Comparative Study of Different Types of RNN in Speech Classification

Ayat N. Ragheb [*1], Amr M. Gody [*2], Tarek M. Said [*3]

[*]*Electrical Engineering Department, Fayoum University, Fayoum, EGYPT*

[1]an1162@fayoum.edu.eg

[2]amg00@fayoum.edu.eg

[3]tms02@fayoum.edu.eg

Abstract: This paper introduces different pre-processing classification models and their performance in the Automatic Speech Recognition system. Other Recurrent Neural Network (RNN) architectures have been tested for this problem, such as RNN cells (RNN), bidirectional RNN (BRNN), Long Short-Term Memory (LSTM), and bidirectional LSTM. Mainly, two features have been considered. First, Mel Frequency Cepstral Coefficient (MFCC) plus delta and delta-delta coefficients (39 parameters) have been used. Second, MFCC quantization using Vector Quantization technique has been used as features. All models have been trained on TIMIT database. Vowels, nasals, fricatives, plosives, and silences have been chosen as syllable classes for classification. Experiment results show that BRNN-MFCC-5- {30,30,20,25,25} system give the highest accuracy. It achieved 92.6%. In similar work of using RNN in classification, 83% accuracy was achieved by [1], and 95% had been achieved by [2]. It is also noticeable that the results obtained by using HMM in a similar problem are 80% by[19] and 81.01% by [17].

Keywords: ASR, Classification technique, RNN, BRNN, LSTM, BLSTM, MFCC, Vector Quantization.

1 INTRODUCTION

For people, the most important and effective communication method is voice, which is used to communicate together. People are very comfortable with speech; hence, people would also like to connect with PCs via speech rather than using keyboards and pointing devices. This can be achieved by establishing an automatic speech recognition system (ASR) that allows the computer to distinguish words spoken by a person on a microphone or phone and convert them into written text. Subtask of syllable classification has been focused because the improvement was believed that it would lead to an improvement in the overall performance of the recognition system. This research is providing a comparative study of different classification methods used to increase the accuracy of ASR.

A more direct model has been used in which the Hidden Markov Model (HMM) is exchanged with one of the RNN architectures (RNN, BRNN, LSTM, and BLSTM); that perform sequence classification directly at the syllable level. The transcription between the features and the required syllable sequence is automatically learned by RNN mechanisms. For each classified syllable, the RNNs scan the input and choose relevant frames. The features used for training the models are MFCC plus delta and delta-delta coefficients (39 parameters) and MFCC quantization with one component for each frame.

The paper is structured as the following steps: Literature survey of relevant topics has been provided in section 2. Section 3 presents the proposed models and features for syllable classification. The database and the experiment are presented in section 4. Results are presented and discussed in section 5. Finally, the concluded is in section 6.

2 LITERATURE REVIEW

For the last six decades, the speech recognition field was an active area of research. The most common methods for improving ASR systems use HMMs [3], feedforward neural networks (FFNNs) [4], hybrid systems using a combination of HMMs and FFNNs [5], and deep neural networks (DNNs) [6], [7], [8]. From the literature review, there are some studies exploring methods to improve the phone recognition classification accuracy.

P. Karjol et al. in [9] presented an algorithm for speech enhancement by a broad phoneme classification using a specific deep neural network. Phonemes have been classified into vowels and non-vowels. TIMIT corpus has been used in the experiment. Results of experiments have shown that the specific deep neural network outperformed the single DNN based speech enhancement with an accuracy of 94.1%.

Christos Antoniou in [10] has proposed a new design for a broad classification by the modular neural network where the observation vector was not fixed in size. Phones have been divided into seven classes (vowels, plosives, fricatives, nasals, diphthongs, semi-vowels, closures). TIMIT database has been used in research. The features that were used were MFCC with 32 coefficients. The accuracy result was 84.1%.

P. Scanlon et al. in [11], have presented a new approach with a neural network multilayer perceptron (MLP) classifier that contains a modular order of experts. Phonemes have been divided into seven classes (vowel, semi-vowels, diphthongs, stops, fricatives, nasals, silence). Features that were used were perceptual linear predictive (PLP). The experiment has been running on a TIMIT database. The best result of accuracy was 74.2%.

W. Rochkitticharеon et al. in [12] have proposed broad phonetic class research for the ASR system. Phones have been divided into five classes (silence, vowel, sonorant consonant, fricative, stop). The continuous speech corpus (LOTUS) has been used in research. It used acoustic parameters that extract the characteristics of each broad manner class. Support Vector Machine Classifier has been used in the study. The best result of accuracy was 80.46%.

T. Jeff et al. in [13] presented a modular multilayer perceptron (MLP) design for acoustic models on broad phoneme classes. Phonemes were divided into seven classes; (vowel, semi-vowels, diphthongs, stops, fricatives, nasals, and silence). Three types of feature techniques have been used in the research: MFCC, PLP, and linear prediction derived cepstrum (LPC). TIMIT database has been used in experiments. The best classification accuracy result was 84.1%.

A. Chittora et al. [14] proposed a phoneme classification method for classifying the phonemes of the Gujarati language by using a modulation spectrogram as feature extraction. The phonemes have been classified by using the Support Vector Machine (SVM) as a classification model. The phonemes have been divided into six classes (vowels, semi-vowels, affricates, fricatives, stops, nasals). The best result of accuracy was 95.70 % when using the proposed features with MFCC features.

G. Deekshitha et al. [15] proposed a classification model using new features extracted at present from a speech signal. Then, a comparison between these features and MFCC features have been proposed. Phone classes were vowels, nasals, fricatives, stops, approximants, and silence. A classifier used in this research was Multilayer feedforward neural network. The accuracy of classification has improved when a combination of the proposed features and MFCC features was used.

M. Aissiou et al. [16] have presented a genetic algorithm on phoneme classification for ASR system. MFCC features have been used in research. Phonemes have been classified into five classes. Experiments have been carried out on an Arabic normalized database. The accuracy result was 90.20%.

G. Kiss et al. [19] have proposed an approach that depends on the segmentation of speech into nine broad classes using the Hidden Markov Model (HMM) in the classification process. Features used in the research were extracted by Bark-scale spectral resolution. The databases used in experiments were KIEL, MRBA, and TIMIT. The accuracy result average was 80% by TIMIT database,83% by MRBA database, and 78% by KIEL database.

M. Antal in [20] has used the Gaussian mixture model (GMM) to improve phone classification. TIMIT database has been used in research. Phone classes were vowels, semi-vowels, affricates, stops, nasals, and fricatives. The feature extraction technique used in the study was MFCC. Results have shown that vowels and nasals gave high identification rates.

Doaa N. Senousy et al. [17] proposed a syllables classification approach for ASR using HMM's variable states. MFCC and Mel Best Tree (MBT) features techniques have been used. Phone classes were liquid, vowels, stops, plosives, nasals, and consonants. The database used in the research was TIMIT. The overall success rate was 81.01% for MBT features and 72.66% for MFCC features.

3 TASK FLOW OF SYLLABLES CLASSIFICATION

The proposed approaches for syllable classification are shown. Speech utterance has been sampled in 16 kHz. 20(ms) frame length has been chosen for analysis. After that, using MFCC with delta and delta-delta parameters in Figure 1 and MFCC with vector quantization in Figure 2, the speech features were extracted and applied to the classification module to classify the speech signal into five classes: vowel, nasals, Fricatives, plosives, and silence. The proposed approaches will be discussed in detail in the following subsections.

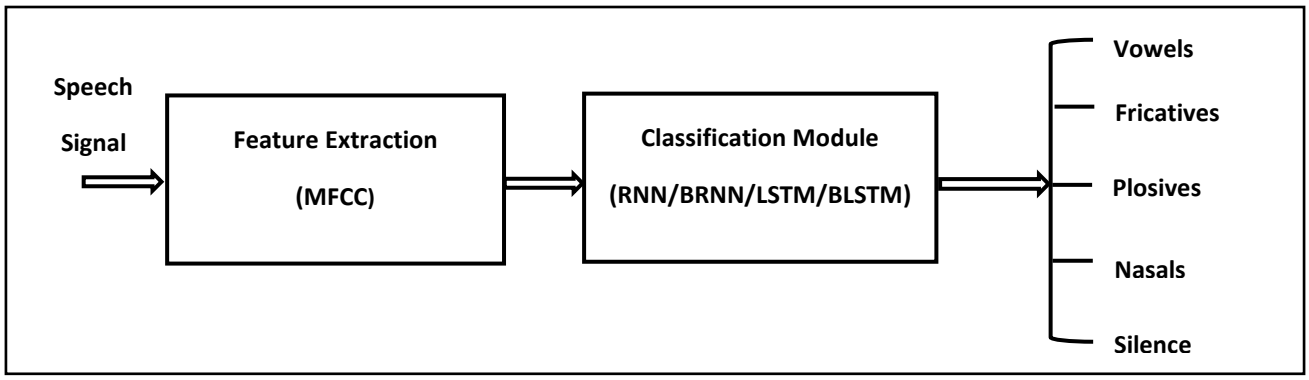

Figure 1: Block diagram of the proposed model

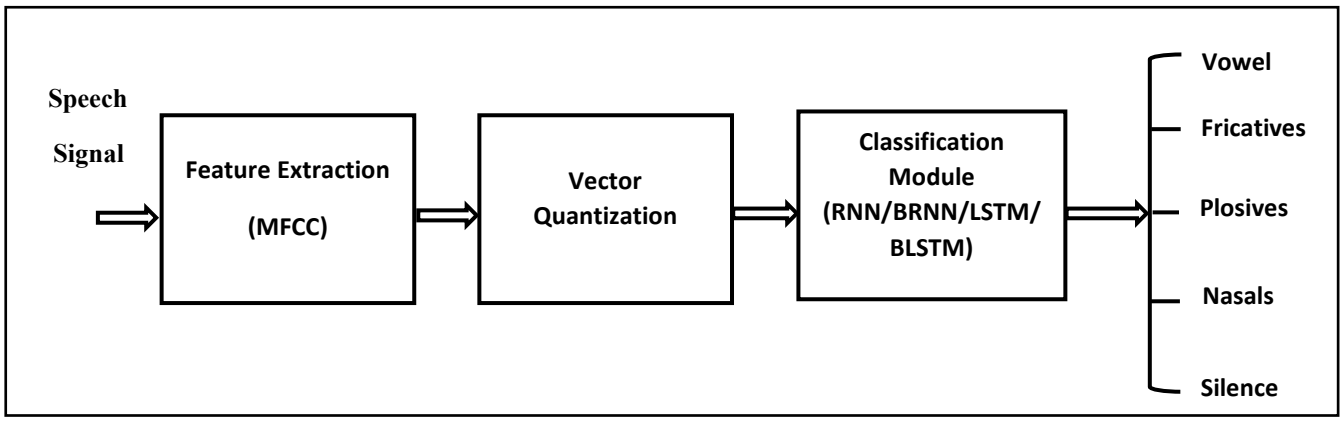

Figure 2: Block diagram of the proposed model with VQ

A. Feature Extraction

The purpose of feature extraction is to transform input data into a set of properties of an utterance with acoustic correlation to the speech signal, that is, parameters that can somehow be computed or estimated through the signal waveform processing. Such properties are termed as features. The four speech recognition models are based on the Mel frequency cepstral coefficient (MFCC) audio data into frames using MFCC vectors plus delta and delta-delta coefficients (39-parameters).

1) MFCC

MFCC technique can extract audio signals' efficient properties in terms of time domain and frequency domain [21]. Therefore, to use the MFCC feature extraction technique, the following steps should be applied. MFCC can be

executed in six steps: pre-processing, framing, Hamming windowing, Fast Fourier Transform (FFT), Mel bank filtering, and Discrete Cosine Transformation (DCT) stages as illustrated in Figure 3.

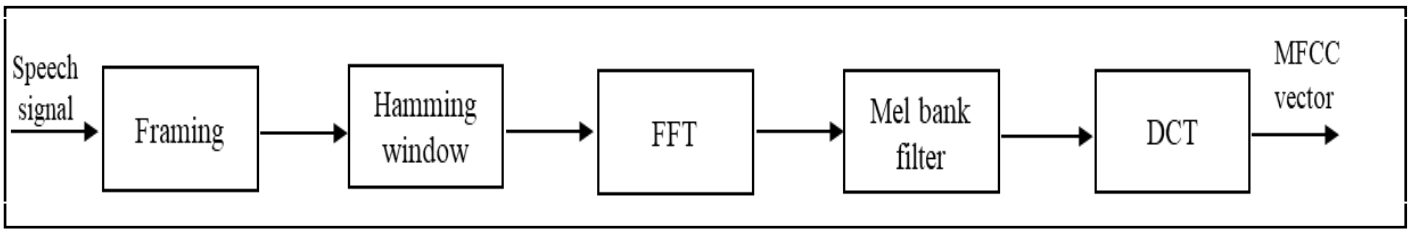

Figure 3: Block diagram of MFCC

In the framing step, the speech signal was segmented into short frames (n) of the length, varied between 20 to 40 (ms) to pass these frames to the Hamming windowing step. Consequently, Hamming windowing was responsible for creating a window shape by considering the next block of the feature extraction processing chain and integrating all the closest frequency lines. Thus, Hamming windows were computed based on equation (1) and equation (2) [22].

$$Y[n] = X[n] * W[n] \tag{1}$$

$$W[n] = 0.54 - 0.46 * cos(\frac{2\pi n}{N-1}) \tag{2}$$

where:

- X: The input signal.
- W: The Hamming window
- Y: The output.

In FFT step, the frame was converted of N samples from the time domain into frequency domain to preserve the convolution of glottal pulse and vocal tract impulse response h(t) in the time domain. Therefore, this step's computation was conducted based on equation (3) [22].

$$Y[w] = FFT\big(h(t) * x(t)\big) \tag{3}$$

Based on the FFT step results, the spectrum frequencies were very wide, and the voice signal does not follow the linear scale. Therefore, the Mel filter bank has been used to ease the conversion to get a Mel frequency signal that is appropriate for human hearing and perception. The Mel frequency was computed in this step based on equation (4) [22].

$$F(Mel) = \left[2595 * log_{10}\left[\frac{1+f}{700}\right]\right] \tag{4}$$

where:

- $F(Mel)$: Frequency on Mel scale.
- f: The frequency in hertz.

Then, the first order derivatives of MFCC (Δ-MFCC) and the second-order derivatives of the MFCC(ΔΔ-MFCC) are added, and these are also called differential (13-coefficients) and acceleration (13-coefficients). The Δ-MFCC coefficients are computed by using equation (5) [23]:

$$d_t = \frac{\sum_{\theta=1}^{\Theta} \theta(c_{t+\theta} - c_{t-\theta})}{2\sum_{\theta=1}^{\Theta} \theta^2} \quad (5)$$

where:

- d_t: The Δ-MFCC coefficient at time t.
- Θ: The window size of the delta.
- $c_{t-\theta}$, $c_{t+\theta}$: Static coefficients.

Then, the ΔΔ-MFCC coefficients are computed by differentiating equation (5).

2) Vector Quantization

Vector quantization is a mapping process between vectors that convert them from big vectors into limited vectors. Each area is named a cluster, which is centered by a codeword. The gathering of all codewords is called a codebook. The widely used LBG [Linde, Buzo, and Gray] algorithm is used to cluster L MFCC extracted vectors into a collection of M codebooks. LBG was performed by the following steps [24]: First, Determine the number of code-vectors N. Second, select N code-vectors at random to be the initial codebook. Third, Using the Euclidean distance measure, cluster the vectors around each code-vector. Fourth, compute the new set of code-vectors (codebook). Fifth, iterate steps 2 and 3 till either of the representative code-vectors do not change.

B. Classification Module

In this research, four classifiers were focused on; (RNN, BRNN, LSTM, and BLSTM). These models were proposed in recent years as an alternative approach to speech recognition systems. That is due to their impressive ability to link input features and improve class discrimination. A brief overview of these approaches has been presented in the next subsections.

1) Recurrent Neural Network Model

RNN model is a type of neural network whose feedback enables them to keep information from the past; this makes RNN a suitable classifier for the speech signal. In Figure 4, block A represents a neural network that takes input x_t at the current time t and provides the value h_t as an output. The loop shown in the structure allows using information from past time to produce output for the present time t. Therefore, the result at time t-1 affects the result at time t. The network response to new data relies on the current input and the output from the recent data. The RNN output calculation is based on iteratively calculating the production of equations (6) and (7) [25]:

$$h_t = \mathcal{H}(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (6)$$

$$y_t = W_{hy}h_t + b_y \quad (7)$$

where:

- h_t: Hidden vector at time t.
- x_t: The input sequence at the current time t.
- y_t: The output sequence at time t.
- W: Weight vector.
- b: Bias vector.
- $\mathcal{H}$: Activation function of the hidden layer.

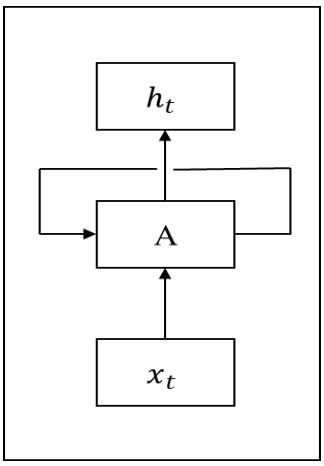

Figure 4: Feedback structure of RNN [21]

2) Bidirectional Recurrent Neural Network Model (BRNN)

In BRNN, data is processed in both directions (forward and backward) using two splitting hidden layers connected to the same output layer [6].

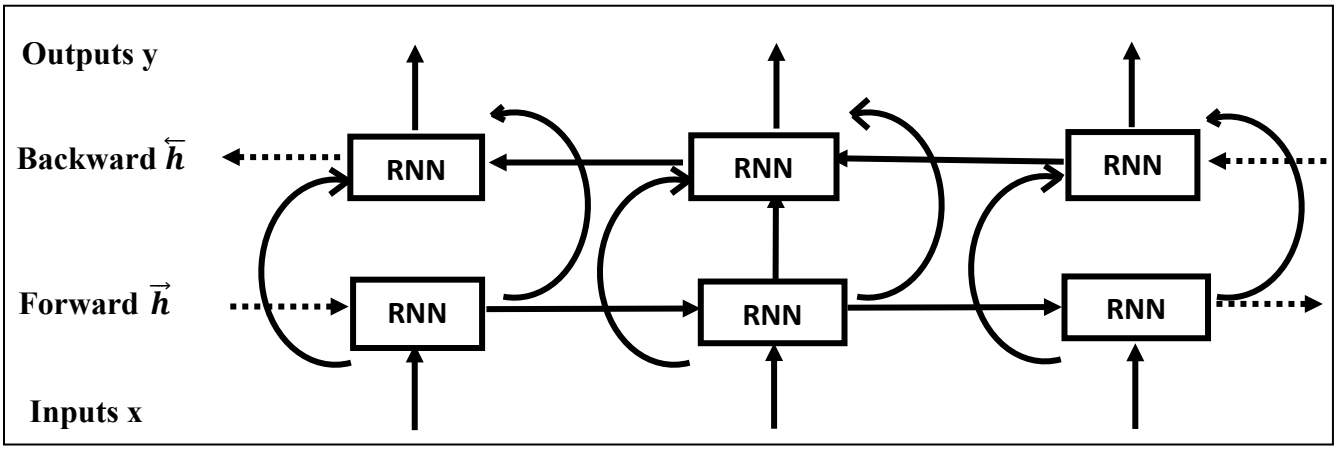

Figure 5: BRNN [6]

As illustrated in Figure 5, BRNN calculates the forward sequence $\vec{h}$, the backward sequence $\overleftarrow{h}$ The output y by duplicate the backward layer from $t = T$ to 1, and the forward layer from $t = 1$ to T. Then update the output layer [6]:

$$\overrightarrow{h_t} = \mathcal{H}\left(W_{x\vec{h}}x_t + W_{\vec{h}\vec{h}}\vec{h}_{t-1} + b_{\vec{h}}\right) \tag{8}$$

$$\overleftarrow{h_t} = \mathcal{H}\left(W_{x\overleftarrow{h}}x_t + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{t-1} + b_{\overleftarrow{h}}\right) \tag{9}$$

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t + b_y \tag{10}$$

where:

- $\overrightarrow{h_t}$: Forward hidden sequence at time t.
- $\mathcal{H}$: Activation function of the hidden layer.

- W: Weight vector.
- b: Bias vector.
- $\overleftarrow{h_t}$: Backward hidden sequence at time t.
- y_t: Output vector at time t.

3) Long Short-Term Memory Model (LSTM)

LSTMs are a kind of RNNs with memory cells, which are essential in handling long-term temporal dependencies in data. Their default behavior is remembering information over a long period. LSTMs also deal with the problem of vanishing/exploding gradient during backpropagation [26]. Thus, they overcome both shortcomings that RNNs face.

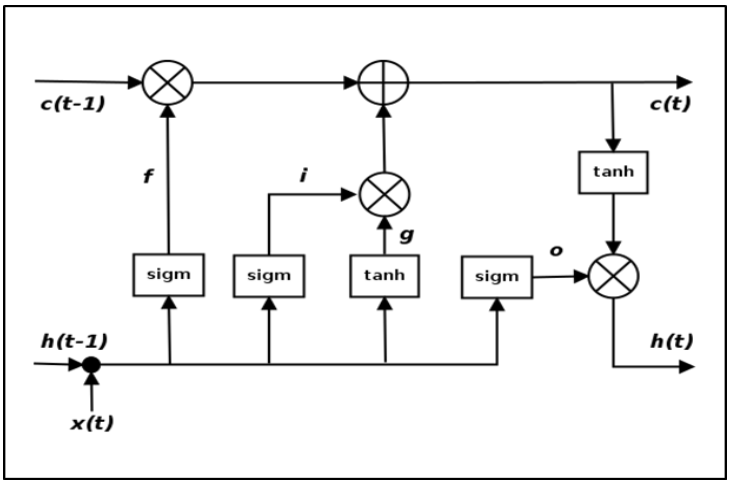

Figure 6: LSTM cell [26]

Figure 6 illustrates the design of the LSTM memory cell. Each big block here represents a memory cell. The cell status is an essential part of LSTM and is displayed in the figure by the horizontal line above the cell. It occurs from every cell in the chain of the LSTM network. LSTM has the choice to add or remove information from this cell state. Another structure in LSTM called gates performs this operation. As shown in Figure 6, there are three gates controlling information going through the cell state as below:

- Forget gate - determines what information to throw away.
- Input gate - determines what new information to save in the cell state.
- Output gate -determines what cell status information goes to the output.

Hochreiter and Schmidhuber first invented LSTM networks in 1997. The calculations of the LSTM cell can be done by the following equations [26]:

$$f_t = \sigma\big(W_f.[h_{t-1}x_t] + b_f\big) \tag{11}$$

$$i_t = \sigma(W_i.[h_{t-1}x_t] + b_i) \tag{12}$$

$$c_t = f_t * c_{t-1} + i_t * \tanh(W_c.[h_{t-1}x_t] + b_c) \tag{13}$$

$$o_t = \sigma(W_o.[h_{t-1}x_t] + b_o) \tag{14}$$

$$h_t = o_t * \tanh(c_t) \tag{15}$$

where:

- f_t: The forget gate.
- W_f: Weight of forget gate.
- x_t: Input at time t.
- b_f: Bias of forget gate.
- b_i: Bias of the input gate.
- b_c: Bias of memory cell content.
- b_o: Bias of output gate.
- h_{t-1}: Hidden vector at time t-1.
- σ: The sigmoid function.
- i_t: The input gate.
- o_t: The output gate.
- c_t: memory cell content

As mentioned earlier, three gates are made up of σ function, and the output of the specific cell is scaled up by using the tan$h$ function.

4) Bidirectional Long Short-Term Memory Model

BLSTM is a combination of BRNNs with LSTM [27] for accessing long-range context by processing input in both forward and backward directions, as in Figure 7.

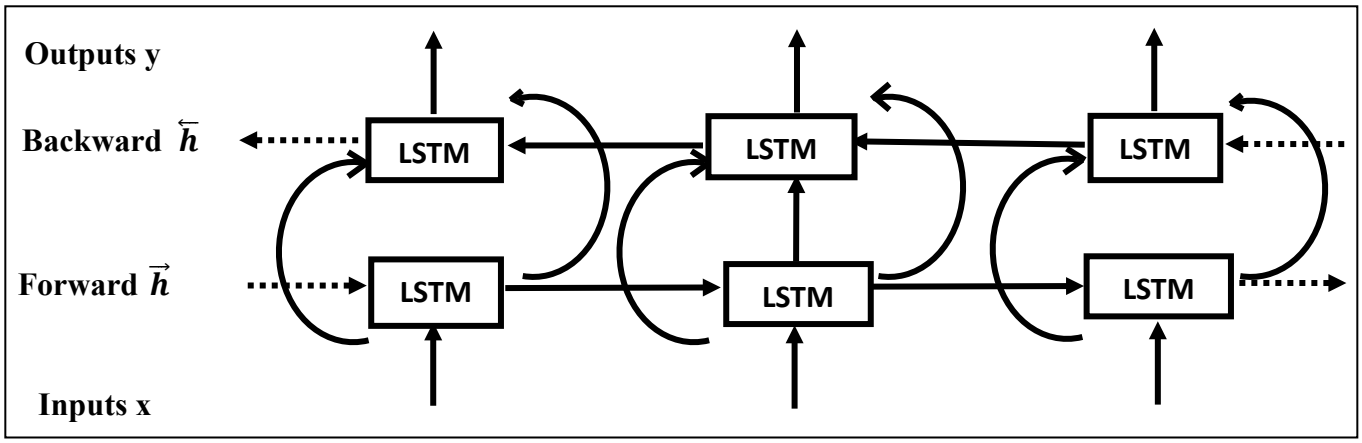

Figure 7: BLSTM [27]

4 EXPERIMENT ENVIRONMENT

A. *Corpus Description and Data Preprocessing*

All experiments were executed on the 16kHz TIMIT database. TIMIT has 630 speakers with 6300 utterances in 8 dialects. It includes sentences of prompted English speech, followed by full phonetic transcripts. It has a dictionary of 61 various phonemes. The training and test of TIMIT files include 4620 and 1680 audio sentences, respectively. All data has been pre-processed into frames using MFCC vectors plus delta and delta-delta coefficients (39 parameters) from a 20 ms window at a 20 ms frame rate. Database has been altered such that transcription files are suitable for the objective of this research. Vowels (V), Plosives (P), Fricatives (F), Nasals (N), and Silences (Si). Table 1 shows each classifier with phones assigned to it.

TABLE 1

PHONE CLASSIFIERS

Classifiers	TIMIT Labels
Vowels (V)	aa, ae, ah, ao, ax, ax-h, axr, ay, aw, eh, el, er, ey, ih, ix, iy, l, ow, oy, r, uh, uw, ux, w, y
Plosives (P)	p, t, k, b, d, g, jh, ch, bcl, dcl, gcl, pcl, tcl, kcl, q, dx
Fricatives (F)	s, sh, z, zh, f, th, v, dh, hh, hv
Nasals (N)	m, em, n, nx, ng, eng, en
Silences (Si)	h#, epi, pau

B. Model Hyperparameters

Common parameters used in RNN, BRNN, LSTM, and BLSTM models have been introduced. All the four models are accomplished by python using a deep learning library called Keras [28]. Keras is a high-level API for neural networks. Keras's advantage lies in its ability to quickly build a prototype of deep learning designs, which are easily modular, where layers are stacked and connected for the computational and extensible interface. It can run on CPU and GPU, supports convolution and recurrent layers, can add packages of machine learning, and it has been widely used by researchers and industrial companies in the past years. These models consist of common parameters, such as the optimization process used to train itself during all the experiments. The Root Mean Square Propagation (RMSProp) optimizer learning how to change parameter θ to reach a minimum loss function $J(\theta)$ with a fixed learning rate of 10e−4 gave the best result using try and error methodology. The batch size, which is the number of samples of the training database used to estimate the error before the model updates the weights, is 32. It is based on the research on [32] and the number of epochs, which is the number of times that the model takes for the learning process is 20; the chosen value has been obtained using the try and error methodology where after this number, there was no noticeable change in the result. In this research, the try and error approach has been applied to select the model's hyper-parameters.

For the output layers, the cross-entropy error function is used to measure the classification process's performance and the softmax activation function with five nodes, which are the classes used. The softmax function is used to calculate the probability distribution of a vector of real numbers, as shown in equation (16). It generates output in a range of values that lie between 0 and 1, and the sum of these probabilities is equal to 1. So, the softmax function is used when classifying data into more than two classes.

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}} \tag{16}$$

where:

- x: the input vector.
- x_i: the i^{th} element in the input vector x.
- e^{x_i}: The exponential of the element x_i in the input vector x.
- $\sigma(x_i)$: The output of the element x_i in the input vector x.
- $\sum_{j=1}^{K} e^{x_j}$: The summation of the exponential of all elements assures that all output values will sum to1.

C. Models Implementation

The experiment has been running on two types of features. Firstly, MFCC with 39 coefficients. Secondly, MFCC and VQ features with one component. Each of these features has been applied to each of the four classifiers (RNN, BRNN, LSTM, BLSTM). The four models were worked in task flow as below:

1. The experiment has been running on one hidden layer with different hidden units each time (10-15-20-25-30).
2. The best result of hidden units has been taken, adding the second hidden layer with different hidden units (10-15-20-25-30).
3. The best result of hidden units has been taken and adding the third hidden layer with different hidden units (10-15-20-25-30).
4. These steps were repeated until they reached five hidden layers. Figure 8 shows the flowgraph of these steps.

System with models, features, hidden layers and hidden units will be represented as (name of model-name of features-number of hidden layers- hidden units used in each hidden layer respectively). For example, RNN-MFCC-3-{10,20,25} means this system used RNN model, MFCC features and three hidden layers with 10,20,25 hidden units in each layer, respectively.

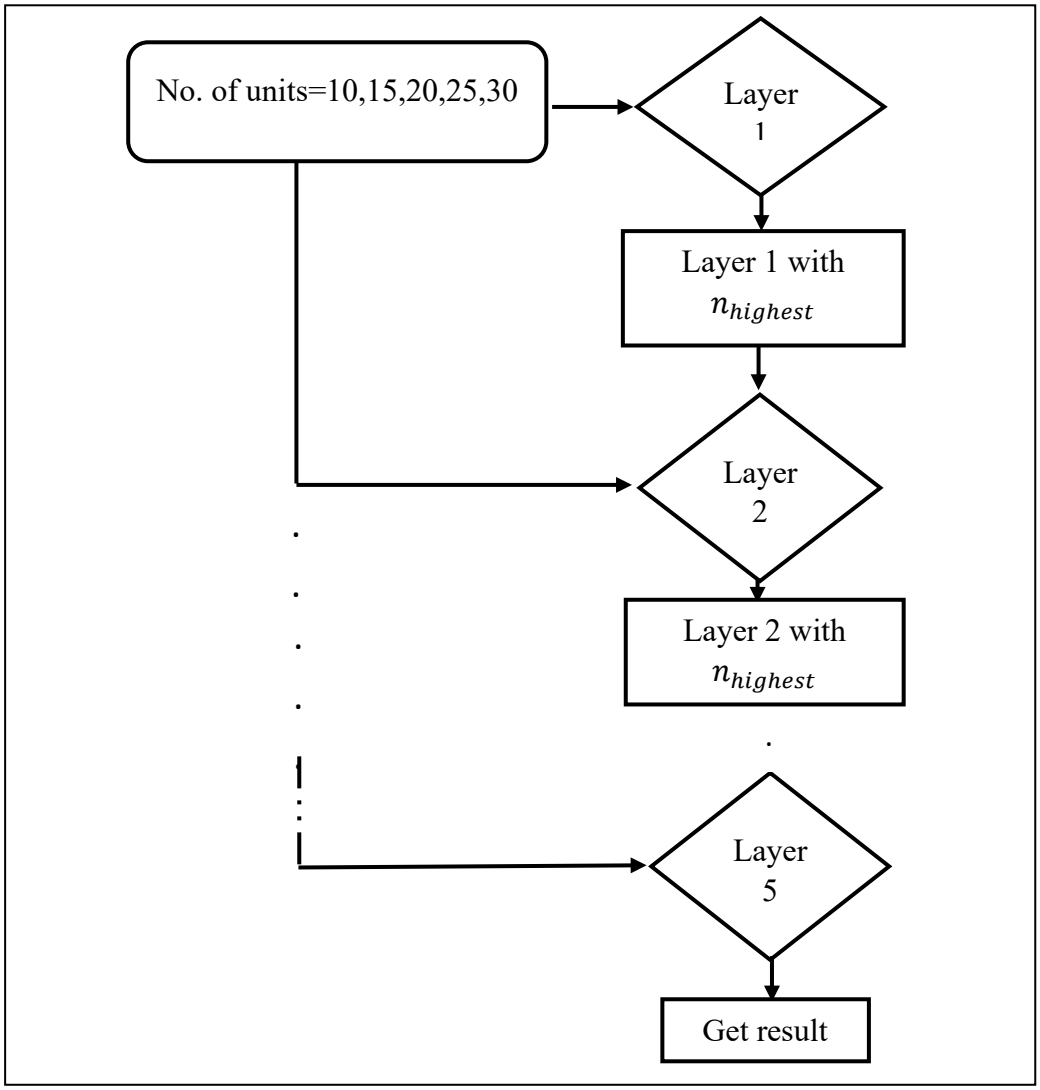

Figure 8: Flowgraph of the four model's implementation

5 RESULTS AND DISCUSSIONS

The models have been evaluated in terms of accuracy. Accuracy measures how often the model makes correct predictions so, it is defined as the accurate predictions divided by the total number of predictions as shown in equation (17).

$$accuracy = \frac{correct\ predictions}{total\ predictions} \tag{17}$$

Figure 9 shows the four models' accuracy in different hidden layers (1-5) based on MFCC-39 coefficient features. All models in hidden layer five have the best results except LSTM, which has the best result in the fourth hidden layer. BRNN-MFCC-5- {30,30,20,25,25} model achieved the best result of them where it is 92.6%.

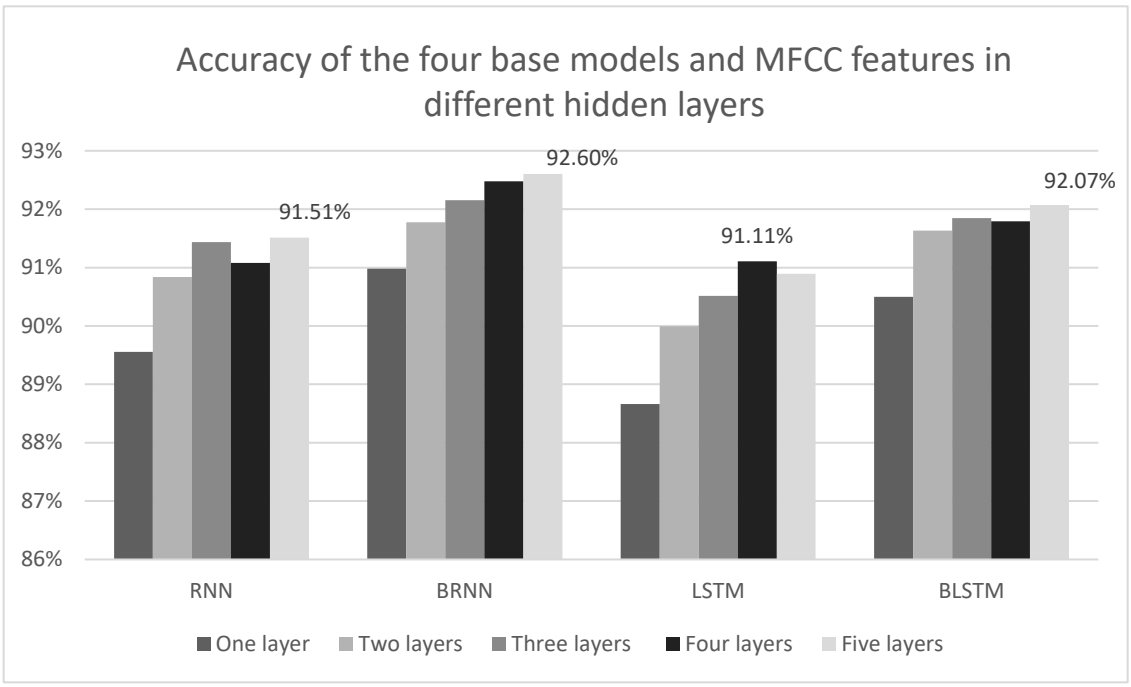

Figure 9: Accuracy results of models and MFCC features with different hidden layers

Figure 10 shows the four models' accuracy in different hidden layers based on MFCC and VQ features, which have one component. Fourth hidden layer achieves higher results in general. For BLSTM, the fourth and fifth hidden layers model has the highest Overall Syllables classification accuracy, which is 78.1445%. From Figure 8 and Figure 9, MFCC-X features achieve higher results than VQ-X, where X is for all classifiers.

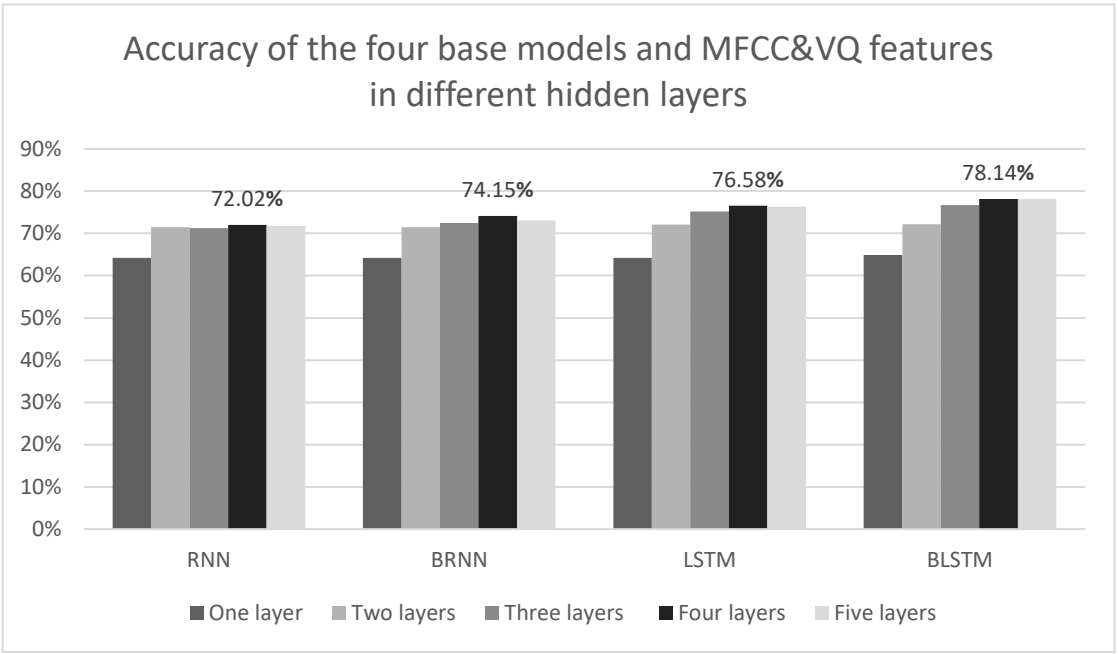

Figure 10: Accuracy results of the four base models with MFCC&VQ features with different hidden layers

TABLE 2

BRNN-MFCC-5- {30,30,20,25,25} CONFUSION MATRIX

	V	P	F	N	SIL	accuracy
V	94000	130	260	890	740	97.9 %
P	190	5500	1800	21	820	66 %
F	600	1700	25000	240	1500	86.1 %
N	1000	12	82	7600	560	82.1 %
SIL	1800	460	930	740	50000	92.7 %

Table 2 shows the confusion matrix and the accuracy of each class of BRNN-MFCC-5- {30,30,20,25,25} system where the experiments give the best result. Best results take the diagonal shape because it has many correct classifications. Table 3, Presents the accuracy of each class in each hidden layer of BRNN-MFCC -5- {30,30,20,25,25} system. Vowels and fricatives have high result in BRNN-MFCC-4- {30,30,20,25}, but plosives and nasals achieve higher results in BRNN-MFCC-5- {30,30,20,25,25}. Silences have high result in BRNN-MFCC-2- {30,30}.

TABLE 3

BRNN-MFCC CLASSES RESULT IN DIFFERENT HIDDEN LAYERS

	V	P	F	N	SIL
one HL	**97%**	**56%**	**83%**	**75.7%**	**92.4%**
two HL	**97.4%**	**58.9%**	**85%**	**76.3%**	**93.2%**
tree HL	**97.6%**	**61.4%**	**85.8%**	**80.4%**	**92.7%**
four HL	**98.3%**	**58.2%**	**89.7%**	**76.9%**	**91.5%**
five HL	**97.9%**	**66%**	**86.1%**	**82.1%**	**92.7%**

BLSTM classes accuracy also has been calculated in each time adding hidden layer where BLSTM model result close to result of BRNN model as shown in Table 4. Vowels, nasals, and silences have higher accuracy in BLSTM-MFCC model than BRNN-MFCC model. Vowels accuracy gives the highest accuracy in BLSTM-MFCC -2- {30,30} with 98.5%, nasals in BLSTM-MFCC -5- {30,30,25,20,30} with 83.6% and silence in BLSTM-MFCC -5 - {30,30,25,20,30} and BLSTM-MFCC -4- {30,30,25,20} with 93.7%. Table 5 illustrate details of each model that achieved the highest accuracy in each class.

TABLE 4

BLSTM-MFCC CLASSES RESULT IN DIFFERENT HIDDEN LAYERS

	V	P	F	N	SIL
one HL	96.5%	48.3%	84.7%	72%	93%
two HL	98.5%	55.7%	87.2%	65.5%	91.8%
tree HL	98.1%	52.9%	86%	74.7%	92.7%
four HL	96.8%	63.1%	87%	69.7%	93.7%
five HL	97%	59.3%	85.2%	83.6%	93.7%

TABLE 5

MODELS OF HIGHER ACCURACY

Classes	Features type	RNN type	Number of hidden layers	Number of hidden units in each hidden layer, respectively	Accuracy %
Vowels	MFCC	BLSTM	2	30-30	98.5
Plosives	MFCC	BRNN	5	30-30-20-25-25	66
Fricatives	MFCC	BRNN	4	30-30-20-25	89.7
Nasals	MFCC	BLSTM	5	30-30-25-20-30	83.6
Silences	MFCC	BLSTM	4 and 5	30-30-25-20 30-30-25-20-30	93.7

By comparing our results with [18], which used the same database (TIMIT) and the same classes, the best results were achieved by the proposed approach (BRNN-MFCC-5- {30,30,20,25,25}), which equal 92.6 % compared with [18], which equal 74.11 %. The vowels class was achieved by 98.5 %; by comparing with [18], which is 91.7 %. Fricatives class was achieved by 89.7 % by comparing with [18], which is 86.9 %. The HMM model has been used in [18]. That is why we used the RNNs model in this research.

6 CONCLUSIONS

It is shown that by increasing the number of hidden layers, accuracy is increased in RNN, BRNN, LSTM, and BLSTM models. Accuracy of LSTM model increased till four hidden layers then decreased. Best result of accuracy was in BRNN-MFCC-5- {30,30,20,25,25} system with overall accuracy and this system gives best result for plosives classes with accuracy 66%. Fricatives gave best result in BRNN-MFCC-4- {30,30,20,25} system with 89.7%. Vowels, nasals, and silences classes give best results in BLSTM -MFCC models. Vowels in BLSTM-MFCC -2- {30,30} system with accuracy of 98.5%, nasals in BLSTM-MFCC -5 - {30,30,25,20,30} with 83.6% and silence in BLSTM-MFCC -4- {30,30,25,20} and BLSTM-MFCC -5- {30,30,25,20,30} with 93.7%.

The highest accuracy (92.6%) is achieved by BRNN compared to similar work that were using the HMM model in [17] and [18] and gave 81.01%, 74.11%, respectively. This indicates that RNN models are more efficient than the HMM model. In the future, training RNNs will be involved with more efficient methods; for example, hybrid models that used convolution neural networks with RNNs [29]. In addition to using more comprehensive features [30] and using different toolkits for RNN training [31] inducing more advanced features.

REFERENCES

[1] Zia, Tehseen, and Usman Zahid. "Long short-term memory recurrent neural network architectures for Urdu acoustic modeling," International Journal of Speech Technology, vol. 22, no. 1, pp. 21-30, 2019.

[2] Ali Shariq Imran, et al. "A Study on the Performance Evaluation of Machine Learning Models for Phoneme Classification," Proceedings of the 2019 11th International Conference on Machine Learning and Computing, Zhuhai, China, pp. 52-58, 2019.

[3] K.-F. Lee, H.-W. Hon, "speaker-independent phone recognition using hidden Markov models," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 37, no. 11, pp. 1641-1648, 1989.

[4] F. Fallside et al., "Continuous speech recognition for the TIMIT database using neural networks," International Conference on Acoustics, Speech, and Signal Processing, Albuquerque, NM, USA, vol.1, pp. 445-448, 1990.

[5] H. A. Bourlard and N. Morgan, "Connectionist speech recognition: a hybrid approach," Kluwer Academic Publishers, ISBN: 978-1-4613-6409-2, vol.247, pp. 4-7, 1994.

[6] A. Graves, A. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, pp. 6645-6649, 2013.

[7] G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82-97, 2012.

[8] A. Mohamed, G. E. Dahl and G. Hinton, "Acoustic modeling using deep belief networks," IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 1, pp. 14-22, 2012.

[9] P. Karjol and P. K. Ghosh, "Broad Phoneme Class-Specific Deep Neural Network Based Speech Enhancement," 2018 International Conference on Signal Processing and Communications (SPCOM), pp. 372-376, 2018.

[10] C. Antoniou, "Modular neural networks exploit large acoustic context through broad-class posteriors for continuous speech recognition," IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 01CH37221), Salt Lake City, UT, USA, vol. 1, pp. 505-508, 2001.

[11] P. Scanlon, D. P. Ellis, and R. B. Reilly, "Using broad phonetic group experts for improved speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 3, pp. 803-812, 2007.

[12] W. Rochkitticharoen, A. Suchato, and P. Punyabukkana, "Broad phonetic class segmentation study for Thai automatic speech recognition," 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Phetchaburi, pp. 1-4, 2012.

[13] Reynolds, T. Jeff, and Christos A. Antoniou, "Experiments in speech recognition using a modular MLP architecture for acoustic modeling," Information Sciences, vol. 156, no. 1-2, pp. 39-54, 2003.

[14] A. Chittora and H. A. Patil, "Classification of phonemes using modulation spectrogram based features for Gujarati language," International Conference on Asian Language Processing (IALP), Kuching, pp. 46-49, 2014.

[15] G. Deekshitha and L. Mary, "Broad phoneme classification using signal-based features," International Journal on Soft Computing, vol. 5, no. 3, pp. 1, 2014.

[16] M. Aissiou and M. Guerti, "Genetic algorithm applied to the standard Arabic phonemes classification," Cybernetics and Systems Journal, vol. 39, no. 3, pp. 99-212, 2008.

[17] Doaa N. Senousy, Amr M. Gody, and S. F. Saad, "Syllables Classification for ASR using Variable State Hidden Markov Model, " in the 18th Conference on Language Engineering, Ain Shams University, pp. 1-11, 2018.

[18] Doaa A. Lehabik, Mohamed H. Merzban, Sameh F. Saad, Amr M. Gody, "Broad Phonetic Classification of ASR using Visual Based Features," vol.7, no. 1, pp. 14-26, 2020.

[19] G. Kiss, D. Sztahó, and K. Vicsi, "Language independent automatic speech segmentation into phoneme-like units on the base of distinctive acoustic features," IEEE 4th international conference on cognitive info-communications (CogInfoCom), Budapest, Hungary, pp. 579-582, 2013.

[20] M. Antal, "Phonetic speaker recognition," Proceedings of the 7th International Conference, Bucharest, Romania, pp. 67-72, 2008.

[21] Hasan, M.R., Jamil, M., Saifur Rahman, "Speaker identification using Mel Frequency Cepstral Coefficients," 3rd International Conference on Electrical and Computer Engineering, Dhaka, Bangladesh, pp. 565-568, 2004.

[22] Kurzekar, P., Deshmukh, R., Waghmare, V., and Shrishrimal, " A Comparative Study of Feature Extraction Techniques for Speech Recognition System, " International Journal of Innovative Research in Science, Engineering and Technology, vol. 3, no. 12, pp. 18006-180016, 2014.

[23] M. A. Hossan, S. Memon, and M. A. Gregory, "A novel approach for MFCC feature extraction," 4th International Conference on Signal Processing and Communication Systems, Gold Coast, QLD, pp. 1-5, 2010.

[24] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," IEEE Transactions on Communications, vol. 28, no. 1, pp. 84-95, 1980.

[25] C. Olah, "Understanding LSTM Networks," Web Site: http://colah.github.io/posts/2015-08-Understanding-LSTMs, 2015.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[27] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural networks, vol. 18, no. 5-6, pp. 602-610, 2005.

[28] F. Chollet, "Keras, " Available from https://github.com/fchollet/keras, 2017.

[29] D. Amodei et al., "End to end speech recognition in English and Mandarin," Proceedings of The 33rd International Conference on Machine Learning, New York, USA, pp.173-182, 2016.

[30] K. Yao, G. Zweig, M.-Y. Hwang, Y. Shi, and D. Yu, "Recurrent neural networks for language understanding," Proceedings of Interspeech Conference, Lyon, France, pp. 2524-2528, 2013.

[31] T. Mikolov, S. Kombrink, A. Deoras, L. Burget, and J. Cernocky, "Rnnlm-recurrent neural network language modeling toolkit," in Proc. of the 2011 ASRU Workshop, Hilton Waikoloa Village, Big Island, Hawaii, pp. 196-201, 2011.

[32] Ibrahim Kandel and Mauro Castelli, "The effect of batch size on the generalizability of the convolution neural networks on a histopathology dataset," ICT Express, vol. 6, no. 4, pp. 312-315, 2020.

BIOGRAPHY

Ayat N. Ragheb received a B.Sc. degree in Electrical Engineering – Communications and Electronics Department with excellent and honor degree from the Faculty of Engineering - Fayoum University in 2014. She joined the M.Sc. program at Fayoum University - Communications and Electronics Department in 2015. Her areas of interest include automatic speech recognition.

Amr M. Gody received the B.Sc. M.Sc., and PhD. from the Faculty of Engineering, Cairo University, Egypt, in 1991, 1995, and 1999. He joined the teaching staff of the Electrical Engineering Department, Faculty of Engineering, Fayoum University, Egypt, in 1994. He is the Acting Chief of the Electrical Engineering Department, Fayoum University, in 2010, 2012, 2013, 2014, and 2016. His current research areas of interest include speech processing, speech recognition, and speech compression. He is author and co-author of many papers in national and international conference proceedings and journals such as Springer(International Journal of Speech Technology), the Egyptian Society of Language Engineering (ESOLE) journal and conferences, International Journal of Engineering Trends and Technology (IJETT), Institute of Electrical and Electronics Engineers (IEEE), International Conference of Signal Processing And Technology

(ICSPAT), National Radio Science Conference(NRSC), International Conference on Computer Engineering &System (ICCES) & Conference of Language Engineering(CLE).

Tarek M. Said received the B.Sc. from Electrical Engineering - Cairo University - Fayoum branch - 1998, M.Sc. from Cairo University - 2004, and PhD. from University of Arkansas - Fayetteville - Arkansas USA - 2009. He is now a lecturer in the Electrical Engineering Department, Faculty of Engineering, Fayoum University. His current research areas include Computational Electromagnetics, Biomedical Imaging, Dielectric properties of Biological Tissues.

TRASLATED ABSTRACT

دراسة مقارنة لأنواع مختلفة من الشبكات العصبية المتكررة في تصنيف الكلام

آيات نبيل حامد*[1] , عمرو محمد جودي *[2], طارق مصطفى سعيد* [3]
* قسم الهندسة الكهربية, جامعة الفيوم , مصر
[1]an1162@fayoum.edu.eg
[2]amg00@fayoum.edu.eg
[3]tms02@fayoum.edu.eg

ملخص

يقدم هذا البحث نماذج مختلفة لتصنيف المعالجة المسبقة وأدائها في نظام التعرف الآلي على الكلام. تم اختبار بنيات مختلفة للشبكة العصبية المتكررة (RNN) لهذه المشكلة، مثل خلايا الشبكة العصبية المتكررة (RNN)، الشبكة العصبية المتكررة ثنائية الاتجاه (BRNN)، الذاكرة طويلة المدى (LSTM)، والذاكرة طويلة المدى ثنائية الاتجاه. تم استخدام طريقتين رئيسيتين لاستخراج المميزات الخاصة للمقاطع. أولاً، تم استخدام معامل cepstral للتردد Mel (MFCC) بالإضافة إلى معاملات دلتا ودلتا دلتا (39 معامل). ثانيًا، تم استخدام تكميم MFCC باستخدام تقنية VQ لاستخراج المميزات الخاصة للمقاطع. تم تدريب جميع النماذج على قاعدة بيانات TIMIT. تم اختيار حروف العلة، والأنف، والمشتقات، والصمت، وفئات المقاطع لتصنيفها. تظهر نتائج التجربة أن نظام {25،25،20،30،30} - BRNN-MFCC-5 يعطي أعلى دقة. حيث حققت 92.6٪.

الكلمات الدالة

التعرف الآلي على الكلام، تقنية تصنيف الكلام، الشبكات العصبية المتكررة، شبكة العصبية المتكررة ثنائية الاتجاه، الذاكرة طويلة المدى، الذاكرة طويلة المدى ثنائية الاتجاه، MFCC المتجهات الكمي (VQ) .