Natural Language Processing

SEMESTER PROJECT : VIDSUM

Submitted By:

Amogh Garg - 2020UCO1688

Mokshi Sharma - 2020UCO1691

Shirishti Jain - 2020UCO1649

INDEX

PROBLEM STATEMENT

The rise of online meetings, webinars, and video conferences has led to an explosion of digital content. While these meetings are an essential part of modern communication and collaboration, the sheer volume of information they generate can be overwhelming. Also, it becomes very cumbersome for the people who missed the meetings due to prior engagements to watch the entire recordings again. Therefore, some efficient mechanism is needed to accurately generate textual summaries of such webinars/meetings/recordings for the people to reference it later.

The objective of this project is to develop an NLP system that can automatically record online meetings, webinars, and videos and generate a summary of the content in text format. The system should be able to summarize the webinar accurately and efficiently, so that the people who missed the webinar can get a gist of all the important things that happened in the meet. Added to this, those people who did attend the meeting can also refer to the summary some time later in order to review the important things. The system should also be able to identify and highlight any important keywords or phrases that are relevant to the discussion.

Overall, the successful development of such a system that can automatically record and summarize online meetings, webinars, and videos will provide significant benefits to organizations and individuals of all types. This system can help improve productivity, enhance collaboration, and increase efficiency by providing a comprehensive overview of important discussions done during the meeting.

OVERVIEW OF THE APPLICATION

FRONTEND

- Frontend is in the form of a browser extension.
- This extension can be used alongside the online meetings/webinars.
- It will be used to record the meeting and send the recording as input to the backend.
- MediaRecorder API is used to record the media stream for which the summary has to be generated. User needs to click on the Start-Recording button and then share the tab when prompted by the browser.
- SocketIO is used to establish a bidirectional communication between the client (frontend) and the server (backend).
- The chunks of video (entire media divided into small streams called chunks) are transmitted from the client to the server (which is running locally) for processing using SocketIO.
- The chunks of video which are in binary format are continuously appended to the “video.webm” file which is stored on the local computer.

BACKEND

Backend mainly consists of the following aspects:

1. Recognising speech
    - Speech is extracted from the “video.webm” file (which consists of the recorded media in binary format) using FFMPEG. (more details about FFMPEG are given under “Models used” section)
    - FFMPEG converts “video.webm” to “audio.wav” which contains just the audio of the entire media.

2. Converting the speech to text
    - VOSK Speech2Text model is used to convert “audio.wav” into text i.e convert speech to text. (More details about the VOSK Speech2Text model are given under the “Models Used” section).
    - Kaldi submodule under VOSK is used for Automatic Speech Recognition (ASR) from the audio file.
    - Chunks of audio are processed at a time and the results are appended to a string which is finally written to a text file.

3. Adding punctuation to the text file
    - Hugging Face Inference API is used to punctuate the text file generated in the previous step.

4. Generating the summary of the text
    - An Extractive summarization model is developed from scratch to generate the summary of the given input file.
    - 2 models have been developed and the results generated by both of them are compared :
    - FREQUENCY BASED APPROACH : This model takes a text file as input and assigns weights to words based on their importance and subsequently assigns scores to sentences based on the weights of the words contained in it. The sentences with scores above a certain threshold are appended to the summary.
    - TEXT RANK ALGORITHM : This model also takes text files as input and after cleaning the text, it creates embeddings for the words. In accordance with the word embeddings, a similarity matrix of sentences is created. The similarity matrix is then converted into a graph, with sentences as vertices and similarity scores as edges, for sentence rank calculation. Finally, a certain number of top-ranked sentences form the final summary.

      (Detailed explanation of the models is provided under “Models used” section)

ROLE OF NLP

1. Speech Recognition

Speech recognition involves recognising the speech of the humans excluding the background noise and converting the speech into some form understandable by the computer. For this purpose we have used FFMPEG which converts the recorded video file in binary format to audio file which is processed later to generate the summary.

2. Conversion of speech to text

Natural Language Processing (NLP) plays a crucial role in conversion of speech to text. It converts spoken language into text or other computer-readable formats, and NLP is used to analyze and process that text.

In our case we need to convert the speech in the audio file (audio.wav) into text format.

Some of the specific ways that NLP is used in conversion of speech to text are:

1. Phoneme segmentation: NLP algorithms are used to break down spoken words into their individual phonemes, or the smallest units of sound in a language. This allows the system to recognize and differentiate between similar-sounding words.
2. Language modeling: NLP techniques are used to analyze the grammar and syntax of a language, allowing the system to predict the most likely words or phrases based on the context of the spoken language.
3. Acoustic modeling: NLP algorithms are used to analyze the sounds and patterns of speech, allowing the system to differentiate between different speakers and accents, as well as to filter out background noise and other interference.
4. Error correction: NLP techniques are used to identify and correct errors in the speech to text conversion process, such as misinterpreted words or phrases.

3. Adding punctuations to the text

NLP plays a crucial role in adding punctuations to text by using various techniques to analyze the grammatical structure of sentences and identify appropriate punctuation marks. Some techniques are :

1. One common technique used in NLP for punctuation prediction is called part-of-speech (POS) tagging. POS tagging involves analyzing each word in a sentence and assigning it a grammatical category, such as a noun, verb, adjective, or

adverb. Based on the assigned grammatical categories, NLP algorithms can predict the appropriate punctuation marks to use.

2. Another technique used in NLP for punctuation prediction is dependency parsing. Dependency parsing involves analyzing the relationships between words in a sentence to determine their grammatical roles and how they relate to each other. Based on this analysis, NLP algorithms can predict the appropriate punctuation marks to use, such as commas to separate phrases or clauses, or periods to end sentences.
3. NLP can also use machine learning algorithms to predict punctuation marks based on patterns in large amounts of text data. These algorithms can learn from examples of correctly punctuated text and use that knowledge to make accurate predictions for new text. Our system uses Hugging Face Inference API which is build on this concept of predicting punctuation marks based on patterns in large amounts of text data.

4. Summarizing the text

NLP plays a significant role in summarizing text by using various techniques to extract the most important information from a document or a passage and present it in a concise and readable format. The two most common mechanisms for generating summary are explained below :

1. Extractive Summarization : One common technique used in NLP for summarization is called extraction-based summarization. In this technique, the algorithm identifies the most relevant sentences or phrases from the text and creates a summary based on those selections. The algorithm uses various metrics such as sentence relevance, word frequency, and topic modeling to determine which sentences are the most important and relevant to the overall content.
2. Abstractive Summarization : Another technique used in NLP for summarization is called abstraction-based summarization. In this technique, the algorithm creates a summary by generating new sentences that capture the essence of the original text. This method is more complex and challenging than extraction-based summarization, as it requires the algorithm to understand the meaning of the text and generate coherent and grammatically correct sentences.

Our application has developed models to generate summary based on word frequency approach and text rank algorithm under extractive summarization.

PROGRAMMING LANGUAGES AND FRAMEWORKS USED

FRONTEND

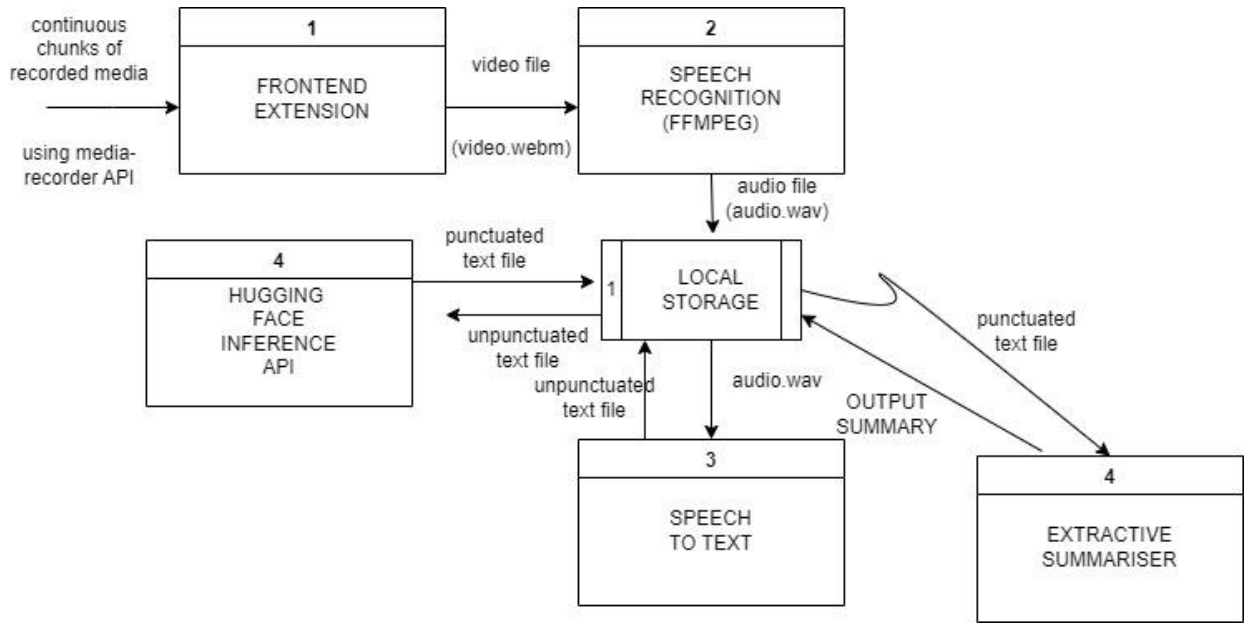
- HTML
- CSS
- JavaScript

BACKEND

- Python
    1. NLTK
    2. SocketIO
    3. Vosk
- Flask

FLOW DIAGRAM

DATA FLOW DIAGRAM



MODELS USED

FFMPEG

- FFmpeg is a free, open-source software project that provides a collection of tools and libraries for handling multimedia files. It is a powerful command-line tool that can be used to convert, edit, and stream audio and video files across a wide range of formats and platforms.
- In this project it is used for video transcoding into audio. Transcoding means to convert audio and video files from one format to another. It supports a wide range of formats, codecs, and containers, making it a versatile tool for media conversion.
- Below command is used to convert video.webm to audio.wav

```
ffmpeg -i video.webm -vn -f wav -ac 1 audio.wav
```

VOSK SPEECH2TEXT

- VOSK Speech2Text is an open-source speech recognition toolkit that is designed to convert speech to text in real-time. It uses deep neural networks to transcribe speech in a wide range of languages, including English, Spanish, French, German, and Russian.
- Kaldi is a free, open-source toolkit for speech recognition that provides a range of tools for developing state-of-the-art speech recognition systems. Some of the reasons why we have used Kaldi API for speech recognition are listed below:
    1. High accuracy: Kaldi has a reputation for providing high accuracy in speech recognition, particularly in challenging acoustic conditions.
    2. Customizable: Kaldi is highly customizable, with a wide range of options for configuring and optimizing speech recognition models to suit specific use cases.
    3. Scalable: Kaldi is designed to be scalable, allowing it to handle large datasets and processing tasks quickly and efficiently.
    4. Extensible: Kaldi is an open-source project with a large community of developers and users, which means it is constantly being updated and improved with new features and capabilities.

5. Integration: Kaldi is designed to be easily integrated with other tools and platforms, such as Python, TensorFlow, and PyTorch

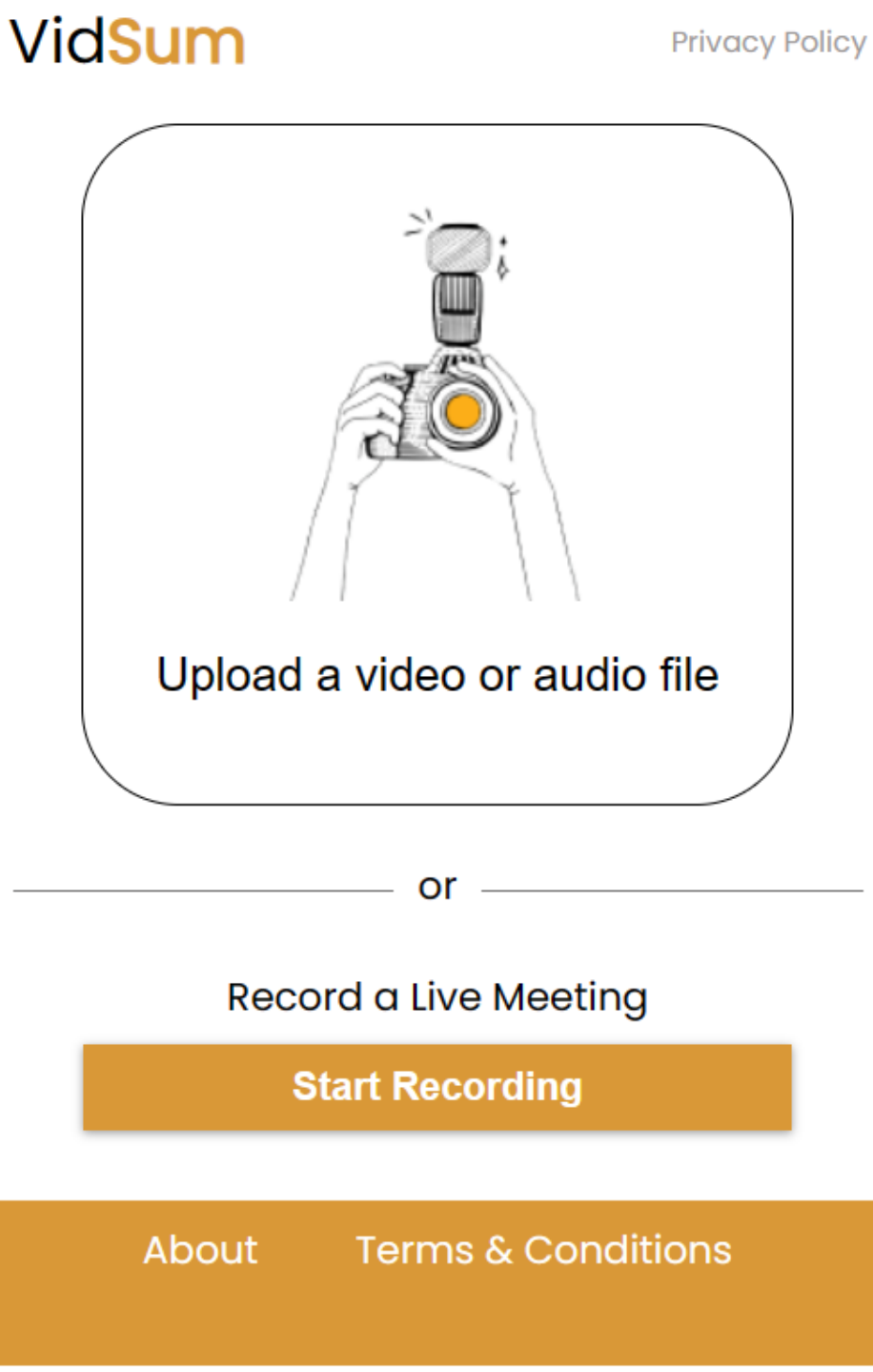
HUGGING FACE INFERENCE API

- Hugging Face inference API is a cloud-based service that provides access to a wide range of natural language processing (NLP) models for various NLP tasks, such as text classification, question-answering, summarization, and more. It is provided by Hugging Face, an open-source community that develops state-of-the-art NLP models and tools.
- The punctuation model of this API can be used to generate predictions for where punctuation should be added to the input text. Once the punctuation predictions have been generated, they can be used to add punctuation to the input text. This involves inserting commas, periods, and other punctuation marks in the appropriate places.

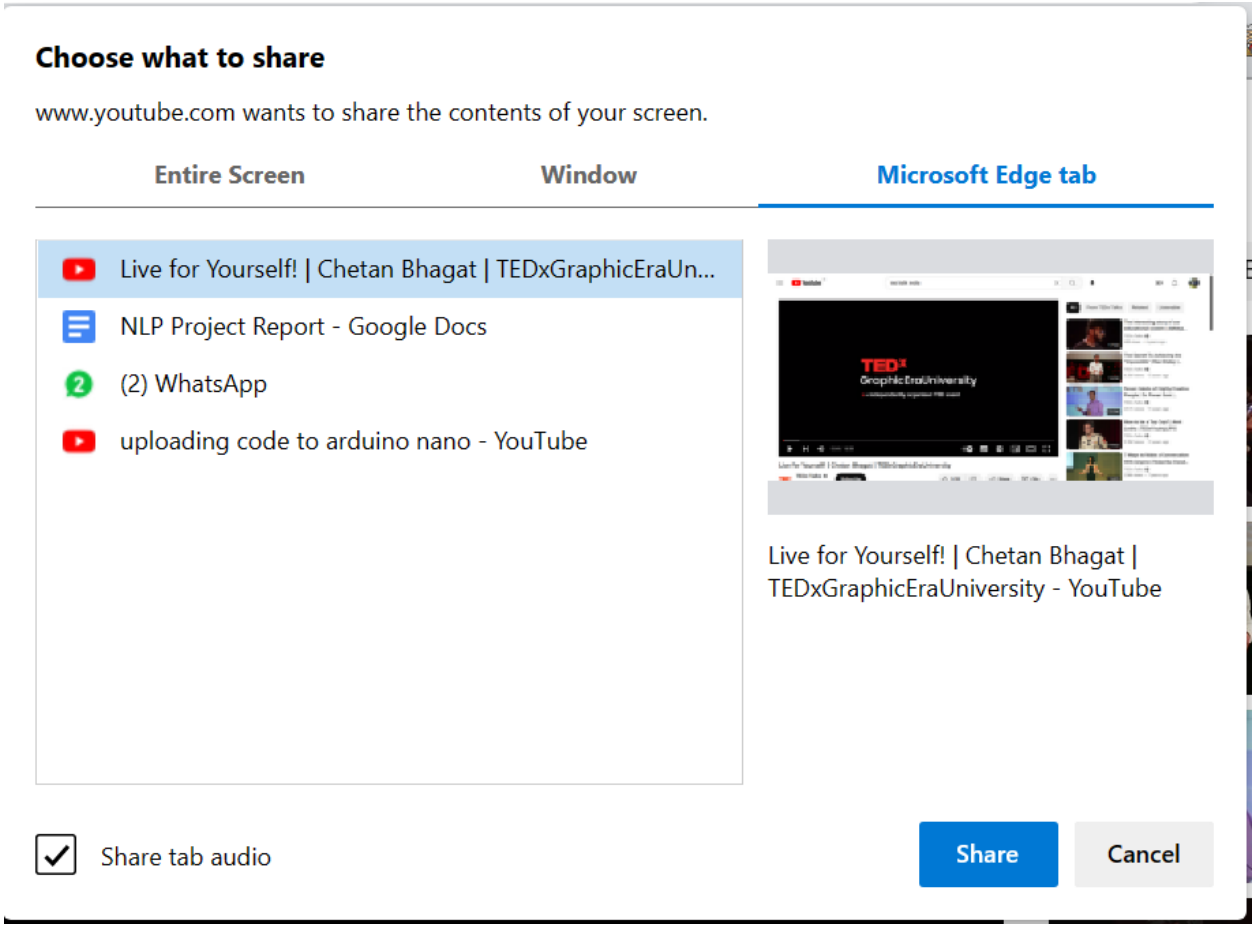
EXTRACTIVE SUMMARIZATION

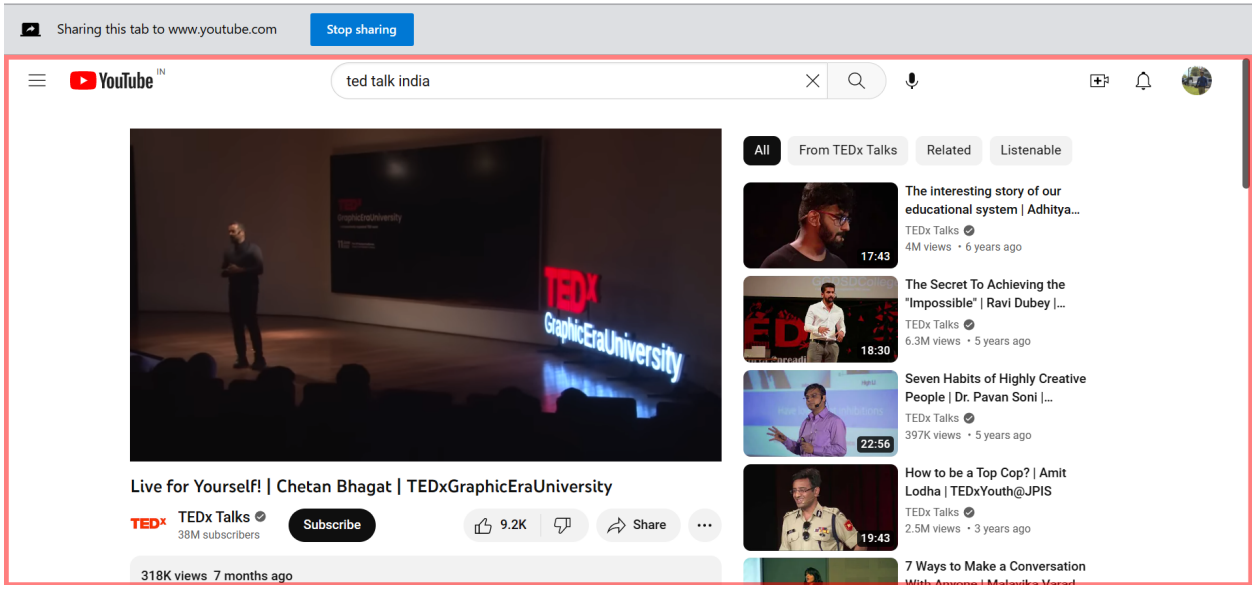
- FREQUENCY BASED APPROACH:
    1. Generate a word frequency hash-map : The frequencies of each word occurring in the text is recorded in a hashmap or dictionary except for the stopwords.
    2. Tokenize the sentence : Single string of input text is broken into a list of sentences using the inbuilt “sent_tokenize()” function.
    3. Using term-frequency method to score sentences : Scoring a sentence according to the words contained in it by adding the frequency of every non-stop word in a sentence. Then to normalize every sentence score is divided by the number of words in the sentence.
    4. Find threshold : Average score of all the sentences is taken as the threshold value and all the sentences above the threshold are considered for summary.
- TEXT RANK ALGORITHM:
    1. After cleaning the input text we will find vector representation (word embeddings) for each and every sentence.
    2. Similarities between sentence vectors are then calculated and stored in a similarity matrix.
    3. The similarity matrix is then converted into a graph, with sentences as vertices and similarity scores as edges, for sentence rank calculation.
    4. Finally, a certain number of top-ranked sentences form the final summary.

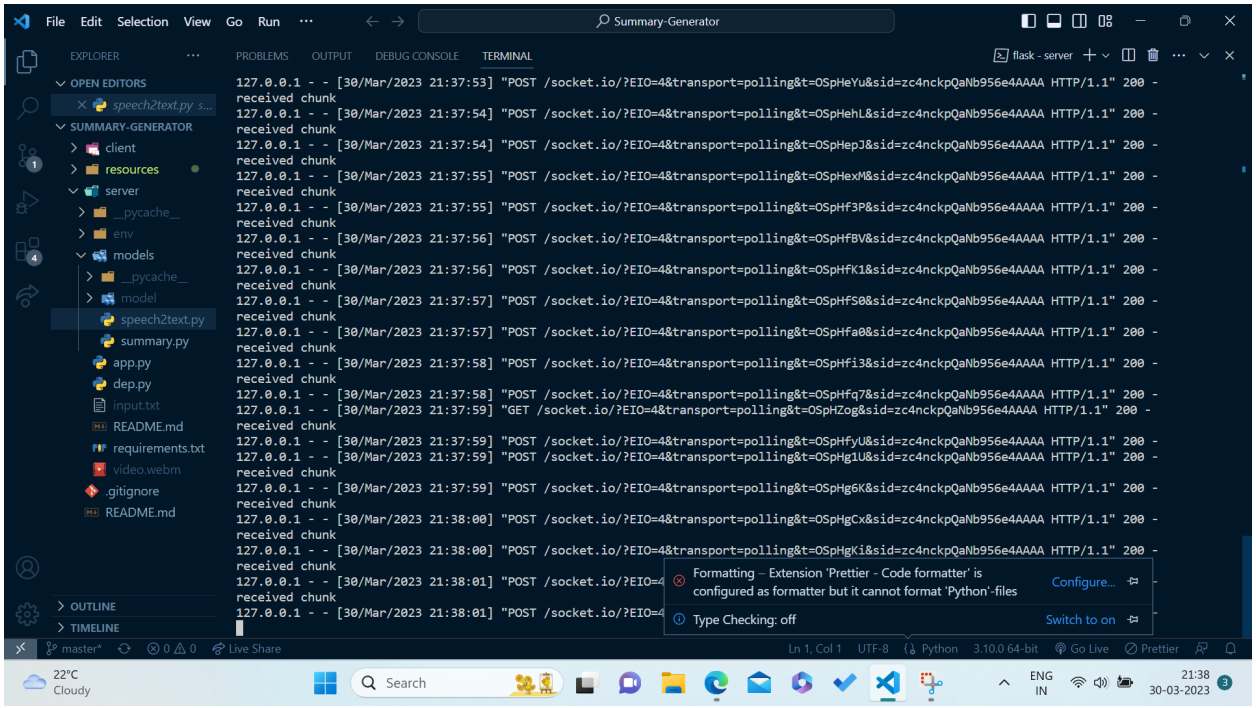
SCREENSHOTS AND OUTPUTS



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS D:\NLP Project\Summary-Generator\server> .\env\Scripts\activate
(env) PS D:\NLP Project\Summary-Generator\server> flask run
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
[nltk_data] Downloading package punkt to C:\Users\AMOGH
[nltk_data]     GARG\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```




Sharing this tab to www.youtube.com
Stop sharing
YouTube IN
ted talk india
All
From TEDx Talks
Related
Listenable
Live for Yourself! | Chetan Bhagat | TEDxGraphicEraUniversity
TEDx Talks
38M subscribers
Subscribe
9.2K
Share
318K views 7 months ago
The interesting story of our educational system | Adhitya...
TEDx Talks
4M views • 6 years ago
17:43
The Secret To Achieving the "Impossible" | Ravi Dubey |...
TEDx Talks
6.3M views • 5 years ago
18:30
Seven Habits of Highly Creative People | Dr. Pavan Soni |...
TEDx Talks
397K views • 5 years ago
22:56
How to be a Top Cop? | Amit Lodha | TEDxYouth@JPIS
TEDx Talks
2.5M views • 3 years ago
19:43
7 Ways to Make a Conversation


File Edit Selection View Go Run
Summary-Generator
EXPLORER
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
flask - server
OPEN EDITORS
SUMMARY-GENERATOR
client
resources
server
__pycache__
env
models
__pycache__
model
speech2text.py
summary.py
app.py
dep.py
input.txt
README.md
requirements.txt
video.webm
.gitignore
README.md
OUTLINE
TIMELINE
127.0.0.1 - - [30/Mar/2023 21:37:53] "POST /socket.io/?EIO=4&transport=polling&t=OSpHeYu&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:54] "POST /socket.io/?EIO=4&transport=polling&t=OSpHehL&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:54] "POST /socket.io/?EIO=4&transport=polling&t=OSpHepJ&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:55] "POST /socket.io/?EIO=4&transport=polling&t=OSpHexM&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:55] "POST /socket.io/?EIO=4&transport=polling&t=OSpHf3P&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:56] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfBV&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:56] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfK1&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:57] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfS0&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:57] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfa0&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:58] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfi3&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:58] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfq7&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
127.0.0.1 - - [30/Mar/2023 21:37:59] "GET /socket.io/?EIO=4&transport=polling&t=OSpHZog&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:59] "POST /socket.io/?EIO=4&transport=polling&t=OSpHfyU&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
127.0.0.1 - - [30/Mar/2023 21:37:59] "POST /socket.io/?EIO=4&transport=polling&t=OSpHg1U&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:37:59] "POST /socket.io/?EIO=4&transport=polling&t=OSpHg6K&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:38:00] "POST /socket.io/?EIO=4&transport=polling&t=OSpHgCx&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:38:00] "POST /socket.io/?EIO=4&transport=polling&t=OSpHgKi&sid=zc4nckpQaNb956e4AAAA HTTP/1.1" 200 -
received chunk
127.0.0.1 - - [30/Mar/2023 21:38:01] "POST /socket.io/?EIO=4
received chunk
127.0.0.1 - - [30/Mar/2023 21:38:01] "POST /socket.io/?EIO=4
Formatting – Extension 'Prettier - Code formatter' is configured as formatter but it cannot format 'Python'-files
Configure...
Type Checking: off
Switch to on
master*
Live Share
Ln 1, Col 1
UTF-8
Python
3.10.0 64-bit
Go Live
Prettier
22°C
Cloudy
Search
ENG
IN
21:38
30-03-2023

VidSum

Privacy Policy

Upload a video or audio file

or

Record a Live Meeting

Generate Summary

About Terms & Conditions

TESTING THE EXTRACTIVE SUMMARIZATION MODEL:

INPUT TEXT:

Natural Language Processing (NLP) is a field of study that deals with the interaction between computers and human languages. It involves the use of computer algorithms to process, analyze, and understand natural language data. NLP is an interdisciplinary field that draws from computer science, linguistics, mathematics, and psychology. The main goal of NLP is to create machines that can understand and interpret human language, allowing for more efficient communication and data processing.NLP is a complex and rapidly growing field that has many practical applications in various industries. Some of the most common applications of NLP include machine translation, sentiment analysis, speech recognition, chatbots, and text classification. Machine translation involves the automatic translation of text from one language to another, while sentiment analysis involves the analysis of text to determine the sentiment expressed by the author. Speech recognition involves the conversion of spoken language into text, while chatbots are computer programs that can simulate human conversation. Text classification involves the automatic classification of text into different categories based on its content.One of the biggest challenges in NLP is the ambiguity of natural language. Natural language is full of ambiguity, and words can have multiple meanings depending on the context in which they are used. For example, the word "bank" can refer to a financial institution or the side of a river. Resolving these ambiguities requires sophisticated algorithms that can take into account the context and other relevant factors. Another challenge in NLP is dealing with the vast amount of data that is available. The internet has made an enormous amount of text data available, and processing this data requires powerful computing resources and sophisticated algorithms. Many NLP algorithms use machine learning techniques to learn from the data and improve their performance over time.NLP has made significant progress in recent years, and many NLP applications are now widely used in various industries. For example, machine translation services like Google Translate have made it possible for people to communicate across language barriers, while sentiment analysis tools are used to analyze customer feedback and improve product development. Chatbots are becoming increasingly popular in customer service, while text classification algorithms are used in a variety of applications, including spam filtering and content moderation.Despite the progress made in NLP, there are still many challenges that need to be addressed. One of the biggest challenges is improving the accuracy of NLP algorithms, especially in dealing with the nuances of natural language. Another challenge is developing algorithms that can handle multiple languages and dialects, as well as different writing styles and formats.In conclusion, NLP is a fascinating and rapidly evolving field that has the potential to revolutionize the way we interact with computers and process data. NLP algorithms are already widely used in various industries, and their use is expected to grow in the coming years. While there are still many challenges to be addressed, the future of NLP looks

promising, and we can expect to see many exciting developments in the field in the years to come.

OUTPUT GENERATED USING FREQUENCY BASED APPROACH:

```
========== RESTART: D:\NLP Project\Summary-Generator\server\models\summary.py ==========
[nltk_data] Downloading package punkt to C:\Users\AMOGH
[nltk_data]     GARG\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
 It involves the use of computer algorithms to process, analyze, and understand natural language data. NLP is an interdisciplinary field that draws from computer science, linguistics, mathematics, and psychology. Some of the most common applications of NLP include machine translation, sentiment analysis, speech recognition, chatbots, and text classification. Machine translation involves the automatic translation of text from one language to another, while sentiment analysis involves the analysis of text to determine the sentiment expressed by the author. Speech recognition involves the conversion of spoken language into text, while chatbots are computer programs that can simulate human conversation. Natural language is full of ambiguity, and words can have multiple meanings depending on the context in which they are used. Another challenge in NLP is dealing with the vast amount of data that is available. One of the biggest challenges is improving the accuracy of NLP algorithms, especially in dealing with the nuances of natural language. NLP algorithms are already widely used in various industries, and their use is expected to grow in the coming years.
```

It involves the use of computer algorithms to process, analyze, and understand natural language data. NLP is an interdisciplinary field that draws from computer science, linguistics, mathematics, and psychology. Some of the most common applications of NLP include machine translation, sentiment analysis, speech recognition, chatbots, and text classification. Machine translation involves the automatic translation of text from one language to another, while sentiment analysis involves the analysis of text to determine the sentiment expressed by the author. Speech recognition involves the conversion of spoken language into text, while chatbots are computer programs that can simulate human conversation. Natural language is full of ambiguity, and words can have multiple meanings depending on the context in which they are used. Another challenge in NLP is dealing with the vast amount of data that is available. One of the biggest challenges is improving the accuracy of NLP algorithms, especially in dealing with the nuances of natural language. NLP algorithms are already widely used in various industries, and their use is expected to grow in the coming years.

OUTPUT GENERATED USING TEXT RANK ALGORITHM:

```
========== RESTART: D:\NLP Project\Summary-Generator\server\models\summary_text_rank.py ==========
[nltk_data] Downloading package stopwords to C:\Users\AMOGH
[nltk_data]     GARG\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!

Warning (from warnings module):
  File "D:\NLP Project\Summary-Generator\server\models\summary_text_rank.py", line 17
    clean_sentences = pd.Series(sentences).str.replace("[^a-zA-Z]", " ")
FutureWarning: The default value of regex will change from True to False in a future version.
It involves the use of computer algorithms to process, analyze, and understand natural language data.The main goal of NLP is to create machines that can understand and interpret human language, allowing for more efficient communication and data processing.NLP is a complex and rapidly growing field that has many practical applications in various industries.Text classification involves the automatic classification of text into different categories based on its content.One of the biggest challenges in NLP is the ambiguity of natural language.The internet has made an enormous amount of text data available, and processing this data requires powerful computing resources and sophisticated algorithms.Many NLP algorithms use machine learning techniques to learn from the data and improve their performance over time.NLP has made significant progress in recent years, and many NLP applications are now widely used in various industries.For example, machine translation services like Google Translate have made it possible for people to communicate across language barriers, while sentiment analysis tools are used to analyze customer feedback and improve product development.Chatbots are becoming increasingly popular in customer service, while text classification algorithms are used in a variety of applications, including spam filtering and content moderation.Despite the progress made in NLP, there are still many challenges that need to be addressed.One of the biggest challenges is improving the accuracy of NLP algorithms, especially in dealing with the nuances of natural language.Another challenge is developing algorithms that can handle multiple languages and dialects, as well as different writing styles and formats.In conclusion, NLP is a fascinating and rapidly evolving field that has the potential to revolutionize the way we interact with computers and process data.
```

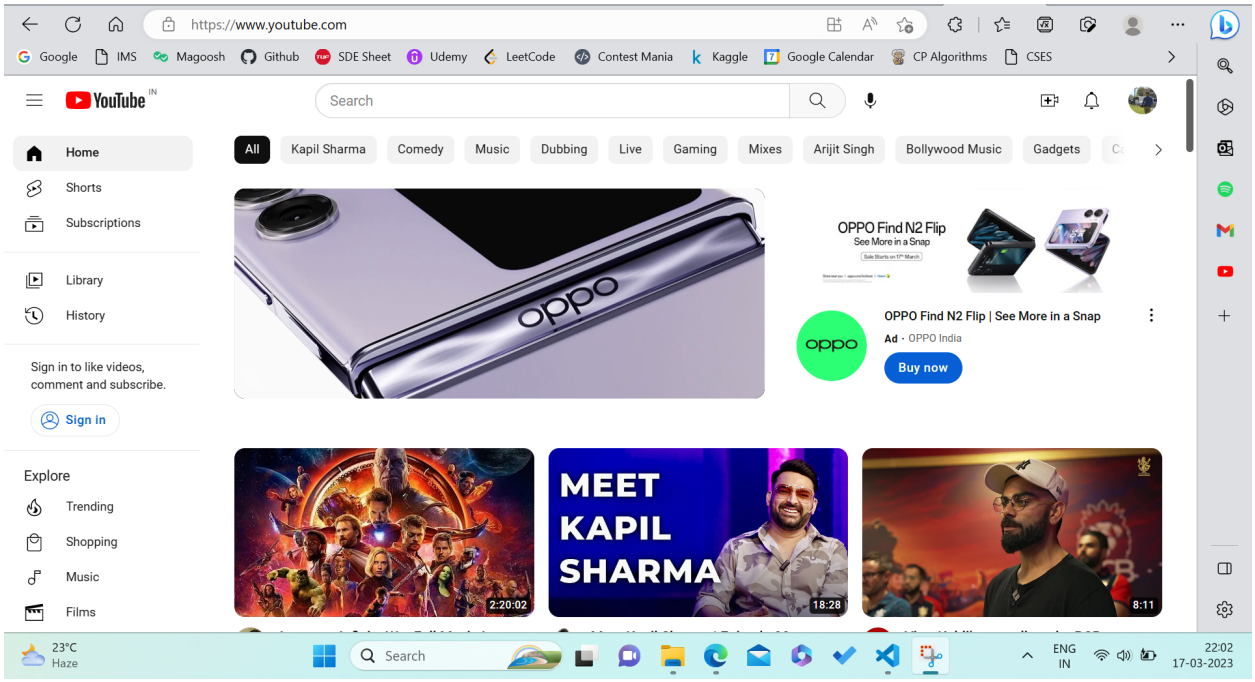
It involves the use of computer algorithms to process, analyze, and understand natural language data.The main goal of NLP is to create machines that can understand and interpret human language, allowing for more efficient communication and data processing.NLP is a complex and rapidly growing field that has many practical applications in various industries.Text classification involves the automatic classification of text into different categories based on its content.One of the biggest challenges in NLP is the ambiguity of natural language.The internet has made an enormous amount of text data

available, and processing this data requires powerful computing resources and sophisticated algorithms.Many NLP algorithms use machine learning techniques to learn from the data and improve their performance over time.NLP has made significant progress in recent years, and many NLP applications are now widely used in various industries.For example, machine translation services like Google Translate have made it possible for people to communicate across language barriers, while sentiment analysis tools are used to analyze customer feedback and improve product development.Chatbots are becoming increasingly popular in customer service, while text classification algorithms are used in a variety of applications, including spam filtering and content moderation.Despite the progress made in NLP, there are still many challenges that need to be addressed.One of the biggest challenges is improving the accuracy of NLP algorithms, especially in dealing with the nuances of natural language.Another challenge is developing algorithms that can handle multiple languages and dialects, as well as different writing styles and formats.In conclusion, NLP is a fascinating and rapidly evolving field that has the potential to revolutionize the way we interact with computers and process data.

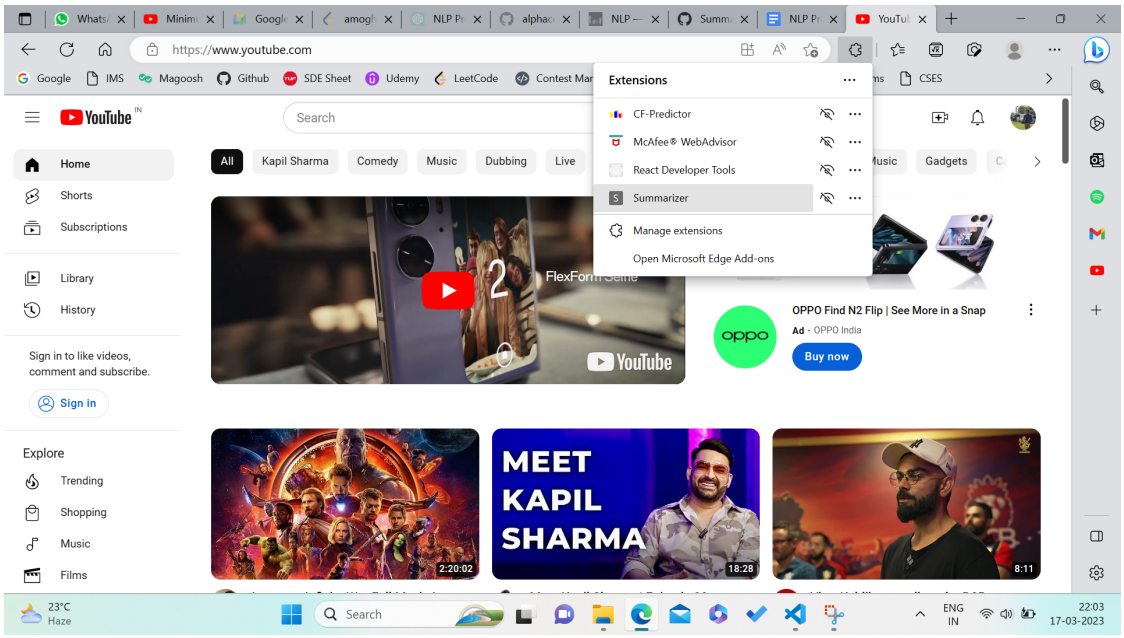
USER MANUAL

After setting up the environment as directed here - CODE, follow the given steps to generate the summary of any video :

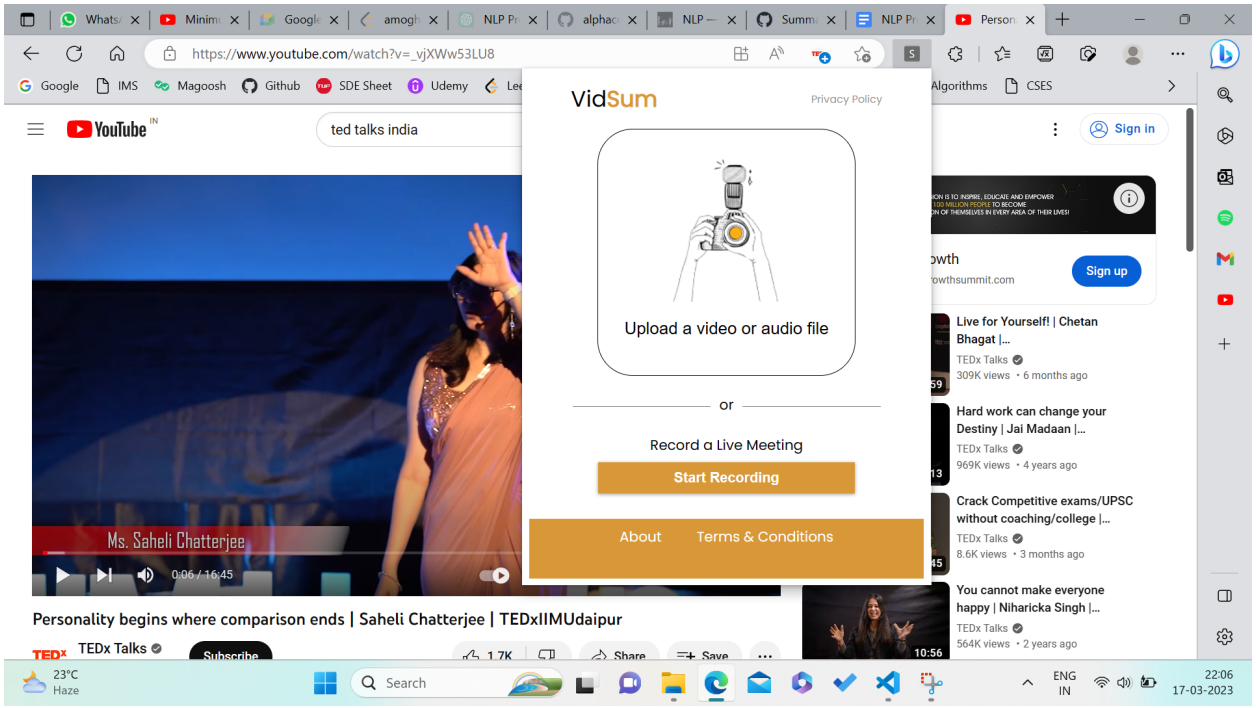
1. Open the tab in which the desired meet/webinar/video is being played.



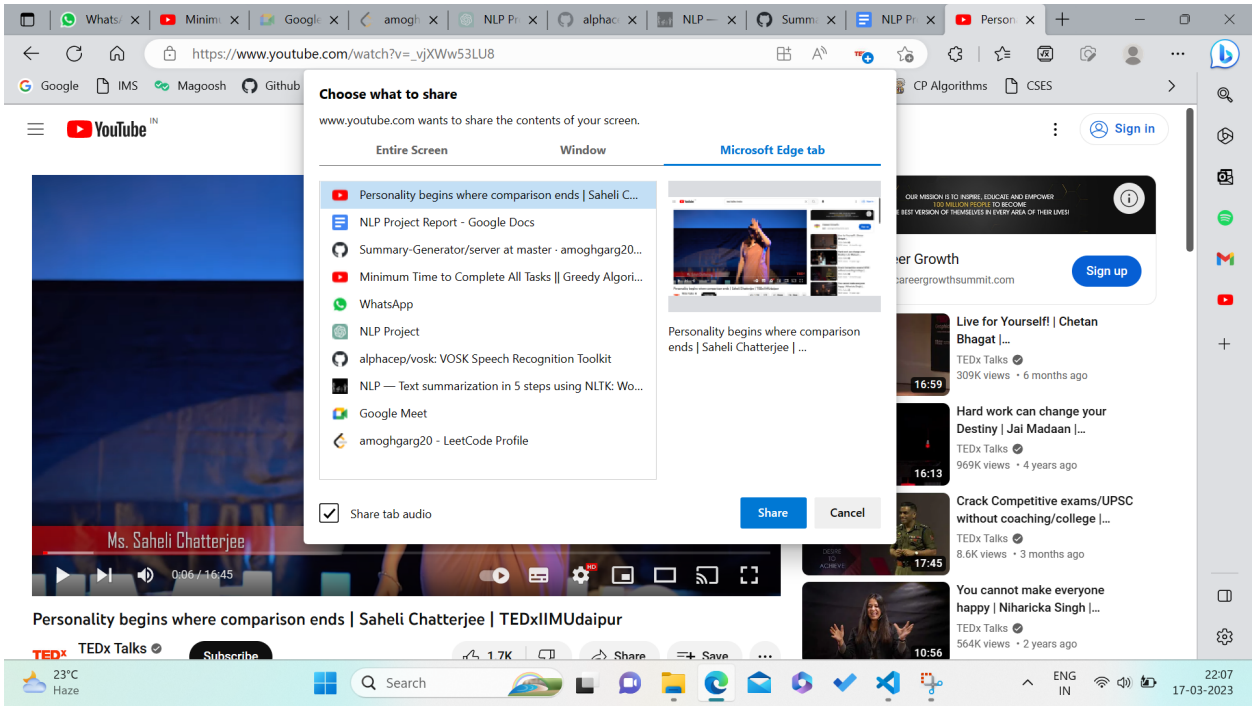
2. From the extensions menu select Vidsum - summariser.



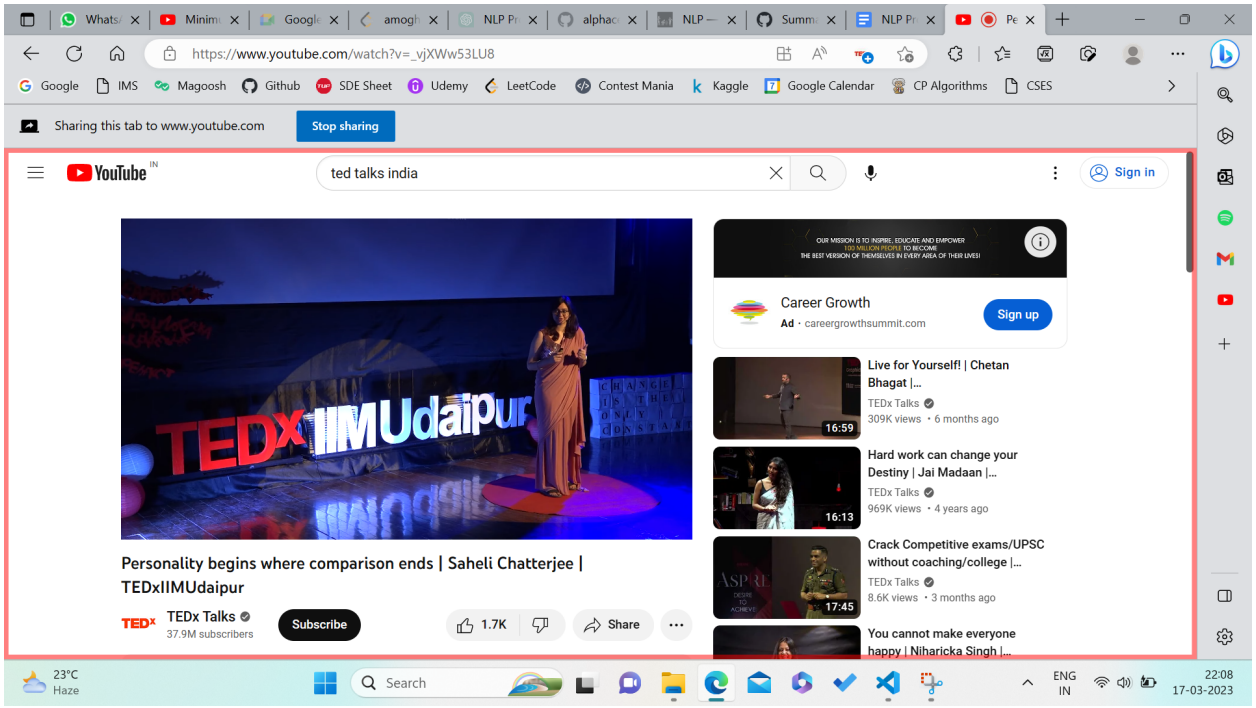
3. Click on the “Start Recording” button. (ensure that the server is running locally)



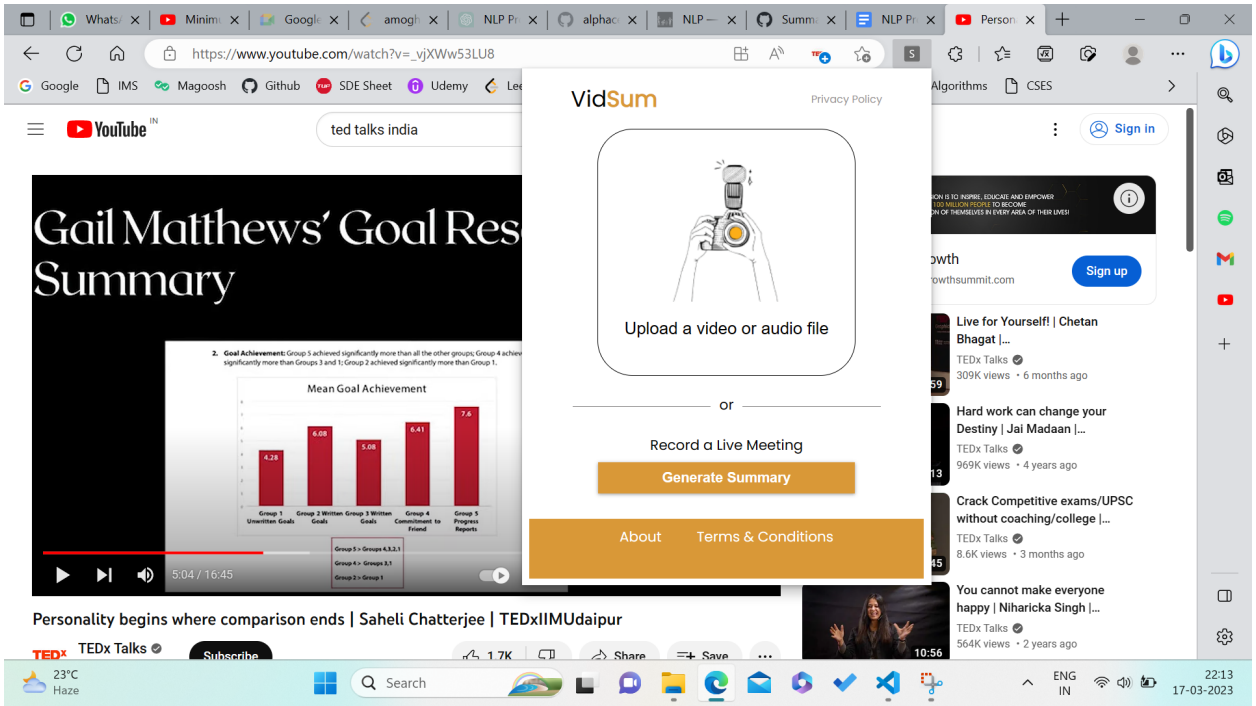
4. Select the desired tab from tabs menu only (otherwise media won’t be recorded).



5. When you want to stop the recording, click on the “Stop sharing tab” option.



6. Open the extension and click on the “Generate Summary” button to generate the summary.



7. Summary can be viewed in the server logs.

SCOPE OF IMPROVEMENT

1. The system can be tested with other summarization techniques like abstractive summarization to see whether there is improvement in performance or not.
2. In future, functionality of providing summary in the form of video clips from the original video recording can be worked upon.
3. Research upon overcoming limitations of Hugging Face Inference API like latency and recording for limited time can be done.
4. A Feature to upload recordings from localhost can also be added.

CODE

The entire code along with the steps to setup the environment can be found on the following link : [amoghgarg20/Summary-Generator (github.com)]

Clone the repository on the local system and then follow the steps given in the **Readme.md** file to setup the environment and test the system.