Batch #6

Prabhjyot Singh Sodhi

Chandradeo Arya

Tanmay Utkarsh

Speaker Recognition

Speaker recognition or Voice Recognition is the identification of a person from characteristics of voices.

We aim to implement the following modules:

- Training a Random Forest Classifier on a dataset of Indian voices
- Test and report the accuracy of the model.
- Allow user to record (using mic) and add to existing datasets
- Allow user to record and test the recorded voice at runtime.

Dataset :-

For our project we have used data set generated by Azarias Reda(university of michigan), Saurabh Panjwani(Microsoft research, India). The data set has been generated from Indian users from all the states using IVR (Interactive voice response) system. All voice samples are in wav format which is the most enriched format for storing speech sample.

Details: The dataset contains separate data for male and female speaking a sequence of numbers in English. The dataset contains background noise, faints and hisses which are commonly observed in anyone's speech up to the extent of 30% which makes it a perfect dataset for training any speaker recognition machine learning system.

Dataset is organized in two separate directories for male and female. Both the directories have been divided into subdirectories for each person and a unique ID in the format M/F+'_personId' is provided. For each person the corresponding directory contains voice samples named in the format ‘personId_’+sequence(1,2,3…).Each voice sample reads a particular sequence of digits like in sample1 it may be something like 02154368 and sample2 6704352918719 . We have used first four such voice samples for training and last voice sample for testing. We are also planning to include the facility of real time testing where you can record your own voice sample and test against the already trained model and since we have trained our system using voice sample from Indian people and also separate for male and female while taking care of the possibility of noise this will be a perfect model for speaker recognition especially in context of India.

Data Preprocessing- Data preprocessing is one of the most important tasks for making a perfect training dataset because unprocessed data can lead to overfitting and wrong prediction. In the used dataset there is considerable amount of noise which has been dealt using constant parameter “noisy” which is used in removing the part of sound without any speech. Noisy is a fractional value of the mean amplitude below which there is no speech. This value needs to be increased when using the system in more noisy area.

In preprocessing we also check for the minimum length of the voice sample to be 400ms because below it relevant features can’t be extracted from voice samples and using it is fruitless. We have also made chunks of given input voice samples for making it manageable while finding features.

Feature Extraction :-

Feature Extraction is an integral part of problem solving using machine learning. Feature extraction involves reducing the number of random variables under consideration, making the problem, increasing the interpretability of the problem. When the input data set for a learning algorithm is conspicuously large, and redundancies are suspected to be present in the data, feature extraction can be applied to leave us a with a reduced and representative set of features, often known as the features vector. The techniques for feature extraction that we plan on using are described as under :-

1) **Mel- Frequency Cepstrum Coefficients (MFCC**)- According to Bogert (1963), The power cepstrum of a signal is defined as the squared magnitude of the inverse Fourier transform of the logarithm of the squared magnitude of the Fourier transform of a signal.

Power Cepstrum of Signal $= \left|\mathcal{F}^{-1}\left\{\log(|\mathcal{F}\{f(t)\}|^2)\right\}\right|^2$

{Source:Wikipedia}

The Mel- Frequency Cepstrum involves plotting the system's response on the **Mel scale,** which contains equally spaced frequency bands rather than the linearly spaced bands in the normal cepstrum. MFCC is one of the most widely used features in Automatic Speech Recognition (ASR). The steps for calculating the MFC are as under :-

a) Take the signal and frame it into small frames. An appropriate frame size should be selected to be able to demonstrate the change effectively. Selecting too small or large a frame may lead to problems. The frames are then windowed via the Hamming Window.

b) Take the Discrete Fourier Transform of the signal to obtain a spectrum

c) Map the spectrum obtained above on the Mel Scale. The Mel Scale is given by :-

M(f)= 1125 ln (1 + f/700)

d) Take the logarithms of the powers at each of the Mel frequencies.

e) Discrete Cosine Transform is then applied to the list of the Mel log powers, obtained in the step above.

f) The MFCCs are the amplitudes of the resulting spectrum. The number of coefficients to be used for training varies from case to case.

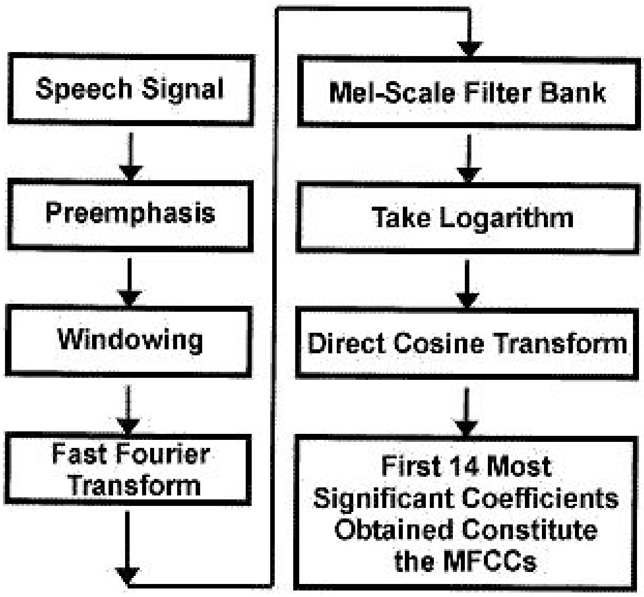
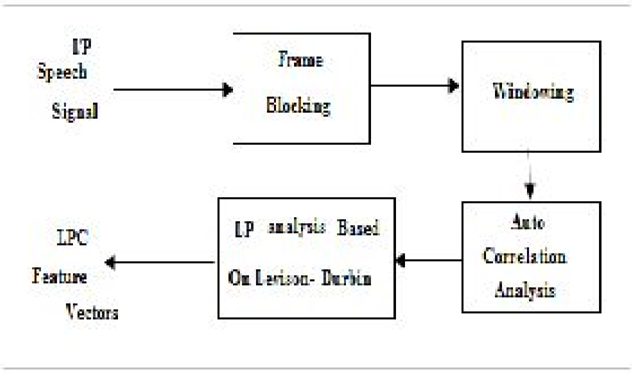


Figure 3.
Steps involved in Mel-frequency cepstral coefficients (MFCCs) extraction.

2) **Linear Predictive Coding (LPC)** - It is one of the most powerful speech analysis techniques and gives very accurate estimates of speech parameters. It involves expressing a current speech sample as a linear combination of the previous signals, i.e, X = Σ $a_i x(n - i)$, where i ranges from 1 to p, where p is the number of previous signals under consideration. For the estimation of the coefficients a_i , the Squared Error Function needs to be minimized. LPC has to be able to withstand transmission errors,as it involves transmission of information of large spectral envelopes. Like MFCC, the input signal is first split into frames, before the coefficients are determined.


I/P Speech Signal
Frame Blocking
Windowing
Auto Correlation Analysis
LP analysis Based On Levison- Durbin
LPC Feature Vectors

Algorithm :-

Random forests :-

- Random forests are an ensemble learning method for classification tasks.

  They work by training a bunch of decision trees (typically around 300, varies from application to application) on a randomized subset of features extracted from the dataset and outputting the mode of the set of classes in the case of classification problems.
- Note that the number of features selected for each are equal to the initial number of features in the dataset and the subset selection is done with replacement.
- Random forest is preferred to tackle over-fitting and thereby produce an accurate model.

Evaluation Techniques :-

Accuracy via confusion matrix can be used to report the efficiency of the trained model.

Technology stack :-

- Python 2.7
- Python libraries:
    - Scipy (for reading and storing wav files)
    - Sklearn (includes an optimized implementation of the random forest classifier)
    - Numpy (for computing the fast fourier transform)
    - Pyaudio (for recording and saving wav files from the mic)