SPEECH EMOTION RECOGNITION MODEL

Submitted By-

Gaurav Mahaur	19095036
Saurabh Yadav	19095087
Varsha Jangir	19095104
Nishant K. Robin	19095121

Under the guidance of:
Dr. Satyabrata Jit
Department of Electronics Engineering

Objective:

- The idea behind creating this project is to build a machine learning model that could detect emotions from the speech using CNN.
- Speech Emotion Recognition(SER) Model, processes and classifies speech signals to detect emotions embedded in them.
- Using deep learning and machine learning algorithms with the help of TESS dataset we aim to design an automatic emotion recognition system.

How a Emotion Detection System helps?

Emotion detection from speech is a relatively new field of research, it has many potential applications. In human-computer or human-human interaction systems, emotion recognition systems could provide users with improved services by being adaptive to their emotions. In virtual worlds, emotion recognition could help simulate more realistic avatar interaction.

Applications

Non-Lexical communication with human

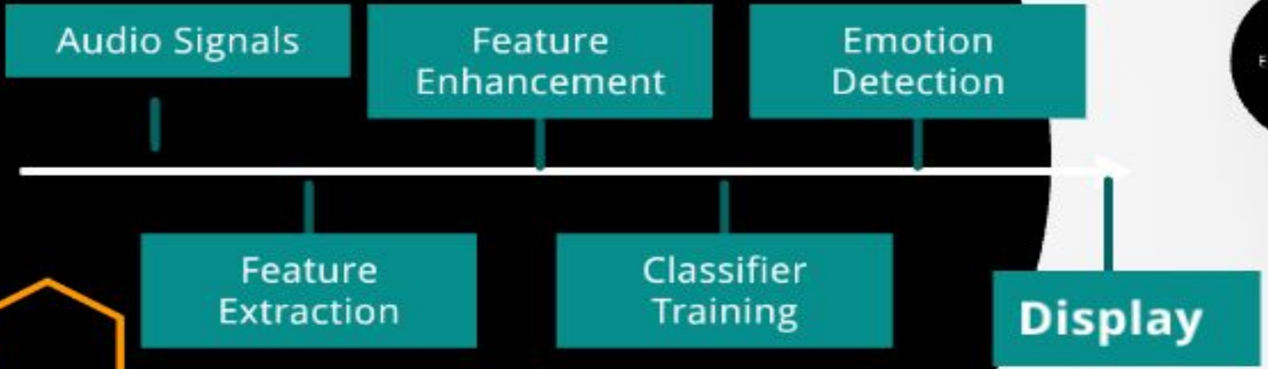
Online Marketing and shopping assistance on websites

Day-to-Day life Human Computer Interaction such as voice assistant like siri ! etc.

Feature Enhancement

Block Diagram

Audio Signals



Audio Signals

Feature Enhancement

Emotion Detection

Feature Extraction

Classifier Training

Display

Emotion Detection

Feature Extraction

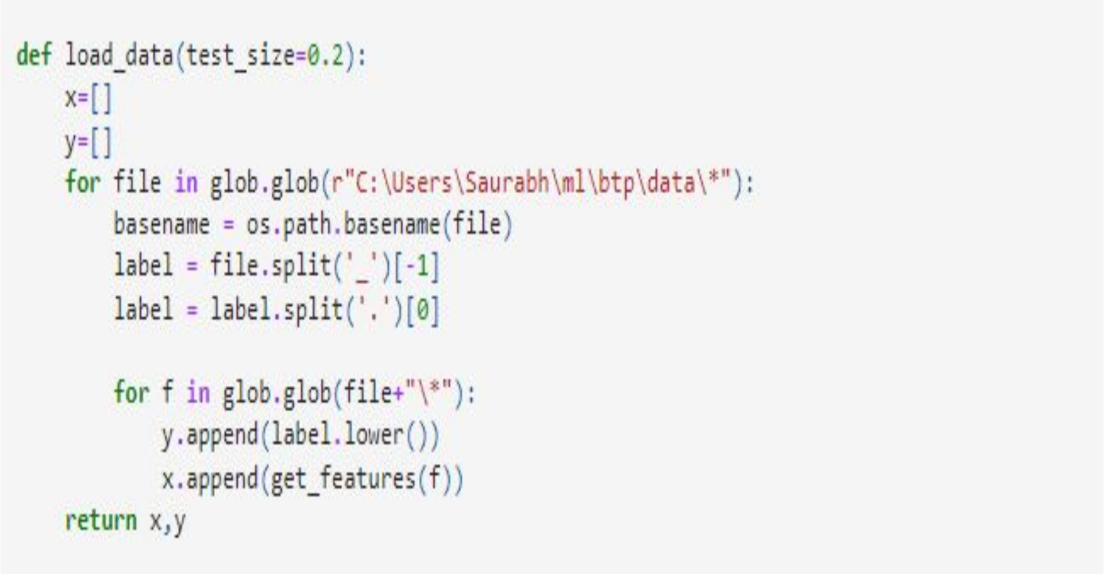
Classifier Training

DataSet:

In this project we have taken the input signal from TESS, which will be trained against the classifier models.

·*TESS (Toronto Emotional Speech Set):*

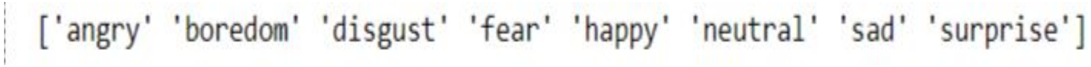
2 female speakers (young and old), 2800 audio files, random words were spoken in 8 different emotions.



```
def load_data(test_size=0.2):
    x=[]
    y=[]
    for file in glob.glob(r"C:\Users\Saurabh\ml\btp\data\*"):
        basename = os.path.basename(file)
        label = file.split('_')[-1]
        label = label.split('.')[0]

        for f in glob.glob(file+"\*"):
            y.append(label.lower())
            x.append(get_features(f))
    return x,y
```

Dataset labels



```
['angry' 'boredom' 'disgust' 'fear' 'happy' 'neutral' 'sad' 'surprise']
```

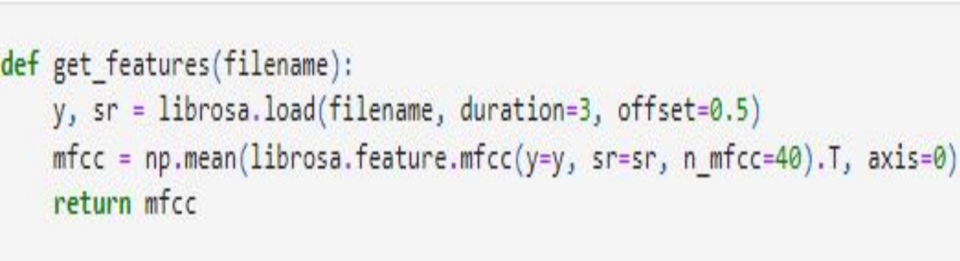
Data Pre-processing:

Data must be cleaned to perform any meaningful analysis. As a next step, the dataset thus collected had to be inspected for its quality. Some of the data quality issues addressed for this experimentation include:

1. **Missing value analysis**
2. **Outlier identification**
3. **Null value handling**
4. **Invalid data**
5. **Duplicate data**

Feature Extraction:

From the Audio data we have extracted three key features which have been used in this study, namely *MFCC (Mel Frequency Cepstral Coefficients), Mel Spectrogram and Chroma*. The Python implementation of Librosa package was used in their extraction.



```
def get_features(filename):
    y, sr = librosa.load(filename, duration=3, offset=0.5)
    mfcc = np.mean(librosa.feature.mfcc(y=y, sr=sr, n_mfcc=40).T, axis=0)
    return mfcc
```

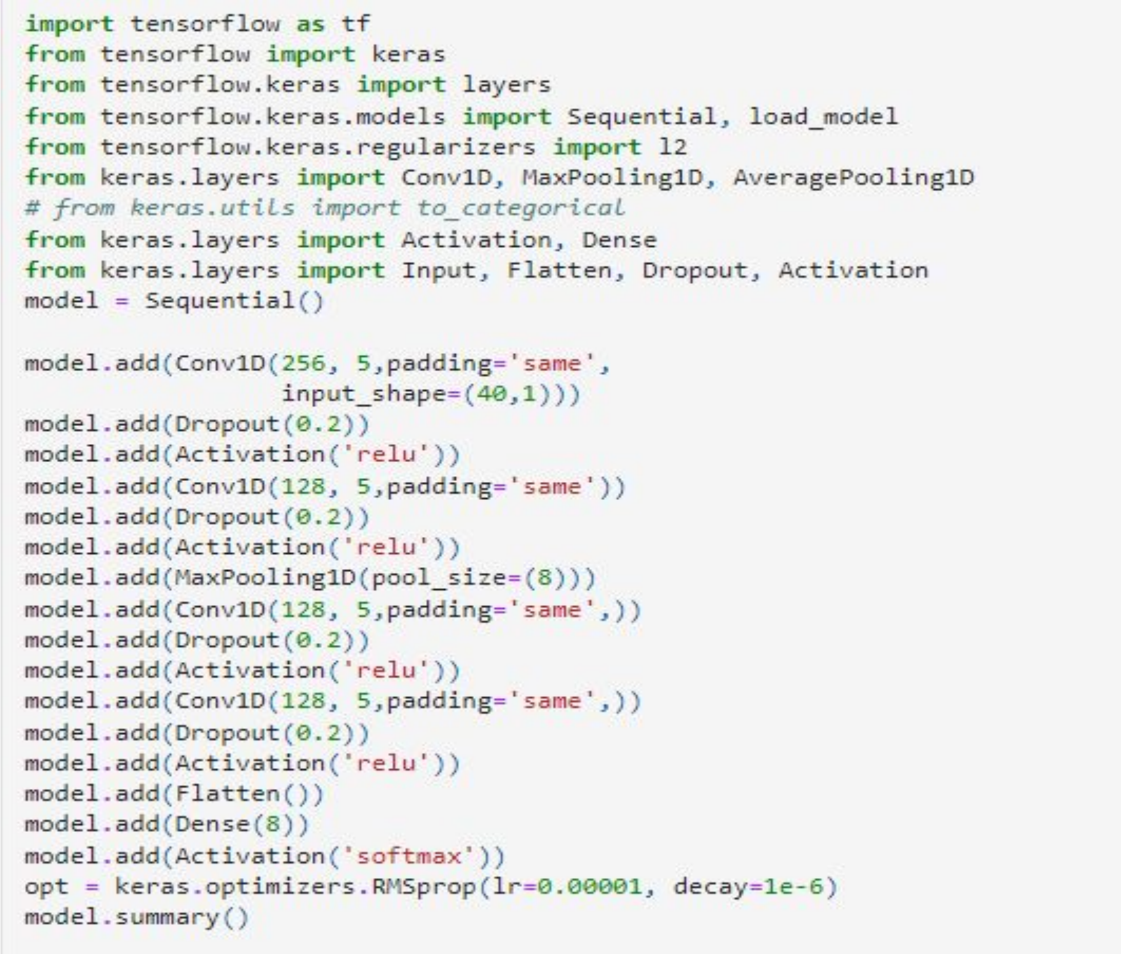
Choice of Features:

- MFCC was by far the most researched about and utilized features in research papers and open source projects.
- Mel spectrogram plots amplitude on frequency vs time graph on a “Mel” scale. As the project is on emotion recognition, a purely subjective item, we found it better to plot the amplitude on Mel scale as Mel scale changes the recorded frequency to “perceived frequency”.
- Researchers have also used Chroma in their projects as per literatures, thus we also tried basic modeling with only MFCC and Mel and with all MFCC, Mel, Chroma. The model with all of the features gave slightly better results, hence we chose to keep all three features

CNN MODEL AND ARCHITECTURE

We have developed the CNN model with Keras and constructed it with 5 layers — 4 Conv1D layers followed by a Dense layer.

Architecture of the CNN Model is as follows:

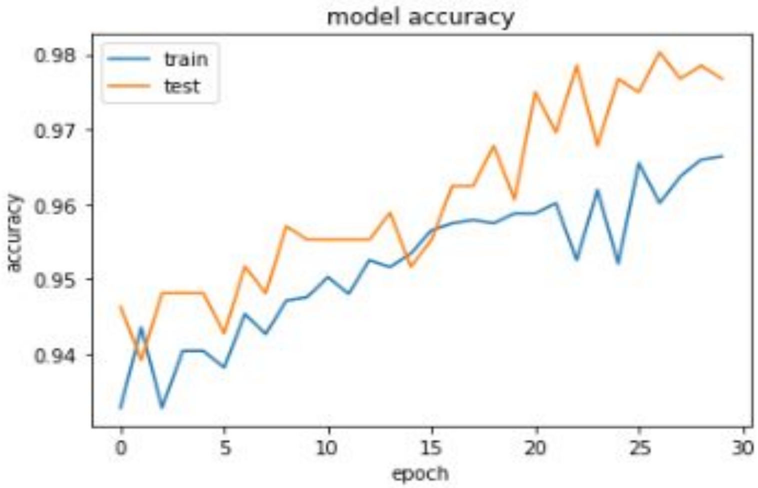


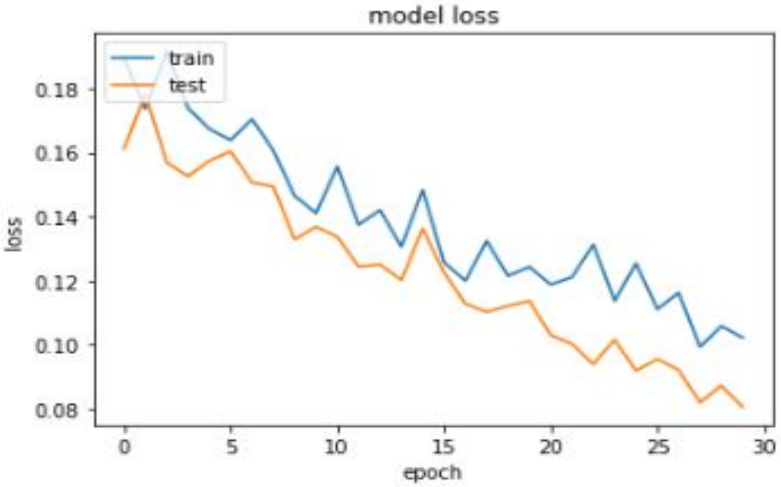
```
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.models import Sequential, load_model
from tensorflow.keras.regularizers import l2
from keras.layers import Conv1D, MaxPooling1D, AveragePooling1D
# from keras.utils import to_categorical
from keras.layers import Activation, Dense
from keras.layers import Input, Flatten, Dropout, Activation
model = Sequential()

model.add(Conv1D(256, 5,padding='same',
                 input_shape=(40,1)))
model.add(Dropout(0.2))
model.add(Activation('relu'))
model.add(Conv1D(128, 5,padding='same'))
model.add(Dropout(0.2))
model.add(Activation('relu'))
model.add(MaxPooling1D(pool_size=(8)))
model.add(Conv1D(128, 5,padding='same',))
model.add(Dropout(0.2))
model.add(Activation('relu'))
model.add(Conv1D(128, 5,padding='same',))
model.add(Dropout(0.2))
model.add(Activation('relu'))
model.add(Flatten())
model.add(Dense(8))
model.add(Activation('softmax'))
opt = keras.optimizers.RMSprop(lr=0.00001, decay=1e-6)
model.summary()
```

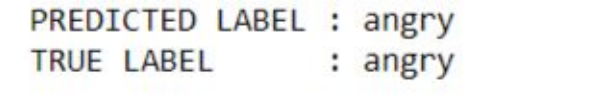
EXPERIMENTAL RESULT

From the TESS dataset, 80 % of the audio files from these are used for training, and 20 % are used for testing the model. The network is trained for 30 epochs. *The training accuracy of the proposed SER model is 96.64% and the validation accuracy is 97.67%.The precision of the model is 97.91% and the recall value is 97.39%.*

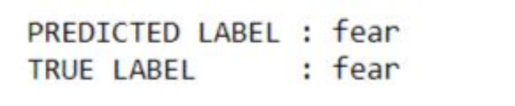




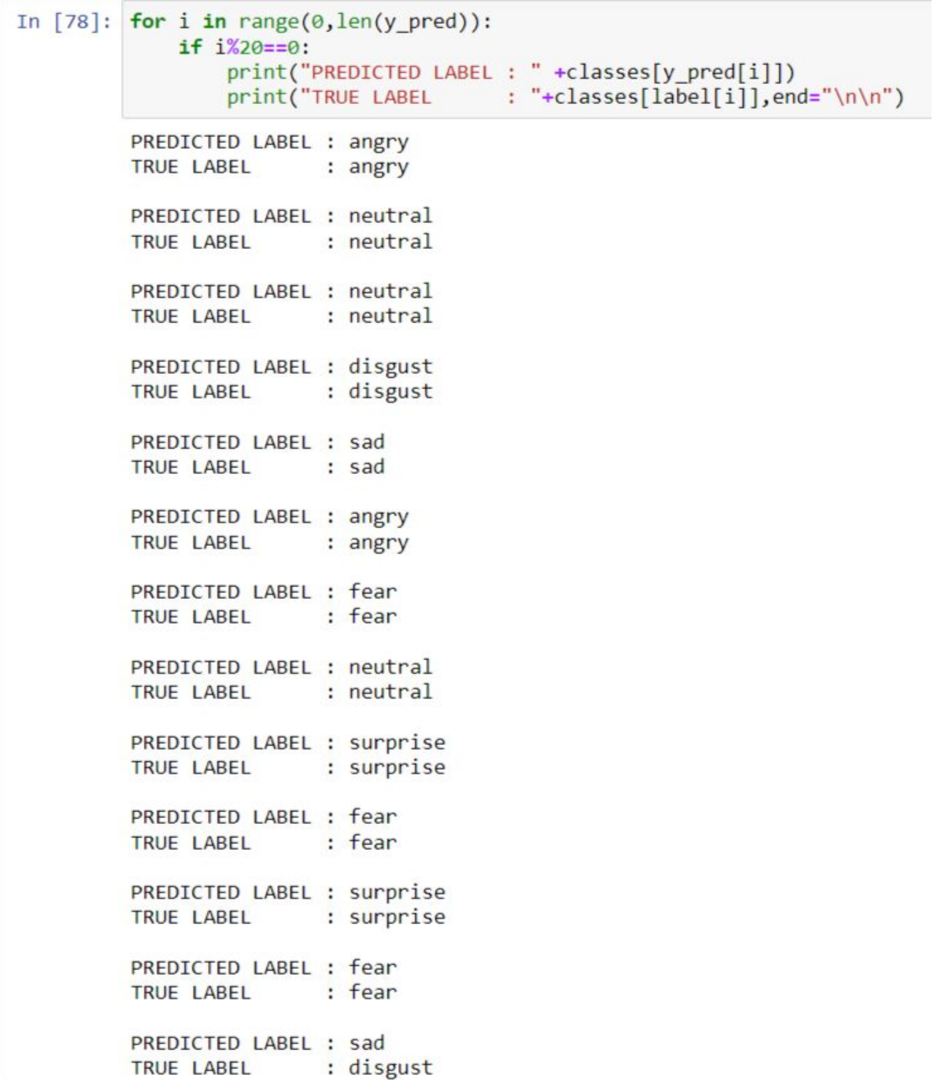
Some Predictions



```
PREDICTED LABEL : angry
TRUE LABEL      : angry
```



```
PREDICTED LABEL : fear
TRUE LABEL      : fear
```



```
In [78]: for i in range(0,len(y_pred)):
             if i%20==0:
                 print("PREDICTED LABEL : " +classes[y_pred[i]])
                 print("TRUE LABEL      : "+classes[label[i]],end="\n\n")
```

```
PREDICTED LABEL : angry
TRUE LABEL      : angry

PREDICTED LABEL : neutral
TRUE LABEL      : neutral

PREDICTED LABEL : neutral
TRUE LABEL      : neutral

PREDICTED LABEL : disgust
TRUE LABEL      : disgust

PREDICTED LABEL : sad
TRUE LABEL      : sad

PREDICTED LABEL : angry
TRUE LABEL      : angry

PREDICTED LABEL : fear
TRUE LABEL      : fear

PREDICTED LABEL : neutral
TRUE LABEL      : neutral

PREDICTED LABEL : surprise
TRUE LABEL      : surprise

PREDICTED LABEL : fear
TRUE LABEL      : fear

PREDICTED LABEL : surprise
TRUE LABEL      : surprise

PREDICTED LABEL : fear
TRUE LABEL      : fear

PREDICTED LABEL : sad
TRUE LABEL      : disgust
```

Conclusion:

- Through this project, we showed how we can leverage Machine learning to obtain the underlying emotion from speech audio data and some insights on the human expression of emotion through voice.
- In this report, the steps of building a speech emotion detection system were discussed in detail Initially, the limited number of publically available speech databases made it challenging to implement a well-trained model
- Finally, the classifier selection involved learning about the strength and weaknesses of each classifying algorithm with respect to emotion recognition.
- At the end of the experimentation, it can be concluded that an integrated feature space will produce a better recognition rate when compared to a single feature.

FUTURE WORK:

- Additional to the emotions, the model can be extended to recognize feelings such as depression and mood changes.
- Use of different dataset (eg. RAVDESS, SAVEE, CREMA-D, Berlin) to make model more robust for real world use.
- Facial Emotion Detection.
- Speech Emotion Detection better accuracy can be achieved.

Therefore, in the future, there would emerge many applications of a speech-based emotion.

REFERENCES:

- https://www.researchgate.net/publication/322873355_Speech_Emotion_Recognition_Methods_and_Cases_Study

- https://www.kaggle.com/datasets/ejlok1/toronto-emotional-speech-set-tess
  --The dataset used in this model

- https://medium.com/@patrickbfuller/librosa-a-python-audio-libary-60014eeaccfb
  - Librosa library article

- https://ieeexplore.ieee.org/document/8308186 - CNN article

Thank You!