Automatic Speech Emotion and Speaker Recognition based on Hybrid GMM and FFBNN

J. Sirisha Devi\textsuperscript{1}, Y. Srinivas\textsuperscript{2} and Siva Prasad Nandyala\textsuperscript{3}

\textsuperscript{1}Assitant Professor, Dept of IT, GRIET, Hyderabad, 501401, Andhra Pradesh, India
\textsuperscript{2}Professor, Dept of IT, GITAM University, Visakhapatnam, 530045, Andhra Pradesh, India
\textsuperscript{3}Research Scholar, Dept of ECE, NIT Warangal, Warangal, 506004, Andhra Pradesh, India

ABSTRACT

In this paper we present text dependent speaker recognition with an enhancement of detecting the emotion of the speaker prior using the hybrid FFBN and GMM methods. The emotional state of the speaker influences recognition system. Mel-frequency Cepstral Coefficient (MFCC) feature set is used for experimentation. To recognize the emotional state of a speaker Gaussian Mixture Model (GMM) is used in training phase and in testing phase Feed Forward Back Propagation Neural Network (FFBNN). Speech database consisting of 25 speakers recorded in five different emotional states: happy, angry, sad, surprise and neutral is used for experimentation. The results reveal that the emotional state of the speaker shows a significant impact on the accuracy of speaker recognition.

KEYWORDS

Mel-frequency Cepstral Coefficient (MFCC), Feed Forward Back Propagation Neural Network (FFBNN), Gaussian Mixture Model (GMM).

1. INTRODUCTION

Speaker recognition refers to recognize the person from their speech. Accuracy of recognition increases if the system is text-independent [1]. The speech signal contains the message being spoken, the emotional state of the speaker and the information of the speaker. Therefore we can use the speech signal for both speech and speaker recognition [2]. Speaker recognition extracts the underlying linguistic message in an utterance. It is the process of automatically recognizing who is speaking on the basis of features present in the speech signal. Speaker recognition is helpful in the areas such as voice dialling, banking by telephone, telephone shopping, database access services, information services, and security control for confidential information areas, voice mail and remote access to computers [9]. Influence of emotional state of human speech in speaker recognition is very high [13]. The term “emotion” can refer to an extremely complex state associated with a wide variety of mental, physiological and physical events. Emotional speech database is valuable for this speaker recognition. In this paper a combination of speech features is considered. The spectral feature Mel frequency cepstral coefficients best suited for speaker recognition and the prosodic feature pitch, which is strongly dependent on the emotional state of the speaker.

In section 2, feature extraction of speech signal is performed. In section 3, classification of the feature vectors is explained. In section 4, experiment shows how training and testing are performed and result analysis is done.

DOI:10.5121/ijcsa.2014.4104
2. **Feature Extraction**

2.1 Pre-processing

Speaker specific features will be present in the voiced part of the speech signals [9]. Pre-processing is a process of separating the voiced and unvoiced speech signals.

2.1.1 Noise removal

While recording the speech signal, it consists of various unwanted piece of signals which are considered to be noise/unvoiced part of speech. This noise part may be due to low quality of source, loss of speech segments, channel fading, and presence of noise in the channel or echo or reverberation.

A low-pass filter allows signal frequencies below the low cut-off frequency to pass and stops frequencies above the cut-off frequency. Speaker specific information is available in the higher frequencies (above 4 kHz), so a low pass filter is used to remove all the frequencies less than 4 KHz as unvoiced data.

2.1.2 Framing

Each speech signal is split into frame segments of size 30ms (milli second) where the sampling rate is 22.05 KHz. Each segment is extracted at every 50ms interval. This implies that the overlap between segments is 10ms.

2.1.3 Windowing

After framing, each and every frame will contain discontinuities at the beginning and end of each frame. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. If we define the window as Eq.1

\[ w(n), 0 \leq n \leq N - 1 \]  

Where, N is the number of samples in each frame, then the result of windowing is the signal, Eq.2

\[ y_1(n) = x_1(n)w(n), 0 \leq n \leq N - 1 \]  

Typically the Hamming window is used, which has the form (Eq.3):

\[ w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{N-1} \right) \quad 0 \leq n \leq N-1 \]

2.1.4 Fast Fourier Transform (FFT)

The next processing step is the Fast Fourier Transform, which converts each frame of N samples from the time domain into the frequency domain. FFT gets log magnitude spectrum to determine MFCC. We have used 1024 point to get better frequency resolution.

2.2 Mel- Frequency Cepstral Coefficients (MFCC)

Prosody is the suprasegmental phonology of speech sounds, involving (but not limited to) aspects such as pitch, loudness, tempo, and stress [10]. Pitch, often referred to as fundamental frequency,
is the vibration rate of the vocal folds. Gold-Rabiner algorithm [3] illustrates pitch extraction based on the fact that locating the position of the maximum point of excitation is not always determinable from the time-waveform. Pitch is elaborated as the time difference between two major peaks in the voiced signal. The method of pitch extraction from the waveforms is described in the following steps:

**Step 1:** Speech signal is given as input to the system.

**Step 2:** Normalized Cross Correlation Function is computed periodically for all lags of \( f_0 \) and the locations of the all local maxima obtained in the first iteration are stored.

**Step 3:** Normalized cross correlation is performed for improved high sampled rate signals in order to determine the accurate location of peaks, with estimations applied to amplitude.

**Step 4:** \( f_0 \) for the particular frame is obtained by the normalized cross correlation related to the particular frame.

**Step 5:** Dynamic programming is used in order to select the set of cross correlation peaks or unvoiced hypothesis across all frames.

Spectral features characterize signal properties in the frequency domain, thus providing useful additions to prosodic features. The frequency domain representation can be obtained by taking the discrete Fourier transform (DFT) of the discrete-time signal. However, the DFT of a signal usually results in a high-dimensional representation, in addition, with fine spectral details that may turn out to impair recognition performance. Therefore, the DFT spectrum is usually transformed into some more compact feature representations for speech related recognition tasks, such as the mel-frequency cepstral coefficients (MFCCs).

The 20 Mel triangular filters are designed with 50% overlapping. From each filter the spectrum is added to get one coefficient each, in this way we have considered the first 13 coefficients as our features. These frequencies are converted to Mel scale using following conversion formula (Eq.4).

\[
f(mel) = 2595 \cdot \log_{10} (1 + \frac{f}{700})
\]

(4)

We have considered 13 MFCC coefficients because; of the fact it gives better recognition accuracy than other coefficients.

### 3. Classification Techniques

In both the training and testing phases, before speaker recognition the emotional state of speaker is identified [11]. A new hybrid model is proposed in which GMM in training and FFBN in testing is used for emotion recognition.

#### 3.1 Emotion Recognition

#### 3.1.1 Gaussian Mixture Model

The feature vector thus obtained is send to GMM classifier for emotion recognition in the training phase. In the GMM based classifier [5] [7], probability density function of the feature space is modeled with a diagonal covariance GMM for each emotion [12]. Probability density function is a weighted combination of K component densities given by Eq.5.

\[
p(f) = \sum_{k=1}^{K} w_k p(f | k)
\]

(5)

where \( f \) : observation feature vector

\( k \) : mixture weight associated with the k-th Gaussian component.
The weights satisfy the constraints, $0 \leq w_k \leq 1$ and $\sum_{k=1}^{K} w_k = 1$

The conditional probability $p(f | k)$ is modeled by Gaussian distribution with the component mean vector $\mu_k$, and the diagonal covariance matrix $\Sigma_k$. The GMM for a given emotion is extracted through the expectation-maximization based iterative training process using a set of training feature vectors representing the emotion.

In the emotion recognition phase [8], posterior probability of the features of a given speech utterance is maximized over all emotion GMM densities. Given a sequence of feature vectors for a speech utterance, $F = \{f_1, f_2, \ldots, f_T\}$, let’s define the loglikelihood of this utterance for emotion class $e$ with a GMM density model $\gamma_e$ as given in Eq. (6),

$$\log p(F | \gamma_e) = \sum_{t=1}^{T} \log p(f_t | \gamma_e)$$

Where, $p(F|\gamma_e)$ is the GMM probability density for the emotion class $e$ as defined in eq. 5.

Then, the emotion GMM density that maximizes posterior probability of the utterance is set as the recognized emotion class is given by Eq. (7).

$$\varepsilon = \arg \max_{\varepsilon} \log p(F | \gamma_{\varepsilon})$$

Where, $E$ is the set of emotions and $\varepsilon$ is the recognized emotion.

3.1.2 Feed Forward Backpropogation Neural Network (FFBNN)

In testing phase, emotion recognition from the speech signals is done by FFBNN. The features are extracted for the speech signal and given to the FFBNN to perform the testing process. The speech signal’s corresponding MFCC features are taken as input to the FFBNN [4]. Here, we have taken three inputs nodes as energy entropy ($E_e$), short-time energy ($S_e$) and Zero crossing Rate ($Z_{CR}$). $N_d$ number of hidden layers and one output layer, which is a prosody effect of the given input signal. The proposed emotion classification FFBNN structure is shown in Figure 1.

Initially, the input feature values are transmitted to the hidden layer and then, to the output layer. Each node in the hidden layer gets input from the input layer, which are multiplexed with suitable weights and summed. The hidden layer input value calculation function is called as bias function, which is described below by Eq. (8),

$$H_i = \beta + \sum_{i=1}^{3} (w_{iE} E_{ei} + w_{iS} S_{ei} + w_{iZ} Z_{CRi} )$$

In Eq. (8), $E_{ei}, S_{ei}$ and $Z_{CRi}$ are the feature values of the $i^{th}$ person speech signal. The activation function in the output layer is given in Eq. (9). The output values from the output layer are compared with target values and the learning error rate for the neural network is computed, which is given in Eq. (10).

$$\alpha = \frac{1}{1 + e^{-H_i}}$$

$$\lambda = \frac{1}{N_d} \sum_{h=0}^{N_d-1} D^i_h - O^i_h$$
In eqn. (10),

\[ \lambda \] - is the learning error rate

\[ D_h^i \] - is the desired output

\[ O_h^i \] - is the actual output.

The error between the nodes is transmitted back to the hidden layer and this process is called the backward pass of the back propagation algorithm. The reduction of error by back propagation algorithm is described in the subsequent steps.

(i) Initially, the weights are assigned to hidden layer neurons. The input layer has a constant weight, whereas the weights for output layer neurons are chosen arbitrarily. Subsequently, the bias function and output layer activation function are computed by using the Eq. (8) and (9).

(ii) Next, the back propagation error is computed for each node and the weights are updated by using the Eq. (11).

\[
 w_{ih} = w_{ih} + \Delta w_{ih} 
\]

Where, the weight \( \Delta w_{ih} \) is changed, which is given by Eq.(12),

\[
 \Delta w_{ih} = \eta, N_{ih}, E^{(\phi)} 
\]

Where, \( \eta \) is the learning rate that normally ranges from 0.2 to 0.5, and \( E^{(\phi)} \) is the BP error. The bias function, activation function, and BP error calculation process are continued till the BP error gets reduced i.e. \( E^{(\phi)} < 0.1 \). If the BP error reaches a minimum value, then the FFBNN is well trained by the speech signals feature values for performing the emotion classification. The well trained FFBNN provides an accurate classification results for the input emotion speech signals. The well trained FFBNN classify the input speech signals based on the prosody which the speech signal belongs to.

3.1.3 Hybrid GMM/FFBN Approach

Therefore, we followed a different approach, namely the extension of a state-of-the-art system that achieves extremely good recognition rates with a GMM that is trained. For recognition purpose used the FFBNN method.

3.2 Speaker Recognition using HMM

Hidden Markov Model is used to recognize the speaker after emotion is recognized by hybrid GMM/FFBN. HMM is a statistical tool for modeling generative sequence to generate observable sequence when characterized by the primary process. The observation is turned to be a probabilistic function (discrete or continuous) of a state instead of a one-to-one correspondence of a state. Each state randomly generates one of M observations (or visible states). To define hidden Markov model, the following probabilities have to be specified:

Matrix of transition probabilities, \( A=(a_{ij}) \), \( a_{ij} = P(s_i | s_j) \)
Matrix of observation probabilities, $B = (b_i(v_m))$, $b_i(v_m) = P(v_m | s_i)$

Vector of initial probabilities, $\pi = (\pi_i)$, $\pi_i = P(s_i)$

Model is represented by $M = (A, B, \pi)$

4. EXPERIMENTS AND RESULTS

MATLAB R2013a has been used for experimentation.

4.1 Training phase

In training phase, the extracted MFCC vector is sent to both the classification techniques. First emotion recognition is done using GMM classifier then the same feature vector is passed to HMM classifier for speaker recognition (SR). For each and every emotion a unique record is generated. Figure 2 explains the block diagram of the entire procedure.

4.2 Testing phase

After feature extraction the feature vector is passed to FFBN classifier for emotion recognition. Once the emotion recognition is recognized, based on the emotional state the corresponding record will be supplied for further recognition part which is speaker recognition using HMM.

4.3 Results

When the emotional state of speaker differs in the testing phase the recognition rate decreased significantly. The results shows that the accuracy rate of speaker recognition has been considerably increased when compared to the recognition rate where emotional state of the speaker was not considered. The accuracy rate of speaker indirectly depends on the accuracy of emotion recognition. As well as when we compared the accuracy rate of emotion recognition using GMM with the proposed system, the accuracy rates have been considerably increased. The recognition rate obtained is 93% with the hybrid model compared to 91% using the same method for training and testing for emotion recognition. Table 1 shows the comparison of speaker recognition results.

The Figure 3 gives the simulation results of speaker recognition for the given emotions.

<table>
<thead>
<tr>
<th>Emotional State</th>
<th>Happy</th>
<th>Angry</th>
<th>Sad</th>
<th>Surprise</th>
<th>Neutral</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recognition Rate (%)</td>
<td><strong>Emotion (GMM)</strong></td>
<td>89</td>
<td>94</td>
<td>89</td>
<td>86</td>
</tr>
<tr>
<td>Speaker (HMM)</td>
<td>88</td>
<td>97</td>
<td>92</td>
<td>90</td>
<td>96</td>
</tr>
<tr>
<td>Recognition Rate (%) of Hybrid Model</td>
<td><strong>Emotion (Hybrid GMM/FFBN)</strong></td>
<td>90</td>
<td>95</td>
<td>89</td>
<td>85</td>
</tr>
<tr>
<td>Speaker (HMM)</td>
<td>90</td>
<td>97</td>
<td>92</td>
<td>92</td>
<td>97</td>
</tr>
</tbody>
</table>
Figure 1: Structure of FFBNN in Emotion Classification

![Structure of FFBNN in Emotion Classification](image)

Figure 2: Process Flow

1. Pre-Processing of Speech Signal
2. Spectrogram Analysis
3. Normalization
4. Emotion Recognition using GMM/FFBNN
5. Recognition using HMM
6. Output Generation

![Process Flow Diagram](image)

Figure 3: Simulation Results

![Simulation Results](image)
REFERENCES