MULTILINGUAL SPEAKER RECOGNITION

Ayush Singh, CSA, IISc
Neha Sharma, DESE, IISc

***Abstract*—The project aims to improve speaker recognition systems through various experiments. Three models (KNN, CNN, and LSTM) will be built and compared using MFCC feature extraction. The effect of sentence structure and content on speaker recognition and the impact of noise on the model's accuracy will be studied. Multilingual speaker recognition will be explored using different feature extraction techniques. Gender bias in speaker recognition will also be analyzed to develop more inclusive models. The goal of these experiments is to enhance accuracy and develop more robust and inclusive speaker recognition systems.**

I. Introduction

Multilingual speaker recognition and language identification are key to developing spoken dialogue systems that can function in multilingual environments. In India, which officially recognizes more than twenty-five languages and whose citizens almost without exception speak more than one of these languages fluently, the development of such a multilingual system is an especially relevant Challenge. This multilingual system should then be adapted to the target language with the help of a language identification system. In this paper, we have developed Speaker Recognition systems based on continuous speech recognition and evaluate these for two Indian regional languages, i.e. Hindi and English spoken by 25 speakers in each language.

This project aims to develop a multilingual speaker recognition system that can recognize the identity of a speaker based on their speech, independent of language. Speaker recognition systems extract individual characteristics from speech utterances to acknowledge the speaker's identity. In this project, we will use neural network models and MFCC feature extraction to improve speaker recognition accuracy.

We will build three models (KNN, CNN, and LSTM) and compare their performance using MFCC feature extraction for speaker recognition. This will help us determine which model best suits our problem and provides the highest accuracy. We will explore how two different types of sentences VCV and Non-VCV sentences, affect speaker recognition by recording the voices of 5 individuals speaking 15 sentences each in English. We will then compare our models' accuracies on two datasets, this will help us understand how sentence structure and content can affect speaker recognition performance.

Our work was motivated by helping early or mid-stage Alzheimer's patients identify who is speaking to them over the phone when multiple people are on the other end. Alzheimer's patient forgetting their loved ones hurts the patient and their family. We apply a deep neural network to identify speakers in noisy environments, which can be used in a phone app to recognize who is currently speaking, show their picture, and give a description to jog the patient's memory. So, we will add noise to the dataset obtained above and compare our models' performance in noisy environments. We will try to modify our models to make them more robust to noise. This will help us understand how noise affects speaker recognition and how we can improve our models to handle noisy environments.

We will perform multilingual audio classification using our existing models and mfcc feature extraction on 25 individuals speaking 15 sentences each, 15 in Hindi and 15 in English. This experiment will help us determine if our existing models can handle multilingual audio classification using only mfcc feature extraction.

If time persists we will also try experimenting with other feature extraction techniques such as lpcc, log area ratio, etc., for our multilingual speaker recognition task. This experiment will help us understand if other feature extraction techniques provide better accuracy than mfcc feature extraction.

Lastly, we will analyze if speaker recognition is biased towards a particular gender using the dataset obtained. This will help us understand if our models are biased towards a specific gender and how to improve them to be more inclusive. Overall, we aim to enhance speaker recognition accuracy and develop more inclusive and robust systems to handle multilingual speaker recognition. The simulated model of a neural network-based multilingual Speaker recognition system for Indian languages Hindi and English has been developed to achieve this goal.

II. Related Work

As a framework from which to build up our implementation, we started with a foundational approach outlined by [Ge et al.] in their paper on Neural network-based classification. This article demonstrates the efficacy of a standard multi-layer perceptron in identifying a single person's voice from a group of two hundred speakers. Despite having considerably less data, we achieved a similar performance level by collecting more data per person.

Much research has been done for multilingual speaker recognition systems using ANN, and there are using different models like statistical methods Hidden Markov Model (HMMs), and Harmonic Product Spectrum (HPS).

The future work section from Alfredo's and Manish's projects suggests implementing a speaker recognition model on a noisy dataset. An approach by Mclaren et al. explores a novel method to leverage MFCC to transform our model. This approach introduces two new data-driven methods of utilizing DCT coefficients for speaker recognition. Nonetheless, this alternative is very complex, and we decide to explore the MFCC coefficients.

An interesting approach is the one by Sainath et al, he proposes that it is possible to modify waveform based model which would be similar to how we try to leverage MFCC in our fully connected model such that the accuracy of this model is equivalent to the Convolutional, long short term memory deep neural network (CLDNN). Although his approach is worth imitating, we will try to create a CNN and LSTM model to achieve higher accuracy.

III. Dataset

The collection of speech utterances consists of sentences in different languages recorded by 25 people. Sound wave files (.wav) are created using a microphone connected to a Personal computer at a sampling rate of 16 KHz with the mono channel. For this purpose, 30 VCV sentences (15 English and 15 Hindi) are recorded from 25 speakers (20 males and 5 females) in 2 different languages, Hindi and English. The sentences are in such a format that every consonant succeeds a vowel in each word and vice versa. The main reason behind this is that vowel sound is always generated with the pronunciation of any letter. Silence and noise are removed from the speech signal. There are two steps in Preprocessing. One reduces noise, and the second separate words from sentences (by removal of silence between words). The proposed approach is done in various steps like the Collection of speech utterances of different speakers of different languages, Preprocessing, Feature extraction, Neural Network training, and Testing, as shown in the figure below.

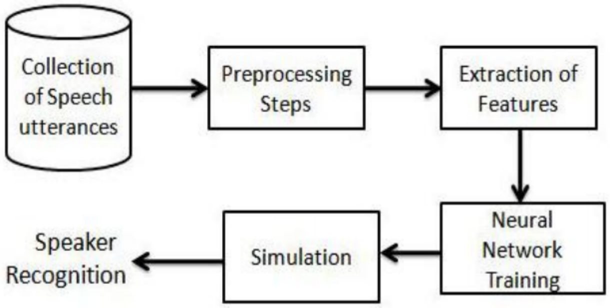

Fig. 1. Various steps in this approach

IV. Speaker Recognition with different Models

A. KNN Model

K-Nearest Neighbors (KNN) is a non-parametric algorithm that can be used for speaker recognition tasks. The main idea behind KNN is to classify new input data based on the class of the nearest neighbors in the feature space. In the case of speaker recognition, the feature space can be represented by the voice characteristics of different speakers.

The KNN algorithm works by first training the model with a set of labelled data, which includes voice recordings of different speakers along with their corresponding labels. The training involves extracting relevant features from the voice recordings, such as storing the mfcc feature vector for each recording. These feature vectors are used to build a database of voice characteristics for each speaker.

When a new voice recording is an input into the system, the KNN algorithm calculates the distance between the feature vector of the input recording and the feature vectors of all the labelled recordings in the database. The KNN algorithm then selects the K nearest neighbours in the feature space based on the calculated distances, where K is a pre-defined hyperparameter. The input recording is then classified based on the majority class of its K nearest neighbours in the feature space. In the context of speaker recognition, the KNN algorithm can identify the speaker of the input recording by selecting the speaker with the highest frequency among its K nearest neighbours.

The performance of the KNN algorithm heavily depends on the choice of hyperparameters, such as the value of K and the distance metric used to calculate the distance between the feature vectors. The optimal choice of hyperparameters can be determined through a grid search or a similar optimization technique.

B. CNN Model

CNNs can extract important features from the audio signals, which can be further processed and classified to identify the speaker.

CNNs are particularly good at extracting localized features from an input signal. For example, in an audio signal, a CNN can identify the unique characteristics of a speaker's voice from specific segments of the input signal.

CNNs can be scaled up or down depending on the size of the input signal, making them flexible for different types of speaker recognition tasks. For example, a small CNN can be used for speaker verification, while a larger CNN may be required for speaker identification.

A Sequential model is created to add all the layers in sequential order. The first layer is a Conv2D layer with 8 filters, a kernel size (2, 2), and a ReLU activation function. This layer takes in the input shape of the data and convolves the input with 8 different filters to produce a set of feature maps.After the first convolutional layer, a MaxPooling2D layer is added to the model. This layer down-samples the previous layer's output by selecting the maximum value in each pooling window. This reduces the spatial dimensions of the output and improves computational efficiency.

A BatchNormalization layer is added to normalize the previous layer's output, which helps stabilise the training process.The next layer is a Conv2D layer with 4 filters and a kernel size of (2, 2). This layer performs another convolution on the previous layer's output to further extract essential features from the input data.Another MaxPooling2D layer is added to the model to down-sample the output of the second convolutional layer.The output from the last MaxPooling2D layer is flattened into a 1D vector using the Flatten layer.

A Dense layer with 5 output units is added to the model. This layer is fully connected to the previous layer and performs a linear transformation on the input. A Softmax activation function is applied to the output of the last Dense layer, which produces a probability distribution over the 5 different classes. The model was compiled with 'Adam' optimizer, 'categorical cross-entropy loss function, and 'accuracy' as the evaluation metric. There are 573 trainable parameters in the CNN model.

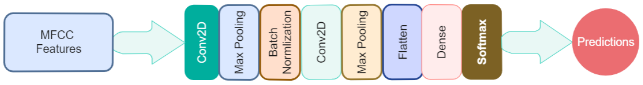

Fig. 2. CNN Model Architecture

C. LSTM Model

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture widely used in speech and language processing due to its ability to model sequential data and long-term dependencies. Speaker recognition identifies an individual based on their unique voice characteristics, such as pitch, tone, and pronunciation. LSTMs have significantly improved speaker recognition tasks compared to traditional machine learning models.

Speaker recognition requires understanding the temporal dependencies in speech signals. LSTMs are particularly good at capturing long-term temporal dependencies in speech signals. Unlike traditional machine learning models that rely on fixed-size feature vectors, LSTMs can maintain a memory of past inputs, allowing them to learn the temporal dynamics of speech signals.

Speaker recognition is often performed in noisy environments with background noise, reverberation, or other distortions. LSTMs are more robust to such noise as they can capture the temporal structure of speech signals, even in the presence of noise.

Overall, Speaker recognition requires understanding the temporal dependencies in speech signals. LSTMs are particularly good at capturing long-term temporal dependencies in speech signals. Unlike traditional machine learning models that rely on fixed-size feature vectors, LSTMs can maintain a memory of past inputs, allowing them to learn the temporal dynamics of speech signals.

The model architecture consists of two LSTM layers with dropout and layer normalization, two dense layers and a Softmax output layer. The first LSTM layer has 128 units, takes the input of shape (timesteps, features), and returns sequences. The second LSTM layer has 64 units. The Dense layer following the second LSTM layer has 32 units, and the final Dense layer has 5 units for the output. The Softmax activation function is used in the output layer. The model has a total of 122,565 parameters and is fully trainable.

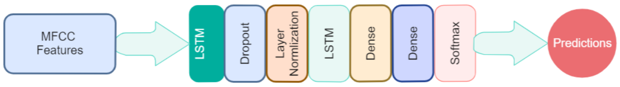

Fig. 3. LSTM Model Architecture

V. Experiments, Results and Observations

We performed the speaker recognition task using 3 models (KNN, CNN and LSTMs) and did the following experiments.

A. Experiment 1

We recorded 15 Non-VCV and VCV sentences from 5 speakers and did speaker recognition for both datasets individually to compare which sentences were better suited for the tasks. Following results were obtained :

We are unlikely to get precisely the same accuracies every time we run a neural network model, even if we use the same settings and dataset. This is because Neural networks usually start with random weights, which can result in different initializations every time we run the model. During training, many optimizers use randomness to determine the direction and magnitude of weight updates. Therefore, even if the initialization is the same, the optimization process may still produce slightly different results each time, so to compare the 2 datasets we made use of KNN average accuracies and observed that VCV (Vowel-Consonant-Vowel) sentences are often used in speaker recognition because they have a consistent phonetic structure and are easy to control in terms of their length and content. VCV sentences consist of three parts: a vowel sound, a consonant sound, and another vowel sound. This structure ensures that each sentence has a similar sound pattern, which makes it easier to analyze the speaker's voice and recognize their unique characteristics.

In contrast, non-VCV sentences can have a wide range of sound patterns, making it harder to analyze and recognize specific characteristics. For example, a sentence like "The quick brown fox jumps over the lazy dog" has a more varied sound pattern than a VCV sentence like "aba apa ava".Overall, using VCV sentences can improve the accuracy and efficiency of speaker recognition systems by providing a consistent and manageable dataset for analysis.

TABLE I
Accuracy of 3 models on VCV and Non-VCV sentences

Model	Accuracy on Non-VCV	Accuracy on VCV
KNN	67.49 %	85 %
CNN	89.99 %	70 %
LSTM	100 %	100 %

B. Experiment 2

We added noise to the VCV sentences audio files obtained in experiment 1 and then observed the model's performances on various noise levels.

From the accuracy vs noise levels plot of the three models, it is clear that noise in datasets affects all the model's performance for speaker recognition tasks. However, LSTM performs better in noisy environments than CNN and KNN models. CNN and LSTM were less affected by the noise as compared to KNN. The MFCC features extracted from audio signals are commonly used as input to CNNs for speaker recognition tasks. Since MFCC features represent the spectral content of audio signals, they are robust to changes in the time domain caused by noise addition.

On the other hand, KNNs are generally not well-suited for handling noisy data. KNN is a distance-based classifier sensitive to noise in the input features. Adding noise can cause the distance between different classes to become closer, which can lead to misclassifications. The addition of noise to audio datasets can affect the accuracy of speaker recognition models since noise can distort the features necessary for speaker recognition, such as the spectral characteristics of the speaker's voice. However, an LSTM model learns the sequential patterns in an audio signal, such as the timing and duration of speech segments and the frequency characteristics of individual phonemes and thus more robust to noise.

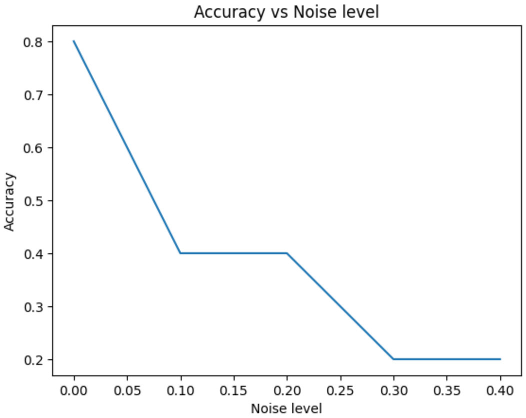

Fig. 4. Accuracy vs Noise Level for KNN

C. Experiment 3

Multilingual audio refers to audio data that contains speech or other sounds in multiple languages. Analyzing and recognizing multilingual audio can be a difficult task as it requires distinguishing between different languages and identifying the language being spoken at any given time.

For instance, a multilingual audio dataset may include speech in English, Spanish, and Mandarin, among others. The audio can be segmented into individual utterances, and each utterance can be transcribed and labeled with its corresponding language.

Once the audio data is labeled, it can be utilized for several applications such as language identification, speech-to-text translation, and speaker recognition. In speaker recognition, the aim is to identify and verify the speaker's identity across different languages, which can be challenging due to the phonetic structures and sound patterns unique to each language.

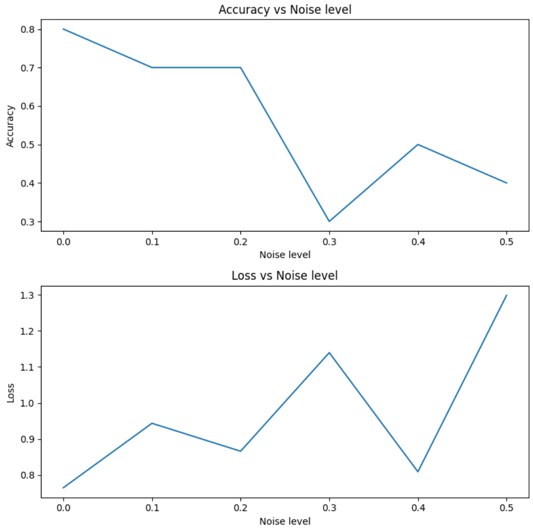

Fig. 5. Accuracy vs Noise Level for CNN

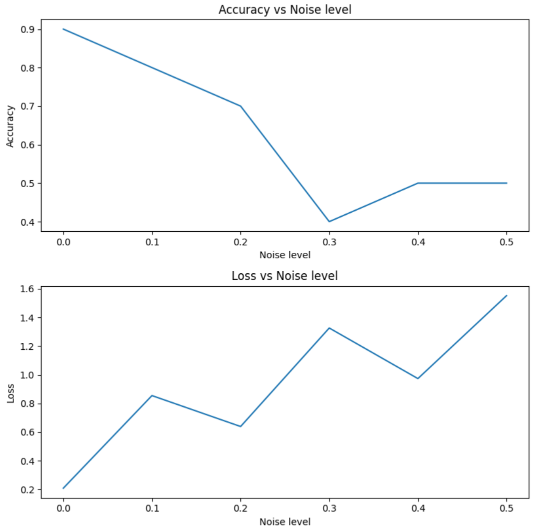

Fig. 6. Accuracy vs Noise Level for LSTM

We conducted an experiment in which we recorded audio samples from 10 individuals in Hindi and English, and used our existing models for multilingual audio classification, reporting their accuracies. Our results show that our models are suitable not only for speaker recognition but also for audio classification.

TABLE II
MULTILINGUAL AUDIO CLASSIFICATION ACCURACY

Model	Audio Classification Accuracy
KNN	62.5 %
CNN	75 %
LSTM	75 %

D. Experiment 4

We recorded 30 sentences (15 Hindi and 15 English) from 25 people to perform multilingual speaker recognition tasks using just MFCC features and our 3 existing models.

Multilingual speaker recognition is different from normal speaker recognition in that it involves identifying and recognizing speakers who speak multiple languages. In normal speaker recognition, the focus is on identifying a speaker's unique voice characteristics regardless of the language they are speaking in. However, in multilingual speaker recognition, the system must be able to recognize a speaker's voice across multiple languages, which can present additional challenges.

One challenge in multilingual speaker recognition is the variation in phonetic patterns across different languages. For example, some languages may have different vowel sounds or consonant clusters that can affect the acoustic characteristics of a speaker's voice. Additionally, multilingual speakers may have different speaking styles or accents when speaking different languages, which can further complicate the task of speaker recognition.

Another challenge is the availability of data in multiple languages. It is important to have a diverse and representative dataset of speakers in each language, as well as enough data to properly train and test the speaker recognition system. Furthermore, it is important to ensure that the data is properly labeled and annotated to accurately reflect the speaker's identity across different languages.

Overall, multilingual speaker recognition is a more complex task than normal speaker recognition due to the additional challenges posed by multiple languages and the need for a diverse and representative dataset.

TABLE III
MULTILINGUAL SPEAKER RECOGNITION ACCURACY

Model	Speaker Recognition Accuracy
KNN	48.33 %
CNN	40 %
LSTM	70 %

We observed that LSTM performs significantly and consistently better in all the Experiments, including multilingual speaker recognition. In speaker recognition tasks, LSTM networks have been shown to perform well because they can capture the long-term dependencies and patterns in speech signals, such as the timing and duration of speech segments and the frequency characteristics of individual phonemes. This ability to capture temporal information is especially important in multilingual speaker recognition, where speakers may use different languages, accents, and dialects. LSTM networks can learn the unique patterns and characteristics of each speaker's voice, regardless of the language they are speaking.

LSTM networks can handle this complexity by learning to extract the most relevant features from the input signals and creating a higher-level representation of the data. This higher-level representation can then be used to classify speakers across multiple languages, making it a powerful tool for multilingual speaker recognition.

E. Experiment 5

We chose a sample of 5 females and 5 males from the dataset and conducted separate tests using our 3 models to determine if there was gender bias in speaker recognition. Although the models performed better on the male group, it would be premature to conclude that speaker recognition is biased towards males. Further investigation is necessary to draw a more definitive conclusion. There can be a gender bias in speaker recognition tasks, as male and female voices can have different characteristics that can impact the accuracy of the recognition. For example, male voices may have lower pitch and frequency, while female voices may have higher pitch and frequency. However, a good speaker recognition system should be able to recognize voices regardless of gender.

Both male and female voices can be used for speaker recognition, and neither is inherently better or worse than the other. The most important factor in speaker recognition is the uniqueness of the individual's voice, rather than their gender. However, it is important to ensure that the dataset used for speaker recognition tasks is diverse and includes samples from a range of genders, ages, and ethnicities to avoid any potential biases in the system.

TABLE IV
SPEAKER RECOGNITION ACCURACY OF 3 MODELS FOR FEMALES AND MALES

Model	Males	Females
KNN	85.41 %	75 %
CNN	87.5 %	75 %
LSTM	87.5 %	100 %

VI. CONCLUSION AND FUTURE WORK

In summary, VCV sentences are preferred in speaker recognition tasks due to their consistent phonetic structure and ease of control regarding length and content. Adding noise to audio datasets can negatively impact the accuracy of speaker recognition models, with KNN being a distance metric the most affected due to its sensitivity to noisy data. Overall, using VCV sentences and LSTM models can improve the accuracy and efficiency of speaker recognition systems in noisy environments. The experiments conducted in this study focused on multilingual audio classification and speaker recognition. Experiment 3 showed that our existing models are suitable not only for speaker recognition but also for multilingual audio classification. Experiment 4 highlighted the challenges of multilingual speaker recognition, such as the variation in phonetic patterns across different languages and the need for a diverse and representative dataset. It was found that LSTM networks consistently outperformed other models, including in multilingual speaker recognition.

In Experiment 5, separate tests were conducted on a sample of males and females to determine if there was gender bias in speaker recognition. Although the models performed better on the male group, further investigation is needed to draw a more definitive conclusion.

Overall, these experiments showed that multilingual audio classification and speaker recognition are complex tasks that require careful consideration of various factors such as the phonetic patterns unique to each language, the availability of diverse and representative datasets, and potential biases in the system. LSTM networks were found to be effective in capturing the temporal information in speech signals, which is especially important in multilingual speaker recognition. To ensure the accuracy and fairness of speaker recognition systems, it is essential to use diverse datasets and take into account potential biases related to gender, age, and ethnicity.

As for future work, we suggest to continue investigating the following :

1. **Are we overfitting the audio source?** We used raw wav audio data from 25 people recorded on computers with moderate ambient noise. Testing the models on audio registered from various other sources will be interesting.

2. **How can this algorithm be applied to real-time streaming?** Currently, the algorithm is trained and tested on existing audio clips; what changes would we have to make to use it in a real-time scenario to identify speakers as they speak?

3. **Other Feature Extraction technique should be explored !** We believe that LPC and LPC-derived features like Line spectral frequencies, linear prediction cepstral coefficients, log area ratios and arcus sine coefficients will perform better or equivalent to mfcc with LSTMs.

VII. Contributions

All members of this team contributed in equal amounts to the overall progress of the project. In particular, the literature review task was shared between both members.

Ayush: Build 3 model architectures (KNN, CNN, LSTM) for speaker recognition task, conducted coding part of all the experiments and prepared the report.

Neha: Recorded audio from 25 people in both English and Hindi (15 sentences in each language), recorded 15 sentences of Non-VCV sentences from 5 people, data preprocessing and prepared presentation.

References

[1] Ge, Zhenhao, et al. ”Neural network based speaker classification and verification systems with enhanced features.” 2017 intelligent systems conference (IntelliSys). IEEE, 2017.

[2] Lukic, Yanick, et al. ”Speaker identification and clustering using convolutional neural networks.” 2016 IEEE 26th international workshop on machine learning for signal processing (MLSP). IEEE, 2016.

[3] Lukic, Yanick, et al. ”Speaker identification and clustering using convolutional neural networks.” 2016 IEEE 26th international workshop on machine learning for signal processing (MLSP). IEEE, 2016.

[4] Mendez, Alfredo. ”Speaker Identification with Deep Neural Networks” Stanford CS 230 Past projects Winter 2019.

[5] Gupta, Rish, Manish Pandit, Sophia Zhen. ”Speaker Identification: Text Independent Context” Stanford CS 230 Past projects Spring 2018

[6] Vyom Pathak, Medium Blog, ”End-To-End Speech Recognition” , https://medium.com/wavey-ai/end-to-end-speech-recognition-f13f0d0197c7

[7] Schmidt-Nielsen, Astrid. Speaker Recognition Using Phoneme Specific Sentences. NAVAL RESEARCH LAB WASHINGTON DC, 1986.

[8] Behic Guven, Blog, ”Speech Recognition in Python- The Complete Beginner’s Guide Simple and hands-on walkthrough”