SPEECH EMOTION DETECTION CLASSIFIER

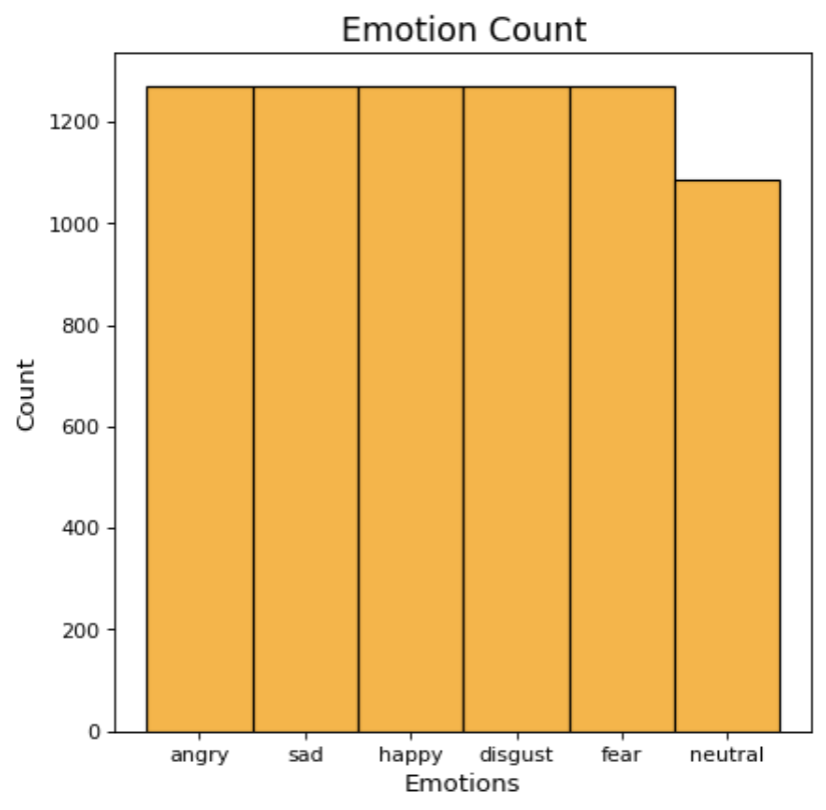
Emotion recognition is an essential aspect of speech analysis and has applications across customer service, healthcare, and virtual assistants. In this repository, we utilize deep learning techniques to recognize emotions from audio data using the CREMA-D dataset.

## **Dataset:**

The primary dataset used for this project is the Crowd-sourced Emotional Multimodal Actors Dataset (CREMA-D). It contains various audio recordings labeled with one of six emotions:

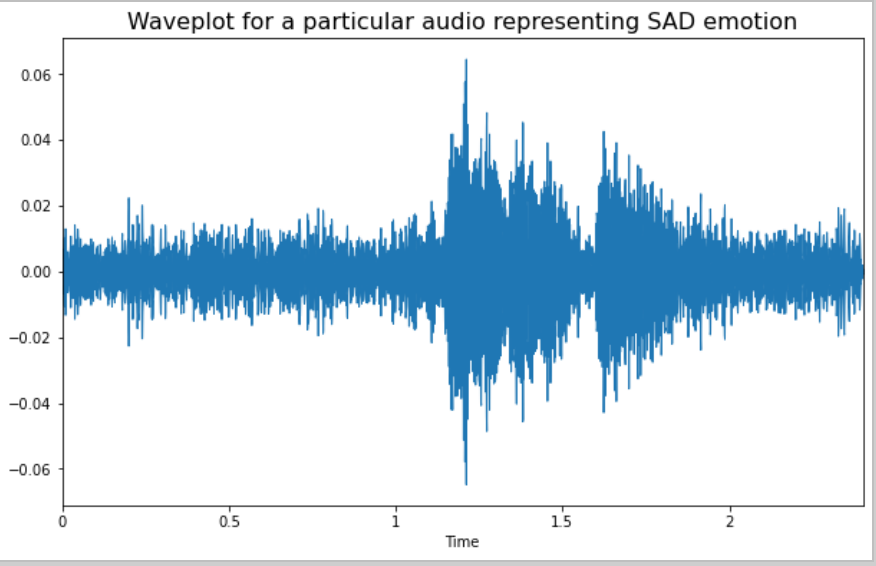
* Emotions in the dataset: sad, angry, disgust, neutral, happy, fear
* Dataset Composition: The dataset includes 7,442 audio samples from 91 actors, capturing a diverse range of expressions with different intensity levels and sentences spoken in various emotional tones.

**Emotions Count in the Dataset:**



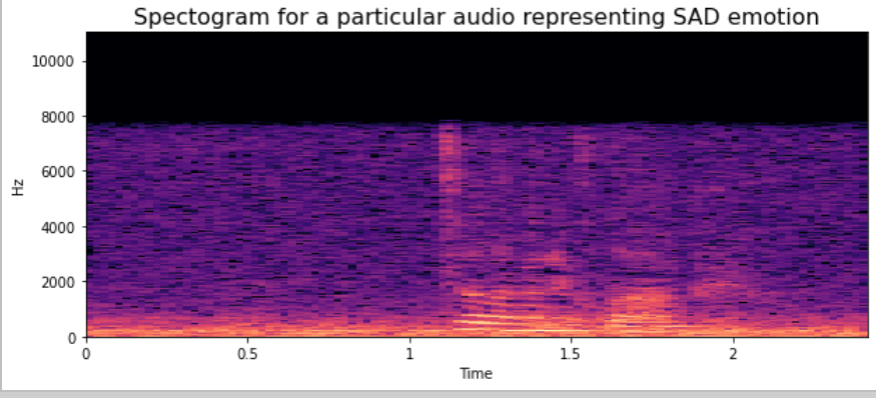
#### **Waveplot of a Sample Audio:**

A waveplot shows the loudness of audio over time:



#### **Spectrogram of a Sample Audio:**

A spectrogram visualizes sound frequencies over time:



## **Project Overview:**

The model used in this project is an **MLPClassifier**, which is trained on features extracted from the audio files, such as **MFCC (Mel Frequency Cepstral Coefficients)** and **mel-spectrogram**. These features enable the model to detect nuanced differences in speech patterns associated with different emotions.

### **Feature Extraction:**

1. **MFCC**: MFCCs are coefficients that collectively represent the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.
2. **Data Augmentation**: To improve the model’s robustness, **Noise Injection** was applied to generate additional synthetic training data

### **Data Processing**

1. **Normalization**: After extracting features, the data is normalized using StandardScaler.
2. **Split for Training/Testing**: The dataset is split into training and testing sets using train\_test\_split().

## **Model & Prediction Accuracy**

* **Model Used**: MLPClassifier
* **Configuration**:

MLPClassifier(

activation='relu',

solver='sgd',

hidden\_layer\_sizes=100,

alpha=0.839903176695813,

batch\_size=150,

learning\_rate='adaptive',

max\_iter=100000

)

### **Sample Predictions:**

The model's predictions for a sample set are shown below:

<<<===========================================>>>

Actual Predict

1011 angry angry

1689 neutral neutral

6092 angry angry

6231 angry disgust

7334 neutral disgust

## **How to Run the Code?**

* Open the Jupyter Notebook or run the Python script in your preferred environment.
* Load and preprocess the dataset, then execute the feature extraction and model training cells.
* **Run Inference Code**: To predict the emotion for a new audio sample, use the inference code provided in the notebook:

predicted\_emotion = mlp\_model.predict(new\_audio\_features)  
 print("Predicted Emotion:", predicted\_emotion)

**Results**: The notebook provides the evaluation results with accuracy scores and performance metrics on test data.

**Link for Trained Model:**

[**https://drive.google.com/file/d/1ZsEddBiTH4l78KKyfiioy-tfO5OAmMgT/view?usp=sharing**](https://drive.google.com/file/d/1ZsEddBiTH4l78KKyfiioy-tfO5OAmMgT/view?usp=sharing)

## **Conclusion:**

This project provides a framework for emotion detection in speech using deep learning techniques on audio data. With further improvements, such as incorporating larger datasets and more complex architectures like CNNs or RNNs, the model's performance can be enhanced for applications in real-world systems.