



Keyword Suggestion tool with Transcript Summarization.

Under the Guidance of :-

Prof. V. B. Kamble

09th July, 2022.

Avdhut Kamble

194033

DCP-TY CWIT, Pune.

ABSTRACT

Due to the Pandemic, the majority of Businesses and Educational institutes have switched over to online teaching and online meetings. Due to Network issues, device compatibility and server-side problems the students attending online classes are affected very badly, moreover the business meetings are also not attended by many professionals, and due to this they face many problems.

'Transcribo' project is created to solve this problem. It transcribes the Online video lectures/meetings to produce their summary/report resp., and also adds the highlights/topics to the content present, with features which add to the productivity of the peoples working online!

This project was developed in Summer 2022 for Python Programming Course, using Python as a Backend Language and HTML & CSS is used for Frontend Development.

Introduction

The project 'Transcribo' was developed during Summer of 2022, for Python Programming Course at CWIT, Pune.

This project is developed to solve and reduce the problems to some extent, faced by the students and all online medium users during Meetings or online teachings.

With the help of this project anyone can get the transcribed file which is not limited to summary only, but also highlights the main key points of the long transcription.

It uses the audio or video format file as an input and produces the transcript file as an output. For further, clarity and wider applicability and use we have given it a form of a Web application, so that it doesn't get a limited to computer or Mobile users.

Objectives

The objective of this project developed is to provide the full transcribed file of audio/video source along with the highlighted key words, so as to grasp the transcription in very short amount of time and henceforth help all the online platform users and specially able's too, to get more soother experience of online meeting & classroom platforms.

System Design

The System design consists of Python as backend, HTML/CSS for frontend and FLASK as web framework.

The Following Pythons libraries were used:-

- Flask
- NLTK
- SCIKIT-LEARN
- GENSIM
- MOVIEPY
- SPEECH-RECOGNITION
- WAVE
- MATH
- CONTEXTLIB

The following technologies were used:

- TfidfVectorizer
- Gensim's Text Summarization tool for keywords
- Speech to text By Google
- Natural Language Processing for the whole block

System Implementation

Firstly the project is based on Flask, so all the data dealing is done by the framework smoothly. The first real task of the program is to convert the video file provided to an .wav audio file for transcription. We did this using a package called moviepy. Then the output audio clip was gone through Google's Speech recognition API in a loop to cover the whole audio.

The transcription from the API was saved in a local .txt file for better use. We tried to append the whole transcription to a single string, but we got some runtime issues (Basically out of RAM).

Though Google API is one of the best speech Recognition API out there, it doesn't punctuate the text in the output. Thus, we came across a pre-trained punctuation model based on theano and a package called "punctuator". We punctuated our text with the model itself.

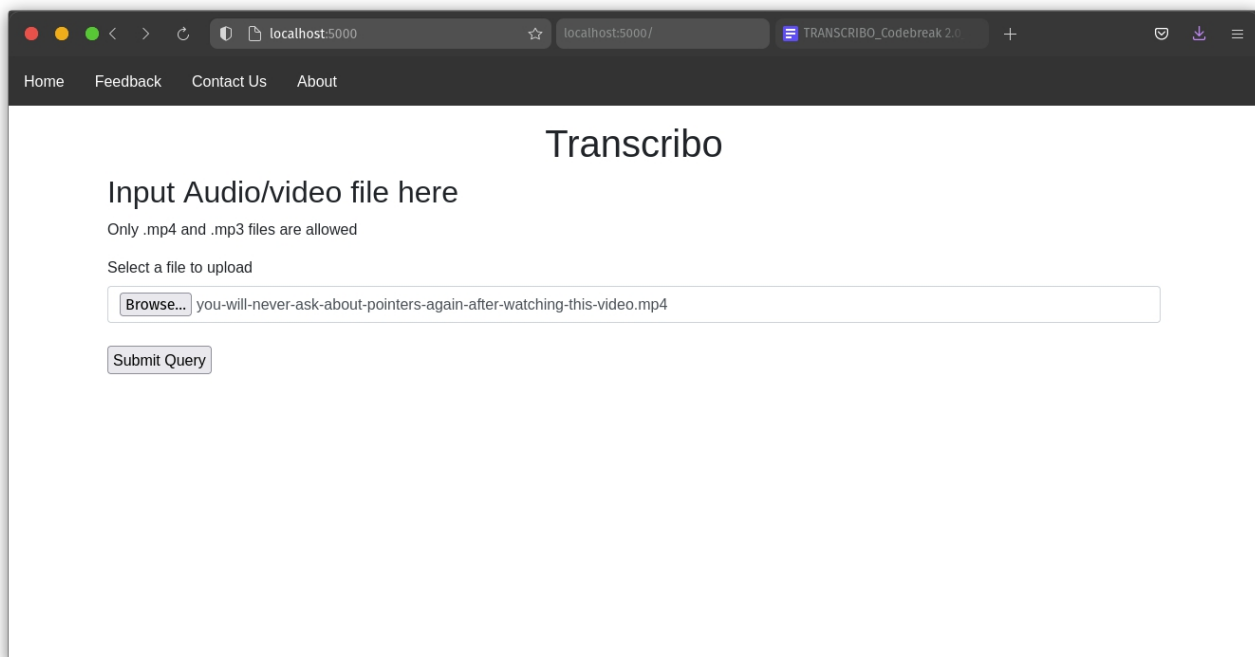
The summarization process: For getting the summary out of the text, we went through several options, and chose Scikit learn's tfidfvectorizer. The process seems to be complicated but was quite easy. First, dividing the text with sentences with the help of punctuations, then tokenize each word with the help of nltk's word_tokenize. Then getting the average of frequency of words and removing stopwords/regular words from the text. Then calculating the importance or Accuracy of the words in the sentence. Then with that accuracy, calculate which sentences are more valuable.

Lastly, adding the sentences with the highest threshold to the cut of final summary. And all the data we obtained from the following system is given out as html with the help of Flask!

Implementation:

```
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
justafolk@redhat ~-> cd Transcribo/flask/app/
justafolk@redhat ~/T/flask app (main)> FLASK_APP="./final.py" flask run
* Serving Flask app './final.py' (lazy loading)
* 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 stopwords to
[nltk_data]   /home/justafolk/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data]   /home/justafolk/nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]   /home/justafolk/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Imports Done
flaskwebgui - [INFO] - * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
```

Fig 1.1 Initialization of Flask Web Application.



Home Feedback Contact Us About

Transcribo

Input Audio/video file here

Only .mp4 and .mp3 files are allowed

Select a file to upload

you-will-never-ask-about-pointers-again-after-watching-this-video.mp4

Fig 1.2 Stage 1: Uploading your source of data (Video / Audio).

```

* Serving Flask app './final.py' (lazy loading)
* 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 stopwords to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package omw-1.4 is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package wordnet is already up-to-date!
Imports Done
flaskwebgui - [INFO] - * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:24] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:25] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:26] "GET /favicon.ico HTTP/1.1" 404 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:33] "GET / HTTP/1.1" 200 -
MoviePy - Writing audio in file.wav
chunk: 65% | 7873/12112 [00:03<00:01, 2635.63it/s, now=None]

```

Fig 1.3. Stage 2 : Converting the Source to Audio wave format.

```

* 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 stopwords to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package omw-1.4 is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package wordnet is already up-to-date!
Imports Done
flaskwebgui - [INFO] - * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:24] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:25] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:26] "GET /favicon.ico HTTP/1.1" 404 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:33] "GET / HTTP/1.1" 200 -
MoviePy - Writing audio in file.wav
MoviePy - Done.
IncrementalBar |### | 1/10

```

Fig 1.4. Stage 3 : Incrementally transcribing audio via Google Speech Recognition API.

```

* Debug mode: off
[nltk_data] Downloading package stopwords to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package omw-1.4 is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data] /home/justafolk/nltk_data...
[nltk_data] Package wordnet is already up-to-date!
Imports Done
flaskwebgui - [INFO] - * Running on http://127.0.0.1:5000 (Press CTRL+C to quit)
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:24] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:25] "GET / HTTP/1.1" 200 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:26] "GET /favicon.ico HTTP/1.1" 404 -
flaskwebgui - [INFO] - 127.0.0.1 - - [10/Jul/2022 15:54:33] "GET / HTTP/1.1" 200 -
MoviePy - Writing audio in file.wav
MoviePy - Done.
IncrementalBar |#####| 10/10
Recognition Done
Punctuating Done

```

Fig 1.5. Stage 4 : After the recognition is done, we punctuate the transcription and initialize the summarization process.

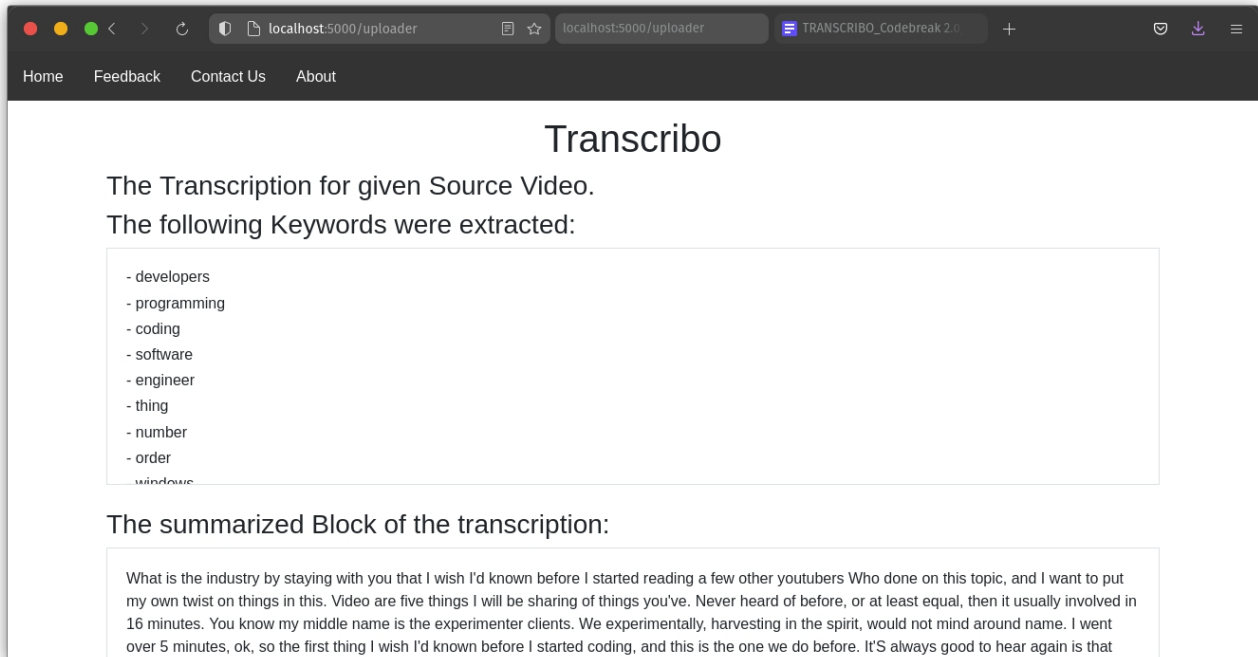


Fig 1.6. Stage 5 : The results are returned to the web application with Keywords, transcript and summarization.

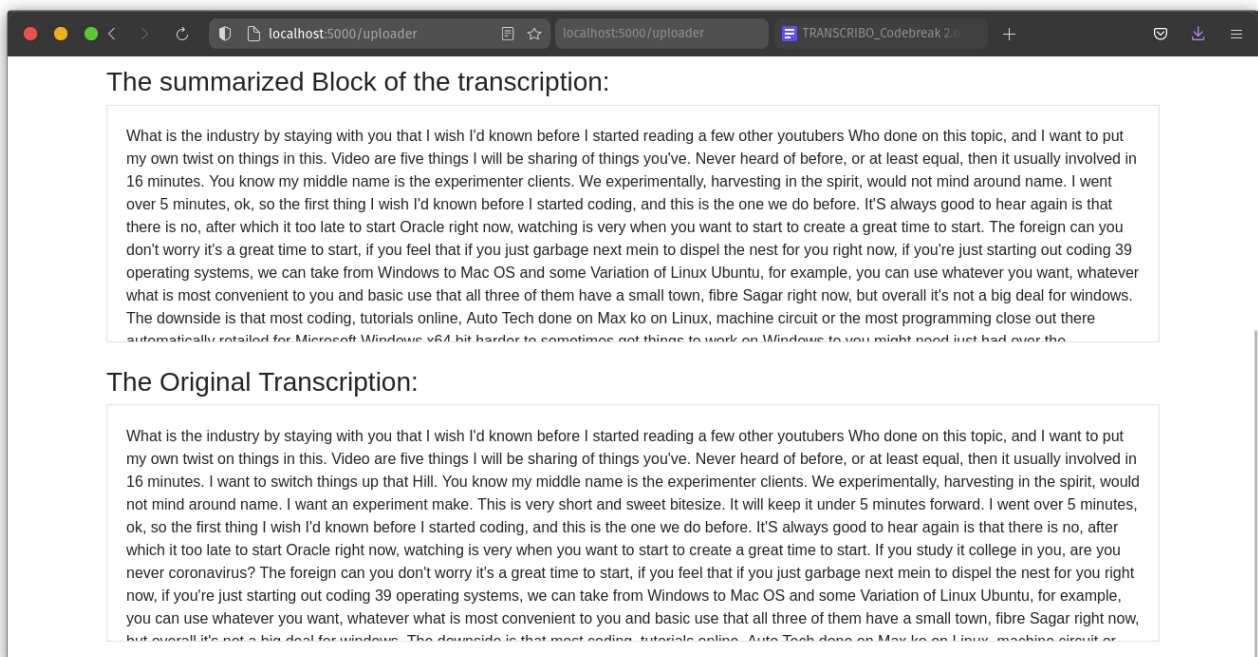


Fig 1.7. Stage 5 : Transcript and Summarization.

Applications

This project has wide range of applications, some of which are listed below:-

- For School/College Students, to get notes, if they missed any class by chance.
- For School/College Students, to get notes with highlighted keywords, to revise the class during examinations.
- For Business professionals to get the minutes of meeting.
- For all the online events and workshops, which happens online, and after that report is to be prepared.
- To know the name of medicines and brief of call, when the doctor consults the patients online using video/audio call.
- To store the financial based startup's (like policy bazaar) call data in transcribed files, for future evidence and reference.
- For all Specially abled who can't hear the audio/video can see the transcription.

Conclusion

This project has many applications and benefits to humans across the globe with its specially designed features and platform independent service, as it is a web application.

The future of the world is Digital in nature, and this project can prove to be the one of the most important and needed to all the users who are digitally connected to the virtual world, while providing virtual services, can use it without any technical knowledge, due to its simple and easy to use operating features.

Within a few years, most of the Corporate Business will get shifted to online platforms, where this project will play a very key role in each and every meeting of theirs.

It will not get limited to the Corporate world, but also prove to be efficient in Educational Institutes as they will also start to use digital platforms for teaching.

References

- [W3Schools Online Web Tutorials](#)
- [Scikit-Learn's Official Documentation](#)
- [Gensim's Text Summarization](#)
- [Speech recognition Guide realpy](#)