Available online at www.elixirpublishers.com (Elixir International Journal)

Advanced Engineering Informatics

Elixir Adv. Engg. Info. 64 (2013) 19398-19400

ISSN: 2229-712X

Artificial Neural Network based Text Dependent Continuous Speech Recognition System

Suma Swamy* and K.V Ramakrishnan

Department of Electronics & Communications Engineering, Anna University, Chennai.

ARTICLE INFO

Article history:

Received: 4 September 2013;

Received in revised form:

5 November 2013;

Accepted: 16 November 2013;

Keywords

ASR,

MFCC,

VQ,

LPC,

BNN.

ABSTRACT

Speaker identification followed by speech recognition system is developed. The system makes use of MFCC (mel frequency cepstrum coefficients) to process the input signal and extract the features. VQ (Vector quantization) is used to identify the speaker. LPC (Linear Predictive Coding) and BNN (Back Propagation Neural Network) technique of hyperbolic tangent function under ANN (Artificial Neural Network) is used for speech recognition system. The implementation is done using MATLAB. The results of the developed system proved to be efficient and faster.

Introduction

A Speaker Identification system uses digitized speech signal [1] and consists of two main modules, viz. a speaker specific feature extractor followed by a speaker modelling technique for generalized representation of extracted features. [2].

The speech recognition system follows speaker identification. Depending upon the speaker, the database is divided into speaker specific regions. Hence this serves as an authenticator for the existing speakers. This also makes the system faster and efficient.

Proposed Model

An efficient Speaker Identification followed by Speech Recognition system is shown in figure.1.

Speaker Identification

Preprocessing

In preprocessing, digital filtering and endpoint detection is done. Filtering is to eliminate/reduce ambient noise. A conventional short time or spectral energy based endpoint detection algorithm is implemented. It is very sensitive to speech artifacts and break down quickly in the presence of noise. Endpoint detection is a process of selecting only a desired speech interval.

To achieve good performance pitch, duration information, use of adaptive thresholds and zero crossings rate resulted in improved performance.

Feature Extraction

MFCC algorithm is used for feature extraction. It is a representation of a short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a non-linear Mel scale of frequency.

Pattern Matching

VQ is used for pattern matching. Clustering the speaker's feature vectors in K non-overlapping clusters forms speaker models. The codebook effectively reduces the amount of data by preserving the essential information of the original distribution [3]. The matching function in VQ-based speaker recognition is typically defined as the quantization distortion between two vector sets.

The feature vectors are passed to the decision logic and then the speaker is identified [4].

Speech Recognition

Preprocessing and Feature Extraction

The speech signal is fed into a pre-processor, which generates a test template based on the features. The test template is then compared with a set of pre-stored reference templates. The reference template that most closely matches the test template is determined based on predetermined decision rules [5].

The preprocessing involves several steps: normalization, parameterization and feature extraction.

Normalization-This step attempts to eliminate the variability in the input speech signal.

Parameterization-This step involves converting the speech signal into a set of statistical parameters that maximize the likelihood of choosing the right output features corresponding to the input signal.

Feature Extraction- Segmenting the speech signal is done by a method that divides the speech signal into a fixed number of frames with variable length. This is done to deal with the non-uniform word length in the speech recordings. The digitized speech signal is segmented into 10 frames of 10 to 40 milliseconds each depending on the length of a given word in the vocabulary. Each frame is weighted by a hamming window [5].

LPC method is used for feature extraction. Covariance method is used to compute the LPC coefficients, which require solving a set of least square based normal equations.

Pattern Matching

BNN is used for pattern matching. The input vector to the back-propagation neural network consists of linear predictive coding coefficients, error variance, short time energy function,

Tele:

E-mail addresses: suma_swamy@yahoo.com

short time average zero crossing rate and voiced/unvoiced discrimination.

Multi-Layered Perceptron *(MLP)* has been adapted for speech recognition. A general neural structure is shown in figure 2. MLP consists of three layers: an input layer, an output layer, and an intermediate or hidden layer.

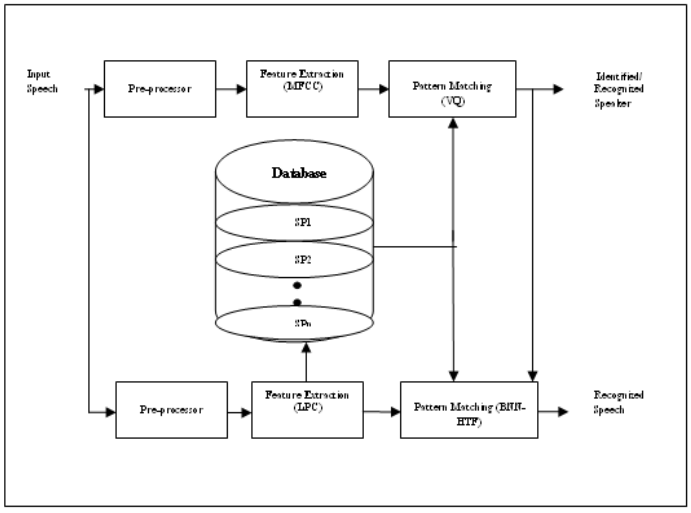


Figure.1 Speaker Identification followed by Speech Recognition

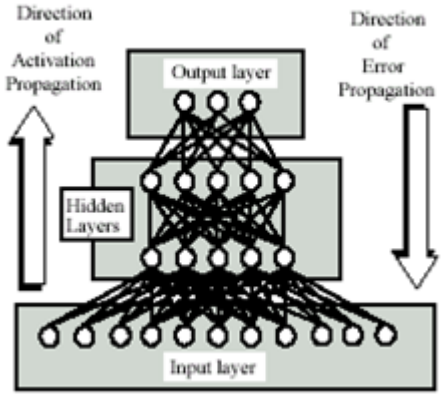


Figure 2. Multilayer Perception [4]

Processing elements or neurons in the input layer only act as buffers for distributing the input signal x_i to neurons in the hidden layer. Each neuron j in the hidden layer sums up its input signals x_i after weighting them with the strengths of the respective connections w_{ji} from the input layer and computes its output y_j as a function f of the sum, viz., [5, 6]

$$Y_j=f(\Sigma w_{ji} x_i) \quad \textbf{(1)}$$

Training a network consists of adjusting its weights using a training algorithm. The training algorithms adopted in this study optimize the weights by attempting to minimize the sum of squared differences between the desired and actual values of the output neurons, namely:

$$E= \tfrac{1}{2} \Sigma (Y_{dj}-Y_j)^2 f\ 1 \quad \textbf{(2)}$$

Where Y_{dj} is the desired value of output neuron j and Yj is the actual output of that neuron. Each weight w_{ji} is adjusted by adding an increment $.w_{ji}$ to it. w_{ji} is selected to reduce E as rapidly as possible. The adjustment is carried out over several training iterations until a satisfactorily small value of E is obtained or a given number of epochs are reached. The computation of w_{ji} depends on the training algorithm adopted. Training process is ended when the maximum number of epochs is reached. .The learning algorithm used in this work is summarized briefly.

Back Propagation Using Hyperbolic Tangent Function

1) ***Training Set***- a collection of input-output patterns that are used to train the network.

2) ***Testing Set***- a collection of input-output patterns that are used to assess network performance.

3) ***Learning Rate***- a scalar parameter, analogous to step size in numerical integration, is used to set the rate of adjustments.

4) ***Network Error***-

Total-Sum-Squared-Error (TSSE) 2[4]

$$TSSE= 0.5 \sum_{patterns} \sum_{outputs} (desired - actual)^2 \quad \textbf{(3)}$$

Root-Mean-Squared-Error (RMSE)-

$$RMSE = \sqrt{\frac{2* TSSE}{\#patterns * \# outputs}} \quad \textbf{(4)}$$

The Back Propagation algorithm looks for the minimum of the error function in weight space using the method of gradient descent. The combination of weights that minimizes the error function is considered to be a solution of the learning problem. These neurons receive signals from the neurons in the preceding layer, l-1. The net input at any neuron is given by the following formula

$$n_j^{(l)} = \sum a_i^{(l-1)} * w_{ij}^{(l)} + b_j^{(l)} \quad \text{where } j=1, 2, 3....N_l \quad \textbf{(5)}$$

The algorithm for Back Propagation is shown below:

1. Set α. Initialize weights and biases.
2. For step time t=1, 2…. repeat the following steps until convergence.
3. Set a (0) = x(t), randomly picked from training set.
4. For l=1, 2,…L where L=Total number of layers
Compute for n=1, 2…. N_L

$$S_{(n)}^{(L)} = 2(a_n^{(L)} - t_n(t))\ F^{(L)}(n_n^{(L)}) \quad \textbf{(6)}$$

5. For l=L-1,….,2,1 and j=1,2,…Nl, compute

$$S_j^{(l)} = F^{(l)}(nj)^{l} \Sigma w_{ji}^{(l+1)} S_i^{(l+1)} \quad \textbf{(7)}$$

6. For l=1, 2,…L update

$$w_{ij}^{(l)}(t+1)= w_{ij}^{(l)}(t) - \alpha\, a_i^{(l-1)}(t)\, s_j^{(l)}(t)$$
$$b_j^{(l)}(t+1) = b_j^{(l)}(t) - \alpha\, s_j^{(l)}(t) \quad \textbf{(8)}$$

Similarity Comparison and Detection- When neural networks accomplish the ASR (Automatic Speech Recognition), the comparison and decision steps are performed through training. The neural network is trained by presenting examples of the reference templates. A learning algorithm based on the

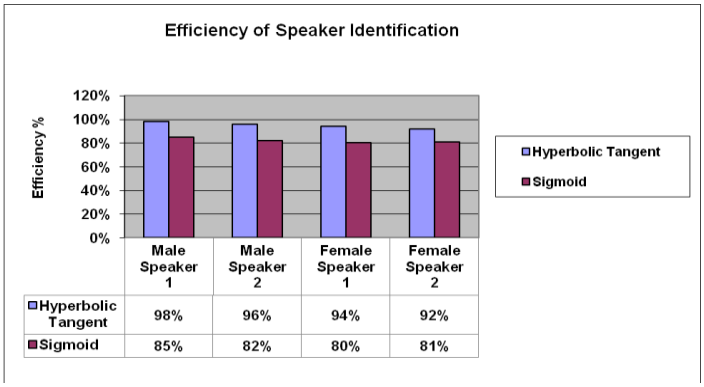
[2] MLP (Multi-Layered Perceptron)

[3] TSSE (Total-Sum-Squared-Error)
RMSE (Root-Mean-Squared-Error()

delta rule is used to minimize a global error function between the current output and the desired output. Once the network is "trained", i.e., the error or the delta weight term (w_{ijnew}– w_{ijold}) is less than a pre-set threshold, a final set of weight values are obtained and used as the network characteristics. The test signal is then passed through this network with fixed weight values; the word corresponding to the highest value in the output vector is the word being recognized. The recognition rate is then calculated based on the percentage of correctly identified outputs [5].

Experimental Results

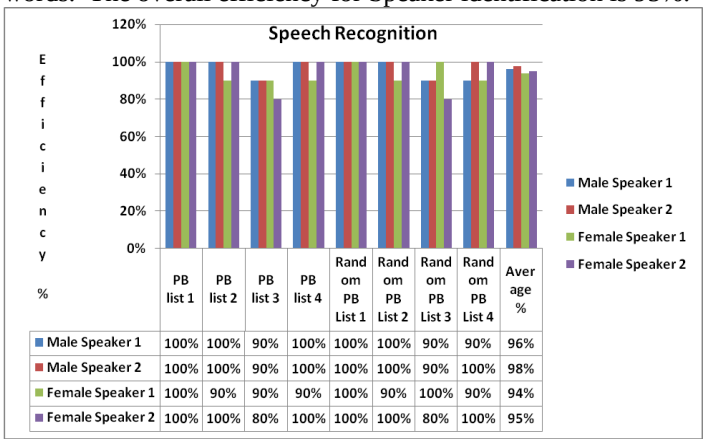
Four different speakers (2 male and 2 female) were asked to speak the set of continuous fifty phonetically balanced words from the four lists. These are recorded and stored as different wave files. The same procedure is repeated for four random lists of same phonetically balanced words. Then these files were used as speech input to the system developed. The speech recognition in % was computed for all the eight lists. The noted values were tabulated. It was compared with that of sigmoid function. It was seen that the speech recognition improved by 10-15% when back propagation neural network with hyperbolic tangent function was used.



	Male Speaker 1	Male Speaker 2	Female Speaker 1	Female Speaker 2
Hyperbolic Tangent	98%	96%	94%	92%
Sigmoid	85%	82%	80%	81%

Figure 3. Bar chart for Speaker Identification

Figure 3 shows the efficiency for Speaker Identification. Figure 4 shows speech recognition for these set of continuous words. The overall efficiency for Speaker identification is 95%.



	PB list 1	PB list 2	PB list 3	PB list 4	Random PB List 1	Random PB List 2	Random PB List 3	Random PB List 4	Average %
Male Speaker 1	100%	100%	90%	100%	100%	100%	90%	90%	96%
Male Speaker 2	100%	100%	90%	100%	100%	100%	90%	100%	98%
Female Speaker 1	100%	90%	90%	90%	100%	90%	100%	90%	94%
Female Speaker 2	100%	100%	80%	100%	100%	100%	80%	100%	95%

Figure 4. Bar chart for Speech Recognition

The overall efficiency for large vocabulary continuous speech recognition is 96%.

Conclusion

Speaker Identification is implemented using MFCC and VQ. Then the authorized speakers are used for speech recognition. LPC and BNN with hyperbolic tangent function are used for speech recognition. The efficiency is 95% for speaker Identification and 96% for. Speech recognition. Three nodes are taken in the hidden layer to obtain better efficiency. LPC and Back Propagation with sigmoid function give the efficiency of only 82% for three speakers. ANN under Hyperbolic Tangent function has enhanced efficiency in comparison with the sigmoid function. The work can be enhanced to text independent speaker identification.

Acknowledgements

The author acknowledges Visvesvaraya Technological University, Belgaum and Anna University, Chennai for the encouragement and the permission to publish this paper. The author would like to thank the Principal of Sir M Visvesvaraya Institute of Technology, Dr.M.S.Indira for her constant encouragement and would like to thank Prof. Dilip.K.Sen, Head of the Department, CSE/ISE for his invaluable guidance and suggestions from time to time.

References

[1] D. Gatica-Perez, G. Lathoud, J.-M. Odobez, and I. McCowan, Audiovisual
probabilistic tracking of multiple speakers in meetings, IEEE Transactions on Speech and Audio Processing, vol. 15, no. 2, pp. 601–616,(February 2007)

[2] Faundez-Zanuy M. and Monte-Moreno E: State-of-the-art in speaker recognition, Aerospace and Electronic Systems Magazine, IEEE, vol.20, No. 5, pp. 7-12, (March 2005)

[3] Y. Linde, A. Buzo, and R. M. Gray: An algorithm for vector quantizer design, IEEE Trans. Commun., vol. 28, no. 1, pp. 84-95,(1980)

[4] R.V Pawar, P.P.Kajave, and S.N.Mali: Speaker Identification using neural Networks, World Academy of Science, Engineering and Technology, (12 2005)

[5] Chau Giang Le: Application of a back propagation neural network to isolated-word speech recognition, (June 1993)

[6] Brian J.Love, Jennifer Vining, Xeuning Sun: Automatic Speech Recognition using Neural Networks, (Spring, 2004)

[7] Suma Swamy, Shalini T., Sindhu P. Nagabhushan, Sumaiah Nawaz, and K.V. Ramakrishnan, "Text Dependent Speaker Identification and Speech Recognition Using Artificial Neural Network", Springer CCIS Journal, Volume 269, ISSN: 1885-0929, 2012.