Neural Network Based Speaker Identification System Using Features Selection

Noor Khaled, Saad Najiam Al Saad

1Department of Computer Science, Al-Mustansiriyah University, Baghdad, Iraqi
2Associate professor, Department of Computer Science, Al-Mustansiriyah University, Baghdad, Iraqi

Abstract

In this paper, a neural-based system for speaker identification is proposed and implemented. It uses selected power discriminative features to lead to the best recognition rate. Mel-frequency cepstral coefficients (MFCC) have been used as set of descriptors for the speaker data. Principle component analysis technique (PCA) is applied to this set of descriptors to achieve dimensionality reduction. The output set is fed to a back-propagation neural net (BPNN) for speaker identification process. The recognition rate achieved is more than 87%. Also, it is found that the effect of noise is reduced when implementing selection process in noisy condition.

Keywords: MFCC, PCA, BPNN, Dimensionality Reduction, Feature Extraction.

1. Introduction

Spoken language is the most natural way used by human to communicate information. Speech signal conveys linguistic information as well as speaker information (e.g. Emotional, regional and physiological characteristics) [1].

Speaker Recognition is a multi-disciplinary technology which uses the vocal characteristics of speakers to deduce information about their identities. As for the importance of speaker recognition; it is noteworthy that speaker identity is the only biometric which may be easily tested (identified or verified) remotely through the existing infrastructure, namely the telephone network. This makes speaker recognition quite valuable and unrivalled in many real-world applications. It needs not be mentioned that with the growing number of cellular (mobile) telephones and their ever-growing complexity, speaker recognition will become more popular in the future[2].

In this paper a speaker recognition system is proposed. It is speaker independent system based on neural network. The tools used for features extraction and selection are MFCC and PCA. The neural net training algorithm used is back-propagation. The rest of paper has been organized as follows: Section 2 talks about speaker recognition system stages and the techniques have been used in our work i.e. MFCC technique, PCA algorithm and BPNN. Section 3 explains in details the proposed speaker recognition system along with the block diagram. Finally, the obtained results and graphs are presented in section 4, followed by conclusion in section 5.

2. Speaker Recognition

Speaker recognition, sometimes referred to as speaker biometrics, includes identification, verification (authentication), and by extension, segmentation, tracking and detection of speakers. It is a generic term used for any procedure which involves knowledge of the identity of a person based on his/her voice [3].

A speaker recognition system first tries to model the vocal tract characteristics of a person. This may be a mathematical model of the physiological system producing the human speech or simply a statistical model with similar output characteristics as the human vocal tract. Once a model is established and has been associated with an individual, new instances of speech may be assessed to determine the likelihood of them having been generated by the model of interest in contrast with other observed models. This is the underlying methodology for all speaker recognition applications [2].

Speaker recognition system can be classified depending on task objective into speaker identification and speaker verification, where speaker identification is the process of determining one of the registered speakers from whom the given utterance comes, and Speaker Verification is to confirm the claim of
identity and declaring the person to be true or impostor. And depending on mode of operation Speaker recognition system can be classified into text dependent model which follows the technique of detecting speakers based on the text and text independent model [4]. In text independent systems a speaker is recognized independent of the speech or language [5].

2.1. Pre-emphasis and Windowing

To flatten the speech spectrum, pre-emphasizer is used to compensate the high frequency component which was suppressed during the human sound production mechanism [6]. A simple digital filter used for such compensation is given as [7]:

\[ Y(n) = x(n) - \alpha x(n-1) , \quad 0 \leq \alpha \leq 1 \]  \hspace{1cm} (1)

Where \( y(n) \) and \( x(n) \) are the output and the input of the filter respectively.

Window technique at each frame is use to reduce signal discontinuity at either end of blocking. Mostly used technique is hamming window technique [3]. The Hamming window of length \( N \) is given as [6]:

\[ W(n) = 0.54 - 0.46 \cos \frac{2\pi n}{N-1}, \quad 0 \leq n \leq N-1 \]  \hspace{1cm} (2)

2.2. Feature Extraction

Feature extraction is the process of transforming the speech signal to a set of feature vectors. The aim of this transformation is to obtain a new representation which is more compact, less redundant, and more suitable for statistical modelling and the calculation of a distance or any other kind of score[8].

MFCCs is based on the human peripheral auditory system. The human perception of the frequency contents of sounds for speech signals does not follow a linear scale. Thus for each tone with an actual frequency \( t \) measured in Hz, a subjective pitch is measured on a scale called the ‘Mel Scale’. The mel frequency scale is a linear frequency spacing below 1000 Hz and logarithmic spacing above 1kHz [9]. The power spectrum is warped according to the Mel-scale in order to adapt the frequency resolution to the properties of the human ear. Then the spectrum is segmented into a number of critical bands by means of a filter bank. The filter bank typically consists of overlapping triangular filters [10]. The steps followed are to find the Mel Spectral coefficients [6, 11, 12]:

First, compute the Fourier transform of each window frame of a signal.

\[ S[k] = \sum_{n=0}^{N-1} s[n] e^{-j2\pi nk/N} , \quad 0 \leq k \leq N - 1 \]  \hspace{1cm} (3)

Where \( s[n] \) is the input of frame speech signal with \( n \) samples, \( S[k] \) is the output of FT to each speech frame with \( k \) samples.

Second, map the powers of the spectrum each frame onto the Mel scale, using triangular overlapping windows. Therefore we can use the following approximate formula to compute the \( \text{Mel} \) for a given frequency \( f \) in Hz.

\[ \text{Mel}(f) = 2595 * \log \left( \frac{1 + f/700}{2} \right) \]  \hspace{1cm} (4)

Third, calculate the logs of the powers at each of the Mel frequencies.

\[ E[m] = \log \left( \sum_{k=0}^{N-1} |X[k]|^2 H_m[k] \right), 0 \leq m \leq M \]  \hspace{1cm} (5)

Where \( x[k] \) is the input of mel filter bank, \( E[m] \) is the output of filter bank, \( M \) is number of mel filter bank channel, \( H_m[k] \) is the transfer function of the filter \( m \).

Fourth, compute the discrete cosine transform (DCT) of the list of Mel log powers.

\[ C[n] = \frac{1}{2M} \sum_{m=1}^{M} F[m] \cos \left( \frac{\pi (m-1)n}{2M} \right), \quad 0 \leq n \leq M \]  \hspace{1cm} (6)

Lastly, the MFCCs are the amplitudes of the resulting spectrum.
2.3. Features Selection

The central assumption when using a feature selection technique is that the data contains many redundant or irrelevant features. Redundant features are those which provide no more information than the currently selected features, and irrelevant features provide no useful information in any context. Feature selection techniques are a subset of the more general field of feature extraction [13]. In this work, Principle Component Analysis Algorithm (PCA) is used for features selection process. The steps of computing PCA algorithm are as follows [14]:

Suppose $x_1, x_2, \ldots, x_M$ are $M \times 1$ vectors

Step 1:

$$\bar{x} = \frac{1}{M} \sum_{i=1}^{M} x_i$$  \hspace{1cm} (7)

Step 2: Subtract the mean

$$\phi_i = x_i - \bar{x}$$  \hspace{1cm} (8)

Step 3: From the matrix $[\phi_1, \phi_2, \ldots, \phi_M]$ ($N \times M$ matrix), then compute:

$$C = \frac{1}{M} \sum_{N=1}^{M} \phi_N \phi_N^T = AA^T$$  \hspace{1cm} (9)

Step 4: Compute the eigenvalues of $C$: $\lambda_1 > \lambda_2 > \ldots > \lambda_N$

Step 5: Compute the eigenvectors of $C$: $u_1, u_2, \ldots, u_N$

Step 6: (dimensionality reduction step) keep only the terms corresponding to the $K$ largest eigenvalues

$$\frac{\sum_{i=1}^{K} \lambda_i}{\sum_{i=1}^{N} \lambda_i} > 0.9$$  \hspace{1cm} (10)

2.4. Back-propagation Neural Network (BPNN)

BP learning is the most popular learning rule for performing supervised learning tasks. It uses a gradient search technique to minimize a cost function equivalent to the Mean square Error (MSE) between the desired and actual network outputs [15].

Basically, the error back-propagation process consists of two passes through the different layers of the network: a forward pass and a backward pass. In the forward pass, input vector is applied to the network, and its effect propagates through the network, layer by layer. Finally, a set of outputs is produced as the actual response of the network. This implements as follows:

Input vector is

$$X_p = (x_{p1}, x_{p2}, \ldots, x_{pN})^t$$  \hspace{1cm} (11)

Where $X_p$ is the input vector.

Calculate the net input values to the hidden layer units:

$$net_{p_j}^h = \sum_{i=1}^{N} W_{Oi}^h x_{pi} + \Theta_{p_j}^h$$  \hspace{1cm} (12)

Where $net_{p_j}^h$ is the net input to hidden layer, $W_{Oi}^h$ is the weight on the connection from $i^{th}$ input unit, $\Theta_{p_j}^h$ is the bias term and “h” refers to quantities on the hidden layer.

Calculate the outputs from the hidden layer:

$$i_{p_j} = f_j^h(net_{p_j}^h)$$  \hspace{1cm} (13)

Where $i_{p_j}$ is the output from the hidden layer and $f_j^h$ is the activation function.
Move to the output layer. Calculate the net-input values to each unit:

$$\text{net}^o_{pk} = \sum_{j=1}^{l} w^o_{kj} i_{pj} + \theta^o_k$$  \hspace{1cm} (14)

Where $\text{net}^o_{pk}$ is the net input to the output layer, $w^o_{kj}$ is the weight in the connection from $j^{th}$ hidden unit, $\theta^o_k$ is the bias term and "o" refers to quantities in the output layer.

Calculate the outputs:

$$O_{pk} = f^o_k(\text{net}^o_{pk})$$  \hspace{1cm} (15)

Where $O_{pk}$ is the output got from the output layer.

During the backward pass, on the other hand, the synaptic weights are all adjusted in accordance with the error-correction rule. The actual response of the network is subtracted from a desired target response to produce an error signal. This error signal is then propagated backward through the network. The synaptic weights are adjusted so as to make the actual response of the network closer the desired response [16, 17]. This pass is implements as follows:

Calculate the error terms for the output units:

$$\delta^o_{pk} = (y_{pk} - O_{pk}) f'_j(\text{net}^o_{pk})$$  \hspace{1cm} (16)

Where $\delta^o_{pk}$ is the error at each output unit, $y_{pk}$ is the desired error and $O_{pk}$ is the actual error.

Calculate the error terms for hidden units:

$$\delta^h_{pj} = f'_j(\text{net}^h_{pj}) \sum_k \delta^o_{pk} w^o_{kj}$$  \hspace{1cm} (17)

Where $\delta^h_{pj}$ is the error at each hidden unit.

Update weights on the output layer:

$$w^o_{kj}(t + 1) = w^o_{kj}(t) + \eta \delta^o_{pk} i_{pj}$$  \hspace{1cm} (18)

Update weights on the hidden layer:

$$w^h_{ki}(t + 1) = w^h_{ki}(t) + \eta \delta^h_{pj} x_i$$  \hspace{1cm} (19)

Where $\eta$ is the learning rate parameter.

2.4.1. Activation function

Every neuron model consists of a processing element with synaptic input connections and a single output. The signal flow of neuron inputs, $x_i$, is considered to be unidirectional. The neuron output signal is given by the relationship $o = f(\Sigma)$, which is illustrated in Figure1.

![Neuron model](Image)
One of the most used activation functions is the sigmoid function which will only produce positive numbers between 0 and 1. The sigmoid activation function is most useful for training data that is also between 0 and 1 [17].

\[
y = \frac{1}{1 + \exp(-2sx)} , 0 < y < 1 , \quad d = 2sy(1-y)
\]

(20)
x is the input to the activation function, y is the output, s is the steepness and d is the derivation.

3. Proposed Speaker Recognition System

Like any other pattern recognition systems, the proposed speaker recognition (identification) system involves two phases namely, training and testing. Figure 2 shows the layout of system. Each phase has its own stages. As shown in the figure, the stages pre-processing and feature extraction are in common. The stages can be summarized as follows:

3.1. Pre-processing

This stage includes the following steps:

- Pre-emphasis
- Framing
- Silence Removal
- Windowing

To pre-emphasis speech signal, a high pass filter which is given in the equation (1) is implemented in this process. The value of \( \alpha \) in this equation is setting to 0.97. To achieve stationary, speech signal is divided into segments (frames) with fixed duration, each with \( N \) samples. This fixed duration value which equivalent to frame length is tested to find its suitable value which leads to the best recognition rate; experimentally 36ms is adopted in this system. Silence is removed by testing the energy value of each frame respect to certain threshold value to determine which frame is not silent. Hamming windowing process is applied on overlapped frames with ratio of 50% of frame size.

![Figure 2. Layout of proposed system](image)
3.2. Features extraction

It is achieved by MFCC technique and implemented as depicted in figure 3. It includes the following steps:

- Convert each frame into frequency domain by FFT.
- Map the power of spectrum of each frame into Mel-scale.
- Take logarithmic to these energies.
- DCT is taken to the logarithmic of these energies.

The features vector that produced from MFCC is with high dimensionality which equal to 216 features.

3.3. Features selection

It is implemented by using PCA; this is done to pre-treatment for the speech feature parameters. The multidimensional feature parameters need to be reduced to a smaller vector and need to maintain the features which are more distinctive between speakers. To find an optimal set of features obtained from MFCC module, a PCA algorithm is applied to implement selection process. In this work the number of best features that constituted the optimal set is tested to find the suitable number that leads to best recognition rate. The first selected number is yielded by implementing this algorithm is 22 features. This number of selected feature will be the start point for neural network input.

3.4. Normalization

The last vector has been normalized because neural network training could be made more efficient by performing certain pre-processing steps on the network inputs and targets. Network input processing functions transforms inputs into better form for the network use. The normalization process for the raw inputs has great effect on preparing the data to be suitable for the training. Without this normalization, training the neural networks would be very slow; in this work Min-Max method has been used.

Min-Max Normalization: This method rescales the features or outputs from one range of values to a new range of values. More often, the features are rescaled to lie within a range of 0 to 1 or from -1 to 1.

\[
\tilde{f} = \left( f - f_{\text{min}} \right) / \left( f_{\text{max}} - f_{\text{min}} \right)
\]

Where:
- \( \tilde{f} \): New value of feature, \( f \): Old value of feature,
- \( f_{\text{max}} \): Maximum value that new feature lie in it,
- \( f_{\text{min}} \): Minimum value that new feature lie in it

3.5. Back-propagation Neural Network (BPNN)

Three layers feed forward neural network architecture had been adopted, input layer, hidden layer, output layer. The determination to the number of input nodes in the input layer are corresponds to the number of features in the features vector that are extracted from MFCC. Feeding these features vectors with their high dimensionality directly to the neural network will increase the computations, training time and performance degradation. By applying selection process the features vector dimensionality
has been reduced. In this work the number of best selected features that constitutes the optimal set that leads to the high accuracy rate is tested. Depending on these tests, six different structures of back-propagation neural net are adopted. Each structure has specific number of input nodes (selected features), number of hidden nodes, and output nodes correspond to the number of speakers. The numbers of input nodes selected are 22, 24, 26, 28, 30, and 32 for structure 1 to 6 respectively. Figure 4 represents the general form of the six structures which N is equal to selected number of features and M represents number of hidden nodes.

![GeneralBPNN structure](image)

Figure 4. GeneralBPNN structure

The main goal of network training is to adjust the weights of neural network nodes such that the network should be able to classify different speakers. The trained network (whose weights are adjusted) is applied on the test speech patterns set to classify their speakers.

4. System performance

Sampling frequency of 16 KHz, sampling resolution of 16-bits, mono recording channel and recorded file format is wav have been considered to capture the speech utterances. A set of tests have been conducted on (10) speakers, (7 male and 3 female).

4.1. Selection Process Effect

Reducing the dimension of inputs vector which correspond to the number of nodes in input layer plays an important role in reducing complexity and the amount of computation in neural network structure. As a result the network training time is shortened.

4.1.1. Time elapsed

Figure 5 illustrates the time elapsed (in second) in training the network without selection process and comparison it with time elapsed in training the network with selection process.
4.1.2. Accuracy rate

Table 1 presents the identification recognition rate based on the number of selected features. The result refers the set of (30) features has high identification rate. This set is adopted in our study. Figure 6 presents comparison of identification rate respect to number of selected features.

<table>
<thead>
<tr>
<th>Structure Number</th>
<th>Number of Selected Features</th>
<th>Recognition Rate %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>22</td>
<td>78%</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>79%</td>
</tr>
<tr>
<td>3</td>
<td>26</td>
<td>82%</td>
</tr>
<tr>
<td>4</td>
<td>28</td>
<td>84%</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>87%</td>
</tr>
<tr>
<td>6</td>
<td>32</td>
<td>84%</td>
</tr>
</tbody>
</table>
4.2. Noisy Condition

Figure 7 illustrates the effect of selection process in presence of noisy condition; the training and tested BPNN with noisy speech signals is implemented by adding White Gaussian Noise with 30db to the same data that used in clean environment which explained in previous subsection. From the experiment, the number of features that lead to high accuracy rate in noisy condition is more than the number of features that lead to high accuracy rate in clean condition. It's clear from the figure that the optimal set that leads to high accuracy in noisy condition by selection process is 32 features.

5. Conclusion

In this study, PCA is used as a tool for selection the more important features and ignoring the redundancy and irrelevant features resulted from MFCC. The small number of features resulted from PCA compared to the large number of features resulted from MFCC encourage us to use them as input...
to neural net. The study reveals that there is an exact number of features that leads to the high recognition rate. The features vector reduction has leads to reduce the time consuming in classification and recognition processing. Also working with exact features leads to reduce sensitivity against noisy compared to MFCC.

6. References