International Journal of Informatics and Computation (IJICOM)
Vol. 7, No. 1, 2025
ISSN: 2685-8711, E-ISSN: 2714-5263 DOI: 10.35842/ijicom

Enhancing Speaker Verification System using MFCC and HMM Method

Alifya Ma'sum[1], Budi Darmawan[2], Giri Wahyu Wiriasto[3]

Abstract

Speaker verification is a system to identify a person's identity using a person's characteristic data. In this study, we utilize the Mel Frequency Cepstral Coefficients (MFCC) to extract voice data characteristics with several stages including pre-emphasis, frame blocking, windowing, Fast Fourier Transform (FFT), Mel Frequency Wrapping, and discrete Cosine Transform (DCT) and employ the Hidden Markov Model (HMM) classify voice data. This study uses 30 speaker voice data with 1500 speaker voice data samples and conducts preprocessing and feature extraction using several state configurations. The test results can harvest values from 40% to 100% with the most accurate voice verification accuracy found in state 4 with 100% accuracy. The test results show that a combination of MFCC and HMM methods can be an accurate approach to the speaker verification process in real applications.

Keywords:
HMM, MFCC, Speaker Verification System, Voice.

1. Introduction

Speaker recognition is the process of recognizing a speaker from their speech. It can be used in many aspects of life, such as remotely retrieving access to personal devices, securing access to voice control, and conducting forensic investigations. In speaker recognition, extracting features from speech is the most important process. The features are used to represent the speech as unique features to distinguish speech samples from each other [1]. Voice verification in the forensic field has also been used in Indonesia. As in several corruption cases in Indonesia that use evidence in the form of voice recordings, it has been successfully proven by voice verification. The method used in Indonesia is to compare the voice recording used as evidence with one or more accused voices. The application of this method requires several experts as respondents to analyze the level of similarity of voices in the recording. However, the accuracy of this method is highly dependent on expert conditions [2].

In voice signals, many parameters need to be considered, ranging from subjective parameters such as the accent produced, and the dialect used, as well as objective parameters (can be measured acoustically). The problem faced is how to extract characteristics from complex voice signals so as to produce new data that is more practical without losing the characteristics of the voice signal [3].

MFCCs are widely used spectral features for speaker recognition and text-dependent

Corresponding Author: 1 Alifya Ma'sum, University of Mataram, maksumalifya@gmail.com
2 Budi Darmawan, University of Mataram, budidarmawan@unram.ac.id
3 Giri Wahyu Wiriasto, University of Mataram, giriwahyuwiriasto@unram.ac.id

speaker recognition systems are the most accurate in voice-based authentication systems [4]. HMM consists of a Markov chain in the first part that hides the state therefore the internal behavior of the model remains invisible. The hidden states of the model capture the temporal structure of the data. HMM is a statistical model that describes a sequence of events.[5]. The MATLAB system will read the WAV file, play it first, and then calculate the characteristic parameters automatically. All the contents of the speech signal have been distinguished in the last step [6].

Two things that are problematic in the speaker verification process are inter-speaker distance and intra-speaker variability. Inter-speaker distance is the characteristic between speakers distinguished by population distribution factors of the stability of the speaker's utterance length in the relevant parameter space. Intra-speaker variability is caused by the dependence between texts, random variations in speaker pronunciation, fatigue effects, clarity in emotions, or illness conditions (cold). In vocal pronunciation under external acoustic conditions (for example noise).[7]

This speech recognition problem is interesting and important because there are problems in identifying speakers due to variations in the pronunciation of each syllable. The very specific characteristics of a voice signal are caused by differences in physiological structure and aspects of innateness in each individual. For this reason, a machine learning algorithm is needed that can be used to extract voice features with the discrete fourier transform (DFT) algorithm so that a digital signal is obtained, then described, the results of feature extraction with the DFT method and then implemented to find the results of the verification [8].

To solve the speaker verification problem, we propose the development of a speaker verification system using MFCC and HMM methods. The proposed solution involves the implementation of the MFCC algorithm for voice feature extraction and the use of HMM for speaker modeling and classification. In addition, we utilize MATLAB as a development platform to speed up the analysis and visualization of results.

2. Related Works

The speaker verification system is a growing research topic with various approaches to achieve promising results with efficient computation [16][17][18][19][20]. An article constructed a speaker verification System with Wavelet-MFCC and HMM Classifier to test and select the best channel from the wavelet-MFCC process that can be used as a new character coefficient. The new feature coefficient is then called the Wavelet- MFCC feature coefficient to find the detail channel (cD) as a feature. It can provide the same accuracy as using the combined channel (cAcD) and is higher than the approximation channel (cA), with an accuracy of 95% [9].

Voice recognition systems also utilize MFCC as feature extraction methods that adopt the principle of human hearing senses. The study method starts from pre-emphasis, frame blocking, windowing, fast fourier transform, mel frequency wrapping, and cepstrum stages. The system can reach 90% and the percentage of system failure was 10% with a top 5 error rate of 0%, while in testing with non-ideal conditions, the percentage of system success was 76.6667% and the percentage of system failure was 23.333% with a top 5 error rate of 0% [10]. Another paper proposed MFCC and DTW to recognize the types of male and female voices. The results obtained are for the accuracy rate in women with alto voice type obtained 80% percentage, for mezzosopran voice type accuracy rate obtained 90%, for soprano voice type accuracy rate obtained 80%[11].

Another research discussed gender recognition of speakers with text-dependent and speaker-dependent speech, in the recognition process an extraction algorithm called

MFCC is used for feature extraction from speech signals while the clustering process uses the Vector Quantization (VQ) method. In the recognition stage, a distortion measure based on Euclidean distance minimization was used to match the test speakers with the speakers in the database. The speech database uses 20 speakers, consisting of 10 male speakers and 10 female speakers with an accuracy rate of 90% for males and 80% for females [12].

Speaker verification aims to determine whether the voice of the speaker who is speaking matches the speaker who has been registered in the system while speaker identification or speaker identification is the task of determining the identity of the speaker from a group of speakers based on the speech being tested, this verification data does not only contain data from a person who is speaking therefore a speaker embedder is needed, The embedder used is x-vectors which are embeddings or embeddings extracted from DNN (Deep Neural Networks) using the TDNN (Time Delay Neural Networks) architecture which is very effective for speaker verification [13].

Speaker verification can also be done using neural networks. LVQ is one type of neural network that uses supervised learning. LVQ learners are quite ideal because inputs that have similarities will be grouped, so that the results given in the learning process will have more accurate results. The purpose of using LVQ is to group inputs to outputs in vector classification in order to minimize the process of errors in classification [14].

3. Proposed Method

The speaker verification system using MFCC and HMM involves steps such as voice feature extraction with MFCC, followed by training an HMM model to recognize voice patterns. The mathematical formula includes calculating the probability of observation in the HMM model to match the voice with the trained model [15].

In the process of calculating the probability of observation, you can use the following equation formulas:

$$b_j(THE_t) = \frac{1}{1 + d\left(THE_{t,}\mu_j\right)} \quad (1)$$

With

$$d\left(THE_t, \mu_j\right) = \sqrt{\sum_{k=1}^{M} (THE_{tk} - \mu_{jk})^2} \quad (2)$$

Where $d\left(THE_t, \mu_j\right)$is the Euclidean distance between the observation series at time t and the average of the observation series at state j. And M is the number of elements of the observation series at state j.

To determine the probability of the HMM model, the following equation is required:

$$P(O|\lambda^k) \text{ for } 1 \leq k \leq K \quad (3)$$

The equation above uses the Viterbi algorithm in its implementation. Before deciding that k* is a speaker who has been successfully verified, it is necessary to make a comparison between the values$P(O|\lambda^k)$with the threshold value. If the value$P(O|\lambda^k)$is greater than the threshold value, then the speaker k* is successfully verified, and vice versa if the value$P(O|\lambda^k)$smaller than the threshold value then speaker k* is not verified.

To determine the threshold value for each voice data, you can use the following equation:

$$Threshold = P(min) + (ab/100) \quad (4)$$

with:

$$a = P(max) - P(min) \quad (5)$$

Where:

Notation	Description
P(max)	The probability value of one of the training data observation series for the HMM model that produces a larger probability value
P(min)	The probability value of one of the training data observation series for the HMM model that produces a smaller probability value
P(min)	The difference between the threshold percentage value (%) and the P(min) value in percent (%)

4. Experimental Setup

In Figure 4.1 it can be explained that the system built is a speaker verification system where this system aims to verify the identity of a speaker from the sound signal entered into the system. In the first stage, MFCC feature extraction is carried out. This stage is done to get a series of observations (O). So that each sound will be converted into an observation sequence (O). Furthermore, the results of sound feature extraction act as input to the HMM speaker verification system. Each observation sequence is then calculated as the probability of the observation sequence against the HMM speaker model P($O|\lambda$) with the Viterbi algorithm. Then a comparison is made between the value of the HMM speaker model P($O|\lambda$) with the threshold value. So, if the calculation result of the speaker probability exceeds the threshold value, then the speaker is verified or recognized and if the calculation result of the probability does not exceed the threshold value then the speaker is not verified or recognized.

For each speaker contained in the database, a separate threshold value is determined which can be found using the equation:

$$Threshold = p(min) + \left(\frac{ab}{100}\right) \quad (6)$$

Where:

$$a = P(max) - P(min)$$

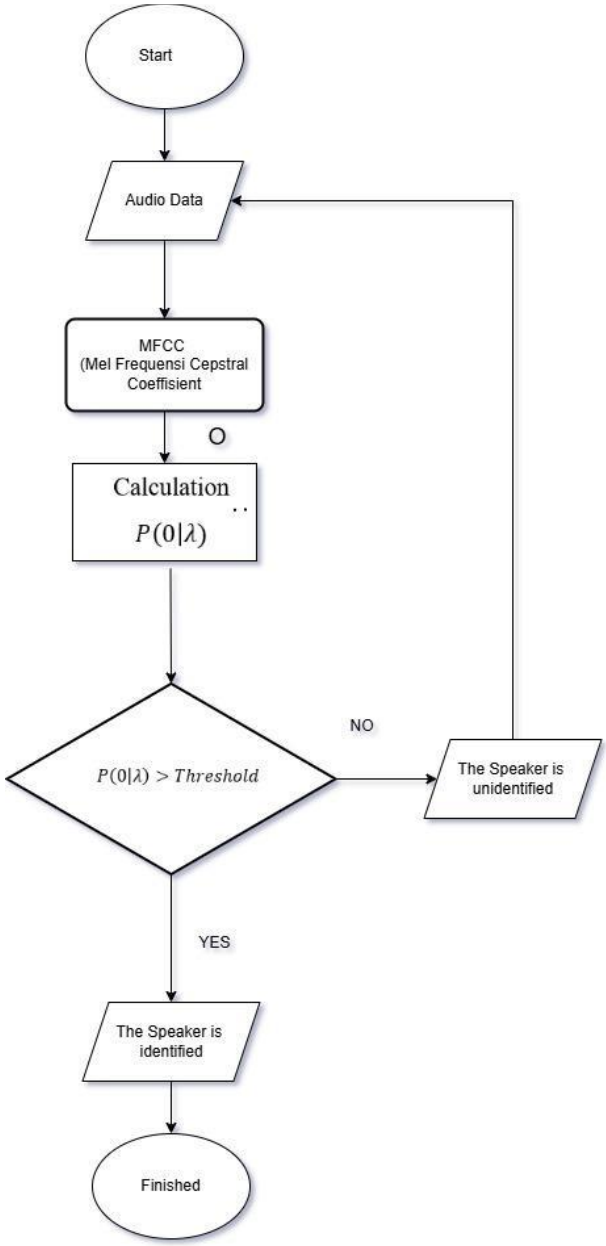

Fig. 4.1 Speaker Verification System Diagram

In this speaker verification system, the Viterbi algorithm is also used, which is closely related to the HMM method. The Viterbi algorithm is one of the core algorithms used in the context of the HMM method. The steps in calculating the Viterbi algorithm are seen in the following equation:

3.1 Initialization

$$\delta_1(i) = \pi_1 b_1(0_1), 1 \leq i \leq N \tag{7}$$

$$\varphi_1(i) = 0 \tag{8}$$

Where:

Notation	Description
$\delta_1(i)$	the maximum probability of all possible paths ending in the state, given the initial probabilities and the first observation. This is the initialization of the Viterbi algorithm.
π_1	the initial probability that the system is in the -th state at the time.
$b_1(0_1)$	emission probability, which is the probability that the state k results in an observation.
$\varphi_1(i)$	part of the backtracking process. Since this is the first step, there is no previous state to track, so it is set to zero.

3.2 Recourse

$$\delta_t(i) = max_{1\le i\le N}\left[\delta_{t-1}(i)^* a_{ij}\right] * b_j(the_{t)} \quad (9)$$

$$\varphi_t(i) = max_{1\le i\le N}\left[\delta_{t-1}(i)^* a_{ij}\right] \quad (10)$$

For:

$2 \le t \le T$

$1 \le j \le N$

Where:

Notation	Description
$\delta_t(i)$	The maximum probability of all possible state paths ending in the -th state at the -th time, considering all previous states.
$\delta_{t-1}(i)^*$	The maximum value of the probability of a path ending in state at time.
a_{ij}	Transition probability from state to state
$b_j(the_{t)}$	The probability that a state produces an observation
$\varphi_t(i)$	Stores the state index from the previous time that gives the maximum probability of reaching the state at a time. This is used to track the optimal path at the final step (backtracking)

3.3 Termination

$$P(\lambda) = max_{1\le i\le N}[\delta_T(i)] \quad (11)$$

$$q_t^* = angry\ max_{1\le i\le N}[\delta_T(i)] \quad (12)$$

Where:

Notation	Description
$P(\lambda)$	It is the maximum probability of all possible state paths that result in a sequence of observations, given an HMM model.

$\delta_T(i)$	The maximum probability of all state paths ending in the - th state at a time.
q_t^*	The state at a time that gives the maximum probability. This is the last state of the optimal path, and will be used for backtracking the path backward.

3.4 Traverse backtracking

$$q_t^* = \varphi_{t+1}(q_t^*) \tag{13}$$

For:

$t = T-1, T-2, \ldots, 1$

Where:

Notation	Description
q_t^*	Shows the optimal state at a time.
φ_{t+1}	It is a back pointer function or back transition function, which provides the previous state that is most likely to result in the state.

5. Result and Analysis

5.1. Discussion

Fig. 5.2 to Fig. 5.5 shows some graphs of the results of testing the speaker verification system with the number of states 2 to 5. From the graph, it can be seen that the speaker verification system has a different level of accuracy in each state. It can be seen that the first speaker is used as a voice sample that has been verified. The speaker accuracy level that is successfully verified is in state 4 with the number of verified voice data as much as 30 voice data (100%) and the lowest accuracy level is in state 3 with the number of verified voice data as much as 12 voice data. This is because there is some voice data that has been successfully verified by the system but is retracted by the system.

In Fig. 5.2 to Fig. 5.5 voice data other than the speaker can be seen that the higher the threshold value, the higher the accuracy of the system. This is caused by the amount of voice data used then from the data a lot of voice data is rejected because it is not able to pass the threshold value used in the speaker verification system.

5.2. Speaker Verification System Testing State 2

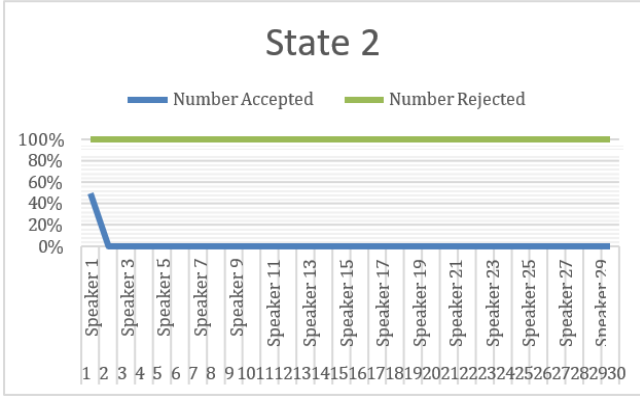

Fig. 5.2 Test result graph of speaker verification system with 2 states

Fig. 5.2 shows the speaker verification system with the number of states 2 has a verified voice data accuracy of 50% (15 voice data) of the total voice data tested as much as 30 voice data.

5.3. Speaker Recognition System Testing State 3

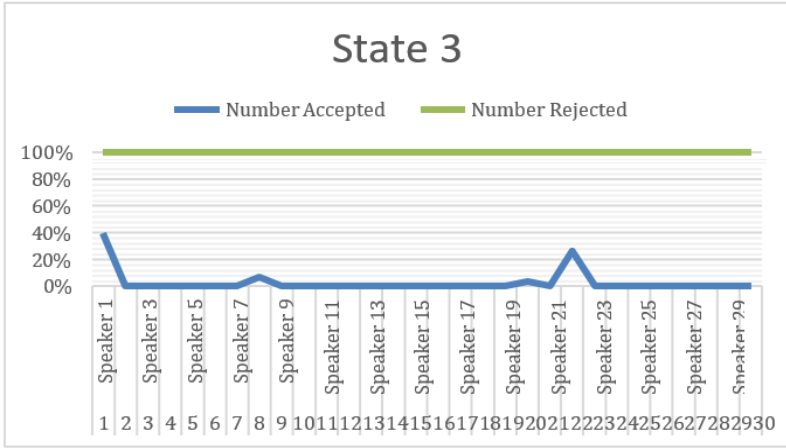

Fig. 5.3 Test result graph of speaker verification system with states 3

Fig. 5.3 shows the speaker verification system with the number of states 3 has a verified voice data accuracy rate of 40% (12 voice data) of the total voice data tested as much as 30 voice data.

5.4. Speaker Verification System Testing State 4

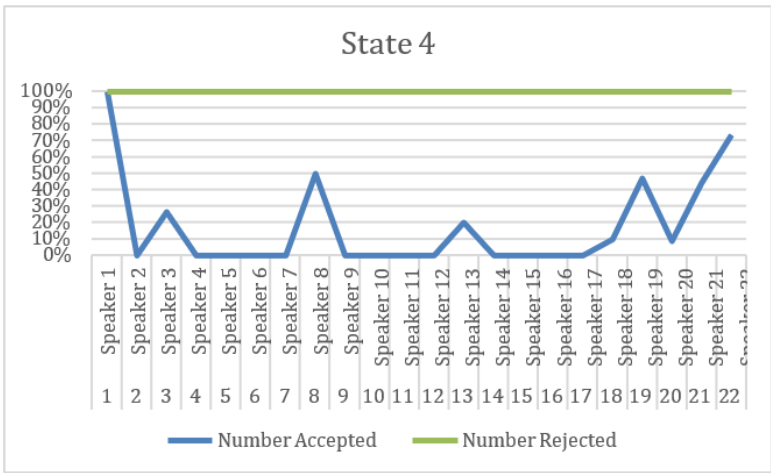

Fig. 5.4 Test result graph of speaker verification system with 4 states

Fig. 5.4 shows the speaker verification system with the number of states 4 has a verified voice data accuracy rate of 100% (30 voice data) or successfully verifies all the voice data tested.

5.5. Speaker Verification System Testing State 5

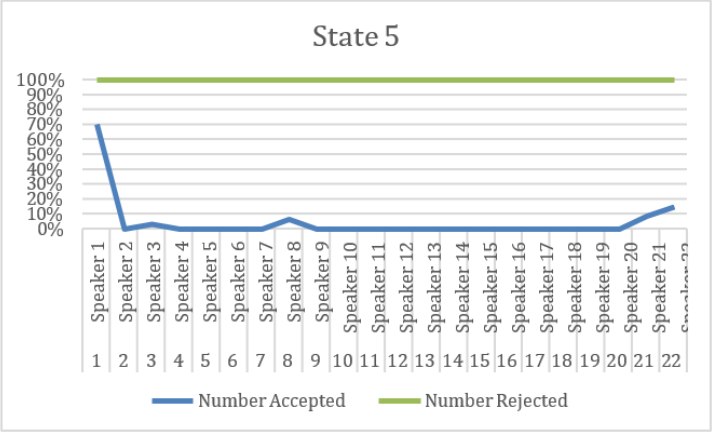

Fig. 5.5 Test result graph of speaker verification system with 5 states

Fig. 5.5 shows the speaker verification system with the number of states 5 has an accuracy rate of 70% (21 voice data) of the total voice data tested as much as 30 voice data.

6. Conclusion

The research with a combination of MFCC and HMM for speaker verification systems can achieve optimal performance by reaching an accuracy of 100%. In this condition, the system successfully verifies all voice samples that belong to registered speakers and indicates that the model effectively learns and identifies voice characteristics when the input matches stored voice patterns. According to the experimental result with many samples, this approach confirms can merely verify voices that exist within its database. Therefore, the technique can reinforce its reliability in rejecting unauthorized inputs to build a secure speaker verification system.

Acknowledgment

Thank you to all friends and people who have provided support for this research, especially to the supervisor, and also do not forget to parents who always provide more support in the process of completing this research.

References

[1] S. Hidayat, R. Hidayat, and T. B. Adji, “Indonesian speech recognition system based on syllables using MFCC, Wavelet and HMM,” 2015.

[2] Kurniawan, “Voice verification using artificial neural networks and Mel Frequency Cepstral Coefficient feature extraction,” *Journal of Business Information Systems*, vol. 7, no. 1, p. 32, May 2017, doi: 10.21456/vol7iss1pp32-38.

[3] H. Muhammad Arkaan, I. Fauzi, L. Windar Al Rosyid, and A. Junaidi, “MATLAB-based human voice characteristic classification using Fast Fourier Transform method,” *Journal of Informatics, Information Systems, Software Engineering and Applications*, vol. 2, no. 1, pp. 1–6, 2019, doi: 10.20895/inista.v2i1

[4] Anegundi, C. D. A, V. V. Pawale, and R. S. B, “A computer-based application for speech recognition in a multi-speaker environment to assist hearing-impaired people,” 2019.
[5] L. Dewi Astuti and W. Wibisono, “Network lifetime improvement on wireless sensor network using Clustered Shortest GeoPath Routing (C-SGP) protocol,” 2017. [Online]. Available: http://www.jtiik.ub.ac.id
[6] L. Pan, “Faculty of Engineering and Sustainable Development,” 2013.
[7] Anonymous, “Speaker identification system using TESPAR method and multilayer perceptron (MLP) architecture.”
[8] Anonymous, *Gender identification through voice using Discrete Fourier Transform (DFT)*, USU Press, 2014.
[9] S. Hidayat, A. Sofyan Anas, S. Agrippina, A. Yusuf, and M. Tajuddin, “Speaker recognition system using Wavelet-MFCC method and Hidden Markov Models (HMM) classification,” *Journal of Information Technology and Computer Science*, vol. 8, no. 1, pp. 119–126, 2021, doi: 10.25126/jtiik.202183284.
[10] G. Ajinurseto, L. O. Bakrim, and N. Islamuddin, “Application of Mel Frequency Cepstral Coefficients method in desktop-based speech recognition system,” *Infomatek*, vol. 25, no. 1, pp. 11–20, Jun. 2023, doi: 10.23969/infomatek.v25i1.6109.
[11] Sidik Permana, Y. I. Nurhasanah, and A. Zulkarnain, “Implementation of MFCC and DTW methods for male and female voice type recognition,”*MIND Journal*, vol. 3, no. 1, pp. 49–63, 2018, doi: 10.26760/mindjournal.
[12] Y. Indrawaty N, Andriana, and D. Permatasari, “Speaker recognition to determine gender using MFCC and VQ methods,” *MIND Journal*, well 1, pp. 34–47, 2017, doi: 10.26760/mindjournal.
[13] Misbullah, M. Saifullah Sani, Husaini, L. Farsiah, Zahnur, and K. Martiwi Sukiakhy, “Indonesian speaker identification system using X-vector embedding,” *Journal of Information Technology and Computer Science*, vol. 11, no. 2, pp. 369–376, Aug. 2024, doi: 10.25126/jtiik.20241127866.
[14] M. Afif Ma’ruf, A. Aranta, and F. Bimantoro, “Student voice verification as an alternative attendance presence using MFCC feature extraction and LVQ classification.” [Online]. Available: http://jtika.if.unram.ac.id/index.php/JTIKA/
[15] N. Zheng, N. Wang, T. Lee, and P. C. Ching, “Speaker verification using complementary information from vocal source and vocal tract,” Springer, Berlin, Heidelberg, 2006, pp. 518–528. doi: 10.1007/11939993_54.
[16] M. M. Homayounpour and I. Rezaian, “Robust Speaker Verification Based on Multi Stage Vector Quantization of MFCC Parameters on Narrow Bandwidth Channels,” International Conference on Advanced Communication Technology, vol. 1, pp. 336–340, Apr. 2008, doi: 10.1109/ICACT.2008.4493773.
[17] K. Mohanaprasad, J. K. Pawani, V. Killa, and S. Sankarganesh, “Real Time Implementation of Speaker Verification System,” Indian journal of science and technology, vol. 8, no. 24, pp. 1–9, Sep. 2015, doi: 10.17485/IJST/2015/V8I24/80193.
[18] Liu, X., & Kinnunen, T. (2021). Learnable MFCCs for Speaker Verification. International Symposium on Circuits and Systems, 1–5. https://doi.org/10.1109/ISCAS51556.2021.9401593
[19] S. Sakai and K. Kameyama, “Text-independent Speaker Verification using Optimized Linear Combination of Local MFCC Features,” Jan. 2010, doi: 10.2316/P.2010.678-063.
[20] Al-Kaltakchi, M. T. S., Al-Nima, R. R. O., Alfathe, M., & Abdullah, M. A. M. (2020). Speaker Verification Using Cosine Distance Scoring with i-vector Approach. Computer Science and Software Engineering, 157–161. https://doi.org/10.1109/CSASE48920.2020.9142088