EXPERIMENTS WITH ARTIFICIAL NEURAL NETWORKS FOR PHONEME AND WORD RECOGNITION

Kjell Elenius
Department of Speech Communication and Music Acoustics
KTH, Box 70014, S-10044 Stockholm, Sweden
kjell@speech.kth.se

Mats Blomberg
Department of Speech Communication and Music Acoustics
KTH, Box 70014, S-10044 Stockholm, Sweden
mats@speech.kth.se

ABSTRACT

An artificial neural network has been trained by the error back-propagation technique to recognise phonemes and words. The speech material was recorded by a male Swedish talker and was labelled by a phonetician. There were 38 output nodes corresponding to Swedish phonemes. Introducing coarticulation information by adding simple recurrency to the net is shown to more effective than expanding the size of the input spectral window. The phoneme recognition network was used with dynamic programming for time alignment to recognise connected digits in a speaker independent way. It was compared to a similar recogniser based on nine quasi-phonetic features instead of 38 phonemes. The phoneme based system performed better than the feature based one for five out of seven speakers.

1. INTRODUCTION

1.1. Aim of the study

In an earlier paper [1] (condensed in [2]) we have published results on experiments concerning phoneme recognition using neural networks. In this paper we do some further investigations on the same type of networks to evaluate different network structures and other features of the recognition system.

1.2. The speech material

The speech material was recorded by a male Swedish speaker and was sampled at 16 kHz using a 6.3 kHz low-pass filter. The sentences were phonetically labelled [3]. The labelling was basically phonemic and did not show allophonic variations. Retroflex variants of the phonemes *r, n* and *d* were not treated as separate phonemes and in contrast to our earlier experiment all the occlusions for the voiceless stops *p, t* and *k* were labelled by the same label: *oc* and not by three separate labels. There were 38 phonemes in all. Fifty sentences were used for training and another fifty were used for testing. The smoothed output of 16 Bark scaled filters in the range 200 Hz to 5 kHz were input to the network. The interval between successive speech frames was 10 ms and the integration time was 25.6 ms - the filter outputs are calculated from a 256 point FFT. The number of phonemes for the training material was 2202 and the total number of 10 ms frames was 15258 (2064 and 13671 for the test material).

1.3. Network training

The standard error back-propagation training algorithm [4] was modified in order to increase the training speed. Outputs are treated as probabilities and we have used the cross-entropy cost function. This leads to a simple error derivative with respect to the total input received by an output node - it is equal to the difference between the target and output activations [5, 6]. For our material this typically yields a faster convergence by a factor of about 5 compared to the more standard quadratic error function [4]. We have used a per pattern updating of weights. This will make the gradient descent path move around in a random fashion, which may help it from getting stuck in local minima, something that otherwise may happen for this material using a per epoch training [7].

We also have used a technique where we do not update weights for patterns with output errors below a (modifiable) threshold. This will speed up training time and at the same time it will focus the training on patterns having larger output errors. The nets have been trained until the total output error has reached an asymptotic value. Our experience is that the performance of the trained networks is within one percent of the mean performance over repeated trainings for the same net, using different initial weights. This indicates that the size of the training set is satisfactory large.

2. PHONEME RECOGNITION EXPERIMENTS

The phoneme recognition performance has been evaluated on the frame level. Each frame has been assigned to the phoneme that has the highest output activation. If this phoneme corresponds to the manually set label of the frame it is evaluated as being correct. If many phonemes have the same maximum activation and one of them corresponds to the manual label, this has also been measured as a correct frame. For the three-digit strings we have counted the number of correct words compared to the total number of words.

2.1. Quasi-phonetic features

In our earlier study [1] using the same speech material we used an input window of 70 ms over seven quasi-phonetic features to recognise phonemes [8, 9]. The performance was 54.2% correctly recognised phonemes on the frame level. Converting that result, where occlusions were labelled separately, to the current 38 phonemes by removing all substitutions between them, increases this performance to 56.3%. To test the effect of the use of features we made some studies with similar network structures but without using explicit features. In our earlier study we first trained a feature net to extract 7 phonetic features for each 16 channel speech frame. The outputs of seven consecutive feature frames centred around the phoneme frame to be recognised were then used as input to the phoneme recognition net. In this study the 16 input spectral nodes for each 10 ms speech frame were connected to 7 ordinary, arbitrary, hidden nodes instead. A window of 10 to 70 ms over these nodes were connected to another hidden layer of 20 nodes that in turn were connected to the output phoneme nodes to make the network structure similar to the earlier net.

The results in Figure 1 imply that introducing features seem to have a restricting rather than supporting effect upon the network. It seems better to let the network decide how the spectral input should be utilised than introducing the phonetically based feature set, at least for the feature set we used. The figure indicates that increasing the input window from 30 to 70 ms has a minor effect. This may arise from the fact that enlarging the window will also increase the number of connections to the upper hidden layer and that 20 hidden nodes are too few to handle all this information. However, this was used for compatibility with the earlier experiment. It should be noted that the feature nodes of the earlier net all had the same weights to the input nodes, whereas each of the seven ordinary nodes were allowed to have arbitrary weights to the input. (The earlier 70 ms network was somewhat more complex, but retraining an identical net structure, without features, also gave a 59.6% recognition score.) One should remember the current result is for one speaker only and one objective for using features in the earlier study was to improve performance for different speakers, something we will deal with in section III.

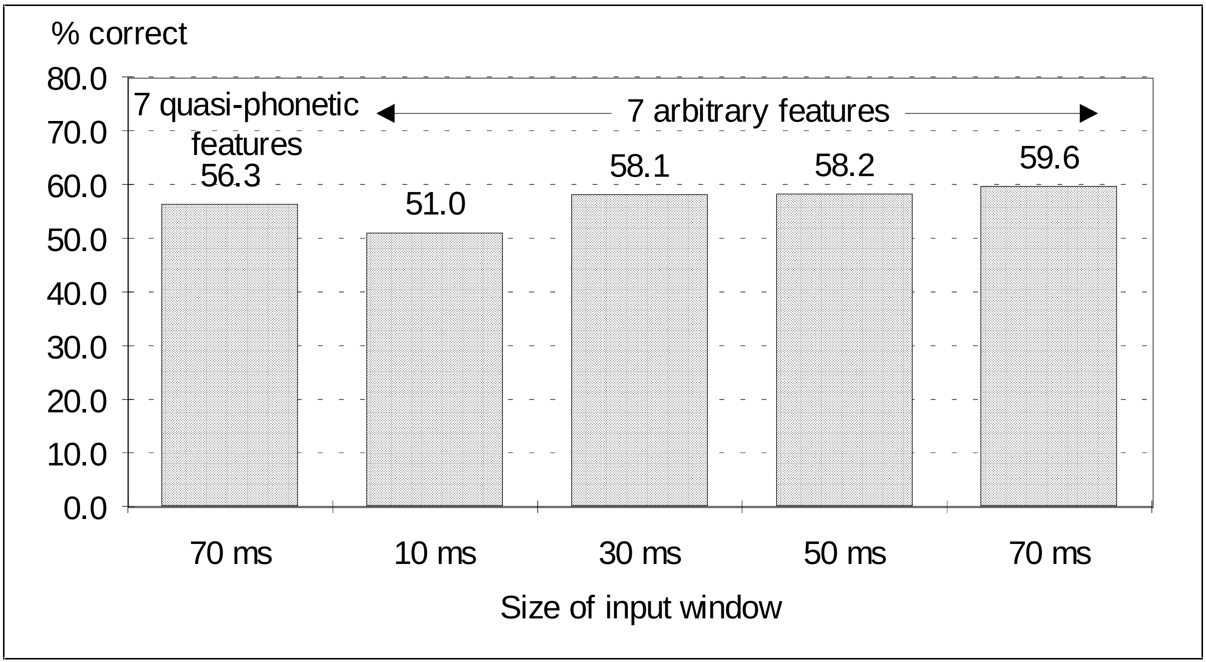

Fig. 1. Comparing quasi-phonetic and arbitrary features for phoneme recognition.

2.2. Recognition without feature nodes

Since the use of features did not enhance phoneme recognition performance we tested a simple net structure where we removed the first hidden layer. The input spectral amplitudes were fed to a hidden layer of 32 nodes for windows 10 and 30 ms and 64 hidden nodes for a 50 ms window. Results in Figure 2 show an improvement compared to the feature nets.

The use of an input window is one way of dealing with coarticulation and context [10, 11, 12]. Adding recurrency is another, compare Jordan [13], Watrous [14] and Robinson and Fallside [15, 16]. We have tried adding simple recurrency to the network using the technique of context nodes, compare Elman [17] and Servan-Schreiber, Cleeremans & McClelland [18]. Recurrency allows the net to build up a "memory" of discrimination relevant features, admitting for different integration times for different features.

We trained a 10 ms net that had 16 input nodes, 32 hidden nodes and 38 output nodes. The 10 ms delayed value of the hidden nodes were connected to themselves and the 10 ms delayed values of the output nodes were also connected to themselves. Introducing recurrency gives a substantial increase in performance as seen in Figure 2. The 10 ms net with context performs about the same as a net with 70 ms input window that has much more weights. Combining a 50 ms input window with context nodes still raises performance, but overall it seems that recurrency is a stronger factor than input window size.

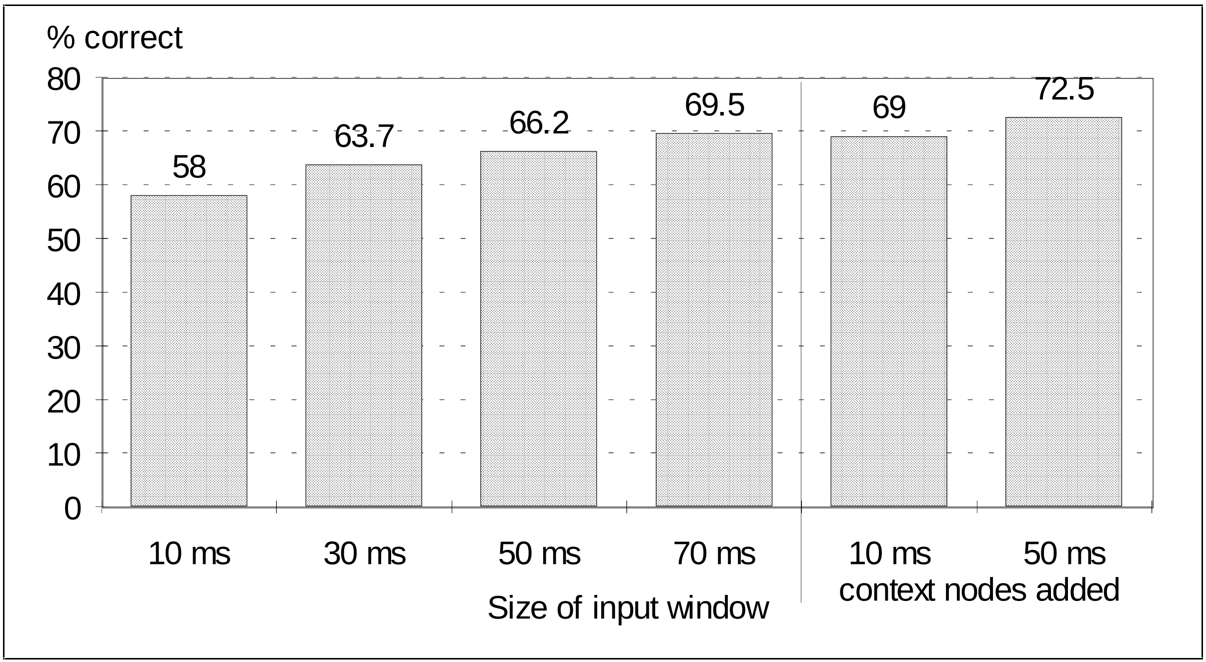

Fig. 2. Phoneme recognition performance per 10 ms frame for some nets trained with only one single hidden layer.

2.3. Sensitivity to misaligning spectral and label frame

The consequences of misaligning the input spectral frame relative to the position of the corresponding label has been examined. The position of the phoneme borders are marked in the time domain and considering the 16 kHz sampling rate they have a theoretical resolution of 1/16th ms, though phoneme borders are of course not really that exact. The spectral frames have a 10 ms resolution and there is a 2.5 ms offset introduced by the way the 25 ms FFT-window is aligned with respect to the frame borders.

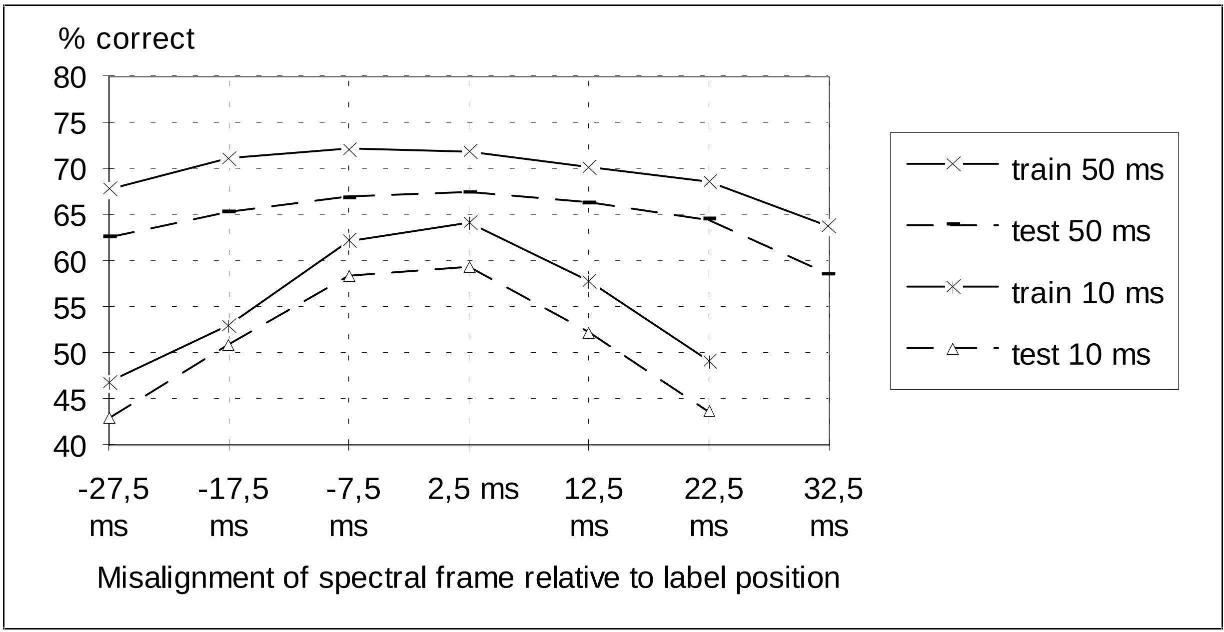

Fig. 3. Performance degradation when misaligning a spectral frame with the corresponding label position. Results for training set and test set and input windows 10 and 50 ms.

The results for the 50 ms input window in Figure 3 show that misaligning the spectral and target frames gives a decrease in performance that is close to symmetric around the correct position. It indicates that the spectral information is symmetrically distributed around the frame to be labelled. The result is not unexpected but still it is an evidence of this fact and implies that the spectral information regarding a phoneme is evenly distributed in the input window, at least as a mean over all phonemes. The optimal target label for an input window should thus be chosen from its midpoint. The 10 ms results show the sensitivity in performance to the number of phoneme borders. A perfect recogniser would give one erroneous frame per phoneme border in this case, which would correspond to 15% frame recognition errors for our test material.

2.4. Number of training frames and phoneme recognition

A typical relation between the percent of frames for each phoneme in the speech material and the recognition rate for that phoneme is shown in Figure 4, which is from the experiment with a 50 ms window with recurrent nodes. It is obvious that the number of training frames has a strong effect on recognition rate. The most frequent phonemes all have high rates.

The recognition rates for all phonemes lie above a line from the origin to the *e*-phoneme. Phonemes furthest away from this line along the Y-axis have a better performance than other equally frequent phones. Some of these seem to have typical spectral shapes, like the fricative sounds *f*, *s*, *tj* [ç] and *sj* [█], and this will of course help their identification. Symbol *oc* stands for the occlusion part of the unvoiced plosives *p*, *t* and *k*. There is no marked difference between vowels and consonants.

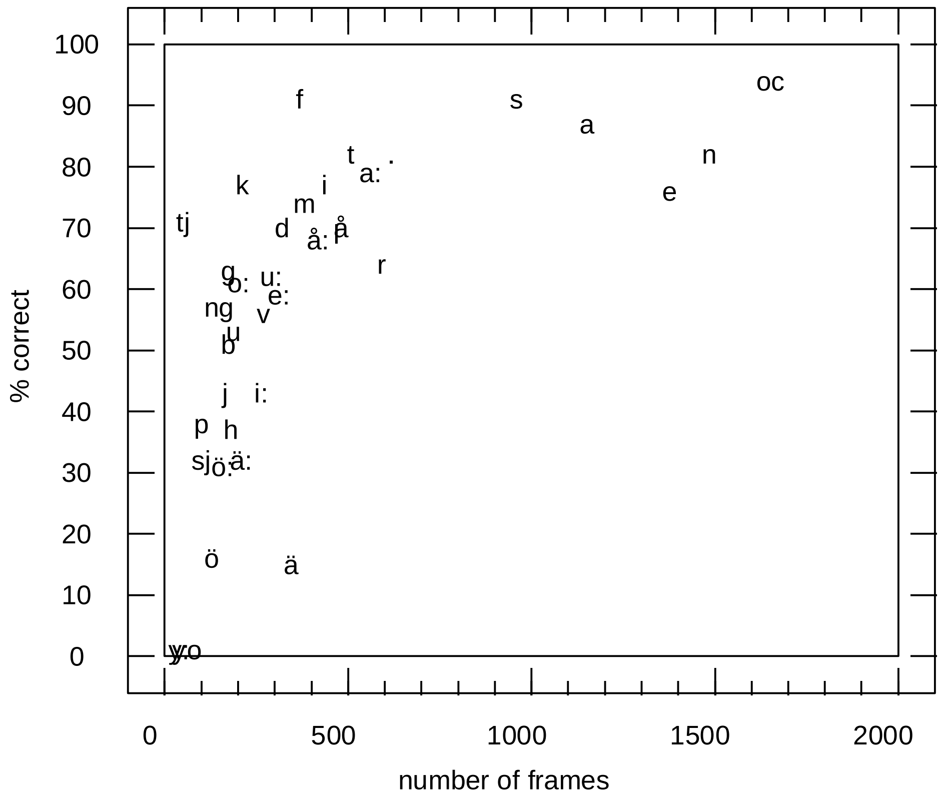

Fig. 4. *Relation between the amount of training frames and the recognition performance for each phoneme.*

2.5. Performance and activation strength

Figure 5 concerns the same experiment as above and shows a strong correlation between the mean activation of a phoneme in the test material and its performance. The mean activation has been calculated over all frames in the test material according to the phonetic labelling done by the phonetician. It is seen that one can interpret the output activation strength as the probability of being correct. Phonemes with low activations are all low frequency, compare Figure 5 above, and have a larger spread in performance.

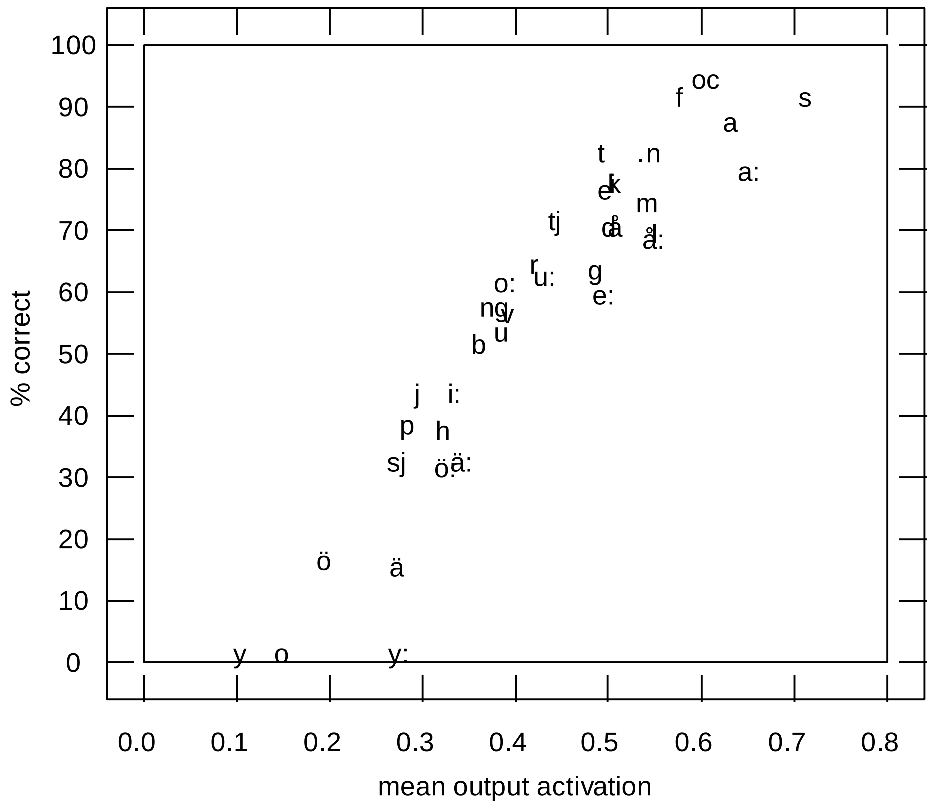

Fig. 5. *Relation between the mean output activation and the recognition performance for each phoneme.*

3. PHONEMES OR FEATURES FOR RECOGNITION

The output of the phoneme network has been used together with dynamic programming for time alignment to recognise connected speech. We used another speech material of three-digit sequences, which was analysed in the same way as the speech above.

3.1. Word and phrase level recognition

The phoneme activations are fed to the word and syntax recognition part of the recognition system, which is based on a dynamic programming (DP) procedure to find the best path through a finite-state phoneme network, [[19]]). The network defines possible word sequences at the phoneme level. Optional pronunciations are realised as parallel branches. Inhalation sounds before the utterance and short silent intervals at word boundaries are included as optional branches in the net.

Phoneme duration information is used explicitly in the DP-algorithm to limit the search. Within the duration limits, uniform distribution densities are assumed. These limits are quite wide, and therefore probably don't influence the recognition result in a significant way.

However, the algorithm is designed for more extensive use of duration information in the future. The local distance in the DP-algorithm is the negative logarithm of the activation value for the phoneme being investigated.

For implementation reasons, the result of the DP-procedure is a phoneme string without any word information. The word sequence is determined by a second search procedure, which maps the phoneme string onto a string of words according to the syntax.

3.2. Phoneme based system

We have tested both phoneme and feature based recognition The networks were similar to the network described earlier with 10 ms input window and context nodes for the output and the hidden layer. The output phoneme activations are treated as probabilities and fed to the word and syntax recognition part of the recognition system.

3.3. Feature based systems

In our earlier paper [1] it was found that a net trained to recognise phonetic features for one speaker was more robust to a speaker change than a phoneme net, also compare [20]. Stevens [21, 22] has also argued for doing speech recognition directly from features instead of phonemes. The feature net we used had nine quasi-phonetic features; voicing, frication, vowel, nasalisation, front, central, back, high-low and rounding. As noted earlier the output activations of the nets are treated as probabilities, when using the cross entropy cost function. The outputs of the feature net are converted to phoneme activations by multiplying the output activation values of the features with each other according to the feature specification of each phoneme. If the feature should be ON the feature activation is taken directly from the net output and if the feature should be OFF the value of [1 - activation] is used instead. All the feature activations are multiplied together for each phoneme in each frame, to get the output activation of the phoneme.

The feature description used does not discriminate between all phonemes, e.g., the long and short [e]-vowel have the same feature setup. Since this was a cause of many recognition errors we tried another feature based system, where we introduced continuous target values. We simply treated the mean output activation for each feature over the training set as the target value during recognition. These values are different for phonemes having different spectral shapes and could be an easy way of handling the problem with the idealised on-off-features. In this case we used the value of [1- |feature activation - mean feature activation|] as the multiplying factor for each feature and phoneme.

3.4. Word recognition results

We have used 100 phonetically labelled three-digit sequences for seven male speakers in these tests. First we trained the net by the 10 first sequences of all speakers except the test speaker. All digit sequences of the test speaker were used for testing and the test speaker was rotated among all speakers (exclusive training). We also did tests where the 10 first sequences of the test speaker were included in the training set (inclusive training) and the rest of his material were used for testing. Exclusive training is of course more interesting in the context of speaker independent speech recognition. The mean word recognition rates for exclusive training were 90.0% for phonemes, 81.6% for features and 84.3 % for mean features and they were 95.5%, 92.1 and 91.1% respectively for inclusive training. This means that including a speaker in the training reduces the error rate by around 50%. Results for exclusive training are shown in Figure 6.

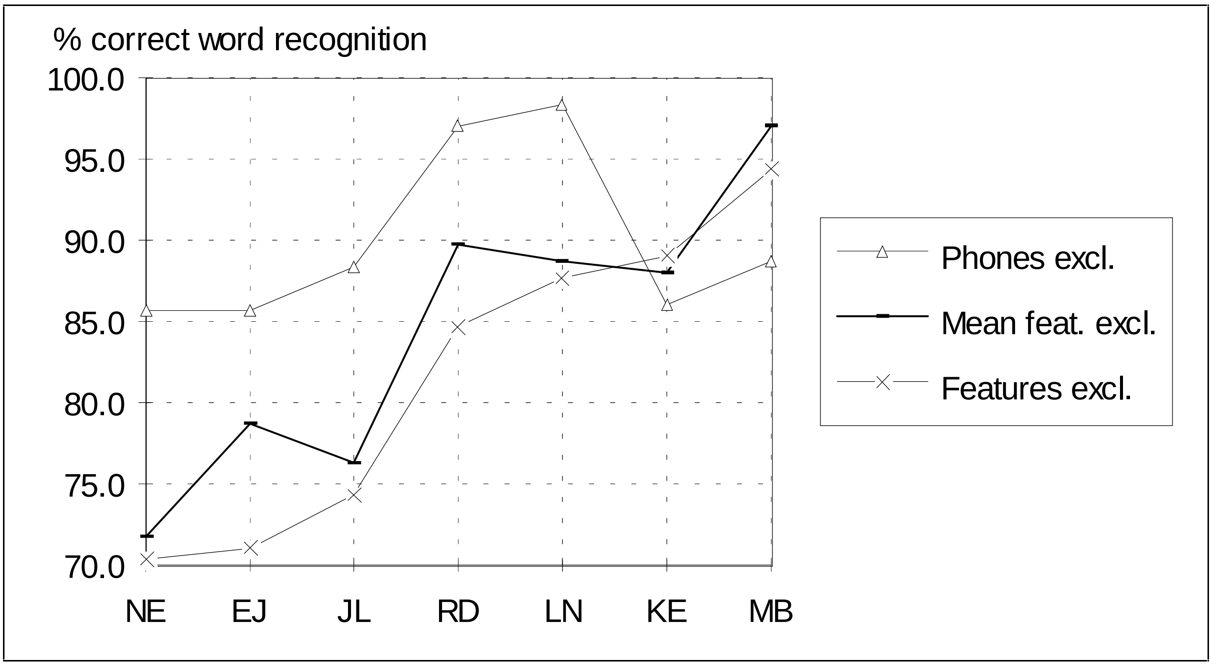

Fig. 6. *Recognition of digits in three-digit strings for seven male speakers. Network training was done on all speakers except the tested one, that was excluded.*

Exclusive training gives a large spread in performance, indicating that there is quite a large individual spread in the voices of the speakers. Phonemes perform better than features for all speakers but *MB* and *KE*, who get better results for features. For the other five speakers there is a rather close correlation between the results for phoneme and feature based training. The speakers *LN* and *RD* perform very well for phonemes, in fact as good as with inclusive training, indicating that their voices are close to the mean of the other speakers. The use of mean features performs somewhat better than "ideal" features for all speakers but *KE*. Most of the errors come from substitutions of 7 with 4, which both start with a noiseless fricative followed by a front vowel, that is acoustically similar. The overall result is that feature based recognition only performs better than phoneme based for two our of seven speakers.

4. DISCUSSION

Our results indicate that using features as an intermediate step for phoneme recognition does not seem to improve the performance. Including context will increase information about the dynamic effects of coarticulation and will always help in speech recognition. Adding simple recurrent nodes is more effective than enlarging the input spectral window. Recurrency will allow for keeping relevant short term information and will make it possible to use different integration times for different phonemes and net extracted features. The spectral information in a window has been shown to lie symmetrically around the window centre. The mean activation strength of the output phoneme nodes is strongly correlated to the recognition performance.

Word recognition based on phonemes performs better than feature based recognition for most speakers, at least for the articulatory based features we have used. It should be possible to design articulatory related feature sets that have better discriminating power. Using features based on spectral characteristics is another possibility. In this study features have been derived from phonemes and in the recognition phase they have been converted back to phonemes before the DP-matching. In doing this the features were treated as being statistically independent, which, of course, is an oversimplification. Another more straightforward approach would be to base the recognition entirely on features, which would give more flexibility in the (feature-) specification of lexical items, e.g., all features of a specific segment need not be aligned in time and context may change only some of the features [21, 22]. This could be the subject of further studies.

Acknowledgements

This project was supported by grants from The Swedish Language Technology Programme.

References

[1] Elenius, K. & Takács G. (1990): "Acoustic-Phonetic Recognition of Continuous Speech by Artificial Neural Networks", STL-QPSR No. 2-3, Dept. of Speech Comm., KTH, Stockholm, 1-44.

[2] Elenius, K., & Takács, G. (1991): "Phoneme recognition with an artificial neural network," in *Proceedings of Eurospeech '91*, Genova, Italy, 121-124.

[3] Nord, L. (1988):, "Acoustic-phonetic studies in a Swedish speech data bank," *Proceedings of SPEECH'88*, Book 3 (7th FASE Symp.), Inst. of Acoust., Edinburgh, 1147-1152.

[4] Rumelhart, D. & McClelland, J. (1986): *Parallel Distributed Processing, Vol. 1,* MIT Press Cambridge, MA, 318-362.

[5] Solla, S., Levin, E. & Fisher, M. (1988): "Accelerated Learning in Layered Neural Networks", Complex Systems 2, 625-640.

[6] van Ooyen, A. & Nienhuis, B. (1992): "Improving the convergence of the Back-Propagation Algorithm", Neural Networks, Vol. 5, 465-471.

[7] Hildebrandt, T. (1992): "Why Batch Learning is Slower than Per-sample Learning", CSEE Department, Lehigh University, Bethlehem, PA.

[8] Jacobson, R., Fant, G. & Halle, M. (1963): Preliminaries to Speech Analysis. *The Distinctive Features and Their Correlates*, MIT Press, Cambridge, MA.

[9] Bimbot, F., Chollet, G. & Tubach, J.P. (1991): "Automatic extraction of phonetic features in speech using neural networks", *Proceedings of the XIIth ICPhS, Aix-en-Provence,* Vol. 5, 394-397.

[10] Waibel, A., Hanazawa, T., Hinton, G., Shikano, K., & Lang, K. (1989): Phoneme recognition using time-delay neural networks," *IEEE ASSP* **37**:3, 626-631.

[11] Boulard, H. & Wellekens, C.J. (1989): "Speech pattern discrimination and multilayer perceptrons", Computer, Speech and Language Vol. 3, 1-19.

[12] Sejnowski, T.J. & Rosenberg, C.R. (1986): "NETtalk: A Parallel Network that Learns to Read Aloud", Technical Report Johns Hopkins University, EECS-86/01.

[13] Jordan, M. I., (1986): "Attractor dynamics and Parallelism in a Connectionist Sequential Machine", *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, Hillsdale, NJ, Erlbaum.

[14] Watrous, R. (1990): "Phoneme discrimination using connectionist networks", JASA, Vol. 87, No. 4, 1753-1772.

[15] Robinson, T.& Fallside, F. (1991): A recurrent error propagation network speech recognition system. *Computer Speech and Language*, 5(3), 259-274.

[16] Borell, J. & Ström, N. (1992): "A Comparison of Speech Signal Representation for Speech Recognition with Hidden Markov Models and Artificial Neural Networks", Proceedings from ESCA Workshop on Comparing Speech Signal Representations", Sheffield, England, 8-9 April.

[17] Elman (1988): "Finding structure in time", Technical Report 8801. Centre for Research in Language, University of California, San Diego.

[18] Servan-Schreiber, D., Cleeremans, A, and McClelland J. L. (1988): "Encoding sequential structure in simple recurrent networks", Carnegie Mellon University, CMU-CS-88-183.

[19] Blomberg, M. (1991):. "Adaptation to a speaker's voice in a speech recognition system based on synthetic references", *Speech Communication* 10, 453-461.

[20] Huckvale, M., Howard, I., Barry, W. (1989): "Automatic phonetic feature labelling of continuous speech", *Proc. Eurospeech 89, Paris, Vol. II*, 565-568.

[21] Stevens, K. N., (1988): "An approach to Lexical Access Based on Distinctive Features", *Proceedings of the Second Symposium on Advanced Man-Machine Interface Through Spoken Language*, Makaha, Oahu, Hawaii, Nov. 19 -22.

[22] Stevens, K.N., Manuel, S.Y., Shattuck-Hufnagel, S. & Liu, S. (1992): "Implementations of a model for lexical access based on features", *Proceedings of the International Conference on Spoken Language Processing*, Oct. 12-16, Banff, vol. 1, 499-502.