CNN based speaker recognition in language and text-independent small scale system

Rohan Jagiasi
Department of Information Technology
VES's Institute of Technology
Mumbai, India
rohan.jagiasi@ves.ac.in

Shubham Ghosalkar
Department of Information Technology
VES's Institute of Technology
Mumbai, India
shubham.ghosalkar@ves.ac.in

Punit Kulal
Department of Information Technology
VES's Institute of Technology
Mumbai, India
punit.kulal@ves.ac.in

Asha Bharambe
Department of Information Technology
VES's Institute of Technology
Mumbai, India
asha.bharambe@ves.ac.in

***Abstract* — Speaker Recognition is the ability of the system to recognize the speaker from the set of speaker samples available in the system. It is of 2 types, one uses a keyword, called text-dependent systems, and another one can recognize the voice in any language/text, also called as text-independent speaker recognition. In this paper, a text-independent, language-independent speaker recognition system is implemented using dense & convolutional neural networks. Speaker recognition has found several applications in upcoming electronic products like personal/home assistants, telephone banking and biometric identification. In this paper, we explore a system that uses MFCC along with DNN and CNN as the model for building a speaker recognition system.**

Keywords* — *speaker recognition, neural network, voice sample, language-independent speaker recognition, independent speaker recognition system

I. INTRODUCTION

Speaker recognition is the identification of a person from the characteristics of voices. Recently, due to a large increase in the field of smart devices, there have been many studies conducted based on how to identify the speaker so as to be able to give a personalized experience to its users. However, most of the studies are dependent on the fact that the user speaks a keyword in order to activate the device used to identify them (Text-dependent speaker recognition) [1]. This makes interacting with the devices a bit monotonous. This study has been done in order to overcome those keyword and language barriers and be able to recognize the user whatever he speaks.

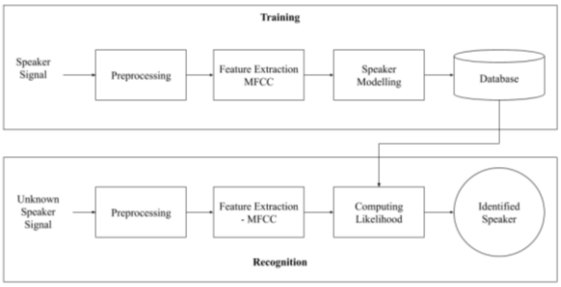
II. EARLIER WORK

There are various approaches to solving the speaker recognition problem. The solution is defined by two parameters : The feature of the voice signal to be used like Mel Frequency Cepstral Coefficients (MFCC), Linear Prediction Cepstral Coefficient (LPCC), Perceptual Linear Prediction (PLP) and the modelling technique used to learn the voice samples like Artificial Neural Networks (ANN), Gaussian Mixture Model (GMM)[12], vector quantization etc. Studies have also been done where multiple techniques are combined, such as using different models for feature extraction and classification.[8] Of these choices, majority approaches use MFCC as the feature to be used since it is most effective for speaker recognition[11].

Using MFCC features and, [10] studied phenomic based speaker recognition systems. However, the accuracies seem to fluctuate based on the words and syllables pronounced. Also, relatively lower accuracies have been achieved using GMM[2], Vector Quantisation[3] and Deep Neural Networks (DNN)[4]

III. OUR APPROACH

Fig 1 - High-level architecture



Expected Input: The speaker signal is expected to be in an 8-44Khz uncompressed audio file in Wav format.

- Preprocessing block: In this block, the voice samples are preprocessed to minimize/filter the noise content and eliminate the silent parts in the signal.
- Feature Extraction block: This block is responsible for extracting MFCCs. It then reduces the dimension of these vectors and passes it to the speaker modelling block.
- Speaker modeling block: This block takes MFCCs as input and builds a model.
- Computing Likelihood: This block is present only in the recognition phase. It searches for a match in the trained model to identify the speaker and returns it as output.

Expected Output: Name of the speaker.

As mentioned before, we use neural networks which is a machine learning approach.
Like every machine learning system, it has 2 phases - training and testing.

The system starts by recording the user speaking a paragraph from any article for a specific amount of time (1 minute in our case). The voice signal is then preprocessed using the SoX framework for eliminating noise and silence from the audio. MFCC (Mel Frequency Cepstral Coefficients) is then extracted from the processed voice signal using a library. These features are then fed to a neural network. After all the speakers have been enrolled, the model is tested.

Testing is fairly simple, the user speaks a couple of words which are recorded, pre-processed and the features extracted are tested on the Convolutional Neural Network (CNN) model built during the training phase of the system.

A. Pre Processing

We use Sound Exchange(SoX), an audio processing framework, to remove the noise and silent parts from the audio. The silence is removed by using a threshold; anything below the threshold is removed.

The human voice has a fundamental frequency range of 85 to 180 Hz for male and 165 to 255 Hz for children and females. However, when a person speaks, their frequency isn't fixed. It varies for different words. The energy also spreads to nearby frequencies which gives a diminishing effect to sound, necessary to utter certain specific words in certain languages. However, this spread needs to be limited in order to distinguish between voice and noise.

The application uses a sampling rate of 44.1Khz (subject to availability) for optimum quality audio files for better results. It has been noticed that the majority of the information is stored in the first 0-8000 Hz bandwidth. A low pass filter is, therefore, applied to remove higher frequency sounds which is mostly ambient noise. The obtained audio signal is saved in the application and will be referred to as processed audio from here on.

As given below, fig 2 shows the original raw voice sample. The processed sample is shown in figure 3

Fig 2 - Original Voice Sample

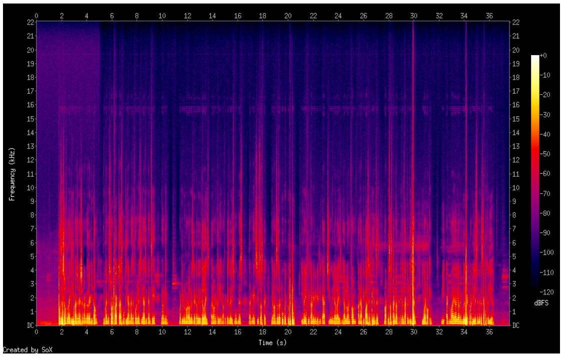
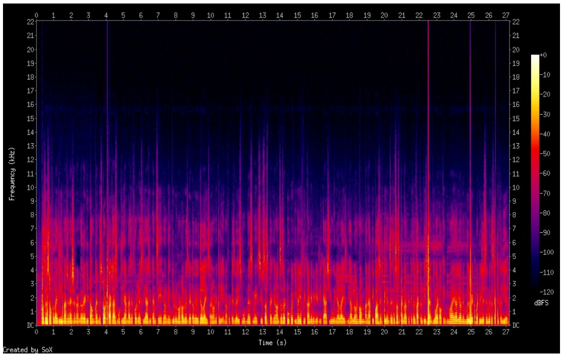


Fig 3 - Processed using SoX script



B. Feature Extraction

[4] This frequency warping can allow for better representation of sound. MFCC has found application in various speech recognition applications.

We have used "python_speech_features", an audio processing library that extracts MFCC features from a given Wav format audio file. Filter bank energies, which is an intermediate step in the extraction of MFCC, is also used along with MFCC feature vectors. The frame size is 32ms with a stride of 16 ms. Only the initial 13 MFC coefficients are useful [10] since the later ones are nearly zero.

C. Neural Network

There are two types of approaches to training the neural network using the MFCC vectors

Every speaker read a paragraph from the english newspaper for the training set. These 1 minute training samples were then pre-processed and each frame was fed to the neural network.

Nearly every paper suggests that features of up to 40 frames be stacked and given collectively as input. This is necessary for speech dependent system, where the sequence of words/frames matter. Since we were focussing only on the text-independent aspect and are not concerned with the order of frames, the size of the input vector corresponds only to the size of features of 1 frame. Every frame has 52 features which include 13 MFCC, 13 deltas derived from MFCCs, 13 acceleration derived from deltas [7] and 13 filterbank energies.[1]

We have adopted an approach which tests two learning models, namely DNN and CNN. We have used DNN because as stated in [5], DNN provides better noise immunity over the next best performing model, i.e. GMM. CNN is tried because CNN is innately used for identifying patterns in the data/features and scales well which is of essence in this problem.[6][9]

The neural network has been implemented on the tensorflow platform using Keras APIs.

DNN: The first layer consists of 3000 nodes followed by 4 layers of 100 nodes each and a dropout of 0.3 between each of them. The dropouts help in avoiding overfitting of data.

CNN: The CNN contains 4 layers, 3 convolution layers and a dense layer in the end. First 2 layers use 52 convolution kernels with filter lengths (window size) of 13 and 7 respectively. The third convolution layer uses 13 output kernels with a window size of 3. The output is then flattened and a dense layer with 1000 nodes follows. 0.25 of dropout is added at this stage to avoid overfitting. Then, the final output layer follows. All layers use ‘tanh’ as the activation function, except the final layer, which uses ‘softmax’.

The loss was calculated using “Categorical Cross Entropy” and the optimizer used was “Adam”

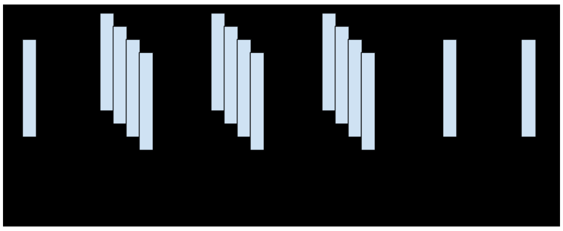


Fig 4 - Diagram of CNN Model

IV. DEPLOYMENT

The application was deployed using the flask framework for the front end. The recordings were taken by the web app. Training samples were of 1 minute length and test samples were recorded for 10 seconds. It was deployed on a laptop with an intel i3 dual-core processor with 8 GB RAM, which could be accessed via a computer or mobile phone.

V. RESULTS AND DISCUSSION

The voice sample is recorded and processed as above. 50 frames are extracted from the processed audio and all of them are tested on the neural net. The probability outputs of all frames are added and the speaker with the highest probability is the output. This test set included voice samples from the speaker in multiple languages, mainly English, Hindi and Marathi.

Table 1 - Observations for voice recorded under lab conditions (Speaker recognition dataset from openslr by Tsinghua University)

No. of Speakers	**DNN accuracy**	**CNN accuracy**
5	70	76
10	67	68
15	70	72
20	55	58
35	55	58
50	61	71

Table 2 - Observations for Real-World Voice Samples

No of Speakers	**DNN accuracy**	**CNN accuracy**
2	100	100
3	75	75
4	75	87

5	80	90
6	65	78
7	64	77
8	58	75

Fig 5 - Observation for Voice recorded under Lab conditions (Speaker Recognition dataset from openslr by Tsinghua University)

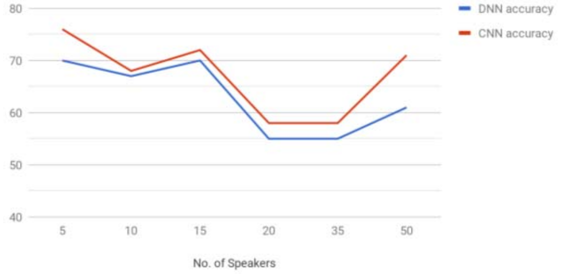
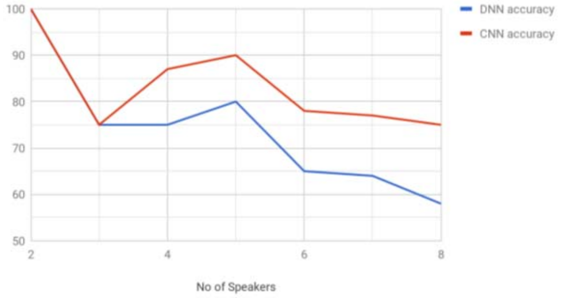


Fig 6 - Observation for Real-World Voice Samples



VI. OBSERVATION

CNN: Upto 10 speakers can be fed into the net for a decent accuracy of 75-80% for real-world samples (Table 2 and fig 6). For voice recorded in lab conditions, the model gives an accuracy of 70% for up to 50 speakers with some inconsistencies as seen from table 1.

DNN: DNN gives an accuracy of 75-80% for 5 speakers and then its performance degrades (Table 1 and Fig 5).

This is consistent with the fact that CNN is a learning model that excels at identifying patterns in the input and can scale much better than DNN.

VII. CONCLUSION AND FUTURE SCOPE

The language independent, text independent speaker recognition system was developed with the idea that the future smart devices should have the ability to recognize their user's voice without the boundaries of languages and keywords. An accuracy of 75-80% was achieved using the CNN model (Table 2). Further studies can be conducted to improve the accuracy of the model and to scale up the number of users.

Speaker recognition can be applied to multiple domains across the industry. In the IoT world, it can change the way we interact with smart devices. Without saying "Ok Google" or "Hey Siri", one would be able to communicate with such devices in natural language. Also, the device would be able to personalize the experience for the recognized user. Speaker recognition is also useful in biometric verification. In its current state, it can be used to build a simple lab attendance system which would not require any specialized biometric device.

REFERENCES

[1] Zhang, Chunlei, and Kazuhito Koishida. "End-to-End Text-Independent Speaker Verification with Triplet Loss on Short Utterances." *In Interspeech*, pp. 1487-1491. 2017.

[2] Li, Chao, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu. "Deep speaker: an end-to-end neural speaker embedding system." *arXiv preprint arXiv:1705.02304* (2017).

[3] Geeta Nijhawan, Dr. M. K. Soni, July 2014, "Speaker Recognition using MFCC and Vector Quantisation", *International Journal on Recent Trends in Engineering and Technology*, Vol 11. No. 1

[4] Shahenda Sarhan, Mohamed Abu ElSoud, Nagham Mohammed Hasan, July 2015, "Text Independent speaker identification based on MFCC and Deep Neural Networks", *https://www.researchgate.net/publication/*291165354

[5] Ahilan Kanagasundaram, David Dean, Sridha Sridharan, Clinton Fookes, October 2016, "DNN based Speaker Recognition on Short Utterances", *arXiv:*1610.03190

[6] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, Zhenyao Zhu, 5 May 2017, "Deep Speaker: an End-to-End Neural Speaker Embedding System",*arXiv*:1705.02304v1 [cs.CL]

[7] Zhenhao Ge, Ananth N. Iyer, Srinath Cheluvaraja, Ram Sundaram, Aravind Ganapathiraju, 7-8 September 2017. "Neural Network Based Speaker Classification and
Verification Systems with Enhanced Features*", Intelligent Systems Conference* 2017

[8] Fred Richardson, Douglas Reynolds, Najim Dehak, October 2015, "Deep Neural Network Approaches to Speaker and Language Recognition", *IEEE Signal Processing Letters*, Vol 22, No. 10

[9] Amirsina Torfi, Nasser Nasrabadi, Jeremy Dawson, 6 November 2017, "Text-Independent Speaker Verification Using 3D Convolutional Neural Networks", *arXiv*:1705.09422v4

[10] Zhang, Shi-Xiong, Zhuo Chen, Yong Zhao, Jinyu Li, and Yifan Gong. "End-to-end attention based text-dependent speaker verification." *In 2016 IEEE Spoken Language Technology Workshop (SLT),* pp. 171-178. IEEE, 2016.

[11] Qiyue Liu, Mingqiu Yao, Han Xu, Fang Wang, 2013, "Research on Different Feature Parameters in Speaker Recognition", *Journal of Signal and Information Processing*, 4, 106-110

[12] Athira Aroon, S.B. Dhonde, 2015, "Speaker Recognition System using Gaussian Mixture Model*", International Journal of Computer Applications (0975 – 8887),* Volume 130 – No.14