EVALUATION OF MACHINE LEARNING ALGORITHMS USING COMBINED FEATURE EXTRACTION TECHNIQUES FOR SPEAKER IDENTIFICATION

ABSTRACT

Aims: To evaluate and compare machine learning algorithms when various feature extraction techniques are employed together and determine the optimal feature combinations for the models studied.

Methodology: The TIMIT online database selected 5 male and 5 female non-native English speakers from five American locations. Each speaker had 10 3-second utterances, totaling 500. Mel frequency cepstral coefficients (MFCC), linear predictive cepstral coefficients (LPCC), gammatone frequency cepstral coefficients (GFCC), discrete wavelet transforms (DWT) and pitch features were extracted using MATLAB and concatenated. A 10-fold cross-validation data split into 85%/15% training/testing datasets was used to train and evaluate three classifier models—Random Forest (RF), Linear Discriminant Analysis (LDA), and Logistic Regression (LR)—using Python software.

Results: LR outperformed the other two models with an average score of accurate predictions of ≈ 76% for the MGL feature combination and 70% for the highest number of feature combinations (MGDLP). All three models improved performance with more concatenated features.

Conclusion: The performance of LR (76% at MGL (39 features) and 70% at MGDLP (53 features)) indicates that speaker-specific training data improves system performance, however, too much data does not translate to even better performance because the system will eventually achieve its peak performance. Also, all-cepstral feature combinations performed better than other feature combinations implying that cepstral features are more robust and improve speaker identification systems.

Keywords: Speaker identification; feature extraction; machine learning; evaluation metrics.

1. INTRODUCTION

According to [1] the human voice contains characteristics that are unaffected by the substance of the speaker's conversation and these voice characteristics such as pitch, tone, rhythm, and pronunciation contain information [2] that can be used by a speaker identification system to automatically identify a speaker from a recording of their voice or speech utterance. It aims to classify an unknown utterance anonymously as belonging to one of a set of N reference speakers [3]. Speaker identification is reliant on the presence of the speaker's voice biometrics in the database of speakers' voice templates or models, which are categorized as either open-set or closed-set. In the case of open-set speaker

identification, the unidentified speaker's utterance is compared with the speaker model that does not contain the input speaker's registered template. If there is no exact match, the input speaker is rejected [4]. Conversely, in closed-set identification, the unidentified speaker's utterance is compared with the pre-existing utterances of registered templates in the model, which includes the input speaker.

The most critical step in speaker identification is feature extraction, where different feature extraction techniques [5] have been applied to extract useful features from the speech data. There are numerous researches [6][7] that have been done to evaluate the performance of machine learning models on extracted features from the speech data. Advances in digital signal processing, algorithms, architecture, and hardware [1] in recent years have pulled significant research interest in the fields of voice processing branching into speech recognition and speaker identification/recognition systems. These advances have made the manipulation of acoustic data for speaker feature extraction from speech wave [2] [8] and applying pre-processing techniques to improve the performance of a recognizer easier and expanded the applications of speaker identification systems in the areas of biometric identification most especially speech and speaker recognition.

As a relatively new research area, due to these advances, there is still so much to be researched on and this paper focuses on contributing knowledge to this area of research by assessing and comparing the performance of some popular machine learning algorithms on the concatenated features obtained from popularly used feature extraction techniques [9][10][11]. This work will find relevance in forensic and security related applications that need high and reliable accuracies. It can also serve as the identification phase of speaker recognition systems that serve as access verification for applications such as cell phone authentication, smart home authentication and remote banking applications.

2. METHODOLOGY

The proposed method for the evaluation of machine learning algorithms using combined feature extraction techniques was executed in the steps summarized in the flow diagram as depicted in Figure 1.

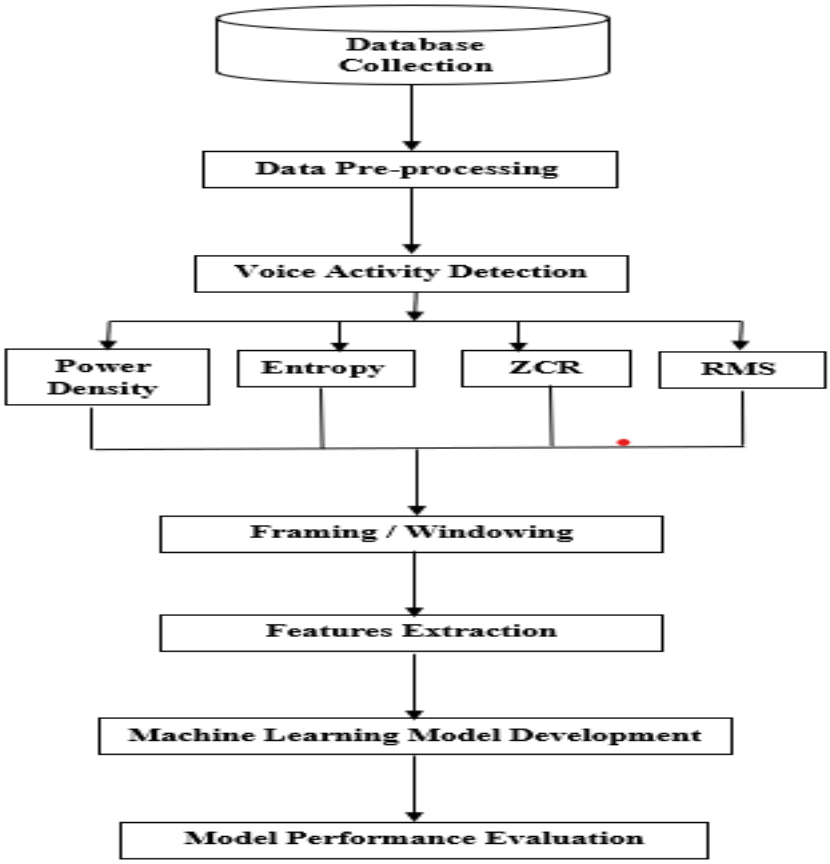

Fig. 1. Flow diagram of the methodology

2.1 Speech Database Collection

The Texas Instruments/Massachusetts Institute of Technology (TIMIT) database was used for data collection. TIMIT is a diverse and well-annotated collection of American English speech data for training, testing, and evaluating speech recognition systems serving as a valuable resource for researchers and developers in the field of speech recognition [12]. A total of 50 non-native English speakers comprising of 5 male speakers from 5 different regions (25 speakers) and 5 female speakers from 5 different regions (25 speakers) were chosen. Each speaker had 10 utterances of 3 seconds duration, making it a total of 500 utterances. Table 1 summarizes the specifications of the TIMIT database.

Table 1. Summary of TIMIT database

Parameter	Values
Language spoken	English
No. of utterances	10 per speaker
No. of speakers	25 Males and 25 Females = 50 speakers
Nationality of speakers	A wide range of dialects and accents, five major dialects of American English found across the United States.
Dataset Distribution	a) 9 utterances from the 50 speakers (450 utterances) for the training dataset b) 1 Utterance from the 50 speakers (50 utterances) for the testing dataset
Duration of utterances	3 secs
Sampling rate	16 kHz
Recording Environment	quiet space

Source: Researcher, 2023

2.2 Data Pre-processing

The 500 utterances of speech data obtained from the TIMIT database were pre-processed by applying voice activity detection (VAD), resampling and normalization.

2.2.1 Voice Activity Detection

Voice activity detection **(**VAD) is a common technique used in speech signal processing to detect voiced and unvoiced portions in speech signals. This method filters the speech signals to exclude silent and particularly noisy segments that may otherwise biased the training stage [13]. The VAD system implemented in this work employed level-crossing sampling for voice activity detection where useless samples and non-speech parts of the signal were eliminated due to the activity-dependent nature of this sampling scheme. Power density, entropy, zero crossing rate (ZCR) and root mean square (RMS) value were the VAD measures implemented and their combined thresholds were used to identify the speech portions in the utterances.

The power density of the speech signal was computed using the Welch method. This method was performed by dividing the framed speech signal into successive blocks K, windowing each frame and computing the modified periodogram for each block or segment.

The Welch power spectrum $(\widehat{PD_s^W})$ was determined using [14] as given in Equation 1 by averaging the modified periodograms to reduce the variance of the individual power measurements.

$$\widehat{PD_s^W}(\omega_k) \triangleq \frac{1}{K}\sum_{k=1}^{K-1} P_{s_k,M}(\omega_k) \quad (1)$$

where $\widehat{PD_s^W}(\omega_k)$ is the Welch power density of the windowed k^{th} block, and $P_{s_k,M}(\omega_k)$ is the modified periodogram of the k^{th} block as given in Equation 2.

$$P_{s_k,M}(\omega_k) = \frac{1}{N}\sum_{n=1}^{N-1}\left|FFT_{N,k}(s_i(n))\right|^2 \quad (2)$$

where $\left|FFT_{N,k}(s_i(n))\right|^2$ is the squared magnitude of the DFT result.

Power density represents the distribution of power in the frequency domain and provides information about the energy distribution across different frequency components of the speech signal [14] . In voiced speech segments, the energy tends to be concentrated around the fundamental frequency (pitch) and its harmonics, resulting in a distinct spectral pattern. On the other hand, unvoiced speech segments exhibit a more uniform distribution of energy across frequencies.

Zero-crossing rate (ZCR) [13] is the number of times the signal changes value, from positive to negative and vice versa, divided by the length of the frame. Voiced speech signals exhibit a relatively low ZCR compared to unvoiced speech or background noise. This is because the vibrations produced by vocal cord vibrations in voiced speech tend to have a more periodic waveform, resulting in fewer zero crossings. On the other hand, unvoiced speech or noise tends to have a higher ZCR due to its more random and turbulent nature. In this work, the ZCR (i) is defined in Equation 3 as

$$ZCR(i) = \frac{1}{2N}\sum_{n=1}^{N}|sgn[s_i(n)] - sgn[s_i(n-1)]| \quad (3)$$

where $s_i(n)$ is the framed speech signal, N is the number of samples in a frame, and $sgn()$ is the sign function given in Equation 4.

$$sgn[s_i(n)] = \begin{cases} 1, & s_i(n) \geq 0 \\ -1, & s_i(n) < 0 \end{cases} \quad (4)$$

Equation 5 was used in the computation of the signal energy entropy [13] for each speech frame. Speech signals have most of its energy collected in the lower frequencies, whereas most energy of the unvoiced speech exists in the higher frequencies. The signal energy entropy was calculated by dividing each short-term frame into K sub-frames of fixed duration and then the energy for each subframe computed.

$$H(i) = -\sum_{k=1}^{K} E_k \; x \; \log_2 E_k \quad (5)$$

where E_k is the given in Equation 6

$$E_k = \frac{E_{subframe}}{E_{frame}} \quad (6)$$

where $E_{subframe}$ is the energy of each subframe given in Equation 7 and E_{frame} is the total energy of the frame given in Equation 8.

$$E_{subframe} = \frac{1}{N}\sum_{n=1}^{N}[s_i(n)]^2 \quad (7)$$

$$E_{frame} = \sum_{k=1}^{K} E_{subframe} \quad (8)$$

where N is the number of samples in the sub-frame.

The RMS [15] value for each frame was calculated by taking the square root of the average of the squared samples within the frame as given in Equation 9. The RMS value of each frame was then compared with the RMS value of the entire speech utterance.

$$RMS_{frame} = \sum_{n=1}^{N}\frac{[s_i(n)]^2}{N} \quad (9)$$

Table 2 summarizes the values of the parameters and the thresholds used in achieving VAD.

Table 2. Summary of parameter values for voiced activity detection

Frame Parameter	Values
Sampling frequency	16,000 samples/s
Frame size	30 ms
Overlap length	60 % of the frame size
Subframe size	10 samples
Thresholds	
Power density	> 50 dB
Zero crossing rate	< 1000
Energy entropy	> 4.5
RMS	> RMS value of the entire utterance

2.2.2 Resampling

After feature extraction from the framed segments of speech obtained from VAD, it was observed that the sample sizes were different for the different speaker utterances. To avoid mismatch during feature classification and identification resampling was done to balance the sample sizes.

2.2.3 Per-emphasis

A high-pass filter was applied to the speech signal to amplify the higher frequency components to achieve a balanced spectrum of voiced sounds which often have a steep roll-off in the high frequency band. Due to the glottal source, voiced sounds typically have a negative 12 dB/octave slope which is offset by a +6 dB/octave boost induced by acoustic energy radiating from the lips [16] . The recordings for this work were done using a mobile phone microphone which introduces a downward slope of approximately -6 dB/octave compared to the true spectrum of the vocal tract obtained. The pre-emphasis filters remove some of the glottal effects in the vocal tract parameters [17] The most commonly used pre-emphasis filter was used which has a transfer function represented by Equation 10.

$$H(z) = 1 - bz^{-1} \quad (10)$$

where the value of b controls the slope of the filter and is usually between 0.4 and 1.0 [18].

2.2.4 Framing and Windowing

The next pre-processing stage was segmenting the samples into short frames of 30ms durations to make the signals quasi-stationary. Each frame was overlapped with the adjacent frame by 60% of the frame size (see information in Table 2) and windowed to prevent discontinuities between successive frames. The Hamming window function, a tapered and smoothing mathematical function, was applied to the edges of the window by multiplying each frame of the signal by the window function to reduce the impact of spectral leakage and artefacts that may arise from framing. Windowing is particularly advantageous when processing signals using Fourier-based algorithms such as the Fourier transform or the discrete Fourier transform (DFT), which assume the signal to be of infinite length. Examples of other window functions are the rectangular window, the Hanning window, the Hamming window, and the Blackman window, each with distinct features and trade-offs.

The mathematical expression of the Hamming window function used is presented in Equation 11.

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1}),\ 0 \leq n \leq N \quad (11)$$

where N = number of samples in each frame.

The result of windowing is presented in Equation 12.

$$y(n) = x(n)\ x\ w(n) \quad (12)$$

Where $x(n)$is the discrete signal, $y(n)$ is the result of windowing

The DFT of each windowed frame is performed using equation 13 to obtain the magnitude spectrum of the signal.

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{\frac{-j2\pi nk}{N}}; 0 \leq k \leq N-1 \quad (13)$$

where N is the number of points used to compute the DFT.

2.2.5 Normalization

The different utterances take from different speakers in the TIMIT database introduces different signal amplitudes and to account for these variations, utterance-level normalization was performed to reduce the influence of irrelevant variations in the speech signals to ensure a uniform scale for comparison, analysis and processing. This also ensures robustness and a more reliable speaker modelling and identification stage.

2.3 Feature Extraction

Different feature extraction techniques were used to extract different sets of features and capture different aspects of the data. A total of 53 short-term features (13 Mel frequency cepstral coefficients, 13 Gammatone frequency cepstral coefficients, 13 linear prediction cepstral coefficients, 13 discrete wavelets transform components and 1 pitch) were extracted for each frame using MATLAB version R2019a software. The unique features identifiers extracted from the speech samples were parameterized into numerical characteristics representing unique entities of each speaker for the purpose of machine learning.

2.3.1 Mel Frequency Cepstral Coefficient Extraction Technique

Fast fourier transform (FFT) is applied to each windowed frame of the speech signal and is transformed to the frequency domain where the Mel spaced filter banks, which imitates the human auditory dynamics, are applied to get the Mel-spectrum. The block diagram of the implemented mel frequency cepstral coefficient (MFCC) feature extraction process is shown in Figure 2.

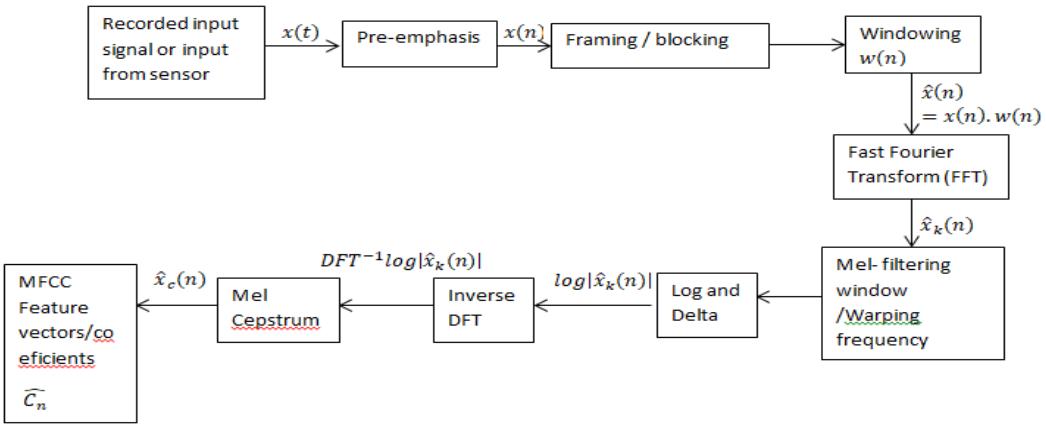

Fig. 2. Block diagram of MFCCs feature extraction

The mel frequency warping is done using Equation 14

$$mel(f) = 2595 \text{ x } log_{10}(1 + f/700) \quad (14)$$

where mel(f) is the frequency (mels) and f is the frequency (Hz).

Hamming function was implemented using Equation 11 and the DFT of each windowed frame was performed using Equation 13. Finally, the MFCC was calculated according to the formula in Equation 15 with the mean of the MFCC coefficients across all frames subtracted from each coefficient to reduce speaker-specific information.

$$\hat{C}(n) = \sum_{m=0}^{M-1} \log_{10}(\hat{X}_k) \cos\frac{\pi n\,(m-0.5)}{M}; n = 0,1,2,\dots,C-1 \quad (15)$$

where $\hat{C}(n)$ are the cepstral coefficients, C is the number of MFCCs, and

$\hat{X}_k$ is the output of filter bank.

Thirteen (13) coefficients for MFCC were extracted and these coefficients are reliable and robust to different speaker and variable recording conditions (19)(20)(21)

2.3.2 Linear Prediction Cepstral Coefficient Extraction Technique

Linear prediction cepstral coefficient (LPCC) are the coefficients of the Fourier transform instances of the logarithmic magnitude spectrum of LPC (Linear Prediction Coding) which imitates the human vocal tract dynamics and the value of the signal is expressed as a linear combination of previous values [9]. The block diagram of the LPCC process is shown in Figure 3 and the linear prediction cepstral coefficients were computed using Equation 16. Thirteen (13) coefficients for LPCC were extracted.

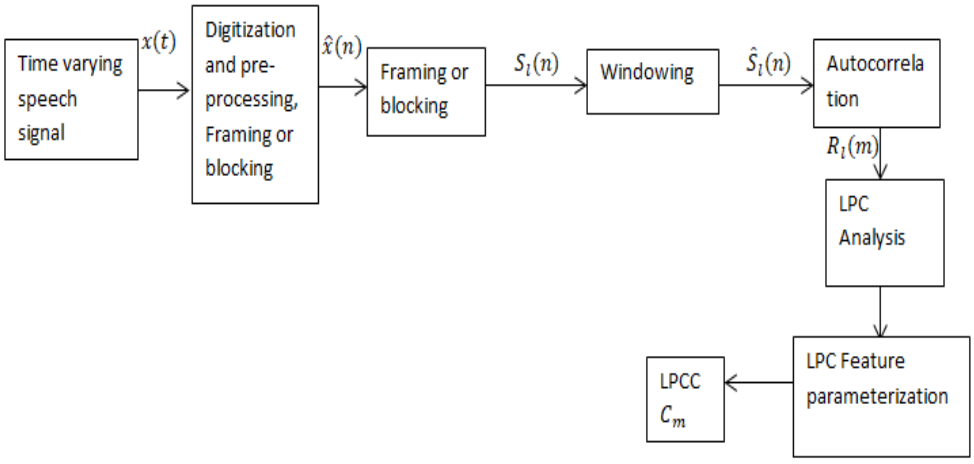

Fig. 3. Block diagram of LPCC feature extraction

$$C_m = a_m + \sum_{k=1}^{m-1} \left[\frac{k}{m}\right] C_k a_{m-k} \quad (16)$$

Where C_m = cepstral coefficient, and

a_m =linear prediction coefficient

2.3.3 Discrete Wavelet Transform Extraction Technique

After pre-processing, framing and windowing to enhance the relevant speech information, the speech signal was decomposed using discrete wavelet transform (DWT) extraction (see Equation 19) into different frequency sub-bands by passing the signal through a series of high-pass filters and a low-pass filters (see Equations 17 and 18) that simultaneously decompose the signal and provide a multi-resolution analysis of the signal [10]. The scaling function $\emptyset(t)$ and wavelet function $\varphi(t)$, facilitate the decomposition, approximation, reconstruction, and analysis of signals at different resolutions. It captures the low-frequency information and enables multiresolution analysis. The outputs give the detail coefficients (from the high-pass filter) and approximation coefficients (from the low-pass). The filter output of the high pass filter is discarded while the output of the low-pass filter is down sampled by 2 and further processed by passing it again through a new low-pass filter and a high- pass filter with half the cut-off frequency of the previous one.

At each level in Figure 4 the signal is decomposed into low and high frequencies. Due to the decomposition process the input signal must be a multiple of 2^n where n is the number of levels. This decomposition is repeated to further increase the frequency resolution while the approximation coefficients get decomposed with high- and low-pass filters and then down-sampled.

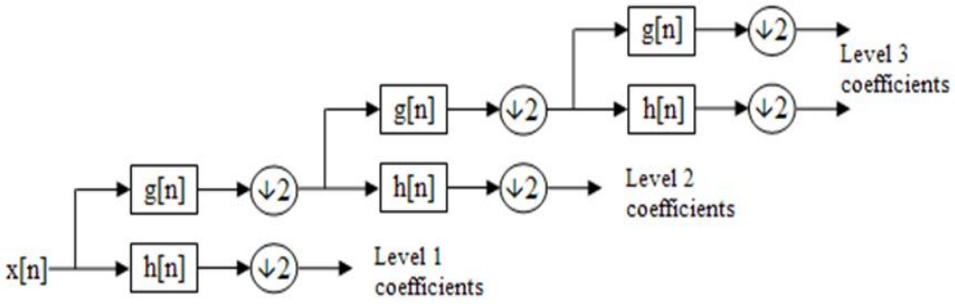

Fig. 4. A 3-level signal decomposition using DWT

The decomposition results in a set of wavelet coefficients that capture the frequency content of the signal at different scales or resolutions. The wavelet coefficients obtained from the DWT decomposition are used as features for speaker identification and the energy distribution is calculated over the coefficients in these sub-bands to extract relevant information [22]. Thirteen (13) DWT features were extracted.

$\emptyset(t) = \sum_{n=0}^{N-1} h[n]\sqrt{2}\emptyset(2t-1)$ (17)

where $\emptyset(t) =$ scaling function, h$[n]$ = impulse function of low pass filter

$\varphi(t) = \sum_{n=0}^{N-1} g[n]\sqrt{2}\emptyset(2t-1)$ (18)

where $\varphi(\mathrm{t})$ = wavelet function, and

$g[n]$ = impulse function of high pass filter

DWT of a discrete signal is given as

$DWT(m,k) = \frac{1}{\sqrt{a_o^m}}\sum_n x[n].\, g(\frac{n - n b_o a_o^m}{a_o^m})$ (19)

where $g(*)$ is the mother wavelet, and $x[n]$ = discrete signal

2.3.4 Gammatone Frequency Cepstral Coefficients Extraction Technique

Gammatone frequency cepstral coefficients (GFCC) are time-frequency features commonly used in audio processing that simulates the human hearing [22] similar to the MFCC technique. By simulating the characteristics of the human auditory system through gammatone filters, the GFCC feature extraction technique enables the extraction of perceptually meaningful features from audio signals. However, instead of triangular filter banks like MFCC, it uses an array of overlapping band pass filters called gammatone filter banks [29] with a high impulse response similar to the magnitude characteristics of the human auditory filter. The superior noise and reverberation robustness exhibited by the GFCC technique [11] makes it a first choice for extricating feature vectors from the corrupted speech sample for use in the recognition phase [23].

After pre-processing, framing and windowing to enhance the relevant speech information, gammatone filter was applied to the speech signal and then passed through an envelope extraction stage where a nonlinear half-wave rectification operation was performed to extract the envelope or the energy information from the filtered signal. The output from this stage was a log-compressed envelope which was transformed into the cepstral domain using a discrete cosine transform (DCT). The DCT coefficients represent the spectral shape of the signal and capture relevant information about the signal's harmonics and temporal dynamics. Since not all DCT coefficients are equally informative for the specific task at hand, a subset of the DCT coefficients, known as GFCCs, is selected based on their relevance and the final output consists of 36 coefficient values made up of cepstral coefficients, first order derivatives and second order derivatives. In this work, 13 cepstral coefficients were extracted. The block diagram of GFCC feature extraction is shown in Figure 5

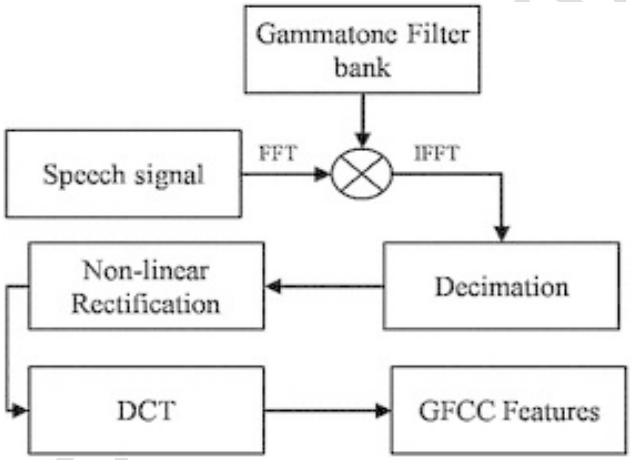

Fig. 5. GFCC feature extraction Block Diagram

2.4 Feature Combinations

The evaluation of machine learning algorithms with combined feature extraction techniques for speaker identification was done using the following feature combinations. The features were concatenated into a unified feature representation to different extents. Table 3 shows the feature combinations and the number of features involved (see appendix).

2.5 Algorithm Selection

The choice of machine learning algorithms was made on the basis of the most popular ones encountered in recent researches [6]. Other considerations made were nature of the problem with respect to speaker identification, the amount of data available for training and testing, and computational requirements of the system used for this work. The machine learning classifiers selected for this work were the random forest (RF), logistic regression

(LR), and linear discriminant analysis (LDA). The testing and training of the selected machine algorithms was performed using Python software.

2.6 Model Training

The random forest (RF), logistic regression (LR), and linear discriminant analysis (LDA) algorithms were used to construct the respective models using the concatenated features representations of the speech signals. The data was split into into training and testing sets. The method employed for data splitting was the 10-fold cross validation method and the models were trained on the training datasets and then used to predict the identity of the speakers from the utterances.

2.6.1 Random Forest Classifier

The supervised learning random forest algorithm was utilized in its classification capacity to classify the speaker features. It operates by constructing numerous decision trees during the training phase and determining the class that appears most frequently (for classification) from the collective outputs of the individual trees [24].

Using bootstrap aggregation, random forest was used to build individual decision trees and to train each decision tree on different subset of the data. The recursive partitioning of the data for each tree based on the selected data was done without any hyperparameter tuning. At the end of building the decision trees, the random forest classifier made predictions by aggregating the outputs of individual trees which was done by majority voting where the class that appeared the most among the trees was selected as the final prediction. This randomness helps to reduce the correlation between individual trees and improves the diversity of the ensemble.

2.6.2 Logistic Regression Classifier

Logistic Regression is a reliable and precise statistical modelling technique. In this paper, logistic regression was chosen because of its limited number of parameters and inclusion of a bias parameter to address overfitting. It also has the advantage of being extended to handle multiclass problems [25] [7] using techniques such as one-vs-rest or softmax regression. It was used to model the relationship between the linear combination of predictors and response using the logistic sigmoid function (see equation 20) on the labelled data. Using the maximum likelihood estimation (MLE) optimization technique, the logistic regression coefficients estimated were used to obtain the maximum likelihood of the observed data. After training the model, new data was introduced by plugging in the values of the predictor variables into the logistic function and allowing the model to predict the probability of the event occurring for new observations because the datasets were labelled

and the values of the predictors and the corresponding outcomes were known. A standard sigmoid function [26] is given by:

$$f(x) = \frac{1}{1+e^{-x}} \quad (20)$$

where x is input to the sigmoid function, and e is Eulers number given by 2.781

2.6.3 Linear Discriminant Analysis Classifier

Linear Discriminant Analysis (LDA) is a statistical technique used for dimensionality reduction and classification tasks. It aims to find a linear combination of features that maximally separates different classes in a datase [27].

In this work, the labelled datasets were fed as input to the LDA machine learning classifier where the mean vectors and covariance matrices for each class in the datasets were computed to obtain the between-class scatter and within-class scatter [28]. These scatter matrices provide information about the separability of the classes. The scatter matrices were then used to find the linear discriminants which define the subspace where a linear classifier is applied to determine the class labels. LDA has several advantages as a classifier when the classes are well-separated in the feature space. It can handle datasets with strong linear relationship between independent variables (multicollinearity), reduces the dimensionality of the data, and is relatively resistant to overfitting. However, it assumes that the data is normally distributed and that the class covariances are equal.

2.7 Performance Evaluation Metrics

The performances of the different models were evaluated using the pandas, numpy and sklearn libraries in python. The aim was to assess the performance of the model [29] and to determine which set of concatenated features best suit which model. It provides insights into the interactions between the different algorithms and the concatenated features. This process will aid in providing understanding of the synergic or complementary nature of the concatenated features and their contribution to enhanced model performance as well as better accuracy, robustness, and generalization capabilities of machine learning models [15]. The following evaluation metrics were used to evaluate the performance of the different models.

2.7.1 Accuracy

The accuracy evaluation metric was used to measure the proportion of correctly classified instances in the datasets. It shows how frequently the classifier predicts the correct values and gives the percentage of the samples which were correctly classified from all the samples given [30]. It was calculated as the number of correct predictions divided by the

total number of predictions. It gives an intuitive measure of the model's performance for a balanced dataset and an evaluation of the overall performance. Accuracy is given by

$$Accuracy = \frac{Number\ \ of\ correct\ \ Predictions}{Total\ \ Number\ \ of\ Predictions} \quad (21)$$

2.7.2 Average Prediction (%)

The average prediction (%) was used to indicate the overall confidence of the model in the predictions made for a specific speaker. After training the model to categorize speech segments as either belonging to a certain speaker or not, each categorization choice was assigned a probability or confidence score by the model that indicates the likelihood that the segment belongs to the designated speaker. This score makes up the average prediction score. A high average prediction score for a speaker indicates that the model consistently assigns high confidence scores to that speaker across multiple speech segments, suggesting that the model is effective at identifying that speaker. On the other hand, a low average prediction score indicates that the model is less confident about the identity of the speaker, or that the speaker's speech segments are more difficult to distinguish from those of other speakers.

3.0 RESULTS AND DISCUSSION

In all the results presented from Figures 7 to 10, the (a) part shows the left axis of the plot representing the speakers' prediction accuracy in percentage, while the number of features is plotted on the right axis. On the bottom axis bars are used to represent the number of concatenated features and the plot markers represent the performance of the three different ML algorithms. In Figure (b), the left axis of the plot represents the average score of accurate predictions in percentage while the number of features is plotted on the right axis. On the bottom axis, bars are used to represent the number of concatenated features, and the plot markers represent the performance of the three different ML algorithms. These results show the effects of different combinations of these extracted features on the performance of the three considered machine learning algorithms.

3.1 Results for Voice Activity Detection (VAD) Stage

The results of voice activity detection (VAD) performed on the datasets obtained from the TIMIT database using power density, entropy, zero crossing rate and root mean square measures are presented in Figures 6.

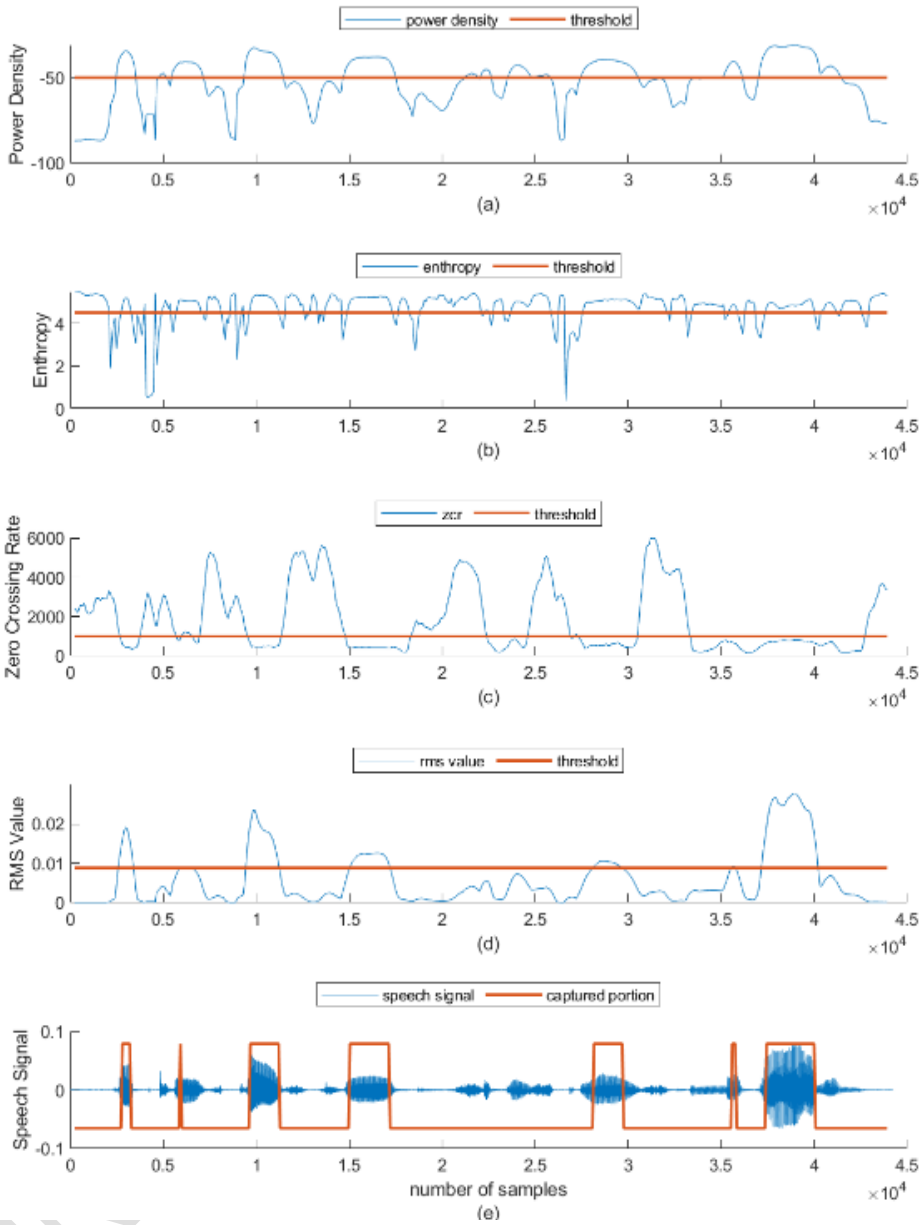

Fig. 6. Plots of voice activity detection results for TIMIT dataset: (a) power density; (b) entropy; (c) zero crossing rate; (d) root mean square value; (e) speech signal with the framed sections over the regions of voice activity. Source: Researcher, 2023.

Figure 6 shows that the portions (frames) of the utterance containing voice activities were successfully detected and enveloped as seen in (e) for TIMIT database. The frames enveloped had a power density value above 50 dB, entropy of more than 4.5, zero crossing rate of less than 1000, and RMS value of more than the overall utterance's RMS value. These enveloped frames were the ones whose features were extracted for model training and testing.

3.2 Evaluation of the performance of the different ML algorithms with LPCC features in combination with other features

Figure 7 shows speaker prediction results with LPCC features and its combinations.

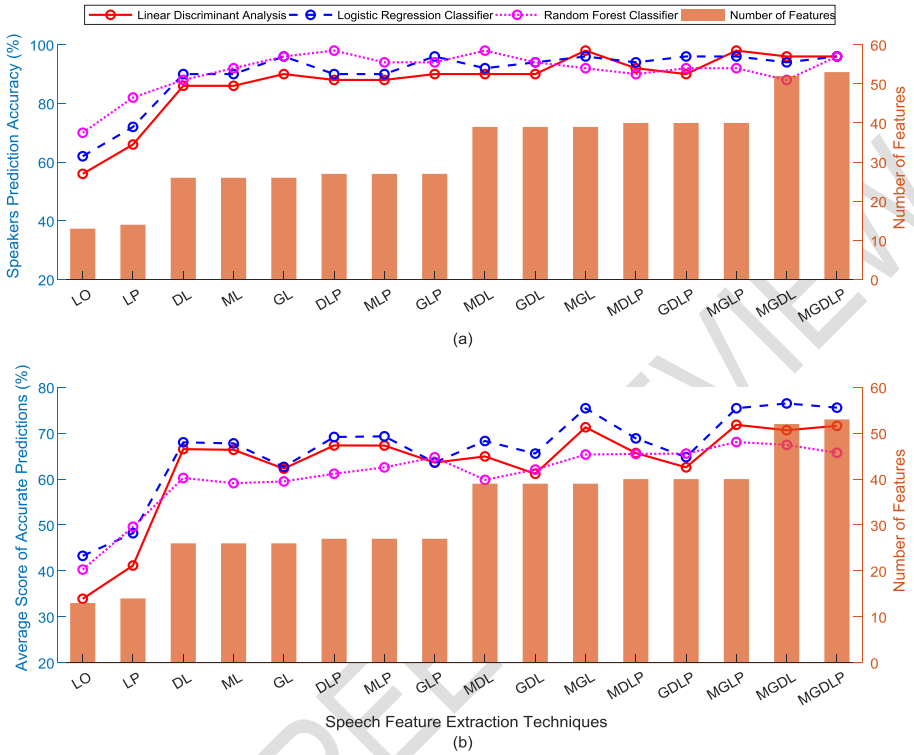

Fig. 7. Speaker prediction results with LPCC features and its combinations

The results indicate that the performance of ML algorithms for speaker prediction accuracy varies based on the number of features used. In figure 7(a), when using a low number of features (LO) with 13 features, all ML algorithms achieved speaker prediction accuracy ranging from 55% to 70%. However, in figure 7(b), the ML algorithms' performances for the average score of accurate predictions were below 30% at LO. On the other hand, when using a higher number of features (MGDLP) with 53 features, all three ML algorithms achieved speaker prediction accuracy above 80%. Additionally, in figure 7(b), only the Logistic Regression (LR) algorithm demonstrated a performance above 70% for the average score of accurate predictions at MGDLP. For the feature combinations between LO and MGDLP, the ML algorithms' performance varied widely, ranging from 30% to 100% for both speaker prediction accuracy and the average score of accurate predictions.

3.3 Evaluation of the performance of the different ML algorithms with GFCC features in combination with other features.

The speaker prediction results with GFCC features and its combinations is shown in Figure 8.

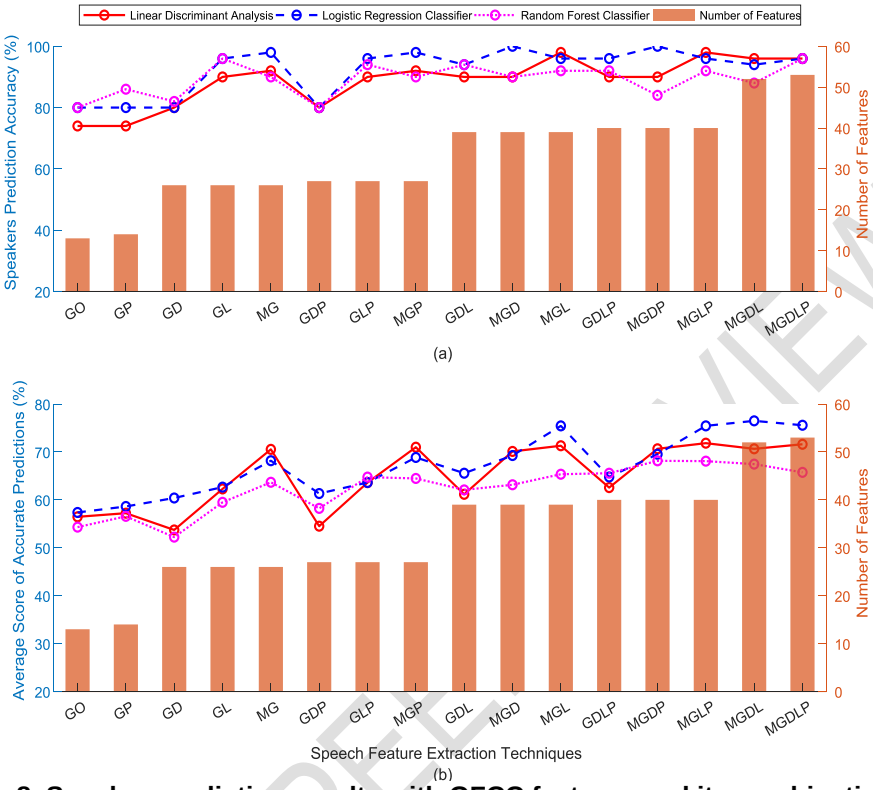

Fig. 8. Speaker prediction results with GFCC features and its combinations

In Figure 8(a), when using a relatively low number of features (GO) with 13 features, all ML algorithms achieved speaker prediction accuracy between 70% and 80%. However, when using a higher number of features (MGDLP) with 53 features, all three ML algorithms achieved a speaker prediction accuracy close to 100%. In Figure 8(b), the performances of all ML algorithms for the average score of accurate predictions were between 50% and 60% at GO. However, at MGDLP with 53 features, the Logistic Regression (LR) algorithm achieved a higher performance of about 76% for the average score of accurate predictions. For the feature combinations between GO and MGDLP, the performance of the ML algorithms varied between 55% and 100% for both speaker prediction accuracy and the average score of accurate predictions.

3.4 Evaluation of the performance of the different ML algorithms with MFCC features in combination with other features

Figure 9 shows speaker prediction results with MFCC features and its combinations.

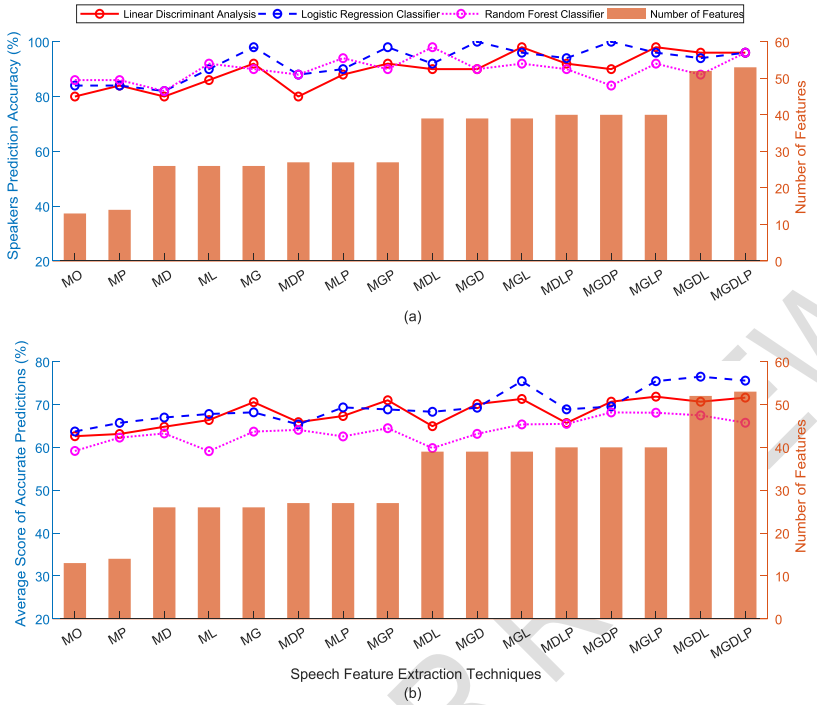

Fig. 9. Speaker prediction results with MFCC features and its combinations

The machine learning (ML) results show that the performance of ML algorithms for speaker prediction accuracy varies based on the number of features used. In figure 9(a), when utilizing a moderate number of features (MO) with 13 features, all ML algorithms achieved speaker prediction accuracy between 80% and 90%. However, when employing a higher number of features (MGDLP) with 53 features, all three ML algorithms achieved speaker prediction accuracy above 90%. In figure 9(b), the performances of all ML algorithms for the average score of accurate predictions were above 55% at MO. At MGDLP with 53 features, the Logistic Regression (LR) algorithm exhibited a performance above 70% for the average score of accurate predictions.

For the feature combinations between MO and MGDLP, the performance of the ML algorithms fell between 55% and 90% for both speaker prediction accuracy and the average score of accurate predictions.

3.5 Evaluation of the performance of the different ML algorithms with DWT features in combination with other features

The speaker prediction results with DWT features and its combinations is shown in Figure 10.

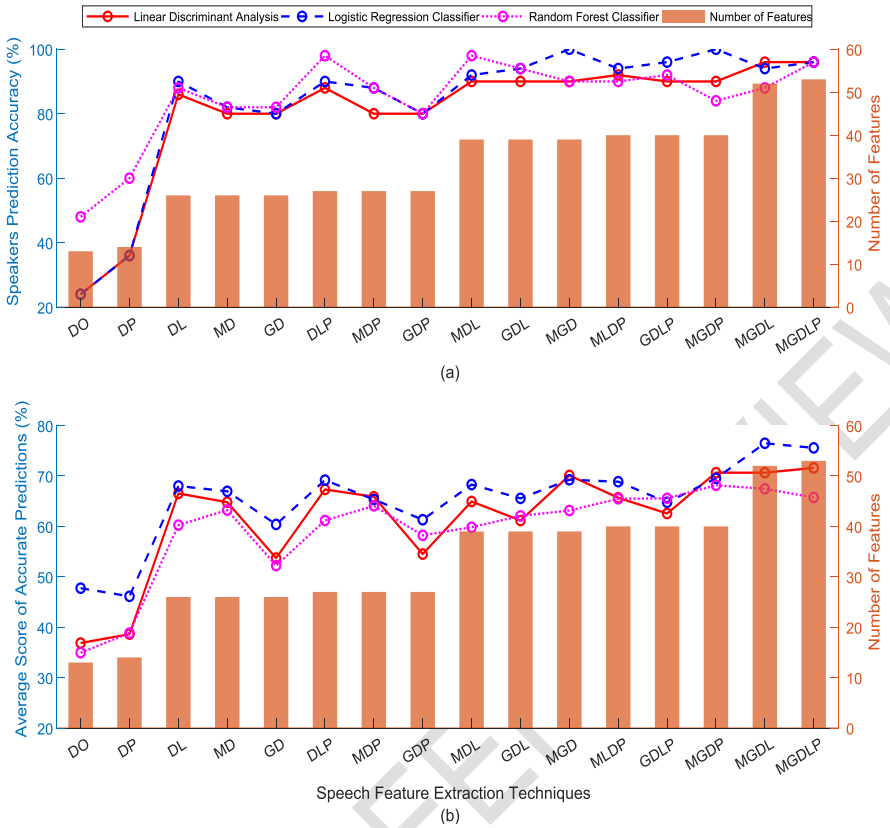

Fig. 10. Speaker prediction results with DWT features and its combinations

The machine learning (ML) results demonstrate that the performance of ML algorithms for speaker prediction accuracy varies depending on the number of features used. In figure 10(a), when utilizing a relatively low number of features (DO) with 13 features, the performances of Logistic Regression (LR) and Linear Discriminant Analysis (LDA) algorithms were below 30% for speaker prediction accuracy. However, when employing a higher number of features (MGDLP) with 53 features, all three ML algorithms achieved speaker prediction accuracy above 90%.

In Figure 10(b), the performances of LDA and Random Forest (RF) algorithms for the average score of accurate predictions were below 40% at DO. At MGDLP with 53 features, the LR algorithm demonstrated a performance close to 80% for the average score of accurate predictions.

For the feature combinations between DO and MGDLP, the performance of the ML algorithms ranged between 30% and 80% for both speaker prediction accuracy and the average score of accurate predictions.

4.0 DISCUSSION

The findings made from these results presented in Fig. 6 to 10 are as follows:

i. The results obtained suggest that increasing the number of features used in the ML algorithms, (specifically from LO to MGDLP or GO to MGDLP or MO to MGDLP or DO to MGDLP), generally improves the speaker prediction accuracy. At MGDLP, all ML algorithms achieved high accuracy levels above 90%. However, there is still some variation in the performance within the feature combinations, indicating the impact of different feature sets on the ML algorithms' performance in speaker prediction tasks. It is worth noting that the Logistic Regression algorithm consistently outperforms the other ML algorithms, particularly in terms of the average score of accurate predictions.

ii. DWT only (DO) is the least performed of all feature combinations for both identification rates (a) and average prediction percentage (b) for all trained machine learning models.

iii. The LR model showed an average score of accurate predictions performance of approximately 70% for the highest number of feature combinations (MGDLP) which represents 53 features and then a best performance score of approximately 76% for average score of accurate predictions at the MGL combination which represents 39 features.

iv. Another observation is that the logistic regression (LR) model performs very well with feature combinations that have all the short-term cepstral features (MFCC, LPCC and GFCC) in combinations (see figure 7, figure 8 and figure 9). Several studies in the literature have shown that MFCC features in particular [31] and cepstral features in general outperform other feature types because they are more robust. This can be related to the conclusion arrived at by [29] that the Cepstral domain features are more superior.

5. CONCLUSION

In conclusion, this work presented the evaluation of machine learning algorithms with combined feature extraction techniques for speaker identification. A brief description of the data collection, data processing, feature extraction, model training, evaluation and results have been presented to show the performance of the different machine learning models under different combinations of datasets.

It is therefore inferred that –

i. cepstral features have a more positive effect towards the performance of speaker recognition systems.

ii. the performance of the machine learning models can indeed be affected by different features sets. This therefore means that the right feature combinations can offer complementary synergy which can improve the performance of the models.

LR model performed best with average score of accurate predictions approximately 70% for the maximum number of feature combinations (MGDLP) and also had the best performance for average score of accurate predictions (approximately 76%) at the MGL. This means that the optimal performance for LR was at the MGL (39) features set and as the features sets increased further the model hit a ceiling and the performance started to decline. This implies that an increase in speaker specific training data will increase the performance of the system, however, too much training data has been proven to be unnecessary because the performance of the system will eventually reach its highest point [32].

The LR model outperformed all other models and it was observed that this performance occurred where all three cepstral features (MFCC, GFCC and LPCC) were part of the concatenated features. Also, as the number of features increased the performance of the LR model hit a ceiling and did not improve further. This work can be applied to biometric applications such as forensics, security access and in remote access for financial institutions.

REFERENCES

1. Furui S. 40 years of progress in automatic speaker recognition. Advances in biometrics. 2009.
2. Hermansky H. Perceptual linear predictive (PLP) analysis of speech. The Journal of the Acoustical Society of America . 1990;87(4):1738–52.
3. Koehler J, Morgan N, Hermansky H, Hirsch HG, Tong G. Integrating RASTA-PLP into speech recognition. In: Proceedings of ICASSP '94 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE; p. I/421-I/424.
4. Garima Sharma, Kartikeyan Umapathy SK. Trends in audio signal feature extraction methods. Applied Acoustics. 2020;158:2–21.
5. Todkar SP, Babar SS, Ambike RU, Suryakar BP, Prasad JR. Speaker Recognition Techniques: A Review. 3rd International Conference for Convergence in Technology (I2CT). 2018;1–5.
6. Ayvaz U, Gürüler H, Khan F, Ahmed N, Whangbo T, Bobomirzaevich A. Automatic speaker recognition using mel-frequency cepstral coefficients through machine learning. CMC-Computers Materials & Continua. 2022;71(3).
7. Mokgonyane, T. B. , Sefara, T. J. , Manamela, M. J. and Modipa TI. The effects of data size on text-independent automatic speaker identification system. In: In2019 International Conference on Advancesin Big Data, Computing and Data Communication Systems (icABCD) IEEE. 2019. p. 1–6.
8. Kaphungkui NK, Kandali AB. Text dependent speaker recognition with back propagation neural network. Int J Eng Adv Technol (IJEAT). 2019;8(5):1431–4.

9. Paulose S, Mathew D, Thomas A. Performance Evaluation of Different Modeling Methods and Classifiers with MFCC and IHC Features for Speaker Recognition. Procedia Comput Sci. 2017;115:55–62.
10. Singh, R. K., Saha, R., Pal, P. K., & Singh G. Novel feature extraction algorithm using DWT and temporal statistical techniques for word dependent speaker's recognition. In: In 2018 Fourth International Conference on Research in Computational Intelligence and Communication IEEE Networks (ICRCICN). 2018. p. 130–4.
11. Hanf RM, Isa K, Mohammad S. Comparative Analysis on Different Cepstral Features for Speaker Identification Recognition. In: IEEE Student Conference on Research and Development (SCOReD). Johor, Malaysia; 2020. p. 487–92.
12. Chakroun R, Frikha M. Robust Text-independent Speaker recognition with Short Utterances using Gaussian Mixture Models. In: 2020 International Wireless Communications and Mobile Computing (IWCMC). IEEE; 2020. p. 2204–9.
13. Faghani M, Rezaee-Dehsorkh H, Ravanshad N, Aminzadeh H. Ultra-Low-Power Voice Activity Detection System Using Level-Crossing Sampling. Electronics (Basel). 2023;12(4):795.
14. Welch P. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. IEEE Transactions on Audio and Electroacoustics. 1967 Jun;15(2):70–3.
15. Mokgonyane TB, Sefara TJ, Manamela MJ, Modipa TI. Development of a text-independent speaker recognition system for biometric access control. In Southern Africa telecommunication networks and applications conference (SATNAC) . 2018;128–33.
16. Jahangir R, Teh YW, Nweke HF, Mujtaba G, Al-Garadi MA, Ali I. Speaker identification through artificial intelligence techniques: A comprehensive review and research challenges. Expert Syst Appl. 2021;171:114591.
17. Alasadi AA, Deshmukh RR, Adhyani HHT, Alahmadi AH, Alshebami AS. Efficient Feature Extraction Algorithms to Develop an Arabic Speech Recognition System. Engineering Technology & Applied Science Research. 2020;10(3):5547-5553.
18. Picone JW. Signal modelling techniques in speech recognition. Proc IEEE . 1993;81:1215–47.
19. Namrata Dave. Feature Extraction Methods LPC, PLP and MFCC In Speech Recognition. International Journal for Advance Research in Engineering and Technology. 2013;1:1–5.
20. Siddhant C. Joshi, Cheeran AN. MATLAB Based Feature Extraction Using Mel Frequency Cepstrum Coefficients for Automatic Speech Recognition. International Journal of Science, Engineering and Technology Research (IJSETR), . 2014;3:1820–4.
21. Tiwari V. MFCC and its applications in speaker recognition. International Journal on Emerging Technologies. 2010;1(1):19–22.
22. Sekkate, S., Khali, M. and Adib A. A Feature Level Fusion Scheme for Robust Speaker Identifiction. BDCA. 2018;289–300.
23. Jain, V.K., Tripathi N. Speech Features Analysis and Biometric Person Identification in Multilingual Environment. IJSRNSC. 2018;6(1).
24. Gupta H, Gupta D. LPC and LPCC method of feature extraction in speech recognition system. In: In: 2016 6th international conference - cloud system and big data engineering (Confluence), Noida, 2016. Noida; 2016. p. 498–502.
25. Harrell JFE, Harrell FE. Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis. Ordinal logistic regression. 2015;311–25.
26. Deniz Tuzsuz. https://www.learndatasci.com/glossary/sigmoid-function/ . 2023. Sigmoid Function.

27. Tronci EM, Beigi H, Feng MQ, Betti R. A transfer learning SHM strategy for bridges enriched by the use of speaker recognition x-vectors. J Civ Struct Health Monit. 2022;12(6):1285–98.
28. Jin Q, Waibel A. Application of LDA to speaker recognition. In Interspeech . 2000;250–3.
29. Mokgonyane TB, Sefara TJ, Manamela MJ, Modipa TI, Masekwameng MS. The effects of acoustic features of speech for automatic speaker recognition. In 2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD), IEEE. 2020;1–5.
30. Ismail M, Memon S, Dhomeja LD, Shah SM, Hussain D, Rahim S, et al. Development of a regional voice dataset and speaker classification based on machine learning. . J Big Data. 2021;8:1–18.
31. Rozario MS, Thomas A, Mathew D. Performance Comparison of Multiple Speech Features for Speaker Recognition using Artificial Neural Network. In: In 2019 9th International Conference on Advances in Computing and Communication (ICACC) . 2019. p. 234–9.
32. Bekli Z, Ouda W. A performance measurement of a Speaker Verification system based on a variance in data collection for Gaussian Mixture Model and Universal Background Model. 2018.