JOURNAL OF EXPERIMENTAL & THEORETICAL ARTIFICIAL INTELLIGENCE
2024, VOL. 36, NO. 2, 273–285
https://doi.org/10.1080/0952813X.2022.2092560

ORIGINAL ARTICLE

Modified layer deep convolution neural network for text-independent speaker recognition

V Karthikeyan[a] and Suja Priyadharsini S[b]

[a]Department of Electronics and Communication Engineering, Kalasalingam Institute of Technology, Krishnankoil, Tamilnadu, India; [b]Department of Electronics and Communication Engineering, Anna University Regional Campus-Tirunelveli, Tirunelveli, Tamilnadu, India

ABSTRACT

Speaker recognition is the task of identifying the spokesman automatically using speaker-specific features. It has been a popular and most involved topic in the field of speech technology. This field opens a wide opportunity for research and finds its application in the areas such as forensics, authentication, security, etc. In this work, a modified deep-convolutional neural network structure has been proposed for speaker identification that has improved convolution, activation, and pooling layers along with Adam's optimiser. The proposed architecture yielded the increase of prediction accuracy and reduction of Loss function when compared to the generic Convolutional Neural Network scheme. The execution of the proposed architecture is validated by various datasets and the outcomes show that the modified CNN performs better than the other state-of-the-art models regarding both accuracy (avg 99%) and loss function (avg 1%). From the analysis, it is found that the Modified-CNN suits the best for real-time speaker identification applications as the efficacy of the model does not degrade due to the effects of noise and interferences that are caused in the recording environment. Relevance of the work: Speaker Recognition is an area of interest in which ML and DL schemes, when combined, have the potential to make history in the areas of Automation and Authentication. Using a modified CNN can enhance the process by ignoring many issues such as false positives, background noise, and so on. This process can be expanded to create a Raga Identification and Therapy mechanism that can be used to treat diseases.

ARTICLE HISTORY
Received 23 August 2020
Accepted 16 June 2022

KEYWORDS
Speaker identification; deep learning; CNN; spectrogram; MFCC

Introduction

The method of identifying the context's speaker is known as speaker recognition. In the disciplines of forensics, authentication, and security, it is the most popular idea (Billeb et al., 2015; Jain et al., 2004). Every human being has the ability to communicate and know who is speaking in a natural manner. However, when it is used artificially, it is referred to as Artificial Intelligence (AI). Many researchers in the field of Artificial Intelligence are now interested in voice and speech-related topics. Speaker Recognition (SR) is divided into two categories: Speaker Identification (SI) and Speaker Verification (SV) based on the application. Speaker identification is the process of identifying a speaker from a closed dataset only based on his voice sample, whereas Speaker Verification is the act of confirming that this voice belongs to this specific speaker (Karthikeyan & Suja Priyadharsini, 2022). The SI and SV are further divided into text-dependent and text-independent approaches based on text-

CONTACT V Karthikeyan velkarthi85@gmail.com Department of Electronics and Communication Engineering, Kalasalingam Institute of Technology, Krishnankoil, Tamilnadu, India

dependency (Prabhakar et al., 2003). The text-dependent approach is completely reliant on a closed text, but the text-independent method is not based on a particular text. Artificial Intelligence, Machine Learning, and Deep Learning, among other technologies, are assisting in the automation of the speaker identification process. Speaker identification can be done using either a model-based or a feature-based approach when using these technologies. The focus of the feature-based method is on training predefined models based on the best aspects of the speech signal, such as MFCC, Spectrogram, etc., (Ai et al., 2012; Kabal & Ramachandran, 1987; Yujin et al., 2010) and the model-based method focuses on creating a parametric stochastic model or a non-parametric template model (Kinnunen & Li, 2010; Martinez et al., 2012; Singh & Rajan, 2011). Many predefined neural network architectures are utilised to train the models in the Deep Learning technique. The deep-convolutional neural network (CNN) is the best model one among them since it outperforms other models. Using pre-set networks, researchers have made numerous advances to the field of voiceprint recognition.

Before a few years, speaker recognition systems used spectral properties like mel frequency cepstral coefficients (MFCC) to embed speakers (Ravanelli et al., 2018; Tiwari, 2010), but modules using i–vectors have rendered these methods obsolete (Dehak et al., 2011; Garcia-Romero & Espy-Wilson, 2011; Kenny, 2010). The i–vector takes the speaker's speech and uses it to adapt a GMM-UBM (Gaussian Mixture Model-Universal Background Model) model that is not dependent on the speaker. The dimensionality of the super vector created by concatenation of the means of the fitted GMM is lowered by using joint factor analysis. End-to-end SR (Sadjadi et al., 2016; Variani et al., 2014; Wan et al., 2018; Xie et al., 2019) systems have recently been developed, which outperform i–vector-based systems and clearly demonstrate state-of-the-art performance. The proposed work uses end-to-end SR in which the layers of deep CNN layers are modified to enhance the identification rate with minimum loss function.

The following is how the rest of the work is laid out: The relevant literature is thoroughly explained in Section 1.2. The proposed model for speaker recognition is depicted in Section 2. Section 3 describes the experimental setup, and the responses of a series of tests are presented in Section 4. Conclusions and areas for future research are given in Section 5.

Related works

Variani et al., (2014) introduced the neural network-based technique for SR in the initial phase. In their work, they used a maxout fully connected network to generate d-vectors, with cosine similarity as the ultimate conclusion. Progressive network designs like CNNs (Kenny, 2010; Sadjadi et al., 2016; Xie et al., 2019) and RNNs (Hajibabaei & Dai, 2018; Li et al., 2017; Wan et al., 2018) are utilised to extract features using d-vector strategy. Many advanced training objectives (Hajibabaei & Dai, 2018; Kenny, 2010; Wan et al., 2018) and temporal aggregation strategies (Cai et al., 2018; Li et al., 2017; Xie et al., 2019) are used in addition to these tactics in order to improve advanced training objectives. The Deep Neural Networks (DNNs) were used in conjunction with the i–vectors to enumerate Baum-Welch statistics (Kenny et al., 2014). This combination can be utilised to extract low- and high-level features well (Yaman et al., 2012). Recent studies show that DNNs are being applied in the field of SI (Heigold et al., 2016; Salehghaffari, 2018; Snyder et al., 2018, 2016). Standard features exclude narrow band attributes such as pitch and formants, but modified features like filter-bank and MFCC (Richardson et al., 2015a); (Salehghaffari, 2018; Snyder et al., 2017), which are based on perceptual data, do not ensure optimal speech-basedtasks.

It is preferable to feed the networks directly with spectrogram-bins (Bhattacharya et al., 2017; Nagrani et al., 2017; Zhang et al., 2018) or the raw voice waveform itself to overcome these limitations. CNN is the best approach for processing original raw audio samples because elements such as weight sharing, use of local filters, and pooling assist the CNN in discovering robust and invariant embodiments (Mehri et al., 2017; Muckenhirn et al., 2018; Palaz et al., 2015). To process speech with CNNs, many recent studies have used low-level speech representations. Magnitude spectrogram characteristics have been used in a number of earlier publications (Bhattacharya et al., 2017; Zhang et al., 2018). Spectrograms provide far more information than hand-crafted features. However, certain crucial hyper-parameters, such as overlap, duration, frame window typology, and the number of frequency bins, require careful tuning. As a result, the current tendency is to skip feature extraction entirely and train the model straight from raw waveforms. Speech recognition (Palaz et al., 2015), speech synthesis (Mehri et al., 2017), speaker recognition (Karthikeyan & Suja Priyadharsini, 2021; Muckenhirn et al., 2018; Ye & Yang, 2021), spoofing detection (Dinkel et al., 2017), and emotion recognition (Trigeorgis et al., 2016) have all benefited from this technology. The convolutional neural network (CNN) is a task-oriented learning depth neural network with many layers and shared weights. CNN offers several advantages over shallow networks, including the ability to characterise complex functions and solve highly abstract AI tasks with a higher level of complexity (Dinkel et al., 2017; Hu et al., 2017, Richardson et al., 2015b; Trigeorgis et al., 2016). Many challenges of categorisation and recognition including handwritten digit, finger vein, action, object, and traffic sign, can benefit from CNN. Both feature extraction and classification audits can be performed at the same time in CNN (Mehri et al., 2017; Muckenhirn et al., 2018; Nagrani et al., 2017; Palaz et al., 2015).

Despite the fact that present approaches have many advantages, a strategy for boosting accuracy and minimising the loss function is still required. As a result, this paper offers a modified layered CNN model with expanded convolutional, pooling, and activation layers, as well as Adam's optimiser, for efficient speaker detection.

The major contribution in this manuscript is listed as below

A multi-layered deep learning methodology is recommended for the imbalanced speaker dataset to recognise the particular Speaker.

- The constructed deep-CNN network contains pre-trained layers at the lower-level to extract the speech feature (Mel's Spectrogram) itself.
- Upper-layer comprises 1-D convolution, pooling, dropout, dense layers by batch normalisation to improve computation efficacy.
- The shallow layers in the modified-CNN offered more granular local attributes that could identify various speakers in the same class, but the deep layers could represent more high-level semantic information that was utilised to classify the speech signals into multiple classes.
- Deep Convolutional Network with a discriminator is applied for minority data augmentation.

To demonstrate improvement in pure voice recognition capabilities, the proposed model is tested using well-known datasets (ELSDSR, TIMIT, 16000PCM) and a recorded experimental dataset.

Materials and methods

In this paper, a unique deep neural learning strategy is used for speaker detection in the dataset, as illustrated in Figure 1. The datasets employed for validation in the proposed work are from a variety of fields, and the features are determined by the network itself. In addition, the layered architecture of a traditional CNN is changed in order to solve the problem of class imbalance and improve the model's overall performance.

Supaporn (Bunrit et al., 2019) proposed a 15-layer traditional convolutional neural network model. In the newly proposed deep-CNN model incorporated the pre-trained layers (First 2 blocks) at the lowest level of hierarchical layers. Convolutional layers, pooling layers, and fully connected layers are all included in the modified CNN architecture. To obtain the bottle-neck attributes, only the first two layers of the pre-trained network are used from the original architecture, rather than all of

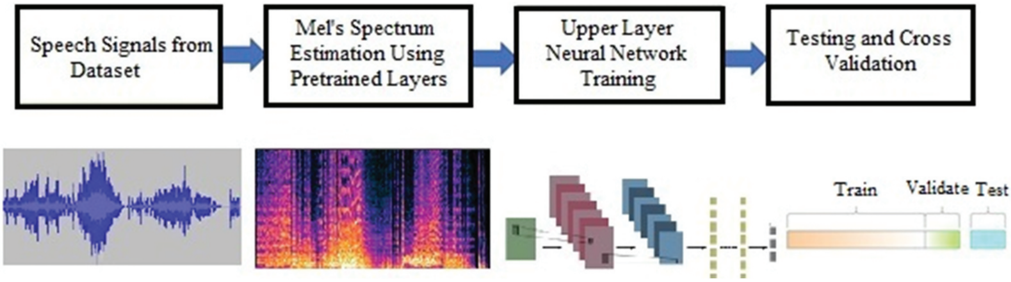
Speech Signals from Dataset
Mel's Spectrum Estimation Using Pretrained Layers
Upper Layer Neural Network Training
Testing and Cross Validation
Train
Validate
Test

Figure 1. Proposed model-block diagram.

the layers of the pre-trained network. Subsequently, all of the layers of the traditional CNN network are changed with additional layers that were chosen to increase the classification model's overall performance. A batch normalising (BN) layer has been connected on top of 1-D convolution, Global-average & Max Pooling, dropout, and dense layers.

Modified convolutional neural network

Convolutions, non-linearities, pooling, and classification are the four fundamental operations that make up a CNN. The models can additionally contain batch normalisation and dropout. These processes are frequently layered together so that convolutions are followed by a non-linearity such as a ReLU, which is then repeated a few times before being combined with a pooling operation. The original input is subsampled by pooling to a size that is manageable when the network is deep enough, and it is passed through to the classification section of the network. Batch normalising is usually applied after the convolution but before the non-linearity. Figure 2 shows the layers structure of the modified-CNN network.

The BN layer stabilises the voiceprint values to reduce the efficacious covariant shift, whereas not performing BN will result in biased outputs. Furthermore, a convolutional-1D layer is used, which is built on the idea that if the input and target datasets are distinct, features from the target-specific domain should be extracted at an upper layer to boost the model's gross response. We have used the speaker dataset as the source domain and a specific speaker as the target domain. So, using the same principle, features particular to the individual speaker is obtained from lower layers relevant to the audio dataset, and characteristics specific to the individual speaker is extracted from higher layers using the convolution layer. Figure 3 represents the various attribute values attained at the output of the CONV-1D layer.

The Discrete Fourier Transform (DFT) is applied to the audio signal after it has been framed and windowed to the frequency range. The output signal is then shifted to Mel's Frequency. The logarithm is then used to calculate the distorted output. The signal is then transformed into Mel Cepstrum using IDFT (Inverse Discrete Fourier Transform; Bunrit et al., 2019). Following that, 1D layers with max and global avg-pooling are utilised to make the network highly powerful to spatial transitions in the voiceprints. Furthermore, the dropout layer is attached to increase the regularisation also minimise the overfitting issue in the modified layered CNN model, and, lastly the dense layer is added to the architecture to make the model precise to resolving the multi-class problem by adjusting the value of the parameter to the number of speakers in the dense network, along with the Relu activation and cross-entropy loss. In comparison to models that use max-pooling, replacing max-pooling layers with convolutions with a greater stride to lower the input size can produce improved results. Batch normalisation layers were utilised in the CNN and discriminator network

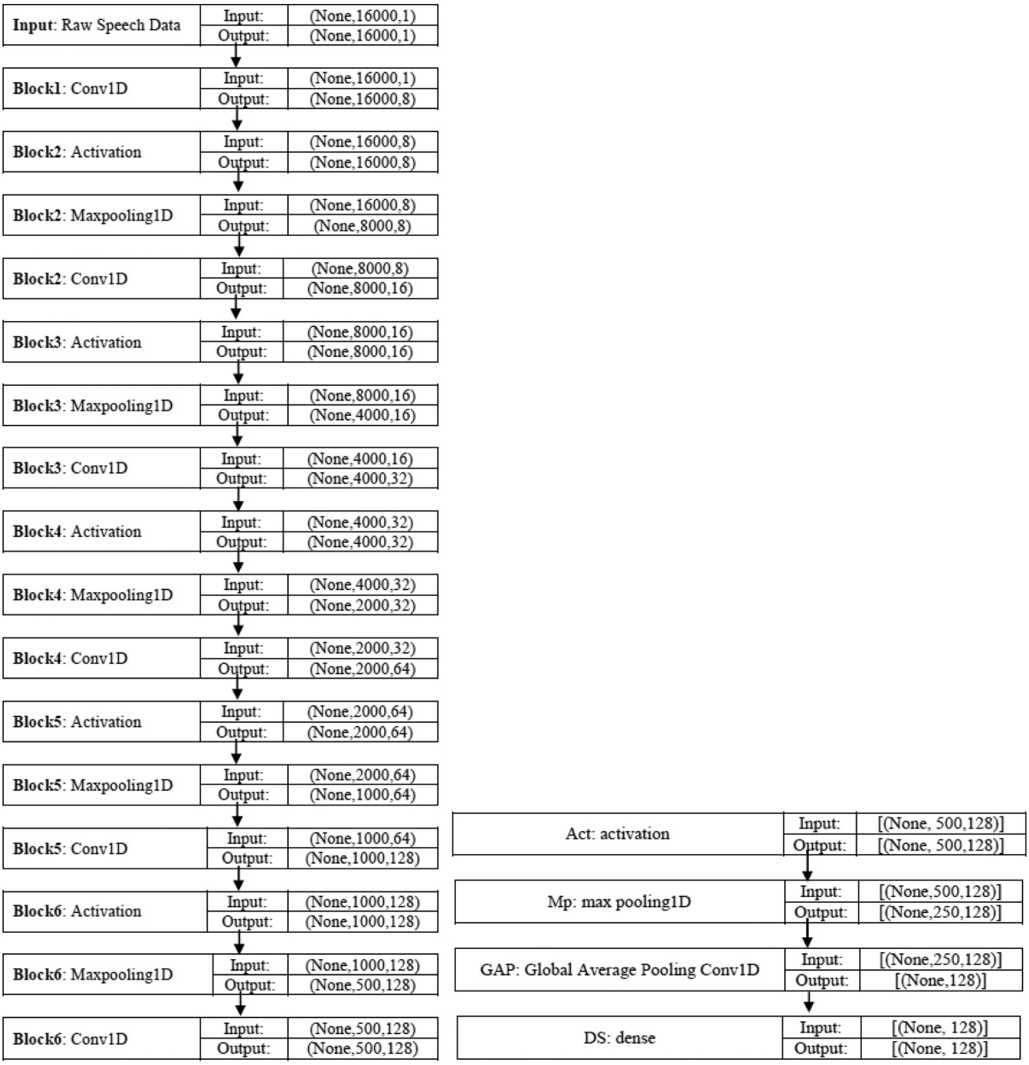
Input: Raw Speech Data | Input: (None,16000,1) | Output: (None,16000,1)
Block1: Conv1D | Input: (None,16000,1) | Output: (None,16000,8)
Block2: Activation | Input: (None,16000,8) | Output: (None,16000,8)
Block2: Maxpooling1D | Input: (None,16000,8) | Output: (None,8000,8)
Block2: Conv1D | Input: (None,8000,8) | Output: (None,8000,16)
Block3: Activation | Input: (None,8000,16) | Output: (None,8000,16)
Block3: Maxpooling1D | Input: (None,8000,16) | Output: (None,4000,16)
Block3: Conv1D | Input: (None,4000,16) | Output: (None,4000,32)
Block4: Activation | Input: (None,4000,32) | Output: (None,4000,32)
Block4: Maxpooling1D | Input: (None,4000,32) | Output: (None,2000,32)
Block4: Conv1D | Input: (None,2000,32) | Output: (None,2000,64)
Block5: Activation | Input: (None,2000,64) | Output: (None,2000,64)
Block5: Maxpooling1D | Input: (None,2000,64) | Output: (None,1000,64)
Block5: Conv1D | Input: (None,1000,64) | Output: (None,1000,128)
Block6: Activation | Input: (None,1000,128) | Output: (None,1000,128)
Block6: Maxpooling1D | Input: (None,1000,128) | Output: (None,500,128)
Block6: Conv1D | Input: (None,500,128) | Output: (None,500,128)
Act: activation | Input: [(None, 500,128)] | Output: [(None, 500,128)]
Mp: max pooling1D | Input: [(None,500,128)] | Output: [(None,250,128)]
GAP: Global Average Pooling Conv1D | Input: [(None,250,128)] | Output: [(None,128)]
DS: dense | Input: [(None, 128)] | Output: [(None, 128)]

Figure 2. Proposed modified-CNN layer structure.

architectures to normalise data through gradient transmission along with forward-pass. Furthermore, this has been found to be useful in addressing the problem of dataset imbalance as well as improving the classification model's performance.

Experimental setup

Dataset

The ELSDSR dataset, the TIMIT dataset, the 16000PCM dataset, and the recorded dataset made using voice samples from students are among the speaker datasets. ELSDSR is a dataset comprising a total of 22 speakers, 12 of whom are male and 10 of whom are female. There are nine samples for each speaker. Seven of the nine samples utilised in this study were used to train the model, while two samples were utilised to test it. TIMIT is a relatively large dataset, comprising 64 speakers and 10 samples per speaker. Seven of the ten samples are utilised to train the Neural Network, while the remaining three are utilised

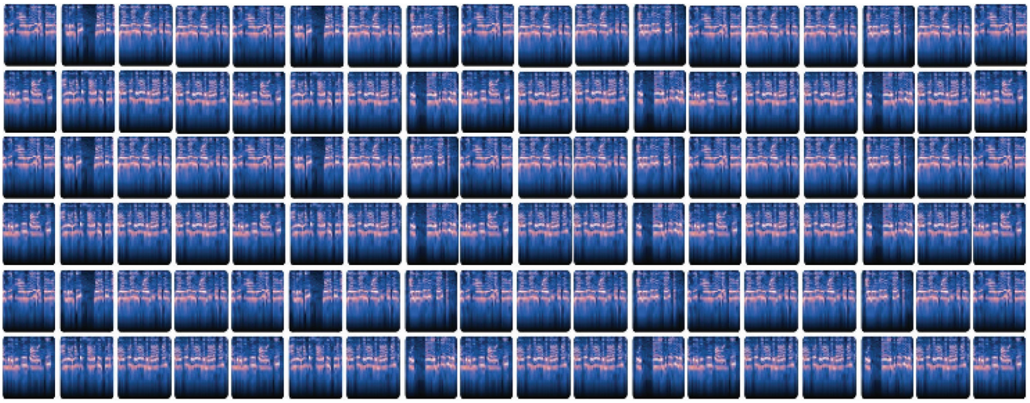

Figure 3. Sample features of recorded speech samples.

to test it. 16000PCM is a useful dataset featuring five important leaders as speakers, including Nelson Mandela, Julia Gillard, and others, three of whom are male and two of whom are women. Each speaker has 1500 speech samples, each of which is 16,000 PCM encoded and lasts 1 s. The recorded dataset is collected from students and faculty members and is divided into 100 samples at a sampling rate of 16 KHz, with each sample being utilised to train and test the model. To distinguish and differentiate background noise from the speaker, a folder with background noise is also supplied.

Implementation details

All of the deep learning experiments were carried out using the Python-based Keras framework. Modified-CNN was also subjected to the Keras framework. Experiments were carried out with Jupyter Notebook Colab (Google Colaboratory), a Python 3.6-based notebook with a graphics processing unit configuration. Various hyperparameters were chosen and reformed to achieve the proposed network design. With softMAX activation function in the dense layer, batch size set to 128 samples per training step, and the model check pointer used to save the best weights, the model was trained for 100 epochs using cross-entropy loss and Adam optimiser. The combination of tanh and ReLU activation functions are used in the generator blocks, and the Leaky ReLU activation function is utilised for entire levels of the discriminator, as well as the sigmoid activation function in the discriminator's final layer.

In the case of the state-of-the-art networks utilised in comparisons, the following implementation details were used during the experiment. All of the state-of-the-art networks were run in Python using the Keras deep learning framework. A fifteen-layer architecture is employed for the traditional CNN utilised as the comparison method, which includes repeating blocks of Convolution and Max-pooling layers, as well as one fully linked layer with the various parameters setting used as the activation functions softMAX and ReLU, as well as the stochastic gradient descent with momentum (SGDM) optimiser and the cross-entropy loss function (Bunrit et al., 2019). In the CNN architecture layer, a dropout layer with a threshold value of 0.6 was also introduced (i.e., 60% of the input units are dropped). The Deep Neural Network is a multi-layer perceptron with ten hidden layers that uses the stochastic gradient descent algorithm to initialise and update the weights in each layer. Stochastic gradient descent employs a single learning rate (0.001) and the ReLU activation function throughout the training procedure (Snyder et al., 2016). Convolution, pooling, and fully connected layers are the three fundamental components of the VGG16 transfer learning model. For the experimental task involving pre-trained networks, it begins with two convolution layers along with pooling, afterwards next two convolutions along with pooling (Masum & Shahriar, 2020), then 3 convolutions along with pooling, plus 3 fully connected layers with Adams-optimiser also sparse cross-entropy loss function. The algorithm of the proposed model is explained below

```
Algorithm: Modified-CNN
Input   : $C_{pq}$ − ConvolutionFilterSize, batchsize(B), DilationRate(d), and Block Size-6;
          Databases: $S_i$ (i = 1,2 . . . . . .k) input speech data and $\widehat{S_t}$ Estimated Speaker
Output: Speaker Features; $S_P$ Predicted Speaker
Step1 : Complete the Initialisation process by setting the needed parameters
Step2 : Apply the speech signal $S_i$ to the input layer
Step3 : Extract the mel-spectrogram of the applied voice signal using the pipelined operations of windowing, FFT, mel-
          scaling and mapping
Step4 : For each epoch
Step5 :   For each batch
Step6 :     For each block
Step7 :         Extract the batch information
Step8 :         Run the Activation, Convolution and Polling functions
Step9 :     End
Step10:   End
Step11:   Run the Adams optimiser and cross-entropy loss operations
Step12:   Measure the accuracy between $\widehat{S_t}$ and $S_P$
Step13: End
Step14: Validate the performance measures
Step15: Display the final outputs
```

Performance parameters

Under this heading, the general performance parameters are discussed. The general parameters covered in this topic include accuracy, loss function, validation accuracy, and validation loss.

Accuracy is a broad term that indicates how well a model performs across all classes. When all classes are equally important, accuracy is taken into consideration. The ratio between the number of right predictions and the total number of speaker samples (Karthikeyan & Suja Priyadharsini, 2021; Ye & Yang, 2021) can be calculated mathematically.

$$Accuracy = \frac{True_{Positive}}{True_{Positive} + True_{Negative} + False_{Positive} + False_{Negative}} \tag{1}$$

Accuracy should be more for an optimal network to classify and recognise the) speaker.

The **Loss function** is a specific parameter in neural networks that are used for optimising parameter values in it. It is a method of evaluating how fair the algorithm models the given dataset. In other words, the loss function is the measure of the absolute difference between the actual output value and the predicted output value. Mathematically,

$$Loss\ Function = Abs\left(Speaker_{predicted} - Speaker_{original}\right) \tag{2}$$

Validation is done by validating the trained model against random samples. While validation, the accuracy obtained is known as Validation Accuracy. In other words, it is also called testing accuracy. While validation, the obtained loss is called Validation Loss. In other words, it is also called testing loss or error function.

Results and discussions

From experimental outputs, it was evident that the modified layered-CNN approach for speaker recognition, performs remarkably well compared to other traditional schemes in the case of experimental, ELSDSR, TIMIT, and 16000PCM datasets illustrated in section 3.1. We have tested the performance of the proposed model against the other state-of-the-art models: (i) Conventional-CNN (15 Layered) (ii) Deep Neural Network (iii) Pre-trained model (VGG16).

Observation1: Experimentation using recorded experimental Dataset

This work focuses to recognise the speaker's identity from their voice samples. The experimental speaker data (recorded dataset) is captured at 16 KHz with 16-bit resolution (Karthikeyan & Suja Priyadharsini, 2021). Six males and five females are represented in the data set. The audio recordings are then converted to wav files using the Audacity software. Finally, each file was cut into 100 samples of 2 seconds each. Sixty samples are for training and the 40 samples are for testing for each speaker.

From the above Table 1, the proposed CNN of modified layered Mel-spectrogram-based scheme outperforms the other DL models in terms of the evaluation metrics accuracy and loss function. Table 1 shows that the modified layered CNNs have a 1.02% loss compared to 7.86% loss for the DNN baseline system.

Observation2: Experimentation using ELSDSR Dataset

English Language Speech Database for Speaker Recognition dataset (ELSDSR) dataset consists of speech data of 22 speakers with 12 males and 10 females. Table 2 indicates the performances of state-of-the-art networks for the ELSDSR dataset.

The accuracy and loss function results are listed in Table 2. It shows that proposed modified-CNN systems perform better with increasing numbers of speakers.

Table 1. Performances of DL models for a recorded experimental dataset.

	Deep Learning Model							
	Deep Neural Network		TL (VGG-16)		Conventional CNN		Modified Layered CNN	
Dataset	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
Experimental Dataset	0.9286	0.0786	0.9818	0.0652	0.9887	0.0212	0.9982	0.0102

Observation3: Experimentation using TIMIT Dataset

Table 2. Performances of DL models for ELSDSR dataset.

	Deep Learning Model							
	Deep Neural Network		TL (VGG –16)		Conventional CNN		Modified Layered CNN	
Dataset	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
ELSDSR Dataset	0.9128	0.0657	0.9712	0.0525	0.9818	0.0196	0.9902	0.0113

Table 3. Performances of DL models for TIMIT dataset.

	Deep Learning Model							
	Deep Neural Network		TL (VGG –16)		Conventional CNN		Modified Layered CNN	
Dataset	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
TIMIT Dataset	0.8968	0.0662	0.9821	0.0621	0.9900	0.0246	0.9956	0.0132

TIMIT Acoustic Phonetic Continuous Speech Corpus dataset contains the audio files of 64 speakers. The dataset includes studio-quality recordings of 64 speakers sampled at 16 kHz, representing the eight key idioms of American English. We use seven for training and three for testing from each speaker's 10 sentences. Table 3 depicts the validation metrics of the proposed modified layered CNN model against the conventional schemes.

Table 4. Performances of DL models for 16000PCM dataset.

	Deep Learning Model							
	Deep Neural Network		TL (VGG −16)		Conventional CNN		Modified Layered CNN	
Dataset	Acc	Loss	Acc	Loss	Acc	Loss	Acc	Loss
16,000 PCM Dataset	0.9352	0.0486	0.9912	0.0281	0.9760	0.0395	0.9931	0.0220

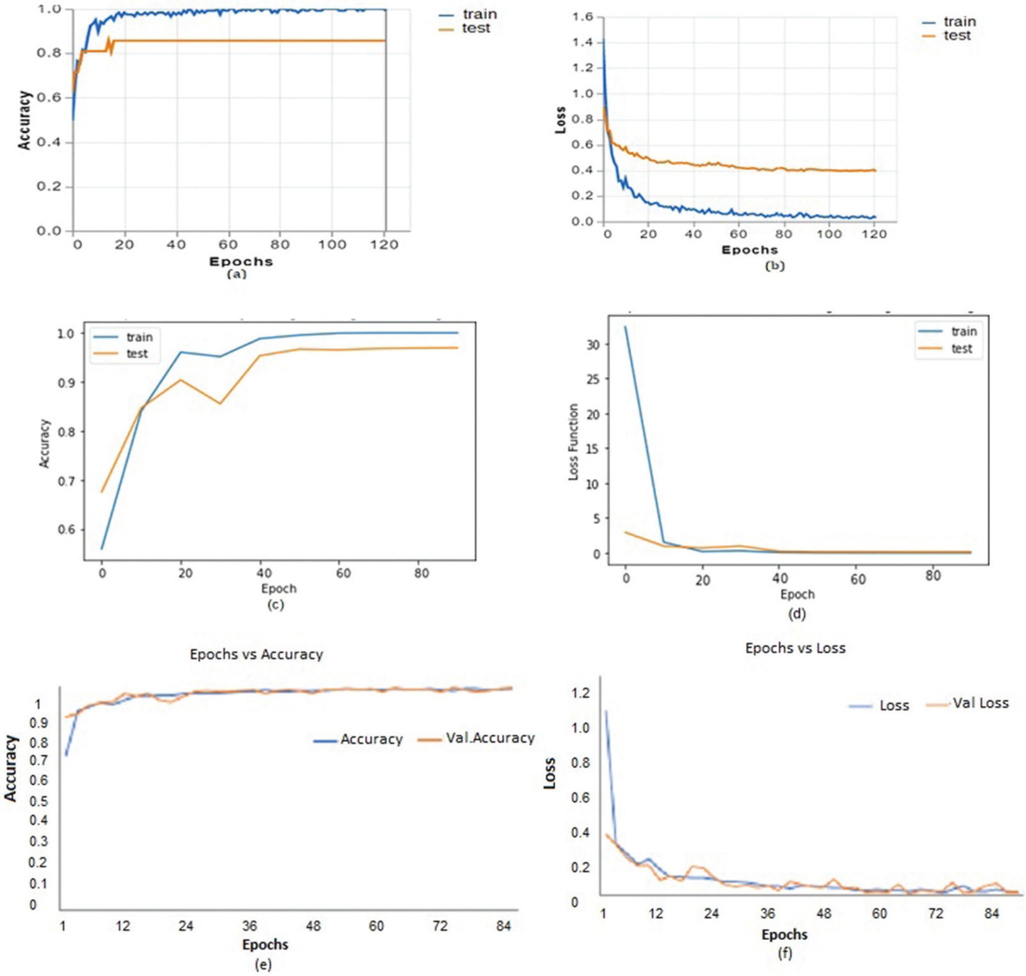

Figure 4. (a)–(f). Sample validation performances of the DL models for the recorded dataset.

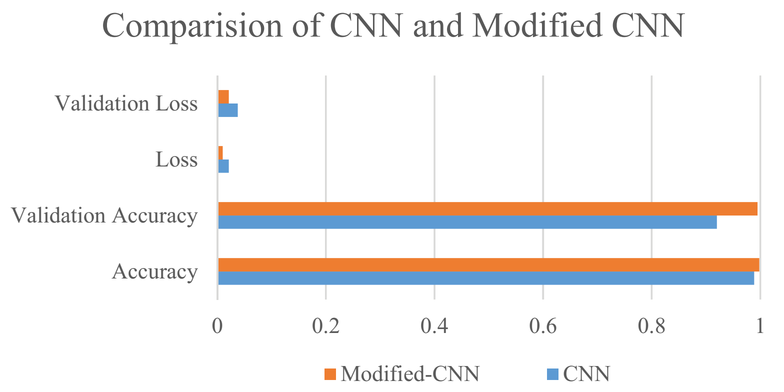

Figure 5. Classification results of the proposed CNN against conventional CNN.

The accuracy and loss function of the proposed system is 99.6% and 1.3% better than our classical convolution neural network system (99% and 2.46%) in the TIMIT speaker dataset.

Observation4: Experimentation using 16000PCM Dataset

The speeches of renowned leaders Benjamin Netanyahu, Jens Stoltenberg, Julia Gillard, Margaret Tacher, and Nelson Mandela are included in this collection. Each audio file in the folder is a PCM encoded one-second 16,000 sample rate wav file. During training, the laughter and clapping (i-e noise) of the audience are blended in with the speech. The suggested model produces 1.7% higher accuracy and a lower loss function than the typical convolutional neural network scheme, as shown in Table 4.

From performance Tables 1–4, it is evident that the proposed modified CNN architecture performs in all the voiceprints with higher accuracy (99% avg) and lower loss function (1% avg).

Observation 5: Validation of proposed CNN model

The Validation is done by validating the response of the trained model against random speaker samples in the dataset. The transfer learning model (VGG-16) is not the best option for SI/SV as the validation results are not satisfactory. The validation accuracy is 0.8571 and the validation loss is found to be 0.0392. The outputs are plotted as graphs in Figure 4(a,b) for better visualisation. The validation accuracy and loss function of the conventional CNN model is plotted against the epochs in Figure 4(c,d), from these graphs it is inferred that the accuracy decreases (0.9200) with the number of epochs, whereas the loss increases with epochs (0.0379). The validation performance measures of the proposed CNN architecture is represented in Figure 4(e,f).

The validation accuracy of modified CNN which is 99.47%, which is much higher than the conventional CNN (92%) and the validation loss is 2.11% is lower than the other DL learning models. Figure 5 shows the overall performance of layer modified CNN against the convention CNN.

Conclusion

Speaker identification is a trend that became a day-to-day necessary task that is applied in the fields of authentication, security and forensic sectors, etc. This work employs layer-modified Convolutional Neural Networks on mel-spectrograms to acquire speaker-specific attributes from a wav audio wave formats. The proposed network is designed with more convolutional, activation, pooling, and dense layers to improve the prediction accuracy. The performance of the proposed prototype is compared with other state-of-the-art models such as Transfer Learning VGG-16, DNN, and generic-CNN. The modified CNN outperforms all other architectures in terms of both accuracy and loss function. Its accuracy is found 1.02% (avg) greater than generic-CNN and the loss function is found 1.20% (avg) lesser than that of generic-CNN. The response of the model is validated using several datasets including the 16000PCM speaker dataset, the TIMIT dataset, the ELSDSR dataset, and the recorded dataset.

The future aims to design a real-time recognising module for identifying Ragas (of Carnatic Music) through Musical notes. It can be applied further for Singer cum Raga Identification and to develop Raga-therapy-based applications to solve Human Diseases without many expense, by automation.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

Suja Priyadharsini S http://orcid.org/0000-0002-3926-5263

References

Ai, O. C., Hariharan, M., Yaacob, S., & Chee, L. S. (2012). Classification of speech dysfluencies with MFCC and LPCC features. *Expert Systems with Applications*, *39*(2), 2157–2165. https://doi.org/10.1016/j.eswa.2011.07.065

Bhattacharya, G., Alam, J., & Kenny, P. (2017). Deep speaker embeddings for short-duration speaker verification. *Proceedings of Interspeech 2017* (pp. 1517–1521). https://doi.org/10.21437/Interspeech.2017-1575

Billeb, S., Rathgeb, C., Reininger, H., Kasper, K., & Busch, C. (2015). Biometric template protection for speaker recognition based on universal background models. *IET* Biometrics, *4*(2), 116–126. https://doi.org/10.1049/iet-bmt.2014.0031

Bunrit, S., Inkian, T., Kerdprasop, N., & Kerdprasop, K. (2019). Text-independent speaker identification using deep learning model of convolution neural network. *International Journal of Machine Learning and Computing*, *9*(2), 143–148. https://doi.org/10.18178/IJMLC.2019.9.2.778

Cai, W., Chen, J., & Li, M. (2018). Exploring the encoding layer and loss function in end-to-end speaker and language recognition system. *Odyssey*. https://doi.org/10.21437/Odyssey.2018-11

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., & Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, *19*(4), 788–798. https://doi.org/10.1109/TASL.2010.2064307

Dinkel, H., Chen, N., Qian, Y., & Yu, K. (2017). End-to-end spoofing detection with raw waveform CLDNNS. *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4860–4864). https://doi.org/10.1109/ICASSP.2017. 7953080.

Garcia-Romero, D., & Espy-Wilson, C. (2011). Analysis of i-vector length normalization in speaker recognition systems. *12th Annual Conference of the INTERSPEECH* (pp. 249–252). https://doi.org/10.21437/Interspeech.2011-53.

Hajibabaei, M., & Dai, D. (2018). *unified hypersphere embedding for speaker recognition*. ArXiv, abs/1807.08312.

Heigold, G., Moreno, I., Bengio, S., & Shazeer, N. M. (2016). End-to-end text-dependent speaker verification. *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 5115–5119). DOI: https://doi.org/10.1109/ICASSP.2016.7472652

Hu, H., Tang, B., Gong, X., Wei, W., & Wang, H. Intelligent fault diagnosis of the high-speed train with big data based on deep neural networks. (2017). *IEEE Transactions on Industrial Informatics*, *13*(4), 2106–2116. 10.1109/TII.2017.2683528. https://doi.org/10.1109/TII.2017.2683528

Jain, A. K., Ross, A., & Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, *14*(1), 4–20. https://doi.org/10.1109/TCSVT.2003.818349

Kabal, P., & Ramachandran, R. The computation of line spectral frequencies using chebyshev polynomials. acoustics, speech and signal processing. (1987). *IEEE Transactions On ASSP*, *34*(6), 1419–1426. 10.1109/ TASSP.1986.1164983. https://doi.org/10.1109/TASSP.1986.1164983

Karthikeyan, V., & Suja priyadharsini, S. (2021). A strong hybrid adaboost classification algorithm for speaker recognition. *Sādhanā*, *46*(3), 1–19. https://doi.org/10.1007/s12046-021-01649-6

Karthikeyan, V., & Suja Priyadharsini, S. (2022). Hybrid machine learning classification scheme for speaker identification. *Journal of Forensic Sciences*, *46*(3), 1033–1048. https://doi.org/10.1111/1556-4029.15006

Kenny, P. (2010). *Bayesian speaker verification with heavy-tailed priors, in odyssey, 2010*.

Kenny, P., Stafylakis, T., Ouellet, P., Gupta, V., & Alam, M. J. (2014). Deep neural networks for extracting baum-welch statistics for speaker recognition *Odyssey, The Speaker Lang. Recognition workshop* Finland, vol 2014. (pp. 293–298).

Kinnunen, T., & Li, H. (2010). An overview of text-independent speaker recognition: From features to supervectors. *Speech Communication*, *52*(1), 12–40. https://doi.org/10.1016/j.specom.2009.08.009

Li, C., Ma, X., Jiang, B., Li, X., Zhang, X., Liu, X., Cao, Y., Kannan, A., & Zhu, Z. (2017). Deep Speaker: An end-to-end neural speaker embedding system. *ArXiv, abs/1705.02304*.

Martinez, J., Perez, H., Escamilla, E., & Suzuki, M. M. (2012). Speaker recognition using mel frequency cepstral coefficients (MFCC) and vector quantization (VQ) techniques. *CONIELECOMP 2012, 22nd International Conference On Electrical Communications And Computers* (pp. 248–251). https://doi.org/10.1109/CONIELECOMP. 2012.6189918.

Masum, M., & Shahriar, H. (2020). TL-NID: Deep neural network with transfer learning for network intrusion detection. *2020 15th International Conference For Internet Technology And Secured Transactions (ICITST)* (pp. 1–7). 10.23919/ ICITST51030.2020.9351317.

Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J. M., Courville, A. C., & Bengio, Y. (2017). SampleRNN: An unconditional end-to-end neural audio generation model. *ArXiv, Abs/1612.07837*.

Muckenhirn, H., Magimai.-Doss, M., & Marcel, S. (2018). Towards directly modeling raw speech signal for speaker verification using CNNS. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4884–4888). https://doi.org/10.1109/ICASSP.2018.8462165.

Nagrani, A., Chung, J. S., & Zisserman, A. (2017). VoxCeleb: A large-scale speaker identification dataset. *INTERSPEECH*.

Palaz, D., Magimai-Doss, M., & Collobert, R. (2015). Analysis of CNN-based speech recognition system using raw speech as input. Proceedings of. *Interspeech 2015* (pp. 11–15). https://doi.org/10.21437/Interspeech.2015-3

Prabhakar, S., Pankanti, S., & Jain, A. (2003). Biometric recognition: Security And privacy concerns. *Security & Privacy, IEEE*, *1*(2), 33–42.

Ravanelli, M., Brakel, P., Omologo, M., & Bengio, Y. (2018). Light gated recurrent units for speech recognition. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *2*(2), 92–102. https://doi.org/10.1109/TETCI.2017.2762739

Richardson, F., Reynolds, D. A., & Dehak, N. (2015,a). A unified deep neural network for speaker and language recognition. Proceedings of *Interspeech 2015* (pp. 1146–1150). https://doi.org/10.21437/Interspeech.2015-299.

Richardson, F., Reynolds, D., & Dehak, N. (2015,b). Deep neural network approaches to speaker and language recognition. *IEEE Signal Processing Letters*, *22*(10), 1671–1675. https://doi.org/10.1109/LSP.2015.2420092

Sadjadi, S. O., Ganapathy, S., & Pelecanos, J. W. (2016). The IBM 2016 speaker recognition system. *Odyssey*. arXiv:1602.07291.

Salehghaffari, H. (2018). *Speaker verification using convolutional neural networks*. abs/1803.05427. ArXiv.

Singh, S., & Rajan, E. (2011). Vector quantization approach for speaker recognition using MFCC and inverted MFCC. *International Journal of Computer Applications*, *17*(1), 1–7. https://doi.org/10.5120/2188-2774

Snyder, D., Ghahremani, P., Povey, D., Garcia-Romero, D., Carmiel, Y., & Khudanpur, S. (2016). Deep neural network-based speaker embeddings for end-to-end speaker verification. *2016 IEEE Spoken Language Technology Workshop (SLT)* (pp. 165–170). https://doi.org/10.1109/SLT.2016.7846260

Snyder, D., Garcia-Romero, D., Povey, D., & Khudanpur, S. (2017). Deep neural network embeddings for text-independent speaker verification. Proceedings of *Interspeech 2017* (pp. 999–1003). https://doi.org/10.21437/ Interspeech.2017-620.

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-Vectors: Robust DNN embeddings for speaker recognition. *2018 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)* (pp. 5329–5333). https://doi.org/10.1109/ICASSP.2018.8461375.

Tiwari, V. (2010). MFCC and its applications in speaker recognition. *International Journal on Emerging Technologies*, *1*(1), 19–22.

Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., & Zafeiriou, S. (2016). Adieu features? End-to-end speech emotion recognition using a deep convolutional recurrent network. In *2016 IEEE international Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE Press* (pp. 5200–5204). https://doi.org/10.1109/ ICASSP. 2016.7472669

Variani, E., Lei, X., McDermott, E., Lopez-Moreno, I., & Gonzalez-Dominguez, J. (2014) Deep neural networks for small footprint text-dependent speaker verification. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4052–4056). https://doi.org/10.1109/ICASSP.2014.6854363.

Wan, L., Wang, Q., Papir, A., & Lopez-Moreno, I., (2018). Generalized end-to-end loss for speaker verification. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 4879–4883). https://doi.org/10. 1109/ICASSP.2018.8462665

Xie, W., Nagrani, A., Chung, J. S., & Zisserman, A. (2019). Utterance-level aggregaton for speaker recognition in the wild. *IEEE international Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 5791–5795). https://arxiv.org/ abs/1902.10107.

Yaman, S., Pelecanos, J., & Sarikaya, R. (2012). Bottleneck features for speaker recognition. *IEEE Odyssey*, *12*(12), 105–108.

Ye, F., & Yang, J. A. (2021). Deep neural network model for speaker identification. *Applied Sciences*, *11*(8), 3603. https://doi.org/10.3390/app11083603

Yujin, Y., Peihua, Z., & Qun, Z. (2010). Research of speaker recognition based on combination of LPCC and MFCC, *2010 IEEE International Conference on Intelligent Computing and Intelligent Systems*, vol *3*, pp. 765–767.

Zhang, C., Koishida, K., & Hansen, J. H. L. (2018). Text-independent speaker verification based on triplet convolutional neural network embeddings. IEEE Transactions on Audio, Speech, and Language, *26* (9), 1633–1644. DOI https://doi.org/10.1109/TASLP.2018.2831456