

# UE20CS390B - Capstone Project Phase - 2

## Project Progress Review # 1

Project Title : FeelSpeak: Generating Emotional Speech with Deep Learning  
Project ID : PW23\_VRB\_07  
Project Guide : Prof. V R Badri Prasad  
Project Team : 235\_320\_345\_362

## Outline

---

- Abstract and Scope of the Project
- Capstone Project Phase - 1
  - Summary of work
  - Inferences drawn from Literature Survey
- List of Tasks/Modules with Individual Contribution
- Design of overall Architecture
- Demonstration and Testing of the completed modules.
- Gantt chart

## Abstract and Scope

---

### Abstract:

- This project aims to develop a system that can generate emotional speech from given input text.
- The system will identify the emotions expressed in the text and generate speech with appropriate prosodic features to convey those emotions effectively.
- The project involves natural language processing, speech synthesis, and emotion recognition tasks.

### Scope:

- The scope of this project includes developing a system that can accurately identify the emotional content of given input text and generate speech with appropriate prosodic features to reflect those emotions.
- The system will involve various tasks such as NLP and text emotion detection, speech synthesis, and emotion recognition.
- The NLP and text emotion detection component will parse the input text to identify its structure, meaning, and emotional content.
- The speech synthesis component will generate speech that accurately reflects the emotions expressed in the input text, and the emotion recognition component will map the emotional content of the input text to appropriate prosodic features.

## Summary of Work Done in Capstone Project Phase - 1

---

### Provide summary of Phase - 1

- The project team developed the architecture of the system, which defines the overall structure and components of the system. They also developed the algorithms that will be used to process the speech data, as well as the tools that will be used to collect and manage the data.
- The team conducted a literature survey to learn about the latest research in speech emotion recognition. Also reviewed existing methodologies for training and evaluating speech emotion recognition models.
- The team identified a speech dataset that contains a variety of emotions.

## Summary of Work Done in Capstone Project Phase - 1

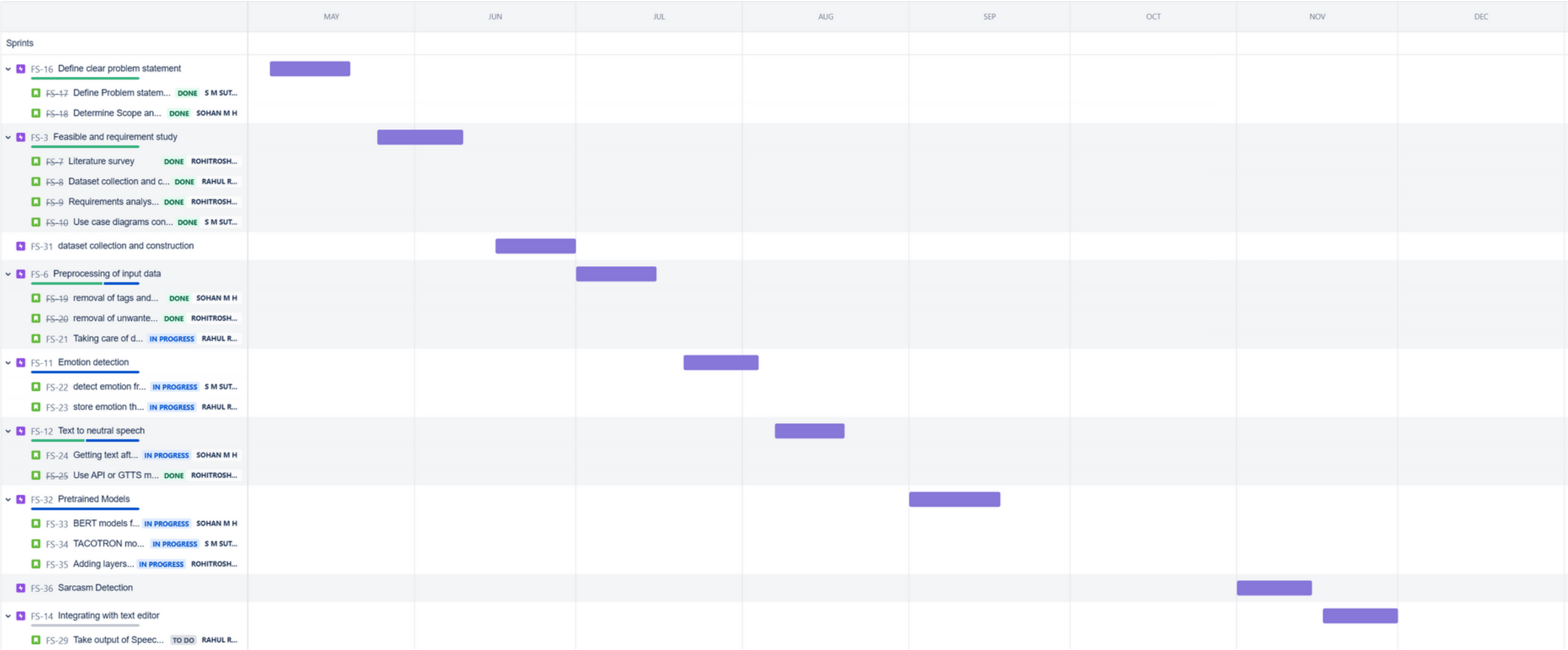
---

Include the suggestions and improvements made.

- Collect a dataset of speech recordings from story books.
- Pre-process the dataset to remove noise and other artifacts.
- Train various pre-trained models on the dataset.
- Evaluate the performance of the models.
- Conduct a literature survey on speech emotion recognition.
- Write a report on the findings of the project.

Architecture

Gantt chart



## List of Tasks/Modules

---

1. List of tasks/Modules to be elaborated in discussion with the guide.
2. For example, If your project consist of Data Preprocessing, the following should be explained,
  - a. Data Collection & Data Preparation
  - b. Data Input
  - c. Data Pre-processing
  - d. Data Visualization
  - e. Data Interpretation
  - f. Storage
3. List the SDK / API / Model / Jar/ DLL / Tools / Technologies used - Open-Source/ Licensed.

# Architecture - S M SUTHARSAN RAJ

## 1. Design of overall Architecture and modules

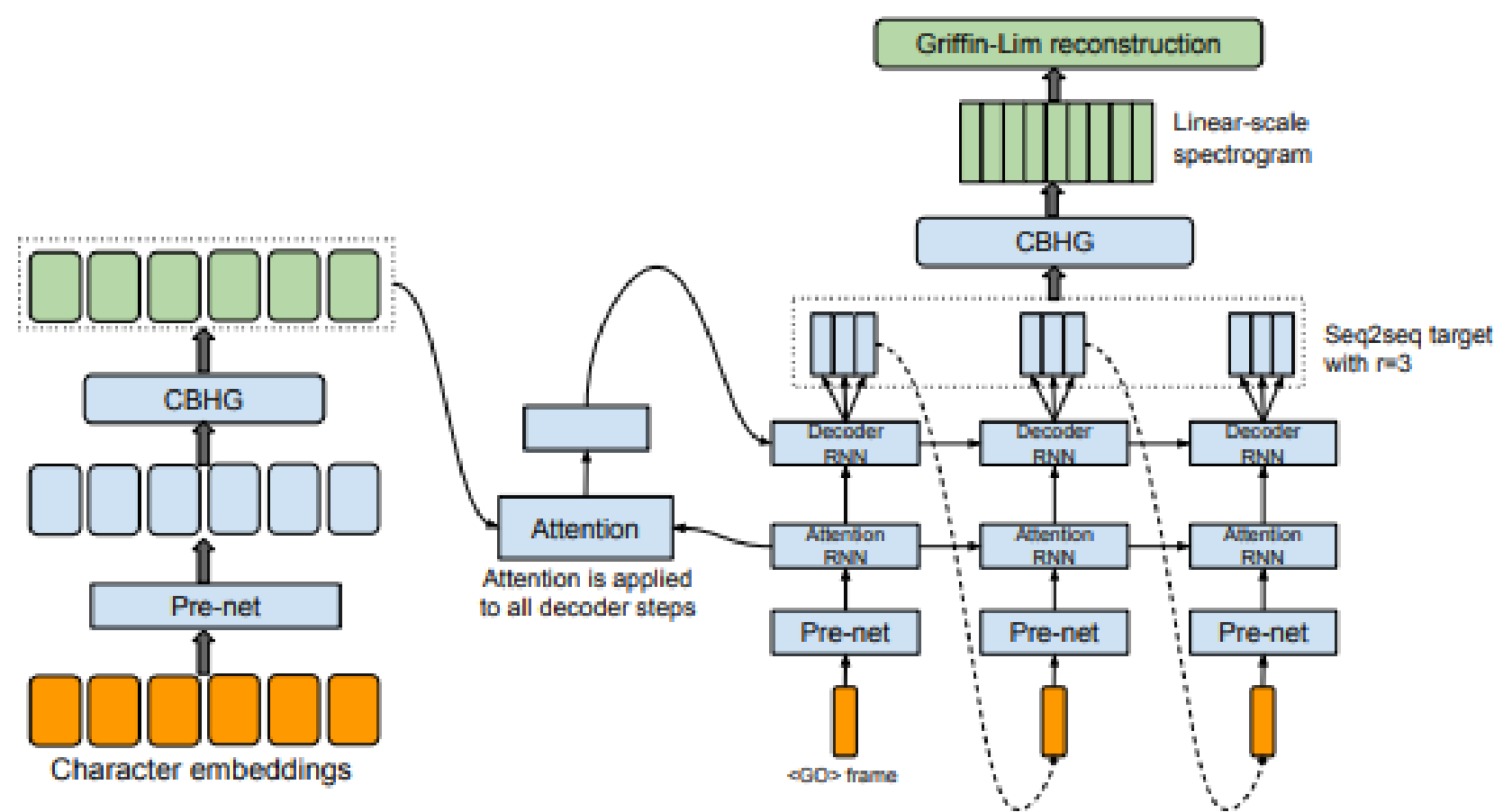


Figure 1: Model architecture. The model takes characters as input and outputs the corresponding raw spectrogram, which is then fed to the Griffin-Lim reconstruction algorithm to synthesize speech.

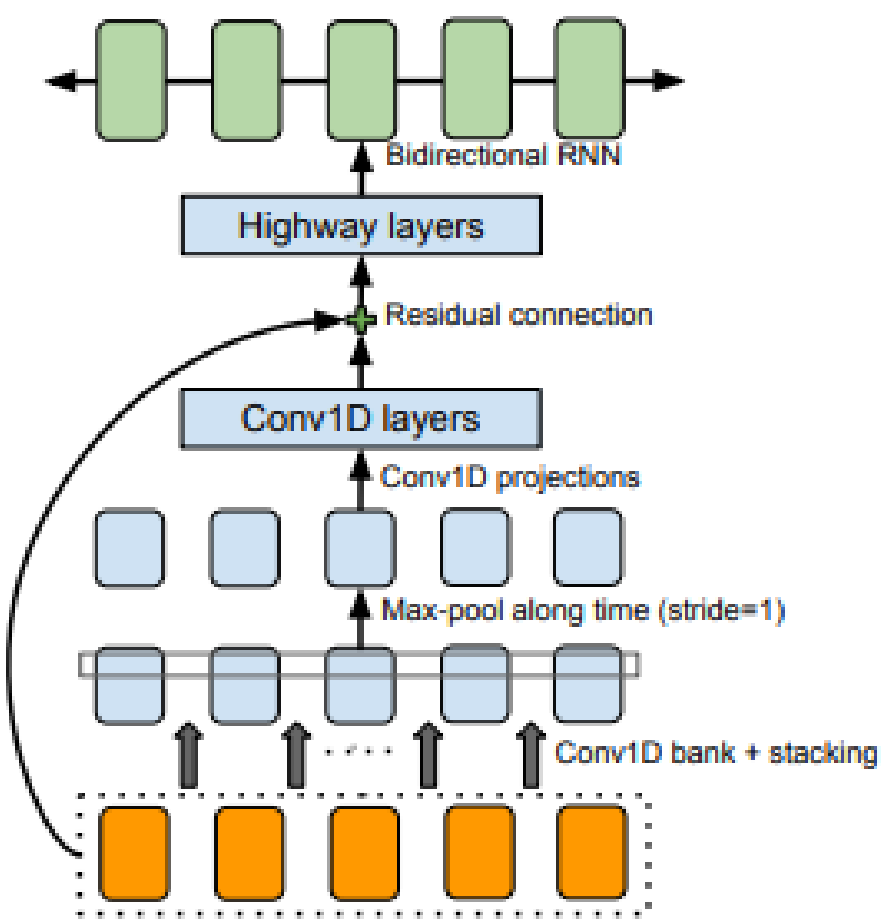


Figure 2: The CBHG (1-D convolution bank + highway network + bidirectional GRU) module adapted from Lee et al. (2016).



## List of Tasks/Modules - S M SUTHARSAN RAJ

---

### Data Collection & Data Preparation:

Data for this project likely includes text data and corresponding speech data.

Text data may have been collected from various sources, and speech data could come from recordings or other audio sources.

#### Data Input:

The text data and corresponding audio data are input into the system.

Text data may be provided in the form of transcriptions, and audio data in the form of WAV files, for example.

#### Data Pre-processing:

Text data is tokenized and processed to prepare it for training.

Audio data may undergo preprocessing, such as converting to mel spectrograms.

#### Data Visualization:

Data visualization might involve inspecting the spectrograms, text representations, or other data representations to ensure they are suitable for training.

## List of Tasks/Modules - S M SUTHARSAN RAJ

---

### Data Interpretation:

Understanding the characteristics of the data is essential to design an effective speech synthesis system. This could involve analyzing the distribution of text and audio data.

### Storage:

Data storage mechanisms are needed to manage and access the large amount of data involved in training. This could include organizing data in directories or databases.

### 3. Technologies and Tools Used:

The project likely uses a variety of open-source tools and libraries for different stages. Here's a list of common technologies and tools that could be used:

- **TensorFlow or PyTorch:** These deep learning frameworks are commonly used for building and training neural networks, including Tacotron.
- **Python:** The primary programming language for implementing and scripting various parts of the project.
- **NumPy and Pandas:** Used for data manipulation and handling.
- **Librosa:** A Python library for audio and music analysis. It can be used for audio file I/O and audio data processing.
- **Matplotlib or Seaborn:** These libraries can be used for data visualization and creating plots.
- **Text Processing Libraries:** For tokenization and text preprocessing.

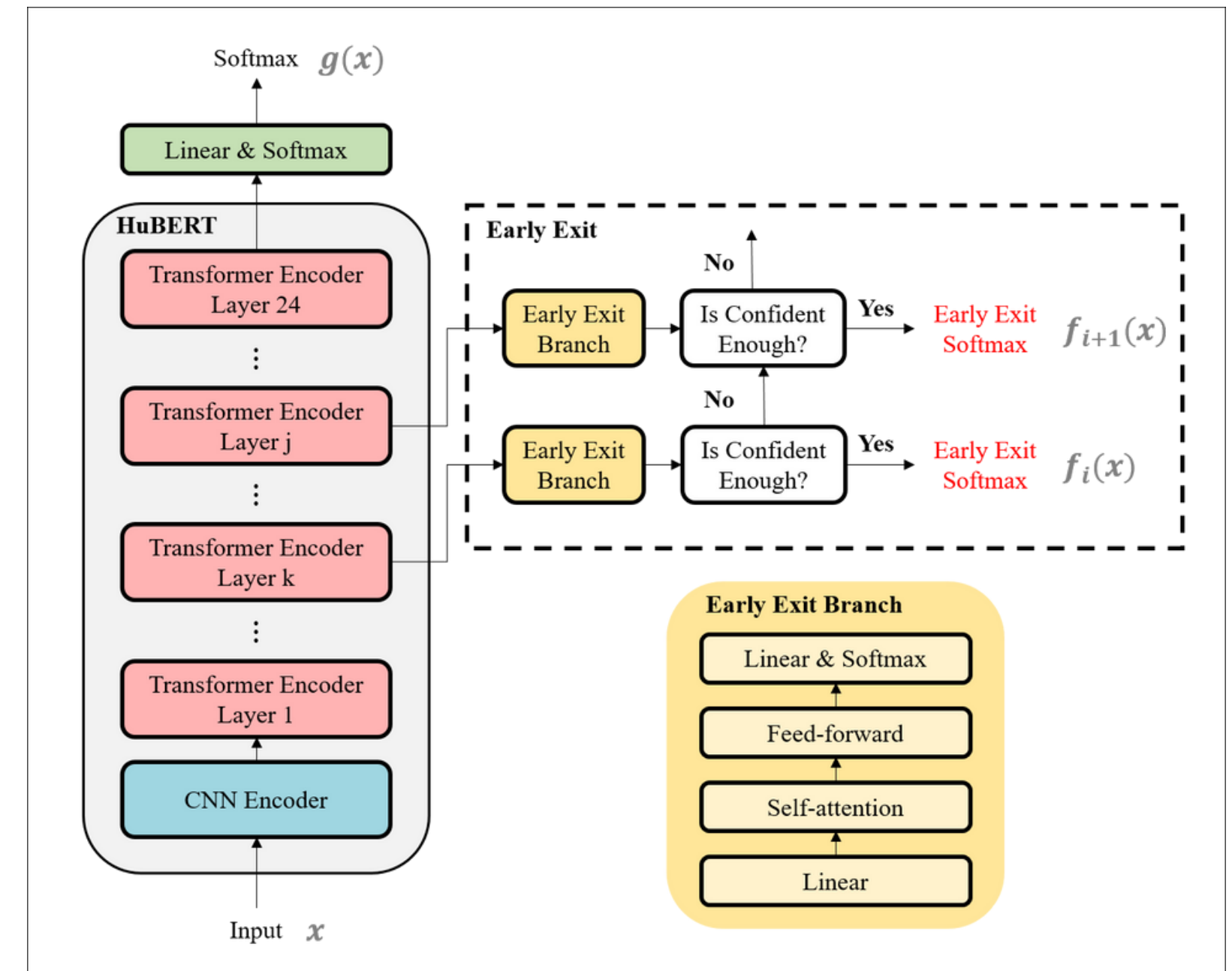
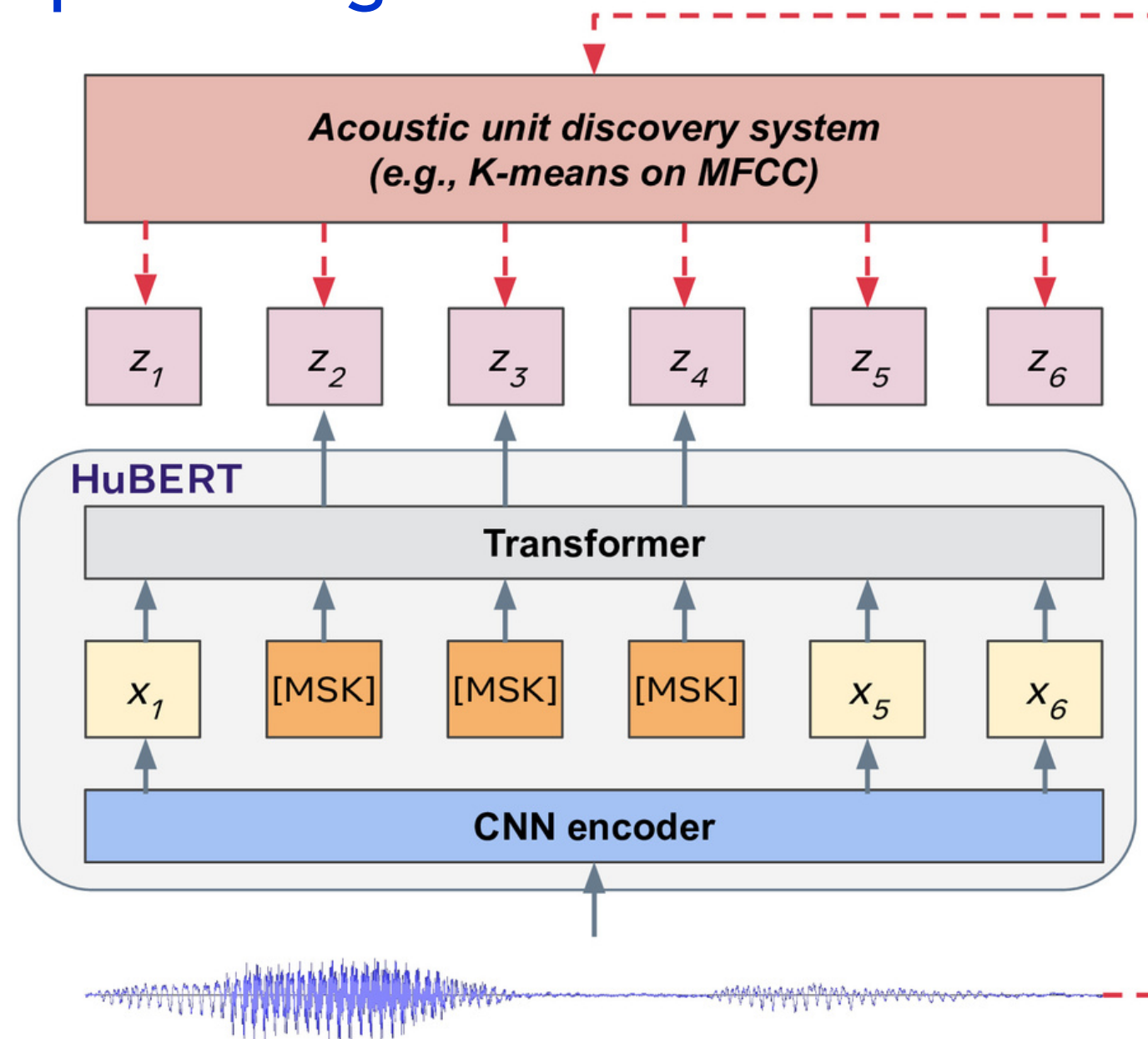
## List of Tasks/Modules - S M SUTHARSAN RAJ

---

```
Initialized Tacotron model. Dimensions:  
embedding: 256  
prenet out: 128  
encoder out: 256  
attention out: 256  
concat attn & out: 512  
decoder cell out: 256  
decoder out (5 frames): 400  
decoder out (1 frame): 80  
postnet out: 256  
linear out: 1025
```

## Architecture - M H Sohan

### HuBERT Model - extract intermediate representations of the speech signal



## List of Tasks/Modules - M H Sohan

- The HuBERT model is a self-supervised model that is trained on the task of masked prediction of continuous audio signals 3.
- It is a variant of the BERT model that is specifically designed for audio signals.
- The model is used to extract intermediate representations of the speech signal from different layers, which are then used to model expressive non-verbal communication cues and generate high-quality speech samples for speech emotion conversion

```
from transformers import AutoProcessor, HubertModel
from datasets import load_dataset
import soundfile as sf

processor = AutoProcessor.from_pretrained("facebook/hubert-large-ls960-ft")
model = HubertModel.from_pretrained("facebook/hubert-large-ls960-ft")

def map_to_array(batch):
    speech, _ = sf.read(batch["file"])
    batch["speech"] = speech
    return batch

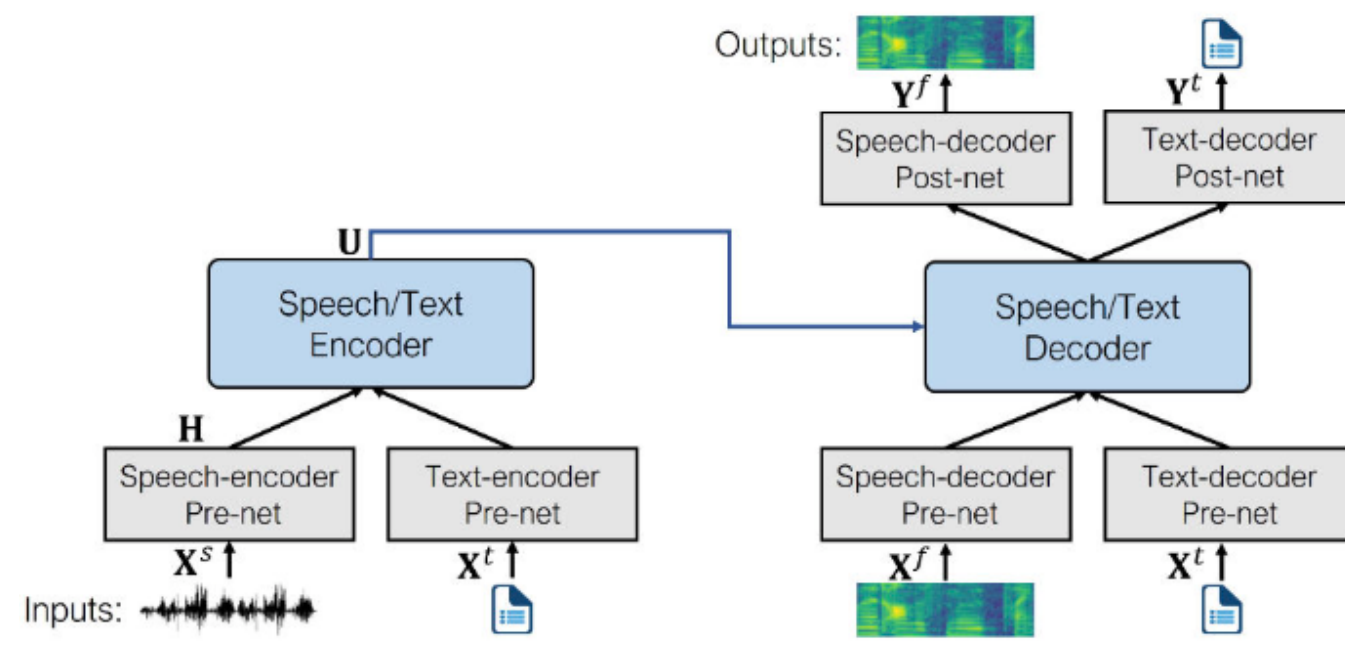
ds = load_dataset("hf-internal-testing/librispeech_asr_dummy", "clean", split="validation")
ds = ds.map(map_to_array)
input_values = processor(ds["speech"][0], return_tensors="pt").input_values # Batch size 1
hidden_states = model(input_values).last_hidden_state
```



## List of Tasks/Modules-Rohit Roshan

### Model - Hugging Face Model Bark

- Bark is a transformer-based text-to-audio model created by Suno.
- The model can also produce nonverbal communications like laughing, sighing and crying.



```

## Install

# install bark (make sure you have torch>=2 for much faster flash-attention)
!pip install git+https://github.com/suno-ai/bark.git

from bark import SAMPLE_RATE, generate_audio, preload_models
from IPython.display import Audio

preload_models()

text_prompt = """
    Hello, my name is Suno. And, uh – and I like pizza. [laughs]
    But I also have other interests such as playing tic tac toe.
"""

audio_array = generate_audio(text_prompt)
Audio(audio_array, rate=SAMPLE_RATE)

```

## List of Tasks/Modules-Rohit Roshan

### Google Text to speech

- gTTS (Google Text-to-Speech) is a Python library and CLI tool that interfaces with Google Translate's text-to-speech API.
- It converts text to neutral speech a

```
from gtts import gTTS  
tts = gTTS('This is the test for google text to speech ')  
tts.save('GTTS.mp3')
```

 GTTS.mp3

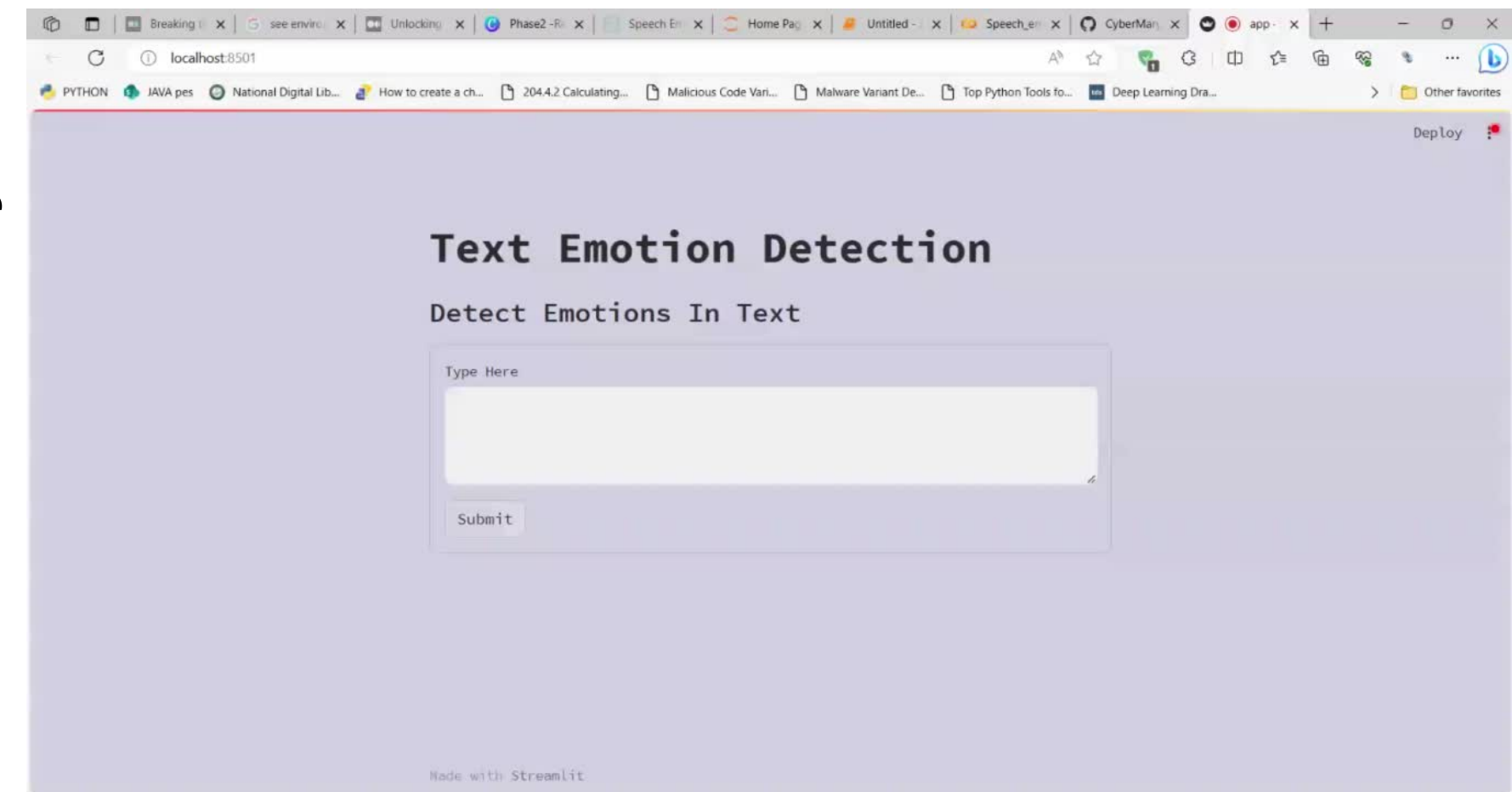
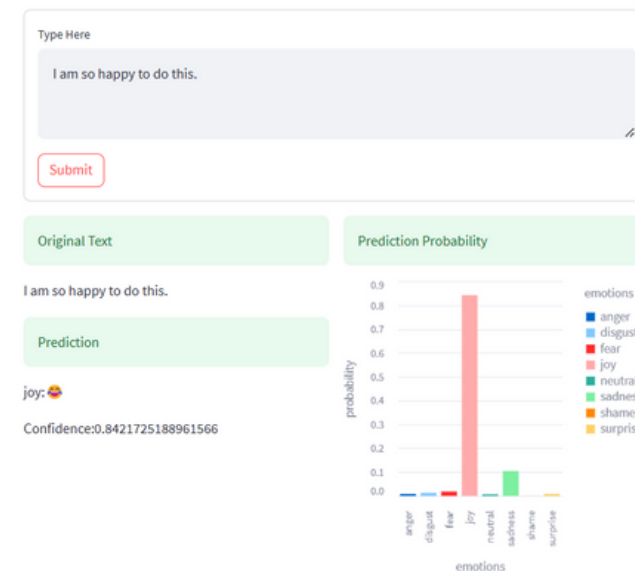
## List of Tasks/Modules-Rahul Roshan G

### Emotion detection from text.

- Detecting emotion from text using models like SVM and Random forest
- Saving the model using pickle library
- Visualizing the model with graph and confidence score using streamlit.

#### Text Emotion Detection

Detect Emotions In Text





## List of Tasks/Modules-Rahul Roshan G

### Emotion detection from text - LEXMO

- LeXmo: The first Python package for classifying emotions in English texts
- The LeXmo package converts the catalog into a pandas data frame, it receives a text, tokenizes it, sums up the number of words for that associate for each emotion, and return a dictionary of the text and emotions weights- the calculation of the emotional association divided with the text word count.
- Find the demo [here](#).

```
[ ] t= """From the beginning, she had sat looking at him fixedly.
    As he now leaned back in his chair, and bent his deep-set eyes upon her in his turn,
    perhaps he might have seen one wavering moment in her,
    when she was impelled to throw herself upon his breast,
    and give him the pent-up confidences of her heart.
    But, to see it, he must have overleaped at a bound the artificial barriers he had for many years been erecting,
    between himself and all those subtle essences of humanity which will elude the utmost cunning of algebra
    until the last trumpet ever to be sounded shall blow even algebra to wreck.
    The barriers were too many and too high for such a leap. With his unbending,
    utilitarian, matter-of-fact face, he hardened her again;
    and the moment shot away into the plumbless depths of the past,
    to mingle with all the lost opportunities that are drowned there."""

[ ] emo=LeXmo.LeXmo(t)

[ ] print(emo)

{'text': 'From the beginning, she had sat looking at him fixedly.\n As he now leaned back in his chair, and bent h

[ ] emo.pop('text', None)

'From the beginning, she had sat looking at him fixedly.\n As he now leaned back in his chair, and bent his deep-s
\n when she was impelled to throw herself upon his breast,\n and give him the pent-up confidences of her heart.\n
any years been erecting, \n between himself and all those subtle essences of humanity which will elude the utmost
o wreck.\n The barriers were too many and too high for such a leap. With his unbending,\n utilitarian, matter-of-
f the past,\n to mingle with all the lost opportunities that are drowned there.'

[ ] print(emo)

{'anger': 0.023255813953488372, 'anticipation': 0.0, 'disgust': 0.005813953488372093, 'fear': 0.023255813953488372,
```

## List of Tasks/Modules-Rahul Roshan G

---

### Emotion detection from text "EmoRoBERTa" Model from Huggingface Transformers

- It uses Emo-Roberta model to detect text from emotions from hugging face transformer.
- It calls the model use this [link](#) and predicts the emotion.
- The models gives dictionary with key as label(emotion) and score.

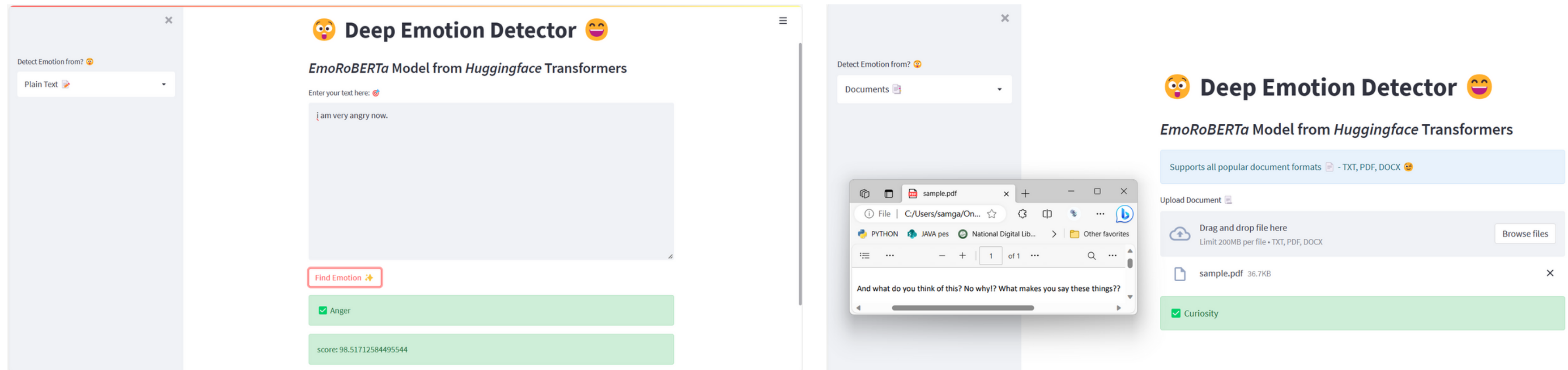
Dataset labelled 58000 Reddit comments with 28 emotions

- admiration, amusement, anger, annoyance, approval, caring, confusion, curiosity, desire, disappointment, disapproval, disgust, embarrassment, excitement, fear, gratitude, grief, joy, love, nervousness, optimism, pride, realization, relief, remorse, sadness, surprise + neutral

**Best results f1: 49.03%**

## List of Tasks/Modules-Rahul Roshan G

### Emotion detection from text "EmoRoBERTa" Model from Huggingface Transformers



The image displays two screenshots of the "Deep Emotion Detector" web application, which utilizes the "EmoRoBERTa" model from Huggingface Transformers.

**Left Screenshot (Text Input):**

- Header:** "Deep Emotion Detector" with a smiley face emoji.
- Model:** "EmoRoBERTa Model from Huggingface Transformers".
- Input:** "Enter your text here:" followed by a text area containing "i am very angry now."
- Action:** A red button labeled "Find Emotion 🚀".
- Output:** A green box showing "✅ Anger" and a score of "98.51712584495544".

**Right Screenshot (Document Upload):**

- Header:** "Deep Emotion Detector" with a smiley face emoji.
- Model:** "EmoRoBERTa Model from Huggingface Transformers".
- Support:** "Supports all popular document formats - TXT, PDF, DOCX 📄".
- Upload:** "Upload Document" section with a "Drag and drop file here" area (Limit 200MB per file • TXT, PDF, DOCX) and a "Browse files" button.
- File:** A file named "sample.pdf" (36.7KB) is shown as uploaded.
- Action:** A green box showing "✅ Curiosity".

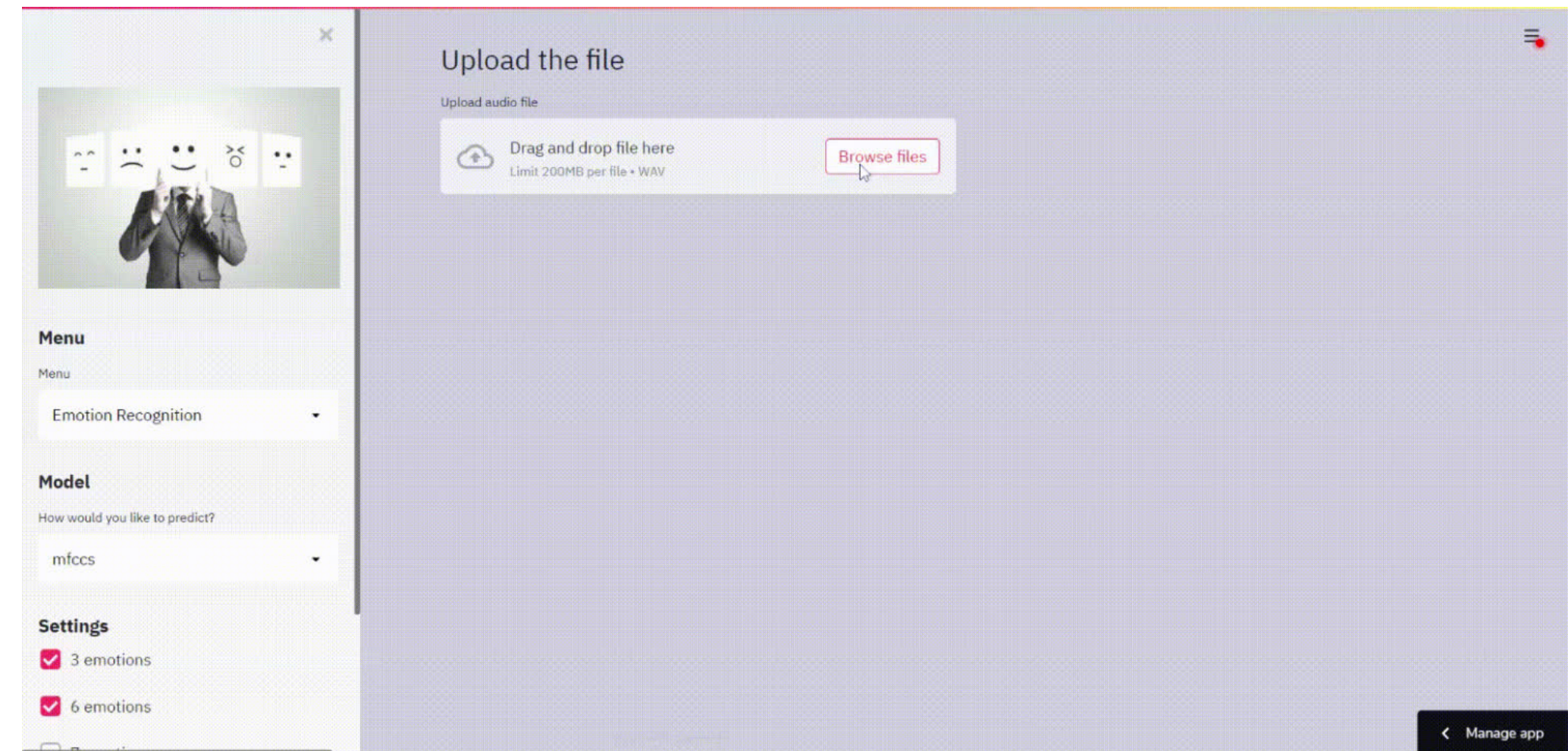
A small browser window titled "sample.pdf" is overlaid on the right screenshot, showing the text: "And what do you think of this? No why!? What makes you say these things??"



## List of Tasks/Modules-Rahul Roshan G

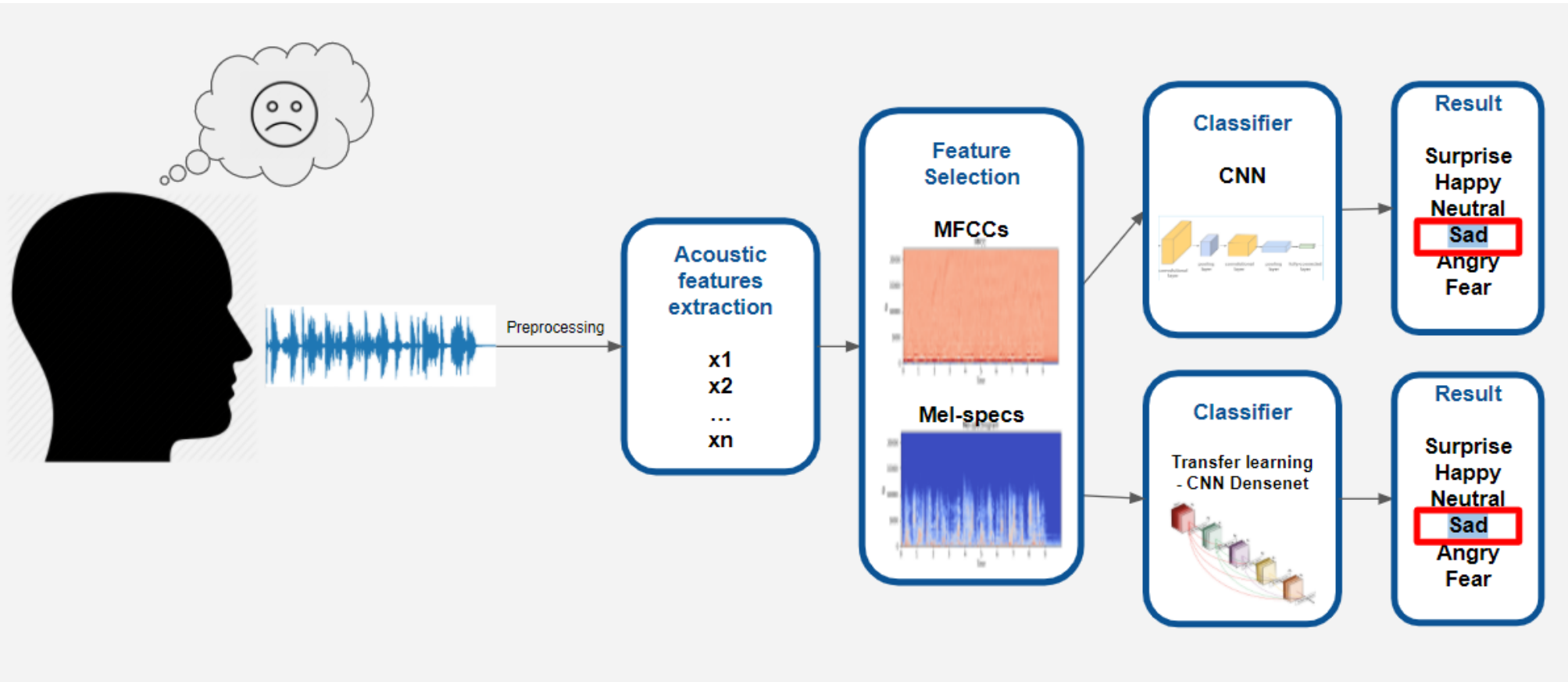
### Emotion detection from Audio/speech.

- Detecting emotion from text using models like CNN for classification.
- Pretrained model used to detect emotion from speech and showcased using Streamlit.



# List of Tasks/Modules-Rahul Roshan G

## Emotion detection from Audio/speech.



```
loaded_model3 = load_model('/content/drive/MyDrive/tmodel_all.h5')
loaded_model3.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
densenet201 (Functional)	(None, 7, 7, 1920)	18321984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1920)	0
flatten (Flatten)	(None, 1920)	0
dense (Dense)	(None, 256)	491776
dense_1 (Dense)	(None, 128)	32896
dense_2 (Dense)	(None, 6)	774

=====  
Total params: 18847430 (71.90 MB)  
Trainable params: 7504006 (28.63 MB)  
Non-trainable params: 11343424 (43.27 MB)



## References

---

- [1] Cai, Xiong, et al. "Emotion controllable speech synthesis using emotion-unlabeled dataset with the assistance of cross-domain speech emotion recognition." ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021.
- [2] Huan, Jeow & Sekh, Arif Ahmed & Quek, Chai & Prasad, Dilip. (2022). Emotionally charged text classification with deep learning and sentiment semantic. Neural Computing and Applications. 34. 10.1007/s00521-021-06542-1
- [3] P. Chandra et al., "Contextual Emotion Detection in Text using Deep Learning and Big Data," 2022 Second International Conference on Computer Science, Engineering and Applications (ICCSEA), Gunupur, India, 2022, pp. 1-5, doi: 10.1109/ICCSEA54677.2022.9936154.
- [4] Speech Emotion Recognition using Convolution Neural Networks and Deep Stride Convolutional Neural Networks - 2020 Published in: 2020 6th International Conference on Wireless and Telematics (ICWT) ISBN Information:INSPEC Accession Number: 20133021  
DOI: 10.1109/ICWT50448.2020.9243622

Thank You