, or medical purposes. Compared with audio content, a text transcript is searchable, takes up less computer memory, and can be used as an alternate method of communication, such as for closed captions.

1.3 INTRODUCTION TO SUMMARIZATION

summarization is the process of shortening a set of data computationally, to create a subset (a summary) that represents the most important or relevant information within the original content. In addition to text, images and videos can also be summarized. Text summarization finds the most informative sentences in a document .

various methods of image summarization are the subject of ongoing research, with some looking to display the most representative images from a given collection or generating a video summarization extracts the most important frames from the video content.

CHAPTER 2

SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

Most of the existing algorithms for the Youtube Transcript Summarization with the several key parameters that need to be set correctly to achieve the best classification results for the problem. For some problem the result may provide good Summarization and accuracy and for some it may results in poor accuracy due to algorithm selection and quality.

2.1.1 Summary Algorithm

Summarization is the task of condensing a piece of text to a shorter version, reducing the size of the initial text while at the same time preserving key informational elements and the meaning of content. Since manual text summarization is a time expensive and generally laborious task, the automatization of the task is gaining increasing popularity and therefore constitutes a strong motivation for academic research**.**

2.1.2 Distance Metrics Algorithm Based on a Position Frequency Matrix

Regulatory sequence detection is a fundamental challenge in computational biology. The transcription process in protein synthesis starts with the binding of the transcription factor (TF) to its binding site. These binding sites are short DNA segments that are called motifs. Different sites can bind to the same factor. This variability in binding sequences besides their low information content and low specificity increases the difficulty of their detection using computational algorithms. This paper proposes a novel algorithm for transcription factor binding sites (TFBSs) detection in the entire genomic structure and allow discovery of new motif sequences. This is achieved by using distance metrics based on a position frequency matrix (PFM) concept that quantify the similitude between the set of conserved sequences belonging to a particular TF and the entire DNA sequence under study. Hence, the PFM in this context can be thought of as a consensus sequence as it provides a representative measure of the said set of binding sites belonging to a particular TF. The algorithm then quantifies the correlation between the PFM and each binding site belonging to a given TF. Same scenario is then applied to the genome sequence under study.

2.1.3 Latent Semantic Analysis — Deduce the hidden topic from the document

Latent Semantic Analysis is an efficient way of analysing the text and finding the hidden topics by understanding the context of the text.Latent Semantic Analysis(LSA) is used to find the hidden topics represented by the document or text. This hidden topics then are used for clustering the similar documents together. LSA is an unsupervised algorithm and hence we don't know the actual topic of the document.Most simple way of finding similar documents is by using vector representation of text and cosine similarity. Vector representation represents each document in the form of vector. This vector is known as document-term matrix.This matrix is of dimension (vocabulary size) * (vocabulary size). It represents the frequency of the words coming together in the dataset. The matrix helps us understand the words which belongs together.

2.1.4 DISADVANTAGES OF ALGORITHMS

1. In the Above Methods and Algorithms more samples will improve the prediction level but fail to improve the accuracy, Accuracy is lesser than we Thought.
2. Transcription Takes To long to Process in These Methods due to their package and dataset .
3. Sometime Initializing Package Can not be done and Complete Code Can be Correpted Due to Package Error.

2.2 LITERATURE SURVEY

In the paper " **Summary and Keyword Extraction From Youtube Video Transcript** " By Shraddha Yadav , Arun Kumar Behra , Chandra Shekhar Sahu, Nilmani Chandrakar From 1Faculty , Department of Computer Science and Engineering , Government Engineering College , Raipur , Chhattisgarh , India .The Paper Reports, For the purpose of extractive summarization, we have used the TF-IDF model with Text Rank Algorithm.TF-IDF(Term Frequency - Inverse Document Frequency) .After the cleaning process, we have to convert the words into it's vectorized form so that our algorithm will process it by using TF-IDF.This is a technique to measure the quantity of a word in documents, we compute a weight to each word whichsignifies the importance of the word in the document and corpus.

TF(Term Frequency): TF calculates the frequency of a word in a document.

TF = No. of repetition of the word in the sentence / No. of words in a sentence

IDF(Inverse Document Frequency): IDF is the inverse of the document frequency which measures the informativeness of term t.

IDF = log(No. of sentences / No. of sentences containing words)

After this, we will multiply both matrices to obtain the vectorized form which tells us which words are the most important.

In The Paper "**Video Summarization using NLP**" By Sanjana , Sai Gagana , Vedhavathi K , Kiran K of Students, Dept. of EC Engineering, BNMIT, Karnataka, India Reports that, Latent Semantic Analysis LSA is an unsupervised approach technique in Natural Language Processing. It is anAlgebraicStatistical method which extracts the features of the sentences that cannot be directly mentioned.These features are essential to data, but are not original features of the dataset.Working of LSA is the Term Co-occurrence and This matrix is of dimension (vocabulary size) * (vocabulary size). It represents the frequency of the words coming together in the dataset. The matrix helps to understand the words which belong together. Similarity between two different summaries is found out using cosine similarity between the summary matrix.

In The Paper **"Survey on Abstractive Transcript Summarization of**

YouTube Videos" By S. Tharun, R. Kranthi Kumar, P. Sai Sravanth, G. Srujan Reddy, B.Akshay Assistant Professor of Department of Computer Science and Engineering

Reports that, They used Latent Dirichlet Allocation (LDA) in Paper [1], which has been shown to be effective in summarization of documents. The suggested LDA summarizing model is divided into three stages. The first phase prepares the subtitle file for modelling by deleting stop words and doing other pre-processing tasks. The subtitles are used to train the LDA model in the second step in order to generate the list of keywords that will be employed to extract relevant sentences. The summary is prepared based on the list of keywords generated in the third phase. The quality of LDA-based generated summaries beats that of TF-IDF and LSA summaries.

In The Paper **"A hybrid end-to-end model ASoVS Using Neural Nettworks"** By Yadav , Arun Kumar Behra , Chandra Shekhar Sahu, Nilmani Chandrakar From 1Faculty , Department of Computer Science and Engineering , Government Engineering College , Raipur , Chhattisgarh , India Reports that, A hybrid end-to-end model, ASoVS, was recommended in the study [12], which uses a deep neural network for generating description and text summary in abstractive method for any video which is given as input. They created a framework which draws people & their traits like gender, age, emotion, etc. as well as scenes, objects, and behaviors for providing a multiple-line video description. They utilized an attention model for abstractive summarization which outperformed baseline techniques and got better results.

They used Latent Dirichlet Allocation (LDA) in Paper [1], which has been shown to be effective in summarization of documents. The suggested LDA summarizing model is divided into three stages. The first phase prepares the subtitle file for modelling by deleting stop words and doing other pre-processing tasks. The subtitles are used to train the LDA model in the second step in order to generate the list of keywords that will be employed to extract relevant sentences. The summary is prepared based on the list of keywords generated in the third phase. The quality of LDA-based generated summaries beats That of TF-IDF and LSA summaries.

Stream Hover which is a platform for explaining and summarizing transcripts of live streamed videos was presented in the paper [2]. They looked at a neural extractive summarization model that learns vector representations of audio file and extracts significant observations from subtitles to construct summaries using a vector quantized autoencoder.

The paper [8] offers a system that generates subtitles for movies in either of three languages: English, Hindi, or Malayalam, depending on the user's preference. Audio extraction, voice recognition, and subtitle generation are the three components in the model. Using the FFMPEG platform, audio extraction converts an entire file of any structure to.wav (Waveform Audio) format. The extracted.wav file is then used to create subtitles in the form of a .srt file, which contains the phrases (lyrics) spoken in the audio file. The Google Translate API is used to achieve speech attention. The srt file is then included in the video in the Subtitle creation module, where the lyrics are synchronized with the time and presented alongside the video. The Moviepy video enhancement library in Python is used to accomplish this.

The paper [9] presents a system for generating abstractive summaries of videos on various topics including cooking, cuisine, software configuration, sports, etc. They used Transfer learning & pre-trained the model on large datasets in English to extend the vocabulary. They also did transcript pre-processing to get better sentence formation and punctuation in ASR systems results. For the How2 and WikiHow datasets, ROUGE and Content-F1 scoring are used to assess the results. Different Text Summarization Methods are classified in the paper [6].

Abstractive Text Summarization is given more weight in the paper. The authors feel that, although being more difficult and computationally intensive than extractive summarizing, abstractive summarization holds more promise in terms of producing more natural and human-like summaries. As a result, we might anticipate further approaches in this subject that provide new viewpoints from computational, cognitive, and linguistic perspectives.

A hybrid end-to-end model, ASoVS, was recommended in the study [12], which uses a deep neural network for generating description and text summary in abstractive method for any video which is given as input. They created a framework which draws people & their traits like gender, age, emotion, etc. as well as scenes, objects, and behaviors for providing a multiple-line video description. They utilized an attention model for abstractive summarization which outperformed baseline techniques and got better results.

The paper [5] discusses methods for converting speech audio files to text files as well as text summarization on the text files. They used Python libraries to convert the audio files to text format in the first scenario. Natural Language Processing modules are utilized for text summarization in the latter situation. The English data functions are implemented using the spaCy Python library. The important sentence obtained when the extraction is studied is used in the summarization approach. Words are allocated weights based on the No. of times they appear in the text file. This method is used to create summaries from the original audio recording.

The paper [11] provides a system in which textual explanations are summarized using extractive methods, with the best results coming from the LSA, LexRank, and SumBasic approaches. The results of human reviews of video summaries were favourable.A conditional recurrent neural network (RNN) was presented in the study [3], which provides the summary of any text which is given as input. The governing is delivered by a unique convolutional encoder, which ensures that decoder targets on correct words during every phase of the generating process. The model is straight forward to train end-to-end on big data sets and depends solely on learned features. The model exceeds the state-of-the-art technique on Gigaword Dataset while competing on the DUC-2004 shared task, according to their experiments.

The paper [10] presents a method in which the application recognizes voice in one language and converts it to a userdefined language for expressive communication. It has four modules: voice recognition, translation, speech synthesis, and visual translation, with audio output of the translated language. In addition, the application receives typed content and translates it into the required language. They employed OCR technology to convert images into text. The abbreviation of OCR is Optical character recognition. It is a technique for extracting text from photographs such as handwritten signs and billboards.

The strategies for generating subtitles are described in the work [7]. Automatically with extensive descriptions of three modules: Audio Extraction, which transforms an MPEG-compliant input file to.wav format, and Speech Recognition, which recognizes extracted speech. The HMM (Hidden Markov Model) determines the likelihood of occurrence of words using the Language model and the Acoustic model. Subtitle generation, which produces a.txt/.srt file that is synchronized with the input file. It is extremely beneficial to persons who are deaf, have reading and literacy difficulties, and are learning to read.

A multilingual speech-to-text conversion method is presented in the work [4]. The information in the voice signal is used to convert. For humans, the most natural and important mode of communication is speech. A human voice utterance is input into a Speech-To-Text (STT) system, which needs a string of words as output. The goal of the system is extracting, characterizing, and recognising speech-related content. For voice classification, the suggested system uses Mel-Frequency Cepstral Coefficient feature extraction methodology, as well as the Minimum Distance Classifier and Support Vector Machine techniques.

ROUGE Metrics for evaluating Text Summaries are provided in article [13]. The abbreviation of ROUGE is RecallOriented Understudy for Gisting Evaluation. It contains measures for deciding the quality of summary by comparing it to other summaries generated by humans. Between the summary generated by the computer to be evaluated and the ideal summaries written by humans, the measurements count the No. of overlapping units like n-grams, word pairs, and word sequences.

The systems proposed in the papers [1], [11] don’t work for videos which do not have readily available subtitles. The work in articles [2], [9] is restricted only to English videos. There is absence of Media player in the work presented in paper [8], the entire video must be uploaded in their system in order to generate the subtitles. The system proposed in the work [12] generates video description based on the content present in the video and doesn’t include its audio, this system bestsuits for generating descriptions for CCTV footages. The work presented in paper [5] is applicable only to audio files and is not compatible with video files. The system proposed in article [3] only works for text input, it cannot neither extract subtitles from videos nor generate subtitles for videos. There is only translation and no summarization of text in the work presented in the paper [10], the text which is generated after speech recognition is translated into the target language as output. The work presented in the paper [7] is limited only to extraction of subtitles from the videos .

2.3 MOTIVATION

The No. of YouTube users in 2020 was approximately 2.3 billion, and has been increasing every year. Every minute,300 hours of YouTube videos are uploaded. Almost one-third of the YouTube viewers in India access videos on their mobiles and spend over 48 hours a month on the website, a Google study said. It is frustrating and time consuming to search for the videos that contains the information we are actually looking for. For instance, there are many Ted Talk videos available online in which the speaker talks for a long time on a given topic, but it is hard to find the content the speaker is mainly focusing on unless we watch the entire video.Many machine learning based video summarization techniques are present but they require devices with large processing powers, this is because each video contains thousands of frames and processing all frames takes a very long time. In this paper we propose to use the LSA Natural Language Processing algorithm, which requires less processing power and no training data required to train the algorithm This is Our Motivation to Do This YouTube Transcript Summarization More Effective and Efficient .

2.4 PROPOSED SYSTEM

we need to get the subtitles or transcript for a given Youtube video id by using the python API known as youtube_transcript_api. Since there are three types of transcript that we can extract - manually generated transcript, automatically generated transcript, and the videos that contain no transcript. We are not considering videos that do not have transcripts. Secondly, when we get the transcript of a given Youtube video since it does not contain any punctuations like comma(,), full stops(.) which is very important for us in finding the boundaries of a sentence, so we will restore punctuations from our extracted transcript by using the python library known as “punctuator”. Now we will apply the text preprocessing methods to clean the extracted transcript by tokenizing the sentences as well as the words, lowercasing it, removing stop words like a, an, the, etc, removing punctuations, and stemming or lemmatization to generate the root form of inflected words.

Performing text summarization: This task consists of shortening a large form of text into a precise summary that keeps all the necessary information intact and preserves the overall meaning. For this purpose in NLP for text summarization, there are two types of methods used: **Extractive Summarization:** In this type of text summarization, the output is only the important phrases and sentences that the model identifies from the original text. For the purpose of extractive summarization, we have used the TF-IDF model with Text Rank Algorithm.

TF-IDF(Term Frequency - Inverse Document Frequency) After the cleaning process, we have to convert the words into it's vectorized form so that our algorithm will process it by using **TF-IDF**. This is a technique to measure the quantity of a word in documents, we compute a weight to each word which signifies the importance of the word in the document and corpus. **TF(Term Frequency):** TF calculates the frequency of a word in a document.

TF = No. of repetition of the word in the sentence / No. of words in a sentence

IDF(Inverse Document Frequency): IDF is the inverse of the document frequency which measures the informativeness of term t.

IDF = log(No. of sentences / No. of sentences containing words)

After this, we will multiply both matrices to obtain the vectorized form which tells us which words are the most important.

2.4.1 Text Rank Algorithm

It is based on the PageRank algorithm which calculates the rank of web pages which is used by search engines such as Google. Using the concept of this we will rank the most important sentences in the text and generate a summary. For the task of automated summarization, TextRank models any document as a graph using sentences as nodes. A function is computing the similarity of sentences to build edges in between. This function is used to weight the graph edges, the higher the similarity between the sentences the more important the edge between them will be in the graph. TextRank determines the relation of similarity between two sentences based on the content that both share.

This overlap is calculated simply as the No. of common lexical tokens between them, divided by the length of each to avoid promoting long sentences. Cosine similarity of two words A & B is computed using following formula:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

2.4.2 FLOW DIAGRRAM OF SUMMARIZATION

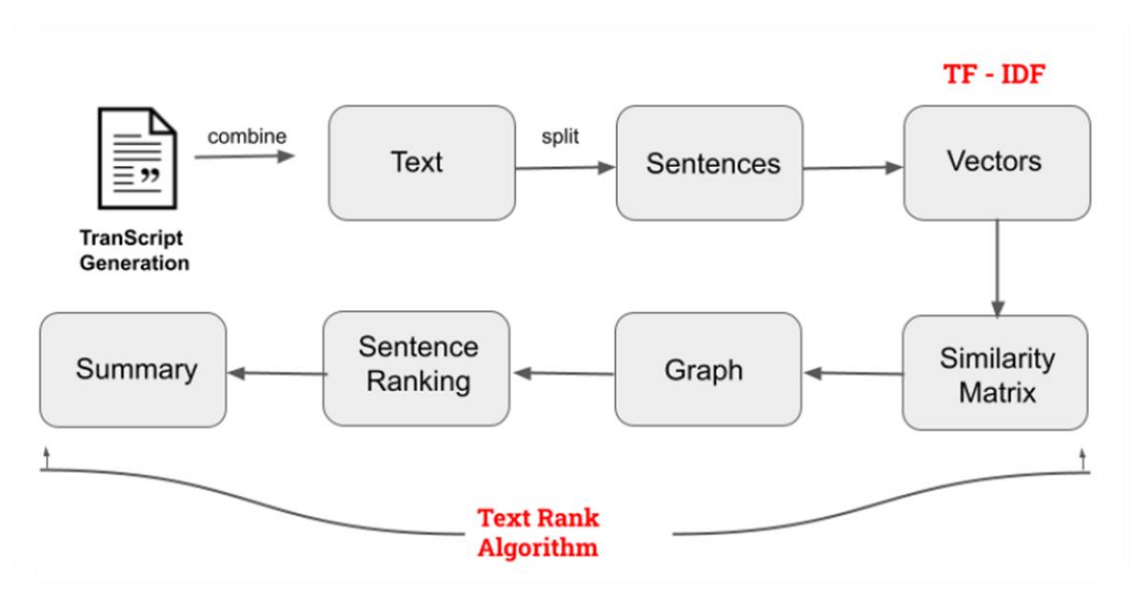


Fig.2.4.2 FLOW DIAGRRAM OF SUMMARIZATION

TextRank models any document as a graph using sentences as nodes. A function is computing the similarity of sentences to build edges in between. This function is used to weight the graph edges, the higher the similarity between the sentences the more important the edge between them will be in the graph. TextRank determines the relation of similarity between two sentences based on the content that both share.

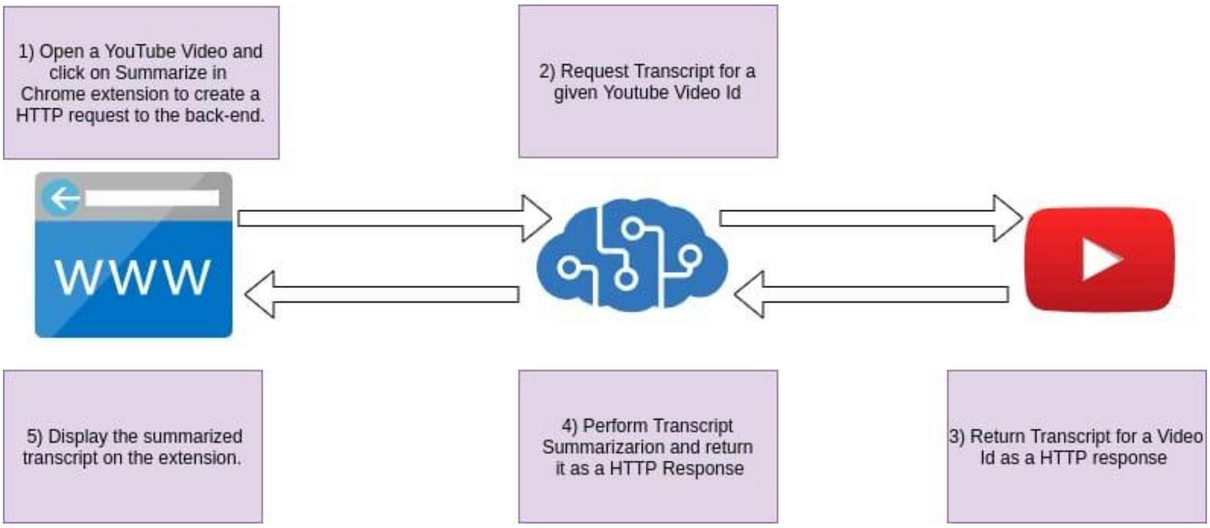
2.4.3 ADVANTAGES OF PROPOSED SYSTEM

1. In the Above Method and Algorithm There is no need of samples to train the algorithm, even there is no need of training the program with samples, The prediction level is high with greater accuracy than the other algorithms used for summarization.
2. Transcription Takes less time to Process in This Method due to their package and dataset.
3. Some Algorithms fails some time due to initializing packages and modules in this Text Rank Algorithm there wont be any error occurance, The Program runs Successfully with great Accuracy within a short time.
4. A Clear and Best Summarized Product will be delivered for the given input.
5. The Text Rank algorithm is based only on word concurrence and does not require any knowledge of the grammar. This excludes the need for creating tools dedicated to particular languages. How-ever the algorithm may be limited by different ways of separating sentences in various languages.
6. It is well defined and developed. At the moment many implementations of the algorithm exist, making it easily available for developers, who wish to utilize its properties.

2.5 SYSTEM ARCHITECTURE

The architectural design of a system emphasizes the design of the system architecture that describes the structure, behavior and more views of the proposed system and analysis.

ARCHITECTURE DIAGRAM OF TRANSCRIPT SUMMARIZATION



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 a given Youtube Video Id

3) Return Transcript for a Video Id as a HTTP response

4) Perform Transcript Summarizarion and return it as a HTTP Response

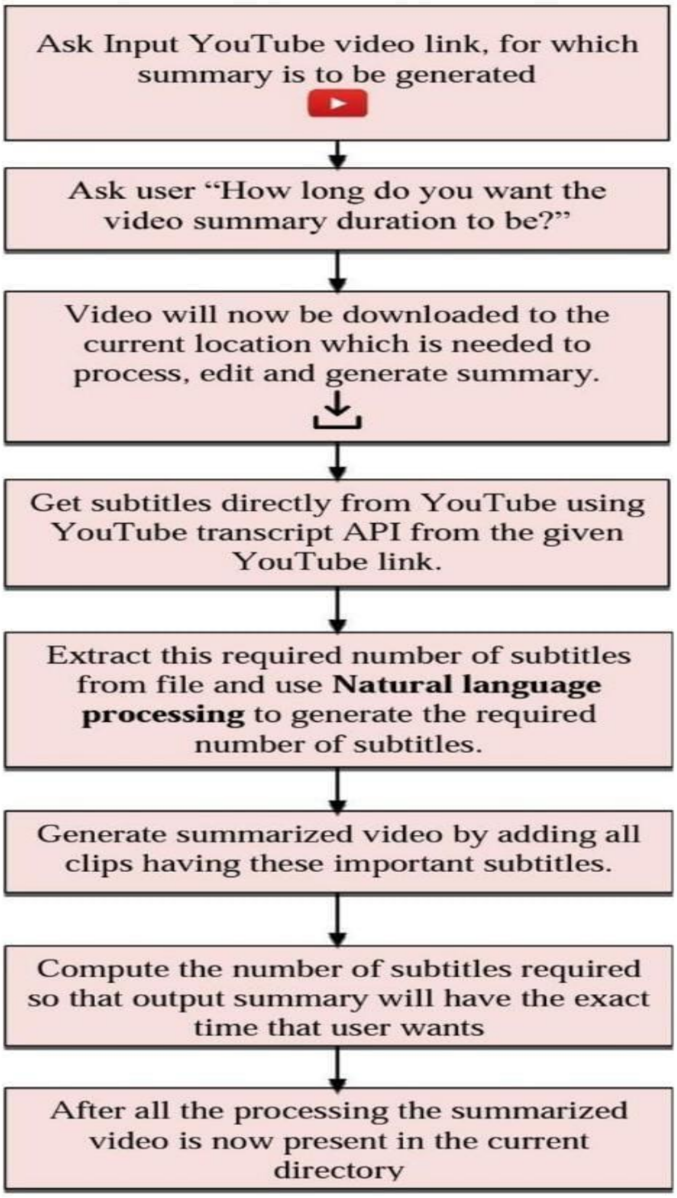
5) Display the summarized transcript on the extension.

Fig.2.5 DIAGRAM OF TRANSCRIPT SUMMARIZATION

High-Level Approach

1. Get transcripts/subtitles for a given YouTube video Id using a Python API.
2. Perform text summarization on obtained transcripts using HuggingFace transformers.
3. Build a Flask backend REST API to expose the summarization service to the client.
4. Develop a chrome extension which will utilize the backend API to display summarized text to the user.

METHODOLOGY CHART


Ask Input YouTube video link, for which summary is to be generated
Ask user "How long do you want the video summary duration to be?"
Video will now be downloaded to the current location which is needed to process, edit and generate summary.
Get subtitles directly from YouTube using YouTube transcript API from the given YouTube link.
Extract this required number of subtitles from file and use **Natural language processing** to generate the required number of subtitles.
Generate summarized video by adding all clips having these important subtitles.
Compute the number of subtitles required so that output summary will have the exact time that user wants
After all the processing the summarized video is now present in the current directory

CHAPTER - 3

SYSTEM REQUIREMENTS

A System Requirements specifies the project hardware and software requirements for developing the project. Supported operating system and run time environments where the project will run is on windows OS with python IDE. Since python IDE provides efficient way to implement AI And Machine Learning learning algorithms .

3.1 HARDWARE REQUIREMENTS :

RAM	:	4 GB and above
Processor	:	2.80 GHz and above
Hard Disk	:	120 GB and above
CPU type	:	Intel I3 Processor or Intel I5 Processor .
Clock speed	:	3.0 GH
Monitor type	:	15 Inch color monitor
Keyboard type	:	Internet keyboard

3.2 SOFTWARE REQUIREMENTS

Operating system	:	Windows 7, Windows 8, Windows 10 , Windows 11 .
Language	:	Python .
Documentation tool	:	Microsoft word 2007 and Above Versions .
Back end	:	JavaScript .
Simulation Tool	:	Visual Studio Code and Chrome .
Software Needed	:	Visual Studio Code , Python , Tensorflow , Youtube .

3.3 SOFTWARE FEATURES

3.3.1 PYTHON

Python is a dynamic, high level, free open source and interpreted programming language. It supports object-oriented programming as well as procedural oriented programming. In Python, we don't need to declare the type of variable because it is a dynamic typed language.

EASY TO CODE

Python is a very developer-friendly language which means that anyone and everyone can learn to code it in a couple of hours or days. As compared to other object-oriented programming languages like Java, C, C++, and C#, Python is one of the easiest to learn.

OPEN SOURCE AND FREE

Python is an open-source programming language which means that anyone can create and contribute to its development. Python has an online forum where thousands of coders gather daily to improve this language further. Along with this Python is free to download and use in any operating system, be it Windows, Mac or Linux.

3.3.2 Visual Studio Code:

Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE .Working with Python in Visual Studio Code, using the Microsoft Python extension, is simple, fun, and productive. The extension makes VS Code an excellent Python editor, and works on any operating system with a variety of Python interpreters .

3.3.3 TENSORFLOW

TensorFlow is an open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google. TensorFlow is Google Brain's second-generation system. While the reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs (with optional CUDA and SYCL extensions for general-purpose computing on graphics processing units). TensorFlow is available on 64-bit Linux, macOS, Windows, and mobile computing platforms including Android and iOS. Theano is a Python library that allows you to define, optimize, and efficiently evaluate mathematical expressions involving multi-dimensional arrays. It is built on top of NumPy.

Tight integration with NumPy: a similar interface to NumPy’s. numpy.nd arrays are also used internally in Theano-compiled functions.

Transparent use of a GPU: perform data-intensive computations up to 140x faster than on a CPU (support for float32 only).

Efficient symbolic differentiation: Theano can compute derivatives for functions of one or many inputs.

Speed and stability optimizations: avoid nasty bugs when computing expressions such as log(1 + exp(x)) for large values of x.

Dynamic C code generation: evaluate expressions faster.

Extensive unit-testing and self-verification: includes tools for detecting and diagnosing bugs and/or potential problems.

CHAPTER 4

SYSTEM DESIGN

The term “Design” is defined as the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application.

DATA FLOW DIAGRAM

DFD are used to Specify Functions of the Information System and how data flow from function to function. A data flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart. The data flow diagram is part of the structured analysis modeling tools.

DATA FLOW DIAGRAM AT THE INITIAL LEVEL (Level 0)

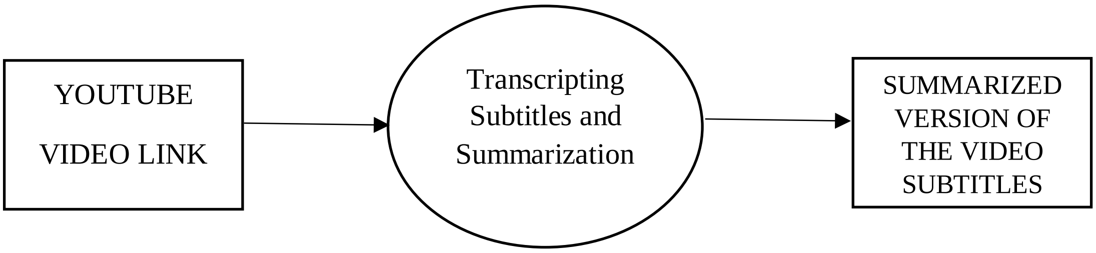

YOUTUBE
VIDEO LINK
Transcripting Subtitles and Summarization
SUMMARIZED VERSION OF THE VIDEO SUBTITLES

Fig 4.1 Flow Diagram Level 0 For The Initial Level

DFD Level 0 is also called a Context Diagram. It's a basic overview of the whole system or process being analyzed or modeled. It's designed to be an at-a-glance view, showing the system as a single high-level process, with its relationship to external entities .

In this the diagram represents the whole system of youtube transcript summarization in a single level process .

DATA FLOW DIAGRAM (LEVEL 1)

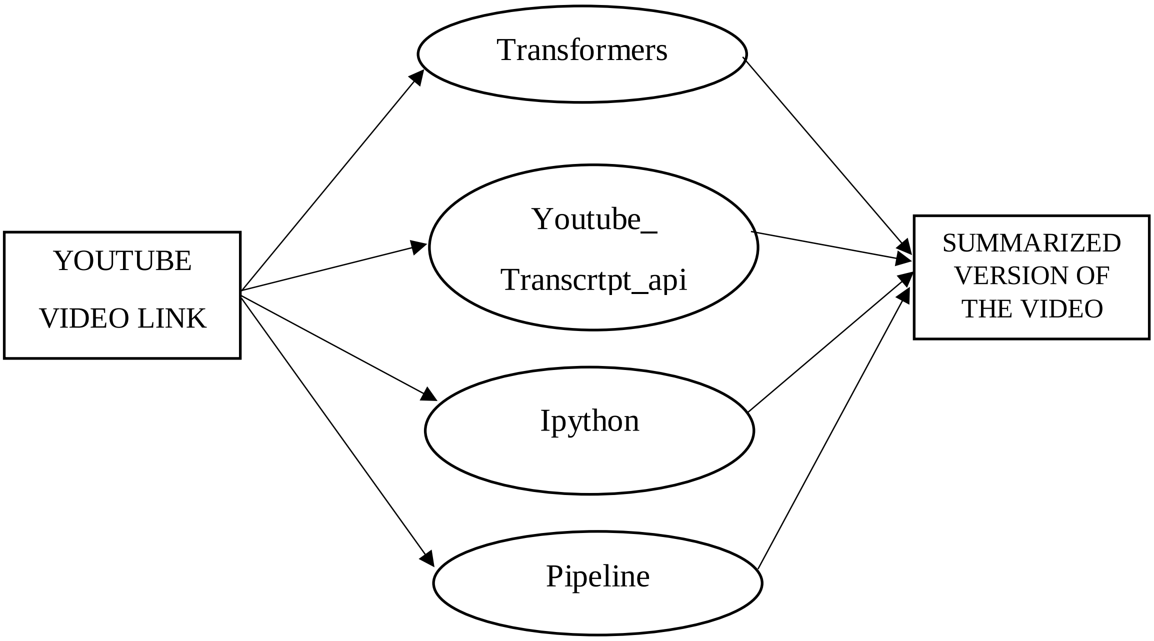


Fig.4.2 Data Flow Diagram Level 1

4.2 Data Flow Diagram Level 1 describes about the overall representation of each module and their functions. The level one data flow diagram has various modules and the respective results.As described previously, context diagrams (level 0 DFDs) are diagrams where the whole system is represented as a single process. A level 1 DFD notates each of the main sub-processes that together form the complete system.

DATA FLOW DIAGRAM (LEVEL 2)

In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses. 2-level DFD: 2-level DFD goes one step deeper into parts of 1-level DFD. It can be used to plan or record the specific/necessary detail about the system's functioning and We are going to see

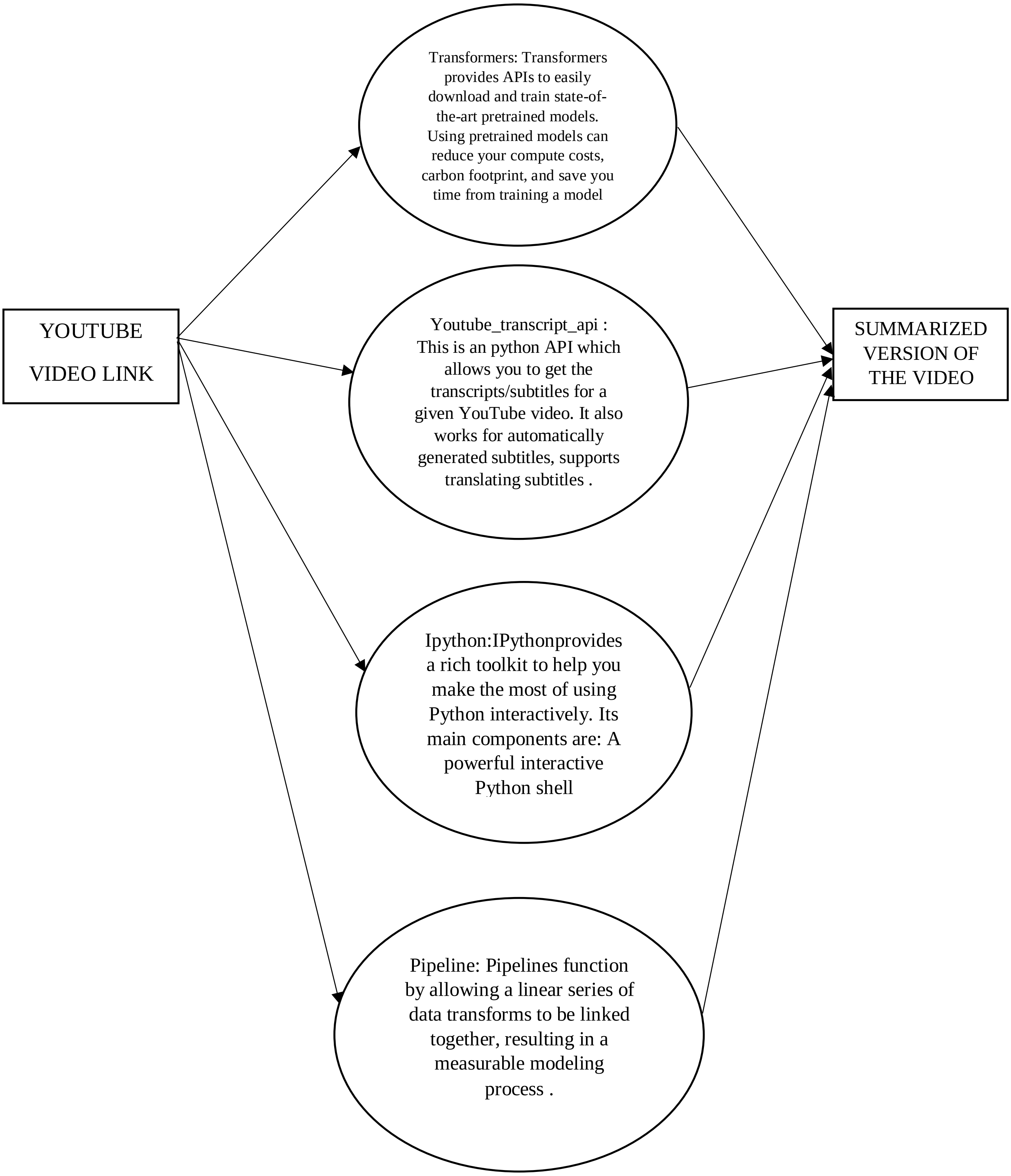

YOUTUBE
VIDEO LINK
Transformers: Transformers provides APIs to easily download and train state-of-the-art pretrained models. Using pretrained models can reduce your compute costs, carbon footprint, and save you time from training a model
Youtube_transcript_api : This is an python API which allows you to get the transcripts/subtitles for a given YouTube video. It also works for automatically generated subtitles, supports translating subtitles .
Ipython:IPythonprovides a rich toolkit to help you make the most of using Python interactively. Its main components are: A powerful interactive Python shell
Pipeline: Pipelines function by allowing a linear series of data transforms to be linked together, resulting in a measurable modeling process .
SUMMARIZED
VERSION OF
THE VIDEO

Fig.4.3 Data Flow Diagram Level 2

DATA FLOW DIAGRAM (LEVEL 3)

Data Flow Diagram Level 3 describes about the overall Transformers Actions of each module and their functions. The level one data flow diagram has various modules and the respective results.As described previously, context diagrams (level 0 DFDs) are diagrams where the whole system is represented as a single process. A level 1 DFD notates each of the main sub-processes that together form the complete system.

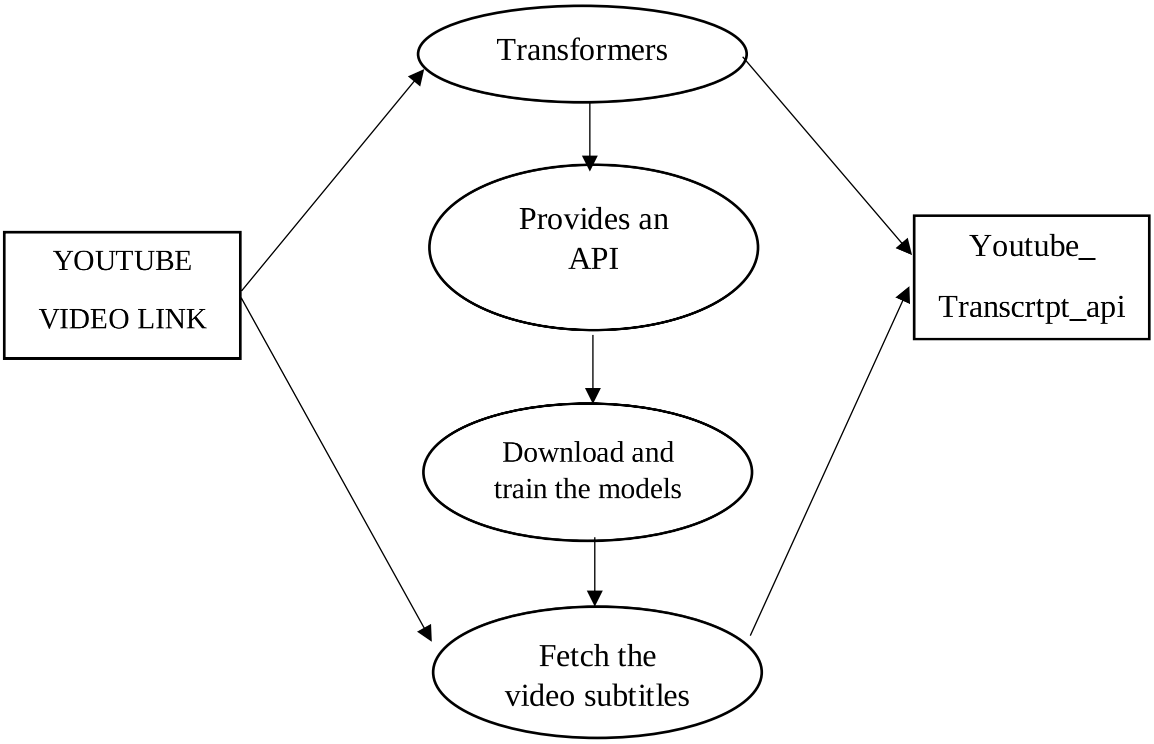


Fig.4.4 Data Flow Diagram Level 3

DATA FLOW DIAGRAM (LEVEL 4)

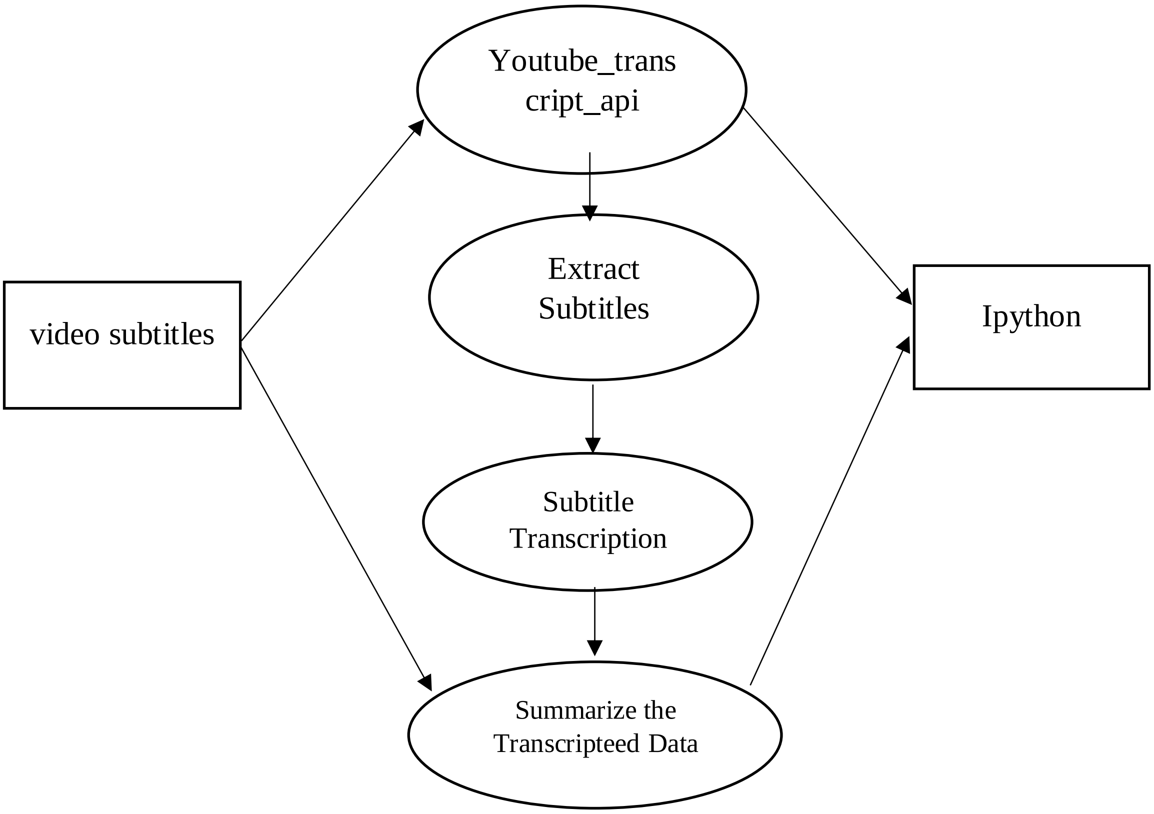


Fig.4.5 Data Flow Diagram Level 4

Data Flow Diagram Level 4 describes about the overall representation of each module and their functions. The level one data flow diagram has various modules and the respective results.As described previously

DATA FLOW DIAGRAM (LEVEL 5)

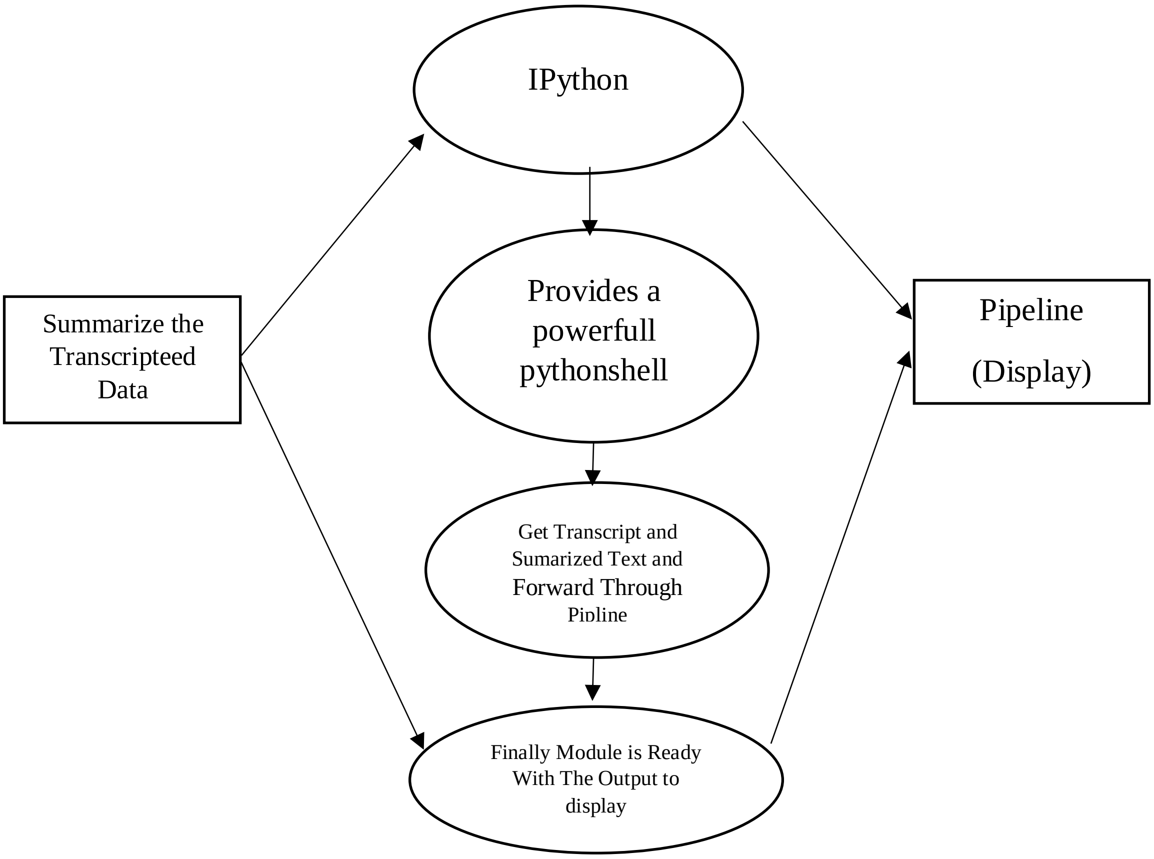

IPython
Summarize the Transcripteed Data
Provides a powerfull pythonshell
Pipeline (Display)
Get Transcript and Sumarized Text and Forward Through Pipline
Finally Module is Ready With The Output to display

Fig.4.5 Data Flow Diagram Level 5

4.6 USECASE DIAGRAM

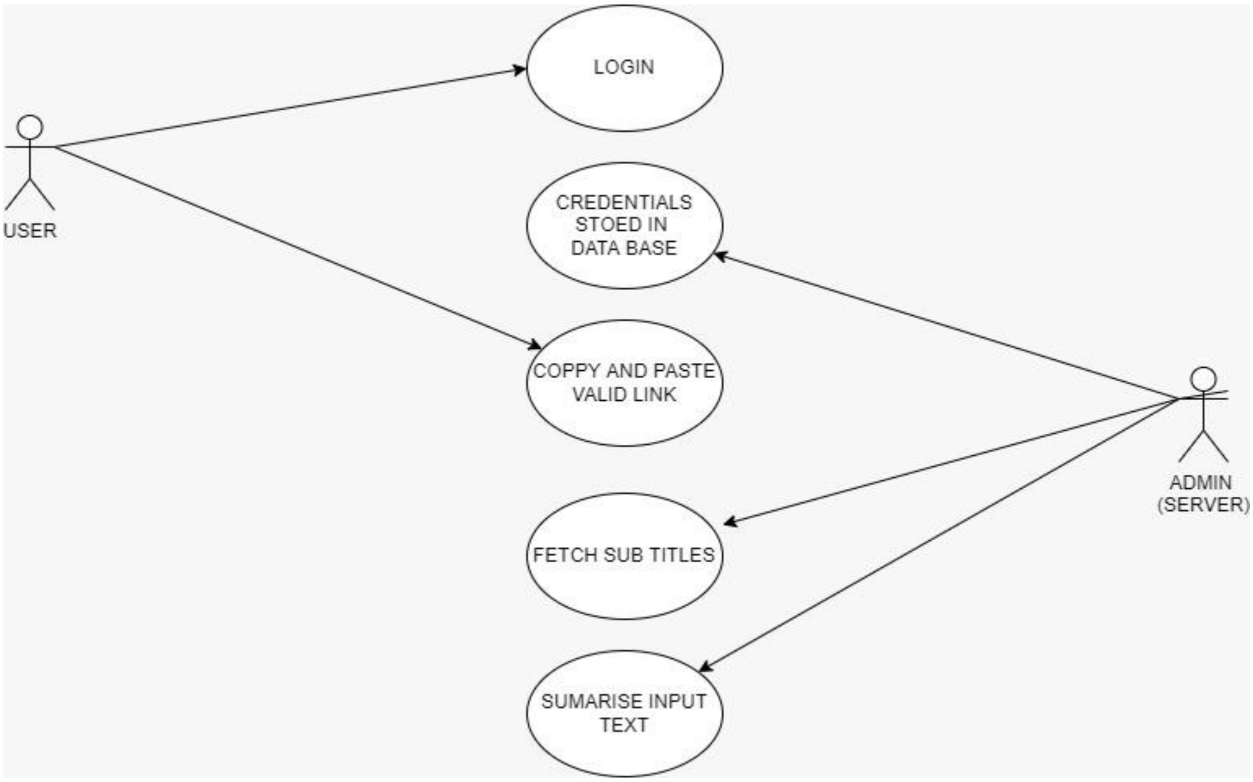


Fig.4.6 Usecase Diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses .

4.7 SEQUENCE DIAGRAM

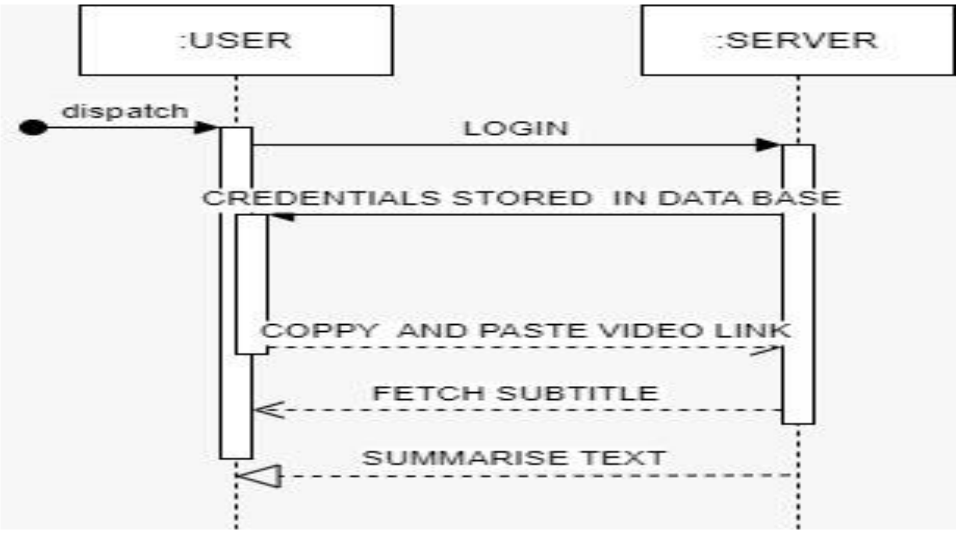


Fig.4.7 Sequence Diagram

A sequence diagram or system sequence diagram shows process interactions arranged in time sequence in the field of software engineering. It depicts the processes involved and the sequence of messages exchanged between the processes needed to carry out the functionality.

4.8 ACTIVITY DIAGRAM

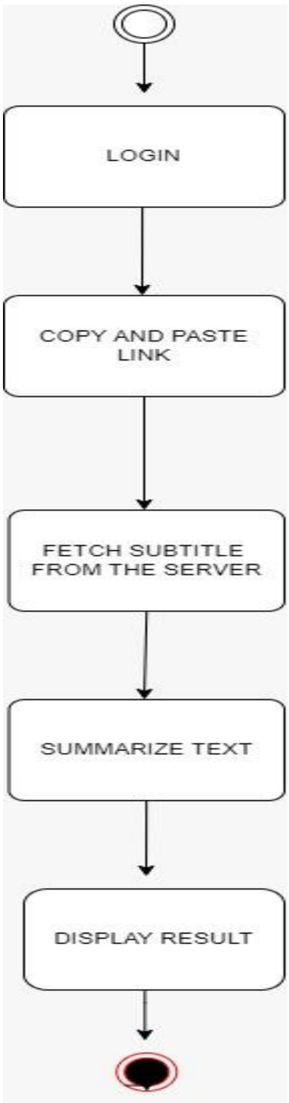


Fig.4.8 Activity Diagram

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

CHAPTER-5
SYSTEM IMPLEMENTATION

5.1 SYSTEM DEVELOPMENT

Since there are three types of transcript that we can extract - manually generated transcript, automatically generated transcript, and the videos that contain no transcript. We are not considering videos that do not have transcripts. Secondly, when we get the transcript of a given Youtube video since it does not contain any punctuations like comma(,), full stops(.) which is very important for us in finding the boundaries of a sentence, so we will restore punctuations from our extracted transcript by using the python library known as “punctuator”. Now we will apply the text preprocessing methods to clean the extracted transcript by tokenizing the sentences as well as the words, lowercasing it, removing stop words like a, an, the, etc, removing punctuations, and stemming or lemmatization to generate the root form of inflected words.

Performing text summarization: This task consists of shortening a large form of text into a precise summary that keeps all the necessary information intact and preserves the overall meaning. For this purpose in NLP for text summarization, there are two types of methods used:
Extractive Summarization: In this type of text summarization, the output is only the important phrases and sentences that the model identifies from the original text. For the purpose of extractive summarization, we have used the TF-IDF model with Text Rank Algorithm.

TF-IDF(Term Frequency - Inverse Document Frequency) After the cleaning process, we have to convert the words into it’s vectorized form so that our algorithm will process it by using **TF-IDF**. This is a technique to measure the quantity of a word in documents, we compute a weight to each word which signifies the importance of the word in the document and corpus.
TF(Term Frequency): TF calculates the frequency of a word in a document.

TF = No. of repetition of the word in the sentence / No. of words in a sentence

IDF(Inverse Document Frequency): IDF is the inverse of the document frequency which measures the informativeness of term t.

IDF = log(No. of sentences / No. of sentences containing words)

After this, we will multiply both matrices to obtain the vectorized form which tells us which words are the most important.

Text Rank Algorithm

It is based on the PageRank algorithm which calculates the rank of web pages which is used by search engines such as Google. Using the concept of this we will rank the most important sentences in the text and generate a summary. For the task of automated summarization, TextRank models any document as a graph using sentences as nodes. A function is computing the similarity of sentences to build edges in between. This function is used to weight the graph edges, the higher the similarity between the sentences the more important the edge between them will be in the graph. TextRank determines the relation of similarity between two sentences based on the content that both share.

This overlap is calculated simply as the No. of common lexical tokens between them, divided by the length of each to avoid promoting long sentences. Cosine similarity of two words A & B is computed using following formula:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Modules which are used in this projects are listed and given a short note on it .

- Transformers ,
- Youtube_transcript_api ,
- Ipython , And
- Pipeline .

Transformers

Transformers provides APIs to easily download and train state-of-the-art pretrained models. Using pretrained models can reduce your compute costs, carbon footprint, and save you time from training a model from scratch.

Youtube_transcript_api

This is an python API which allows you to get the transcripts/subtitles for a given YouTube video. It also works for automatically generated subtitles, supports translating subtitles.

Ipython

IPythonprovides a rich toolkit to help you make the most of using Python interactively. Its main components are: A powerful interactive Python shell .

Pipeline

Pipelines function by allowing a linear series of data transforms to be linked together, resulting in a measurable modeling process .

Therefore,

These are the list of modules used in this youtube transcript summarization.

CHAPTER-6

CONCLUSION

6.1 CONCLUSION

In this Project, we propose two different methods to generate summary and important keywords from the given Youtube video - extractive and abstractive. We have made a simple user interface through which users can easily get their summaries through these methods, and surely find it easy to interact with our user interface and get what they want. We are sure that our project will surely satisfy the users and solve all the problems that it's supposed to tackle which is saving time and efforts, by providing only the useful information about the topic which interests them so that they don't have to watch those long videos and the time that saved can be used in gaining more knowledge.

6.2 FUTURE SCOPE

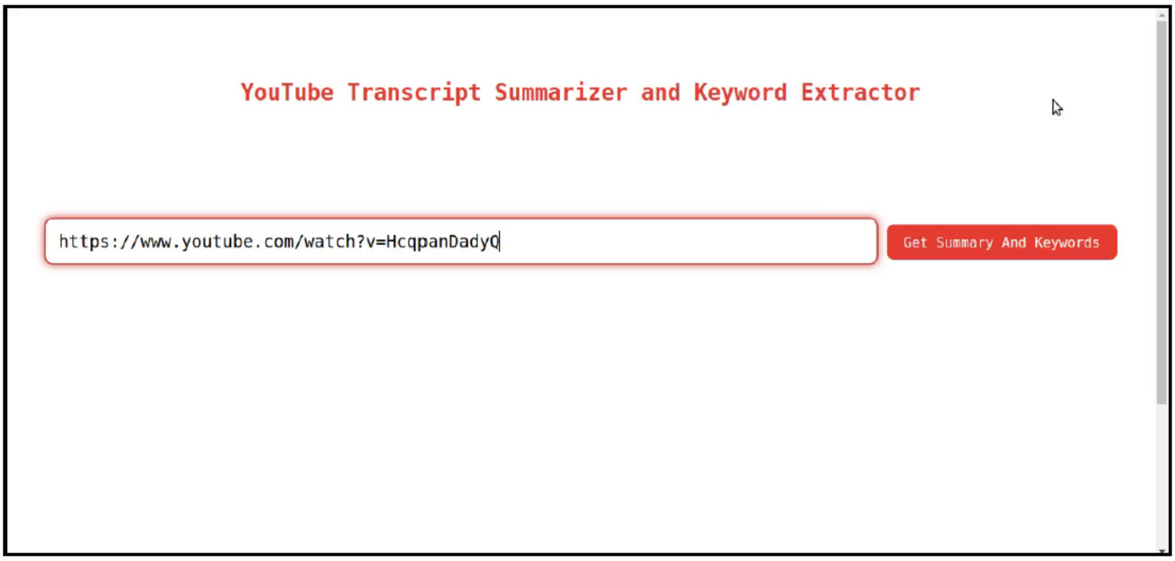
The hearing-impaired individuals and the students are the groups of people in the society that would be benefitted the most with the Transcript Summarization of YouTube videos. The hearing-impaired people find it difficult to understand the videos without the transcripts or subtitles. It would be helpful for them if summary is generated even for the videos which do not have readily available transcripts. Summarising the transcripts of the YouTube videos would also help the students to pick lecture/tutorial videos based on their preferences. The concept of Transcript Summarisation can also be extended to other streaming services as well.

APPENDICES

A. SOURCE CODE

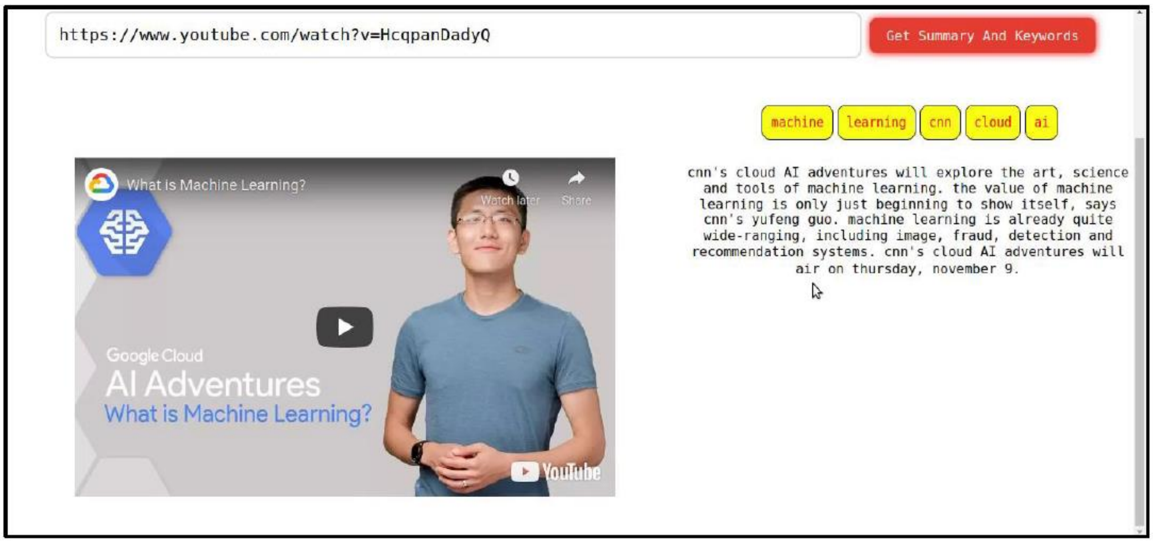
```
# Youtube Transcript Summarization Code!!!
from transformers import pipeline
from youtube_transcript_api import YouTubeTranscriptApi
print("Give a Youtube Video Link:")
# Example Link :  https://www.youtube.com/watch?v=L0t69f8UcmU
youtube_video = input()
video_id = youtube_video.split("=")[1]
from IPython.display import YouTubeVideo
YouTubeVideo(video_id)
YouTubeTranscriptApi.get_transcript(video_id)
transcript = YouTubeTranscriptApi.get_transcript(video_id)
result = ""
for i in transcript:
result += ' ' + i['text']
print("Input Text: \n" + result)
print(len(result))
summarizer = pipeline('summarization')
num_iters = int(len(result)/1000)
summarized_text = []
print("\nSummarized text :\n")
for i in range(0, num_iters + 1):
  start = 0
  start = i * 1000
  end = (i + 1) * 1000
  out = summarizer(result[start:end],min_length=10 , max_length=45)
  out = out[0]
  out = out['summary_text']
  print(out)
```

B. UI SAMPLE OUTPUT



YouTube Transcript Summarizer and Keyword Extractor

https://www.youtube.com/watch?v=HcqpanDadyQ

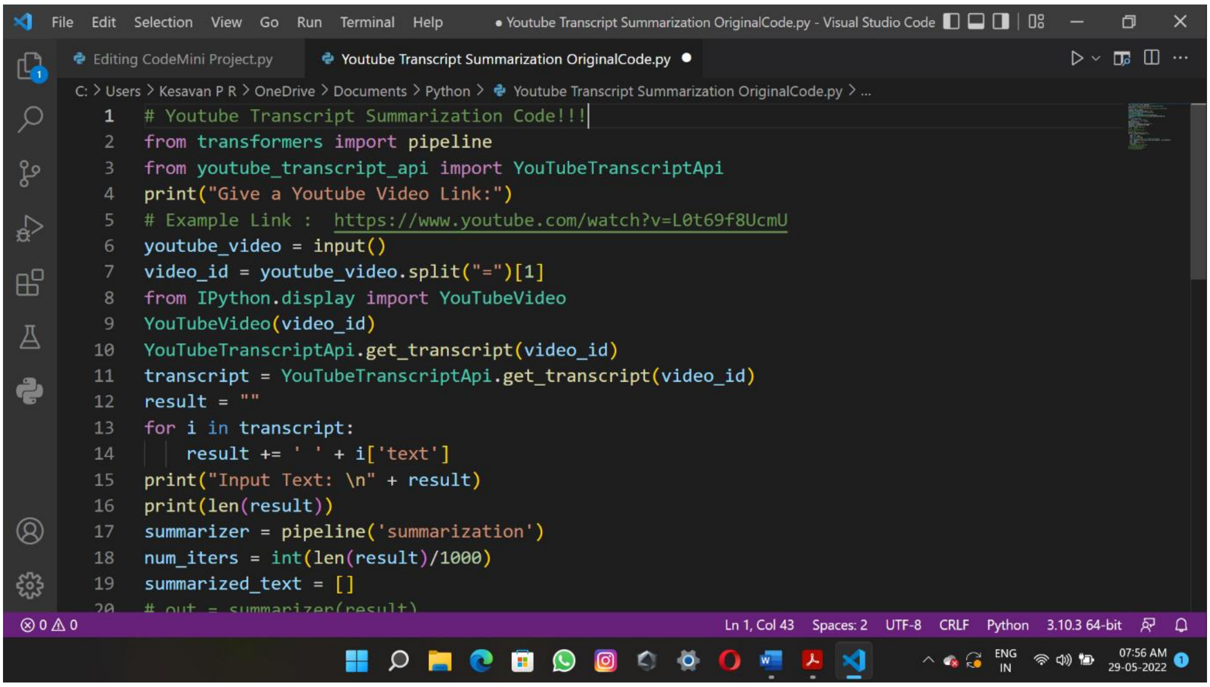


https://www.youtube.com/watch?v=HcqpanDadyQ

machine learning cnn cloud ai

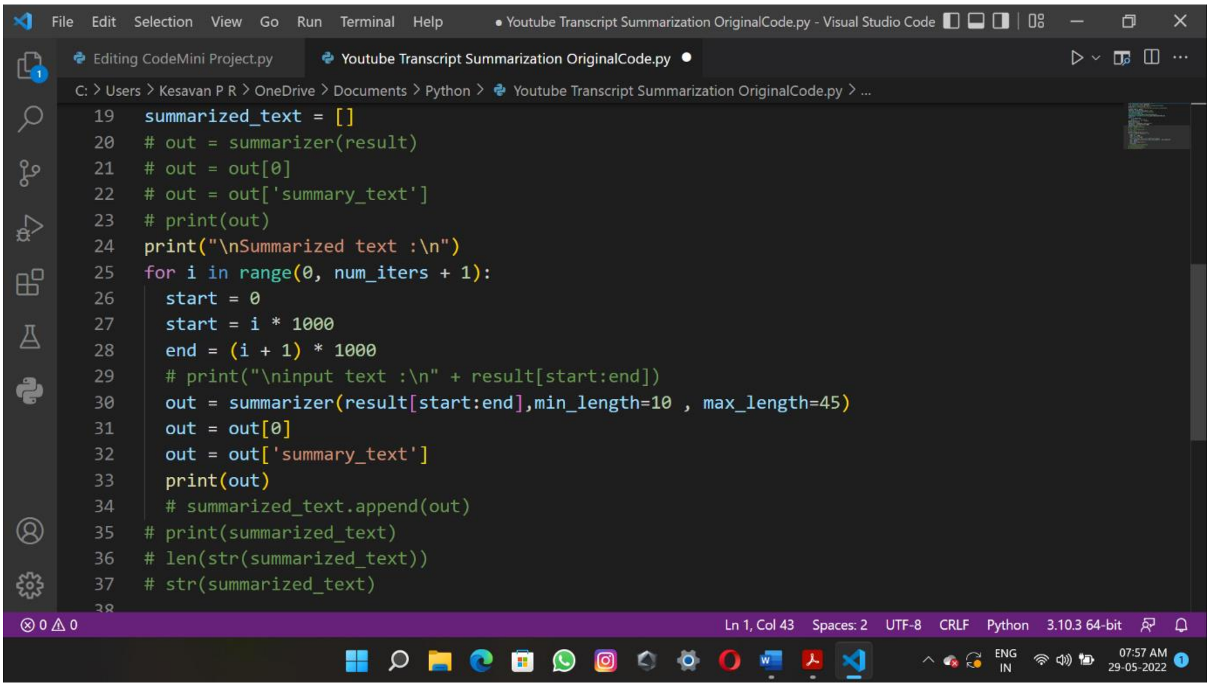
cnn's cloud AI adventures will explore the art, science and tools of machine learning. the value of machine learning is only just beginning to show itself, says cnn's yufeng guo. machine learning is already quite wide-ranging, including image, fraud, detection and recommendation systems. cnn's cloud AI adventures will air on thursday, november 9.

C. CODE



File Edit Selection View Go Run Terminal Help
● Youtube Transcript Summarization OriginalCode.py - Visual Studio Code
Editing CodeMini Project.py
Youtube Transcript Summarization OriginalCode.py ●
C: > Users > Kesavan P R > OneDrive > Documents > Python > Youtube Transcript Summarization OriginalCode.py > ...
⊗ 0 ⚠ 0
Ln 1, Col 43 Spaces: 2 UTF-8 CRLF Python 3.10.3 64-bit
ENG IN
07:56 AM 29-05-2022

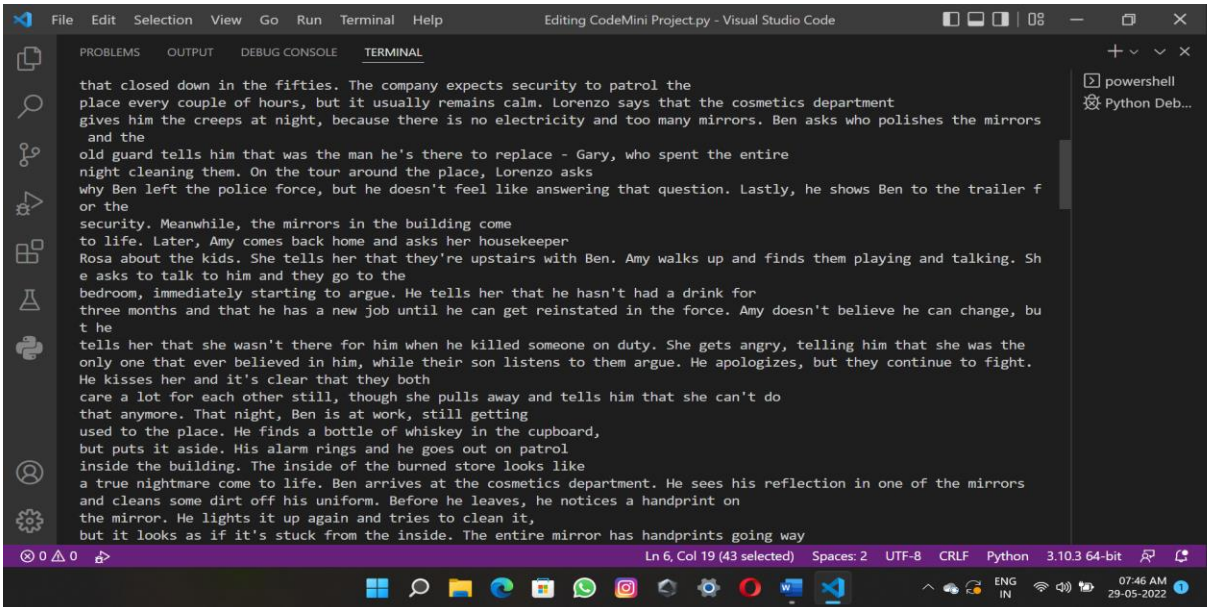
```python
# Youtube Transcript Summarization Code!!!
from transformers import pipeline
from youtube_transcript_api import YouTubeTranscriptApi
print("Give a Youtube Video Link:")
# Example Link :  https://www.youtube.com/watch?v=L0t69f8UcmU
youtube_video = input()
video_id = youtube_video.split("=")[1]
from IPython.display import YouTubeVideo
YouTubeVideo(video_id)
YouTubeTranscriptApi.get_transcript(video_id)
transcript = YouTubeTranscriptApi.get_transcript(video_id)
result = ""
for i in transcript:
    result += ' ' + i['text']
print("Input Text: \n" + result)
print(len(result))
summarizer = pipeline('summarization')
num_iters = int(len(result)/1000)
summarized_text = []
# out = summarizer(result)
```



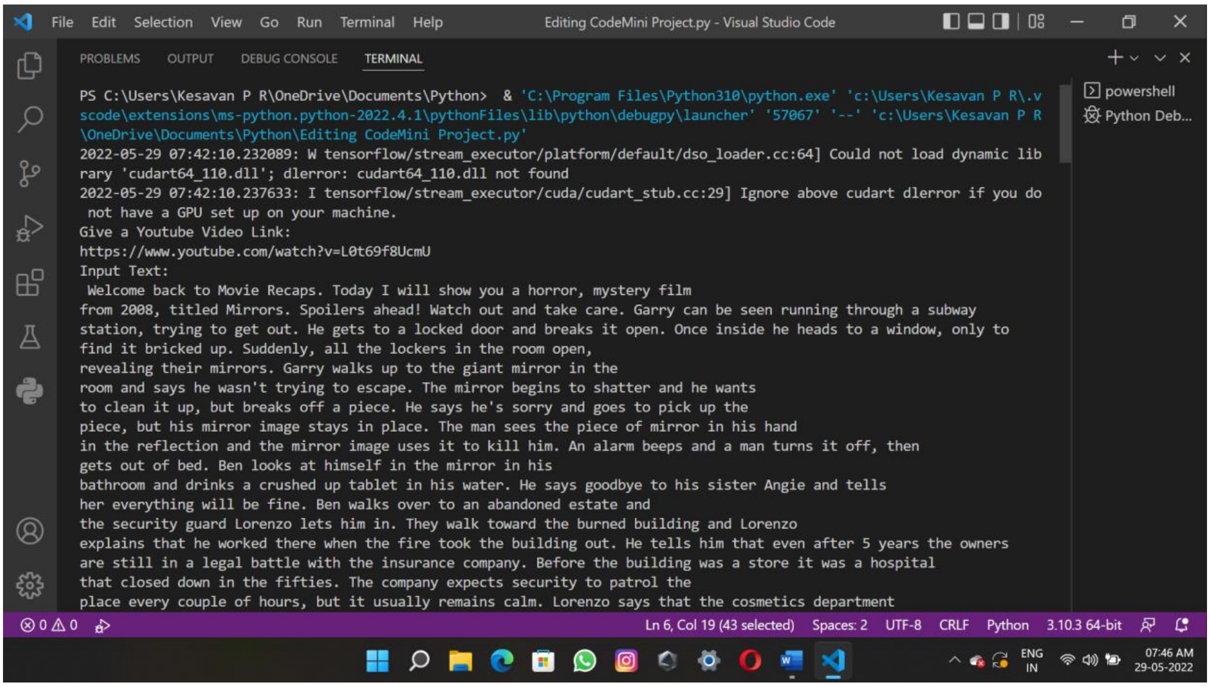
File Edit Selection View Go Run Terminal Help
● Youtube Transcript Summarization OriginalCode.py - Visual Studio Code
Editing CodeMini Project.py
Youtube Transcript Summarization OriginalCode.py ●
C: > Users > Kesavan P R > OneDrive > Documents > Python > Youtube Transcript Summarization OriginalCode.py > ...
⊗ 0 ⚠ 0
Ln 1, Col 43 Spaces: 2 UTF-8 CRLF Python 3.10.3 64-bit
ENG IN
07:57 AM 29-05-2022

```python
summarized_text = []
# out = summarizer(result)
# out = out[0]
# out = out['summary_text']
# print(out)
print("\nSummarized text :\n")
for i in range(0, num_iters + 1):
  start = 0
  start = i * 1000
  end = (i + 1) * 1000
  # print("\ninput text :\n" + result[start:end])
  out = summarizer(result[start:end],min_length=10 , max_length=45)
  out = out[0]
  out = out['summary_text']
  print(out)
  # summarized_text.append(out)
# print(summarized_text)
# len(str(summarized_text))
# str(summarized_text)
```

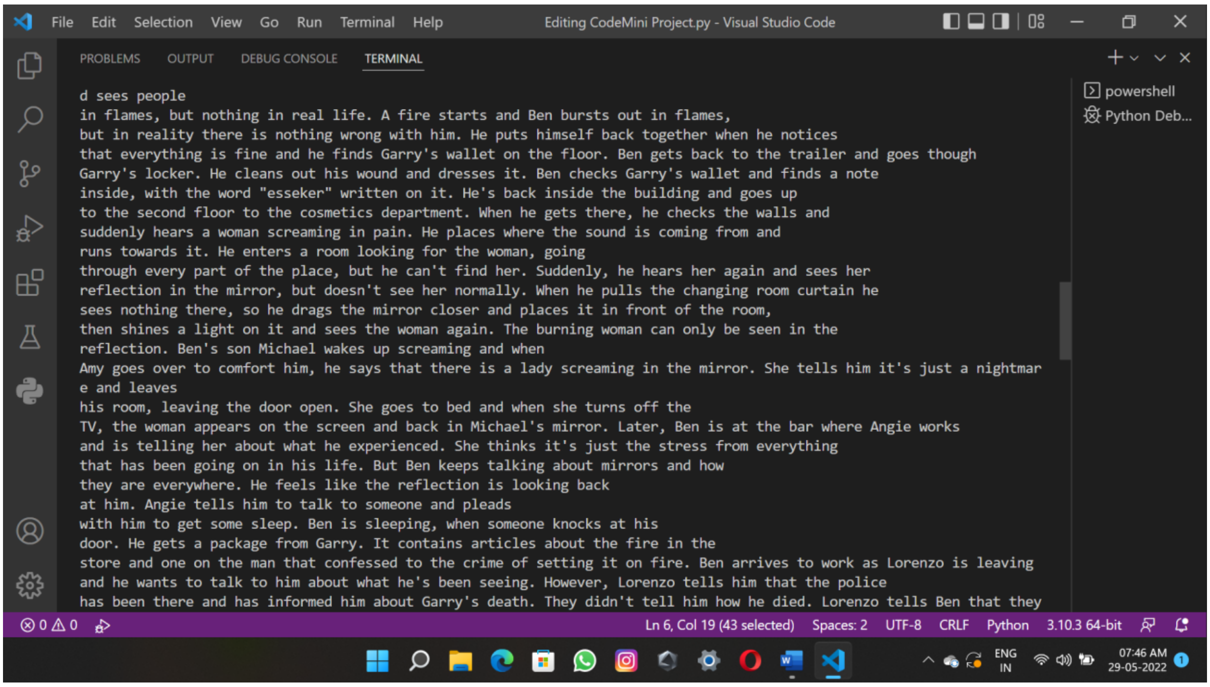
D. CODE OUTPUT



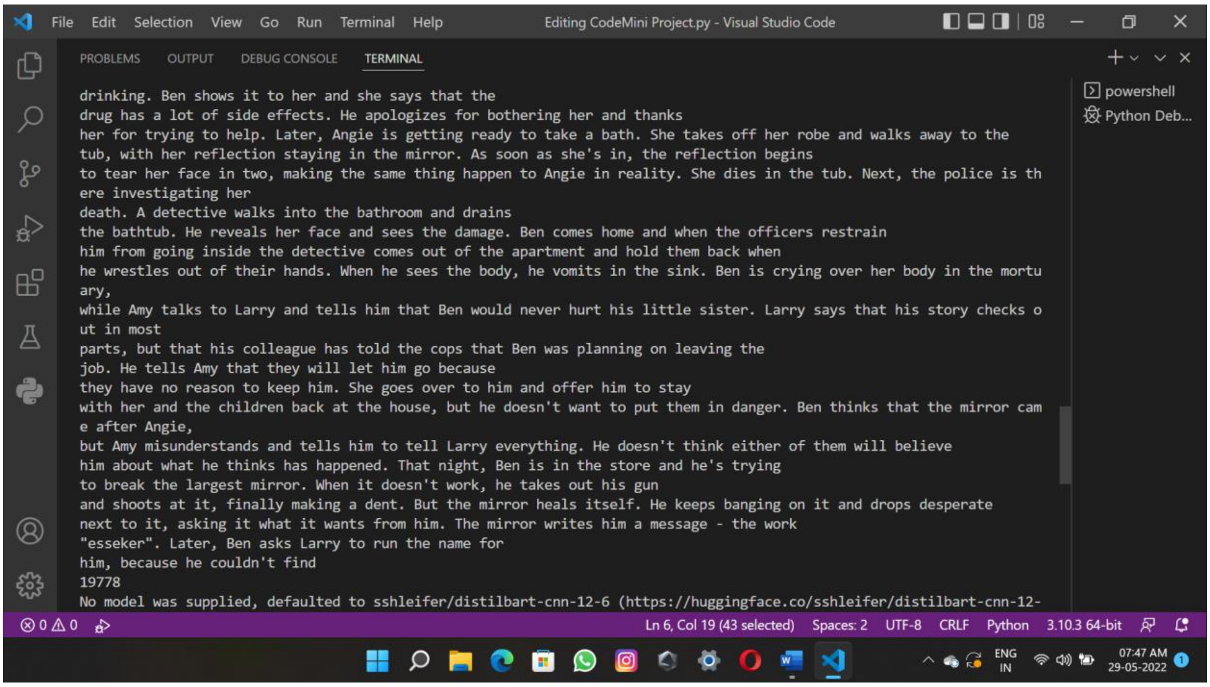
```
that closed down in the fifties. The company expects security to patrol the
place every couple of hours, but it usually remains calm. Lorenzo says that the cosmetics department
gives him the creeps at night, because there is no electricity and too many mirrors. Ben asks who polishes the mirrors
 and the
old guard tells him that was the man he's there to replace - Gary, who spent the entire
night cleaning them. On the tour around the place, Lorenzo asks
why Ben left the police force, but he doesn't feel like answering that question. Lastly, he shows Ben to the trailer f
or the
security. Meanwhile, the mirrors in the building come
to life. Later, Amy comes back home and asks her housekeeper
Rosa about the kids. She tells her that they're upstairs with Ben. Amy walks up and finds them playing and talking. Sh
e asks to talk to him and they go to the
bedroom, immediately starting to argue. He tells her that he hasn't had a drink for
three months and that he has a new job until he can get reinstated in the force. Amy doesn't believe he can change, bu
t he
tells her that she wasn't there for him when he killed someone on duty. She gets angry, telling him that she was the
only one that ever believed in him, while their son listens to them argue. He apologizes, but they continue to fight.
He kisses her and it's clear that they both
care a lot for each other still, though she pulls away and tells him that she can't do
that anymore. That night, Ben is at work, still getting
used to the place. He finds a bottle of whiskey in the cupboard,
but puts it aside. His alarm rings and he goes out on patrol
inside the building. The inside of the burned store looks like
a true nightmare come to life. Ben arrives at the cosmetics department. He sees his reflection in one of the mirrors
and cleans some dirt off his uniform. Before he leaves, he notices a handprint on
the mirror. He lights it up again and tries to clean it,
but it looks as if it's stuck from the inside. The entire mirror has handprints going way
```



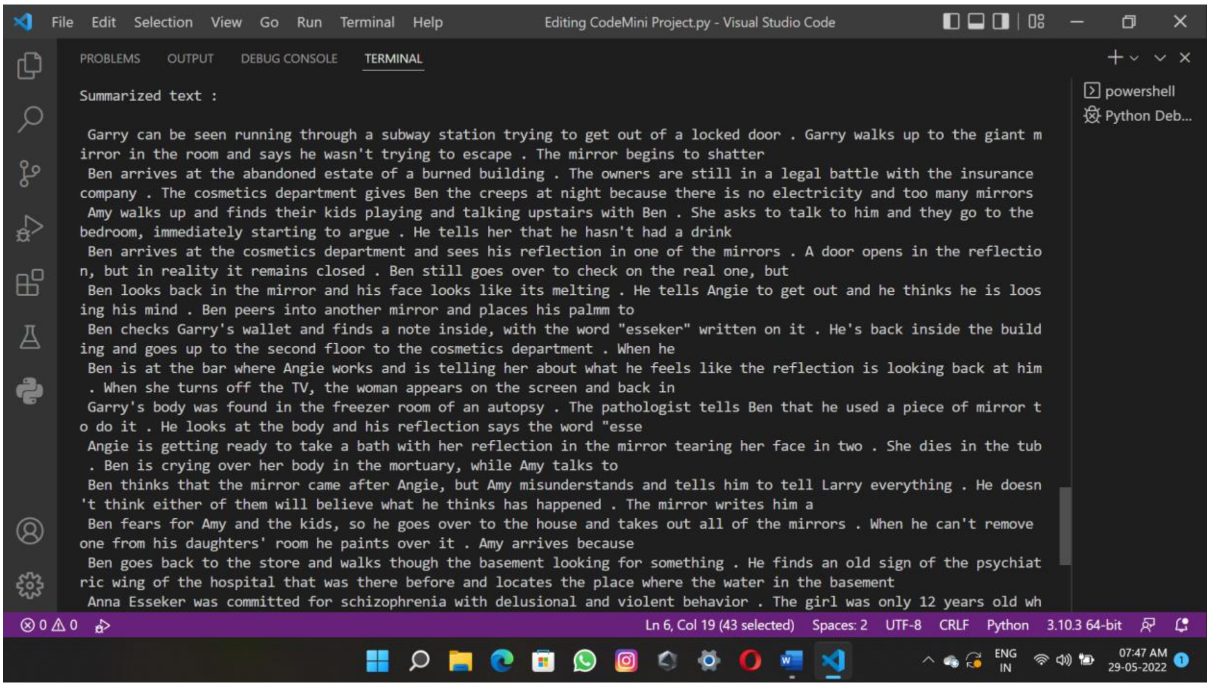
```
PS C:\Users\Kesavan P R\OneDrive\Documents\Python>  & 'C:\Program Files\Python310\python.exe' 'c:\Users\Kesavan P R\.v
scode\extensions\ms-python.python-2022.4.1\pythonFiles\lib\python\debugpy\launcher' '57067' '--' 'c:\Users\Kesavan P R
\OneDrive\Documents\Python\Editing CodeMini Project.py'
2022-05-29 07:42:10.232089: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic lib
rary 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-05-29 07:42:10.237633: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do
 not have a GPU set up on your machine.
Give a Youtube Video Link:
https://www.youtube.com/watch?v=L0t69f8UcmU
Input Text:
 Welcome back to Movie Recaps. Today I will show you a horror, mystery film
from 2008, titled Mirrors. Spoilers ahead! Watch out and take care. Garry can be seen running through a subway
station, trying to get out. He gets to a locked door and breaks it open. Once inside he heads to a window, only to
find it bricked up. Suddenly, all the lockers in the room open,
revealing their mirrors. Garry walks up to the giant mirror in the
room and says he wasn't trying to escape. The mirror begins to shatter and he wants
to clean it up, but breaks off a piece. He says he's sorry and goes to pick up the
piece, but his mirror image stays in place. The man sees the piece of mirror in his hand
in the reflection and the mirror image uses it to kill him. An alarm beeps and a man turns it off, then
gets out of bed. Ben looks at himself in the mirror in his
bathroom and drinks a crushed up tablet in his water. He says goodbye to his sister Angie and tells
her everything will be fine. Ben walks over to an abandoned estate and
the security guard Lorenzo lets him in. They walk toward the burned building and Lorenzo
explains that he worked there when the fire took the building out. He tells him that even after 5 years the owners
are still in a legal battle with the insurance company. Before the building was a store it was a hospital
that closed down in the fifties. The company expects security to patrol the
place every couple of hours, but it usually remains calm. Lorenzo says that the cosmetics department
```



```
d sees people
in flames, but nothing in real life. A fire starts and Ben bursts out in flames,
but in reality there is nothing wrong with him. He puts himself back together when he notices
that everything is fine and he finds Garry's wallet on the floor. Ben gets back to the trailer and goes though
Garry's locker. He cleans out his wound and dresses it. Ben checks Garry's wallet and finds a note
inside, with the word "esseker" written on it. He's back inside the building and goes up
to the second floor to the cosmetics department. When he gets there, he checks the walls and
suddenly hears a woman screaming in pain. He places where the sound is coming from and
runs towards it. He enters a room looking for the woman, going
through every part of the place, but he can't find her. Suddenly, he hears her again and sees her
reflection in the mirror, but doesn't see her normally. When he pulls the changing room curtain he
sees nothing there, so he drags the mirror closer and places it in front of the room,
then shines a light on it and sees the woman again. The burning woman can only be seen in the
reflection. Ben's son Michael wakes up screaming and when
Amy goes over to comfort him, he says that there is a lady screaming in the mirror. She tells him it's just a nightmar
e and leaves
his room, leaving the door open. She goes to bed and when she turns off the
TV, the woman appears on the screen and back in Michael's mirror. Later, Ben is at the bar where Angie works
and is telling her about what he experienced. She thinks it's just the stress from everything
that has been going on in his life. But Ben keeps talking about mirrors and how
they are everywhere. He feels like the reflection is looking back
at him. Angie tells him to talk to someone and pleads
with him to get some sleep. Ben is sleeping, when someone knocks at his
door. He gets a package from Garry. It contains articles about the fire in the
store and one on the man that confessed to the crime of setting it on fire. Ben arrives to work as Lorenzo is leaving
and he wants to talk to him about what he's been seeing. However, Lorenzo tells him that the police
has been there and has informed him about Garry's death. They didn't tell him how he died. Lorenzo tells Ben that they
```



```
drinking. Ben shows it to her and she says that the
drug has a lot of side effects. He apologizes for bothering her and thanks
her for trying to help. Later, Angie is getting ready to take a bath. She takes off her robe and walks away to the
tub, with her reflection staying in the mirror. As soon as she's in, the reflection begins
to tear her face in two, making the same thing happen to Angie in reality. She dies in the tub. Next, the police is th
ere investigating her
death. A detective walks into the bathroom and drains
the bathtub. He reveals her face and sees the damage. Ben comes home and when the officers restrain
him from going inside the detective comes out of the apartment and hold them back when
he wrestles out of their hands. When he sees the body, he vomits in the sink. Ben is crying over her body in the mortu
ary,
while Amy talks to Larry and tells him that Ben would never hurt his little sister. Larry says that his story checks o
ut in most
parts, but that his colleague has told the cops that Ben was planning on leaving the
job. He tells Amy that they will let him go because
they have no reason to keep him. She goes over to him and offer him to stay
with her and the children back at the house, but he doesn't want to put them in danger. Ben thinks that the mirror cam
e after Angie,
but Amy misunderstands and tells him to tell Larry everything. He doesn't think either of them will believe
him about what he thinks has happened. That night, Ben is in the store and he's trying
to break the largest mirror. When it doesn't work, he takes out his gun
and shoots at it, finally making a dent. But the mirror heals itself. He keeps banging on it and drops desperate
next to it, asking it what it wants from him. The mirror writes him a message - the work
"esseker". Later, Ben asks Larry to run the name for
him, because he couldn't find
19778
No model was supplied, defaulted to sshleifer/distilbart-cnn-12-6 (https://huggingface.co/sshleifer/distilbart-cnn-12-
```



```
Summarized text :

 Garry can be seen running through a subway station trying to get out of a locked door . Garry walks up to the giant m
irror in the room and says he wasn't trying to escape . The mirror begins to shatter
 Ben arrives at the abandoned estate of a burned building . The owners are still in a legal battle with the insurance
company . The cosmetics department gives Ben the creeps at night because there is no electricity and too many mirrors
 Amy walks up and finds their kids playing and talking upstairs with Ben . She asks to talk to him and they go to the
bedroom, immediately starting to argue . He tells her that he hasn't had a drink
 Ben arrives at the cosmetics department and sees his reflection in one of the mirrors . A door opens in the reflectio
n, but in reality it remains closed . Ben still goes over to check on the real one, but
 Ben looks back in the mirror and his face looks like its melting . He tells Angie to get out and he thinks he is loos
ing his mind . Ben peers into another mirror and places his palmm to
 Ben checks Garry's wallet and finds a note inside, with the word "esseker" written on it . He's back inside the build
ing and goes up to the second floor to the cosmetics department . When he
 Ben is at the bar where Angie works and is telling her about what he feels like the reflection is looking back at him
 . When she turns off the TV, the woman appears on the screen and back in
 Garry's body was found in the freezer room of an autopsy . The pathologist tells Ben that he used a piece of mirror t
o do it . He looks at the body and his reflection says the word "esse
 Angie is getting ready to take a bath with her reflection in the mirror tearing her face in two . She dies in the tub
 . Ben is crying over her body in the mortuary, while Amy talks to
 Ben thinks that the mirror came after Angie, but Amy misunderstands and tells him to tell Larry everything . He doesn
't think either of them will believe what he thinks has happened . The mirror writes him a
 Ben fears for Amy and the kids, so he goes over to the house and takes out all of the mirrors . When he can't remove
one from his daughters' room he paints over it . Amy arrives because
 Ben goes back to the store and walks though the basement looking for something . He finds an old sign of the psychiat
ric wing of the hospital that was there before and locates the place where the water in the basement
 Anna Esseker was committed for schizophrenia with delusional and violent behavior . The girl was only 12 years old wh
```

REFERENCES

[1]. Alrumiah, S. S., Al-Shargabi, A. A. Educational Videos Subtitles' Summarization Using Latent Dirichlet

Allocation and Length Enhancement. CMC-Computers, Materials & Continua, 70(3), 6205–6221. (2022)

[2]. Sangwoo Cho, Franck Dernoncourt , Tim Ganter, Trung Bui, Nedim Lipka, Walter Chang, Hailin Jin, Jonathan Brandt, Hassan Foroosh, Fei Liu, "StreamHover: Livestream Transcript Summarization and Annotation" , arXiv : 2109.05160v1 [cs.CL] (11 Sep 2021)

[3]. S. Chopra, M. Auli, and A. M. Rush, ''Abstractive sentence summarization with attentive recurrent neuralnetworks,'' in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol., , pp.93–98. (June 2016).

[4]. Ghadage, Yogita H. and Sushama Shelke. "Speech to text conversion for multilingual languages." International Conference on Communication and Signal Processing (ICCSP) 0236-0240. (2016).

[5]. Pravin Khandare, Sanket Gaikwad, Aditya Kukade, Rohit Panicker, Swaraj Thamke, " Audio Data Summarization system using Natural Language Processing, " International Research Journal of Engineering and Technology (IRJET) Vol. 06, Issue 09,, e-ISSN: 2395-0056; p-ISSN: 2395-0072. 2019.

[6]. Hugo Trinidad and Elisha Votruba, " Abstractive Text Summarization Methods "

[7]. Prof. S. A. Aher , Hajari Ashwini M , Hase Megha S, Jadhav Snehal B, Pawar Snehal S, " Generating Subtitles Automatically For Sound in Videos, " International Journal of Modern Trends in Engineering and Research (IJMTER) Vol. 03, Issue 03, ISSN (Online):2349–9745; ISSN (Print):2393-8161[March – 2016]

[8]. Aiswarya K R, " Automatic Multiple Language Subtitle Generation for Videos, " International Research Journal of Engineering and Technology (IRJET) Vol. 07, Issue 05, , e-ISSN. 2395-0056, p-ISSN: 2395- 0072. [May - 2020].

[9]. Savelieva, Alexandra & Au-Yeung, Bryan & Ramani, Vasanth.. Abstractive Summarization of Spoken and Written Instructions with BERT. (2020).

[10]. Patil, S. et al. "Multilingual Speech and Text Recognition and Translation using Image." International journal of engineering research and technology. (2016).