# Performance Evaluation of Feed-forward Neural Network Models for Speech Recognition

*Vikas Pandey*
*Assistant Professor, Department of Computer Science & Application,*
*MCNUJC University, Bhopal, M.P., India.*

***Abstract-*** Automatic signal recognition (ASR)is one of the most appealing application domains of pattern recognition tasks. Due to its applicability in various engineering and scientific fields, automatic speech recognition has become the most explored researched area now-a-days. In this paper, we are evaluating and analyzing the performance of various feed-forward neural network models for the generalized classification of speech signals. This analysis presents an investigation of the speech classification performance of the various feed-forward neural network models. The work is conducted with ten samples of speech signals for first five alphabets, i.e. two speech signals of each alphabet, of English language. Digital signal processing operations are applied on analog speech signals to convert them into digital form and then to make them suitable for further processing by neural network models. Performance of each neural network model is measured and analyzed for the pattern samples of speech signals. This performance analysis is performed on five standard feed-forward neural network architecture as the classifiers. The utilized standard feed-forward neural network models include multilayer feed-forward neural network with back propagation algorithm, Radial basis function neural network, Exact radial basis function neural network, Cascade feed-forward neural network and Elman back propagation neural network model. It has been observed that the Cascade feed-forward neural network gives highest recognition accuracy for most of the given input speech signal samples. It has been also examined that Radial basis function network and Exact radial basis function network gave lowest recognition rate among all the selected networks for the given samples of speech signals.

***Keywords-*** *Pattern recognition, Signal processing, Speech classification, Feed-forward neural network, Radial basis network.*

## I. INTRODUCTION

Speech signal is the most important and basic medium of human communication, which carries valuable information like speaker identity and language information [1].It has been observed that speech could be a useful interface to interact with machine in the form of voice or speech recognition [2]. It becomes the most useful interface to interact with the machines because there is no need to interact with the machine using command line interface through keyboard or graphical interface through mouse. A speech signal is a wave which is one-dimensional and has temporal structure. Thus, speech analysis is based on harmonic analysis or waveform analysis. Compared to the image analysis, there is very few information to be used directly for the recognition in the waveform in speech analysis. A number of speech processing and recognition techniques have been applied to identify and use this medium efficiently in many application areas [3]. Automatic speech recognition is a technology which allows a machine to understand the words or text spoken by a person, identify them and convert them into the text form. ASR can be defined as the independent, computer–driven transaction of spoken language into readable text in real time [4]. Therefore, due to the increasing popularity and demand, automatic speech recognition has become one of the emerging and most interesting areas in the problem domain of pattern recognition [5].This area is receiving much attention now-a-days and is being broadly used in a variety of engineering and scientific fields to solve various useful pattern recognition problems like telephone directory assistance, office dictation devices, automatic voice translators into foreign languages, spoken database querying services, automatic words or phrases speaking systems, words pronunciation systems, etc.[6].Thus, automatic speech recognition technology provides valuable and remarkable services to diverse application areas to make machines capable to respond correctly and reliably to human voices [7]. Automatic speech recognition is a very challenging task because human spoken words or text depends on many characteristics of the person such as speaking style, speed, surrounding noise elements, pitch, etc. A person may speak the same text in different ways on various times. Thus, inputs provided to the system may be highly variable. Also, an individual speaker or different speakers may use widely varying speaking levels for the same word or sequence of words, which may produce variable-length speech signals [8]. Another major problem is that the speech of a person may vary with respect to age, accent, emotional state, etc. [9]. Hence, to handle these problems, the modular structure of speech recognition system can be considered which is similar to the human mechanism of speech perception. The basic structure of ASR consists of feature extraction module and classification module [10]. It is considered that a speech recognition system can be an isolated word recognition system

,a connected word recognition system, a continuous speech recognition system or a spontaneous speech recognition system[11]. Isolated word recognition is considered as a classification problem of machine learning in which the machine accepts single words or sounds at a time and have time boundaries in both training and testing data [12]. The continuous speech may be considered as structural machine learning and there is no time boundary limitation on the input speech[13,14].Thus continuous speech recognition is more difficult than the isolated word speech recognition. The accuracy of any speech recognition system depends on various parameters such as size of the vocabulary, presence of confusable words like B, C, E, T& V, speaker dependent or independent system, environmental noise, acoustical distortions such as echoes, difference in microphones, etc., various ways of speaking like shouting, whispering, speaking quickly or slowly, etc. [15,16]. ASR has been an interesting area of research to segment and classify a speech signal as a voiced speech signal, an unvoiced speech signal or silence (absence) of speech signal based on different measurements made on the signal [17]. Silence in a speech recognition system is equally important as speech because it defines the start and end of the speech or the utterance [18]. Lots of efforts have been done in the field of speech recognition since last eighty years. In 1939, a system model for the analysis and synthesis of speech signals was proposed [19]. Beside this further, five measurements such as zero-crossing rate, the speech energy, the correlation between adjacent speech samples, the first predictor coefficient from a 12-pole LPC (linear predictive coding) analysis and the energy were considered for error to analyze the speech signals [20]. In this approach, it was assumed that the measured parameters are distributed according to the multidimensional Gaussian probability density function. The input speech segment is assigned to one of the three output classes voiced, unvoiced and silence by applying the minimum distance rule. The training set consisted of two sounds of sentences spoken by a male speaker and the testing set comprised of two different sentence sounds spoken by another male and a female speaker. The training set was used to compute the means and the covariance matrices for the three classes. In the early eighties, connectionist machines and Boltzmann machines were also used by speech scientists for performing the task of speech recognition [21, 22]. It has been observed that the major problem with automatic speech recognition systems is to handle the variable length speech sequences. Hidden Markov Models (HMMs) are good in handling such data and modeling temporal behavior of the speech signals using a sequence of states. HMMs based phonemes recognition techniques were proposed in which feature pattern vector was created by extracting features from a single phoneme [23]. Further, HMMs and neural network models have also been used for speech recognition tasks [24, 25]. It is reported that the neural networks have been used extensively for many applications in speech processing and recognition such as speech synthesis, speaker adaptation and recognition, keyword spotting, etc. [26-28]. Wavelet transformation based improved feature vector formation technique for the classification of audio signals is applied [29]. A secure neural network based Interactive Voice Response system was presented in which a user can use the system only after entering the correct password. After that, user was requested to input the voice sample which is used to identify the user. These voice samples are transformed into frequency domain to extract speaker specific features using the Mel Frequency Cepstral Coefficient (MFCC) feature extraction method [30]. Therefore, based on the extracted features, system recognized between the known and unknown speaker's identity. Hence, results were based on the false accept and false reject criteria. Back propagation neural network with identification rate above 90% was used to perform and improve the accuracy of the classification of audio signals [31].A multilayer perceptron neural network system optimized with genetic algorithm for classifying audio into speech and music was presented and wavelet transformation method has been applied for feature extraction [32]. It is reported that in this model system achieved 96.49%recognition accuracy. A Back-propagation neural network based recognition system with 96.33% and 92% recognition accuracy for known and unknown speakers respectively was implemented for isolated Bangla speech [33]. In this system, 10 Bangla digits (0-9) from 10 speakers were recorded as .wav files at the sampling rate of 8.00 KHz, then 58 bytes from the beginning were discarded for deleting the header information &extracting the wave data, and finally start and end-point detection technique was applied on the speech signal to extract the voiced data. Mel Frequency Cepstral Coefficient (MFCC) method was applied for feature extraction to construct the training set for neural network. A Convolutional Neural Network (CNN) is used for speech recognition and reduced the error rate by 6% -10% as compared to the Deep Neural Networks (DNNs) in which the input data is organized as a number of feature maps[34]. The CNN contains a pair of hidden layers- a convolution layer and a pooling layer. In this scheme a smaller number of units in pooling layer are used, which results in smaller model size and lower computational complexity than the full weight sharing scheme. Despite of the number of sincere efforts and research work done in the area of automatic speech recognition using artificial neural network, still there is some space left for the selection of optimal neural network architecture for recognition of speech with good accuracy. Therefore, in the present paper, we are investigating the performance of different feed-forward neural network models to select the optimal and suitable model for the speech recognition of first five alphabets of English language. Five feed-forward neural network models like Multilayer feed-forward network, Radial basis function network, Exact radial basis function network, Cascade feed-forward network and Elman back propagation network have been selected for the experiment. A comparative analysis for the recognition accuracy of the selected neural network models for noiseless and noisy input speech samples is also conducted to explore the analysis of performances of the networks for the given speech samples. This paper is further organized in five sections. Section 2 discusses the feature extraction process of the input speech samples presented for the experiment. In section 3, implementation details of neural network models used for the speech classification are provided. Section 4 presents the simulation results, comparative study of recognition accuracy, performances of the selected neural network models and a complete discussion of the results. Section 5 considers the conclusion followed by references.

## II. FEATURE EXTRACTION

Ten (10) speech signal samples (two of each) of English letters 'A', 'B', 'C', 'D' and 'E' spoken by a single female speaker are considered as input data samples. Individual alphabet is spoken in two different ways(quickly and slowly) for the time duration of 4 seconds each. All input speech signals are collected as audio files. The first five or quickly spoken speech signals are named as A1, B1, C1, D1 & E1; while the second set of speech signals, which are spoken slowly, are named as A2, B2, C2, D2 & E2. The set of collected input signals is presented in figure 1.

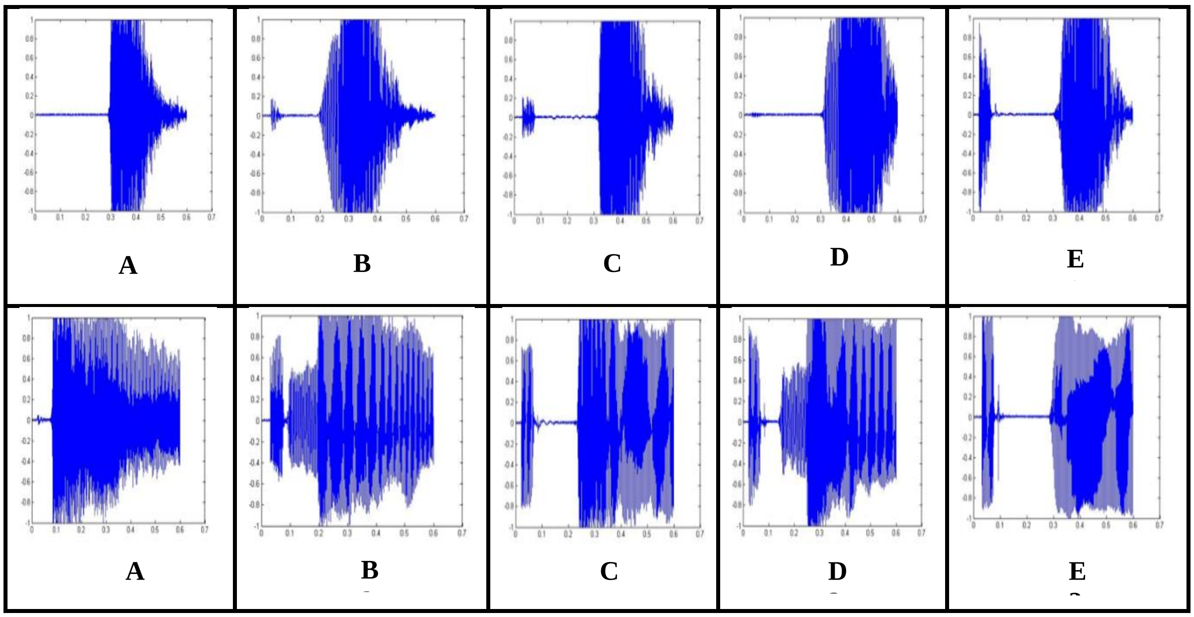



Fig.1: Set of collected input signal samples

These input signals are converted into digital form by applying two important digital signal processing operations, i.e. sampling and quantization, to make these speech input signals suitable for further processing of speech classification by neural network models. It has been widely accepted that most of the signals such as speech, radar signals and various communication signals are analog. Therefore, to process these analog signals it is necessary to convert them into digital form, i.e. to convert them into a sequence of numbers having finite precision. Generally this conversion needs three step process- sampling, quantization and coding as shown in figure 2.

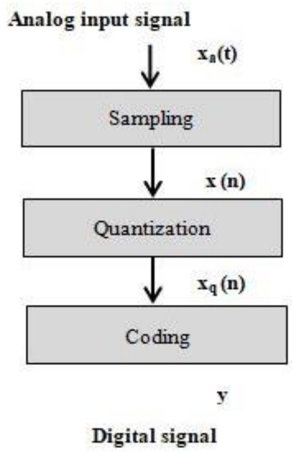



Fig.2: Applying digital signal processing operations on an analog signal

Here, Sampling is the process of obtaining signal values from continuous signal at regular time intervals or at discrete time instants as[35]:

$$x(n) = x_a(nT) \qquad (1)$$

Where $x_a(t)$ is the input continuous signal, T is the sampling instance and n is the sample index.

Hence, the time interval T between successive samples is called the sampling period or sample interval, and

$$\frac{1}{T} \equiv F_s \qquad (2)$$

or

$$T = nT = \frac{n}{F_s} \qquad (3)$$

Where $F_s$ is the sampling rate or sampling frequency.

In the sampling process, only the time elements are discretized, amplitude values are remained continuous. The values obtained can take on any number from a continuous range and thus there may be infinitely many possible values for each obtained value of the discrete time sequence signal. The amplitude of x[n] is known with infinite precision. Thus, to represent each value, an infinite number of digits are required instead of a finite number of digits. Therefore to solve this type of ambiguity, the quantization process is applied.

Quantization is a process of converting a discrete-time continuous-amplitude signal into a digital signal by expressing each sample value as a finite number of digits [36]. Let $x_q(n)$ denotes sequence of quantized samples after the quantization process as [37]:

$$x_q(n) = Q[x(n)] \qquad (4)$$

where x(n) represents the samples.

The quantization process includes an error which is introduced in representing the continuous-valued signal by a finite set of discrete value levels. This quantization error $e_q(n)$ is defined as:

$$e_q(n) = x_q(n) - x(n) \qquad (5)$$

The quantization error $e_q(n)$ in rounding is limited to the range of $-\Delta/2$ to $\Delta/2$ as:

$$-\Delta/2 \le e_q \le \Delta/2 \qquad (6)$$

where D is the quantization step size.

Also, $$\Delta = \frac{x_{\max} - x_{\min}}{L - 1} \qquad (7)$$

Hence, if the dynamic range is fixed, increasing the number of quantization levels, L, results in a decrease of the quantization step size. Thus, the quantization error decreases and the accuracy of the quantizer increases. With the presence of quantization error, we can represent the quantized information as:

$$x_q(n) = x(n) + e_q(n) \qquad (8)$$

Now, in order to use the quantization error for the construction of pattern information, we consider the following assumptions [38]:

- The error $e_q(n)$ is uniformly distributed over the range $-\Delta/2 < e_q(n) < \Delta/2$
- The error sequence [ $e_q(n)$ ] is a stationary white noise sequence.
- The error sequence [ $e_q(n)$ ] is uncorrelated with the signal sequence x(n)
- The signal sequence is of zero mean and stationary.

Therefore, with these assumptions, the effect of the additive noise $e_q(n)$ on the desired

signal can be quantified by evaluating the signal-to-quantization noise ratio (SQNR) as:

$$SQNR = 10\log_{10}\frac{Px}{Pn}$$

where

$$Px = \sigma_x^2 = E[x^2(n)] \; P_n = \sigma_e^2 \qquad (9)$$

So, $$P_x = E[e_q^2(n)] \qquad (10)$$

Hence, the signal information is processed and the digital form of input signal is constructed. These digitized signals of five different alphabets from A to E are presented in pattern vector form of size 4800 x 1. Thus, the training set is of size 4800 x 10. These digitized signals in pattern vector form are used for training with feed-forward neural network models to generate the required signal classification. Further, to evaluate the performance of trained neural network models the test patterns are constructed. These test patterns are constructed by introducing 10%, 20%, 30%, 40%, 50%, 60% & 70% noise or error respectively to the signals used for training. These noisy analog signals are processed with sampling, quantization and coding steps for digitized presentation. These noisy digital signals are further presented as test pattern vectors and used to evaluate the performance of trained neural network models.

## III. IMPLEMENTATION OF NEURAL NETWORK MODELS

An artificial neural network or ANN is a computational model which is designed to perform the complex pattern recognition tasks such as pattern classification, pattern mapping, pattern

Let us assume that the quantization error is uniformly distributed in the range (-$\Delta$/2, $\Delta$/2). Hence the mean value of the error is zero and the variance is:

$$P_n = \sigma_e^2 = \int_{-\Delta/2}^{\Delta/2} e^2 p(e)de = \frac{1}{\Delta}\int_{-\Delta/2}^{\Delta/2} e^2 de = \frac{\Delta^2}{12} \qquad (11)$$

The third process for the digital representation of the signal is coding. Coding process assigns a unique binary number to each quantization level. Therefore, if we have L levels, we need at least L different binary numbers. Thus, for a word length of L+1 bits, $2^{L+1}$distinct binary numbers can be represented. Hence, we should have $2^{L+1} \geq L$ or $L+1 \geq \log_2 L$.

Thus, the step size of the quantization can be expressed as:

$$\Delta = \frac{\mu}{2^{L+1}} \qquad (12)$$

where μ is the range of quantization.

Now from equation (9), (10) and (11), we have:

$$SQNR = 10\log\frac{P_x}{p_n} = 20\log\frac{\sigma_x}{\sigma_e} \qquad (13)$$

Thus,

$$SQNR = 6.02e + 16.81 - 20\log\frac{\mu}{\sigma_x} dB \qquad (14)$$

association, etc. Thus, a neural network can be characterized as a computing architecture, which consists of a large number of simple highly interconnected data processing element called neurons, designed to resemble the learning and storing capability of human brain for performing the task of pattern recognition [39, 40]. In this paper, five feed-forward neural network models are explored. These models are trained with variants of gradient descent method of generalized delta learning rule. The basis of all delta learning rule is back propagation learning rule [41].Back propagation (BP) is a supervised learning algorithm and belongs to a class of "learning with the teacher"[42]. Back propagation is a systematic method of training multilayer artificial neural networks in which a predefined desired target output (t) for each input pattern is prepared. This target output is compared with the actual output (o) and difference is termed as error (E). The value of the error term is propagated backward from the output layer to hidden layer/s to update the weights in the hidden and output layer respectively as shown in equations 15 & 16 as [43]:

$$v_{kj} = v_{kj} + c\lambda(t_k - z_k)z_k(1 - z_k)o_i \qquad (15)$$

And

| Parameter | Value |
|---|---|
| Number of hidden layers | 3 |
| Number of neurons in first hidden layer | 7 |
| Number of neurons in second hidden layer | 11 |
| Number of neurons in third hidden layer | 15 |
| Number of neurons in output layer | 10 |
| Transfer function for first layer | Hyperbolic Tangent Sigmoid |
| Transfer function for second layer | Hyperbolic Tangent Sigmoid |
| Transfer function for third layer | Hyperbolic Tangent Sigmoid |
| Training function | Levenberg-Marquardt |
| Maximum number of epochs | 1000 |
| Performance function | Mean squared error |
| Error goal | 0.00001 |
| Adaption rate | 1.0 |
| Backpropagation learning rate | 0.1 |
| Initial weights and biased term values | Values generated randomly between 0 and 1 |

Table 1: Parameters used for creating the Multilayer Feed-forward network

$$w_{ji} = w_{ji} + c\lambda^2 o_i(1-o_i)x_i\left(\sum_{k=1}^{K}(t_k - z_k)z_k(1-z_k)v_{kj}\right) \quad (16)$$

Mean of the error at the $k^{th}$ iteration is computed as:

$$E = E + \frac{1}{K}\sum_{k=1}^{K} \left(t_k - z_k\right. \quad (17)$$

where $w_{ji}$ is the weight connecting input layer $i^{th}$ neuron to the $j^{th}$ neuron of the hidden layer, $w_{kj}$ is the weight connecting hidden layer $j^{th}$to the $k^{th}$neuron in the hidden layer, $\lambda$is the parameter used to control the gradient of the function ,$t_k$ is the output of the $k^{th}$ target vector and $o_i$ is the output of the net output of the hidden layer neuron. In the present work, the Multilayer feed-forward neural network model is considered first and this is trained with Levenberg-Marquardt learning rule. In this experiment, size of input pattern vector P and target output vector T are considered same because the input and output patterns are the signal pattern information. The parameters used for the architecture and training of the network are presented in table

Second neural network model used in this work is Radial basis function network (RBF). RBF network is a three layer feed-forward neural network and consists of a single hidden layer in its structure, as shown in figure 3. In this architecture, the hidden layer is non-linear and output layer is linear. Hence, due to the non-linear characteristics, RBF is able to model the complex pattern mapping problems and exhibit the better generalization [44]. In this network, the number of neurons in the first layer is less than the number of samples and each unit implements a radial basis function such as Gaussian radial function, Quadratic function, Inverse quadratic function, Thin plate spline, etc. Hence, activation function of the hidden layer computes the Euclidean distance between the input vector and center of that unit and the value of the function increases or decreases monotonically with the distance from a center point.

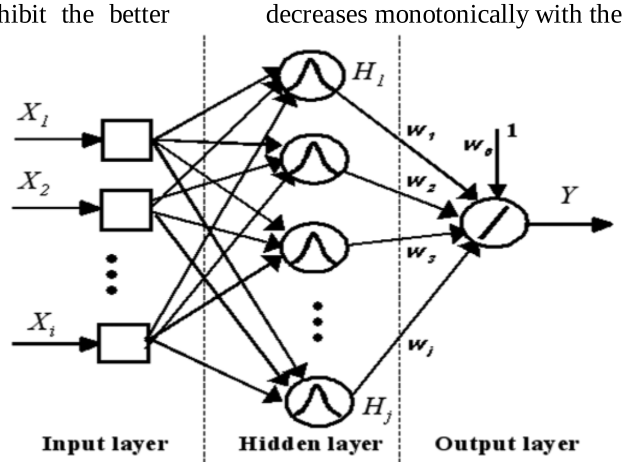


Fig.3: Radial basis function neural network

Here, the Gaussian function is widely used to compute the activation value for the unit of the middle layeras [45]:

$$z_k(x_l) = \exp\left[-\frac{\| x_l - \mu_k \|^2}{2\sigma_k^2}\right] \quad (15)$$

for j = 1,2,3,…………………, M

where ‖.‖ is the Euclidean norm,x is the N-dimensional input vector,$\sigma_k$is the width of the neuron and $\mu_k$is the mean of the k$^{th}$ Gaussian function.

Thus, for an input pattren x, the output of the j$^{th}$ node of the output layer can be expressed as[46]:

$$y_j(x) = \sum_{k=1}^{K} w_{kj} z_k(\| x_i - \mu_k \|) + w_{0j} \quad (16)$$

for j = 1, 2, 3, ………….., M
where $y_j(x)$ is the output of the j$^{th}$ processing element of the output layer in the network, $w_{kj}$is the connection weight from the k$^{th}$ hidden unit to the j$^{th}$ output unit and $\mu_k$ is the center of the k$^{th}$ hidden unit. In feed-forward RBF network the weight vector between the input layer and the k$^{th}$ hidden layer can be taken as the center $\mu_k$ [47]. In the RBF network, learning takes place in two stages. In the first stage, the input data set is used todetermine the parameters of the selected basis function and weights of the fisrt layer are updated by the unsupervised learning. In the second stage of learning, weights of the second layer are calculated with the gradient descent approach. Therefore, for the optimization of the basis function, a number of techniques such as clustering algorithms, unsupervised learning, supervised learning, etc. have been proposed. But, to obtain the optimal performance, suprevised learning method is applied because it is required to include the target pattern vector in the training procedure. Thus, the weight vector modification and basis function parameters update is performed in iterative manner to accomplish the learning in supervised way. Hence, the update in weight and bais parameters at the mth step of iteration can be expressed as [48]:

$$w_{ij}(m) = w_{jk}(m-1) + \eta_1 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)s_j^l(y_j^l).\exp\left(-\frac{(\| x_i^l - \mu_{ki}^l \|)^2}{2\sigma_k^2}\right)$$

$$\mu_{ij}(m) = \mu_{jk}(m-1) + \eta_2 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)s_j^l(y_j^l).w_{jk}.\phi_k(x_i^l).\left(\frac{x_i^l - \mu_{ki}^l}{\sigma_k^2}\right)$$

$$\sigma_k(m) = \sigma_k(m-1) + \eta_3 \sum_{j=1}^{M}\sum_{k=1}^{K}(d_j^l - y_j^l)s_j^l(y_j^l).w_{jk}.\phi_k(x_i^l).\left(\frac{\| x_i^l - \mu_{ki}^l \|^2}{\sigma_k^2}\right)$$

where η is the learning rate parameter, is the weight between the i$^{th}$ unit of output layer and j$^{th}$ radial unit of the middle layer. The next neural network model used for the simulation design is Exact Radial Basis Network. The Exact Radial basis network is a radial basis network with exact approximation which produces a network with zero error on training vectors. The difference between Radial basis function and Exact radial basis function network is that the RBF network does not necessarily produce the correct output, whereas Exact RBF network gives the exactly correct outputs for all the training data. This network does not perform well when the network is defined in terms of several input vectors. In this network, there are as many neurons in the first layer as the number of samples. The parameters used for the architecture and training of the Radial basis function network and Exact radial basis function network are presented in table 2.

| | **Radial basis Function Network** | **Exact radial basis function network** |
|---|---|---|
| **Parameter** | **Value** | **Value** |
| Performance function | Mean squared error | Mean squared error |
| Spread of Radial basis function | 1.0 | 1.0 |
| Number of neurons in layer 1 | 10 | 10 |
| Number of neurons in layer 2 | 4800 | 4800 |
| Transfer function in layer 1 | Radial basis transfer function | Radial basis transfer function |
| Transfer function in layer 2 | Linear transfer function | Linear transfer function |
| Backpropagation learning rate | 0.1 | 0.1 |

Table 2: Parameters used for creating the Radial basis function network and Exact radial basis function network

The next neural network model selected for the simulation is Cascade feed-forward neural network. The architecture of this network is similar to the back propagation neural network, the only difference lies that this network also has additional weight connection from the input layer to each forwarding layer along with from each layer to the successive layers as shown in figure4.

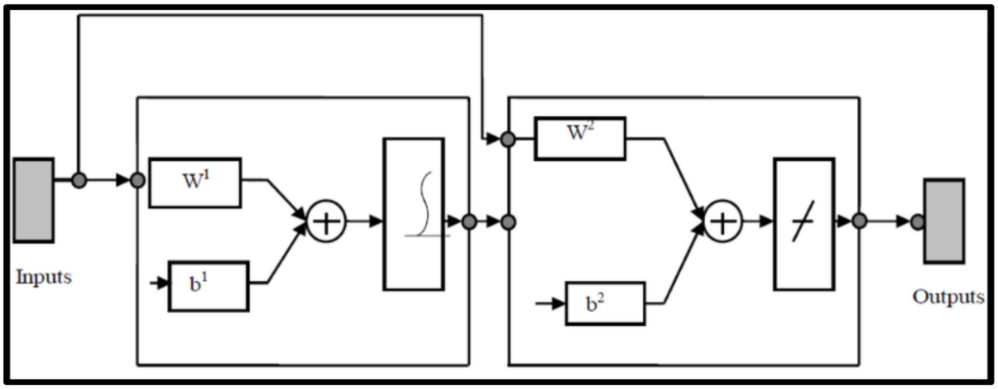



Fig.4: Cascade feed-forward neural network

Hence, due to these additional connections, speed of cascade network for convergence may increase, because each of the cascade neural network's neuron is trained independently to each other. Thus, this network learns faster than the back propagation neural network. Therefore, the weight and bias updating for this network can be expressed as [49]:

$$\Delta b^k = \eta \delta^k \quad (23)$$

For k=1….M

$$\Delta W^k = -\eta \delta^k (a^{k-1})^T \quad (24)$$

where a is the output vector, η is the learning rate and δ is the sensitivities defined in relation to the changes of the activation level

The last neural network used in this work for simulation is Elman back propagation neural network, which is a type of recurrent network[50]. It is a two layer network in which a recurrent connection exists from the output of the hidden layer to its input as shown in figure5. The parameters used for the architecture and training of the network are presented in table 3.

.

| **Parameter** | **Value** |
|---|---|
| Number of hidden layers | 1 |
| Number of neurons in the hidden layer | 20 |
| Number of neurons in output layer | 10 |
| Transfer function | Linear transfer function |
| Training function | Levenberg-Marquardt |
| Maximum number of epochs | 1000 |
| Performance function | Mean squared error |
| Error goal | 0.00001 |
| Backprpagation learning rate | 0.1 |
| Initial weights and biased term values | Values generated randomly between 0 and 1 |

Table 3: Parameters used for creating the Cascade feed-forward network

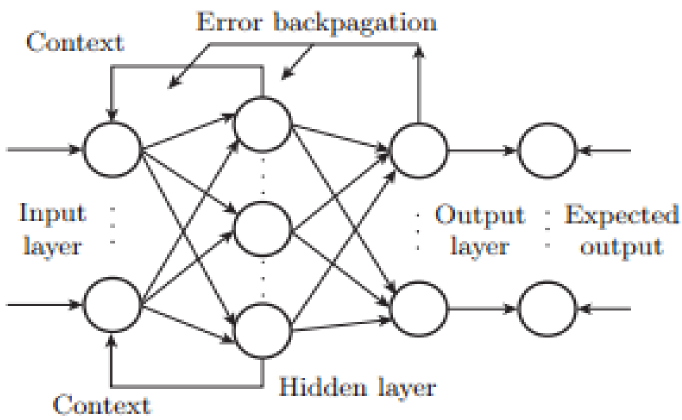



Fig.5: Elman back propagation neural network

In addition to the input, output and hidden units, network also consists of context units. These additional units work as a function of time-delays and are used to learn the previous activations of the hidden units[51].Thus, the input information is continuously recovered and processed over multiple time points. Hence, the total input to the i^th hidden unit can be expressed as [52]:

$$v_i(k) = \sum_{j=1}^{n} w_{i,j}^{x}(k-1)x_j^c(k) + w_i^u(k-1)u(k) \quad (25)$$

where $x_j^c$ is the output of the j^th context unit, $w_{i,j}^x$ is the weight between the context layer and the hidden layer, $w_i^u$ is the weight between the input unit and hidden layer and $u(k)$ is the external input.

Output of the i^th hidden unit and j^th context unit can be represented by equations 26& 27 respectively:

$$x_i(k) = f(v_i) \quad (26)$$

$$x_j^c(k) = x_j(k-1) \quad (27)$$

Thus, output of the network can be represented as:

$$y(k) = \sum_{i=1}^{n} w_i^y(k-1)x_i(k) \quad (28)$$

where $w_i^y$ is the weight between the output layer and the hidden unit.

Parameters used to create and train the network are shown in table 4.

| **Parameter** | **Value** |
|---|---|
| Number of hidden layers | 1 |
| Number of neurons in the hidden layer | 10 |
| Number of neurons in output layer | 10 |
| Transfer function | Linear transfer function |
| Training function | Gradient descent backpropagation with adaptive learning |
| | 10000 |
| Maximum number of epochs | Mean squared error |
| Performance function | 0.00001 |
| Error goal | 0.1 |
| Backprpagation learning rate | Values generated randomly between 0 and 1 |
| Initial weights and biased term values | |

Table 4: Parameters used for creating the Elman back propagation network

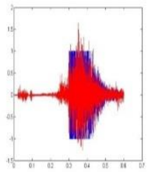
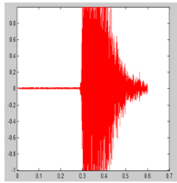
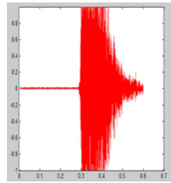
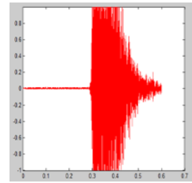
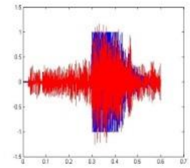
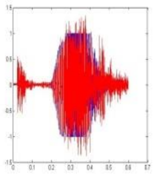
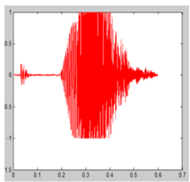
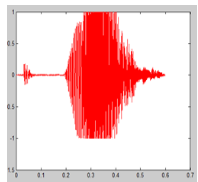
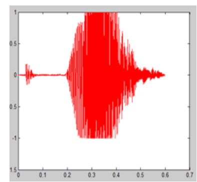
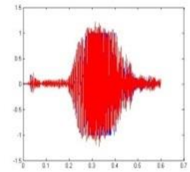
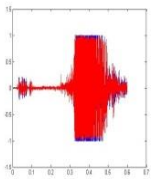
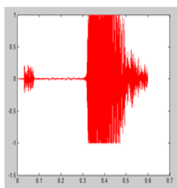
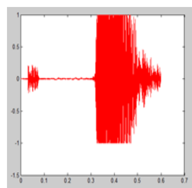
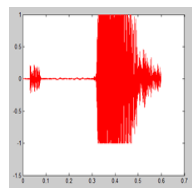
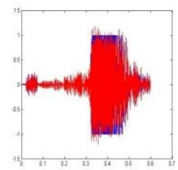
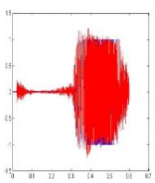
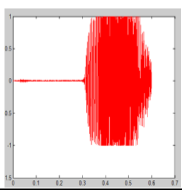
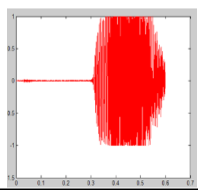
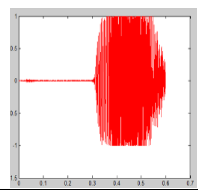
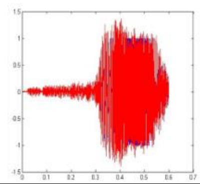
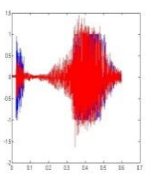
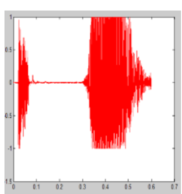
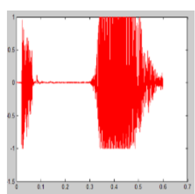
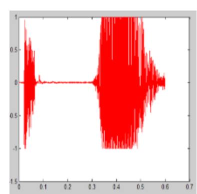
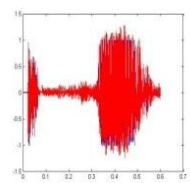
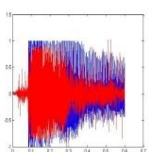
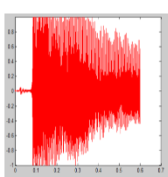
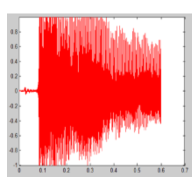
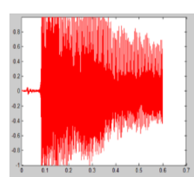
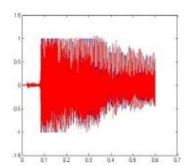
## IV. RESULTS AND DISCUSSIONS

In the proposed simulation, the performance of all the five neural network models are analyzed for created training pattern vectors and test pattern vectors of speech signals. The results presented in the simulation are considered from the selected feed-forward multilayer neural network models. Performance of these neural network models for the training patterns are presented in table 5. Performances are presented on the basis of regression value after the complete training cycle.

| Network | Signal | | | | |
|---|---|---|---|---|---|
| **Multilayer feed-forward network** | **A1** | **B1** | **C1** | **D1** | **E1** |
| | 0.6862 | 0.8699 | 0.9717 | 0.975 | 0.8364 |
| | **A2** | **B2** | **C2** | **D2** | **E2** |
| | 0.7183 | 0.9495 | 0.9718 | 0.9032 | 0.9697 |
| **Radial basis function network** | **A1** | **B1** | **C1** | **D1** | **E1** |
| | 1 | 1 | 1 | 1 | 1 |
| | **A2** | **B2** | **C2** | **D2** | **E2** |
| | 1 | 1 | 1 | 1 | 1 |
| **Exact radial basis function network** | **A1** | **B1** | **C1** | **D1** | **E1** |
| | 1 | 1 | 1 | 1 | 1 |
| | **A2** | **B2** | **C2** | **D2** | **E2** |
| | 1 | 1 | 1 | 1 | 1 |
| **Cascade feed-forward network** | **A1** | **B1** | **C1** | **D1** | **E1** |
| | 1 | 1 | 1 | 1 | 1 |
| | **A2** | **B2** | **C2** | **D2** | **E2** |
| | 1 | 1 | 1 | 1 | 1 |
| **Elman Backpropagation network** | **A1** | **B1** | **C1** | **D1** | **E1** |
| | 0.6983 | 0.9606 | 0.9528 | 0.9642 | 0.9552 |
| | **A2** | **B2** | **C2** | **D2** | **E2** |
| | 0.947 | 0.6679 | 0.9303 | 0.8603 | 0.8712 |

Table 5: Regression value of training pattern vectors for all signals of the selected networks

The performance of these networks for the pattern vectors of input signals can also represent graphically in signal form. These signals are obtained as simulated output from the trained neural network models after presenting the digital input signals as shown in figure 6. Figure 6 is representing both the signals i.e., simulated output signal and actual input signal.

| Signal | Multilayer feed-forward network | Radial basis function network | Exact radial basis function network | Cascade feed-forward network | Elman backpropagation network |
|---|---|---|---|---|---|
| **A1** |  |  |  |  |  |
| **B1** |  |  |  |  |  |
| **C1** |  |  |  |  |  |
| **D1** |  |  |  |  |  |
| **E1** |  |  |  |  |  |
| **A2** |  |  |  |  |  |

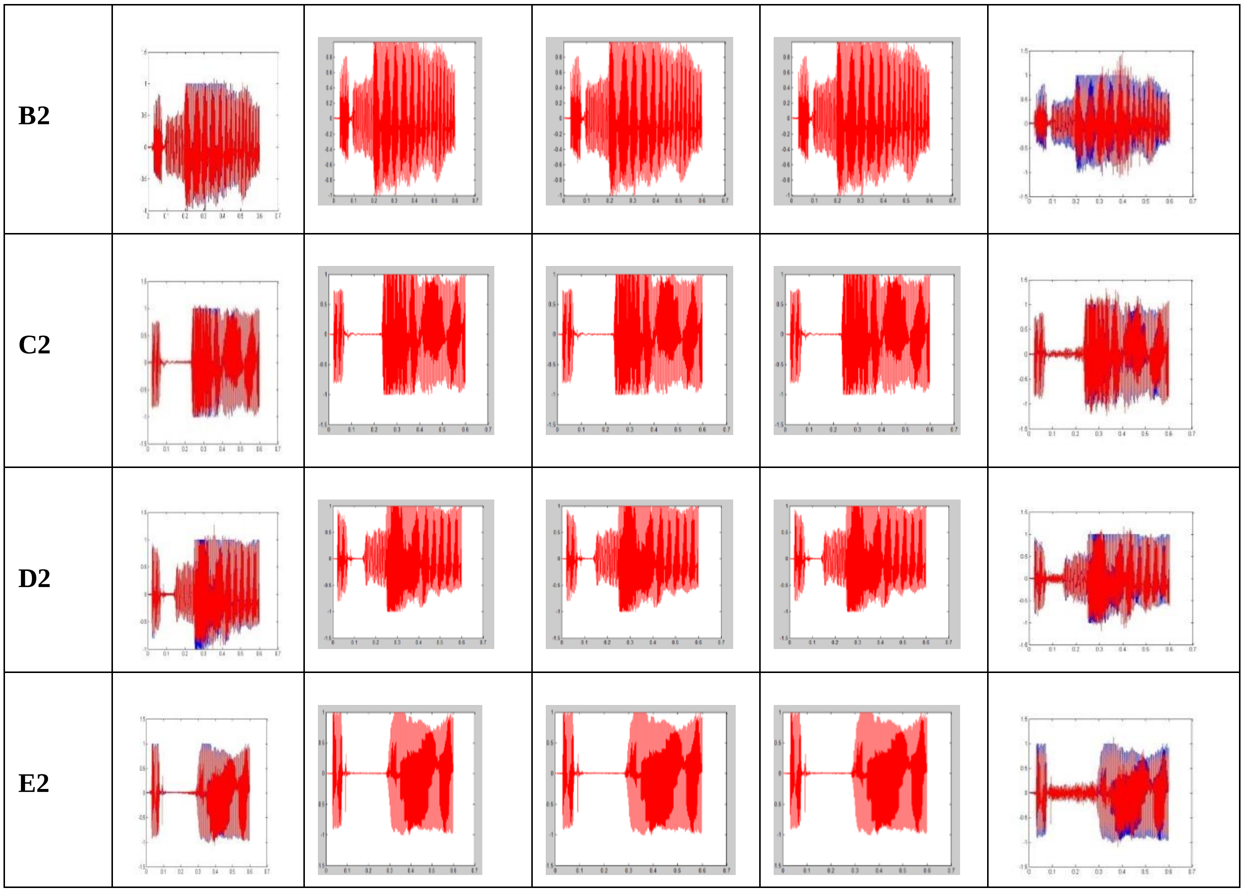



Fig.6: Graphical representation of the performances of networks

To analyze the performance of trained neural networks for their generalized classification behavior the testing pattern vectors are created by introducing 10%, 20%, 30%, 40%, 50%, 60% and 70% error or noise in training pattern vectors of all signals. Performances of neural networks for test pattern vectors are presented in table 6.

| Signal | Presence of error or noise | Multilayer Feed-forward Network | Radial Basis Function Network | Exact Radial Basis Function Network | Cascade Feed-forward Network | Elman Backpropa-gation Network |
|---|---|---|---|---|---|---|
| A1 | 10% | .70377 | .99039 | .99681 | 1 | .8588 |
| | 20% | .70377 | .9766 | .98878 | 1 | .8588 |
| | 30% | .70377 | .96511 | .98555 | 1 | .8588 |
| | 40% | .70377 | .95264 | .09767 | 1 | .8588 |
| | 50% | .70155 | .01541 | .05473 | .9976 | .85697 |
| | 60% | .47603 | .01541 | .00477 | .71216 | .59273 |
| | 70% | .16863 | .01541 | .00477 | .31699 | .23158 |
| B1 | 10% | .75584 | .08975 | .44955 | 1 | .93475 |
| | 20% | .75584 | .0829 | .44109 | 1 | .93475 |
| | 30% | .75884 | .08065 | .43502 | 1 | .93475 |
| | 40% | .7486 | .00273 | .11749 | .97976 | .92034 |
| | 50% | .61433 | .00273 | .11749 | .79901 | .75698 |
| | 60% | .36273 | .00273 | .11749 | .50045 | .44963 |
| | 70% | .05019 | .00273 | .11749 | .1045 | .07623 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **C1** | 10% | .76881 | .01107 | .27175 | 1 | .95643 |
| | 20% | .76881 | .00003 | .09201 | 1 | .95643 |
| | 30% | .76881 | .00102 | .09037 | 1 | .95643 |
| | 40% | .76881 | .003 | .08314 | 1 | .95643 |
| | 50% | .76881 | .00398 | .08303 | 1 | .95643 |
| | 60% | .66084 | .00398 | .02553 | .85845 | .81137 |
| | 70% | .39857 | .00398 | .02553 | .57546 | .52031 |
| **D1** | 10% | .7715 | .9865 | .99448 | 1 | .94671 |
| | 20% | .7715 | .97951 | .99172 | 1 | .94671 |
| | 30% | .7715 | .97353 | .98894 | 1 | .94671 |
| | 40% | .7715 | .96539 | .98685 | 1 | .94671 |
| | 50% | .7715 | .95859 | .98302 | 1 | .94671 |
| | 60% | .73883 | .00696 | .01754 | .93385 | .89039 |
| | 70% | .60227 | .00696 | .01754 | .76863 | .72655 |
| **E1** | 10% | .76559 | .00715 | .0235 | .96551 | .91292 |
| | 20% | .76559 | .00715 | .0235 | .96551 | .91292 |
| | 30% | .76559 | .00715 | .0235 | .96551 | .91292 |
| | 40% | .76559 | .00715 | .0235 | .96551 | .91292 |
| | 50% | .76559 | .00715 | .0235 | .96551 | .91292 |
| | 60% | .71062 | .00715 | .0235 | .88832 | .84324 |
| | 70% | .51039 | .00715 | .0235 | .64813 | .60267 |
| **A2** | 10% | .64027 | 1 | .93127 | 1 | .77752 |
| | 20% | .58488 | 1 | .00076 | .92155 | .71603 |
| | 30% | .48806 | 1 | .00076 | .79738 | .60804 |
| | 40% | .43487 | 1 | .00076 | .6919 | .52623 |
| | 50% | .34327 | 1 | .00076 | .55598 | .42113 |
| | 60% | .26023 | 1 | .00076 | .41866 | .3154 |
| | 70% | .20727 | 1 | .00076 | .3316 | .24509 |
| **B2** | 10% | .63424 | .00445 | .01298 | .99447 | .83306 |
| | 20% | .63338 | .00445 | .01298 | .98817 | .82799 |
| | 30% | .63766 | .00445 | .01298 | .98297 | .83385 |
| | 40% | .59011 | .00445 | .01298 | .91265 | .77964 |
| | 50% | .5094 | .00445 | .01298 | .80685 | .69141 |
| | 60% | .43029 | .00445 | .01298 | .69128 | .59632 |
| | 70% | .34165 | .00445 | .01298 | .55804 | .4777 |
| **C2** | 10% | .69589 | .00076 | 1 | .96517 | .88741 |
| | 20% | .69467 | .00076 | 1 | .95954 | .88258 |
| | 30% | .69467 | .00076 | 1 | .95954 | .88258 |
| | 40% | .69053 | .00076 | 1 | .95596 | .88029 |
| | 50% | .58352 | .00076 | 1 | .83151 | .7536 |
| | 60% | .48048 | .00076 | 1 | .70695 | .63567 |
| | 70% | .41278 | .00076 | 1 | .60189 | .54029 |
| | 10% | .63109 | .01973 | .03040 | .96503 | .8414 |
| | 20% | .62871 | .01973 | .03040 | .96223 | .83981 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **D2** | 30% | .63148 | .01973 | .03040 | .96083 | .83998 |
| | 40% | .6319 | .01973 | .03040 | .94356 | .83812 |
| | 50% | .55621 | .01973 | .03040 | .84801 | .7491 |
| | 60% | .45592 | .01973 | .03040 | .71375 | .61861 |
| | 70% | .37579 | .01973 | .03040 | .60172 | .50896 |
| **E2** | 10% | .67228 | .00173 | .19291 | .9573 | .85681 |
| | 20% | .65794 | .00173 | .19291 | .93661 | .8381 |
| | 30% | .65794 | .00173 | .19291 | .93661 | .8381 |
| | 40% | .65794 | .00173 | .19291 | .93661 | .8381 |
| | 50% | .65935 | .00173 | .19291 | .93661 | .83851 |
| | 60% | .5906 | .00173 | .19291 | .83872 | .74309 |
| | 70% | .5153 | .00173 | .19291 | .72876 | .64437 |

Table 6: Signal-wise regression values of all selected networks for test pattern vectors

Accuracy of recognition for test pattern vectors is taken as a measurement to analyze the performance of the neural networks. Therefore, to do a comparative analysis for the performances of network models, the average of recognition accuracy is calculated for all signals with all chosen percentage of errors respectively as shown in table 7.

| Network | Average percentage of error | | | | | | |
|---|---|---|---|---|---|---|---|
| | 10% | 20% | 30% | 40% | 50% | 60% | 70% |
| Multilayer Feed-forward Network | .70393 | .69651 | .68753 | .67636 | .62735 | .51666 | .35828 |
| Radial Basis Function Network | .31115 | .30729 | .30541 | .29576 | .20145 | .10633 | .10629 |
| Exact Radial Basis Function Network | .49037 | .37742 | .37604 | .25457 | .24988 | .14259 | .14259 |
| Cascade Feed-forward Network | .98475 | .97336 | .96028 | .9386 | .87412 | .72626 | .52357 |
| Elman Backprop-agation Network | .88058 | .87141 | .86122 | .84576 | .78838 | .64965 | .45738 |

Table 7: Average percentage of recognition accuracy with chosen percentage of errors of networks

Hence, the performance analysis is considered by comparing the average of regression value between simulated output of the network performance for test pattern signal and expected output for the input signal. This comparison is presented in figure7.

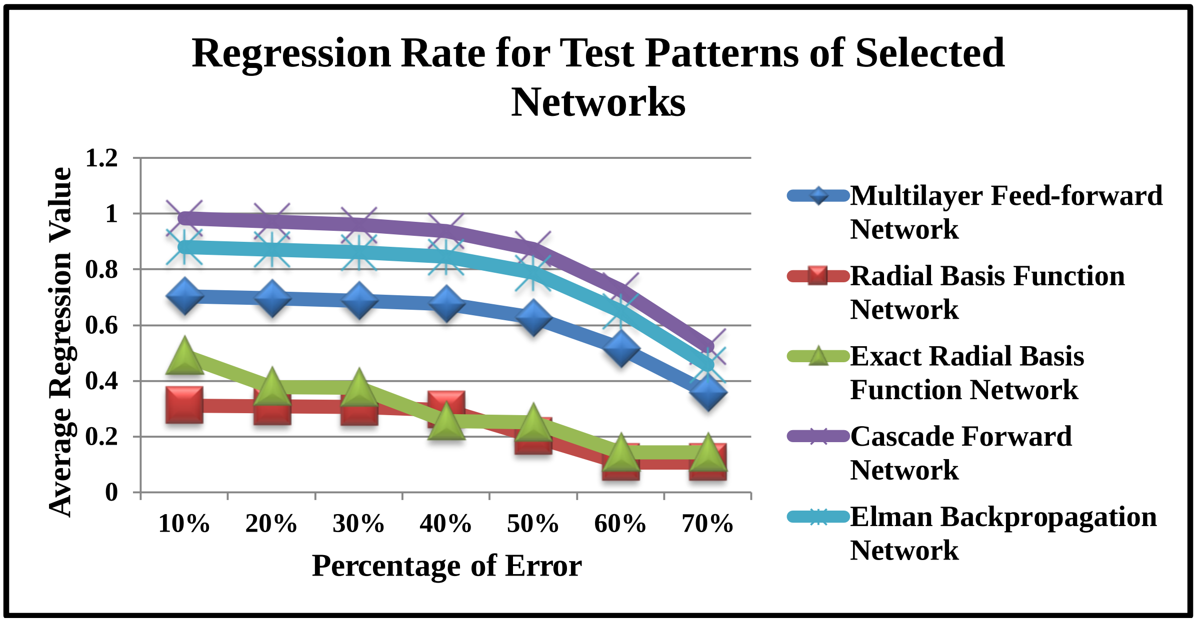



Fig.7: Comparison of regression values of testing patterns for all selected neural network models

Tables 5 and 6 are exhibiting the comparison of performance of the selected neural network models for the training and testing pattern signals respectively. Results mentioned in table 5 exhibit that ,Radial basis function network model, Exact radial basis function network model and Cascade feed-forward network model show 100% approximation for all the training signal pattern vectors. Results also exhibit that Multilayer feed-forward network model performs better than Elman back propagation network model for training input signals. The interesting results are obtained during the simulation, which show that the multilayer feed-forward network model shows better approximation for most of the speech signals which are spoken slowly, i.e. B2, C2, D2 and E2; whereas Elman back propagation network model show better approximation for most of the speech signals spoken fast, i.e. A1, B1 and E1.

Results indicate that the Cascade feed-forward neural network model gives highest average recognition accuracy for all test pattern signal vectors as compared to other network models. For test patterns, the highest recognition accuracy is exhibited by the network is 100% and the lowest recognition accuracy showed by the network is 10%for signal B1 with 70% error in signal, which is better than the lowest recognition accuracy value given by other network models. Thus, network is behaving as good approximator and classifier for the speech signals.

Results show that Radial basis function network performs with 100% recognition accuracy only for the training patterns of signals; but for test pattern signals, network shows poor classification accuracy for all test patterns if noise or error in signal is more than 10%. Results also display that Exact radial basis function network gives 100% recognition accuracy for all training patterns and all test pattern vectors of signal C2, while 90% or above recognition accuracy for few test patterns of signals A1, C1, D1and A2; but for most of the test pattern vectors, it shows reasonably low recognition accuracy. Thus, overall performances of Radial basis function network and Exact radial basis function network are poor for test pattern vectors, so these network models are not generalized well for most of the test pattern vector signals. Hence, Radial basis function network does not perform well for signal classification due to its large sample size or for large number of features in the input data.

Results indicate that Elman back propagation neural network model exhibit above 85% classification accuracy for signals A1, B1, C1, D1, E1, C2 and E2. Results also indicate that this network gives approximately same recognition accuracy for most of the signals even though the input signals contain 10% - 50% noise. It can also be seen from table5 that Multilayer feed-forward network and Elman back propagation neural network shows better recognition accuracy for few test patterns than the training patterns for signals A1 and A1 & B1 respectively.

The simulated results are clearly indicating the better signal classification for Cascade feed-forward and Elman back propagation neural networks and bad performance for Radial basis function networks. The Elman back propagation and Cascade feed-forward neural networks performs well for signal classification problems whose sample size is large.

## V. CONCLUSION

In this paper we analyzed the performance of five feed-forward neural network models trained with variants of back propagation algorithm like Multilayer feed-forward network, Radial basis function network, Exact radial basis function network, Cascade feed-forward network and Elman back propagation network for the classification of speech signals of first five alphabets of the English language. Training pattern vectors are created by applying digital signal processing operations like sampling, quantization and coding to convert the continuous analog speech signal to discrete-time signal and digital signal respectively. Test pattern vectors are created by introducing 10%, 20%, 30%, 40%, 50%, 60% and 70% noise or error respectively in the input signals used for training. Simulated results of the performance evaluation of the selected networks are presented and discussed. The following observations have been drawn from the simulated performance evaluation.

- The simulated results are showing that the Cascade feed-forward neural network is exhibiting good signal classification for the training and test pattern vectors. The network shows 100% recognition accuracy for all the training patterns and for test patterns with noise percentage from 10% to 40%, network gives approximately the same recognition accuracy. Also, network is exhibiting above 50% recognition accuracy for most of the test patterns with50%, 60% and 70% noise. The highest and lowest recognition accuracy presented by the network is 100% and 10% respectively. The average test pattern recognition accuracy shown by the network for each of the selected percentage of noise is better than other networks selected for the simulation. Thus, the cascade feed-forward neural network performs better for signal classification whose samples are of large size.
- The simulated results are also exhibiting that the Radial basis function neural network model shows 100% recognition accuracy for training pattern vectors and all test pattern vectors created for signal A2. The average test pattern recognition accuracy of the network is below 35%. Thus, the network is showing poor behavior of generalization for large data samples.
- Simulated results are also showing that the performance of Exact radial basis function network is better than the Radial basis function network. Network model shows 100% recognition accuracy for training pattern vectors and all test pattern vectors created for the signal C2. The lowest test pattern recognition accuracy given by the system is .076% for signal A2. Hence, like Radial basis function network, this network also exhibit poor generalization for large signal samples.
- Simulation results also indicate that Elman back propagation network shows almost 80% recognition accuracy for most of the test patterns with 10%-60% noise in input signal. Performance of the network degraded when percentage of noise become 70%. The highest and lowest recognition accuracy presented by the network is 95% and 7% respectively. The average test pattern recognition accuracy is good for patterns with 10%-40% error.
- It can also be seen from the results that Multilayer feed-forward network, Cascade feed-forward network and Elman back propagation network shows highest recognition accuracy for speech signals spoken fast, whereas Radial basis function network and Exact radial basis function network show highest recognition rate for speech signals spoken slowly.
- It is clear from the results that the networks with extra connection/sin layers like Cascade feed-forward network and Elman back propagation network perform well for speech signals collected for the experiment. Thus, the Cascade feed-forward network and Elman back propagation networks are found most optimal model for speech classification problem, even if the input signals contain more than 50% noise from the original signal that has been used for training. Therefore, for the large samples of signals Cascade feed-forward network and Elman back propagation network are recommended for signal classification with better accuracy.

## VI. REFERENCES


[1]. S. Katagiri, "Applications of Artificial Neural Networks (ANNs) to Speech Processing", Chapter of Handbook of Neural Network Signal Processing, Edited by Yu Hen Hu and Jenq-Neng Hwang, CRC Press, 2002.

[2]. G.Wouter, G. Tsenov and V. Mladenov, "Neural Networks used for Speech Recognition", Journal of Automatic Control, pg. 1-7, Vol. 20, 2010.

[3]. S. Furui, "50 years of Progress in Speech and Speaker Recognition Research", ECTI Transactions on Computer and Information Technology,Vol.1, No.2, 2005.

[4]. R. Stuckless, "Developments in Real-time Speech-to-text Communication for People with Impaired Hearing", In M Ross (Ed.), Communication Access for People with Hearing Loss, pp.197-226, MD: York Press, 1994.

[5]. J. K. Basu, D. Bhattacharyya and T. Kim, "Use of Artificial Neural Network in Pattern Recognition", International Journal of Software Engineering and Its Applications, pg. 23-34, Vol. 4, No. 2, 2010.

[6]. G. Potamianos, C. Neti, G. Gravier, A. Garg and A. W. Senior, " Recent Advances in the Automatic Recognition of Audio-Visual Speech', Proceedings of the IEEE, pp. 1-18, Vol. 91, No. 9, 2003.

[7]. H. Yashwanth, H. Mahendrakar and S. David, "Automatic Speech Recognition Using Audio Visual Cues", IEEE India Annual Conference, pp. 166-169, 2004.

[8]. P. Rani, S. Kakkar and S. Rani, "Speech Recognition Using Neural Network", In Proceedings of International Conference

on Advancements in Engineering and Technology, International Journal of Computer Applications, pp. 11-14, 2015.

[9]. N. Srivastava, "Speech Recognition Using Artificial Neural Network", International Journal of Engineering Science and Innovative Technology (IJESIT), pg. 406-412, Vol. 3, Issue 3, 2014.

[10]. S. K. Gaikwad, B. W. Gawali and P. Yannawar, "A Review on Speech Recognition Technique", International Journal of Computer Applications, pg. 16-24, Vol. 10, No. 3, 2010.

[11]. M. A. Anusuya and S. K. Katti, "Speech Recognition by Machine: A Review", International Journal of Computer Science and Information Security, pg. 181 – 205, Vol. 6, No. 3, 2009.

[12]. Z. N.Karam and W. M.Campbell "A New Kernel for SVM MIIR Based Speaker Recognition", MIT Lincoln Laboratory,Lexington, MA, USA.

[13]. T. Landauer, C. Kamm, and S. Singhal, "Learning a Minimally Structured Backpropagation Network to Recognize Speech," In Proceedings of Ninth Annual Conference of Cognitive Science Soc., pp. 531–536, 1987.

[14]. L. Deng and X. Li, "Machine Learning Paradigms for Speech Recognition", IEEE Transactions on Audio, Speech and Language Processing, pg. 1-30, Vol. 21, No. 5, 2013.

[15]. C. Bregler, S. Manke, H. Hild and A. Waibel, "Bimodal Sensor Integration on the Example of Speech-Reading", In the Proceedings of ICNN, pg. 667-677, Vol. 11, 1993.

[16]. S. Furui, "Digital Speech Processing, Synthesis, and Recognition", Marcel Dekker Inc. Publications, New York, 1989.

[17]. J. D. Markel, "The SIFT Algorithm for Fundamental Frequency Estimation", IEEE Transactions Audio Electroacoustic, pp. 367-377, Vol. AU-20, 1972.

[18]. M. M. Sondhi, "New Methods of Pitch Extraction", IEEE Transactions Audio Electroacoustic, pp. 262-266, Vol. AU-16, 1968.

[19]. H. Dudley, "The Vocoder", Bell Labs Record, pp. 122-126, Vol. 17, 1939.

[20]. B. S. Atal and L. R. Rabiner, "A Pattern Recognition Approach to Voiced-Unvoiced-Silence Classification with Applications to Speech Recognition", IEEE Transactions on Acoustics, Speech and Signal Processing, pp 201-212, Vol. ASSP-24, No. 3, 1976.

[21]. G.E. Hinton, T. Sejnowski and D.H. Ackley, "Boltzmann Machines Constraint Satisfaction Networks That Learn", Technical Report, CMU-CS-84-119, Carnegie Mellon University, 1981.

[22]. R.W. Prager, T.D. Harrison and F. Fallside, "Boltzmann Machines for Speech Recognition", Computer, Speech and Language, pp. 3-27, Vol. 1, Academic Press, 1986.

[23]. X. Cui et.al., "A Study of Variable-Parameter Gaussian Mixture Hidden Markov Modeling for Noisy Speech Recognition", IEEE Transactions On Audio, Speech, And Language Processing, Vol. 15, No. 4, 2007.

[24]. G.E. Dahl, M. Ranzato, A. Mohamed and G.E. Hinton, "Phone Recognition with the Mean-covariance Restricted Boltzmann Machine", Adv. Neural Inf. Process.Syst., No. 23, 2010.

[25]. D. Yu, L. Deng and G. Dahl, "Roles of Pre-training and Fine-tuning in Context-dependent DBN-HMMs for Real-world Speech Recognition", In Proceedings of NIPS Workshop Deep Learn, Unsupervised Feature Learn, 2010.

[26]. H. Bourland and C.J. Wellekens, "Multilayer Perceptrons and Automatic Speech Recognition", IEEE First International Conference on Neural Networks, San Diego, California IV-407-IV-416, June 21-24, 1987.

[27]. H.Bourland and C.J. Wellekens, "Speech Pattern Discriminations and Multilayer Perceptrons", Computer, Speech and Language, 1987.

[28]. Robinson and F. Fallside, "A Recurrent Error Propagation Network Speech Recognizer System", Computer, Speech and Language, Vol. 5, No. 3, 1991.

[29]. R. Shantha, D. Sugumar and V. Sadasivam, "Audio Signal Classification Based on Optimal Wavelet and Support Vector Machine", IEEE International Conference on Computational Intelligence and Multimedia Applications, pp. 544-548, Vol. 2, 2007.

[30]. S. A. A. Shah, A. Asar and S.F. Shukat, " Neural Network Solution for Secure Interactive Voice Response", World Applied Sciences Journal, pg. 1264-1269, Vol. 6, No. 9, 2009.

[31]. L. Yang and Z. Yang, "Study on Audio Signal's Classification Based on BP Neural Network", IEEE Conference Publications on Artificial Intelligence, Management Science and Electronic Commerce, pp. 5153-5155, 2011.

[32]. S. Balochian, E. A.Seidabad and S. Z. Rad, " Neural Network Optimization Genetic Algorithms for the Audio Classification to Speech and Music", International Journal of Signal Processing, Image Processing and Pattern Recognition, pg. 47-54, Vol. 6, No. 3, 2013.

[33]. Md. A.Hossain, Md. M. Rahman, U. K.Prodhan and Md. F. Khan, " Implementation of Back-Propagation Neural Network for Isolated Bangla Speech Recognition", International Journal of Information Sciences and Techniques, pg. 1-9, Vol. 3, No. 4, 2013.

[34]. O.Abdel-Hamid, A.R. Mohamed, H. Jiang, L. Deng, G. Penn and D. Yu, " Convolutional Neural Networks for Speech Recognition", IEEE/ ACM Transactions on Audio, Speech and Language Processing, pg. 1-13, Vol. 22, No. 10, 2014.

[35]. S. M Kuo and B. H Lee, "Real time Digital Signal Processing-Implementations, Applications and Experiments with TMS320C55X", John Wiley & Sons Publications Ltd., 2001.

[36]. S. W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing", California Technical Publishing, Second Edition, 1999.

[37]. J. G. Proakis and D. G. Manolakis, "Digital Signal Processing – Principles, Algorithms and Applications", Prentice Hall Publication, Third Edition, 2005.

[38]. D. Manolakis and V. Ingle, "Applied Digital Signal Processing – Theory and Practice", Cambridge University Press, First Edition, 2011.

[39]. T. Lefteri H. and A. U. Robert, "Fuzzy and Neural Approaches in Engineering", John Wiley and Sons Publications, 1997.

[40]. R. Hecht-Nielsen, "Theory of Backpropagation Neural Network", International Joint Conference on Neural Networks, pp. 593-605, Vol. 1, 1989.

[41]. B. Yagnanarayana, "Artificial Intelligence", Prentice Hall Publications, Ninth Edition, 2004.

[42]. O.D. Jesus and M. T. Hagan, "Backpropagation Algorithms for a Broad Class of Dynamic Networks", IEEE Transactions on Neural Networks, pg. 14-27, Vol. 18, No. 1, 2007.

[43]. S. Kumar and A. Rawat, " Implementation of Feed-forward Neural Network Models for Pattern Classification Using Transformation Based Feature Extraction Methods", International Journal of Scientific Research in Science and Technology, pg. 119-126, Vol. 2, Issue 3, 2016.

[44]. M.J.D. Powell, "Radial Basis Functions for Multivariate Interpolation: A Review", In Algorithms for the Approximation of Functions and Data, J.C. Mason and M.G. Cox, eds., Clarendon Press, pp. 143-167, 1987.

[45]. Y. Diao and K. M. Passino, "Adaptive Neural / Fuzzy Control for Interpolated Nonlinear Systems", IEEE Transactions on Fuzzy Systems, pg. 583-595, Vol. 10, No. 5, 2002.

[46]. S. N. Sivanandam and S.N. Deepa, "Principles of Soft Computing", Wiley India Publications, Second Edition, 2012.

[47]. S.Chen, C.F.N.Cowan and P. M. Grant "Orthogonal Least Square Learning Algorithm for Radial Basis Function Networks", IEEE Transactions on Neural Networks, pg. 302-309, Vol.2, No. 2, 1991.

[48]. S. N. Parappa and M. P. Singh, "Conjugate Descent of Gradient Descent Radial Basis Function for Generalization of Feed-forward Neural Network", International Journal of Advancements in Research & Technology, pg. 112-125, Vol. 2, Issue 12, 2013.

[49]. Ah-H.Tan, "Cascade ARTMAP:Integrating Neural Computation and Symbolic Knowledge Processing", IEEE Transactions on Neural Networks, pg. 237-250, Vol. 8, No. 2, 1997.

[50]. J. L. Elman, "Finding structure in Time", Cognitive Science, pg. 179-211, Vol. 14, 1990.

[51]. A.Gruning, "Elman Backpropagation as Reinforcement for Simple Recurrent Networks", Neural Computation, IEEE Transactions, pg. 3108-3131, Vol. 19, Issue 11, 2007.

[52]. D. T. Pham and X. Liu, "Training of Elman Networks and Dynamic System Modelling", International Journal of Systems Science, pg. 221-226, Vol. 27, No. 2, 1996.


Dr. Vikas Pandey PH.D in computer scince from University of Rdvv,jabalpur (m.p), india in 2015 and Master of Computer Application from Rdvv,jabalpur (m.p), india in year 2009. currently working as Assistant Professor in Department of Computer Science& Application Makhanlal Chaturvedi National University of Journalism and Communication (MCNUJC), Bhopal(M.P) India. I have interest research area in image processing, wavelet application & neural network.