SoC Summer 2021 Final Documentation

**** This will be considered as your final documentation and will be made available for online usage.***

UnScripted

Annirudh Mittal, Siddesh Pawar

Keywords (Include 7 or more keywords that will help others find your documentation easily)

Speech recognition, STT, ML, NLP, Python, Unscripted, Journalism, Meeting Notes, Article

Team Member Name	Roll Number	Email-Id
Hiren Bavaskar Harshit Modi	190050049 190100075	190050049@iitb.ac.in 190100075@iitb.ac.in

Brief Description

The project was aimed at making a pipeline which could transcribe the speech of a journalist to text and then extract important sentences from the raw transcript and form a summary or journal/article out of it. The code uses Google Speech Recognition library for the Speech to Text part. It subsequently breaks the audio into chunks and then transcribes it part by part. Further, the code creates an array of sentences and then uses the NLTK library for summarising. Sentence similarity is checked based on which the final article is given as output. As of now, the code gives users the liberty to choose the size of the article.

Progress

Describe how the work was done
Work-Flow distributed across the duration between the review meets
Challenges (Difficulties faced and how you overcame it)
Calculations involved
(min.300 words)

Initially, for the learning part, we learned about the basics of python and Machine learning. Various ML models and their uses. Then, we learnt about NLP and its uses. Moving further, each one of the team members examined various different speech to text open source models like DeepSpeech, IBM Watson, Google Speech recognition, Wav2Vec2. We used WER(Word error rate), Rogue and Bleu error metrics to compare the performance of these pre-trained models and then choose the best out of it. We concluded to use Google Speech Recognition for our purpose. Then, we made various attempts to process the transcript text and finally using NLTK, we were able to come up with better results.

1st - 2nd meet : Basic library learning like Numpy, Pandas and using git
2nd - 3rd meet : Learning basics of ML and python
3rd - 4th meet : Speech to text model analysis
4th - 5th meet : Distributing in subteams for Journalism, meeting minutes, movie dialogues
5th - 6th meet : Learning transformers and named entity recognition

6th - 7th meet : Writing down code for STT and text cleaning

Challenges:

1. *Reduce the error for STT model.*
2. *Fine tune the STT model for Indian accent*
3. *Find the corresponding relevant dataset for fine-tuning*
4. *Removal of gibberish and extra characters in the transcribed text*
5. *Grammatically correcting the transcribed text*
6. *Group the similar sentences based on the context of the transcribed text*
7. *Write an extractive summary of the processed transcribed text*
8. *Format the text neatly*

Solutions used:

By selecting the best STT model, we could reduce the speech to text errors. Fine tuning the model for Indian accent proved to be difficult because the datasets are scarcely available on the internet. Extra spaces and special characters were removed by the use of inbuilt text features of the library. Sentence similarity was used to filter out important and relevant sentences out of transcription for summary.

Results

A detailed description of the results you have achieved

Include:

Images of your working prototype

GitHub Repository (if any)

All files (documents or sheets) used (eg.meeting notes)

Simulations/ Project Video

Final Presentation Link

Add these as hyperlinks with a proper name for the field

Github repo: https://github.com/Hiren6/Unscripted_Journalism

Learning Value

This project has enhanced our understanding of the ASR system. Along with it, the project has given us a glimpse that there is still a lot to do for a speech to text model to work accurately. Also, while working in a team, we enjoyed coding in collaboration and discussed possible ideas together.

Software used

Github
Google Collaboratory
Jupyter Notebook
Python and libraries:

- *SpeechRecognition*
- *NLTK*
- *Sys*
- *Pydub*

Suggestions for others

Include what you would like to share to someone who starts with this project

We would recommend that learning NLP, ML would be helpful before starting. Basic knowledge of python would be helpful. For further improvements to our project, we would recommend adding features like speech to text for different languages and fine tuning it for various accents for better transcription.

References and Citations

CITE ALL THE RESEARCH PAPERS AND AUTHORS (Whatever is demanded by the paper) and also add the link to it
Mention all GitHub repositories used for reference. If you have used code from other GitHub repositories make sure you mention and cite the authors properly and INCLUDE THE GITHUB LICENSE in the disclaimer section.

References used:
Speech to text code and tutorial
Text summarizer resource
Author of text summarizer github repo (Author provided the sample code and tutorial as a Medium article)

Disclaimer

Fair Use Disclaimer: Include “Fair use of … “ wherever necessary (instances where you have directly used materials or taken words from other sources)
Copyright Disclaimer: Make sure it is well within the limits of use allowed in their respective copyright policies.

Licenses

Include licenses of all the software and tools (if it is free to use, open-source, proprietary, paid) used and mention GitHub licenses

Google Colab and Jupyter Notebooks are free to use

Python - FOSS

GitHub, Git - Free, FOSS

Python and libraries:

- *SpeechRecognition - Free*
- *NLTK - Free*
- *Sys - Free*
- *Pydub - Free*