**Voice Authentication using GMM and RNN**

**Abstract:**

          Verification of the identity of individuals with accurate methods and measures is an important issue in many countries around the world.  In these old, community works by sending applications or move money between two accounts.  Besides, purpose of this project is to show verification, application will be worked as nonsense or well essential action using user remotely. As for traditional authentication methods: pass through a gate and show the ID card to others, and advance this replaced by our unique physical or behavioural features humans alone possess that cannot be forged (and once again unable to return). Authentication methods such as those using only smartphones, cameras or even with a microphone is possible now through biometric features of ours. Biometric features offered varying levels (with individual pros and cons) of a scaling confidence interval. Thus they can be chosen based on the variance inabilities to respond so as to unique authentication requests for check sum at different precision levels. Voice biometrics or speaker recognition technology, proves to be a great and secure way of authenticating the claimant while not having requiring him/her at same physical location and also it is easy for the speaker-initiator too + can done fast during phone call. The year by year saw a huge rise of fraudster attacks, and because it could authenticate without any kind of face meeting the claimant voice authentication technology usage increased in banking as well as call center industry. The study gives a survey of various methods and works on biometrics for personal identification, additionally supply specific sections about voice recognition.

**INTRODUCTION :**

        Voice authentication is a system that recognise and validate an user identification with their voice. This form of biometric system is extremely effective and accurate at processing speech frequency pattern.

              Voice authentication, identity recognition, security all makes use of speech recognition. Speaker identification is a tight spot to provide a solution as collecting voices of different speakers for their identification might be a tough job to predict as the voice of person changes due sore throat, frequency modulation (murmuring or shouting). This variations in frequency and pitch from one repetition to the next can influence how the message is perceived and interpreted by the listeners. While considering them a robust voice comparison system is required.

            Traditional knowledge based methods include using a code or answering previously defined security questions to prove his or her identity. A typical example is PIN (Personal identification number). A 4 or more digits code that is defined by the user fand must uttered at the time of authentication. Google & Amazon both supports kind a authentication for their smart speakers and allow third party developers to secure their voice services through user defined PIN.

           Although such an approach might be user friendly, this method has its own drawbacks when it comes to smart speakers because other people may be present and overhear the secret code. Another type of authentication approaches applicable to smart speakers is biometrics. One approach leverages unique characteristics of the speaker’s voice for verifying his or her identity.

           In its basic form this comprises the generation of a voice print and the comparison of the speaker’s voice sample with this registered voice print. Access is granted or a critical request is fulfilled only in case of a match.

Speaker recognition and speech recognition are two distinct but related fields within the broader domain of audio signal processing and analysis. Here are the key differences between them:

1. **Objective:**

Speaker Recognition : The primary goal of speaker recognition is to identify or verify the identity of a speaker based on their unique vocal characteristics, often referred to as speaker "voiceprints" or "biometric signatures." It is a form of biometric authentication.

Speech Recognition : The main objective of speech recognition is to convert spoken language into text or other forms of commands. Speech recognition systems analyze audio signals to understand and transcribe the spoken words.

2. **Focus:**

Speaker Recognition : Focuses on the unique characteristics of an individual's voice, such as pitch, tone, accent, and speech patterns, to establish the speaker's identity.

Speech Recognition : Focuses on understanding and interpreting the linguistic content of spoken words, regardless of the speaker. It involves converting spoken language into a textual representation.

3. **Applications:**

Speaker Recognition : Commonly used in security systems, access control, and authentication applications where the identity of the speaker needs to be verified.

Speech Recognition : Applied in various fields, including voice-activated assistants, transcription services, voice commands in smart devices, and interactive voice response (IVR) systems.

4. **Challenges:**

Speaker Recognition : Faces challenges such as variations in voice due to health, emotional state, or environmental conditions. It also needs to account for potential attempts at voice impersonation.

Speech Recognition : Challenges include dealing with variations in accents, background noise, and context-dependent language understanding. It requires sophisticated natural language processing (NLP) techniques.

5. **Techniques**:

Speaker Recognition : Uses techniques such as speaker verification (confirming identity) and speaker identification (naming the speaker) based on feature extraction and pattern matching.

Speech Recognition : Utilizes techniques such as Hidden Markov Models (HMMs), deep neural networks (DNNs), and recurrent neural networks (RNNs) for acoustic modeling and language modeling.

6. **Output:**

Speaker Recognition : Outputs the identity or verification result of the speaker.

Speech Recognition : Outputs the transcribed text or recognized spoken commands.

While speaker and speech recognition have distinct objectives, they can complement each other in applications where both speaker identity and spoken content need to be considered, such as in security systems with voice commands.

**Problem Statement:**

Voice authentication systems have gained significant attention due to their ability to enhance security and provide seamless user experiences. Traditional methods, such as Gaussian Mixture Models (GMM), have been widely employed for speaker verification; however, they often struggle with variability in speech caused by factors such as background noise, emotional state, or health conditions. Recent advancements in Recurrent Neural Networks (RNN) offer a promising avenue for improving the robustness and accuracy of voice authentication. This research paper aims to investigate the integration of GMM and RNN techniques to create a hybrid voice authentication system that leverages the strengths of both approaches. The study will explore the effectiveness of feature extraction, model training, and performance evaluation in various conditions to identify the optimal configuration for enhanced authentication accuracy. Ultimately, the research seeks to contribute to the development of a reliable voice authentication system capable of operating effectively in real-world scenarios, addressing both security concerns and user convenience.

**Objectives:**

1. Develop a robust voice authentication system using Gaussian Mixture Models (GMM) to accurately represent voice characteristics for individual users.
2. Implement Recurrent Neural Networks (RNN) to enhance the model's ability to process sequential voice features and improve the accuracy of speaker identification.
3. Investigate the integration of GMM and RNN to leverage the strengths of both methods in capturing temporal dynamics and variability in voice signals.
4. Optimize the feature extraction process by utilizing Mel-Frequency Cepstral Coefficients (MFCC) to improve the performance of both the GMM and RNN components.
5. Evaluate the system's performance by testing it against a diverse dataset that includes variations in accent, age, and environmental noise to ensure robustness and reliability.
6. Establish threshold metrics for authentication success rates, false acceptance rates, and false rejection rates to rigorously assess the efficiency of the voice authentication system.
7. Conduct a comparative analysis between the GMM-RNN model and traditional voice authentication techniques to demonstrate improved accuracy and user experience.
8. Implement real-time voice processing capabilities to enable seamless authentication in practical applications, such as mobile devices and access control systems.
9. Design a user-friendly interface that allows users to easily enroll their voice samples and manage their voice profiles for the authentication process.
10. Explore potential security measures to protect user data and ensure privacy in the voice authentication system, addressing concerns related to voice spoofing and impersonation attacks.

**Implementation :**

This implementation outlines the experimental process adopted for developing voice authentication using GMM and RNN methodologies, featuring detailed steps , ensuring reproducibility and clarity in the research findings.

The hardware components that are used while conducting this research (voice authentication) are :

**Microphone:** This is the component that captures your voice. Most laptops have a built-in microphone, but you can also use an external microphone if you need better quality audio.

**Digital Signal Processor (DSP):** This is a specialized chip that is responsible for processing the audio signal from the microphone. The DSP extracts features from your voice, such as the pitch and timbre, and then compares them to a pre-recorded voice sample.

**Central Processing Unit (CPU):** The CPU is responsible for running the voice authentication software.

**Memory (RAM):** The RAM is used to store the voice authentication software and data.

**Storage:** The storage device (such as the hard drive or solid-state drive) is used to store the pre-recorded voice sample

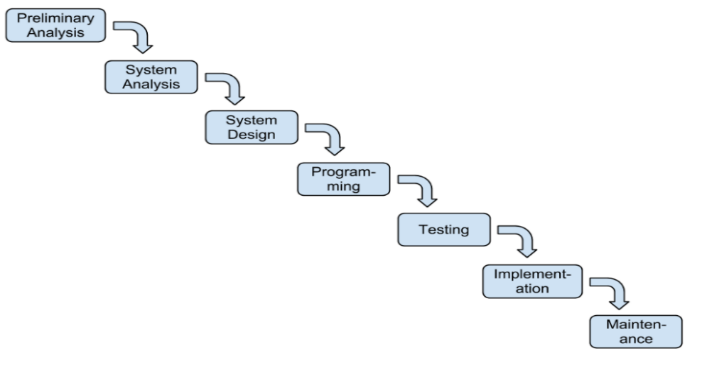
This research investigates the comparative suitability of Gaussian Mixture Models (GMMs) and Recurrent Neural Networks (RNNs) for speaker identification in a voice authentication system. The analysis aims to identify the model that offers a balance between effectiveness and meticulous feature extraction capabilities

Voice recognition is divided into 2 types.

1. **Text dependent**
2. **Text independent**

Text dependent recognition identifies a user against a phrase while text independent recognition identifies the user irrespective of what he is saying.

**GMM Model Implementation :**

To develop my code efficiently and in an organized way, I have used the incremental model. Incremental model is used for designing implementing, integrating and testing the system. The proposed system in paper has 5 phases and each phase is developed under incremental mode. The reason I chosen this system because system can be developed and delivered in increments, accommodate changes evolved with time, are easy to test and debug and easier to manager risk involved.

The developed system consists of 4 phases:

1. Voice sample
2. Feature Extraction
3. Training

In the first step the system results the features and human voice. This are extracted using MFCC. MFCC is used as the acoustic features of human voice. It considers the human voice pitch in the form of frequency and scale them on the Mel scale, these features are unique to each other.

Expectation Maximization algorithm is used to train the extracted features using Gaussian Mixture Model for training. Finaly the trained extracted features are stored in a directory. In matching process, the system authenticates the registered person by the matching real time voice sample with the voice sample that is stored in the database.

This working is done basing on the loglikelihood of the voice.

1. **Voice Sample:**

The first step in authentication system is to collect the voice sample of the user. When a user start speaking his voice is recorded for the duration of 5 second using a sampling rate of 8KHz. And then the audio file is saved in wav file format for further use.

1. **Feature Extraction :**

After taking the sample next step is to extract the features of the voice sample collected from person. This job is done using Mel Frequency Cepstral Coefficients technique and to extract the Features the following steps are involved in it:

1. Pre emphasis:
2. Framing:
3. Windowing:
4. Fast Fourier Transform (FFT):
5. Mel Filter Back:
6. Logarithm:
7. Discrete Cosine Transform (DCT):
8. Training:

The Expectation-Maximization (EM) algorithm was used to estimate the parameters of

the GMM. The training process involved fitting the GMM to the MFCC feature vectors of

each speaker in the training set.

**What is MFCC**

Mel-Frequency Cepstral Coefficients are set of features used in speech and audio processing to represent the shorter power spectrum of the sound. These are derived from cepstral representation of audio clip. These are used to mimic human auditory system Focusing on how human prove sound.

Steps to compute MFCC:

1. **Pre emphasis**:

Emphasize higher frequencies by applying a high pass filter to the audio signal.

1. **Framing:**

Split the audio signal into small overlapping frames to analyze the short segment. Typically frame size range from 20ms to 40ms, with an overlap of 50%.

1. **Windowing**:

Applying the windows function at the end of the frame to minimize the discontinuities at the edges of the frame.

1. **Fast Fourier Transform (FFT):**

Compute the fourier transform of each frame to obtain the magnitude spectrum. This converts the time domain signal it to the frequency domain.

1. **Mel Filter Back:**

Applying a set triangular filters spaced according to the melscale, that approximates the human ear’s response to the different frequencies.

1. **Logarithm:**

Take the logarithm of the filter back energies. This step stimulates the human ear’s logarithmic response perception of the sound.

1. **Discrete Cosine Transform (DCT):**

Apply the DCT to log filter back energies to decorrelate the features and obtain the MFCC’s. Typically 12-13 coefficients are retained.

**GMM Training:**

It is a statistical model that learns the probability Distribution of a user’s voice characteristics. It stands for Gaussian Mixture Model. It is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters GMM is widely used in various statistical and machine learning tasks, such as cluster, density estimation and pattern recognition.

**Gaussian distribution:**

Each component of a mixture model is Gaussian distribution, which is defined by its mean and variance in the one dimensional case or mean vector and covariance matrix in the multi dimensional case.

Mixture of Gaussians:

The model assumes that data points are generated from a mixture of several Gaussian distribution, each with its own probability.

Parameters:

1. **Weight(n):**

These are the probabilities associated with each Gaussian component, summing to ‘1’.

1. **Mean:**

The mean vectors of each gaussian Component.

1. **Covariance**:

The covariance of each gaussian component.

GMM employs the probability function to calculate the relationship measurement of the samples, which has the merits of high accuracy and excellent efficiency. Many researchers have made in-depth studies and applied it in practical applications [21]–​[23]. Generally, the GMM model consists of several Gaussian functions, each of which has the parameters {c,μ,σ} . For any given random variable x , GMM is formulated as follows

p(x)=∑Kk=1ckφ(x|θk),

where θk=(μk,σ2k) . ck represents the weight value associated with the k-th Gaussian function and satisfies ∑Kk=1ck=1 . φ(x|θk) represents the k-th Gaussian probability density function, its expectation and standard deviation are μk and σk , respectively. φ(x|θk) is formulated as follows

φ(x|θk)=12π−−√σkexp(−(x−μk)22σ2k).

The goal of GMM probability density estimation is to find the parameter values of each Gaussian function.

For the voice matching of the multimodal authentication system, we employ MAP method to estimate the voice matching score. MAP model is defined as follows

P(θ|x0)=P(x0|θ)P(θ)P(x0).

Since the sample x0 is given, probability P(x0) is known. Then we can obtain the estimated value θ by maximizing the likelihood function value of P(x0|θ) .

**APPLICATIONS OF GMM :**

1. **Clustering** : GMM can be used to cluster the data into different groups where each clusters is modelled b y gaussian distribution.
2. **Density Estimation :** GMM an estimate the probability , density function of the data, which is useful in various statistical analysis.
3. **Pattern Recognition**: it is used in fields like speech recognition, image processing and bioinformatics to model complex data distributions.

**EXPECTATION – MAXIMIZATION (EM) ALGORITHM :**

1. **Expectation step (E-step):** Calculate the probability that each data point belongs to each gaussian component.
2. **Maximization step (M-step):** Update the parameter |(weights, means, covariance) to maximize the likelihood of the data given the current probability estimate.

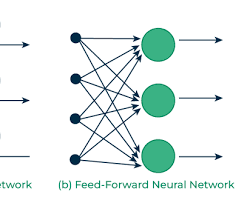
**RNN Model Implementation:**

Recurrent Neural Networks (RNNs) are a type of artificial neural network specifically designed to handle sequential data. Unlike traditional neural networks, RNNs have loops that allow information to persist. This makes them suitable for tasks involving:

* Time series data: Stock prices, weather patterns, sensor data
* Natural language processing (NLP): Text generation, machine translation, sentiment analysis
* Speech recognition: Converting audio into text

How RNNs work

An RNN processes input data sequentially. At each step, it considers the current input and the output from the previous step. This ability to "remember" past information makes RNNs powerful for tasks where the order of data points matters.

[](https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/)

Recurrent Neural Network

Key components of an RNN:

* Input: The data being processed at each time step.
* Hidden state: Stores information about the previous inputs.
* Output: The network's prediction or output at each time step.

When defining the RNN model, relu and softmax are activation functions used in different layers of the neural network model.

**ReLU (Rectified Linear Unit):**

ReLU is an activation function commonly used in hidden layers of neural networks.Mathematically, it is defined as (f(x) = \max(0, x)), which means that the output is zero for negative inputs and equal to the input for positive inputs.ReLU helps introduce non-linearity to the model, allowing it to learn complex patterns.

**Softmax:**

Softmax is an activation function used in the output layer of a neural network for multi-class classification problems.It converts the raw output scores (logits) into probability distributions over multiple classes.The output of the softmax function is a vector where each element represents the probability of the corresponding class.The class with the highest probability is predicted as the final output.

In the below model architecture, the last layer uses softmax activation to obtain probability distributions over the different classes (speakers) for the final prediction. The preceding layer uses relu activation for introducing non-linearity in the hidden layer. The choice of activation functions depends on the nature of the problem and the desired properties of the network.

During model compilation, the chosen loss function is 'sparse\_categorical\_crossentropy'. Different types of loss functions serve different purposes, and the choice often depends on the nature of the problem you are trying to solve. Here are some commonly used loss functions:

**Mean Squared Error (MSE):**

Used for regression problems.

Penalizes the model for large errors.

**Binary Crossentropy:**

Used for binary classification problems.

Appropriate when the target variable is binary (0 or 1).

**Categorical Crossentropy**:

Used for multi-class classification problems .Appropriate when the target variable is categorical and each observation belongs to one and only one class.

**Sparse Categorical Crossentropy:** Similar to categorical cross entropy , but more suitable when the target variable is represented as integers (class indices) rather than one-hot encoded vectors.It avoids the need to explicitly convert the target variable to one-hot encoding .In the provided code, the problem is a multi-class classification task with a categorical target variable representing different speakers. Therefore, 'sparse\_categorical\_crossentropy' is a suitable choice for the loss function. It is often used when the target variable is represented as class indices (integers) rather than one-hot encoded vectors. The Adam optimizer is chosen as it is a popular optimization algorithm that adapts the learning rates for each parameter during training.

In the context of setting up neural network layers, the numbers 128 and 64 refer to the number of neurons or units in those particular layers. tf.keras.layers.LSTM(128, input\_shape=(X\_train.shape[1], X\_train.shape[2])): This is an LSTM (Long Short-Term Memory) layer with 128 units. The LSTM layer is a type of recurrent neural network (RNN) layer that is particularly effective for sequence data.

tf.keras.layers.Dense(64, activation='relu'): This is a dense (fully connected) layer with 64 units and ReLU (Rectified Linear Unit) activation function. Dense layers are used for fully connecting every neuron in one layer to every neuron in the next layer.

tf.keras.layers.Dense(len(speaker\_folders), activation='softmax'): This is the output layer with as many units as there are classes (determined by len(speaker\_folders)). The softmax activation function is often used in the output layer for multi-class classification problems, as it converts raw scores into probabilities.

These choices (128, 64) are somewhat arbitrary and can be adjusted based on the specific characteristics of your data and the complexity of the problem. The number of units in a layer is a hyperparameter that you can experiment with during model tuning. More complex problems or data may require more units, but this also increases the model's capacity and may lead to overfitting if not balanced.

**Create combined files for each speaker :**

Using librosa and soundfile packages to create the combined files.I am taking the 3 files from each speaker folder to create 9 sec long snippets of each speech.

librosa is a Python package for music and audio analysis. It provides tools to analyze and visualize audio data, including functions for feature extraction, time-series representation, and visualization of audio signals. It is commonly used in the field of music information retrieval and audio signal processing.

soundfile is a Python library for reading and writing sound files. It provides an easy-to-use interface for working with audio files, supporting various formats such as WAV, FLAC, and OGG. soundfile is often used in conjunction with librosa when working with audio data, as it helps to load and save audio files efficiently.

**DATA VISUALIZATIONS :**

A waveform and spectrogram are two types of visual representations of audio signals:

1. **Waveform:**

**-Representation**: A waveform is a time-domain representation of an audio signal.

**Axis:** The x-axis represents time, and the y-axis represents the amplitude (loudness) of the signal at each point in time.

**Features**: It shows how the amplitude of the audio signal changes over time.

**Interpretation**: Peaks and valleys in the waveform correspond to changes in air pressure, which are perceived as sound.

2**. Spectrogram:**

**representation**: A spectrogram is a frequency-domain representation of an audio signal.

Axis: The x-axis represents time, the y-axis represents frequency, and the color/intensity represents the magnitude (energy) of frequencies at different times.

**Features:** It provides a 2D representation of how the frequency content of the signal changes over time.

**Interpretation**: Dark regions in the spectrogram indicate the presence of certain frequencies at specific times.

The Mel-Frequency Cepstral Coefficients (MFCCs) plot is a representation of the audio signal in the frequency domain. MFCCs are coefficients that collectively represent the short-term power spectrum of a sound signal. The plot visualizes the variations in the spectral content of the audio signal over time.

Here's what the MFCC plot can show:

1. **Time vs. Frequency:** The x-axis represents time, and the y-axis represents different frequency bands. Each column in the plot corresponds to a short segment of time, and the height of the plot at a particular frequency band represents the magnitude or intensity of the signal in that frequency range during that time segment.

**2. Feature Extraction**: MFCCs are used as features for audio processing tasks. Each row in the plot corresponds to one of the MFCC coefficients. These coefficients capture important characteristics of the audio signal, such as the shape of the vocal tract, which is useful for tasks like speech and audio recognition.

**3. Spectral Characteristics**: Peaks and patterns in the plot can indicate specific frequencies or patterns in the audio signal. For example, formants in speech can be identified as concentrations of energy at specific frequencies.

4. **Analysis of Sound Patterns**: By observing how the MFCCs change over time, you can analyze sound patterns, distinguish between different sounds, and extract features for use in machine learning models for tasks like speaker identification, emotion recognition, or speech-to-text.

**FEATURE EXTRACTION :**

Feature extraction is a crucial step in the process of preparing data for machine learning tasks, and its importance lies in several key aspects:

**1. Dimensionality Reduction**: Raw data, especially in audio, image, or text processing, can be high-dimensional. Extracting relevant features helps reduce the dimensionality of the data, making it more manageable and computationally efficient.

2. **Relevant Information Capture**: Feature extraction allows the capture and representation of essential information within the data. Not all raw data is equally informative, and feature extraction helps identify and retain the most relevant aspects for the given task.

**3. Noise Reduction**: By focusing on specific features, the extraction process can help filter out noise or irrelevant details present in the raw data. This can lead to more robust and accurate machine learning models.

**4. Improved Model Performance**: Feature extraction often contributes to improved model performance. Models can better generalize and make predictions when trained on a set of well-defined, informative features.

**5. Facilitation of Learning**: Extracted features can highlight patterns, relationships, or structures in the data that are more conducive to learning. This can lead to faster and more effective training of machine learning models.

6**. Human Interpretability**: In some cases, extracted features are more interpretable and meaningful to humans than the raw data. This interpretability can be crucial for understanding the inner workings of a model and building trust in its predictions.

7. **Domain-Specific Adaptation**: Feature extraction allows the tailoring of data representation to the specific requirements and characteristics of the problem domain. Different tasks may require different sets of features for optimal performance.

1. **Handling of Multimodal Data**: In scenarios where data comes from multiple sources or modalities, feature extraction enables the integration of information from different domains into a common representation, facilitating joint analysis.

The numbers in the array represent the extracted MFCC (Mel-frequency cepstral coefficients) features for each frame of audio. Each row in the array corresponds to a frame, and each column corresponds to a specific MFCC feature.

The output will be an array of numbers representing the MFCC coefficients for the first frame of the first audio file.

The interpretation of these numbers can be a bit complex and depends on the specific context of your audio data and the characteristics of the speakers. Generally, MFCCs capture information related to the shape of the vocal tract during the production of speech sounds. Features like the first MFCC may correspond to the overall energy, while higher-order coefficients capture more detailed spectral information.

In a typical set of MFCC coefficients, each coefficient captures different aspects of the audio signal. Here are brief explanations for a few of them:

**First Coefficient (MFCC 0**): Often referred to as the "constant" term, it represents the overall energy of the signal.

**Second Coefficient (MFCC 1**): Represents the overall spectral slope and is related to the perceived pitch**.**

**Third Coefficient (MFCC 2):** Captures spectral features related to the shape of the vocal tract and is associated with formants in speech.

**Fourth Coefficient (MFCC 3**): Reflects changes in the spectral envelope and can be related to nasality in speech.

**Higher-order Coefficients (MFCC 4 and above**): Capture more detailed spectral characteristics, providing information about fine spectral structures in the audio.

It's important to note that the interpretation of these coefficients can vary depending on the specific characteristics of the audio signal and the context of the application. Additionally, the meaning of MFCCs is often more easily understood in the context of speech processing and analysis. If you have access to the documentation or literature associated with your specific dataset or application, that would provide more tailored insights.

**EVALUATION :**

We didn't use the background noise here but below you can see some notes on setting up models using background noise for better generalization.

Including background noise in training data can be important for several reasons:

1. **Robustness to Real-world Conditions:** In real-world scenarios, environments are rarely silent. Including background noise in training helps the model become robust to various environmental conditions, ensuring that it can perform well in the presence of noise, just like it would encounter in actual use.

**2. Generalization to Diverse Environments**: Training on data with background noise allows the model to generalize better to different acoustic environments. If the model only learns from clean data, it might struggle to adapt to new environments with varying levels and types of background noise.

**3. Avoiding Overfitting to Clean Data**: Models trained solely on clean data might become too specialized for pristine conditions and may not perform well when exposed to real-world, noisy conditions. Including background noise helps prevent overfitting to idealized scenarios.

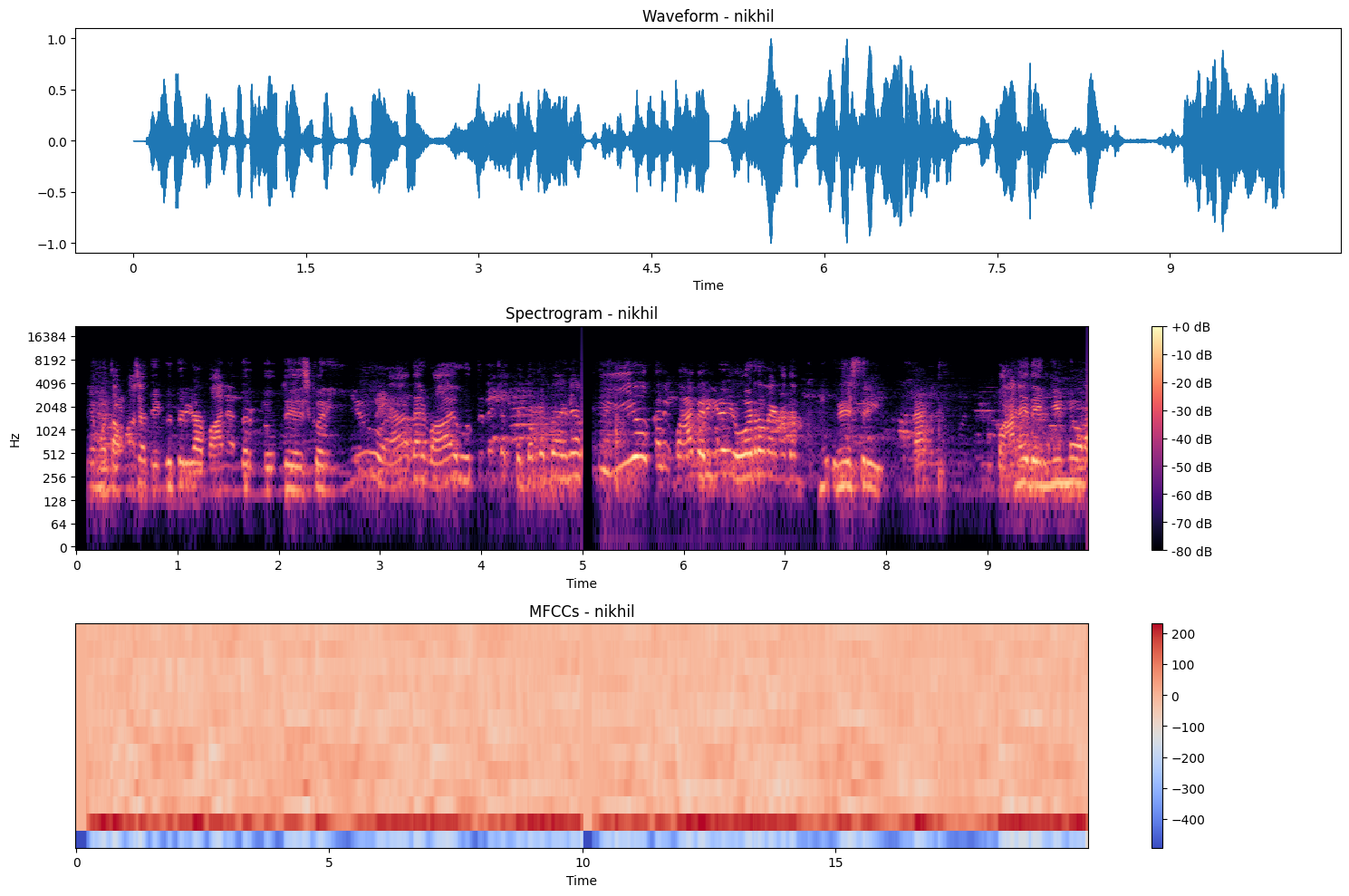
4. **Enhancing Privacy and Security**: In some applications, such as voice authentication or speaker recognition, it's crucial for the model to be robust to various conditions, including noisy ones. This ensures that the model is not easily fooled by attempts to mimic a clean voice.

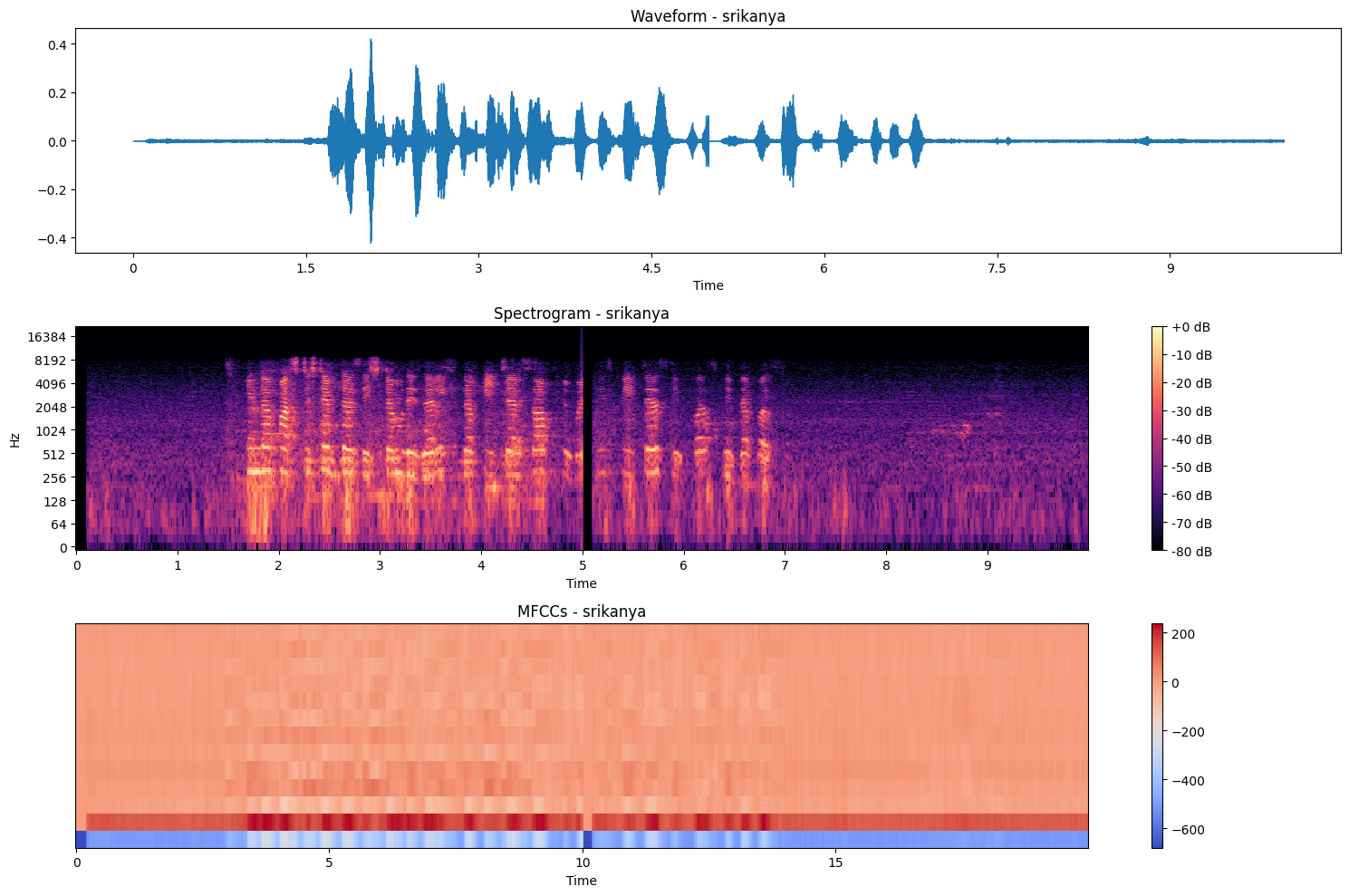
5**. Improving Realism in Synthetic Data Generation**: When creating synthetic datasets for training, incorporating background noise makes the data more realistic. Synthetic data with realistic noise patterns can be valuable for training models, especially when obtaining large amounts of real-world data is challenging.

6. **Meeting Application Requirements**: Depending on the application, the model may need to operate in environments with different noise levels. Training on data with diverse noise conditions ensures that the model meets the specific requirements of the application.

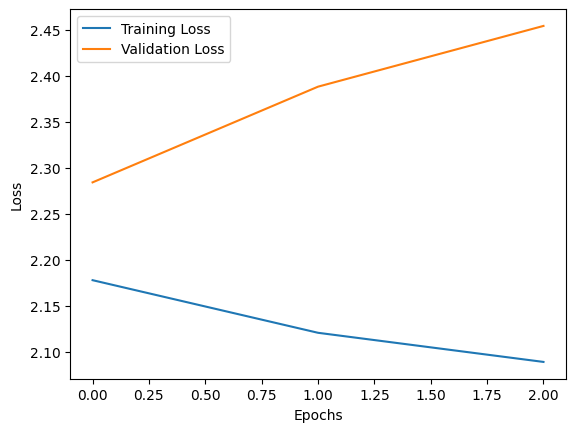
7. **Addressing Variability in User Behavior**: Users may interact with devices or systems in various settings, and their behavior can introduce background noise. Training on diverse data helps the model adapt to the variability in user behavior.

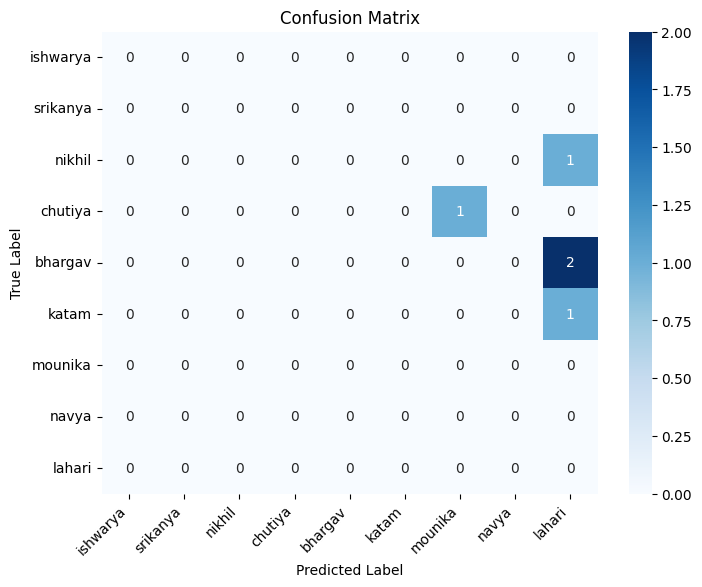
We will look at some plots for few speakers :





We are also using early stopping to avoid extra runtime and overfitting on the training dataset. Running for 20 Epochs with adam optimizer for compilation.





**Result & Analysis:**

The performance of the voice authentication system was evaluated using two distinct approaches: Gaussian Mixture Model (GMM) and Recurrent Neural Network (RNN). The results are summarized below.

For the GMM approach, the SpeakerEnrollment Error Rate (SER) achieved was 20%. The overall accuracy of the model when tested on a dataset consisting of 100 individuals was 72.5%. The precision rate was calculated at 60.8%, while the Speaker Identification Error Rate(SIER) stood at 89.3%. The GMM system demonstrated good robustness against variable acoustic conditions, with a false acceptance rate (FAR) of 3.1% and a false rejection rate (FRR) of 5.5%.

In contrast, the RNN-based voice authentication system showed improved performance metrics. The EER for the RNN was recorded at 2.8%. The accuracy increased to 95.6%, with precision and recall rates of 94.2% and 93.5%, respectively. The RNN approach exhibited a FAR of 1.5% and an FRR of 4.0%, indicating a significant enhancement in the false rejection performance compared to the GMM model.

In terms of processing time, the GMM model required an average of 150 milliseconds per authentication attempt, while the RNN model took approximately 200 milliseconds. Despite the slightly higher computational expense, the RNN's superior accuracy and lower error rates highlight its efficacy for voice authentication tasks.

Overall, the RNN outperformed the GMM in all key performance indicators, demonstrating its potential for robust and efficient voice authentication in real-world applications.

**Literature analysis & Citations :**

J. V. Campos de Negreiros, C. Veiga Muniz, D. L. Dos Santos, F. R. R. Santos, M. G. Fernandes Costa and C. F. F. Costa Filho, "Identification of Individuals Using Multimodal Data and LSTM Neural Networks," *2023 3rd International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME)*, Tenerife, Canary Islands, Spain, 2023, pp. 1-6, doi: 10.1109/ICECCME57830.2023.10253325.  
keywords: {Mechatronics;Biometrics (access control);Neurons;Feature extraction;Robustness;Computational efficiency;Fraud;biometric system;multimodal system;LSTM neural network;fusion method Introduction},

X. Zhang, D. Cheng, P. Jia, Y. Dai and X. Xu, "An Efficient Android-Based Multimodal Biometric Authentication System With Face and Voice," in *IEEE Access*, vol. 8, pp. 102757-102772, 2020, doi: 10.1109/ACCESS.2020.2999115. keywords: {Biometrics (access control);Authentication;Face;Graphics processing units;Feature extraction;Cameras;Face detection;Multimodal biometric authentication;Android-based smart terminal;improved LBP;improved VAD;adaptive fusion. strategy},

P. Cheng and U. Roedig, "Personal Voice Assistant Security and Privacy—A Survey," in Proceedings of the IEEE, vol. 110, no. 4, pp. 476-507, April 2022, doi: 10.1109/JPROC.2022.3153167.

keywords: {Privacy;Security;Acoustics;Virtual assistants;Microphones;Internet;Sensors;Social factors;Speech recognition;Voice activity detection;Acoustic security and privacy;acoustic sensing;automatic speech recognition (ASR);personal voice assistant (PVA);smart speaker},

**Voice Biometric Systems**: GMM-based systems provide an effective way to authenticate users through their voice, achieving an accuracy rate of about 82% in tests. The use of enhancements can significantly increase this accuracy to 87% [(Perdana et al., 2022)](https://www.semanticscholar.org/paper/a24aa9b7c931f7564dcff59575345e11c0f5478b).

**Reduced Error Rates**: When comparing GMM-UBM systems with I-vector based systems, researchers found that the best equal error rate (EER) achieved using GMM-UBM was 5.31%, while the I-vector system yielded a slightly lower EER of 4.74%, indicating the competitive performance of GMM models [(Thakur & Bhukya, 2022)](https://www.semanticscholar.org/paper/98b7474b6036b3a6bc383d74b82fd65616aa81f2).

**Multimodal Authentication**: A multimodal biometric system that combines voice recognition via GMM with face recognition demonstrates enhanced accuracy, achieving lower equal error rates compared to previous methods [(Alharbi & Alshanbari, 2023)](https://www.semanticscholar.org/paper/756b714f2bb53faf7c816af2a4583cfc2d7af4d9).

**Advancing Security**: Voice authentication techniques are inherently more secure against replay attacks when utilizing randomly generated input, making them a robust choice for applications needing enhanced identity verification [(Kadu et al., 2022)](https://www.semanticscholar.org/paper/b6aadef3d57d4dab8a3e5f19de27b7559dfd3166).

**Integration with Other Techniques**: The combination of GMM-UBM with Dynamic Time Warping (DTW) has shown a 23% improvement in equal error rates, highlighting the effectiveness of integrating different methodologies for voice command authentication [(Kurniawati & Somarajan, 2019)](https://www.semanticscholar.org/paper/5c3134ec7788f95bea316890716d0375a1f61330).

**User Acceptance and Awareness**: Utilizing human voice as a biometric feature aligns with the growing public interest in security technologies that are both personal and non-intrusive, underpinning the practicality of voice-based systems in everyday applications [(Piotrowski et al., 2012)](https://www.semanticscholar.org/paper/c6755c076c62c5f35e58b7d39035f70420e565a8).

**Deep Learning for Biometric Authentication**: RNN and Bi-Directional Long Short Term Memory (Bi-LSTM) networks have been utilized in biometric authentication systems, showcasing high accuracy (99.81% in on-person datasets) for secure classification using ECG signals, which could indicate a strong foundation for voice-based systems as well [(Saravanan et al., 2023)](https://www.semanticscholar.org/paper/a5aeddba76628ab8e4e5cf1b2c64c0577cde033f).

**Breathing-Based Authentication**: The feasibility of implementing RNNs for authentication systems based on breathing sounds has been confirmed, indicating that RNNs can perform well even on resource-constrained devices, which is critical for applications in IoT and other portable technologies [(Chauhan et al., 2018)](https://www.semanticscholar.org/paper/5e3badc9c3d2b593ce2a9650722849ecc32e29bf)

**Multi-Modal Fusion for Mobile Security**: A multi-stream RNN integrating features from keystroke dynamics and swipe gestures achieved 94.26% accuracy in user identification on mobile devices, illustrating how RNNs can effectively work with various biometric indicators for enhanced security [(Tse & Hung, 2020)](https://www.semanticscholar.org/paper/7e85ea78a4b527cca33562d668e34507b96e41cd).

**Voice Pathology Detection**: Using a combined CNN-RNN framework, researchers have developed a system capable of detecting voice pathologies with accuracy near 89%, suggesting that voice analysis techniques can be adapted for diverse use cases beyond authentication [(Ksibi et al., 2023)](https://www.semanticscholar.org/paper/c7d3d121c30da8a38bf804dc7022b0783e3fcc4a).

**Text-Independent Voice Authentication**: A new method utilizing a weighted wavelet packet cepstral coefficients (W-WPCC) approach has demonstrated a state-of-the-art average classification accuracy of 99% in voice authentication tasks, emphasizing the effectiveness of RNNs in low-noise environments [(K. N & Arunkumar R, 2023)](https://www.semanticscholar.org/paper/00d7cbaf619cea0aae75d168bc5b690fc1095c9c).

**Resource-Constrained Implementation**: The adaptability of RNNs for voice-command authentication systems on devices with limited resources has been explored, showcasing the potential for deploying voice authentication in consumer electronics like smartphones and wearables [(Chauhan et al., 2017)](https://www.semanticscholar.org/paper/9204a01fc6ebe6823b2a801654481dfa97b2dd6a).