FOREX Publication
Open Access | Rapid and quality publishing

International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 4 | Pages 1502-1507 | e-ISSN: 2347-470X

A Robust Deep Learning-Based Speaker Identification System Using Hybrid Model on KUI Dataset

Subrat Kumar Nayak[1], Ajit Kumar Nayak[2], Suprava Ranjan Laha[3,6]*, Nrusingha Tripathy[4], and Takialddin AI Smadi[5]

[1]Siksha 'O' Anusandhan (deemed to be University), Bhubaneswar, India, email: subratsilicon28@gmail.com
[2]Siksha 'O' Anusandhan (deemed to be University), Bhubaneswar, India, email: ajitnayak@soa.ac.in
[3]Siksha 'O' Anusandhan (deemed to be University), Bhubaneswar, India, email: supravalaha@gmail.com
[4]Siksha 'O' Anusandhan (deemed to be University), Bhubaneswar, India, email: nrusinghatripathy654@gmail.com
[5]Faculty of Engineering, Jerash University, Jordan
[6]Brainware University, Barasat, Kolkata, India, email: supravalaha@gmail.com

*__Correspondence:__ *supravalaha@gmail.com; Tel.: +91-8217471921*

ABSTRACT- *Background:* Speaker identification, detecting human voices using speech characteristics and acoustics, is essential in security, biometrics, IoT, and human-computer interaction (HCI). As technology advances, more innovative software and robust hardware enhance these applications. This study evaluates feature extraction, pre-processing, and deep learning methods for speaker identification in natural settings. *Methods:* We compared deep learning algorithms, including Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and a proposed Hybrid model. Audio files were processed using different feature extraction and pre-processing techniques. *Results:* The proposed Hybrid model achieved the highest accuracy at 95%, surpassing other models. LSTM followed with an accuracy of 93%. Performance metrics, including accuracy, recall, and F1 score, were used to evaluate the models. *Conclusions:* The study demonstrates that the Hybrid model is the most effective for speaker identification in natural settings, highlighting its potential for improved human-computer interaction and security applications.

Keywords: Mel Frequency Cepstral Coefficients, CNN, LSTM, ANN, RNN, Speaker Identification.

ARTICLE INFORMATION
Author(s): Subrat Kumar Nayak, Ajit Kumar Nayak, Suprava Ranjan Laha, Nrusingha Tripathy, and Takialddin AI Smadi;
Received: 03/07/24; **Accepted:** 08/12/24; **Published:** 30/12/24
E- ISSN: 2347-470X
Paper Id: IJEER240412
Citation: 10.37391/IJEER.120446
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-12/ijeer-120446.html

Publisher's Note: FOREX Publication stays neutral with regard to Jurisdictional claims in Published maps and institutional affiliations.

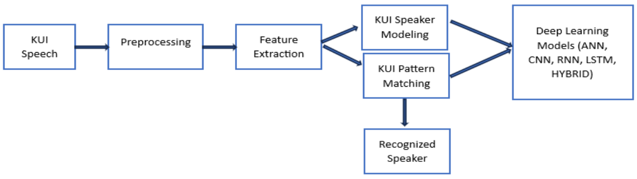
1. INTRODUCTION

A machine can take in speech or audio, process it, and identify the speaker using computing. While speaker identification has long been a study topic, the Internet of Things (IoT) and the rapid advancement of technology have increased the popularity of smart devices, voice assistants, and home assistants. As previously mentioned, voice is one of people's most fundamental forms of interaction. Therefore, voice technology facilitates the smoothest integration of human-to-machine communication [1].

Speaker identification is the ability to distinguish between human speech and the process of identifying or verifying an individual's identity via the use of their voiceprints and other auditory characteristics [2]. Speech recognition increases user accessibility by enabling more accessible communication with the system, whereas speaker identification verifies an individual's identity so the system knows who is speaking to it [3]. Speech recognition is language and corpus-dependent since it converts audio to text. However, to detect individual differences in speech patterns, speaker identification often needs to pay more attention to language and concentrate on unprocessed audio perceptions and the associated information.

The first stage in such apps is authentication. Thus, this research presents a deep learning-based speaker identification system. The training and testing stages in this type of network are crucial for precise outcomes. We must establish a quiet setting for each stage to ensure reliable findings. We can provide a clean atmosphere for training, but it is not feasible to maintain a noise-free setting for testing every time. Therefore, this research aims to develop a speaker identification system to produce the best results while dealing with external noise.

Diagrammatically, *fig.1* illustrates the KUI speaker identification system. Numerous studies have been conducted in several languages [4]. However, limited research has been conducted on language datasets with limited resources. Language is the most crucial tool for integrating indigenous people from their seclusion into society [5]. *Table 1* compares our techniques to pertinent historical investigations. The method we offer works better. Our approach has a higher bandwidth than earlier literature-described methods [6, 7].



. **Figure 1.** The architecture of KUI Speaker Identification

Table 1. Speaker Identification in different languages

Ref	Methods	Dataset	Language	Accuracy (%)
[8]	VQ	Private	English	90
[9]	CNN	Private	English	91
[10]	CNN	TIMIT	English	81
[10]	CNN	LibriSpeech	English	85
[10]	LSTM	TIMIT	English	72
[11]	CNN	LibriSpeech	English	76
[11]	CNN	LibriSpeech	English	81
[12]	CNN	SITW	English	83
[12]	CNN	TIMIT	English	86
[13]	CNN	Private	English	91
[13]	ANN	Private	English	80
[14]	CNN	TIMIT	English	94

2. MATERIALS & METHODS

Many neural network approaches are being used for speaker identification. Nevertheless, there is no research on speaker identification in a KUI language. Four distinct neural network models, Artificial Neural Network (ANN), Convolutional Neural Network (CNN), Long short-term Memory (LSTM), and Hybrid model, have been compared throughout this research [15].

2.1 Artificial Neural Network (ANN)

Artificial neural networks can handle a non-linear model with more competence. It is far more effective in developing data models [16]. Regression and classification issues are handled by artificial neural networks (ANN). This is one of the most basic mathematical models for advancing data analysis technology [17]. The flattened layer in this model has no learnable parameters. A one-dimensional array is created from the input sequence [18, 19]. The different layers of this model are stated in *Eq. (1) to Eq. (4).*

$$Flatten(X) = Flatten(X_{train}) \tag{1}$$

After the flattened layer,the first dense layer is present.

$$Z^{(1)} = Flatten(X).W^{(1)} + b^{(1)}, A^{(1)} = RELU\left(Z^{(1)}\right) \tag{2}$$

The result of the first linear transformation is used element by element to the rectified linear unit activation function while $W^{(1)}$ and $b^{(1)}$denote the weight matrix and bias associated with the dense layer. The second dense layer is connected to the first dense layer.

$$Z^{(2)} = A^{(1)}.W^{(2)} + b^{(2)}, A^{(2)} = RELU\left(Z^{(2)}\right) \tag{3}$$

Here $W^{(2)}$ and $b^{(2)}$denote the weight matrix.

$$Z^{(3)} = A^{(2)}.W^{(3)} + b^{(3)}, A^{(3)} = Softmax\left(Z^{(3)}\right) \tag{4}$$

Here $W^{(3)}$ and $b^{(3)}$denote the weight matrix and bias associated with the output dense layer, and the output layer activation process uses Softmax to turn the raw output scores into probabilities.

2.2 Convolutional Neural Network (CNN)

Convolutional Neural Networks are deep learning systems that can receive inputs and give importance to each item [20, 21]. Multiplexing two functions produces a third function called convolution that reveals how the second function modified the first function [22, 23]. CNN convolution is this math. Convolutional methods are power-hungry, bandwidth-intensive, and sluggish. *Equations (5–11)* describe this model's layers.

The input layer is first linked to the first convolutional layer, as shown below.

$$Z^{(1)} = Conv1D(X), A^{(1)} = RELU\left(Z^{(1)}\right) \tag{5}$$

Here, X is the input sequence. The first Convolutional layer, Conv1D, has a RELU activation function and 64 filters with a kernel size of 3. The max-pooling layer was linked to the first Convolutional layer.

$$A^{(2)} = MaxPooling1D\left(A^{(1)}\right) \tag{6}$$

The Maxpooling1D layer decreases the first Convolutional layer's output's spatial dimensions. The max-pooling layer is then connected to the second Convolutional layer with 128 filters.

$$Z^{(3)} = Conv1D\left(A^{(2)}\right), A^{(3)} = RELU\left(Z^{(3)}\right) \tag{7}$$

The first dense layer is connected to the global max pooling layer, which connects to the second Convolutional layer.

$$A^{(4)} = GlobalMaxPooling1D\left(A^{(3)}\right), \quad Z^{(5)} = A^{(4)}.W^{(5)} + b^{(5)}, A^{(5)} = RELU\left(Z^{(5)}\right) \tag{8}$$

Here, the weight matrix and bias for the first dense layer are represented as $W^{(5)}$ and $b^{(5)}$the dropout layer joins the first and second dense layers.

$$A^{(6)} = Dropout\left(A^{(5)}\right), Z^{(7)} = A^{(6)}.W^{(7)} + b^{(7)}, A^{(7)} = RELU\left(Z^{(7)}\right) \tag{9}$$

Similarly, another dropout layer is connected between the second and third dense layers.

$$A^{(8)} = Dropout\left(A^{(7)}\right), Z^{(9)} = A^{(8)}.W^{(9)} + b^{(9)}, A^{(9)} = RELU\left(Z^{(9)}\right) \tag{10}$$

The third dense layer consists of 32 units connecting to the final dense output layer.

$$Z^{(10)} = A^{(9)}.W^{(10)} + b^{(10)}, A^{(10)} = Softmax\left(Z^{(10)}\right) \tag{11}$$

The weight matrix and bias for the output dense layer are represented as $W^{(10)}$and$b^{(10)}$, and Softmax is used for the output layer activation to convert the raw output into probabilities.

2.3 Long short-term Memory (LSTM)

The LSTM Recurrent Neural Network (RNN) manages long-term reliance. Vanishing gradients can be fixed. An LSTM has three gates: forget, input, and output [24]. One LSTM hidden state. *Equations (12–14)* describe this model's layers [25, 26].

According to this approach, the input layer is connected to the first LSTM layer and the second LSTM layer to the first dense layer.

$$H^{(1)} = LSTM(X), H^{(2)} = LSTM\left(H^{(1)}\right), Z^{(3)} = H^{(2)}.W^{(3)} + b^{(3)}, A^{(3)} = RELU\left(Z^{(3)}\right) \tag{12}$$

Here the first LSTM layer consists of 256 units whereas second LSTM layer consist of 128 units. For the first dense layer $W^{(3)}$ and $b^{(3)}$ are the weight matrix and bias. Two dropout layers are used here in between the three dense layers as follows.

$$A^{(4)} = Dropout\left(A^{(3)}\right), Z^{(5)} = A^{(4)}.W^{(5)} + b^{(5)}, A^{(5)} = RELU\left(Z^{(5)}\right), A^{(6)} = Dropout\left(A^{(5)}\right) \tag{13}$$

A 32-unit connection exists between the third dense layer and the output dense layer.

$$Z^{(7)} = A^{(6)}.W^{(7)} + b^{(7)}, A^{(7)} = RELU\left(Z^{(7)}\right), Z^{(8)} = A^{(7)}.W^{(8)} + b^{(8)}, A^{(8)} = Softmax\left(Z^{(8)}\right) \tag{14}$$

Here $W^{(8)}$ and $b^{(8)}$ are weight matrix and bias for the output dense layer.

2.4 Hybrid Model (CNN+LSTM)

Equations (15) through *(18)* specify the layers of our proposed hybrid model, which combines CNN and LSTM. We initially take an LSTM layer.

$$H^{(1)} = LSTM(X) \tag{15}$$

In this instance, X represents the input sequence, the 256 units of the first LSTM layer, and the 128 units of the second LSTM layer, initiating the LSTM operation. In between two LSTM layers, one convolutional layer and Maxpooling layer are present.

$$Z^{(2)} = Conv1D\left(H^{(1)}\right), A^{(2)} = RELU\left(Z^{(2)}\right), A^{(3)} = MaxPooling1D\left(A^{(2)}\right), H^{(4)} = LSTM\left(A^{(3)}\right) \tag{16}$$

In between the second LSTM layer, one Flatten layer, three dense layers, and two dropout layers are present. By randomly changing a portion of the input units to 0 at each update during training, the dropout layer helps prevent overfitting.

$$Flatten\left(H^{(4)}\right), Z^{(6)} = Flatten\left(H^{(4)}\right).W^{(6)} + b^{(6)}, A^{(6)} = RELU\left(Z^{(6)}\right), A^{(7)} = Dropout\left(A^{(6)}\right), Z^{(8)} = A^{(7)}.W^{(8)} + b^{(8)}, A^{(8)} = RELU\left(Z^{(8)}\right), A^{(9)} = Dropout\left(A^{(8)}\right), Z^{(10)} = A^{(9)}.W^{(10)} + b^{(10)}, A^{(10)} = RELU\left(Z^{(10)}\right) \tag{17}$$

The three dense layers each have a RELU activation function and 128, 64, and 32 units, respectively. The third dense layer used Softmax to link to the final output dense layer.

$$Z^{(11)} = A^{(10)}.W^{(11)} + b^{(11)}, A^{(11)} = Softmax\left(Z^{(11)}\right) \tag{18}$$

Here, $W^{(11)}$ and $b^{(11)}$ stand in for the weight matrix and bias for the output layer.

3. EXPERIMENTAL ANALYSIS

The KUI speech dataset, developed in the KUI language, was used in our investigations. We have designed a platform for the dataset's development, shown in *fig. 2*. The platform consists of an admin and user parts [27]. It is recorded in a studio or dark room to reduce noise. For this experiment we have taken 10 speakers having 500 sentences each. The dataset collection is an ongoing process. Initially we have collected 5000 audio data using this platform.

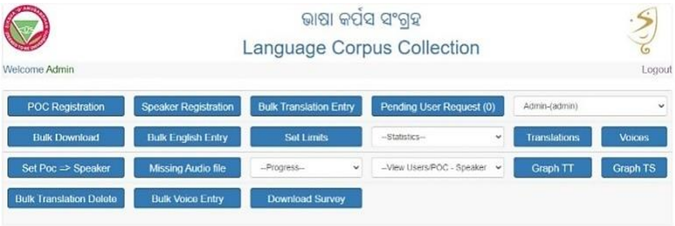


Figure 2. The platform for data collection in KUI language

3.1. Data Preprocessing

A 16 KHz recording rate is used for each audio data. Lower frequencies in audio signals change more than higher ones. Therefore, a part of the stream might reveal numerous aspects. The Mel-frequency crystal co-efficient (MFCC) represents audio data most accurately of all feature extraction methods. A cosine conversion of the spectrum's natural logarithm produces MFCC [28]. Thus, MFCC feature extraction is employed with neural networks. Delete any audio files without helpful information. The MFCC architecture is shown in *fig. 3*.

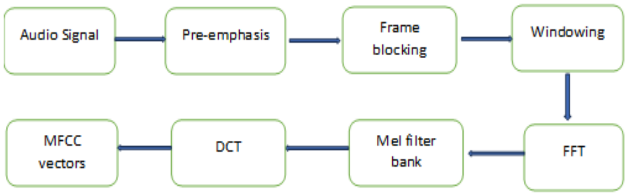


Figure 3. Architecture of Mel-frequency crystal co-efficient (MFCC)

3.2 Experimental Setup

As previously mentioned, we test different deep-learning models on our KUI dataset. Three distinct percentages are used for training and testing: 70:30, 80:20, and 90:10. Every experiment uses a CPU, a Core i5 processor, 16 GB of RAM, and Python 3.9. The Adam optimizer, whose learning rate is 0.01, and the criterion loss = "categorical_crossentropy" was the foundation for the loss objective function. The experiment used five epoch sizes: 20, 50, 100, 200, and 500. Without any additional data, every model is trained using the same dataset. We contrast our suggested approach with the other four approaches. We use a 16,000-sample rate and 2D normalization. RELU is used as the activation function. From the experiment, we found that the KUI dataset is the best fit for our proposed model.

4. RESULT ANALYSIS

We train the models with varying settings to achieve optimal performance for speaker identification on the KUI dataset. We start our experiment with an epoch size of 20, having a learning rate of 0.01. We assume a 70-30 training-to-testing ratio in this period size. The training accuracy of ANN is 65, while the testing accuracy climbs vertically for the first 10 epochs, then gradually increases to 51. The testing accuracy of CNN and LSTM is 73 and 88, respectively. Our proposed model gives the highest accuracy of 90. *Figure 4* displays a graphic representation of accuracies up to 500 epochs having a split ratio of 70-30. *Figure 5* displays a graphical representation of accuracy up to 500 epochs with a split ratio of 80-20. CNN provides excellent accuracy as well, almost 91. However, our suggested model yields an accuracy of 95. Similarly, we use 80-20 and 90-10 as the split ratios for the 500 epoch. Our suggested model is more accurate than the other models in the epoch size 500. CNN provides a superior 80-20 split ratio outcome for epoch size 200 than the other two ratios. In *table 2*, details of the testing accuracy are provided. Additionally, our suggested model provides greater accuracy in 100 epochs than the other four models. The accuracy is shown as a graphical representation in *fig. 6*. Except for ANN, every model shows parallel accuracy at an epoch size 200. In the 90-10 split ratio, the CNN model performs better than the other two split ratios. Furthermore, our proposed model produces better accuracy in 200 epochs. In the end, for an era size of 500, our proposed model has the highest accuracy. Below are the findings from the various methods. Several metrics are used to calculate these findings.

Table 2. Accuracy of all the Models

No of Epoch	Epoch Size	ANN	CNN	RNN	LSTM	HYBRID (CNN+ LSTM)
200	70-30	80	88	81	89	91
	80-20	82	88	83	91	92
	90-10	80	89	81	91	92
500	70-30	83	91	85	90	95
	80-20	82	88	84	89	93
	90-10	78	92	88	93	94

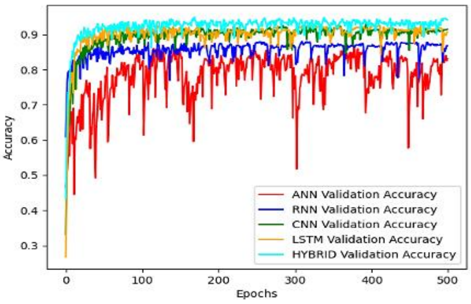


Figure 4. Model accuracy with split ratio of 70-30 and epoch size of 500

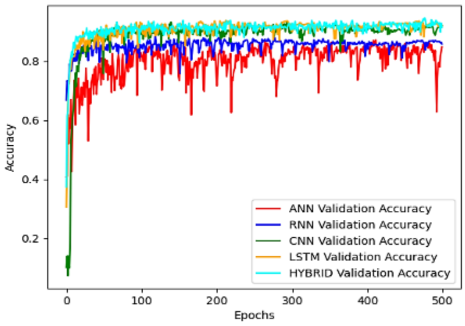


Figure 5. Model accuracy with split ratio of 80-20 and epoch size of 500

When we use the epoch size of 200 and the split ratio of 70–30 our suggested model has the least amount of loss. The loss for the CNN model steadily dropped. Our suggested model likewise has a lower loss value when the epoch size is increased. We infer from our experiment that our suggested model has the most minor loss compared to the other four models.

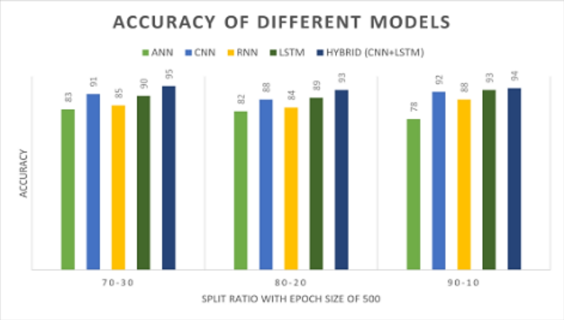


Figure 6. Accuracy of different models

Precision, recall, and F1-score are the classification metrics evaluated to ascertain accuracy. Every statistic has a different split ratio and occurs in a separate era size. The various categorization metrics with split ratios of 70-30, 80–20, 90-10, and epoch sizes of 200, and 500 are displayed in *table 3*. According to the analysis results, our suggested model outperforms the other three regarding accuracy and other metrics. In *figure 7-8*, our proposed model's confusion matrices are drawn with the epoch size of 500 with a split ratio of 70-30 and 80-20.

Table 3. Performance of all the Models

Epochs	Parameters	ANN			CNN			LSTM			HYBRID Model		
		70-30	80-20	90-10	70-30	80-20	90-10	70-30	80-20	90-10	70-30	80-20	90-10
200	**Precision**	79	79	82	89	85	88	88	89	91	93	95	95
	Recall	80	83	83	88	88	91	92	95	89	92	87	89
	F1-Score	79	81	82	88	87	90	90	92	90	92	91	92
500	**Precision**	82	80	78	93	92	88	88	91	90	91	93	94
	Recall	83	82	81	91	90	87	90	90	93	93	95	92
	F1-Score	82	81	79	92	91	88	89	91	92	92	94	93

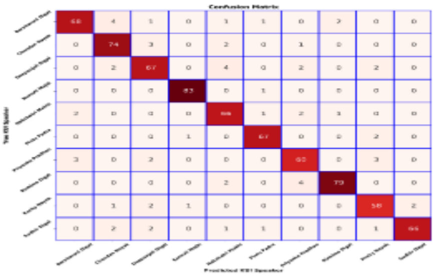


Figure 7. Confusion matrix for HYBRID model with epoch size 500 and split ratio 70-30

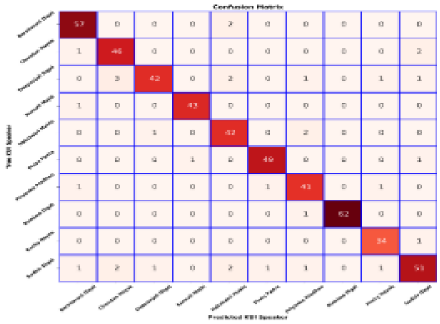


Figure 8. Confusion matrix for HYBRID model with epoch size 500 and split ratio 80-20

5. CONCLUSION & FUTUREWORK

The Speaker identification in a self-created KUI dataset is the primary goal of this work. We tried to enhance the performance by utilizing various epochs and learning rates. We have utilized MFCC to train the models. According to the experimental findings, speaker identification accuracy in this study was 97% during training and 95% during testing. This study contributes to the hybrid CNN, ANN and LSTM studies.

As our dataset is low-resourced and tribal language, 95% accuracy is still out of the mark. Additionally, we examined our findings for precision, recall, F1 score, and confusion matrix with the proposed model. This approach still has much scope for improvement:

1. The architecture for extracting additional features from KUI speech samples may be more profound and less computationally intensive.
2. Combining this method with additional feature extraction techniques may improve the model's performance. We intend to investigate the potential applications of our training technique in the future, such as gender recognition with our dataset.
3. We plan to experiment with slot filling and dialog act recognition.
4. We will research ways to make our models more resilient so that they may be applied in real-world situations.

Speaking recognition systems will grow increasingly precise, safe, and indispensable to our everyday lives as artificial intelligence, machine learning, and signal processing progress. We may further develop a model using this dataset by substituting a Transformer model for the LSTM.

6. ACKNOWLEDGMENTS

I would like to express my sincere gratitude to everyone who supported and guided me throughout this project.

REFERENCES

[1] Hansen, J. H., & Hasan, T. (2015). Speaker recognition by machines and humans: A tutorial review. *IEEE Signal processing magazine*, *32*(6), 74-99.

[2] Nassif, A. B., Shahin, I., Hamsa, S., Nemmour, N., & Hirose, K. (2021). CASA-based speaker identification using cascaded GMM-CNN classifier in noisy and emotional talking conditions. *Applied Soft Computing*, *103*, 107141.

[3] Simić, N., Suzić, S., Nosek, T., Vujović, M., Perić, Z., Savić, M., &Delić, V. (2022). Speaker recognition using constrained convolutional neural networks in emotional speech. *Entropy*, *24*(3), 414.

[4] Meftah, A. H., Mathkour, H., Kerrache, S., & Alotaibi, Y. A. (2020). Speaker identification in different emotional states in Arabic and English. *IEEE Access*, *8*, 60070-60083.

[5] Nayak, S. K., Nayak, A. K., Mishra, S., & Mohanty, P. (2023). Deep learning approaches for speech command recognition in a low resource KUI language. *International Journal of Intelligent Systems and Applications in Engineering*, *11*(2), 377-386.

[6] Bimbot, F., Bonastre, J., Fredouille, C. et al. A Tutorial on Text-Independent Speaker Verification. EURASIP J. Adv. Signal Process. 2004, 101962 (2004). https://doi.org/10.1155/S1110865704310024

[7] Sztah´o, D´avid, Gy¨orgySzasz´ak, and Andr´as Beke.” Deep learning methodsin speaker recognition: a review.” arXiv preprint arXiv:1911.06615(2019).

[8] Tripathi, S., & Bhatnagar, S. (2012, November). Speaker recognition. In *2012 Third International Conference on Computer and Communication Technology* (pp. 283-287). IEEE.

[9] Wang, M., Sirlapu, T., Kwasniewska, A., Szankin, M., Bartscherer, M., & Nicolas, R. (2018, July). Speaker recognition using convolutional neural network with minimal training data for smart home solutions. In *2018 11th International Conference on Human System Interaction (HSI)* (pp. 139-145). IEEE.

[10] Prachi, N. N., Nahiyan, F. M., Habibullah, M., & Khan, R. (2022, February). Deep Learning Based Speaker Recognition System with CNN and LSTM Techniques. In *2022 Interdisciplinary Research in Technology and Management (IRTM)* (pp. 1-6). IEEE.

[11] Pentapati, H. K., & Sridevi, K. (2022). Dilated Convolution and MelSpectrum for Speaker Identification using Simple Deep Network. In *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)* (Vol. 1, pp. 1169-1173). IEEE.

[12] Chowdhury, A., & Ross, A. (2017, October). Extracting sub-glottal and supra-glottal features from MFCC using convolutional neural networks for speaker identification in degraded audio signals. In *2017 IEEE International Joint Conference on Biometrics (IJCB)* (pp. 608-617). IEEE.

[13] Gade, V. S. R., & Sumathi, M. (2023, May). Hybrid Deep Convolutional Neural Network based Speaker Recognition for Noisy Speech Environments. In *2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC)* (pp. 920-926). IEEE.

[14] Nainan, S., & Kulkarni, V. (2021). Enhancement in speaker recognition for optimized speech features using GMM, SVM and 1-D CNN. *International Journal of Speech Technology*, *24*, 809-822.

[15] Shahin, I., Nassif, A. B., & Hindawi, N. (2021). Speaker identification in stressful talking environments based on convolutional neural network. *International Journal of Speech Technology*, *24*, 1055-1066.

[16] Kabir, M. M., Mridha, M. F., Shin, J., Jahan, I., & Ohi, A. Q. (2021). A survey of speaker recognition: Fundamental theories, recognition methods and opportunities. *IEEE Access*, *9*, 79236-79263.

[17] Abbood, Z. A., Yasen, B. T., Ahmed, M. R., & Duru, A. D. (2022). Speaker identification model based on deep neural networks. *Iraqi Journal for Computer Science and Mathematics*, *3*(1), 108-114.

[18] Tripathy, N., Hota, S., Mishra, D., Satapathy, P., & Nayak, S. K. (2024). Empirical Forecasting Analysis of Bitcoin Prices: A Comparison of Machine learning, Deep learning, and Ensemble learning Models. *International journal of electrical and computer engineering systems*, *15*(1), 21-29.

[19] Bai, Z., & Zhang, X. L. (2021). Speaker recognition based on deep learning: An overview. *Neural Networks*, *140*, 65-99.

[20] Hourri, S., Nikolov, N. S., &Kharroubi, J. (2021). Convolutional neural network vectors for speaker recognition. *International Journal of Speech Technology*, *24*(2), 389-400.

[21] Lukic, Y., Vogt, C., Dürr, O., & Stadelmann, T. (2016, September). Speaker identification and clustering using convolutional neural networks. In 2016 IEEE 26th international workshop on machine learning for signal processing (MLSP) (pp. 1-6). IEEE.

[22] Costantini, G., Cesarini, V., & Brenna, E. (2023). High-Level CNN and Machine Learning Methods for Speaker Recognition. *Sensors*, *23*(7), 3461.

[23] Tomar, S., &Koolagudi, S. G. (2023, April). CNN-MFCC Model for Speaker Recognition using Emotive Speech. In *2023 IEEE 8th International Conference for Convergence in Technology (I2CT)* (pp. 1-7). IEEE.

[24] El-Moneim, S. A., Nassar, M. A., Dessouky, M. I., Ismail, N. A., El-Fishawy, A. S., & Abd El-Samie, F. E. (2020). Text-independent speaker recognition using LSTM-RNN and speech enhancement. *Multimedia Tools and Applications*, *79*, 24013-24028.

[25] Dua, M., Jain, C., & Kumar, S. (2022). LSTM and CNN based ensemble approach for spoof detection task in automatic speaker verification systems. *Journal of Ambient Intelligence and Humanized Computing*, *13*(4), 1985-2000.

[26] Prachi, N. N., Nahiyan, F. M., Habibullah, M., & Khan, R. (2022, February). Deep Learning Based Speaker Recognition System with CNN and LSTM Techniques. In *2022 Interdisciplinary Research in Technology and Management (IRTM)* (pp. 1-6). IEEE.

[27] Nayak, S. K., Nayak, A. K., Mishra, S., Mohanty, P., Tripathy, N., Pati, A., & Panigrahi, A. (2024). Original Research Article Speech data collection system for KUI, a Low resourced tribal. *Journal of Autonomous Intelligence*, *7*(1).

[28] Prabakaran, D., &Sriuppili, S. (2021). Speech processing: MFCC based feature extraction techniques-an investigation. In *Journal of Physics: Conference Series* (Vol. 1717, No. 1, p. 012009). IOP Publishing