Deep Learning Model CNN With LSTM For Speaker Recognition

Bassel Alkhatib[1] Mohammad Madian Kamal Eddin[2]
[1]Web Master Director- Syrian Virtual University
Damascus Syria and the Faculty of Information Technology
Engineering-Damascus University, Syria
drbasselalkhatib@gmail.com

[2]Student at the PhD program- Syrian Virtual University
Damascus Syria
2k.madian123@gmail.com
ORCID: https://orcid.org/0000-0003-3806-2920
k.madian123@gmail.com

JDIM
Journal of Digital Information Management

ABSTRACT: *Speech recognition is one of the most important research fields nowadays because of its necessity in our daily lives and to raise the fields of security to the highest level, It's a task of speech processing, and our main scope in this paper is on speaker verification, which is to identify persons from their voices where the process depends on digitizing the sound waves into a form that allows the system to deal with it. The verification process is based on the characteristics of the speaker's voice (voice biometrics) and sends it to a further process to extract the features of that voice using the feature extraction method and using AI techniques to perform the task of identification. MFCC is used for the task of features extraction and obtains the spectrogram of a given voice signal where it represents a bank of information about the voice and sends it to the CNN model for further processing for training the model on that signal to verify if the voice belongs to a user in the system or it's a new enrollment.*

Subject Categories and Descriptors: [I.2.7 Natural Language Processing]; Speech recognition and synthesis [I.5 PATTERN RECOGNITION]; Neural nets [B.4 INPUT/OUTPUT AND DATA COMMUNICATIONS];

General Terms: Speech Recognition, Neural Networks, Feature Extraction

Keywords: ASR, Speech Verification, MFCC, CNN

Received: 19 August 2022, 18 September 2022, Accepted 11 November 2022

Review Metrics: 0/6, Review Score: 4.95, Inter-reviewer Consistency: 92%

DOI: 10.6025/jdim/2022/20/4/131-147

1. Introduction

Sound is a mechanical frequency, or a wave capable of moving in several physical mediums such as solid bodies, liquids, and gases, does not spread in a vacuum, and the organism can sense it through a special organ called the ear. From the perspective of biology, the sound is a signal that contains a tone or several tones issued by the organism that owns the emitting organ, used as a means of communication between it and another organism of its kind or another sex, through which it expresses what it wants to say or do, consciously or unconsciously. The sensation caused by these vibrations is called the sense of hearing.

Sound is the basis of many experiences that humans acquire, and the speed of sound in a normal antenna is estimated at 343 meters per second or 1224 kilometers per hour. The speed of sound is related to the stiffness factor and density of the material in which the sound is moving.

Sound is the primary form of communication between humans, so humans developed sounds until they reached their formal use, calling it a language that they used to communicate with each other using their voices to express what they wanted. The voice is unique to each individual [27] as research has proven, and advanced technology in the past few years has opened up a new field of research to study the characteristics of voices and this field is called Natural Language Processing (NLP).

NLP is the ability of a machine (computer, phone, etc.) to

interact with humans using spoken or written language. Where NLP is a sub-field of Artificial Intelligence (AI) [28]. NLP has many research subfields such as (text classification, text extraction, machine translation, natural language generation, and processing) and these elements of NLP are used in many real-world applications.

A direct benefit we can get from NLP is speech tasks such as customer service automation (where voice assistants can use speech recognition to understand what a customer is saying and detect whether it is a new or an old customer using his voice) [28], so NLP is useful To be a part of personal assistants like Alexa by enabling it to understand the spoken word (text-to-speech), or enable chatbots that have a full conversation with users or customers using technologies techniques like (text-to-speech and speech-to-text), So, we can get the advantages of using NLP to serve the concept of speech recognition and speaker identification where the machine interprets the important elements of human language and voice that correspond to a specific feature in the dataset and returns the answer.

The main types of natural language processing (NLP) for speech:

• Speech Recognition (SR):
SR is the methodologies and technologies that enable the recognition and translation of spoken language into text by Computers. It is also known as "automatic speech recognition" (ASR), or just "speech to text" (STT) [1].

• Speaker Verification (SV):
SV is the process of automatically identifying who is speaking by using the speaker's information contained in speech waves to verify the identity claimed by people who access the systems that have been trained on a specific person's voice or can be used to authenticate or verify the identity of the speaker as part of a security operation [2].

• Speech Synthesis (SS):
SS is the computer-generated simulation of human speech or Text-to-Speech System (TTS) which converts plain language text into speech. Other systems make linguistic representations symbolic, such as phonemic transcription into speech [3].

Automatic speaker verification (SV) is the process of identifying the speaker from the voice signal, where the voice contains a special characteristic like speaker pronunciation, accent, vocal tract, and rhythm [4], so it's the task to recognize the identity of someone based on the speech signal (voicePrint). Like fingerprints, gestures, retina, iris, and faces, voice is the most direct way of human communication. But unlike the other verification methods that depend on images, ASR is based on variable values (voice signal) that change over time.

There are two main types of Speaker Verification (SV):

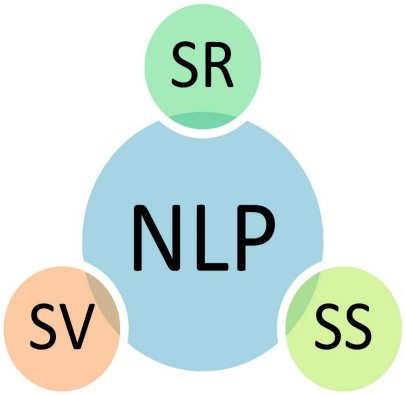


Figure 1. NLP main types

• Text-Dependent Speaker Verification (TD_SV)
The process of identifying the speaker by repeating the same utterances that are used in the enrollment phase when the user registered into the system, the system is based on the speech signal and extracts the word content and must be the same as that stored in the database files.

• Text-Independent Speaker Verification (TI-SV)
The process of identifying the speaker with no constraint on the speech signal where the user can say anything and speak freely.

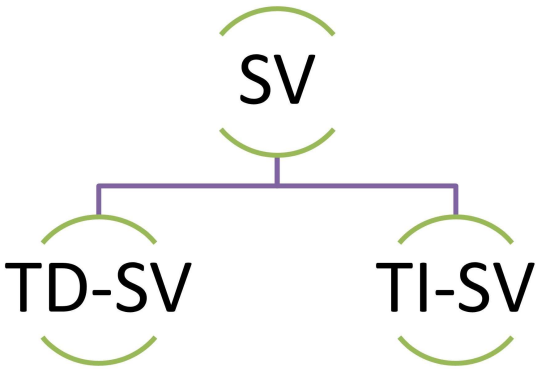


Figure 2. SV main types

In this paper, text-independent speaker Verification (TI-SV) will be used because it is more convenient when the speaker can speak freely to the system. However, it requires more processes to achieve good performance.

Many researchers have been prepared for the task of speaker identification and verification, but the research presented assumed that a large amount of training data should be available to train the system [5,6] on it before starting to work. The process of verifying the identity of the speaker needs less data [7,8] than the process of identification (training), but it also needs a huge amount of voice data for the verification process, but there are few works based on the presence of little training data, and from here it is necessary to use new techniques to imitate the human decision process to identify the speaker, as the human being, by nature, depends on little informa-

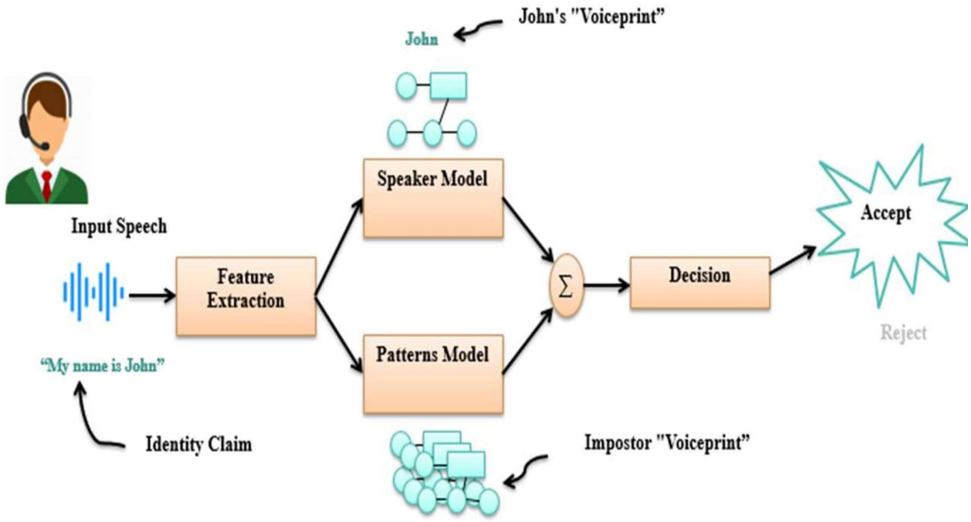


Figure 3. Speaker identification framework

tion to make the decision. here is the role of artificial intelligence to mimic the human decision-making process, and one of the most important techniques to imitate these processes is neural networks, which are based on simulating the human mind.

2. Related Works

In the last few years, lots of research work has been done in the field of speaker identification, where different techniques have been used for feature extraction (LPCC, MFC, LPC, MFCC) and classification models. many classification methods are used to classify these features like (GMM, HMM, ANN, DNN, VQ, and SVM) [29]. Jorge M and others developed a speaker recognition system where they used MFCC for feature extraction and VQ for the task of classification, where their system generate a codebook for each speaker clustering the acoustic vectors of each one in the system's environment, the system matches the incoming voice with each trained codebook and the total VQ distortion is calculated, the speaker with the smallest distortion is the one that matches with the incoming voice. the system achieved 82% overall accuracy [30]. Hong Yu and others developed a speaker identification system where they improved the way that MFCC work and called it super Mel-frequency cepstrum coefficients by cascading three MFCC frames together. where the Histogram transform method estimated the probability density function of these coefficients and achieved good results of identification [31]. A text-independent speaker identification system was developed by Amar Aggoun and others based on wavelet analysis and neural networks. Where the wavelet analysis comprises many techniques to process the signal (MFCC, sub-band coding, discrete transform, packet transform). a combination of Neural networks was used for learning the model. The identification rate was improved by 15% and reduced the identification time by 40% in the system compared to the traditional MFCC method [32]. Kevin R and others developed statistics pattern recognition and ANN (an independent speaker recognition system) using various classifiers, where they modified a Neural tree network called (MNTN), they used two methods to achieve and compare the results where the error rates were achieved by both the modified ANN (MNTN) and VQ (vector quantization), but the MNTN demonstrates a logarithmic saving in retrieval time [29]. The most relevant work was developed by Emry C et al., who developed and applied a recurrent convolutional neural network to the task of detecting polyphonic acoustic events. It was a hybrid network that combined both a convolutional neural network (CNN) and a recurrent neural network (RNN), the new hybrid system achieved a remarkable result as it improved performance and achieved better accuracy compared to the single methods, but it took a long time to detect the event for large-scale sound because it depends on the recursive working method [33]. Kartik Mahto and others developed a Singer identification system where they used different techniques and compared them where they used LCP and MFCC for feature extraction and used ANN and Naïve Bayes for classification. MFCC proved to provide a better result as compared to LPC for both classification methods, the best results were achieved by using MFCC and Naïve Bayes classifiers with 77% percentage of identification [34].

3. Overall System Design

Studying the techniques of analyzing voices of users' speeches to use them as a means based on biometric models (voicePrint) and studying the possibility of extracting audio features from the voices of users within the system environment, developing and improving their results by linking them with deep learning techniques to generate and structure new models and features, Train network models on them and characterize them as probabilistic and reference models for users that enable the system to predict users and process new sounds to identify users through a verbal message to enable HCI (Human Computer Interaction) processes to protect their identities and

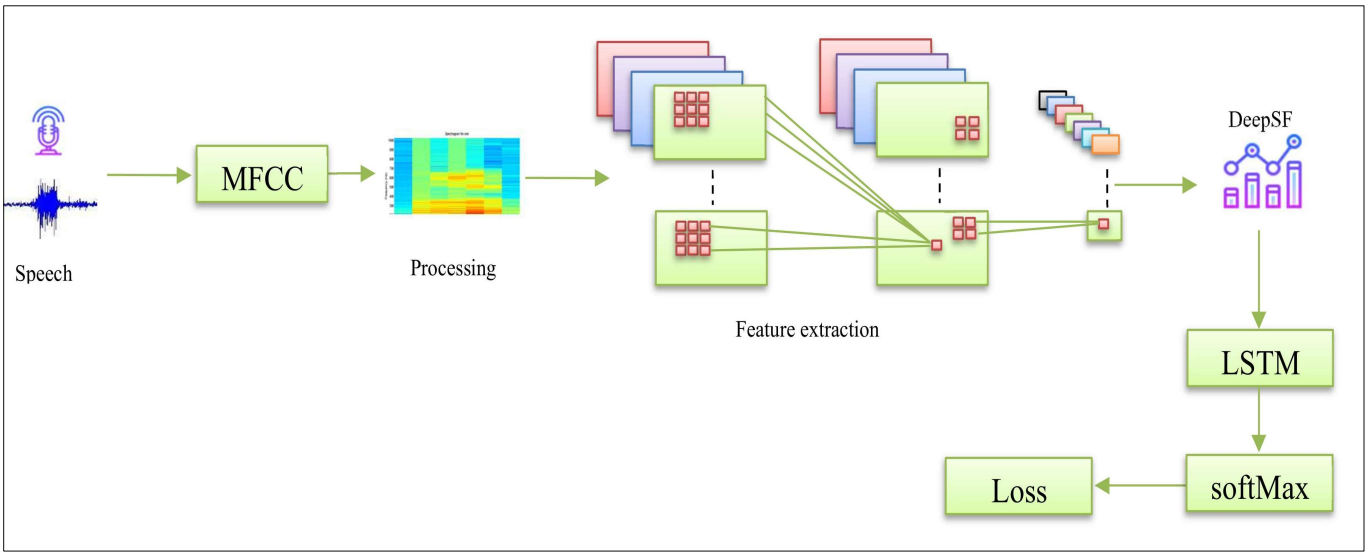


Figure 4. CNN Speaker identification structure

personal information which is the main goal from the model.

In this paper, we will present a methodology of a model that analyzes the voices of users, where the system will extract the distinctive signs of each voice from the users and then process and equip it with an increase in the values of high frequencies against its low counterpart to increase the predictability of the sounds that are described by these frequencies [9] and present it as an input to other models that processing this sound and extracting its patterns for training and representing them as a knowledge network, which constitutes an understanding of the sound patterns and compare them with new patterns when using the system later [10].

A new feature that we will call Deep Learning Speaker Features (DeepSF), where converts the audio features extracted by the MFCC (feature extraction method) into Deep Learning Speaker Features (DeepSF) using a deep neural network technique CNN (Convolutional Neural Network).

The deep speaker features (voice information) are captured by the CNN kernel to perform much processing on it where the model convolution the signal and generates the feature map in the convolution layer after that the model minimizes the number of parameters that the network needs to learn using maxPooling model, this step of processing is followed by the LTSM model where it represents a powerful temporal modeling ability by learning the voice features (signal context information) [29] and maximize the voice content's retention, after that the Relu function is applied where it replaced the negative values with zero, then push the result of the last steps to the fully connected layer and softMax layer to do the job of recognizing speakers' voices and obtain the final classification result. Then the loss function is used as a training criterion where it compares the result of the current cycle to the previous cycle of processing.

This aims to learn and understand the distribution of features and vocal intensity of the speaker, which is itself a characteristic of the speaker, that is, based on learning how these features were distributed in the speaker's voice. Then the model converts each feature of the MFCC into a deep learning feature map corresponding to DeepSF, but before moving forward a question should be asked? How can a voice (audio signal) be converted into coefficients and what do they represent?

MFCCs are the representation of the compressed form of the audio signal spectrum(when the sum of an infinite number of sinusoids is represented by a waveform) [35], and the coefficients contain information about the rate of change in different spectrum bands [35]. Voice has many features and characteristics that are unique to each person [27], and the use of MFCC is to find these features, which can automatically learn a wide range of acoustic features that are represented by (spectrum, pitch, tone, formant, Intensity, Frequency modulation, Group Delay, etc.) where they are all part of the sound source and vocal tract, which greatly improves the accuracy of speaker identification.

To get these features 7 steps must be applied (MFCC steps):

- Process the incoming signal (Digitalization, Pre-emphasis..voice active detection).
- Frame the signal into short frames.
- Windowing.
- Calculation of the Discrete Fourier Transform (DFT).
- Applying Filter Banks.
- apply the log of these spectrogram values to get the log filterbank energies.
- Discrete cosine transforms (DCT).

Convolutional Neural Networks (CNN)

It was known a few years ago to use the approaches of similarity measurement techniques as a criterion for speaker verification after classifying and creating a reference for each speaker using Hidden Markov Models and Gaussian mixture models, but with continuous research and development of new methods to reduce the error rate and reach an acceptable accuracy standard, and with the contemporary artificial intelligence, the aforementioned similarity measurement theories have been replaced by the Deep neural network learning techniques [11], due to fully connected nature of DNN to mimic the human mind [15].

With the increasing development and research within the scope of DNN, there are many problems with its use where the structural location is not captured from the feature space, which is a fundamental defect. In Addition, the DNN encounters the gradient vanishing problem in the training time of stochastic gradient descent (SGD) [12]. Hence, it was necessary to search for solutions to face the problems that resulted within the DNN, and the result was access to the CNN technology, the Convolutional Neural Networks (CNN) succeeded in designing the structural site from the feature space [13]. It also reduces transition contrast and takes care of small perturbations and shifts in the feature space because it adopts clustering (pooling) in a local frequency region and by exploiting the prior knowledge of the speech signal, it was able to take advantage of the long dependencies between speech frames.

A convolutional neural network (ConvNet) is a deep learning network architecture that learns directly from data eliminating the need for feature extraction methods. It's useful for finding a pattern to recognize objects in an image (faces, fingers) and non-image data (signal data, time series) [14].

Advantages of CNN

• CNN is used for classification and recognition because of its high accuracy (Produces highly accurate recognition results).

• CNN automatically detects important features without any human supervision.

• To visualize the speech signal, the signal is converted into spectrogram grid like data and then applied to the CNN method.

• Eliminating the need for manual feature extraction methods (features are learned directly by the CNN).

• CNN can be retrained for new recognition tasks, enabling you to build on pre-existing networks.

• Provides an optimal architecture for uncovering and learning key features in image and time series and it represents the key technology in audio processing keyword detection when a certain word or phrase is spoken like "Search" where CNN can detect the keyword and ignore the other phrases [14].

The CNN can have tens of hundreds of layers that tech to learn to detect different features of the time series, filters applied to each training section at different frames, and the output of each convolved frame is used as the input to the next layer. The filters can start at very simple features and increase in complexity to features that uniquely define the signal.

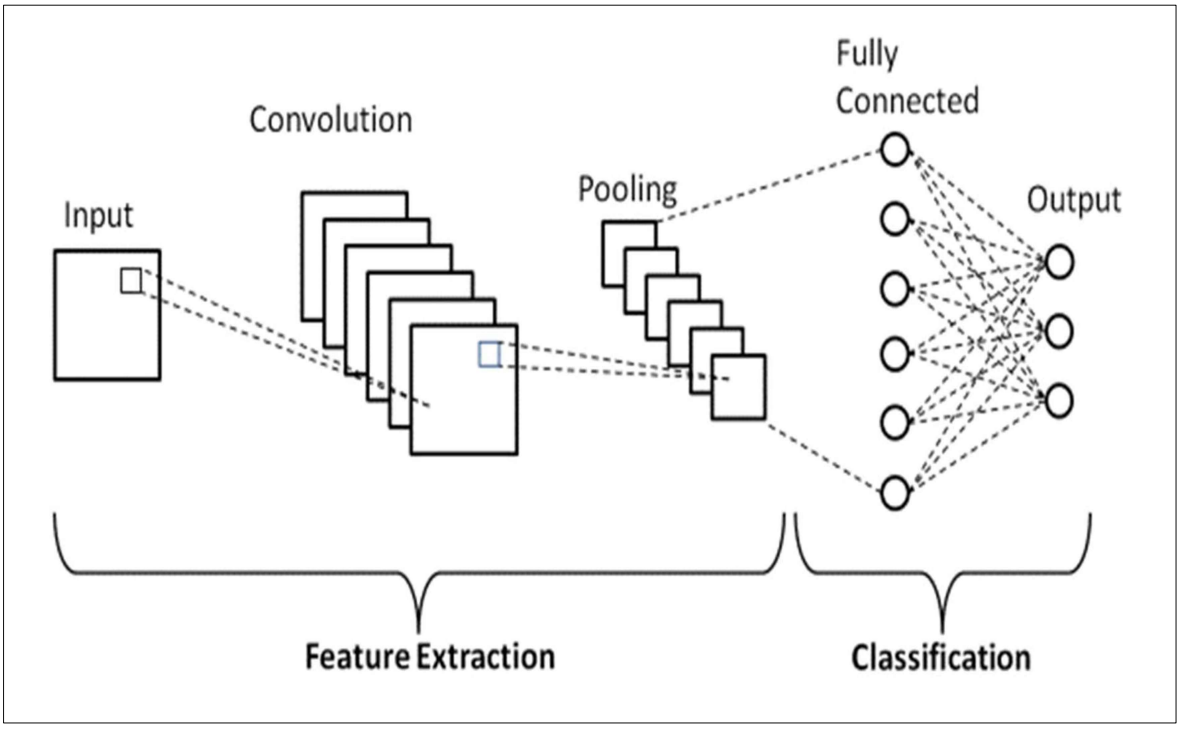


Figure 5. CNN basic structure [25]

CNN Layers

These layers perform operations that modify data to learn data-specific features:

• Convolution Layer

The input data is processed by an array (weights array) of filters (kernels) where these filters refer to a matrix of numbers (pattern) that the filter is looking for, each filter is randomly initialized to normal (Gaussian distribution) and learn to detect different features in the input source (image, time-series, audio), and then get the features map (output of convolution operation).

• Pooling Layer

the main object in this layer is to reduce the number of parameters (spatial size of features) that the network needs to learn which minimizes the learning time, the most popular type of pooling is maxPooling.

• Relu Function

Rectified linear unit (Relu) that employs a non-saturating activation function, where the negative values would be removed from the filtered polling and replaced with zero, in other words if the input is above a certain value activated it, otherwise output it as a zero, so that allows training to be faster and more effective. So, the activated value will represent the input to the next layer.

$$f(x) = \begin{cases} 0 & f(x) < 0 \\ x & f(x) > 0 \end{cases}$$

5	1	-7
2	-2	0
4	-1	3

Filter output

5	1	0
2	0	0
4	0	3

Filter output after Relu

• Fully connected Layer

This layer represents the last classification step where the input of this layer is the flattened vector (matrix flattened to single vector, matrix of X dimension "X is the number of classes that the network will be able to predict") where this vector represents the output of the previous layers (convolutional and pooling) to feed forward the neural network. And defines as:

$$g(Wx + b)$$

Where:

x: the input vector with dimension

W: the weight matrix with dimension.

b: the bias vector with dimension.

• Output Layer

Where the SoftMax activation function is employed for classification in this layer, Which is used to get probabilities of the input being in a particular class:

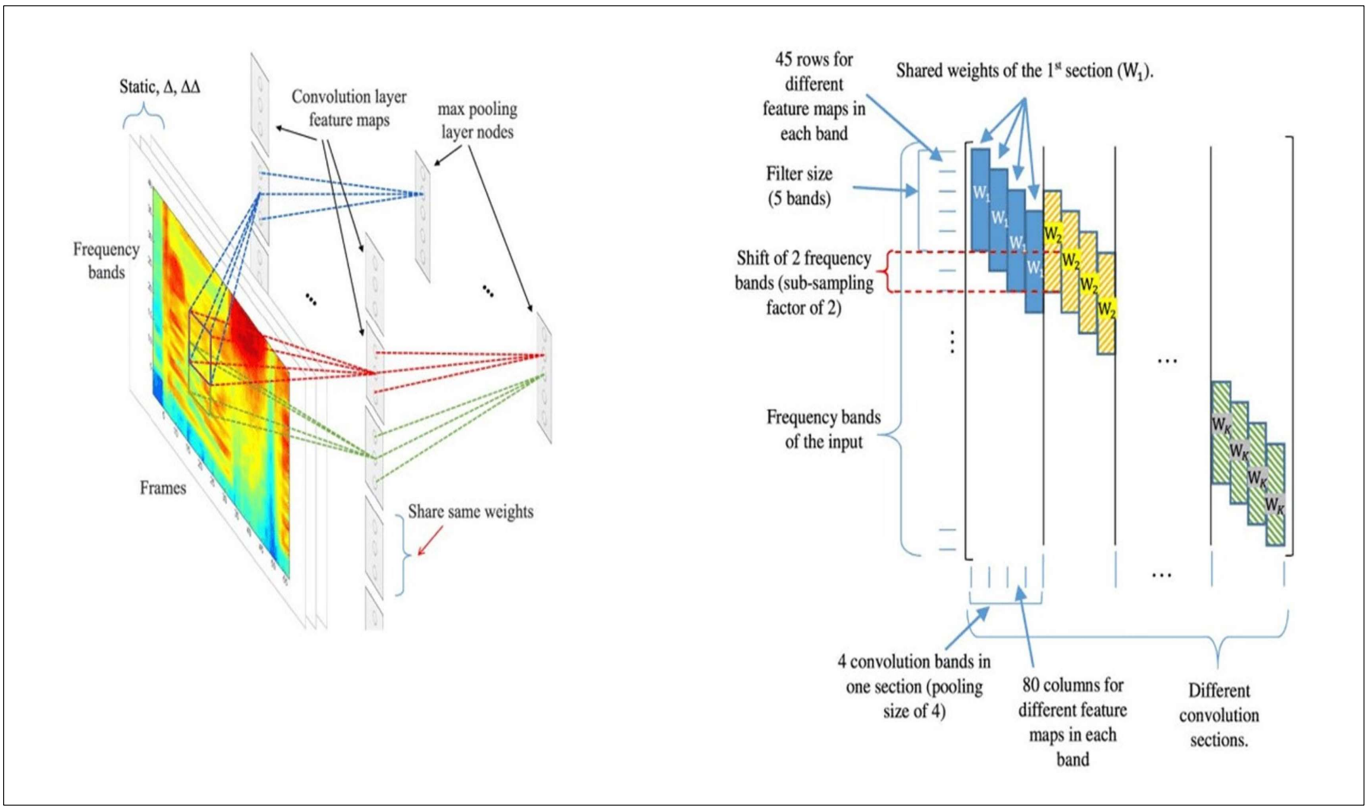


Figure 6. CNN method architecture [19] Long Short-

$$f_i(x) = \frac{\exp(x_i)}{\sum_j \exp(x_j)}$$

Long Short-Term Memory(LSTM):
At this stage of processing the data is a sequence form which is the result of the CNN pooling where the system can't deal with it, before it can be recognized by the model it needs further processing, to process these sequences of data another technique must be used from another method, the best method that can process a sequence of data is the RNN (Recurrent Neural Network) [23] but it has its problem where the gradient can easily disappear during its learning phase because it's weak by learning the context of a continuous signal over time (voice signal) [24]. So, to address this problem and to avoid the vanishing gradient problem, LSTM must be used to prevent over-learning the network for a time series (continuous signal over time) and to improve the recognition rate [24].

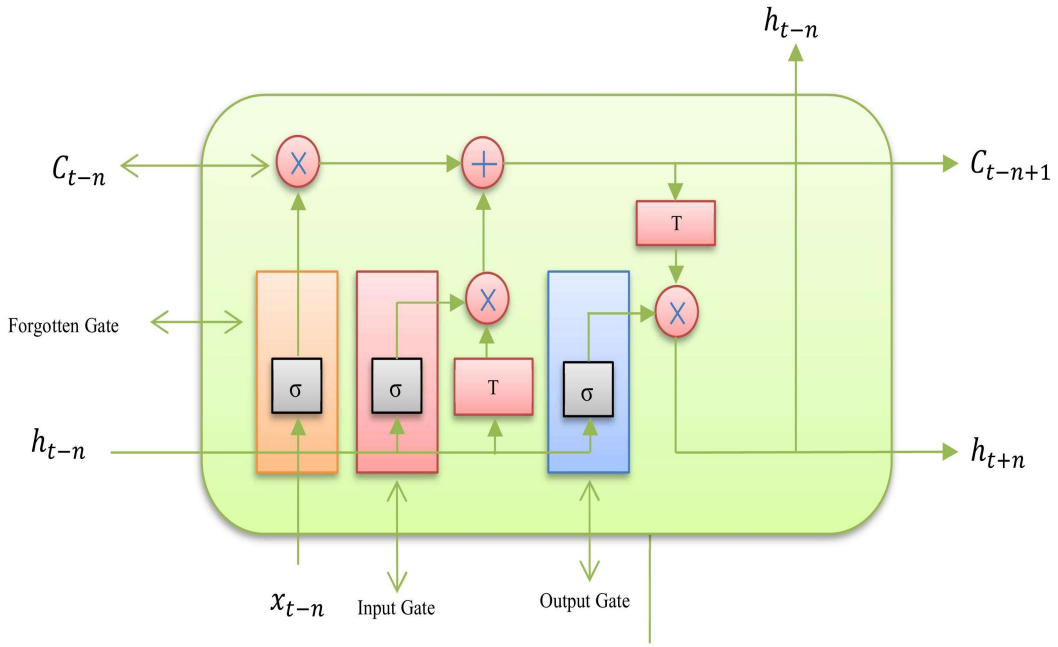


Figure 7. LSTM Basic Structure

To prevent the gradient from disappearing, inside the circulating neural units the dropout was introduced to disconnect the neurons' connections that were made by the processing by about 10%, after that the fully connected layer (softMax) processed the output and do the job by recognizing the claimed identity where it's the last layer to access the network.

The LSTM is defined by the following formulas:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}C_{t-1})$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}C_{t-1})$$
$$C_t = f_t \odot C_{t-1} + C'$$
$$C' = i_t \odot \mathrm{T}(W_{xc}x_t + W_{hc}h_{t-1})$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}C_t)$$
$$h_t = o_t \odot \mathrm{T}(C_t)$$
$$m_t = \mathrm{T}(C_t)$$

Where:

i_t : Input gate.

f_t : Forget gate (features that are discarded in the model).

C_t: neuron activation (current layer's memory unit).

C': Input speaker information.

o_t : Output gate.

h_t : Final speaker information (by the current layer).

σ : sigmoid activation function.

W: weight matrix (for each stage).

Loss Function
The loss function is a process that reacts as assistance to tell how good is the model that we are working on at predicting a giving signal, it has its cost function that represents a unique curve and gradients, which helps in making the model more accurate by showing the result of the current parameters and how to update them, the weights of the neural net can be updated by the help of the gradients which calculated by the loss function.

4. Experiment Analysis and Results

As described earlier in the previous sections of this paper the idea is to take the advantage of the deep learning methods to implement and test a voice verification model

that can identify the speakers from their voices signals (Voice Biometric), for this kind of model the signal must pass through several procedures and processing to reach the final goal of identification and verification of the given data.

The proposed model is implemented using python language with the help of a deep learning library (TensorFlow) which is written also in python and can be executed on GPU (graphics processing unit) which works on parallel processing, usually, it's more efficient than working on CPU (central processing unit) for the neural network. The model was trained using the CNN along with the LTSM model for 50 epochs (224 samples for the training phase and 57 samples for the test phase) where the training pairs are re-shuffled in each training epoch with a batch size of 100.

Speech Dataset

To test the model's performance described in this paper, the speaker identification run with the help of the VoxCeleb dataset which it's a public speech dataset, that consists of several English speeches recorded files with a high-fidelity microphone (44.1 kHz, 16-bit) divided into sections (training set, validation set, test set). The recorded files were sampled down to 16 kHz to produce the dataset. And it contains speech files from different English accents.

Speech Signal Processing

The first step is how to get the voice signal, two methods could be used, the first one is to capture the signal from a microphone, and the second one is to upload a file into the system which contains the voice of the speaker, we will go with the second one where numbers of files uploaded to the system to achieve the desired goal.

The second step is analyzing the uploaded files to clear the signal and applying the feature extraction method to get the physical characteristics of each speaker. Modifying the signal values is the first elementary process that can be applied where the signal is modulated according to the mean since the main objective of this step is to reduce the effect of any continuous frequency [36] produced by the recording devices. This processing step modifies the frequencies slightly as it does not change the shape of the signal, where a certain threshold is determined from the mean and subtracted from the signal values.

Dataset	Speakers	Utterances	Trails
VoxCeleb	1251	145375	579818

Table 1. VoxCeleb dataset information

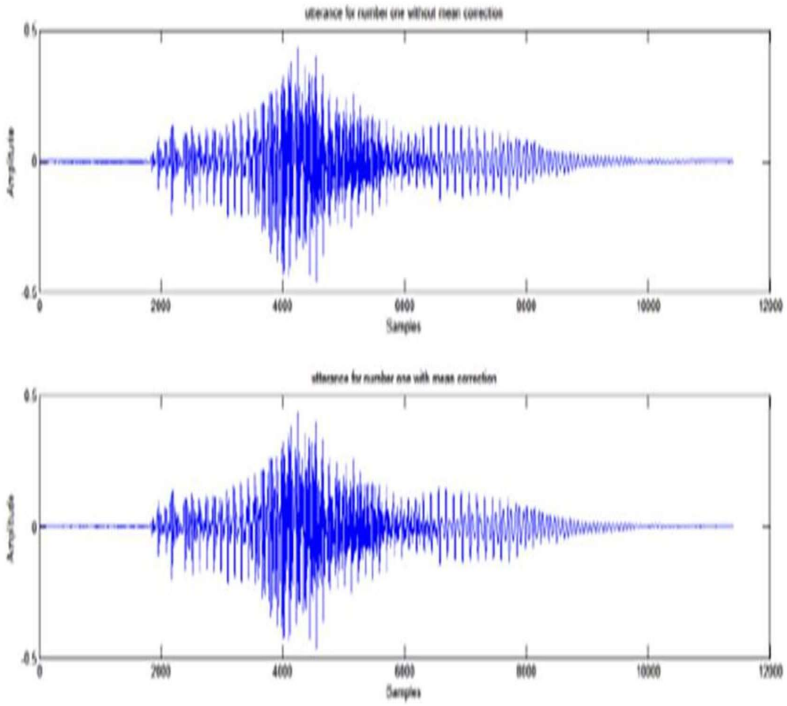


Figure 8. Number one with a threshold of 0.05

As shown in the previous figure, there is no difference between the two signals because, as we mentioned before, the purpose of this step is to reduce the effect of any continuous frequency. Another thing that can affect the quality of the audio samples is the moments of silence before or after the recording session, and these moments of silence can change the quality of the speech contents. So, these moments of silence (static samples) are removed from the signal. Removing the static samples from the signal isolates samples that contain speech information, Therefore, voice active detection (VAD) is used to give us the ability to distinguish between sounds and silences, and one of the most widely used techniques that serve this purpose of edge detection algorithms is short-term energy measurement [37] as this method depends on the signal energy.

$$E_{log} = \sum_{i=1}^{n} log(s(i)^2)$$

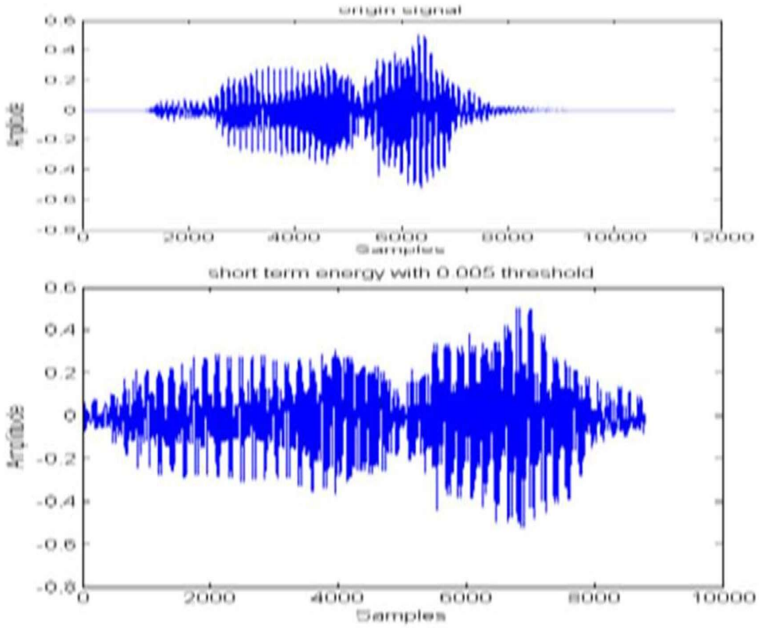


Figure 9. The signal after cutting the silence edges

MFCC (Mel Frequency Cepstral Coefficient) is used for the process of extracting the features of each speaker. The main goal of this step of processing is to obtain the spectrogram which contains special features of the speaker's voice, to obtain the spectrogram the FFT method must be performed (an inner step in the MFCC method) at a sampling frequency 16kHz and the number of FFT points is 256 after framing and windowing the signal [20], the FFT is the computing of the discrete Cosine Transform (DCT) of the given signal over the overlapping windows blocks, which enables tracking the characteristics [16] of the signal.

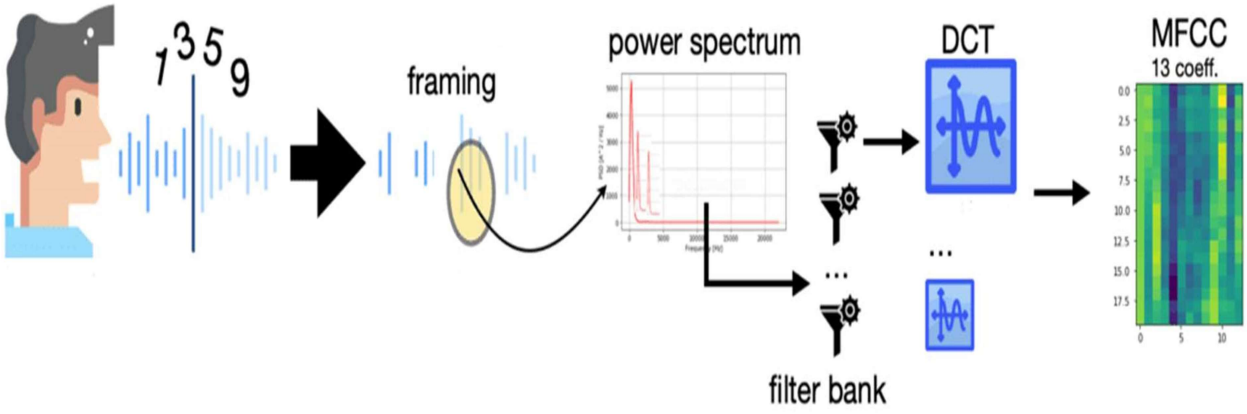


Figure 10. MFCC process (The main goal of the preprocessing is to obtain a spectrogram) [26]

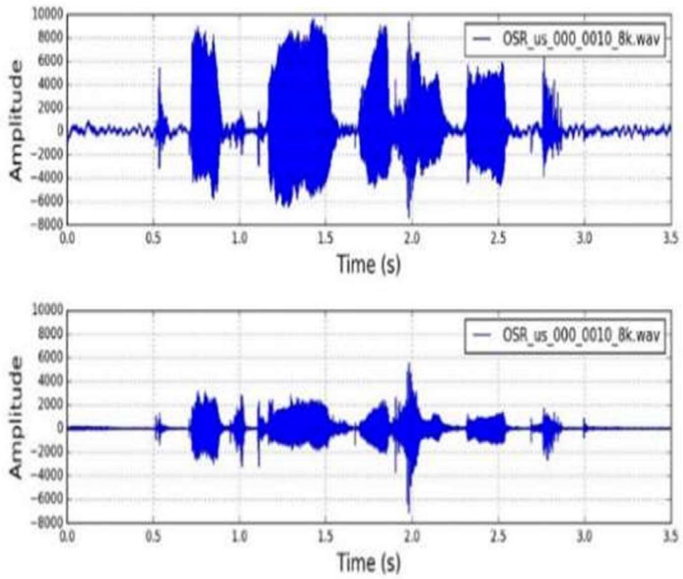


Figure 11. The signal before and after Pre-emphasis

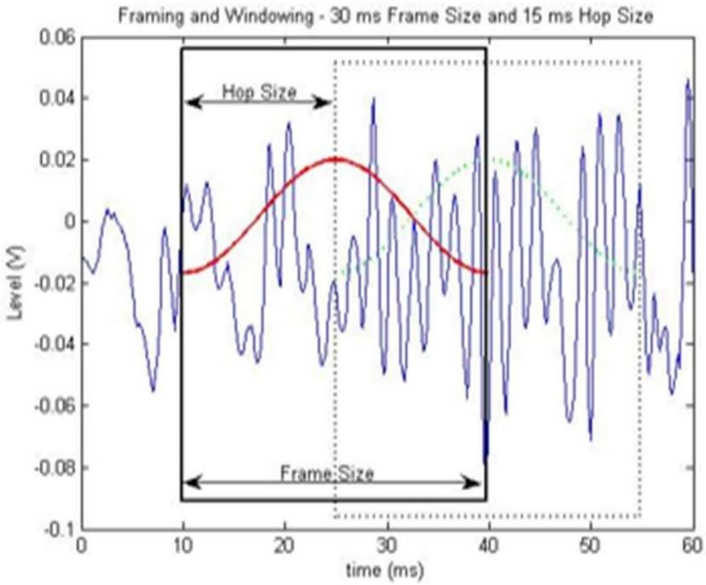


Figure 12. Framing & windowing

Firstly, the signal is filtered by equilibrating the whole frequency (pre-emphases) where it raises the high frequency against the low frequency to increase the predictability of the sounds [16].

Then the signal is divided into a small frame (frame size) of (25 - 30 ms) and overlapping (hope size) by (10 - 15 ms) which enables tracking the characteristics [16] of the signal and then windowed to minimize the distortion and prevent losing spectral energy that could be made on the signal by the previous step (divide the signal) by using the Hamming window.

Then the FFT (Fast Fourier Transform) is applied to convert the signal from the time domain into the frequency domain [17] to generate the feature vector. The FFT of the filtered and processed voice signal is calculated by the following:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi nk}{N}} \qquad k = 0,1,2\ldots,N-1$$

Where x: the framed signal with n as

n: the sequence number in the frame,

N the frame size.

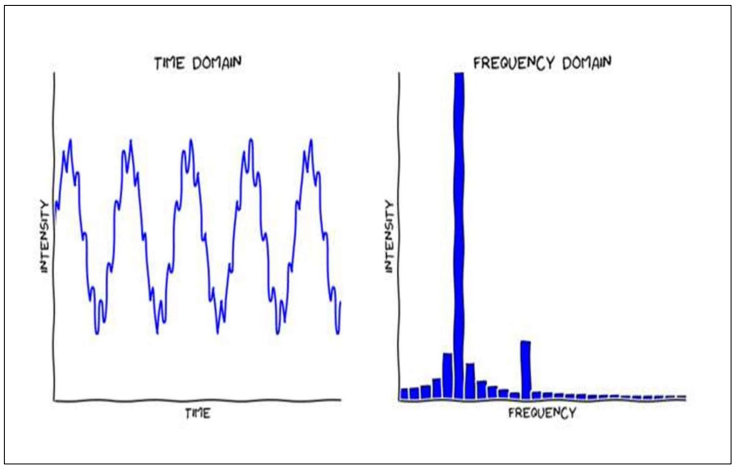
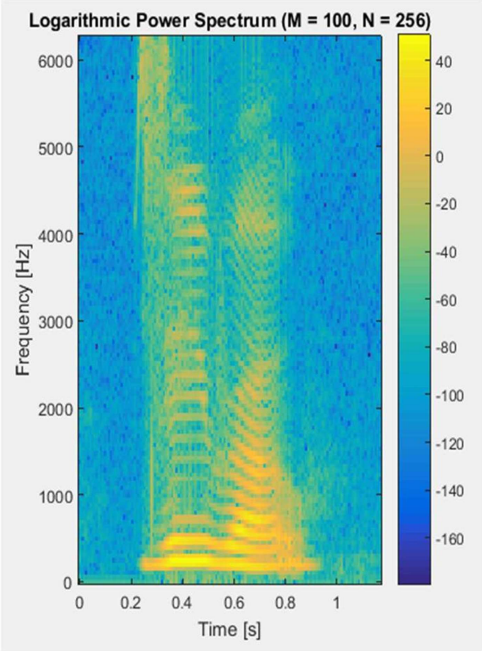
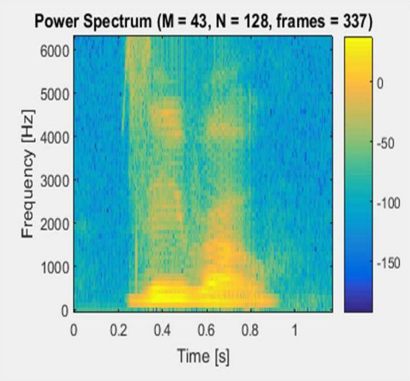


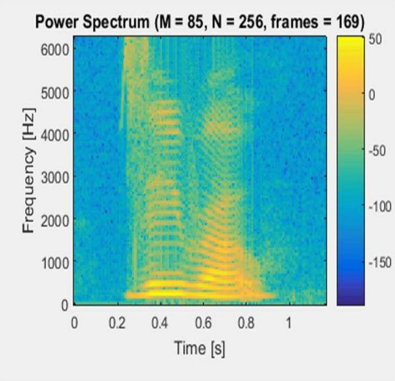
Figure 13. Fast Fourier Transform

After converting the signal to the time domain we got the advantage of having a lot of information about the signal which enables us to figure out the initial state of the signal (voice) and draw its spectrogram, The power spectrum is defined as:

$$P(k) = |X(k)|^2$$







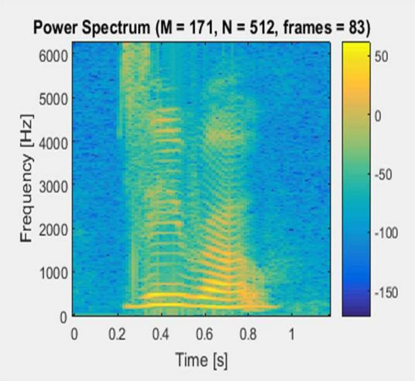


Figure 14. Spectrogram of speech

As we can see in the above figure the spectrogram contains information about the signal which is displayed in time (x-axis) and frequency domains (y-axis) at the same time, it shows the change of the signal spectrum over time, the x-axis is the time domain, the energy of the signal of different frequency can be seen in the figure, the spectrogram represents the most basic form of the voiceprint [21] that shown by the different degrees of colors in the figure.

Then we can get the Mel filter (Mel scale) by passing the FFT signal to the Mel filter bank [22], After that step, the DCT (Discrete Cosine Transform) is applied to get the MFCC features (32 features, including 40 log of the distribution of energy coefficients) along with the first and second temporal derivatives that represent each speaker with zero mean after normalizing each vector dimension and save it into the system environment then these values are sent into the convolutional layer after that we can build our model which is trained through iterative computation.

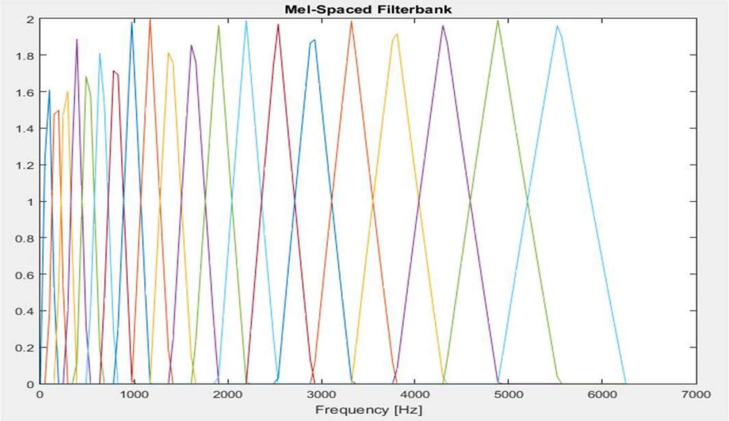


Figure 15. Mel-filter bank

The following figure shows the whole process of MFCC:

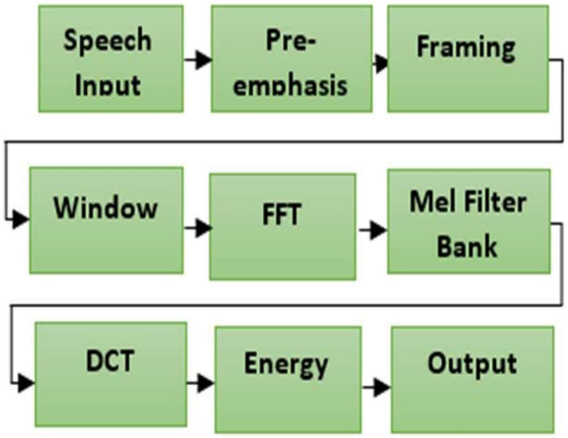


Figure 16. MFCC Block diagram

Enrollment and Training

The third step is to perform the learning method where convolutional neural networks (CNNs) that are described in the previous section are used to perform the task of verification (learning and testing) where CNN was initially used in the field of visual recognition tasks when it proved its ability to recognize faces and achieved high accuracy, it was widely used in the field of natural language processing (NLP) [18].

In this step, the result of the previous step of extraction of the voice features is passed as an input to the CNN model for further processing, as mentioned before the CNN consist of several convolutional layers with pooling layers and fully connected layers, and the input is convolved by the kernels (convolution filters) that connected the convolutional layers where the kernel split up the input signal into smaller blocks (receptive domain) which represent the core of the kernel.

In speech recognition, the pooling filters share the same

weight that is attached to the convolution filters and that is called limited weight sharing (LWS), at the start of the training, the weight in the model was initialized randomly and updated through the process steps, this technique is used because the voice signal changes across different frequencies and has special feature pattern in different filters.

To obtain the convolution results a multiplication operation is made by multiplying the values of the kernel by the identical values of the input that are in the receptive domain. The dropout layer is applied in the CNN layers along with the dense layer to address the difficult problem of overfitting in the training phase of the model which is assigned the value of 0.05 for the normal CNN layer and 0.20 for the LTSM layer (recurrent dropout) version of the model. The MFCC features are organized in 2-dimension (vector dimension) where the x-axis and y-axis represent the frequency and time domains and that is called the feature map. when applied convolution and pooling operations, in the upper layers the size of feature maps becomes smaller, in between the input and the output layers there are a lot of hidden layers added (fully connected layers) where its work is to combine the features across all frequencies band before pass it to the output layer.

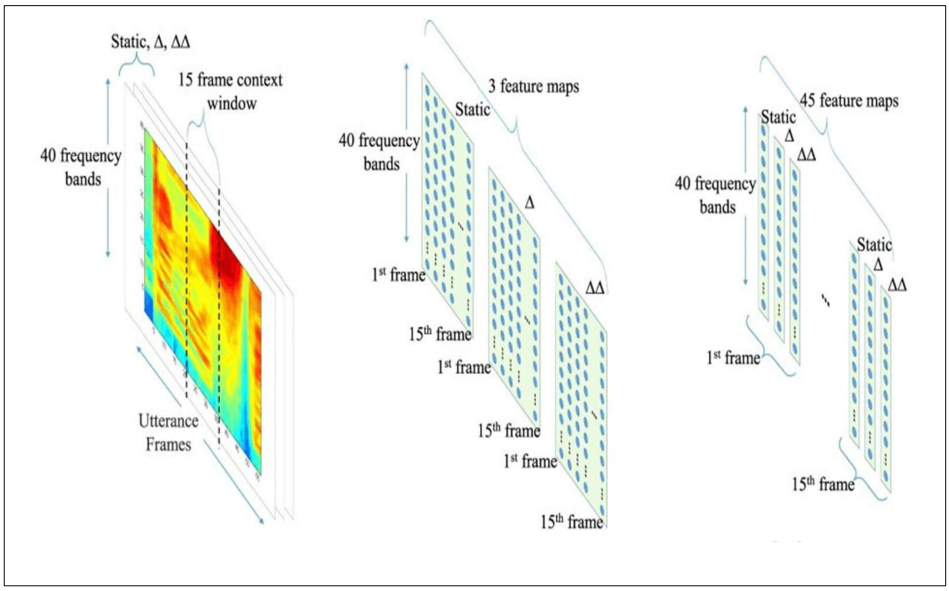


Figure 17. Feature map for speech recognition [19]

If the input feature map is $y = y_1 \ldots y_i$ that is connected to many features map (CNNs metrics) $Z = z_1 \ldots z_j$ and w is the local weight. The convolution operation is defined as:

$$Z_j = \sigma\left(\sum_{i=1}^{I} Y_i w_{i,j}\right), \quad j \in (1, J)$$

5. Experiment Results

In the figure below the training and testing results are shown to verify the ability of the learning method for both accuracy and loss rate for one training and testing process which shows the proposed model before using the LSTM technique:

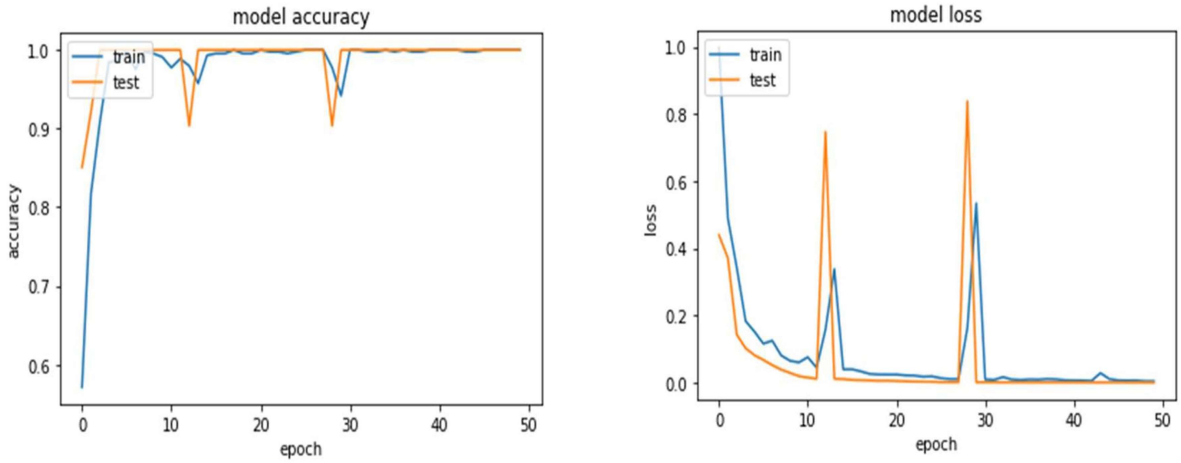


Figure 18. Loss and accuracy of the DeepSF

The table below (Table 2) shows some of the epochs values that were generated during the training and testing . phases which shows the loss and accuracy rate validated by the system

#	Loss	Accuracy	Validated loss	Validated accuracy
1	0. 6923	0.5714	0.4387	0.8509
2	0.4886	0.8170	0.3706	0.9211
3	0.3420	0.9085	0.1426	0.9385
4	0.1827	0.9844	0.1020	0.9852
5	0.1513	0.9866	0.0813	0.9946
6	0.1157	0.9978	0.0672	0.9993
7	0.1248	0.9754	0.0519	0.9864
8	0.0808	0.9978	0.0385	0.9926
9	0.0648	0.9955	0.0288	0.9979
10	0.0599	0.9911	0.0190	0.9994

Table 2. Loss and accuracy for some epochs in the model

Table 3 reports the overall accuracy of different models units and output shapes being tested during the process of verification and validation, the reported accuracy ranged from 96.64% to 98.38% the proposed model archived as follows:

Layer (type)	Units	Output Shape	#Param	Overall accuracy
LSTM	64	(None, 128, 64)	22272	98.38%
LSTM	32	(None, 32)	12416	96.64%
Dense	2	(None, 2)	66	96.89%

LTSM with units 64 archived the highest rate at 98.38%

Table 3. Performance comparison with several models

The figure below shows an improvement after using the LTSM model in both phases (training and testing) and the ability of the model had updated the accuracy of verification and validation for loss and accuracy models:

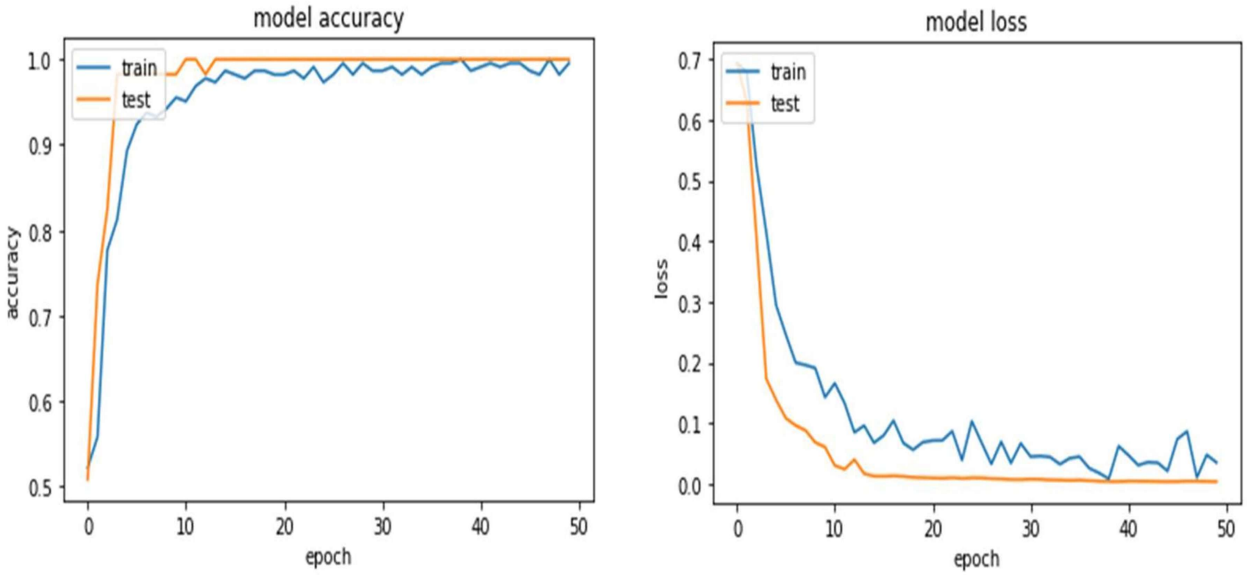


Figure 19. Loss and accuracy of the DeepSF with LSTM

Table 4 shows the improvement that happened to the loss which reduced from 0.69237 to 14271 affected the accuracy rate over the model which increased by about 3% over the second version which did not use the LTSM model:

#	loss	loss improved
1	0.69237	0.67958
2	0.67958	0.52432
3	0.52432	0.41415
4	0.41415	0.29453
5	0.29453	0.24562
6	0.24562	0.19957
7	0.19957	0.19571
8	0.19571	0.19077
9	0.19077	0.14271
10	0.14271	0.14271

Table 4. improvement Loss for some epochs in the model

Table 5 shows the Overall Accuracy and loss values generated in the represented model after the training, test, and validation phases of the given speeches signals:

#	Loss	Accuracy
Train model	0.0274753491726837	0.9664357142857143
Test model	0.0035152380199482045	0.9838999142857143

Table 5. Overall accuracy rate in training and test phases

6. Conclusion

In this paper, a speaker verification model has been briefly introduced. To achieve the desired goal of the system several techniques have been used as a core of the model. Where the input of the system is a voice signal (continuous signal over time) that needs to be processed before getting to start on the model, so the first step is to process the incoming signal by several methods discussed in detail in the previous sections. MFCC (Mel Frequency Cepstral Coefficient) and CNN (Convolution neural network) with the help of RNN-LSTM to deal with a sequence of data (speech signal) all used together to extract the voice features and train the model on them. Moreover, both gave good performance and accuracy results as shown in the result section. MFCC is used to process the signal and extract the voice spectrogram to introduce it to the CNN model as a bank that contains a lot of information about the speaker's voice (spectrum, pitch, tone, formant, Intensity, Frequency modulation, Group Delay, etc.), then the CNN model gets the advantages of this information to generate the new Deep Learning Speaker Features (DeepSF) where The important part is to learn and understand the distribution of features and vocal intensity of the speaker, which itself is a characteristic of the speaker, i.e. it is based on learning how these features were distributed in the speaker's audio clips. Then the algorithm converts each feature of the MFCC into a deep learning feature corresponding to DeepSF. DeepSF represents a new pure form of the previous features extracted by the MFCC algorithm. DeepSF is used to build the feature map which will be used by the kernel of the model to build the required knowledge network that constitutes an understanding of the sound patterns of the speaker's voice and minimize the map by deleting unnecessary data to increase the recognition rate of the model and decrease the required learning time and the amount of the audio samples where LTSM used to prevent over learning the network for the speech signal. The final output is the result of the SoftMax activation function that classified the speaker voice to use when the model is used later for verification. The overall accuracy rate of the system is archived at 98% in the test phase. The model has good performance in short and middle duration of speaker recognition which is considered good performance for such systems. Even though the model achieved high accuracy for short and middle speech duration but it takes a long time for processing a long speech duration under a complex noise environment and the performance is not that good, to overcome this situation and adapt the model to work under a long noisy environment many points need to be done:

• Optimize the model to learn the context of the long speech duration and deal with it which increases the accuracy of the recognition rate.

• Improve the noise elimination methods in the preprocessing phase by incorporating new skills which play an important part in the recognition rate and have a direct effect on the final result of the model's accuracy.

Authors' declaration

- **Conflicts of Interest:** None.

- We hereby confirm that all the Figures and Tables in the manuscript are mine/ours. Besides, the Figures and images, which are not mine/ours, have been given permission for re-publication and attached to the manuscript.

- The author has signed an animal welfare statement.

- Ethical Clearance: The project was approved by the local ethical committee at the Syrian Virtual University.

References

[1] Speaker independent connected speech recognition, Retrieved from Fifthgen. Fifthgen.com (15 June, 2013).

[2] Bimbot, F.J., Bonastre, F., Fredouille, C., Gravier, G. & Magrin-Chagnolleau, I. (2004) A tutorial on text-independent speaker verification. EURASIP *Journal on Applied Signal Processing*, 430–451.

[3] Reynolds, D.A.A. (1992) Gaussian mixture modeling approach to text. Independent Speaker Identification.

[4] Kinnunen, T. & Li, H. (2010) An overview of text-independent speaker recognition from features to super vectors. *Speech Communication,* 52, 12–40.

[5] Torfi, A., Dawson, J. & Nasrabadi, N.M. (2018) Text-independent speaker verification using 3D convolutional neural networks, *IEEE International Conference on Multimedia and Expo (ICME)*, San Diego, CA, pp. 1–6.

[6] Jung, J.w., Heo, H.S., Kim, J.h., Shim, H.j. & Yu, H.j. (2019). Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification, arXiv Preprint ArXiv:1904.08104.

[7] Anand, P. et al. (2019). "*Few Shot Speaker Recognition using Deep Neural Networks*." arXiv Preprint ArXiv:1904.08775.

[8] Vélez, I., Rascon, C. & Fuentes-Pineda, G. (2018). "One-Shot Speaker Identification for a Service Robot using a CNN-based Generic Verifier", arXiv Preprint ArXiv:1809.04115.

[9] Mostafa, E. (2019). Advanced Intelligent Systems for Sustainable Development. Springer: Berlin, (AI2SD 2018).

[10] Rudresh, M.D., Latha, A.S., Suganya, J. & Nayana, C.G. Performance analysis of speech digit recognition using cepstrum and vector quantization. *IEEE, Computer and Optimization Techniques* (ICEECCOT) (15 Dec 2017).

[11] Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N. & Kingsbury, B. (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine. *IEEE Publications*, 29, 82–97.

[12] Glorot, X. & Bengio, Y. (2010) Understanding the difficulty of training deep feedforward neural networks. *In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics,* pp. 249–256.

[13] Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. (1998) Gradient-based learning applied to document recognition. Proceedings of the IEEE. *IEEE Publications*, 86, 2278–2324.

[14] https://www.mathworks.com/discovery/convolutional-neural-network-matlab.html.

[15] "A Hybrid of Deep CNN and Bidirectional LSTM for Automatic Speech Recognition". (degruyter.com).

[16] Mostafa, E. (2019) "Advanced Intelligent Systems for Sustainable Development", springer. Available from: https://www.springer.com/gp/book/9783030119270.

[17] Karpov, E. (2003). Real-Time Speaker Identification [University of Joensuu, Department of Computer Science Master's Thesis].

[18] https://theaisummer.com/speech-recognition/convolutional-models.

[19] Abdel-Hamid, O., Mohamed, A.-r., Jiang, H., Deng, L., Penn, G. & Yu, D. (2014) Convolutional neural networks for speech recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22, 1533–1545.

[20] Wen-kai Lu & Qiang Zhang (2009) Deconvolutive short-time Fourier transform spectrogram. IEEE Signal Processing Letters, 16, 576–579.

[21] Li, B. (2011) On identity authentication technology of distance education system based on voiceprint recognition. In: *Proceedings of the 30th Chinese Control Conference*, Yantai, China, Vols. 22–24, pp. 5718–5721.

[22] Gowdy, J.N. & Tufekci, Z. "Mel-scaled discrete wavelet coefficients for speech recognition", in Proceedings of the 2000 IEEE International Conference on Acoustics [Speech], and Signal Processing. *Proceedings (Cat. No. 00CH37100). Istanbul, Turkey* (5–9 June 2000), Volume 3, pp. 1351–1354.

[23] Huangkang, C. & Ying, C. (2019) Speaker identification based on multimodal long short-term memory with depth- gate, *[J]. Laser and Optoelectronics Progress*, 56, 031007.

[24] Miao, X. & Mcloughlin, I. (2019) Multi- Genre Broad-

cast Challenge*[J] LSTM-TDNN with convolutional front-end for Dialect Identification*, in the, 2019.

[25] https://medium.com/techiepedia/binary-image-classifier-cnn-using-tensorflow-a3f5d6746697.

[26] https://www.gosmar.eu/machinelearning/2020/05/25/neural-networks-and-speech-recognition/.

[27] Shan, Shuaijie, Liu, J. & Dun, Y. (2021) Prospect of voiceprint recognition based on deep learning. *Journal of Physics: Conference Series*, 1848, 012046.

[28] Tucci, L., Lutkevich, B.," A guide to artificial intelligence in the enterprise: natural language processing (NLP). Tech Accelerator (09 Jul 2021).

[29] Farrell, K.R., Mammone, R.J., Assaleh, K.T. (1994) Speaker networks recognition using neural and conventional classifiers. *IEEE Transactions on Speech and Audio Processing*, 2, 194–205.

[30] Martinez, J., Perez, H., Escamilla, E. (2018). Speaker Recognition Using Mel Frequency Cepstral Coefficients (MFCC) and Vector Quantization (VQ) Techniques, Vol. 4. *IEEE Publications*, pp. 978–971-61284-1325-5/12/.

[31] Ma, Z., Yu, H., Tan, Z.H. & Guo, J. (2016) Text-inde pendent speaker identification using the histogram transform model. *IEEE Access*, 4, 9733–9739

[32] Almaadeed, N., Aggoun, A. & Amira, A. (2015) Speaker identification using multimodal neural networks and wavelet analysis. *IET Biometrics*, 4, 18–28.

[33] Emre, C.¸ akir, Giambattista Parascandolo, Toni Heittola (2017) Heikki Huttunen, and Tuomas Virtanen, Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25, 1291–1303.

[34] Ranjan, Kumari R., Mahto, K., Kumari, D. & Solanki, S.S. (2017) Singer Identification using MFCC and LPC and its comparison for ANN and Naïve Bayes Classifiers. *International Journal of Latest Engineering Research and Applications (IJLERA)*, 02, (I04): PP – 25-30.

[35] Singh, T. (2019) MFCC's made easy. https://medium.com/@tanveer9812/mfccs-made-easy-7ef383006040.

[36] Pejman, M., Ku, J., Johannes, S. & Florian, M. (2016) Single channel phase-aware signal processing in speech communication. *Theory into Practice*. Wiley: Chichester, UK, 53–55.

[37] Verteletskaya, E., Sakhnov, K. (2010). Voice Activity Detection for Speech Enhancement Applications. *Acta Polytechnica*, 50 No. 4-2010.