Speech Emotion Recognition System

Samruddhi Khairnar - kbtug20170@kbtcoe.org
Avichal Sharma - avichalsharma2003@gmail.com
Yash Rokade - rokadeyash34@gmail.com
Araya Gupta - arayagupta28@gmail.com
Priyanshu Bisht - bisht.priyanshu05@gmail.com

Introduction

As a part of our internship at InternsElite, we chose ***Speech Emotion Recognition*** as the topic of our major project. This report briefly outlines the system design of our project - from consolidation of the CREMA-D dataset, till the possible real-world applications of our system, for deployment.

Problem Definition

Speech Emotion Recognition (SER) comprises audio techniques and deep learning methods in an act to recognize human emotion / mood from speech. SER allows machines to understand human emotions and finds its place in a wide variety of applications like - psychotherapy bots, medical research etc. Hence, we intend to design an SER system using deep learning techniques.

Objectives

- To learn to visualize and extract features from audio data, using *Librosa*.
- To learn and implement Deep Learning techniques on the CREMA-D dataset.
- To train deep learning models to classify human audio files as having emotions - anger, disgust, fear, happy, neutral or sad.
- To build a simple dashboard to record voice and classify it using the trained model, ***if time permits***.

Project Category

Data Science – *Deep Learning **(ML)** and Audio Processing*

Software Tools Required

- **Python 3 -** It is a high-level, general-purpose programming language.
- **JupyterLab -** It is a web-based interactive computing platform.
- **Libraries** :
    1. **Pandas -** Open source data analysis and manipulation tool, using Python.
    2. **Librosa** (+its imports) **-** Library for audio signal analysis using Python.
    3. **Tensorflow/Keras -** Libraries for implementing deep learning in Python.

Hardware and Software Requirements

- **Microprocessor -** Intel Core i5 / i7 (>= 6th gen)
- **RAM -** 8 GB or more
- **Operating Systems -** Windows / Linux / Macintosh

Requirement Specifications

Functionality

- Users shall be able to record their voice and get it classified in any one of the emotion buckets.
- They shall be able to visualize audio files and get an analysis of the audio signals.

Platform

The trained model shall be deployed on any cloud platform, to be accessible to all users.

Deliverables

1. This report.
2. Dataset Source Link - https://www.kaggle.com/datasets/ejlok1/cremad
3. Audio Signal Analysis + Feature Extraction source code as a Jupyter Notebook.
4. Model Training source code as a Jupyter Notebook.
5. The trained model - a pickle file.
6. The Dashboard source code as a Jupyter Notebook - *(**will be implemented only if time permits**)*.

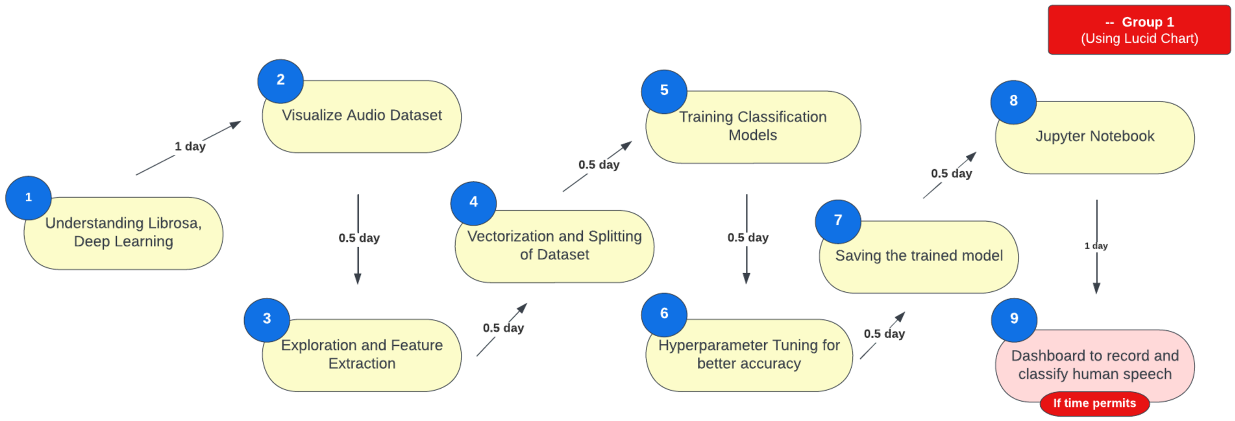
Project Scope

The scope of this project is restricted to extracting features like - Mel Frequencies and MFCCs from an audio dataset and training deep learning classification models to classify audio files based on emotions. **If time permits**, the scope will include building a dashboard to record and classify human speech into emotion buckets.

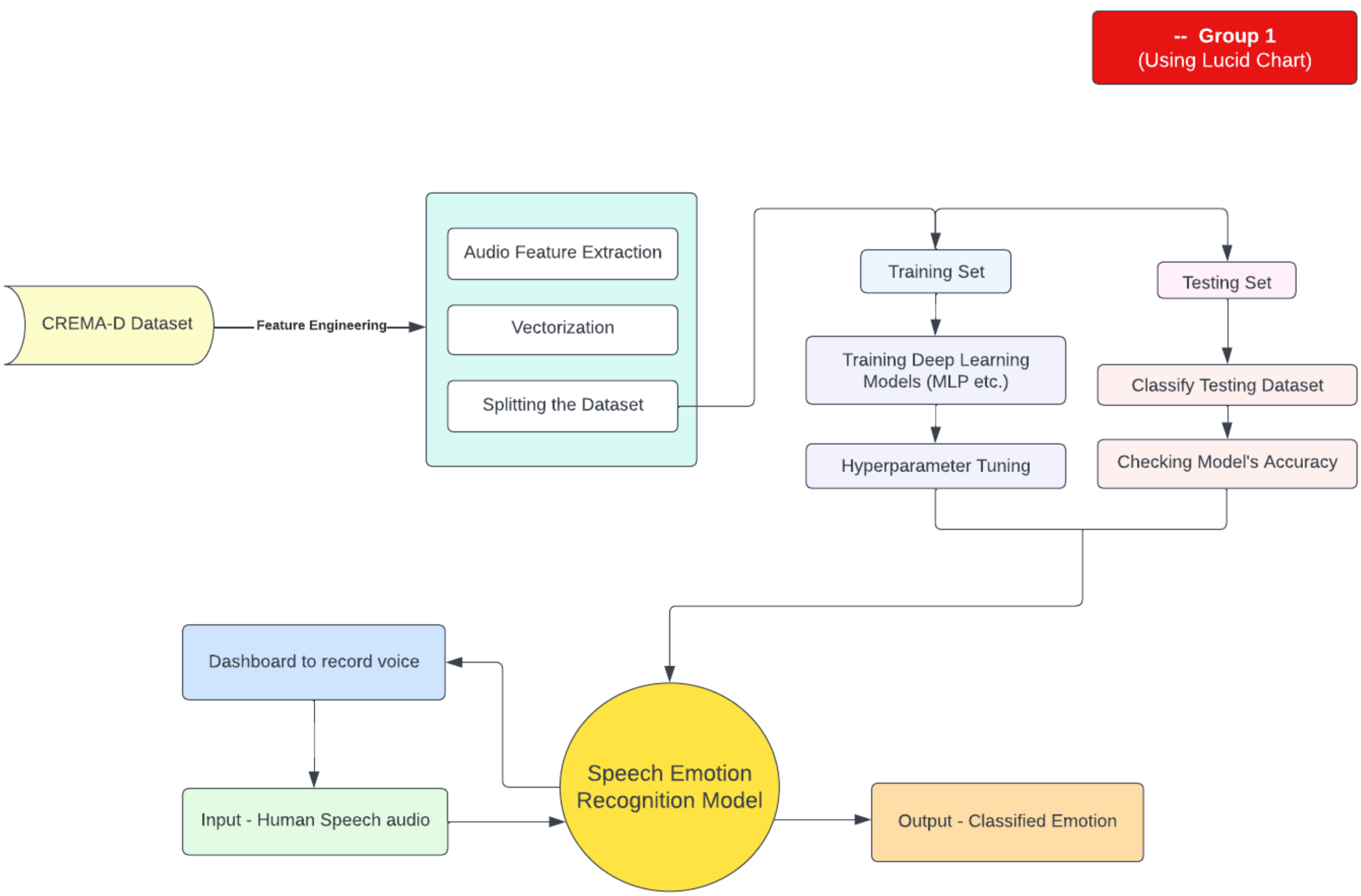
Audio Feature Extraction Theory (**Reference**: https://www.youtube.com/watch?v=PYlr8ayHb4g)

1. Audio files must be converted into frequency domain, to be interpretable. So initially **time series** data *(Amplitude vs time)* gets converted into a **frequency series** *(Frequency vs time)* using **Fast Fourier Transform**, to extract frequencies from the dataset.
2. Then, for smaller frequencies to be noticeable, the Y-axis *(frequency)* is scaled to be Mel or even better - **Log Mel**.
3. Then, MFCCs are extracted. **Mel-frequency cepstral coefficients** *(MFCC)* are **compressible** representations of the Log Mel Spectrogram.
4. **Chroma Feature** can also be extracted - chromagram bins the audio points into twelve different **pitch classes** *(CC#DD#EFF#GG#AA#B notes)*.
5. We will be using these features *(extracted using Librosa)* in our project, to train our models.
6. Data augmentation *(creating new data by adding nuances into existing data)* can also be done for better training of the model but we chose not to include it in our project.

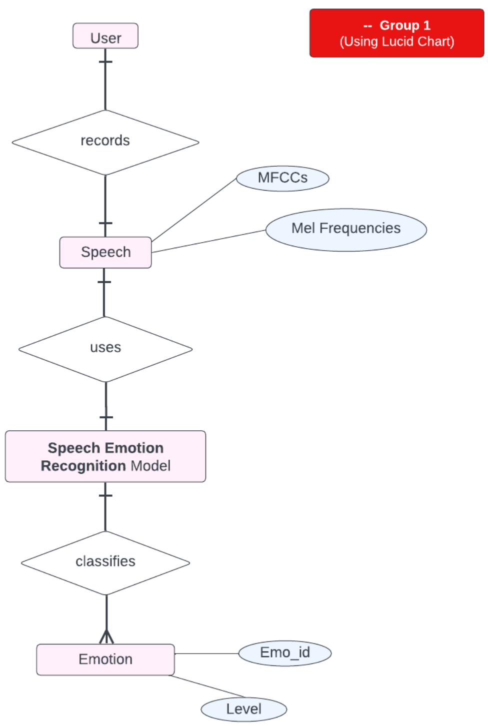
Project Scheduling - PERT



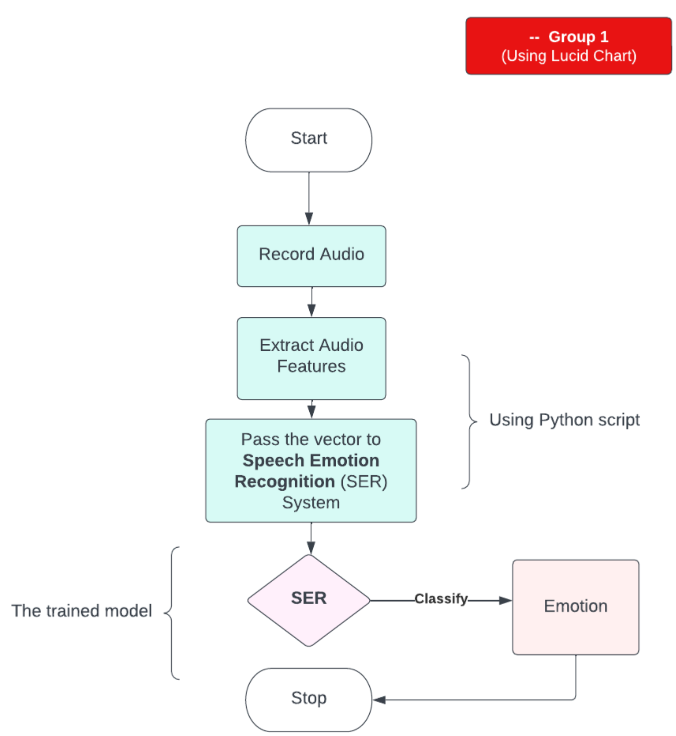
Analysis - DFD level 0



ER Diagram



System Design Flowchart



Modules

- **Audio Signal Analysis and Feature Extraction** *(using Librosa)* : This module will contain visualization of the audio files, followed by the extraction of features as explained in the theory part, above.
- **Deep Learning** *(using Tensorflow/Keras)* : This module will contain the model training and testing details, along with hyperparameter Tuning.
- **Dashboard** *(using Streamlit)* : This module will provide an interface to record real-time speech and show insights *(**will be implemented only if time permits**)*.

Dataset

References *:*

1. CREMA-D *(Zipped)* : https://www.kaggle.com/datasets/ejlok1/cremad
2. CREMA-D Dataset info : https://github.com/CheyneyComputerScience/CREMA-D

Overview :

- CREMA-D dataset contains 7,442 .wav audio files, recorded by 91 actors, speaking one sentence from a selection of 12 sentences, with 6 different emotions and 4 emotion levels.
- **Emotions** : Anger (ANG), Disgust (DIS), Fear (FEA), Happy/Joy (HAP), Neutral (NEU), Sad (SAD).
- **Emotion Levels** : Low (LO), Medium (MD), High (HI), Unspecified (XX).
- **Naming of files** : Actor id_Sentence_Emotion_Level.wav
- **Eg** : 1001_IEO_ANG_MD.wav

Implementation procedure

1. We will perform audio feature extraction and vectorization for the audio files in the dataset.
2. Then, we will train 1-2 deep learning classification models (MLP etc.) and evaluate their accuracy, preceded by tuning of their hyperparameters.
3. After selecting the most accurate model, we will save it and test it on unseen speech data.
4. Lastly, we will try to build a dashboard to record real-time speech and classify its emotion, **if time permits**.

Security Concerns

In future, if we plan to deploy our ***Speech Emotion Recognition*** model on any cloud platform, we will have to employ certain encryption techniques to secure the audio data of our users, in order to prevent privacy breaches.