Mini Project Report

Topic : Speaker Recognition

Group Members:

1)Anooj Sarvankar

2)Shail Shroff

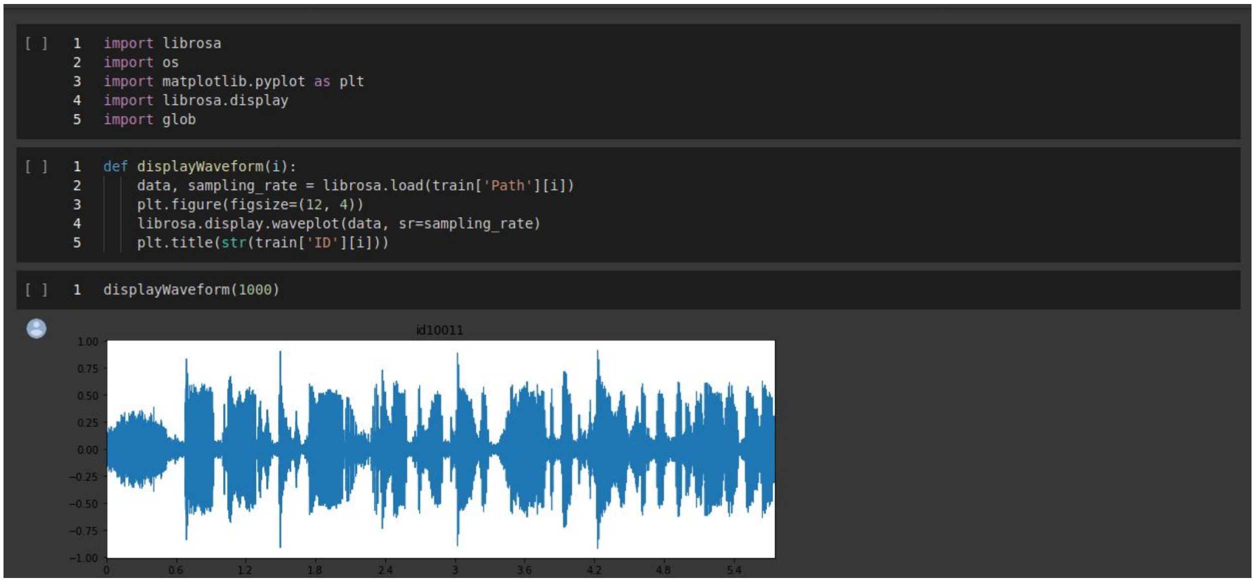
Abstract:

Each human voice has its own characteristic features. The features that describe the human voice can be: the pitch, the energy of the signal, the mean value, the median value, the MFCC coefficients, and others. Some of the natural questions that come to mind are: which are the features that best characterize the human voice? What is the minimum number of features that can completely characterize the human voice? In order to answer these questions a database of human voices is used, along with different feature extraction algorithms. Just extracting the features is not enough. Classification algorithms will be tried and used for classification of distinct speakers.

Literature Survey:

The literature survey for research was done by referring to the journal papers, conference papers, articles, books, internet and databases.Overall, this chapter describes a review of voice recognition tasks, voice recognition approaches, current voice recognition system as well as different types of methods applied to voice recognition systems. Based on the review of the advantages and disadvantages, this thesis discusses the most suitable techniques and methods to develop a voice recognition system.

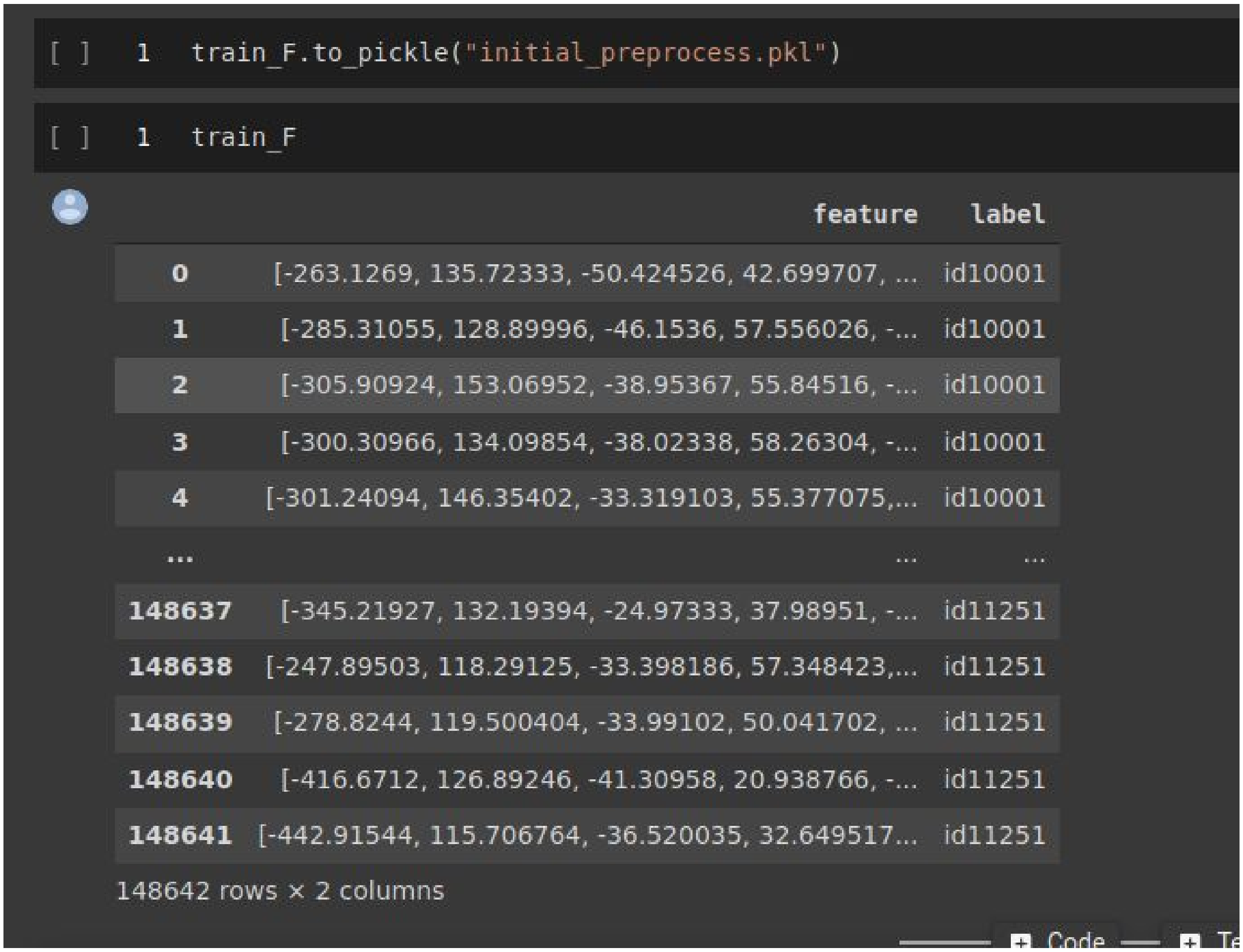
Data Preprocessing:



```
[ ]  import librosa
     import os
     import matplotlib.pyplot as plt
     import librosa.display
     import glob
```

```
[ ]  def displayWaveform(i):
         data, sampling_rate = librosa.load(train['Path'][i])
         plt.figure(figsize=(12, 4))
         librosa.display.waveplot(data, sr=sampling_rate)
         plt.title(str(train['ID'][i]))
```

```
[ ]  displayWaveform(1000)
```



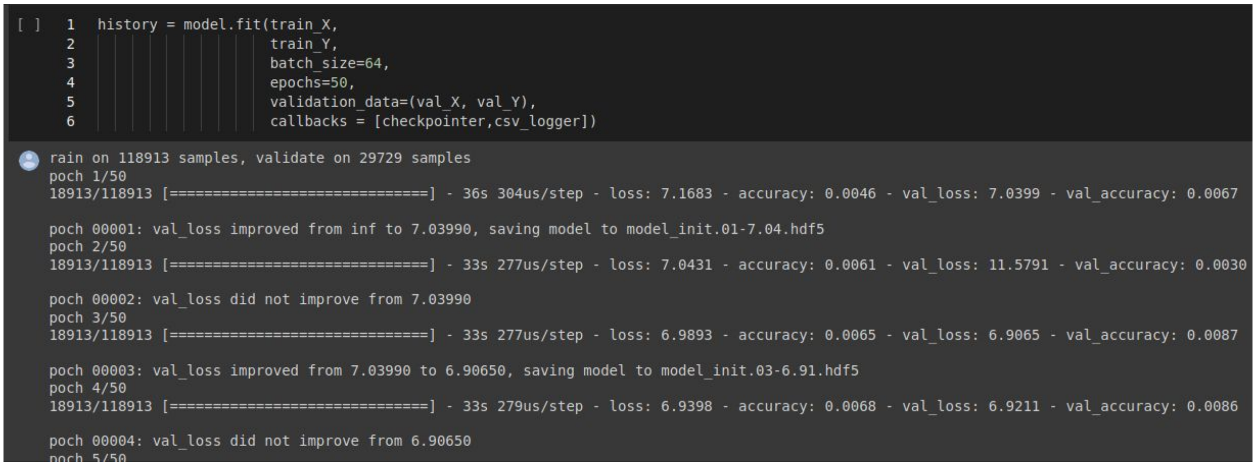
```
[ ]  train_F.to_pickle("initial_preprocess.pkl")
```

```
[ ]  train_F
```

	feature	label
0	[-263.1269, 135.72333, -50.424526, 42.699707, ...	id10001
1	[-285.31055, 128.89996, -46.1536, 57.556026, -...	id10001
2	[-305.90924, 153.06952, -38.95367, 55.84516, -...	id10001
3	[-300.30966, 134.09854, -38.02338, 58.26304, -...	id10001
4	[-301.24094, 146.35402, -33.319103, 55.377075,...	id10001
...	...	...
148637	[-345.21927, 132.19394, -24.97333, 37.98951, -...	id11251
148638	[-247.89503, 118.29125, -33.398186, 57.348423,...	id11251
148639	[-278.8244, 119.500404, -33.99102, 50.041702, ...	id11251
148640	[-416.6712, 126.89246, -41.30958, 20.938766, -...	id11251
148641	[-442.91544, 115.706764, -36.520035, 32.649517...	id11251

148642 rows × 2 columns

Results on Sample set of data:



```
history = model.fit(train_X,
                    train_Y,
                    batch_size=64,
                    epochs=50,
                    validation_data=(val_X, val_Y),
                    callbacks = [checkpointer,csv_logger])
```

```
rain on 118913 samples, validate on 29729 samples
poch 1/50
18913/118913 [==============================] - 36s 304us/step - loss: 7.1683 - accuracy: 0.0046 - val_loss: 7.0399 - val_accuracy: 0.0067

poch 00001: val_loss improved from inf to 7.03990, saving model to model_init.01-7.04.hdf5
poch 2/50
18913/118913 [==============================] - 33s 277us/step - loss: 7.0431 - accuracy: 0.0061 - val_loss: 11.5791 - val_accuracy: 0.0030

poch 00002: val_loss did not improve from 7.03990
poch 3/50
18913/118913 [==============================] - 33s 277us/step - loss: 6.9893 - accuracy: 0.0065 - val_loss: 6.9065 - val_accuracy: 0.0087

poch 00003: val_loss improved from 7.03990 to 6.90650, saving model to model_init.03-6.91.hdf5
poch 4/50
18913/118913 [==============================] - 33s 279us/step - loss: 6.9398 - accuracy: 0.0068 - val_loss: 6.9211 - val_accuracy: 0.0086

poch 00004: val_loss did not improve from 6.90650
poch 5/50
```

Algorithms and Techniques:

1. Librosa
2. Keras
3. Numpy
4. Pandas
5. Sklearn
6. Sequential model

To-Do:

1.Implement as many different architecture models to check for increase in accuracy.

2.Experiment with preprocessing for potential features.

3.Try Transfer Learning