Neural Network Speech Enhancement for Noise Robust Speech Recognition

Bert de Vries[1], Chi Wei Che[2], Roger Crane[1], Jim Flanagan[2], Qiguang Lin[2], and John Pearson[1]

[1]*David Sarnoff Research Center*
CN5300 Princeton, NJ 08543-5300
{bdevries, rcrane, jpearson}@sarnoff.com

[2]*CAIP Center, Rutgers University*
CoRE Building, Busch Campus
Piscataway, NJ 08855-1390
{werche, qlin, jlf}@caip.rutgers.edu

Abstract

We have developed neural net equalizers that compensate for the effects of mismatched acoustics in the training and operational environments of speech recognizers. We show that neural nets can be used to significantly boost recognition accuracy, without retraining the speech recognizer.

1 Introduction

The performance of current state-of-the-art Hidden Markov Model (HMM) based speech recognizers has improved to the point that commercial applications are feasible. These recognition systems work best for high quality, "close-talking" speech and require consistent environments for training, testing and operation. Typically, the performance of speech recognizers drops drastically if training and operational conditions are not matched.

Degradation of speech signal quality by environmental conditions is due to noise, room acoustics, non-linearities in recording equipment, and compensation by the speaker himself.

Robust speech recognition refers to the art of producing graceful performance degradation when training and testing data set conditions differ. Research on reducing the data set mismatch takes three forms ([1] Gong and Treurniet, 1993). In the first approach the search is for *noise-resistant speech features* which are used in the recognizer pre-processor. It is currently believed that cepstral features are more resistant to noise than the more common discrete Fourier transform coefficients ([2] Erell and Weintraub, 1993). In the second approach the goal is to map noisy speech into cleaner speech. These methods are collectively called *speech enhancement* techniques. The third approach does not work on the received speech signal but instead adapts the recognizer's model parameters to accommodate the changed environmental conditions. Such methods are referred to as *model compensation* techniques.

In this study we explore neural net room acoustics equalizers to expand the use of existing speech recognizers to practical environments where users need not be encumbered by hand-held or attached microphone systems. Applications include combat information centers, large group conferences (tele-conferencing) and mobile hands-busy, eyes-busy maintenance tasks.

In the next section we present our approach to neural net speech enhancement. In section 3 we report on experimental results.

2 Systems Analysis

2.1 Experimental Methodology

We recorded speech data from 50 male and 30 female speakers at a sampling frequency of 16 KHz and used 16-bit linear quantization. Each person spoke 20 isolated commands, 10 digits and 10 continuous sentences. Data were recorded in two sessions. During the first session, speech was simultaneously recorded by a high-quality head-mounted microphone (type HMD-224; these data will be referred to as the close-talking or head-mounted data) and a one-dimensional beamforming line-array microphone located 3 meters from the speaker. In the second session the data were recorded simultaneously by the same head-mounted microphone and a desk-mounted microphone (type PCC 160) at 3 meters distance from the speaker. The recording environment was a regular hard-walled laboratory room (6x6x2.7 meters). Ambient noise in the room was from several workstations, fans, and a large video display machine. The recording and data processing configuration is shown in Figure 1.

To quantify performance improvements achieved with neural networks, we used SPHINX-I ("sphinx-one"), a HMM continuous speech recognition system, developed by Carnegie Mellon University, and a simple dynamic time warping (DTW) recognizer, which we developed ourselves. Before the received speech data is fed to the speech recognizer, a feature extractor converts the speech waveform signal into a cepstral vector signal, which in our case consists of 13 coefficients. Whereas the speech waveform is sampled at 16 KHz, the feature extractor produces a cepstral feature vector at a frequency of 0.1 KHz (the "frame rate" is 10 msecs). As is shown in Figure 1, there are two locations in this configuration where a speech enhancement filter can be placed. We can enhance speech before the feature extractor (the light-shaded box in Figure 1) or after preprocessing (the dark-shaded boxes). Before presenting experimental results, we discuss the consequences of either choice.

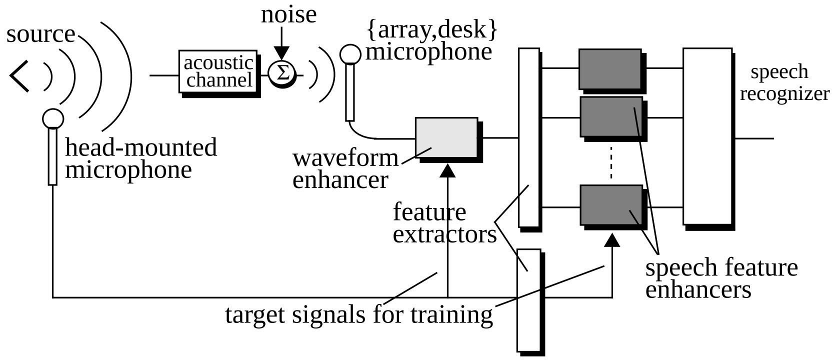

Figure 1 The recording and data processing configuration.

2.2 Speech Waveform versus Feature Enhancement

If a speech enhancement filter is placed before the feature extractor, we might anticipate that a linear equalizer would suffice. We could justifiably use a received speech signal model of the form

$$x(t) = h(t)_* u(t) + n(t) \tag{1}$$

where *h(t)* is the impulse response model for the acoustic channel, *u(t)* the speech signal at the speaker's end, *n(t)* a noise source and *x(t)* the received signal at the distant microphone. Without any information about *n(t)*, the best an equalizer can do is approximate *g(t)* such that $g(t)*h(t) = \delta(t-k)$, that is, the speech enhancement filter is the inverse channel response (modulo some delay).

The speech enhancement filter can also be positioned after the feature extractor, a configuration which we call (speech) feature enhancement. Since the feature extractor already focuses on rather noise-robust features, the task of the enhancement filter seems easier than in the case of waveform enhancement. For this reason we elected to perform our experiments on feature enhancement rather than waveform enhancement.

The feature extraction stage performs irreversible operations such as time averaging (since we go from 16Khz to a 0.1KHz sample rate), as well as non-linear operations (logarithms in the case of cepstral synthesis). Consequently we should not expect that a linear filter is optimal at this stage. In this study we have compared the benefits of non-linear and recurrent nets over linear filters for feature enhancement.

While the ultimate goal of speech enhancement is to improve recognition performance (in terms of word recognition accuracy), we cannot use the word accuracy, a piecewise constant function, as a training cost function for the speech enhancements filters. Instead we must use related continuously differentiable cost functions such as output signal mean square error (MSE). In this paper we study experimentally the relation between signal MSE and recognition performance.

Feature enhancement in comparison to waveform enhancement adds another difficulty since we are dealing now with a vector signal (13 channels in our case) instead of a scalar signal. Should we use a single net with 13 inputs and 13 outputs, or is it better to assign different networks to the individual channels? There are arguments for either approach. We do not know if a particular “number of channels per network” assignment is consistently better than other alternatives. Experimental results are reported only for the one-net-per-channel approach.

3 Experiments

3.1 Comparative Evaluation across Microphone Types

In the first set of experiments we used the SPHINX-I speech recognizer. We used one neural network per channel for a total of 13 different neural nets. The networks had similar architectures for all channels. The network configuration is shown in Figure 2. The input layer has 9 taps, covering the current frame, the four preceding and four following frames. The input layer fully connects to a 4-node hidden layer of sigmoid nodes, which connect in turn to a single linear output node.

testing dataset identified by microphone type	accuracy (%) on **pre-trained** SPHINX	accuracy (%) after **re-training** on close-talking	accuracy (%) after **re-training** on line-array
close-talking	88	95	-
line-array (LA)	16	21	82
LA+neural net	82	85	-

Table 1: Results on SPHINX for varying dataset environmental conditions.

The input data is drawn from the (cepstral coefficients of the) line array data and the target data is taken from the simultaneously recorded head-mounted microphone data set. These neural nets were individually trained to minimize the MSE between corresponding frames of the input and target data set. We trained on approximately 4 seconds of speech material (isolated words) recorded from a single speaker. The networks were trained in speaker-dependent mode. The goal of the experiments is to estimate the maximal performance gain achievable with neural network enhancement. Results for speaker-independent training would very likely be less favorable. We also want to compare results across different microphone types. The results are captured in Table 1. See also ([3], Che et al. 1994) for a more elaborate account. When we used the SPHINX recognizer, which was pre-trained by the developers, on data recorded by the head-mounted microphone, 88% of the words were correctly classified. When we used line-array data and no neural nets, the performance dropped to 16%. A substantial improvement to 82% performance is observed when the line array data were used in conjunction with feature enhancement neural nets.

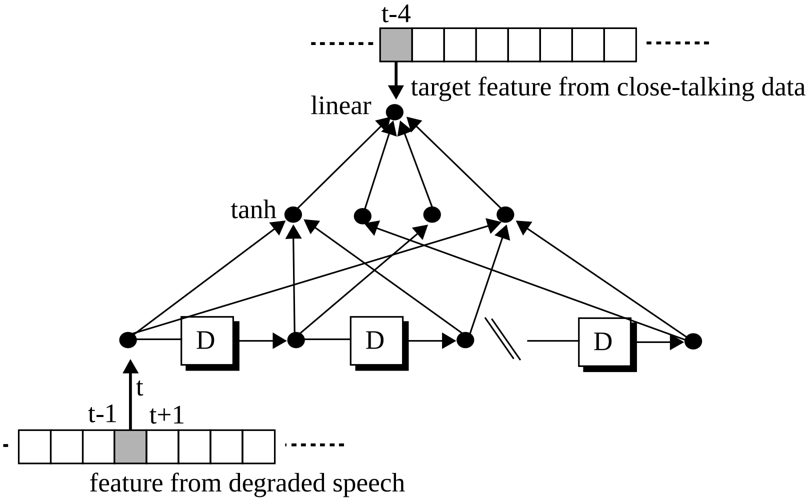
t-4
target feature from close-talking data
linear
tanh
D
D
D
t
t-1
t+1
feature from degraded speech

Figure 2 The basic configuration of the neural nets used in our experiments.

Next we re-trained the SPHINX system both on data from the head-mounted microphone as well as on line array data. As expected the performance is very good when the training and testing sets are matched. We found 95% and 82% word accuracy on the head-mounted and line-array data respectively. The results in Table 1 illustrate the main result of this experiment: *whereas it is possible to re-train the HMM recognizer on a matching dataset in order to recover good performance, it is faster to train a (set of) small neural nets with comparable performance results* (relative to HMM re-training). Furthermore it is remarkable that re-training on the (noisy) line array data (82%) instead of the cleaner (better SNR ratio) head-mounted data (21%) greatly improves the performance when we test on line array data. Apparently, matching training and testing data sets is preferable over training on clean data.

3.2 Comparison across Network Architectures

In a second set of experiments we used a simple dynamic time warping recognizer which we developed ourselves. The purpose of these experiments was to determine the benefits of non-linear and recurrent nets relative to linear filters. Furthermore we want to determine the relation between the MSE criterion for neural net training and the DTW word accuracy performance.We trained three different architectures, a linear feedforward net (adaline, adaptive transversal filter), a non-linear layered feedforward net, and a non-linear recurrent network. We used 13 nets per experiment, one for each channel. The transversal filter had 9 taps. The non-linear feedforward net had 9 taps in the input layer, 5 sigmoidal hidden units and 1 linear output node. The recurrent net was similarly configured but the hidden layer was fully connected to itself as well (an "Elman" network). The line array data were used as input data and the head-mounted data set as target data for neural net training. We used batch training, backpropagation with adaptive learning rate and no momentum. Twenty words from 6 different speakers (120 words total) were used both for training and testing (where, of course, the testing set made use of different words than the training set). The DTW recognizer was trained speaker-dependently on the head-mounted data set. The recognizer performance was tested after every neural net training epoch on data from the line array. Results are displayed in Figure 3, where we plotted both the sum of the MSE over the 13 nets as well as the DTW performance as a function of training epoch. These graphs provide some insight in the relation between neural net MSE and DTW performance. The graphs in Figure 3 are all normalized between 0 and 1. For all networks, the DTW performance asymptotes at approximately 0.85, i.e., 85% word recognition accuracy. When we did not use a neural net, a performance of 38% was observed when the DTW recognizer, trained on close-talking data, was tested on line array data. Hence, the feature enhancement nets were able to boost recognition performance from 38% to approximately 85%. It is surprising that the linear transversal filter works as well as the non-linear feedforward and recurrent nets. In the previous section we hypothesized that we needed non-linear processing for feature enhancement. These experiments seem to indicate that linear processing performs just as well, although we do not want to draw generally valid conclusions from one experiment.

An interesting phenomenon is the unstable performance of the DTW recognizer at early stages of learning. It is clear from these graphs that recognition accuracy is not monotonically related to neural net MSE. Yet, if we train long enough the DTW recognizer performance stabilizes for all networks. In Figure 3(d) we plotted the MSE for the individual adaline nets for channels 0, 6, and 12. This plot in combination with Figure 3(a) suggests that the recognizer performance stabilizes only after the "slowest-learning" network (for channel 12) has converged (after about 100 epochs). Apparently the higher-order channels are instrumental for reliable recognition.

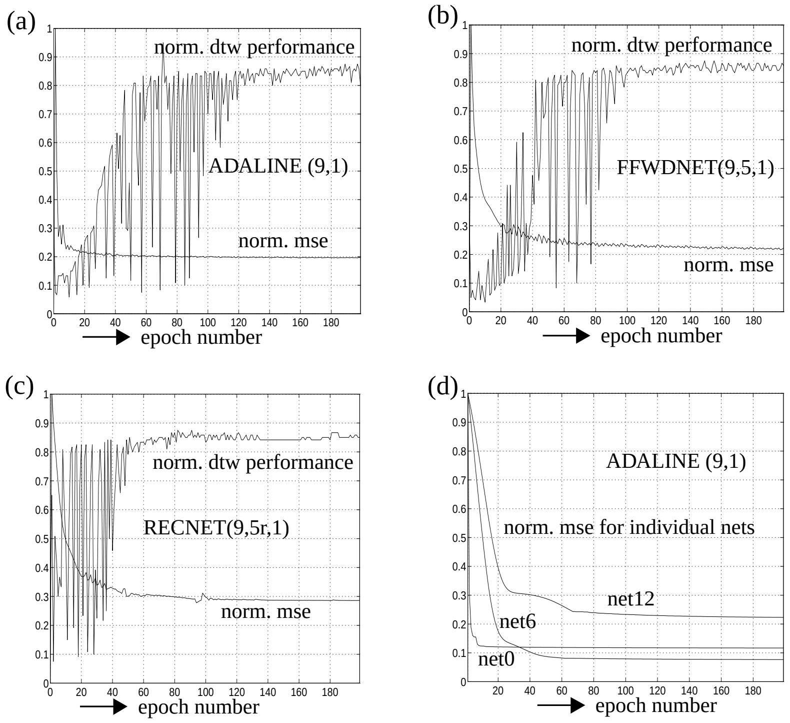

Figure 3 Results on DTW word accuracy using different neural nets as speech feature enhancers. All curves are normalized between 0 and 1. Neural net training proceeded in batch mode, with adaptive learning rates (and no momentum term). The DTW performance curves are relative to the test set. The MSE curves are with respect to the training data set.

4 Conclusions

We have used neural networks for speech feature enhancement in a speech recognizer. From a more general perspective, these experiments are about boosting performance when the training and testing sets are not well matched. This theme is unfortunately rather common in practical applications for adaptive systems, since real world environments are typically non-stationary. We do not claim general truths here, but we noticed some interesting results in our experiments. In general, the technique works very well. In our first set of experiments we were able to boost recognition performance from about 20% to about 80%. In our second set of experiments, on another recognizer, with different data sets, recognition accuracy was improved from 38% to about 85%.

An important conclusion is that recognition performance with neural net speech enhancers in mismatched training and testing data sets appears comparable to re-trained HMM recognizers (without neural nets). The neural net approach avoids re-training of the recognizers, which is a tedious and time-consuming process. We also noticed that it is not always a priority to train on the cleanest data available. It is more important to have matching conditions for the datasets. If the testing environment is noisy, it appears to be better to also train on data with low signal-to-noise ratio. We found that for the case of cepstral feature enhancement, quite surprisingly, the linear transversal adaptive filter performs as well as non-linear and recurrent networks. This result needs further investigation. Furthermore, as expected, the MSE cost criterion for neural net training appeared not to be monotonically related to word recognition accuracy. On the other hand, when training continued long enough, i.e., until the MSE for all nets converged, the DTW performance was stable and MSE training did lead to an excellent performance boost.

Acknowledgments

We are grateful to Dr. Richard Stern and the speech group at Carnegie Mellon University for providing us with a copy of the SPHINX speech recognition system. This work is supported by ARPA contract no. DABT63-93-C-0037.

References

[1] Gong Y. and Treurniet W.C., Speech recognition in noisy environments: a survey, Technical report CRC-TN 93-002, Communications Research Centre, Department of Communications, Ottawa, 1993.

[2] Erell A. and Weintraub M., Filterbank-energy estimation using mixture and Markov model recognition of noisy speech, *IEEE transactions on speech and audio processing*, vol.1, no.1, 1993.

[3] Che C., Lin Q., Pearson J., de Vries B., and Flanagan J., Microphone arrays and neural networks for robust speech recognition, *proceedings of ARPA workshop on Human Language Technology*, Plainsboro, NJ, 1994.

Neural Network Speech Enhancement for Noise Robust Speech Recognition

Bert de Vries[1], Chi Wei Che[2], Roger Crane[1], Jim Flanagan[2], Qiguang Lin[2], and John Pearson[1]

[1]David Sarnoff Research Center
CN5300 Princeton, NJ 08543-5300
{bdevries, rcrane, jpearson}@sarnoff.com

[2]CAIP Center, Rutgers University
CoRE Building, Busch Campus
Piscataway, NJ 08855-1390
{werche, qlin, jlf}@caip.rutgers.edu

keywords

speech enhancement, channel equalization, room acoustics, speech recognition