Speaker Identification For Bengali Speakers Using Deep Neural Networks

Riddhick Dalal

*Subhayan Moitra*

*ABSTRACT: The easiest way for two persons to communicate is through speech. Using feature extraction techniques to extract unique features then combining with various models is the way one can identify speakers through machine learning. Many researchers have proposed their way of thinking through implementing different techniques which includes the combination of feature extraction techniques and classification models.*

*We proposed a novel approach which can be used to uniquely identify a speaker through their voice sample. So, our speaker recognition system uses Deep Neural Networks as classifiers and MFCC, LFCC, LPC, Rasta-PLP as feature extraction techniques. The used dataset is a custom dataset consisting of 10 different speakers, speaking Bengali or English mixed with Bengali language and recorded in a natural environment. Our proposed method achieved 98% accuracy, which is better from previous works.*

***Keywords- DNN, CNN , MFCC ,LFCC , LPC , RASTA-PLP***

1. **INTRODUCTION**

Speech is the major part of human communication. It is a one dimensional function of time. Speech contains many different levels of information such that gender, age , emotion , identification of speaker etc. Like fingerprints , iris and face, every person’s voice also contains some unique features. So speech can also be used to identify/recognise a person as a biometric identifier.

We can extract different features like MFCC, LFCC, LPCC ,RPLP ,PNCC from a given speech signal. These features contain unique sequences which can be used to identify speakers.   
In recent times Deep Neural Networks along with ANN, RNN, SVM and CNN performed very accurately in Speaker Identification. Combining Deep Neural Networks along with the various feature extraction algorithms we can build more accurate and efficient models which can identify speakers in real time.

**2.RELATED WORKS**

In recent years a lot of works have been done in Speaker Identification using different techniques and methods.

**2.1 Speaker Identification Using a Hybrid CNN-MFCC Approach:** The architecture used here is CNN combined with MFCC. It identifies speakers without converting it into text and in a noisy environment. It uses DNN for classification and gets an accuracy of 87.5% on a self-made dataset of 60 speakers. CNN is very helpful as it can do both feature extraction and classification. Speaker acknowledgement is done using a neural network here. The used dataset is purely homemade and contains background noises. The reason to use this kind of dataset is to observe real life occurring voices. The voices are taken from classmates and YouTube speaking in Urdu. 20 seconds is the time length of each of these. At first a comparison between 2 approaches is done then combined into a better hybrid approach. First the CNN based approach is used and then the approach that uses MFCC as feature extractor and DNN as classifier. Individually, promising accuracy was not found in the results for an unknown speaker. So, in a hybrid approach the feature of both the models is combined and then DNN is used on that single feature file. According to the results, the CNN approach yielded 75% accuracy and 77.5% precision, the MFCC-DNN approach yielded 80% accuracy and 85% precision, the hybrid approach yielded 87.5% accuracy and 91% precision.

**2.2 Speaker Identification by GMM based i Vector:** A Gaussian mixture model is being used which is built by extracting some acoustic feature vectors from the voice. Compression on the basis of an i-vector yields better predicted results. An order pair of speakers is created where the unknown speaker resides in the first co-ordinate and the test speaker resides in the second coordinate. Voices which are independent of text and language are identified by vocal tract. From the existing speaker model, an unknown speaker is identified using a vocal track as it is identified first for identifying the unknown speaker. It gives score prediction by following the postulates of Bhattacharyya. The GMM creation is done by quantizing the analog signals and then sampling the quantized results for doing pre emphasis on the previous sampled results and then windowing is done and on the result FFT is applied and then some band pass filters are applied for getting an average value and after that MFCC is used for feature extraction from it to get the final GMM result. A probabilistic compression process is applied by using linearity of GMM and generative equations. Simulating the results is being done in two stages. The first stage is thresholding and the next stage is cosine based score predicting. For a particular value of threshold, false accepts can be detected and for the highest predicted score this model yields better results.

**2.3 A review on Deep Learning approaches in Speaker Identification:** Deep learning approaches are more successful in speech recognition and identifying the speaker, than the traditional approaches. The paper aims to promote deep learning implementation techniques for identifying the speaker. It categorized various applications and implementations of Deep Learning (DL) according to the process of identifying a speaker. Deep Neural Network (DNN) is a layer-greedy training technique to train multiple neural networks (NN) of hidden layers, at least three. The method of training DNN is known as Deep Learning (DL). SID (Speaker Identification) is a Natural Language Processing (NLP) technique. Major implementations of Deep Learning are CNN, also known as Convolutional Neural Network, DBN, also known as Deep Belief Networks and SAE, also known as Stacked Auto Encoders. Speaker Identification can be distinguished as two categories. In the first category, identification is done based on the speaker's voice print which is further categorized as a closed set in which the speaker is verified with some existing voice prints and an open set in which a new speaker is verified. In the second category, identification is done based on the level of user control which is further categorized as text dependent and text independent. The two phases of the Speaker Identification process are, training phase and matching phase. At the training phase, the speaker's voice prints are taken for feature extraction and then a model is trained based on those features. At the matching phase, test speaker’s voice prints are taken and the features are extracted and then it is matched with the extracted features of the trained model. GMM (Gaussian Mixture Model) were useful for text independent matching and HMM (Hidden Markov Model) were used for text dependent. There are three categories in which speakers can be identified using DNN. In the first category, DNN is used as a feature extractor and then GMM can be used for matching. In the second category, DNN is used as a classifier for matching and MFCC can be used for feature extraction. In the third category, DNN is used for the entire feature extraction and matching in the Speaker Identification process. Here PCA (Principal Component Analysis) is often used for dimensionality reduction. Stacked Bottleneck Features (SBN) is also used for dimensionality reduction. The Bottleneck layer has significantly lower dimensionality as two cascading Neural Networks are used. i-vector based approaches also give food results in the field of Speaker Identification. DL can be used to extract i-vectors as well as a classifier after extracting the i-vectors. Unsupervised Deep Belief Networks (UDBN) are suitable for unknown Speaker Identification which uses i-vectors.

**2.4 Comparative Study of Different Techniques in Speaker Recognition: Review:** Speakers are always recognised by the individual information which is already present in their speech signals. Speech gives information about the emotion and identity of the speaker. Every human voice has unique vocal characteristics like pitch, frequency, and tone, which has to be extracted and then enrolled by training the voice model and then acknowledged or confirmed. Feature extraction is the technique in which distinctive elements are distinguished from the information set. It is done after the pre-processing. A basic speech recognition system has the input speech signal which is pre-processed and then features are extracted for classification and after that, a decision for speaker recognition is taken. MFCC is the most popular feature extraction technique. It is the classical approach for analyzing speech signals. MFCC has a high rate of performance and low complexity. LPC system is used to decide fundamental speech parameters. Previous speech tests are blended and approximated. LPC technique is reliable and accurate for providing parameters for representing vocal tract. LPC has good computational speed and encodes speech at low bitrate. DTW algorithm is used to determine similarities between two time series. It depends on element programming. Lt is a coordinate acknowledgement strategy. Delta and double delta of MFCC features can also be used as extracting techniques. For classification GMM, ANN and SVM can be used. GMM is useful when less memory and dataset are used. ANN is useful when extracting features and modeling is combined into a single network. SVM is effective in binary classification.

**2.5 Speaker Verification using Convolutional Neural Networks:** The developed architecture for speaker verification is a CNN based architecture which captures and discards the information of the speakers and non-speakers simultaneously. Background model is created by training to differentiate between speakers. Previous approaches averaged the results from the background model. The problem is overturned here by using the Siamese framework for fine tuning the trained model. Discriminative feature space is generated for differentiating the same and different speakers. This method outperforms the previously formed traditional methods for verification. Three phases are involved in the general procedure for speaker verification namely: i) Development ii) Enrolment iii) Evaluation. In this work, Siamese Neural Network is used to operate. Public VoxCeleb dataset is used for the experiments. Here, 140000 utterances are there for 1211 speakers and for testing there are 6000 utterances for 40 speakers. The audios have different ethnicities, accents and also have background chatter, channel noise, overlapping speech and different recording qualities. In the input pipeline, the SpeechPy library is used for feature extraction. VGG-M architecture is being used and the size is being reduced for training it faster. In time dimension, performing pooling degraded the performance. The Siamese architecture is used while verifying which consists of two identical CNNs. The general idea behind this is if there are two pairs which belong to the same identity then the common feature subspace distance would be close. TensorFLow is used and the model is trained on the NVIDIA Pascal GPU. GMM-UBM and I-vectors models have been used for comparison. EER is the Equal Error Rate which is the least for this architecture as compared to other models. The EER result is 10.5.

**2.6 Text-independent speaker recognition using LSTM-RNN and speech enhancement:** In the given paper speaker is recognised in a text independent manner in presence of noise and reverberation. For feature extraction, MFCC, spectrum and log spectrum are used. For classification, LSTM-RNN (Long-Short Term Memory Recurrent Neural Network) is used. When MFCC is used, recognition rate is 95.33%, when spectrum or log spectrum is used, recognition rate is 98.7%. Spectral subtraction and wavelet de noising, speech enhancement techniques are used for improving the performance of recognition. The DNN is less effective than LSTM-RNN in the acousting model. RNNs are cyclic. MFCC techniques are sensitive to external noises. Here spectrum and log spectrum are used as extracting features and then compared to MFCC with and without noise. Short Time Fourier Transform (STFT) is applied on the signal for computing the spectrogram. A RNN is a feed forward network and it faces vanishing gradient problem. It consists of one input layer, one output layer and 1-2 hidden layers. The system decides the output based on the previous and on the present inputs. Noise reduction is the key point of speech enhancement which can be achieved by techniques like spectral subtraction and wavelet de noising. The proposed system recognises speakers from any given spoken phrases. At first the speech signals are pre-processed and used as input for LSTM-RNN. Then some features are extracted for training the LSTM-RNN. The data used here is a subset of Chinese Mandarin Corpus Dataset. This dataset was recorded using cell phones. A total of 100 utterances from 5 female speakers were taken, from which 70 were used for training and 30 for testing. 3 seconds reverberation time was used. Accuracy was best when spectral subtraction was at 30 dB. When the utterances are undistorted, LSTM-RNN had an accuracy of 98.7% and when distorted, using spectrum features, 90% accuracy was achieved and when reverberation was added, using spectrum features 62.7% accuracy was achieved.

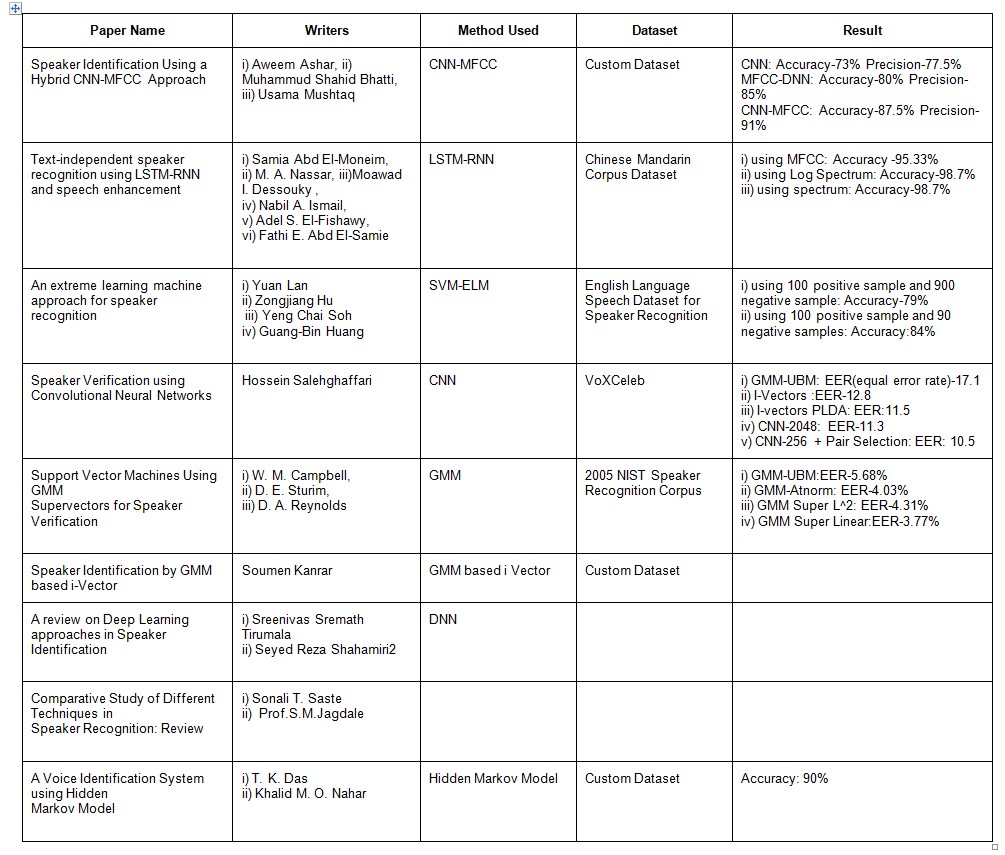
**2.7 Support Vector Machines Using GMM Supervectors for Speaker Verification:** In speaker recognition, GMMs are extremely successful. GMM Mean Super vectors are formed by stacking the means from the output of the GMM model. The GMM super vector in a SVM classifier is used here. Two new SVM kernels are proposed which are based on the GMM models’ distance metrics. Speaker recognition is a two-class problem, so is SVM, which is a two-class classifier. Two natural methods are shown for calculating the distance between GMM super vectors. The ideal outputs for SVM are either 1 or -1 and it depends upon the support vector whether it lies in class 0 or 1. The training of the GMM UBM is performed by MAP adaptation. The first proposed kernel is the GMM Super vector Linear Kernel. Here, only a single inner product has to be computed between GMM and target model for obtaining a score. The second proposed kernel is GMM L2 Inner Product Kernel. Here, the assumption is that mixture components are far from each other. It uses inner products of function space. Experiments are performed on the 2005 NIST Speaker Recognition (SRE) corpus. EER (Equal Error Rate) and minDCF (minimum Decision Cost Value) are used as metrics for evaluation. RASTA filtering is used for processing Cepstral vectors. 8 GMM super vectors from eight conversations were produced for target speaker enrolment. Model compaction was applied for the Linear Kernel for obtaining a smaller representation. The determinant was discarded due to ill conditioning in the L2 Inner Product Kernel. Standard GMM configuration is outperformed by the Linear GMM Super vector kernel.

**2.8 An extreme learning machine approach for speaker recognition:** Extreme Learning Machine (ELM) is used in this paper for verification of text independent speakers and is compared to the SVM classifier. ELM are extremely fast learners. ELSDSR corpus database is used and MFCC was extracted as input for ELM and SVM. Variants of ELM are, i) Optimization based ELM, ii) Regularized ELM, iii) Kernelized ELM. They can run faster than SVM, Optimization based ELM is very much similar to SVM but it does the work with fewer optimization constraints. For making the resultant solution stabler, a positive value of 1/λ is added while calculating the output weights in the Regularized ELM. Kernelized ELM comes into play when the feature mapping of the hidden layer is unknown. The user doesn’t need to know the feature space and its dimensionality. From the database ELSDSR (English Language Speech Database for Speaker Recognition) corpus voice messages were collected from 22 speakers of age varying from 24 to 63. The training set had 154 utterances and the testing set had 44 utterances. Training data had a duration of 83s each and testing data had a duration of 17.6s each. After MFCC, the speech signals from the 20 speakers were converted to 28 dimensional samples. The evaluation of SVM and ELM classifiers and its variants had two stages. First stage builds three classifiers, SVM, Optimised based ELM and Regularized ELM. Second stage compares the performance of the classifiers with the ROC curve. Both Optimized ELM and Regularized ELM are found to spend less time than SVM for training. Optimized ELM are much better in classes 7 to 10 and Regularized ELM are better for all classes 1 to 10. Kernelized ELM classifiers spend least training time for all classes. So, it can be concluded that ELM classifiers and its variants have better performance than SVM classifiers.

**2.9 A Voice Identification System Using Hidden Markov Model:** In this experiment MFCC technique has been used over the voice signals to extract features and a set of feature vectors was created. To train the and classify the features The Vector Quantization techniques were used.

A Hidden Markov Model is a statistical Markov Model in which the system being modeled is assumed to be a Markov process with unobserved states.

***Table 1:*** *Summary Of Related Works*



The system was mainly divided into two modules- a) Voice recognition b) Speech Recognition.

Modeling: The modeling was done over the features extracted from the mfcc and vq methods. For each HMM state there may be a set of output symbols which can be described as output probabilities, and a finite number of states. A set of probabilities known as transition probabilities controls the relation between one process and the transitions among the states

An observation is produced by the other process representing the current state, for each instance of time while assuming the process to be in some state .

Result: For the result an one dimensional power spectrum was plotted which indicates power of a signal at each frequency that it contains. With this model Speech Recognition has been accomplished with a success rate of 90% .

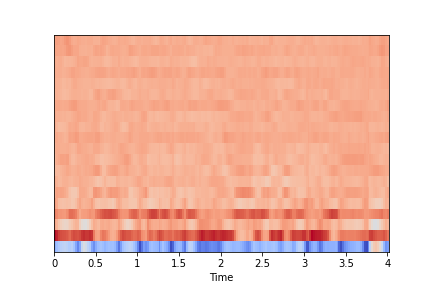
**3. DATA PREPARATION**

For this paper we created our own dataset. We took voices from 10 different people, aged between 18 to 50. For each person there are 750 audio samples with a duration of 4 seconds each. All the samples are in Bengali or Bengali-English mixed language. The male-female ratio of this data set is 1:1. All the voice samples are collected in natural environmental conditions using different types of recording devices.

**4. FEATURES EXTRACTION**

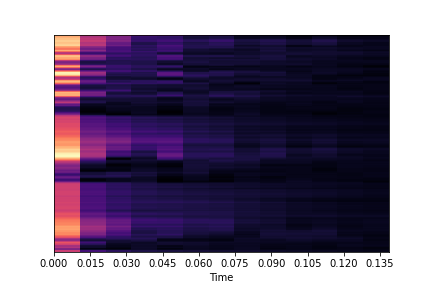
In this paper we worked with four types of features, these are Mel-frequency cepstral coefficients(MFCCs), Linear prediction coefficients(LPCs), Linear Frequency Cepstral Coefficients(LFCCs), Rasta Perceptual Linear Prediction coefficients(RPLP).

**4.1 MFCC:** Mel Frequency Cepstral represents short-term power spectrum of a sound . In this paper we extracted 20 mfccs for each audio clip using Librosa library.

****

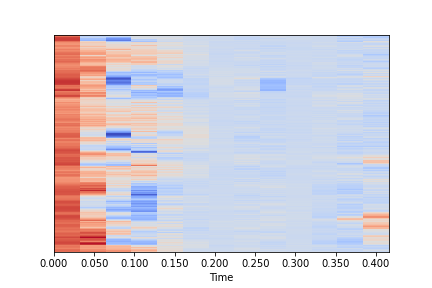
***Fig 1:*** *MFCC Spectrogram*

**4.2 LPC:** Linear Prediction Coefficients calculate power spectrum of a given audio signal. In this paper we extracted 13 lpcs for each audio clip using Spafe library.



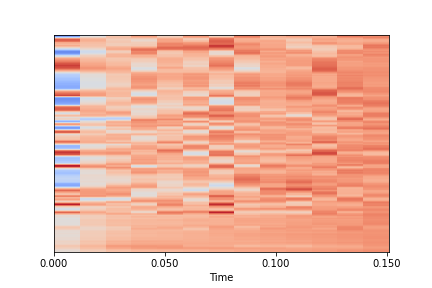
***Fig 2:*** *LPC Spectrogram*

**4.3 LFCC:** Linear Frequency Cepstral Coefficients are very identical to MFCCs, except it covers all frequency ranges equally and gives them equal importance. In this paper we extracted 13 lfccs for each audio clip using Spafe library.



***Fig 3:*** *LFCC Spectrogram*

**4.4 RPLP:** Rasta PLP analysis is done using Single Value Decomposition. In this paper we extracted 13 Rasta PLP coefficients for each audio clip using the Spafe library.



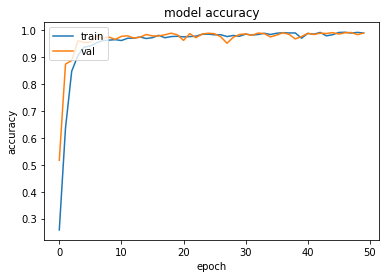
***Fig 4:*** *Rasta-PLP Spectrogram*

**5. PROPOSED MODEL**

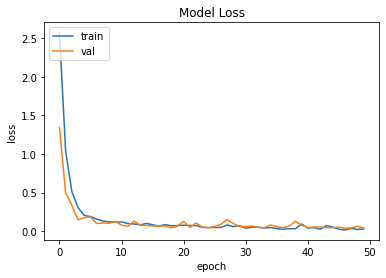
First we created four fully connected Deep Neural Network Models. Then we combined these models for ensemble learning, where the mode of the predictions from all four models will be counted as the final prediction. Thus it increases the correct prediction probability.

**5.1 Model 1:** It has two 2D Convolutional Layers with 128 perceptrons each which are followed by another two 2D Convolutional Layers with 64 perceptrons each. Then It has two fully connected dense layers with 64 perceptrons and 32 perceptrons respectively and an output layer with 10 possible outcomes .

We used the MFCCs as inputs for this model. After 50 epochs this model gives training accuracy of 98.95% with loss 0.0301 and validation accuracy of 98.93% with validation loss 0.0413.



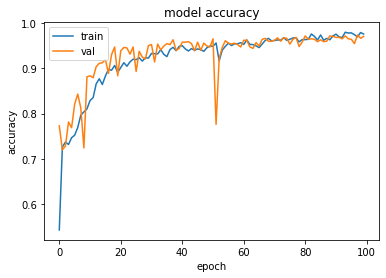
***Fig 5:*** *MODEL 1 Accuracy Plot*



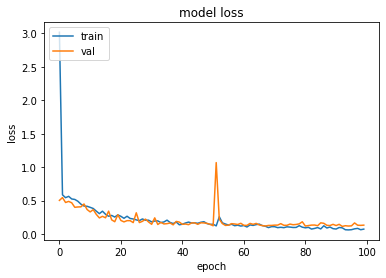
***Fig 6:*** *MODEL 1 Loss Plot*

**5.2 Model 2:** It has one 2D Convolutional Layer with 128 perceptrons which is followed by another two 2D Convolutional Layers with 64 perceptrons and another 2D Convolutional Layer with 32 perceptrons. Then It has a fully connected dense layer with 32 neurons and an output layer .

We used the LFCCs as inputs for this model. After 100 epochs this model gives training accuracy of 97.56% with loss 0.0742 and validation accuracy of 97.03% with validation loss 0.1336.



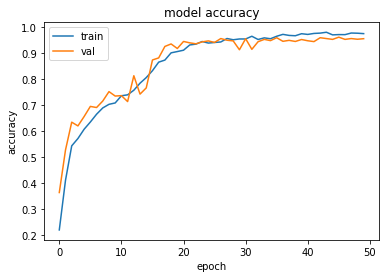
***Fig 7:***  *MODEL 2 Accuracy Plot*



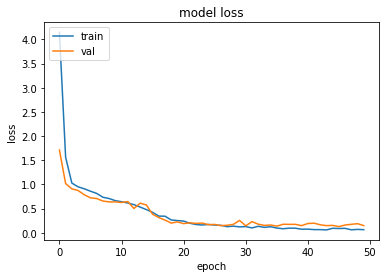
***Fig 8:*** *MODEL 2 Loss Plot*

**5.3 Model 3:** It has a 2D Convolutional Layer with 128 perceptrons followed by two 2D Convolutional Layers with 64 perceptrons and 32 perceptrons respectively. Then It has two fully connected dense layers with 64 and 32 perceptrons, and an output layer with 10 possible outcomes .

We used the LPCs as inputs for this model. After 50 epochs this model gives training accuracy of 97.59% with loss 0.0629 and validation accuracy of 95.63% with validation loss 0.1469.



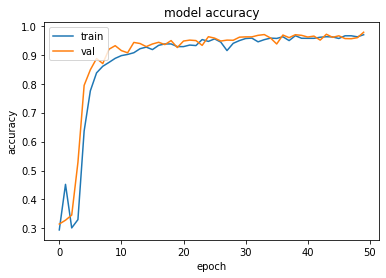
***Fig 9:*** *MODEL 3 Accuracy Plot*



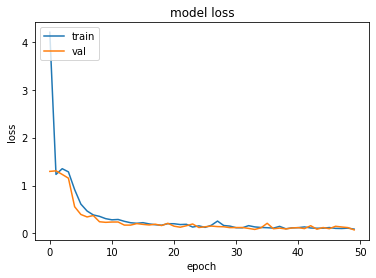
***Fig 10:*** *MODEL 3 Loss Plot*

**5.4 Model 4:** It has a 2D Convolutional Layer with 128 perceptrons each which is followed by another two 2D Convolutional Layers with 64 perceptrons each. Then It has two fully connected dense layers with 64 and 32 perceptrons respectively. Then it has an output layer.

We used the Rasta-PLPs as inputs for this model. After 50 epochs this model gives training accuracy of 96.91% with loss 0.0911 and validation accuracy of 97.76% with validation loss 0.0744.

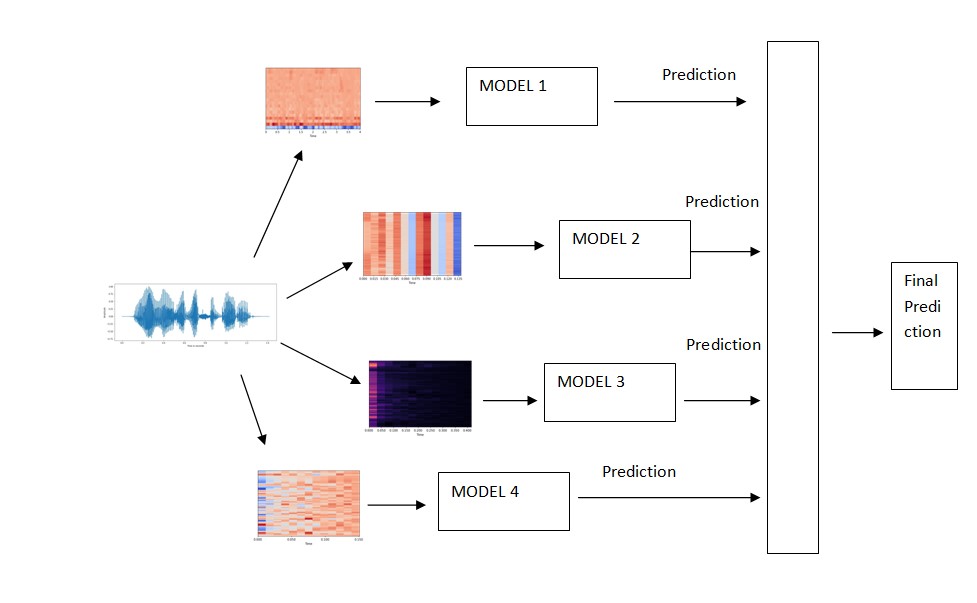


***Fig 11:*** *MODEL 4 Accuracy Plot*



***Fig 12:*** *MODEL 4 Loss Plot*

**5.5 Appling Ensemble Learning:** Now for a given audio sample we will extract all the four above mentioned features and feed those features to their respective model for predictions. Then we consider the mode of the predictions as the final prediction for given audio data.



**6. RESULTS**

| **Feature** | **Epochs** | **Training Accuracy** | **Validation Accuracy** |
| --- | --- | --- | --- |
| MFCC | 50 | 98.95% | 98.93% |
| LFCC | 100 | 97.56% | 97.03% |
| LPC | 50 | 97.59% | 95.63% |
| RPLP | 50 | 96.91% | 97.76% |

The accuracy will increase further after implementing the Ensemble Learning, as the probability of getting wrong predictions will be reduced in this process.

***Fig 13:*** *Diagram of Ensemble Learning Algorithm*

**7. CONCLUSION**

The model we proposed consists of Deep Neural Networks , which take MFCC, LFCC, LPC and Rasta-PLP spectrograms as input and achieve validation accuracy of 98.93%, 97.03%, 95.63% and 97.76% respectively.  Further using ensemble learning techniques we can combine the outputs of these models and identify the speaker more accurately . Currently we worked with a dataset of 10 unique speakers, in future we can expand the dataset to create a more general and accurate speaker identification system which can identify speakers in real time.