MLP NETWORK FOR ENHANCEMENT OF NOisy MFCC VECTORS

Hemmo Haverinen\(^1\), Petri Salmela\(^1\), Juha Hääkkinen\(^2\), Mikko Lehtokangas\(^1\) and Jukka Saarinen\(^1\)

\(^1\) Tampere University of Technology
Signal Processing Laboratory
FIN-33101 Tampere, Finland
Email: hemmo@cs.tut.fi

\(^2\) Nokia Research Center
Tampere, Finland

ABSTRACT

The performance of voice dialling systems often degrades rapidly as the intensity of the background noise increases. In this paper, we describe a neural network based speech enhancement technique for improving the speech recognition performance of a voice dialling system in very noisy real world type conditions. The speech samples were recorded in laboratory conditions and afterwards corrupted by adding car noise or babble noise recorded in a cafe. These noise corrupted speech samples were enhanced in cepstral domain by a context dependent multilayer perceptron (MLP) network before performing the recognition using a hidden Markov model (HMM) based speech recognition system. The accuracy of the test set increased 58\%, 55\% and 46\% in the car noise environments having -5 dB, 0 dB and 5 dB SNRs, respectively. The accuracy of the test set increased 44\%, 48\% and 39\% in the babble noise environments having SNR 5 dB, 10 dB and 15 dB, respectively. The accuracy remained approximately same for both car and babble noise environments when having SNR of 20 dB.

1. INTRODUCTION

So far the various noise compensation techniques that have been proposed in the field of speech recognition have not been able to solve the problem. Precise modelling of the noise is often impossible because of the varying intensity and the nonstationary nature of the background noise. One way to approach the problem is to try to enhance corrupted feature vectors before they are fed into the recognition unit. This has been done using e.g. spectral subtraction, Kalman filtering and hidden Markov model (HMM) based noise compensation models [1]. However, the filter and model based techniques above usually suffer from complexity of structure. Also, the training of the models or defining the proper filters can be rather difficult. An alternative approach to above methods have been the techniques that try to modify the parameters of the recognition unit. An example of such a technique is the parallel model combination (PMC) technique [2].

The performance of many traditional noise compensation techniques, such as spectral subtraction, depends strongly on the accuracy of the estimated background noise intensity. The approximation of the noise intensity often requires that it should be possible to classify the signal into speech and non-speech segments. This speech/non-speech classification of the signal is known to be a difficult task especially when having SNRs below 0 dB. Voice activity detectors (VADs) are these kind of classifiers but currently they do not perform adequately for the speech recognition purposes. The performance of e.g. the PMC technique is based on the accuracy of this kind of signal classification [3].

This paper proposes an enhancement technique for noisy mel-frequency cepstral coefficients (MFCCs) using a context dependent multilayer perceptron (MLP) network. This enhancement technique was chosen since contextual information can be easily incorporated in the MLP network, and it has been shown that the MLP can approximate any continuous function within certain constraints [4]. The enhancement of cepstral feature vectors using neural networks has been researched earlier but differing from our method these methods did not use e.g. context vectors and derivative coefficients of MFCCs [5, 6, 7].

2. ENHANCEMENT SCHEME

The block diagram of the recognition system used in the experiments is depicted in Figure 1. The figure shows that the speech signal is corrupted by additive car or babble noise before it is transformed to MFCC vectors in the front-end. This front-end produces one feature vector at every 10 ms. These noise corrupted feature vectors consist of 12 static MFCCs and an energy coefficient. Since the noise and speech are combined in a nonlinear way in these noise corrupted MFCC feature vectors, i.e. through a log function, a nonlinear system is required for noise compensation. A multilayer perceptron (MLP) network can approximate the required nonlinear function to some extent [4]. Therefore, as illustrated in Figure 1 an MLP network is included as a part of the signal preprocessing unit and it is used to enhance the original corrupted MFCC vectors.

The noisy input vector to MLP at time \( n \) is denoted by \([C_{n}(1)...C_{n}(13)]\) and the corresponding network output vector by \([C'_{n}(1)...C'_{n}(13)]\) which is the estimate of the
12 clean static MFCCs and an energy coefficient. These estimated coefficients were used for the calculation of their first order derivative coefficients $\Delta C'_n$ such that

$$\Delta C'_n(i) = \sum_{r=-3}^{3} r \cdot C'_{n+r}(i), 1 \leq i \leq 13 \quad (1)$$

where $i$ denotes $i$th element of vector $C'_n$. Moreover, the second order coefficients $\Delta \Delta C'_n$ were obtained as

$$\Delta \Delta C'_n(i) = \sum_{r=-2}^{2} r \cdot \Delta C'_{n+r}(i), 1 \leq i \leq 13. \quad (2)$$

Then, all calculated derivative coefficients were appended to the $C'_n$ and the resulting noise compensated feature vector of 39 elements was fed into the hidden Markov model (HMM) based speech recognition system explained in Section 2.2.

However, the use of MLP noise compensation does not come without a cost. It is a well known fact that model based approaches such as an MLP cannot extrapolate usually outside the space of training data [8]. However, this can be compensated to some extent by using an extensive training data set. Another cost of using MLP noise compensation is the large number of parameters as well as long training time compared to e.g. PMC.

### 2.1 Structure of MLP

In order to provide the MLP with the context information the input of the MLP consisted of nine consecutive feature vectors, specifically 4 previous, the current and next 4 feature vectors. None of these vectors nor the output feature vector did include derivative coefficients but only 12 static MFCCs and an energy coefficient. The MLP was fully connected and the number of input, hidden and output neurons in our tests were 117, 200 and 13, respectively, resulting 26213 parameters. The number of hidden neurons was determined by a small number of tests. The activation functions were hyperbolic tangent and linear in the hidden and output layers, respectively.

### 2.2 Recognition unit

The word recognition unit contained a whole word HMM for each vocabulary word. Each word model had 12 to 18 states proceeding from left to right such that only a self-loop and forward pass were allowed. Moreover, each state in the word model was duration constrained by using loose maximum and minimum duration bounds [9]. These HMMs were one mixture models having constant variances set to one. The training of each HMM was performed using one utterance of 20 dB SNR. After finding the endpoints of this utterance, the feature vectors were evenly assigned for each state of the model. The mean vector of the HMM state was simply obtained as an average of the feature vectors assigned to the state. The duration constraints were also loosely determined for each state according to the number of feature vectors assigned to the state. During recognition of an utterance an $n$th best garbage model was used along the word models.

### 3. DATA SETS

The utterances of the data set were recorded in laboratory conditions and afterwards corrupted by adding real car noise or babble noise. The SNR intensities of added noise ranged from approximately 20 dB to -5 dB for the car noise and from 20 dB to 5 dB for the babble noise. The database consisted of isolated utterances of names pronounced by 1 female and 4 male speakers. The male speakers are denoted by male1, male2, male3, male4 and the female by female. The training, cross-validation and test sets contained 1920, 270 and 2400 utterances, respectively. The number of speakers in these sets were 4, 3 and 5, respectively. The training set consisted of utterances from male1, male2, male3 and female, the validation set from male1, male4 and female and the test set contained utterances from all the five speakers. However, neither of the sets contained utterances in common.

There were four background noise environments for both noise types differing from SNR. In each of these environments every speaker pronounced 12 repetitions of the vocabulary of 30 names. Every speaker had his/her own 30 name vocabulary. One utterance consisted of three parts, specifically pure noise parts in the beginning and at the end and noisy speech part in between. The parts of the utterances that contained only noise were shortened in order to increase the portion of the speech in the data and to reduce the computation time. Hence, the data of the training and validation sets were cut in a way that the resulting sets contained one third of speech and the rest was pure noise. The test data set was not touched.

### 4. TRAINING OF MLP

Before training the MLP, the HMM recognizer was trained using the utterances of the training set having approximately 20 dB SNR. After that the parameters of the HMM were kept fixed and the weight and bias param-

![Figure 1. The structure of the speech recognition system](image-url)
eters of the MLP were initialized according to the
Nguyen-Widrow initialization algorithm [10]. Then the
weight parameters were adjusted with the momentum
backpropagation training algorithm using the mean
square error (MSE) as a cost function [11, 12]. During
each training iteration the gradient was calculated over
one utterance of the training set. The decision, whether
the adjustment to the weights was made, was based on
the validation set. If the MSE over validation set
decreased then the weight change was approved, the
learning rate parameter was increased by a small factor
and the momentum parameter was set to a small value.
Otherwise the weight change was discarded, the learning
rate parameter was decreased by a small factor and the
momentum parameter was set to zero. After the learning
rate parameter had been decreased to a very small value,
the training was stopped. The samples of the training set
were usually processed about once during a training ses-
sion. The weights of the MLP were kept fixed during the
determination of the test set recognition percentages.

The training data set consisted of the input and the corre-
sponding target vector pairs explained in Section 2.1.
Since the elements of the original MFCC vectors differ in
the magnitude of the variances, the element values need
to be normalized in order to get proper convergence with
the MLP. This is due to the fact that the linear activation
function and MSE cost function tend to emphasize the
elements having large variances. This phenomenon was
verified with the very first MLP training experiments
which did not improve the baseline recognition accuracy
at all. This problem vanished when each feature vector
component was normalized by the corresponding mean
and standard deviation calculated over the utterance.

5. SIMULATION RESULTS

The recognition results for the car noise and babble noise
are illustrated in Figures 2 and 3, respectively. The results
are averages of 10 simulations. The simulations in car
noise did not include any babble noise utterances and
vice versa. The dashed line and the dotted line in Figures
2 and 3 are the average recognition rates without MLP
noise compensation and with MLP noise compensation,
respectively. The solid line illustrates the rates when the
HMM parameters were trained further with the MLP
compensated feature vectors. For each of these three
cases the feature vectors were normalized in a way told in
Section 4. All utterances of the sets were classified to the

Figure 2. The average recognition rates in car noise for (a) training, (b) validation and (c) test sets. The dashed lines depict the rates
without MLP noise compensation. The dotted lines depict the rates with MLP noise compensation and the solid lines depict the
results when the HMM parameters were trained further with MLP compensated feature vectors.

Figure 3. The average recognition rates in babble noise for (a) training, (b) validation and (c) test sets. The dashed lines depict the rates
without MLP noise compensation. The dotted lines depict the rates with MLP noise compensation and the solid lines depict the
results when the HMM parameters were trained further with MLP compensated feature vectors.
best matching name model which means that all recognition errors were substitutions.

Without the MLP noise compensation the average recognition rates for the test set were 74.8%, 90.2%, 95.5% and 99.8% in the car noise environments having approximately -5 dB, 0 dB, 5 dB and 20 dB SNRs, respectively. The results show that when using the MLP compensation technique, the recognition rates increased on average to 87.2%, 93.8% and 96.3% in the environments of -5 dB, 0 dB and 5 dB, respectively. Moreover, the recognition rate remained almost the same in 20 dB SNR i.e. in 99.4%. These results improved further when the HMM parameters were trained further with the MLP compensated feature vectors. The training was done with the training set utterances having approximately 20 dB SNR. The improved recognition rates of the test set were 89.3%, 95.5%, 97.6% and 99.4% in the environments having -5 dB, 0 dB, 5 dB and 20 dB SNRs, respectively. Figure 3 shows that the MLP noise compensation improves the recognition percentages also in babble noise. The relative improvement in the recognition accuracy was on average 44%, 48% and 39% for the test set environments having SNR 5 dB, 10 dB and 15 dB, respectively. The accuracy remained approximately the same in environment having SNR 20 dB. These results show significant improvement in the recognition accuracy in very noisy conditions.

Previously we have tested the MLP noise compensation with different context structure in the car noise environment [13]. In those tests the MLP noise compensation was tested without context vectors in the input of the MLP and also when only one previous and one future context vector were used in the input of the MLP. All the input vectors including the context vectors included 1st and 2nd order derivative coefficients though the output vector of the MLP had same form as presented in this paper. The output vector was used for the calculation of the derivative coefficients as told in Section 2. The performance without the context vectors was distinctly lower than with the context structure. Furthermore, the context structure presented in this paper improved slightly the performance compared to the one tested earlier.

6. CONCLUSIONS

The results show that a context dependent multilayer perceptron (MLP) network can compensate noise from mel-frequency cepstral coefficients and hence improve recognition accuracy in noisy environments. The improvement in the recognition accuracy was for the car noise on average 58%, 55% and 46% for the test set environments having SNR -5 dB, 0 dB and 5 dB, respectively. The improvement for the babble noise was on average 44%, 48% and 39% for the test set environments having SNR 5 dB, 10 dB and 15 dB, respectively. The accuracy remained approximately the same with both noise types in environments having SNR 20 dB.

REFERENCES