EMOTION CONVERSION OF SPEECH SIGNAL USING NEURAL NETWORK

[1]AMRITA, [2]BAGESHREE PATHAK

[1,2]Department of Electronics and Telecommunication, Cummins College of Engineering for women
Email: virgo.aku2005@gmail.com, bvpathak100@yahoo.com

Abstract-We investigate a neural network approach to transform source emotion to target emotion. We achieve this emotion transformation by changing prosodic parameters (such as duration, pitch and intensity) learnt from the target emotion onto the neutral emotion. Mapping techniques are used for transformation of one emotion to another emotion. We then use an Artificial Neural Network (ANN) to learn the mapping between the neutral speech signal and the target speech signal and performed the emotion transformation.

Keywords- ANN, Praat software, Prosodic parameters.

I. INTRODUCTION

Human speech is an acoustic waveform generated by the vocal apparatus, whose parameters are modeled by the speaker to convey information. Emotion transformation in speech and its solution have been attracting much attention in the field of text-to-speech synthesis (TTS) for rapid generation of target expressive styles. A rudimentary emotion conversion frame work can be implemented using rules which modify an input utterance in a deterministic way. In GMM-based spectral conversion techniques were applied to emotion conversion but it was found that spectral transformation alone is not sufficient for conveying the required target emotion. In the use of GMM and CART-based F0 conversion methods were evaluated for mapping neutral prosody to emotional prosody in Mandarin speech. In a unified conversion system was proposed using duration embedded Bi-HMMs to convert neutral spectra and decision trees to transform syllable F0 segments. Report methods specific to an HMM-based speech synthesis framework, where emotional prosody and spectra were modeled and adapted jointly using phone HMMs. In an emotion conversion system for English which is independent of the underlying synthesis system is described.

In recent years, expressive speech synthesis has become an important subject due to requests to realize more familiar human interface of a spoken dialogue system, more advanced output of TTS (Text-to-Speech) system and so on. Emotional speech synthesis is of particular interest modifying the acoustic parameters of the input speech in accordance with a learned set of mapping from input emotion to target emotion.

We transform speech from one emotion to another by modifying the acoustic parameters of the input speech in accordance with a learned set of mappings from input emotion (tone) to target emotion (tone). Our method does not involve speech recognition or speech synthesis. Speech from one emotion to another by modifying the acoustic parameters of the input speech in accordance with a learned set of mapping from input emotion to target emotion. . After that we use ANN to get transformation of one emotion into another emotion.

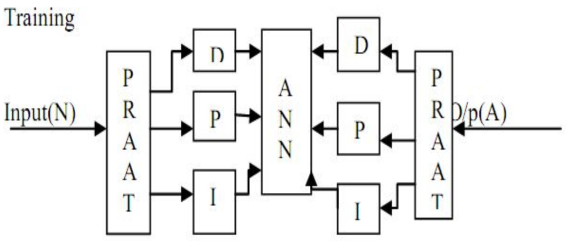
II. DATABASE CREATION

The database was created by recording data in four different emotions namely Neutral, Angry, Sad and Happy. Each sentence is recorded in each emotion twice. Different set of sentences will be used for the training and testing purposes. Voice data of five female speakers were recorded in noise free environment. They are recorded with sampling rate 44100samples/second and bit resolution is equal to 16 bits per sample.
The following sentences are used for the transformation purpose.

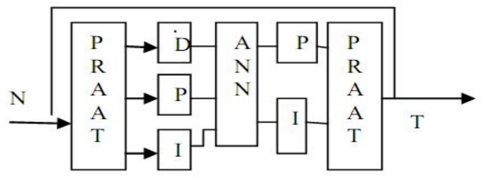
1. Come Here
2. Sit Down
3. Well Done
4. Great News
5. I am Bored

The recording has been done in DON studio karvenagar; Pune. The software used for editing is Protools the preamp used was HDOmni and UAD interface was used to get the microphone inputs. The editing was done on the MAC computer.

III. BLOCK DIAGRAM



Testing



Here N=neutral , T=target emotion.
P=Pitch of speech signal
D=Duration of speech signal
I=Intensity of speech signal

In training phase we give neutral sentence as an input to the PRAAT software to extract the feature (pitch=P, duration=D, intensity=I). These features are given to the ANN. By the same procedure we take target speech as an output. Extract the features and give it to ANN. ANN will map between source and target speech. We had taken 2N samples from one speaker speaking N sentences. Each of these 2N sentences will have a neutral counterpart (N sample) and the target tone/emotion counterpart (N sample), for example neutral and angry. The prosodic parameters from these N inputs and N outputs will be extracted using PRAAT. We train the ANN by applying these 2N samples (N sets) as training set. Once the ANN is trained, we apply some test sets (we give an input speech sample in neutral tone and the ANN should give out a speech sample in angry tone.

IV. PRAAT SOFTWARE

PRAAT (talk) is a free scientific computer software package for the analysis of speech in phonetics. It was designed by Paul Boersma and David Wee ink .It can run on a wide range of operating system. For extraction of features software used, is PRAAT. Here features that are going to be extracted are pitch, duration and intensity. We copy the durational prosodic parameter by marking the duration of the words in the target samples. After that mapping them to the corresponding points in the input sample. Fig1 shows the durations of the word marked in the input sample (left) and the target sample(right).

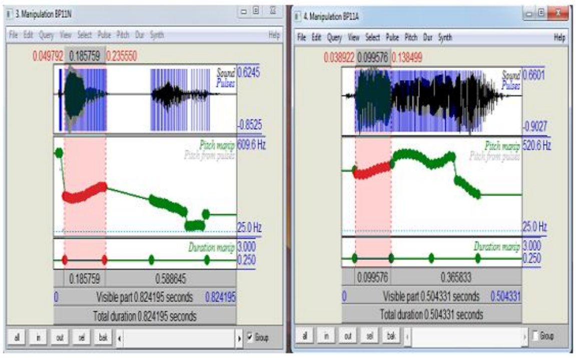

3. Manipulation BP11N
4. Manipulation BP11A
0.049792 0.185759 0.235550
0.038922 0.099576 0.138499
Pitch manip
Duration manip
Visible part 0.824195 seconds
Total duration 0.824195 seconds
Visible part 0.504331 seconds
Total duration 0.504331 seconds

Fig 1: Duration calculation of source and target speech

We notice that the word duration for angry have reduced in the target sample as compared to the input sample. After the duration is matched, the words in the input tone now take the same amount of time as the words from the target tone by manupullation. Fig 2 shows the input speech waveform with the duration is reduced.

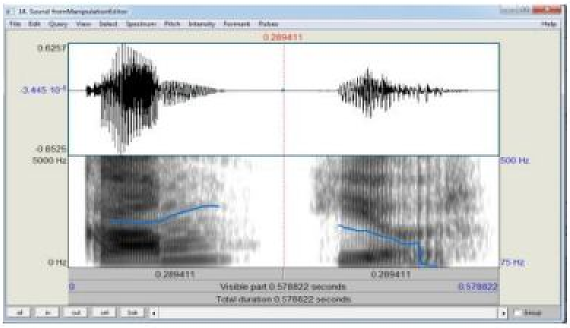

0.289411
5000 Hz
500 Hz
75 Hz
0 Hz
Visible part 0.578822 seconds
Total duration 0.578822 seconds

Fig 2: Final speech sample with reduced duration

We then replace the pitch tier of the output from the input speech sample with duration changed. To conform pitch change, we resynthesize the output of the pitch replacement, and conform that the pitch has indeed changed. The default range for intensity is 50 to 100 db, but it is possible to change it. To do so, we first extract the intensity from the target tone using PRAAT. Finally, Fig:3 shows the comparision of the original input speech sample and the result of applying the concepts: duration change, pitch tier replacement and intensity tier multiplication.

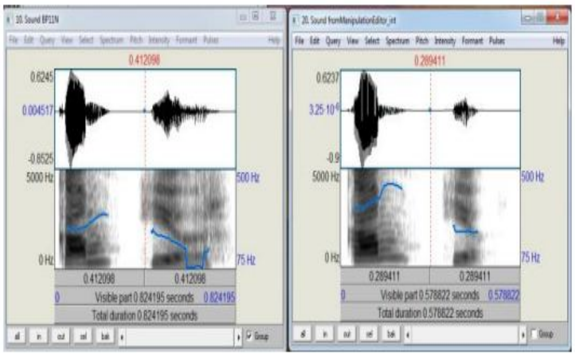

0.412098
0.289411
5000 Hz
500 Hz
75 Hz
Visible part 0.824195 seconds
Total duration 0.824195 seconds
Visible part 0.578822 seconds
Total duration 0.578822 seconds

Fig 3: Source and Target speech after transformation

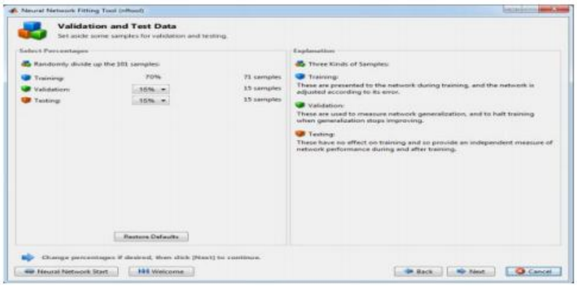
V. ANN USING MATLAB NEURAL NETWORK TOOLBOX

We use MATLAB's Neural Network Toolbox for performing the ANN related act ivities. MATLAB Neural Network Toolbox has several neural network topologies such as fitting tool (nftool), pat tern recognition (nprtool), clustering tool (nctool) and time series tool (ntstool). We use the MATLAB approach to form our ANN and train and test it. It is invoked by the command "nnstart".

The important steps in ANN are:

1. Select dataset (input, output pairs)
2. Train the ANN for minimal MSE
3. Evaluate the trained ANN for a test input

We train an ANN for neutral to angry tone conversion to start with. We use the fitting tool (nftool) for our problem. nftool needs a dataset to operate on. The dataset is a set of 2 matrices: matrix1 (1xN) input, and matrix2 (1xN) output. We have N samples, of which some will be used for training, some for validation, and the rest for testing. We have saved the input/output parameter set as a MAT file, and loaded them into the data input section. Firstly, we use the duration parameters (input is the duration of a spoken word in the neutral tone, output is the duration of the same spoken word in the target tone).The next step is to train the network: some Part of the dataset will be used for training, some part for validation, and the rest for testing. We use the Levenberg-Marquardt back-propagation method to train the ANN. The ANN is said to be trained when the MSE goes below a particular threshold. We use default values for all the thresholds in this training. The start of the training for the ANN, and Fig 5 shows the network architecture selection panel. We use the default architecture, which means 10 neurons in the hidden layer.


Neural Network Fitting Tool (nftool)

Validation and Test Data

Set aside some samples for validation and testing.

Select Percentages

Randomly divide up the 101 samples:

Training:	70%	71 samples
Validation:	15%	15 samples
Testing:	15%	15 samples

Explanation

Three Kinds of Samples:

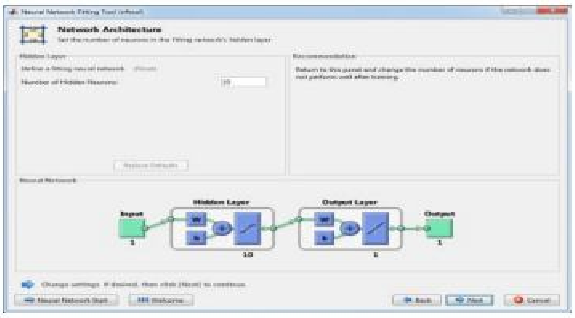
Training:
These are presented to the network during training, and the network is adjusted according to its error.

Validation:
These are used to measure network generalization, and to halt training when generalization stops improving.

Testing:
These have no effect on training and so provide an independent measure of network performance during and after training.

Change percentages if desired, then click [Next] to continue.

Fig 4: nftool: Starting training of the ANN


Neural Network Fitting Tool (nftool)

Network Architecture

Set the number of neurons in the fitting network's hidden layer.

Hidden Layer

Define a fitting neural network. (fitnet)

Number of Hidden Neurons: 10

Recommendation

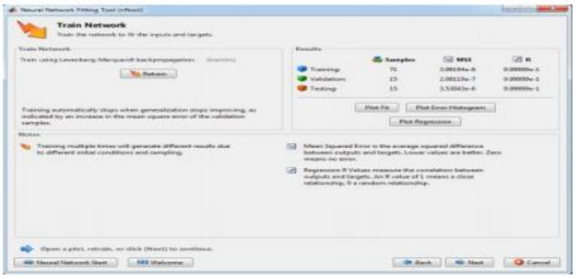
Return to this panel and change the number of neurons if the network does not perform well after training.

Neural Network

Change settings, if desired, then click [Next] to continue.

Fig 5: nftool: Networking architecture selection

Fig 6 shows the trained ANN with corresponding MSE. Now the network is trained and able to accept any test inputs.


Neural Network Fitting Tool (nftool)

Train Network

Train the network to fit the inputs and targets.

Train Network

Train using Levenberg-Marquardt backpropagation. (trainlm)

Training automatically stops when generalization stops improving, as indicated by an increase in the mean square error of the validation samples.

Results

	Samples	MSE	R
Training:	71	[illegible]	9.99999e-1
Validation:	15	[illegible]	9.99999e-1
Testing:	15	[illegible]	9.99999e-1

Notes

Training multiple times will generate different results due to different initial conditions and sampling.

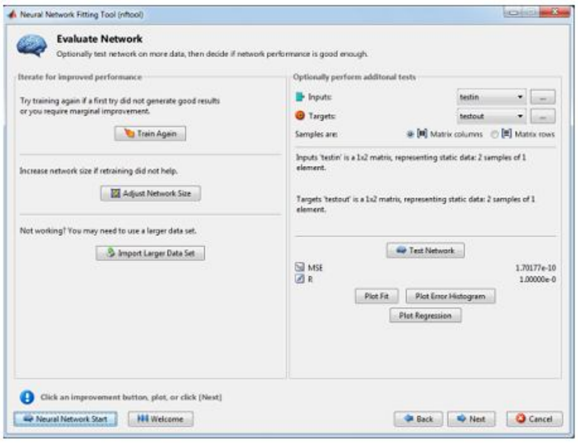
Mean Squared Error is the average squared difference between outputs and targets. Lower values are better. Zero means no error.

Regression R Values measure the correlation between outputs and targets. An R value of 1 means a close relationship, 0 a random relationship.

Open a plot, retrain, or click [Next] to continue.

Fig 6: nftool: ANN fully trained

The next step is to evaluate the ANN by giving it a test input and finding the corresponding output. Fig 7 shows the ANN in evaluation mode where we give it a test input (1x2 vector) and get the resultant output.


Neural Network Fitting Tool (nftool)

Evaluate Network

Optionally test network on more data, then decide if network performance is good enough.

Iterate for improved performance

Try training again if a first try did not generate good results or you require marginal improvement.

Increase network size if retraining did not help.

Not working? You may need to use a larger data set.

Optionally perform additonal tests

Inputs: testin

Targets: testout

Samples are: ● Matrix columns ○ Matrix rows

Inputs 'testin' is a 1x2 matrix, representing static data: 2 samples of 1 element.

Targets 'testout' is a 1x2 matrix, representing static data: 2 samples of 1 element.

MSE 1.70177e-10

R 1.00000e-0

Click an improvement button, plot, or click [Next]

Fig 7: nftool: Evaluating the trained ANN

CONCLUSION

Initial trials with PRAAT and the basic idea of “copying” prosodic parameters from target tone to input tone seem to be able to transform the input emotion to the target emotion.

We then implemented an ANN approach to verify the same “copying” of prosodic parameters from target tone to input tone, and we observe that the input emotion is indeed transformed to the target tone.

REFERENCE

[1] Schroder, M., “Emotional Speech Synthesis - A Review”, Proc. of EUROSPEECH, vol.1:561–564, 1999.

[2] Kawanami, H., Iwami, Y., Toda, T., Saruwatari, H., and Shikamo, K.”GMM-based Voice Conversion Applied to Emotional Speech Synthesis”, IEEE Trans. Speech and Audio Proc., 7(6):697–708, 1999.

[3] Wu, C.H., Hsia, C.-C., Liu, T.-E., and Wang, J.-F., ”Voice Conversion Using Duration-Embedded Bi-HMMs for Expressive Speech Synthesis”, IEEE Trans. Audio, Speech and Language Proc.,vol.14(4):1109–1116, 2006.

[4] Tao, J., Yongguo, K., and Li, A. ”Prosody Conversion from Neutral Speech to Emotional Speech”, IEEE Trans. Audio, Speech and Lang Proc., vol.14:1145–1153, 2006.

[5] Tsuzuki, H., Zen, H., Tokuda, K., Kitamura, T., Bulut, M. and Narayanan, S. ”Constructing emotional speech synthesizers with limited speech database”, Proc. of ICSLP vol.2:1185-1188, 2004.

[6] Yamagishi, J., Onishi, K., Masuko, T., Kobayashi, T., ”Modeling of various speaking styles and emotions for HMM-Based Speech Synthesis”, Proc. EUROSPEECH, vol.3:2461-2464, 2003.

[7] Inanoglu, Z., Young, S., ”A System for Transforming the Emotion in Speech: Combining Data-Driven Conversion Techniques for Prosody and Voice Quality”, Proc. of Interspeech, 2007.

[8] I. R. Murray, et al.: Towards the Simulation of Emotion in Synthetic Speech: A Review of the Literature of Human Vocal Emotion, J. of ASA, 93, No.2, pp.1097-1108 (1993).

[9] A. Iida, et al.: A Speech Synthesis System with Emotion for Assisting Communication, Proc. ICSA Workshop on Speech And Emotion, pp.167-177 (2000).