Emotion recognition using neural networks

Mehmet Unluturk
Izmir University of Economics
29 PUBLICATIONS **126** CITATIONS
SEE PROFILE

Coskun Atay
Izmir University of Economics
15 PUBLICATIONS **30** CITATIONS
SEE PROFILE

Emotion Recognition Using Neural Networks

MEHMET S. UNLUTURK, KAYA OGUZ, COSKUN ATAY
Department of Software Engineering
Izmir University of Economics
Sakarya Cad No.156, Balcova, Izmir 35330
TURKEY
suleyman.unluturk@ieu.edu.tr kaya.oguz@ieu.edu.tr coskun.atay@ieu.edu.tr

Abstract: - Speech and emotion recognition improve the quality of human computer interaction and allow more easy to use interfaces for every level of user in software applications. In this study, we have developed the emotion recognition neural network (ERNN) to classify the voice signals for emotion recognition. The ERNN has 128 input nodes, 20 hidden neurons, and three summing output nodes. A set of 97932 training sets is used to train the ERNN. A new set of 24483 testing sets is utilized to test the ERNN performance. The samples tested for voice recognition are acquired from the movies “Anger Management” and “Pick of Destiny”. ERNN achieves an average recognition performance of 100%. This high level of recognition suggests that the ERNN is a promising method for emotion recognition in computer applications.

Key-Words: - Back propagation learning algorithm, Neural network, Emotion, Speech, Power Spectrum, Fast-Fourier Transform (FFT)

1 Introduction

Speech is one of the oldest tools humans use for interaction among each other. It is therefore one of the most natural ways to interact with the computers as well. Although speech recognition is now good enough to allow speech to text engines, emotion recognition can increase the over all efficiency of interaction and may provide everyone a more comfortable user interface.

It is often trivial for humans to get the emotion of the speaker and adjust their behavior accordingly. Emotion recognition will give the programmer a chance to develop an artificial intelligence that can meet the speaker's feelings that can be used in many scenarios from computer games to virtual sales-programs.

Three base emotions, angry, happy and neutral are taken into account. Various speech sets that belong to these emotion groups are extracted from different movies and used for training and testing. The ERNN is capable of distinguishing these test samples.

Neural networks are chosen for the solution because a basic formula cannot be devised for the problem. The neural networks are also quick to respond which is a requirement as the emotion should be determined almost instantly. The training takes a long time but is irrelevant as the training is mostly done off-line.

The paper is organized as follows; part 2 is about ERNN design, part 3 tells about the results and discussion, part 4 includes the conclusion and future work.

2 ERNN Design

Emotion recognition is not a new topic and both research and applications exist using various methods most of which require extracting certain features from the speech [1]. A common problem is determining emotion from noisy speech, which complicates the extraction of the emotion because of the background noise [2]. To extract the emotion signatures inherent to voice signals, the back propagation-learning algorithm [3,4,5,6] is used to design the emotion recognition neural network (ERNN).

The block diagram of ERNN is shown in Figure 1. Segmented data is applied to the input of the power spectrum processor utilizing the Fast Fourier Transform algorithm. The output of it is normalized and is presented to a three layer, fully interconnected neural network for classification. The output layer of the neural network is inputted by the weighted sum of outputs of the hidden and bias nodes in the hidden layer. These weighted inputs are processed by a hyperbolic tangent function. A set of desired output values is then compared to the estimated outputs of the neural network for every set of input values of the power spectrum of the voice signals. The weights are appropriately updated by back propagating the gradient of the output error through the entire neural network.

The experimental data, used for both training and testing the ERNN, is obtained from movies in WAV format at 48000 Hz with only one channel (mono). Each experimental data segment is composed of 65536 data points.

Removing the mean and dividing it with the standard deviation normalize the segmented power spectrum data. This normalization is highly desirable because it desensitizes the neural network to the signal offset and/or signal gain. Normalization is given as

$$X_p(k\Omega) = \frac{R_p(k\Omega) - \mu}{\sigma}, k = 1, \cdots, K$$
$$\mu = \frac{1}{K}\sum_{k=1}^{K} R_p(k\Omega), \qquad (1)$$
$$\sigma = \sqrt{\frac{1}{K-1}\sum_{1}^{K}(R_p(k\Omega) - \mu)^2},$$

where $R_p(k\Omega)$ (Ω is the frequency sampling interval) is the segmented power spectrum of the voice signal, μ and σ are the estimated mean (i.e., signal offset) and the estimated standard deviation (i.e., measurement scale) respectively.

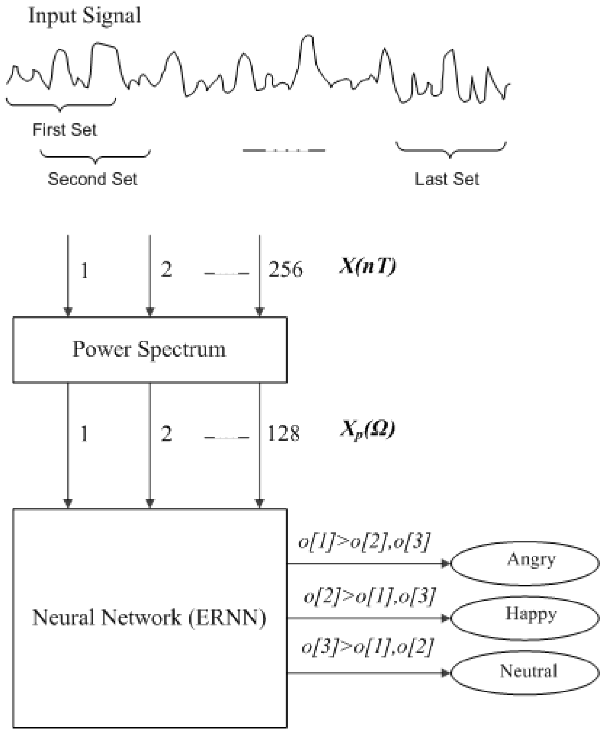

Figure 1. Block diagram of the emotion recognition neural network (ERNN) system. The output neuron that has the highest value indicates which type of emotion signal is present at the input.

The input signal to the power spectrum block (see Figure 1) is created using a sliding window. The size of the sliding window is 256 samples, and the step between two successive windows is 8 samples. The first set is taken from the beginning of the voice data. The second set is 8 samples to the right of the first set, and this is repeated until the window covers the entire 65536 samples of the voice signal. The power spectrum, $R_p(k\Omega)$, of each input set is calculated by

$$R_p(k\Omega) = FFT(x(nT)) \bullet Conj(FFT(x(nT))) \quad (2)$$

where $x(nT)$ is the sampled value of the voice signal, T is the time sampling interval, FFT is the Fast Fourier Transform, Ω is the frequency-sampling interval. The first 128 samples of $R_p(k\Omega)$ which span the entire frequency range is taken into consideration as an input to the neural network for signal classification. Hence, the neural network needs 128 inputs and 3 neurons in its output layer to classify the voice signals. The hidden layer has 20 neurons. This number was picked through experimentation and experience. During testing, ERNN accomplished a recognition performance of 100% with that many hidden neurons. And also, if the network has trouble classifying, then additional neurons can be added to the hidden layer. Figure 2 shows a three layer neural network that is designed to classify the voice signal power spectrum. The hidden neuron's output of this neural network, y_j, is given as

$$y_j = \phi(\sum_{i=1}^{N} X_{Pi} w_{ji}{}^{h} + \theta^{j}) \qquad (3)$$

where the activation function for the hidden and output layer, $\phi()$, is a tangent hyperbolic function defined as

$$\phi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \qquad (4)$$

The term X_p is the normalized input vector, $w_{ji}{}^{h}$ is the weight from j^{th} hidden neuron to the i^{th} input neuron.

$$X_p = [X_p(\Omega), \cdots, X_p(i\Omega), \cdots, X_p(N\Omega)] \quad (5)$$

Back propagation algorithm is used to estimate the hidden layer and output layer weights and biases for the optimal design of ERNN. Then, the output vector *o*, is applied to the decision block in order to classify the voice signal,

$$\begin{aligned} o[1] > o[2], o[3] &\longrightarrow Angry \\ o[2] > o[1], o[3] &\longrightarrow Happy \\ o[3] > o[1], o[2] &\longrightarrow Neutral \end{aligned} \qquad (6)$$

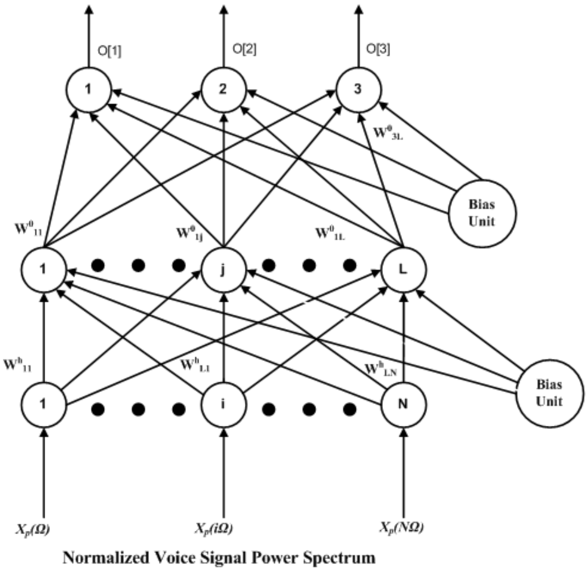

Figure 2. A three-layer model of emotion recognition neural network (N=128 and L=20).

Before the creation or training of the neural network, the samples and their outputs are prepared. As each file is read, they are positioned in the input matrix, as the correct outputs are positioned in the output matrix.

When the files are read through built-in MATLAB functions, they are put in arrays that have values between -1 and 1. This array is then scanned by a window, which is 256 bytes long, with a step of 8 bytes. Each file is scanned thoroughly, and they are read up to the 16th power of 2, 65536; as a result of this process, each file gives 8161 samples.

$$sampleCount = \frac{fileLength - windowSize}{step} \qquad (7)$$

The input matrix is generated in a way that the columns follow the angry, happy and neutral samples with the output matrix corresponding to the related values to help the back propagation algorithm to give better results.

Before placing the samples into the matrix, they are transformed from time domain to frequency domain using the Fast Fourier Transform. After the transformation, their power spectrum is generated. This power spectrum is symmetric so the first half is placed into the matrix. With four WAV files for each emotion that is scanned to give more than 8K samples, the input matrix is made up of almost 100K columns and 128 rows. The output matrix has 3 rows.

Once the matrices are ready, they are fed into the neural network for training. After the training is completed, the samples are simulated on the neural network. The samples are also taken through the same procedure. As there are almost 8K simulations in one test, the number of correct answers as percentage is taken as a final result (see Equation 6).

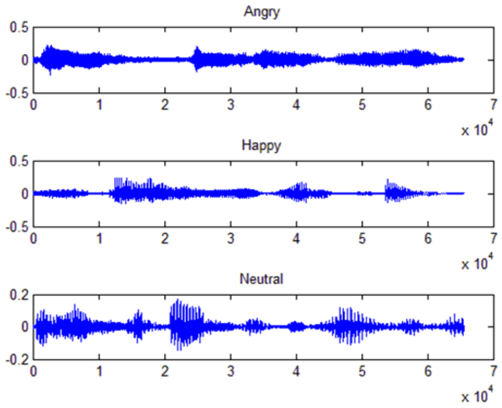

Figure 3. Samples from each emotion.

3 Results and Discussion

The voice data, applied for both training and testing the ERNN, is obtained from the movies “Anger Management” and “Pick of Destiny”. Figure 3 shows three types of emotion voice signals.

To reach the performance goal, a sliding window is used to segment the data in order to arrange as many sets of data as possible for training and testing the neural network. From voice signals, a set of 97932 training sequences, 128 samples each, was assembled to train the emotion recognition neural network. A new set of 24483 testing sequences was utilized to test the ERNN performance. Figure 4 shows a sample of 3 voice signals of angry, 3 voice signals of happy and 3 neutral voice signals. The corresponding power spectra of these voice signals are shown in Figure 5. Both the amplitude and their power spectra exhibit random pattern and a set of features that can be used for classification is not recognizable. Therefore, a trained neural network is conceivable to recognize the emotion buried into the voice signals.

ERNN achieves an average recognition performance of 100%. This performance is impressive and statistically reliable because 97932 data segments are used in training the neural network.

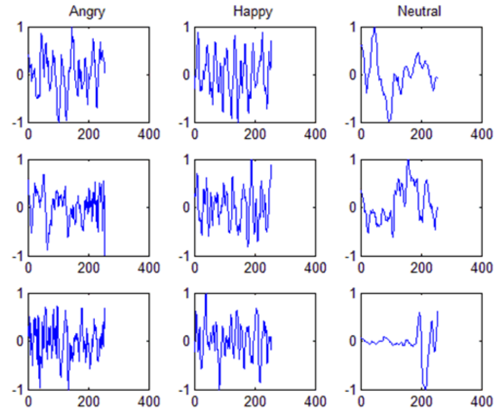

Figure 4. 256 time-samples X(nT) from each of angry, happy and neutral emotions respectively.

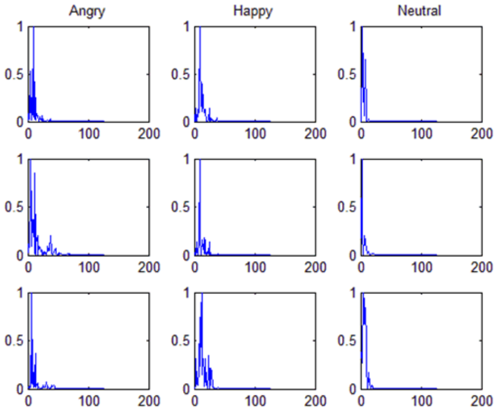

Figure 5. 128 power-samples $X_p(\Omega)$ from each of angry, happy and neutral power domain emotions respectively.

4 Conclusion

In this study we have developed a neural network that is designed to classify the power spectrum of the voice signals. Voice signals are random signals and are not readily quantifiable and lack uniquely recognizable features. Therefore, the neural network becomes appealing for classifying these signals because they are trainable.

The optimal values for neural networks weights are estimated using the back propagation algorithm. Experimental measurements of voice signals are utilized to train and test the emotion recognition neural network. This network shows a remarkable 100% classification performance.

These results are encouraging and suggest that neural networks are potentially useful for emotion recognition. Furthermore, the ERNN renders practical advantages such as real-time processing, adaptability and training capability. It is important to point out that similar neural network designs can be used in medical ultrasonic imaging for tissue characterization and diagnosis.

In the next step we are willing to work on getting this neural network working with near real time applications, like computer games. Role-playing games can utilize the player's acting capabilities as another input that might improve gaming experience.

References:

[1] Razak A., et al., “Comparison Between Fuzzy and NN Method for Speech Emotion Recognition”, Proceedings of the Third International Conference on Information Technology and Applications, 2005.

[2] Mingyu You, et al., “Emotion Recognition from Noisy Speech”, 2006 IEEE International Conference on Multimedia and Expo, Toronto, 2006.

[3] J. A. Freeman and D. M. Skapura, Neural Networks: Algorithms, Applications and Programming Techniques, Addison Wesley Publishing Company, 1991.

[4] T. Masters, Practical Neural Network Recipes in C++, Academic Press Publishing Company, 1993.

[5] A. Cichocki and R. Unbehauen, Neural Networks for Optimazing and Signal Processing, John Wiley & Sons Publishing Company, 1993.

[6] Werbos, P. The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting, John Wiley & Sons, New York, 1994.