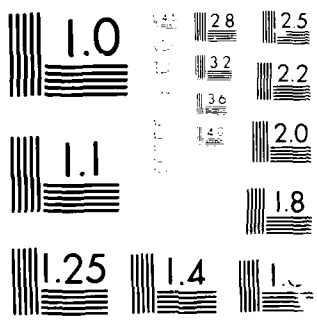
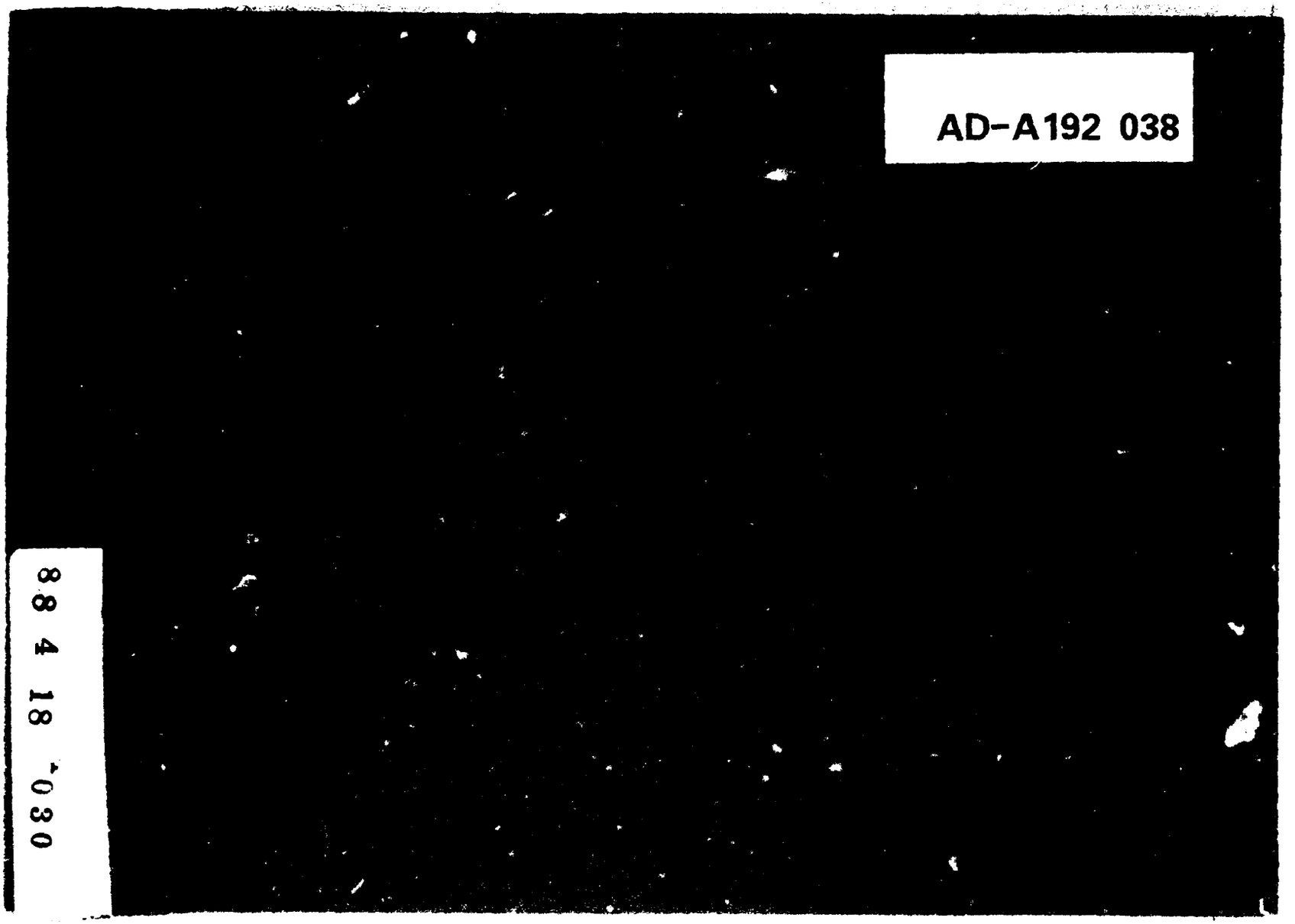
AD-A192 038 EXPERIMENTS IN ISOLATED DIGIT RECOGNITION USING THE MULTI-LAYER PERCEPTRO..(U) ROYAL SIGNALS AND RADAR ESTABLISHMENT MALVERN (ENGLAND) J W PEELING ET AL. 1/1

UNCLASSIFIED 17 DEC 87 RSRE-MEMO-4073 DRIC-BR-105252 F/G 25/4 NL

[redacted]


1.0
1.1
1.25
1.4
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.8



R.S.R.E. Memorandum 4073

EXPERIMENTS IN ISOLATED DIGIT RECOGNITION USING THE MULTI-LAYER PERCEPTRON

S.M. Peeling and R.K. Moore

December 17, 1987

Abstract

The *multi-layer perceptron* is investigated as a new approach to the automatic recognition of spoken isolated digits. The choice of the parameters for the multi-layer perceptron is discussed and experimental results are reported. A comparison is made with established techniques such as dynamic time-warping and hidden Markov modelling applied to the same data. The results, for this particular task, show that the recognition accuracy obtained using the multi-layer perceptron is comparable with that from using hidden Markov modelling.

Contents

List of Figures

List of Tables

1 Introduction

This report is concerned with the investigation of the *multi-layer perceptron* (MLP) [14] as a new approach to speech pattern processing and, in particular, to the problem of isolated digit recognition. The choice of the parameters used by the MLP are discussed. Experimental results are quoted to show how the choice of these parameters can influence the performance of the MLP.

A comparison is made with the established techniques of dynamic time-warping and hidden Markov modelling, applied to the same data.

2 The Multi-Layer Perceptron

Like the Boltzmann machine [1], the MLP is a member of the class of self-organising machines known as *adaptive parallel distributed processing networks* [11]. In this formalism, a-priori speech knowledge is expressed in the pattern of weighted connections in a network of very simple processing units. Input data is presented to the network as a pattern of activity on the input units, and the interpretation of the input data is represented by the resulting activity on the output units. The information embedded in the network is refined by adjusting the weights in order to produce the required input-output relationship. The advantage of the MLP over the Boltzmann machine is that it is more tractable computationally.

2.1 The Perceptron

As its name suggests, the MLP is related to work done in the 1960's on simple two-layer associative networks known as *perceptrons* [12]. In the perceptron, a set of input patterns is mapped directly to a set of output patterns and a learning algorithm is used to adjust the weights on the input-output connections in order to optimise the accuracy of the mapping. However, it was found that two-layer networks are not able to compute many important functions (for example, it is impossible to perform an exclusive-or operation without at least one intermediate or hidden unit) and there was no known learning algorithm for networks with hidden units. Such a learning algorithm has recently been reported; Rumelhart et al [14] have shown that it is possible to generalise the original perceptron learning algorithm to handle *multi-layer* feedforward networks.

2.2 Multi-Layer Networks

The units in a multi-layer perceptron are configured in layers such that there is a layer of input units, any number of intermediate layers, and a layer of output units. Connections within a layer or from higher to lower layers are not permitted. Each unit has a real-valued output (between 0 and 1) which is a non-linear function of its total input. For example, the total input, x_j, to unit j is given by:-

$$x_j = \sum_i y_i w_{ij}$$

where w_{ij} is the value of the weighted connection between unit i and unit j. The output of unit j, y_j, is given by:–

$$y_j = \frac{1}{1 + e^{-x_j}}$$

Thus, given an input pattern, the output pattern can be computed in a single *forward pass* through the network.

2.3 Adapting the Weights

If a unit j is an output unit then, for a given target value t_j, the total error E at the output is defined by the following expression:—

$$E = \frac{1}{2} \sum_c \sum_j (t_{jc} - y_{jc})^2$$

where c is an index over input-output pairs. The learning algorithm minimises E by gradient descent. This involves changing the weights according to the following rule:—

$$\triangle w_{ji}(n+1) = \epsilon \delta_j y_i + \alpha \triangle w_{ji}(n)$$

where $\triangle w_{ji}$ is the change to be made to the weight on the connection from the ith to the jth unit, ϵ is the learning rate, α is a 'momentum' term and δ is a measure of the local error at unit j. For an output unit, the error term is given by the expression:—

$$\delta_j = (t_j - y_j)\, y_j\, (1 - y_j)$$

and for an internal (hidden) unit the expression is:—

$$\delta_j = \sum_k \delta_k w_{kj}\, y_j\, (1 - y_j)$$

From the foregoing it can be seen that the learning algorithm changes the weights by apportioning the error at the output using a *backward pass* from the output layer to the input layer. This process is termed '*error back-propagation*'.

The effect of the learning algorithm is thus to 'discover' a set of weights which produce an appropriate *non-linear* transformation between input and output. The MLP is thus a powerful technique for deriving high-order internal representations.

2.4 Application To Speech Pattern Processing

The MLP has been applied to a range of problems in speech pattern processing. In this laboratory these include the recognition of single vowel spectra, whole-word patterns and visual speech information based on lip shape [13]. Elsewhere, MLP's have been used for:–

1. the analysis and recognition of speech [4]. Elman and Zipser use nine syllables, each made up of a consonant/vowel pair, and train MLPs to recognise the syllable, consonant or vowel. They also experiment with an identity mapping, i.e. the MLP output is the same as the input, and study the weights learnt by the system.

2. the recognition of spoken and handwritten digits [3]. Burr looks at ten examples of each of the digits 0 to 9, both handwritten and spoken. The effect of using different numbers of hidden units is discussed.

3. the recognition of the each letter of the alphabet spoken by 11 different speakers [5]. Franzini proposes methods of reducing the learning time by modifying the learning rate adaptively and using a different error measure.

4. isolated word recognition [8]. Lang looks at the four words "bee", "dee", "ee" and "vee" and uses an MLP to discriminate between them which does not have total connection between the layers.

5. illustrative classification problems and vowel data [7]. Huang and Lippmann are more concerned with discussing the capabilities of the MLP and some of its properties. They compare the performance from MLPs with that from quadratic Gaussian and k-nearest neighbour classifiers.

6. spoken digit classification [9]. Lippmann and Gold discuss the number of hidden units necessary for various classes of problem then go on to discuss the effect of more hidden units and layers on the digit classification problem.

7. the automatic derivation of orthographic-to-phonetic conversion rules for text-to-speech synthesis [17]. Sejnowski describes an MLP which learnt about the phonological rules associated with the pronunciation of English text.

The rest of this report is devoted to reporting the results obtained from applying MLPs with different parameters to the recognition of spoken isolated digits and comparing the results obtained with those obtained using more traditional methods.

3 Isolated Digit Recognition Experiments

3.1 Recorded Speech Data

All the data used was taken from the 'RSRE/40-speaker digit database' [18]. The recorded material consists of lists SB, 1A, 1B and 1C of the NATO RSG10 spoken digit database [19]. Each list consists of ten examples of each of the isolated digits "zero" to "nine". List SB was always one of the lists used for training purposes. One of the remaining three sets was used for testing purposes in all the experiments, although it was not the same list for all the speakers.

3.2 Data Representation

The data were obtained by passing speech signals through a 19 channel filter-bank analyser [6] with a 20ms frame rate. The output from each channel corresponded to the amplitude of the signal at a particular frequency. The data were segmented so that the start and end points of each digit were known, and each digit was also labelled according to its classification, i.e. "1", "2" etc. The words ranged in length between 17 and 60 frames.

3.3 Experiments

Two different types of recognition experiment were conducted. The first type was performed on data taken from each of the twenty 'least consistent' speakers in the forty speaker database [18]. Here, each MLP was trained to recognise the digits spoken by one of the speakers and tested on digits from the same speaker. This set of experiments is referred to as the *speaker dependent experiments*.

The second experiment used data from all forty speakers. Here, one, or more, examples of each digit were taken from all the speakers. The MLP was then trained to recognise the digit, irrespective of speaker. This set of experiments is referred to as *multiple speaker experiments*.

As well as the MLP, two "standard" speech recognition techniques were applied to the same data. They were dynamic time-warping (DTW), a template matching technique, and hidden Markov modelling (HMM), a statistical method. Results for DTW are only available for the speaker dependent experiments.

4 Software Implementation of the MLP

The MLP program was written in Coral66 and run on a VAX8600. The program was written to allow most of the parameters to be user changeable. For both sets of experiments the MLP was trained by looping through the training data. The digits were presented in groups of ten, i.e. each group contained one example of each of the digits, although the digits were not necessarily in order. After each set of ten numbers, the MLP weights were updated. Since the data set was of a finite length it was necessary to loop through it repeatedly until the termination criterion was satisfied.

The main output from the program consisted of a data file containing details of all the parameters involved in a particular run, plus the set of weights which had been generated. As the program ran it displayed the error per word summed over all the output units. Obviously it wasn't practical to print this error after each pattern presentation so it was only printed after some number of patterns had been presented to the system. Usually, the error was summed over ten or twenty sets of ten pattern presentations, then the average error displayed. In the remainder of this report, the term *presentation* refers to the cycle of passing 10 digits through the MLP and adjusting the weights on the connections using error back-propagation.

The format of the data file which was produced allowed for repeated training passes through the MLP using the weights which had been generated in the previous pass. It was thus easily possible to assess the effect of more training on the model (see Section 7.2).

5 Experimental Strategy

This section gives greater detail of the strategy used in all the experiments for each of the different techniques applied to the data.

5.1 Dynamic Time-warping

In the speaker dependent experiments, the DTW errors were computed, for each speaker, for a 300 digit test set (lists 1A, 1B and 1C) over 5000 randomly chosen reference sets from the 100 digit training set (list SB) [18]. The error quoted is the result of summing these errors over all 20 speakers.

Results from the DTW algorithm are only available for the speaker dependent experiments.

5.2 Hidden Markov Modelling

The HMM speaker dependent tests used an 8 state HMM with multivariate Gaussian states and diagonal covariance matrices, trained on 10 examples of each digit [16].

For the multiple speaker experiments a 16 state hidden semi-Markov model with Gaussian state output probability density functions and non-parametric (Ferguson) state duration probability distribution functions [15] was trained on one example of each digit from each of the 40 speakers. (It is assumed that better performance could be obtained by using more training data).

For both types of experiment, the same testing files were used by the HMM, HSMM and the MLP.

5.3 The Multi-Layer Perceptron

For all the experiments reported here each MLP had a 19 channel x 60 time frames array of inputs. Words shorter than 60 frames were padded with silence (zeros) and randomly positioned within the input array. (Hence when looping through the data, the digits were not always in the same position in the input array). There were 10 output units, i.e. one for each digit. The number of hidden units could be varied. In some cases there were no hidden units, i.e. the input and output units were directly connected as in the original *perceptron*. Other experiments involved either 1 or 2 hidden layers, each containing 8 or 50 hidden units. In this report, "1 x 8" refers to a single hidden layer containing 8 units and "2 x 8" refers to two hidden layers each containing 8 units.

The determination of parameters such as learning rate, momentum scaling term and number of hidden units, will now be discussed.

The problem of choosing a set of suitable parameters for any experiment is non-trivial since the parameters are dependent on the problem and the MLP configuration. For example, values of ϵ the learning rate, and α the momentum term, which are suitable for an MLP with one hidden layer may not be suitable for a system with two hidden layers. Hence, suitable values for ϵ and α can only be found by experimentation for each configuration. Given the size of the data set involved in this study, a comprehensive search for the optimum set of parameters was not feasible. Details are given in Appendix A of the experiments that were conducted in order to find suitable values of ϵ and α for an MLP with one layer of eight hidden units.

Similarly, in order to determine the appropriate number of hidden units it is impossible to conduct an exhaustive search. Instead it is necessary to rely on experience and previous published work; although "good" values for one experiment are not necessarily so for another. Some initial speaker dependent experiments were performed on a small subset of the data using varying numbers of hidden units. From these, it appeared that less than eight hidden units gave very poor results. There was no significant improvement obtained from using twelve rather than eight hidden units.

In the initial experiments it was believed that the values of the *start-up weights* was crucial to the successful convergence of the MLP. (These start-up weights are the small random values which are assigned to the weights on all the connections before the first pass through the network). A strategy was therefore evolved in which each experiment was repeated five times with different start-up weights. It soon became clear that this was not necessary, provided that the convergence criteria, discussed below, were satisfied.

There are two methods of terminating the training phase for an MLP: the MLP is either presented with a specified number of training examples, or the training continues until the total error E at the output units falls below some pre-defined value. Both strategies were employed but with a further limitation - the MLP so trained must give zero errors when tested on the training data. This criterion was relaxed in some of the experiments when after using five different sets of start-up weights the MLP still did not give zero errors on the training data. Also, due to the computational load of the multiple speaker problem, only two different sets of start-up weights were used in those experiments.

Rumelhart et al [14] state that setting the target outputs to be 0 or 1 forces the weights on the connections to become infinitely large. This is not a problem which has materialised in these experiments. The target outputs were always set to 0 or 1.

6 Computational Requirements

A comparison of the computational requirements for the MLP, 8-state HMM and 16-state HSMM are shown in Table 1. The program times come from computer programs written in Coral66 and run on a VAX8600.

Technique	No. of parameters estimated	Training (mfpos)	Training Program time (hours)	Recognition (mfpos per word)
MLP	9000 - 60000	5800 - 46200	8 - 50	9000 - 60000
8-state HMM	3500	94	0.25	150000
16-state HSMM	9500	2400	8	290000

Table 1: Comparison of the computational requirements of MLP, 8-state HMM and 16-state HSMM. Operations counted in million floating point operations (*mfpos*).

From a computational point of view more parameters need to be estimated for the MLP than for the HMM and HSMM. Similarly, more computation is involved in the training phase for the MLP. However, the situation is reversed in the testing, or recognition, phase where the MLP typically involves at least 50% fewer operations than the HMM or HSMM.

7 Speaker Dependent Results

7.1 Learning Rate and Momentum Term

Initial speaker dependent experiments were conducted to investigate the effect of different values of the learning rate, ϵ, and momentum scaling term, α. Each experiment used an MLP with a single layer of eight hidden units. For each of the twenty least consistent speakers five different sets of start-up weights were used and nine different pairs of ϵ and α values. These pairs of values were 0.25, 0.5 and 0.75 in all possible combinations for ϵ and α. Each MLP was trained on 1000 pattern presentations, i.e. it was shown examples of each of the 10 digits 1000 times. The weights were updated after each set of 10 digits. The MLP was then used to recognise the training and test data sets. Detailed results for all the experiments are shown in the graphs in Appendix A. A summary of the results is shown in Tables 2 and 3. These are the "best" results for each pair of ϵ and α in the sense that they come from the MLPs which gave the smallest number of errors on the training set. (When more than one MLP gave the same number of errors the results quoted come from the one with the smallest error per pattern after 1000 presentations).

ϵ	α		
	0.25	0.50	0.75
0.25	5	2	8
0.50	4	5	189
0.75	5	28	553

Table 2: Total speaker dependent errors from a 2000 digit training set for MLPs with 1 x 8 hidden units and ϵ and α as shown.

ϵ	α		
	0.25	0.50	0.75
0.25	97	65	91
0.5	82	76	274
0.75	78	113	634

Table 3: Total speaker dependent errors from a 2000 digit test set for MLPs with 1 x 8 hidden units and ϵ and α as shown.

From Tables 2 and 3 it is clear that the best training and test results for these experiments came from using ϵ=0.25 with α=0.5. For all the other ϵ/α combinations in Table 2 which lie on or above the diagonal, there were less than 10 errors on the training data. The corresponding test set results in Table 3 are also similar to each other. Obviously for those cases below the diagonal, where the errors on the training set are in the twenties or even hundreds, then the test results are much less meaningful.

Other work in this laboratory [2] has reported good results from using a large momentum term and a small learning rate. For the speaker dependent experiments, some initial tests were conducted which showed that good results were obtained using a learning rate of 0.05 and a momentum scaling term of 0.9. However it was discovered that more presentations

were necessary to train the MLP; 1500, rather than 1000, were used. After 1500 presentations the final error per pattern was very similar to that which had been obtained in the earlier experiments after 1000 presentations. Again, five experiments were performed and these graphs can also be found in Appendix A. The results proved to be very similar to those obtained from the combination ϵ=0.25, α=0.5. A summary of the "best" results, from 1000 and 1500 pattern presentations, is shown in Table 4.

ϵ	α	No. of Presentations	Training Errors	Testing Errors
0.25	0.50	1000	2	65
0.05	0.90	1500	1	71

Table 4: Total speaker dependent errors from 2000 digit training and test sets for MLPs with 1 x 8 hidden units and ϵ and α as shown.

In view of the results shown in Appendix A, the strategy of using five different sets of start-up weights per experiment was considered unnecessary. All further experiments used just one set of start-up weights unless the MLP failed to converge, i.e. it did not give zero errors on the training set, when another set were used. An overall limit of 5 different sets of start-up weights, per experiment, was applied. The MLP's usually converged with the first run - difficulties mainly arose in systems without hidden units. When any difficulties did arise, it was usually with the same few speakers and up to 5 different sets of start-up weights could be needed, although on average it was about 3.

7.2 Termination Criteria

As mentioned previously, the training phase can be terminated either after a pre-determined number of pattern presentations, or when the error per pattern falls below some pre-specified value. From the results shown in Table 4, it can be seen that even the "best" results after a fixed number of pattern presentations are still returning a large number of errors on the test data. In view of this, the alternative termination condition was tried. Experiments were conducted where an MLP with a single layer of eight hidden units remained in the training phase until the error per pattern was less than 0.001 when averaged over 200 patterns.

Figure 1 shows the behaviour of the error per pattern at the output units for one of the speakers in the database. This graph is typical of those obtained from other speakers - the main difference being the number of presentations before the error falls below the 0.001 level. It is worth noting that the error after 1500 presentations happens to be in a local minimum but, in general, the error is still oscillating quite dramatically. However, after 4000 presentations, although there is still oscillation the error is much smaller. Also, although the error is still decreasing it is doing so at a much slower rate.

Two pairs of learning rate and momentum term values were used and the results are shown in Table 5. The results for individual speakers are shown in Appendix B.

From the results in Table 5 it is clear that a significant improvement in performance is

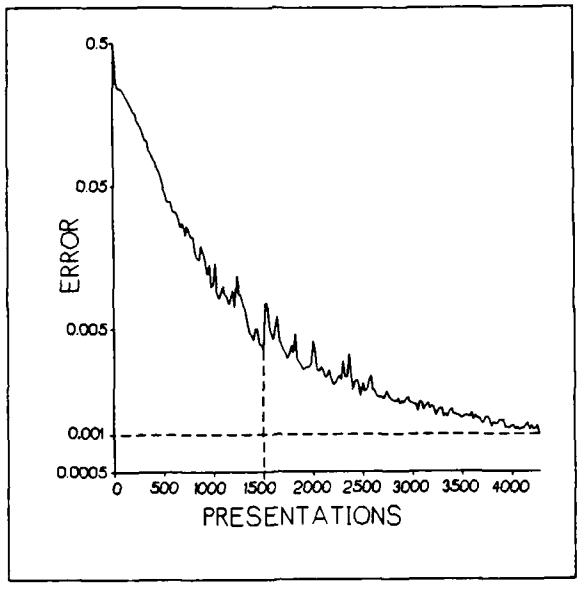


Figure 1: Graph showing typical behaviour of the error per pattern at the output units for a speaker dependent experiment.

obtained by using more pattern presentations during the training phase. It is worth noting that whilst the average number of pattern presentations was 4276 the range covered was from 2900 to 5920. Clearly, in order to obtain the best performance, without unnecessary training, it is advisable to use the error per pattern as a termination condition.

All the remaining experiments mentioned in this report used the error per pattern as a termination criterion, rather than the number of pattern presentations.

ϵ	α	Average no. of Presentations	Training Errors	Testing Errors
0.25	0.50	1500	2	65
0.05	0.90	1500	1	71
0.25	0.50	4276	0	48
0.05	0.90	4967	0	47

Table 5: Effect of using different termination conditions for MLPs with 1 x 8 hidden units – total speaker dependent errors from 2000 digit training and test sets

7.3 Number of Hidden Units

There is no easy way to decide how may hidden units are necessary to solve a specific problem. However, it is known that two hidden layers are sufficient [10].

No. of Hidden	ϵ	α	Training Errors	Testing Errors
0	0.07	0.50	0	36
1 x 8	0.05	0.90	0	47
2 x 8	0.05	0.90	0	73
1 x 50	0.15	0.75	0	29
2 x 50	0.25	0.50	0	47

Table 6: Speaker dependent errors from a 2000 digit test set with varying numbers of hidden units.

In this study, initial experiments concentrated on using just a single layer of eight hidden units. In order to try and assess the performance obtained, speaker dependent experiments were conducted which used zero hidden units, i.e. the input and output layers were directly connected. These results (shown in Table 6) were better than the best obtained from a single layer of 8 hidden units. (This may be because there are more weights involved in the system without hidden units - there are 10 rather than 8 units connected to the 19 x 60 input array). This prompted a set of experiments using 2 x 8 hidden units. The results (also shown in Table 6) again showed a worsening performance. The most likely explanation was that there were too few weights in the system for it to perform the task.

The numbers of weights involved in MLPs with varying number of hidden units are shown in Table 7. From this it can be seen that with 50 hidden units there are significantly more weights in the system. In order to test the hypothesis that more weights would improve the performance, MLPs with 50 hidden units were used. From the results in Table 6 it can be seen that a single layer of 50 hidden units gave fewer errors than zero hidden units. The 2 x 50 results do not shown any improvement over the 1 x 50 because the large number of weights in the former tend to become specialised for the training set. They are then less able to generalise to the test set.

No. of Hidden	No. of Weights
0	11400
1 x 8	9200
2 x 8	9300
1 x 50	57500
2 x 50	60000

Table 7: Number of weights in MLPs with the numbers of hidden units shown

7.4 Number of Training Examples

Experiments with the multiple speaker data (reported later) showed that the more training data which the MLP was shown, the better the results were. As a result of this, experiments were conducted into the effect of increasing the amount of training data in the speaker dependent case. As previously mentioned, the database contained four sets of ten examples of each digit for all the speakers. Initial experiments used one set of 100 words for training (always list SB) and a different set of 100 for testing purposes. Later experiments kept the same testing set but used all of the remaining 300 words for training purposes. Table 8 shows a comparison of the errors obtained from MLPs trained using 10 or 30 examples of each digit, for varying numbers of hidden units. The corresponding results for the individual speakers are in Appendix C.

No. of Hidden	ϵ	α	10 Examples		30 Examples	
			Train	Test	Train	Test
0	0.07	0.50	0	36	2	7
1 x 8	0.05	0.90	0	47	1	5
2 x 8	0.05	0.90	0	73	5	19
1 x 50	0.15	0.75	0	29	1	5
2 x 50	0.25	0.50	0	47	1	9

Table 8: Speaker dependent experiment: effect of using 10 and 30 examples of each digit during training. Errors shown from 2000 and 6000 digits respectively.

The results demonstrate a significant improvement in test set recognition performance by using more training data.

7.5 Overall Speaker Dependent Results

Figure 2 shows a comparison of the total errors from speaker dependent isolated word recognition experiments on 2000 digits spoken by 20 speakers. The MLP results quoted are those which correspond to the fewest errors on the training data. The best performance was obtained from an HMM, but both the single hidden layer MLP's gave almost identical results. The worst performance was obtained from DTW.

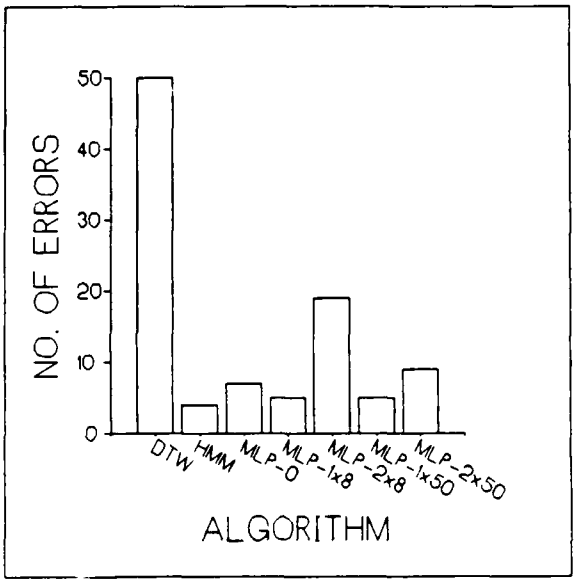


Figure 2: Results of Speaker Dependent Experiments (errors out of 2000)

8 Multiple Speaker Results

To a large extent the multiple speaker experiments were based on the experience gained from the speaker dependent experiments. Some initial experiments, using one example of each digit from all forty speakers in the database, were performed on pairs of ϵ/α values. From these, three pairs were chosen for the full set of experiments. They were ϵ=0.15, α=0.75 and ϵ=0.07, α=0.5 and ϵ=0.25, α=0.5. The results of using these three pairs of values for varying numbers of hidden units are shown in Table 9. Note that the training errors are from 400 digits and the testing errors from 4000 digits.

The results shown in Table 9 were somewhat disappointing since an 8 state HMM was producing about 60 errors on the test set. However, by looking at the number of pattern presentations involved in the training phase (see Table 10) it was clear that the MLP with 2 x 50 hidden units was, in general, learning much faster than the one with 1 x 50. Since the MLPs with 1 or 2 layers of 50 hidden units were also giving very few errors on the training data it was clear that they were specialised for the training set.

In an effort to improve the MLP's performance, it was shown more training words - two examples of each digit per speaker. In the cases of 0 and 1 x 8 hidden units only the ϵ/α pair which had given fewest errors on the 400 digit training set were used. The results are shown in Table 11.

Although the results from the training set, shown in Table 11 are worse than those from the 400 digit training set, the test set results are better. From the number of presentations involved in the training phase, shown in Table 12, it is clear that the learning phase has

No. of Hidden	ϵ=0.15 α=0.75		ϵ=0.07 α=0.50		ϵ=0.25 α=0.50	
	Train	Test	Train	Test	Train	Test
0	72	903	29	477	79	887
1 x 8	19	493	8	389	25	549
2 x 8	37	635	26	552	37	637
1 x 50	0	151	0	124	0	134
2 x 50	1	206	1	179	0	159

Table 9: Multiple speaker results: errors from a 400 digit training set and 4000 digit test set

No. of Hidden	ϵ=0.15 α=0.75 No. of Presentations	ϵ=0.07 α=0.50 No. of Presentations	ϵ=0.25 α=0.50 No. of Presentations
0	30000	30000	30000
1 x 8	30000	44180	30000
2 x 8	30000	30000	30000
1 x 50	10400	30960	10680
2 x 50	12600	12600	9520

Table 10: Number of pattern presentations in the training phase of multiple speaker experiments using a 400 digit training set.

taken longer for the 800 word training set than for the 400 one.

In view of this improvement in test set performance, the training set was increased to 1200 words, i.e. 3 examples per digit. This was only tried for the two 50 hidden unit cases since their performance was far better than any of the others. The results are shown in Table 13.

Again, the number of pattern presentations has increased, but the training and test sets recognition performance has decreased for the 2 x 50 MLP.

The overall performance of MLP's in the multiple speaker experiments are shown in Figure 3. This shows the recognition performance obtained from a 16 state HSMM in comparison with that from the MLPs with the fewest errors on the training data.

The best performance came from the HSMM and the best MLP performance from an MLP with a single layer of 50 hidden units. The worst performance came from an MLP with no hidden layer.

No. of Hidden	ϵ=0.15 α=0.75		ϵ=0.07 α=0.50		ϵ=0.25 α=0.50	
	Train	Test	Train	Test	Train	Test
0	-	-	60	444	-	-
1 x 8	-	-	29	397	-	-
2 x 8	99	633	60	453	75	515
1 x 50	1	99	0	85	4	107
2 x 50	17	192	1	131	5	117

Table 11: *Multiple speaker results: errors from an 800 digit training set and 4000 digit test set*

No. of Hidden	ϵ=0.15 α=0.75 No. of Presentations	ϵ=0.07 α=0.50 No. of Presentations	ϵ=0.25 α=0.50 No. of Presentations
0	-	30000	-
1 x 8	-	30000	-
2 x 8	30000	30000	30000
1 x 50	15760	38000	15960
2 x 50	14140	18240	14440

Table 12: Number of pattern presentations in the training phase of multiple speaker experiments using an 800 digit training set.

9 Discussion

This section attempts to relate the results obtained in this study to results quoted elsewhere for using MLPs on speech related tasks.

In [8], Lang has used an MLP with a spectrogram input to distinguish between the words "bee", "dee", "ee" and "vee", referred to as B, D, E and V. He started with spectrograms containing 128 frequency bands but combined them to reduce to 16 bands. The initial data contained 50 time frames but adjacent frames were summed to reduce to 12 time slices. As in this study he tried using zero hidden units to establish a baseline performance then went on to use a single layer of 8 hidden units. However, in the 1 x 8 experiments, there was not total connection between the layers in that the hidden units did not see all the inputs at any one time, similarly the output units did not see all the hidden units at any one time. A data set of 700 words was used for training purposes and 100 for test purposes. Lang quotes 20000 iterations for training but it is not clear what he counts as an iteration - one word or one example of each class of word. After this training he quotes 93% correct on training data and 93% on test data. These results are certainly worse than the best quoted here - for example this study can give 99% correct discrimination over 10 classes after 4500 iterations, where each iteration includes 10 words. However, it may be argued that B, D, E and V are more confusable than the digits.

No. of Hidden	ϵ	α	No. of Presentations	Training Errors	Testing Errors
1 x 50	0.07	0.50	39640	2	78
2 x 50	0.25	0.50	19500	7	142

Table 13: Multiple speaker results: errors from a 1200 digit training set and a 4000 digit test set

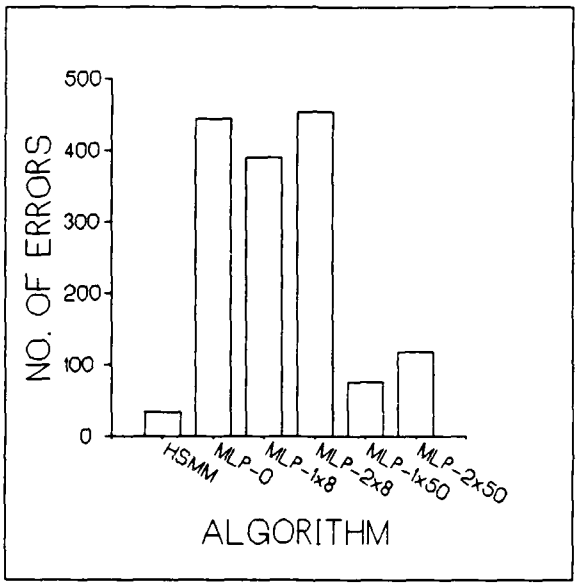


Figure 3: Results of Multiple Speaker Experiments (errors out of 4000)

One possible explanation of Lang's performance is the argument he quotes for deciding how to downsample his input spectrograms. He has 700 training cases, each of which requires an output choice which can be specified with 2 bits. He knows from experience that each weight in a network can fairly easily learn one and a half bits. Hence, he argues that a network with more than 1000 weights could memorise the training cases and fail to generalise to the test set. Applying this argument to the speaker dependent experiments with 300 training words and 10 output classes implies less than 800 weights are needed. Similarly for the multiple speaker with 400 training words less than 1100 weights would be needed. The number of weights, including biases, which are used are shown in Table 7.

From this it can be seen that according to Lang's argument digit discrimination should be possible without any hidden units. The results have shown that this is not true and that in fact an MLP with 1 x 50 hidden units performs better than one with 1 x 8. It is interesting to speculate whether with fewer input units Lang's arguments might hold for the digit discrimination task.

In [9] Lippmann and Gold use MLPs to distinguish between the first seven monosyllabic digits, i.e. 1, 2, 3, 4, 5, 6 and 8. They use two sets of eleven cepstral coefficients for each word, i.e. 22 inputs, and experiment with varying numbers of hidden units. They performed speaker dependent experiments on 16 different speakers using 70 training and 112 testing words per speaker. They reported the best results for using two hidden layers with an error rate of over 7% averaged over all the speakers. This again is worse performance than quoted in this report. Also, they state that the more hidden units which are used, the more iterations are necessary in the training phase. This has certainly not been true in the experiments reported here, although Lippmann and Gold do start with 16 hidden units and increase to 256, which is many more than have been used here. However they report that the increase is noticeable from the start.

10 Conclusions

It is clear that in both speaker dependent and multiple speaker recognition of isolated spoken digits the MLP is capable of a level of performance comparable to HMM.

The results from MLP's with two hidden layers tend to be worse than those from a single hidden layer. This is because there are more weights to be learnt in the two hidden layer case and they tend to be optimised for the particular training set. Using a larger training set decreases the number of errors.

Experience suggests that the choice of learning rate and momentum term are crucial, as to a lesser extent is the number of hidden units.

The choice of start-up weights was not found to be very important. However, in cases where the MLP failed to converge during the training phase, the experiment was always repeated with a different set of weights.

It is very encouraging that the MLPs have proved capable of performance so close to that achieved by HMMs. It is also worth stressing that a large amount of a-priori knowledge goes into an HMM, whereas none is necessary for the MLP. If a-priori knowledge was included in MLPs, in some way, the performance could no doubt be dramatically increased.

It is also worth noting that the performance from MLPs reported here is at least as good as that obtained in other laboratories when using MLPs on similar tasks.

References

[1] D.H. Ackley, G.E. Hinton, and T.J. Sejnowski. A Learning Algorithm for Boltzmann Machines. *Cognitive Science*, 9:147–169, 1985.

[2] M.D. Bedworth and J.S. Bridle. *Experiments with the Back-Propagation Algorithm: A Systematic Look at a Small Problem.* Memo 4049, Royal Signals and Radar Establishment, 1987.

[3] D.J. Burr. A Neural Network Digit Recogniser. *Proc. IEEE International Conf. on Systems, Man and Cybernetics*, 1986. Atlanta, GA, October 14–17.

[4] J.L. Elman and D. Zipser. *Learning the Hidden Structure of Speech.* ICS Report 8701, Institute for Cognitive Science, University of California, 1987.

[5] M.A. Franzini. Speech Recognition With back Propagation. *Proc. IEEE Ninth International Conf. of Engineering in Medicine and Biology Society*, November 1987.

[6] J.N. Holmes. The JSRU Channel Vocoder. *Proc. IEE*, 127 Pt.F(1):53–60, 1980.

[7] W.Y. Huang and R.P. Lippmann. Comparisons Between Neural Net and Conventional Classifiers. *Proc. IEEE First International Conf. on Neural Networks*, 1987. San Diego, CA, June 21–24.

[8] K. Lang. *Connectionist Speech Recognition.* Technical Report, Computer Science Dept., Carnegie-Mellon University, June 1987. Thesis Proposal.

[9] R.P. Lippmann and Gold B. Neural-Net Classifiers Useful for Speech Recognition. *Proc. IEEE First International Conf. on Neural Networks*, 1987. San Diego, CA, June 21–24.

[10] I.D. Longstaff and J.F. Cross. *A Pattern Recognition Approach to Understanding the Multi-layer Perceptron.* Memo 3936, Royal Signals and Radar Establishment, 1986.

[11] J.L. McClelland, D.E. Rumelhart, and the PDP Research Group. *Parallel Distributed Processing.* MIT Press, 1986.

[12] M.L. Minsky and S. Papert. *Perceptrons.* MIT Press, 1969.

[13] S.M. Peeling, R.K. Moore, and M.J. Tomlinson. The Multi-layer Perceptron as a Tool for Speech Pattern Processing Research. *Proc. IoA Autumn Conf. on Speech and Hearing*, 8(7):307–314, 1986.

[14] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. Learning Internal Representations by Error Propagation. In D.E. Rumelhart and J.L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol. 1: Foundations*, MIT Press, 1986.

[15] M.J Russell and A.E. Cook. Experimental Evaluation of Duration Modelling Techniques for Automatic Speech Recognition. *Proc. IEEE International Conf. on Acoustics, Speech and Signal Processing*, :2376–2379, 1987.

[16] M.J. Russell and A.E. Cook. Experiments in Speaker-Dependent Isolated Digit Recognition using Hidden Markov Models. *Proc. IoA Autumn Conf. on Speech and Hearing*, 8(7):291–298, 1986.

[17] T.J. Sejnowski and C.R. Rosenberg. Parallel Networks That Learn to Pronounce English Text. *Complex Systems*, 1:145–168, 1987.

[18] D.C. Smith, M.J. Russell, and M.J. Tomlinson. *Rank-ordering of Subjects Involved in the Evaluation of Automatic Speech Recognisers.* Memo 3926, Royal Signals and Radar Establishment, 1986.

[19] R.S. Vonusa, J.T. Nelson, S.E. Smith, and J.G. Parker. Nato ac/243 (Panel iii rsg10) Language Database. *Proc. NBS Speech I/O Stand. Workshop*, 1982.

Appendix A Effect of Varying Learning Rate and Momentum

The figures here show the errors obtained for the 20 least consistent speakers in the database [18] using an MLP with a single layer of 8 hidden units. In each case the MLP saw each of the ten digits 1000 times (for the first nine graphs) and 1500 times (the last graph). The digits were randomly positioned within the input array. There were five runs per speaker for each experiment and the errors for both test and training data are shown. Each speaker is identified by his, or her, initials.

The first nine graphs show the results from using values of 0.25, 0.5 and 0.75 for both the learning rate, ϵ, and the momentum term, α. These graphs are in order of worsening recognition performance, taken over all the speakers. The last figure shows the results from using ϵ=0.05 and α=0.9.

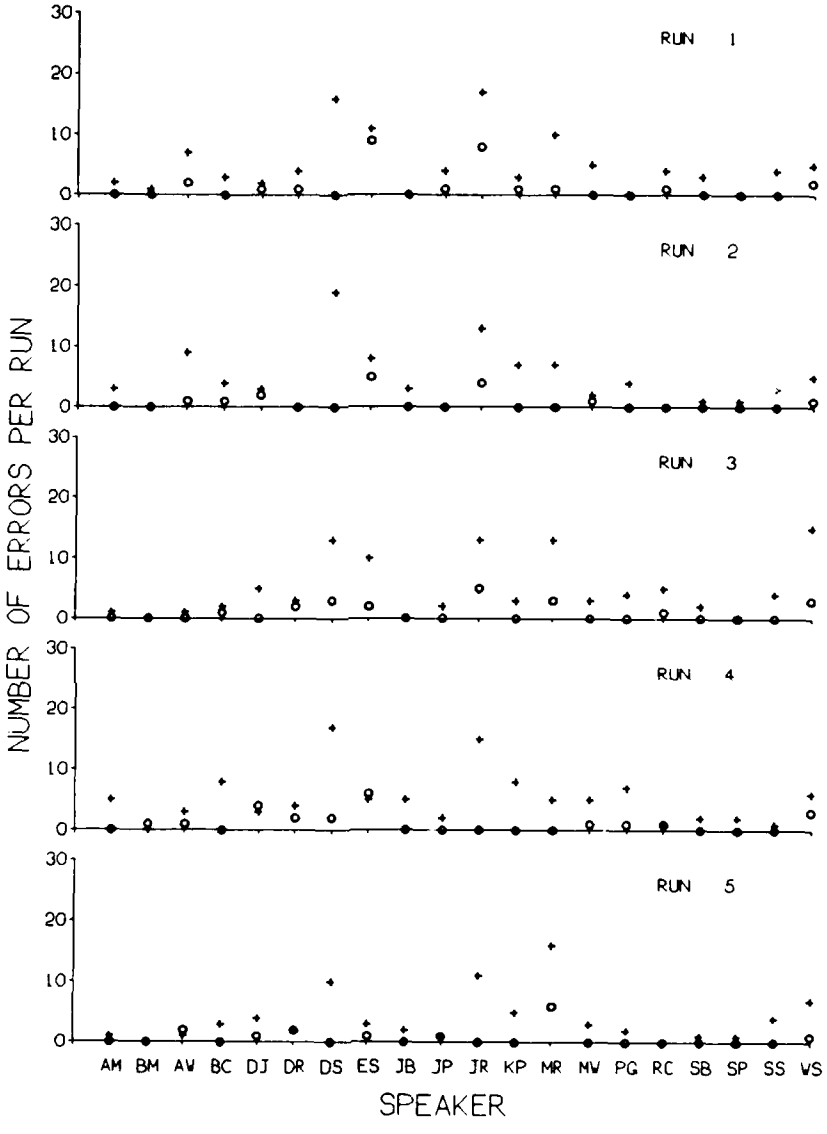


Figure 4: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.25 and α=0.5. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

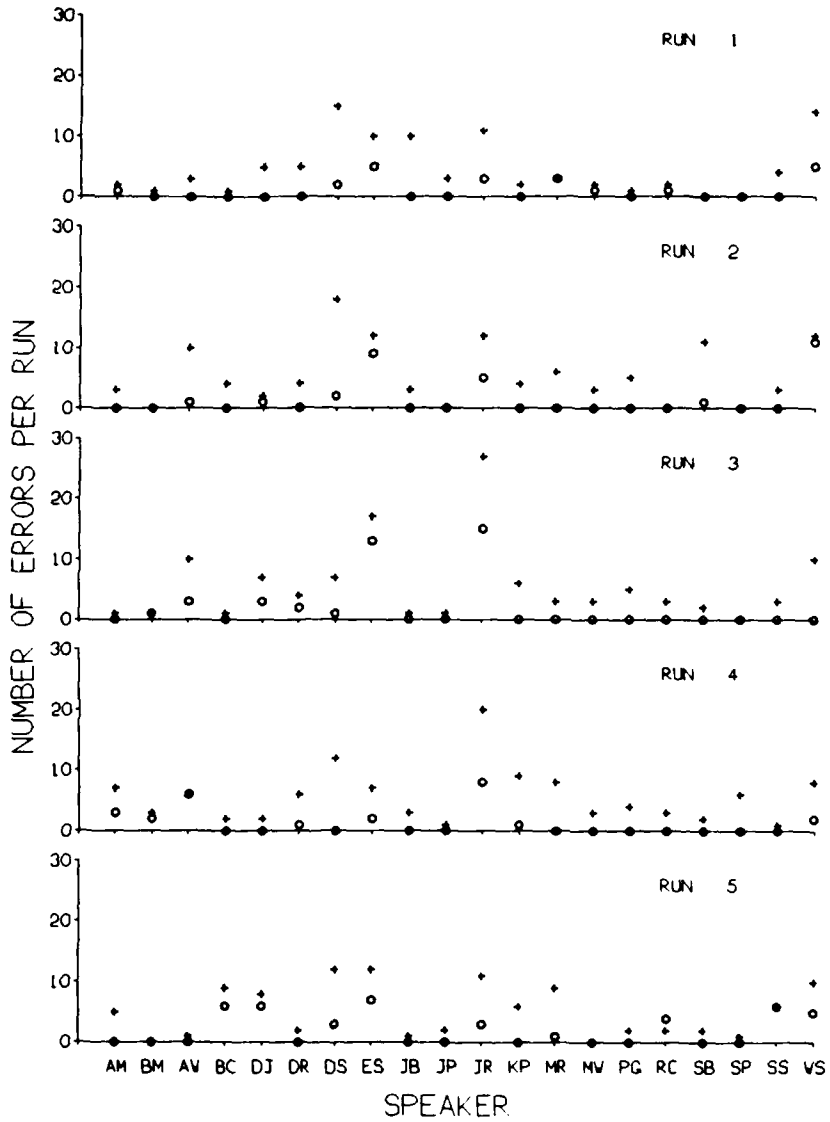


Figure 5: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.5 and α=0.5. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

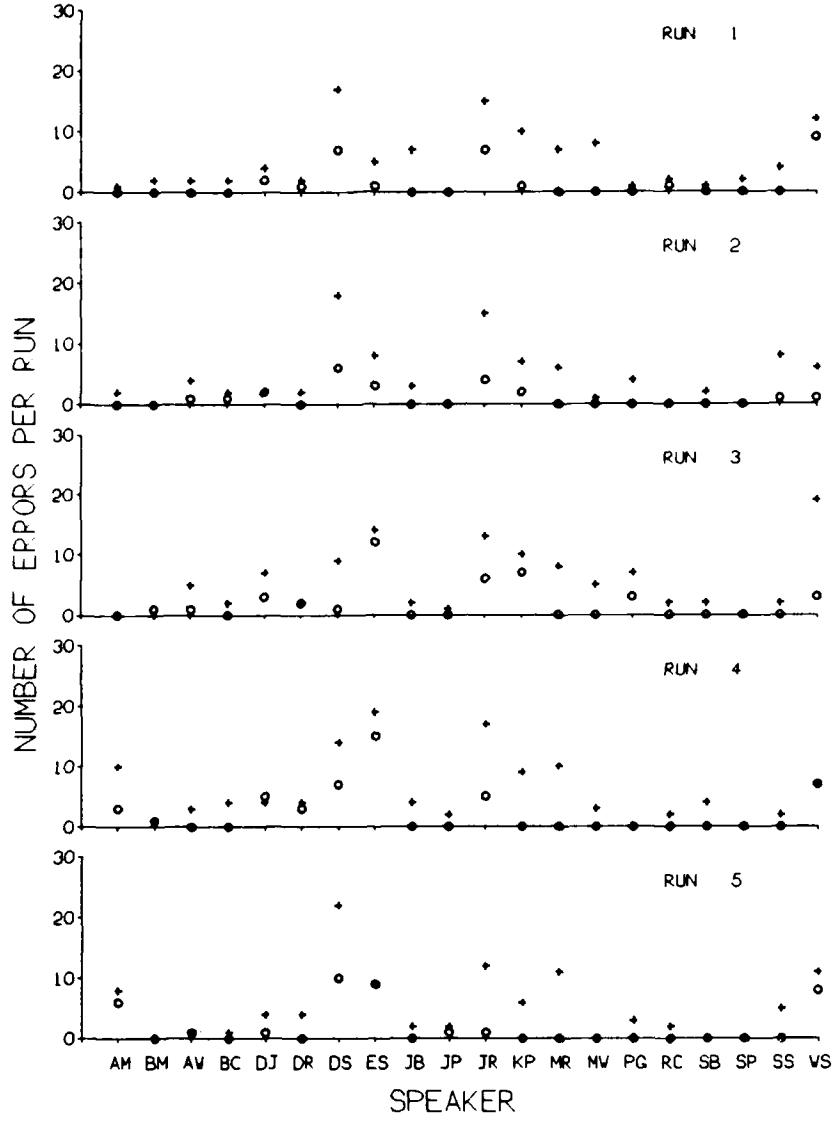


Figure 6: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.75 and α=0.25. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

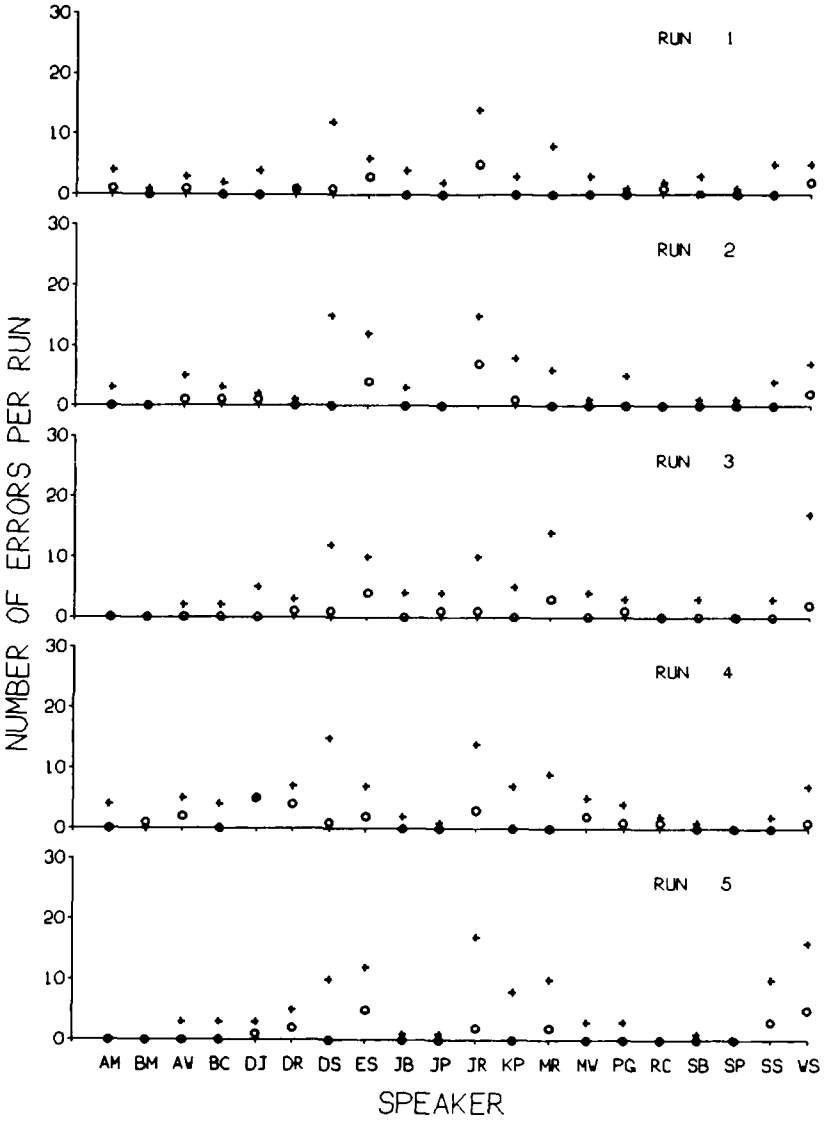


Figure 7: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.5 and α=0.25. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

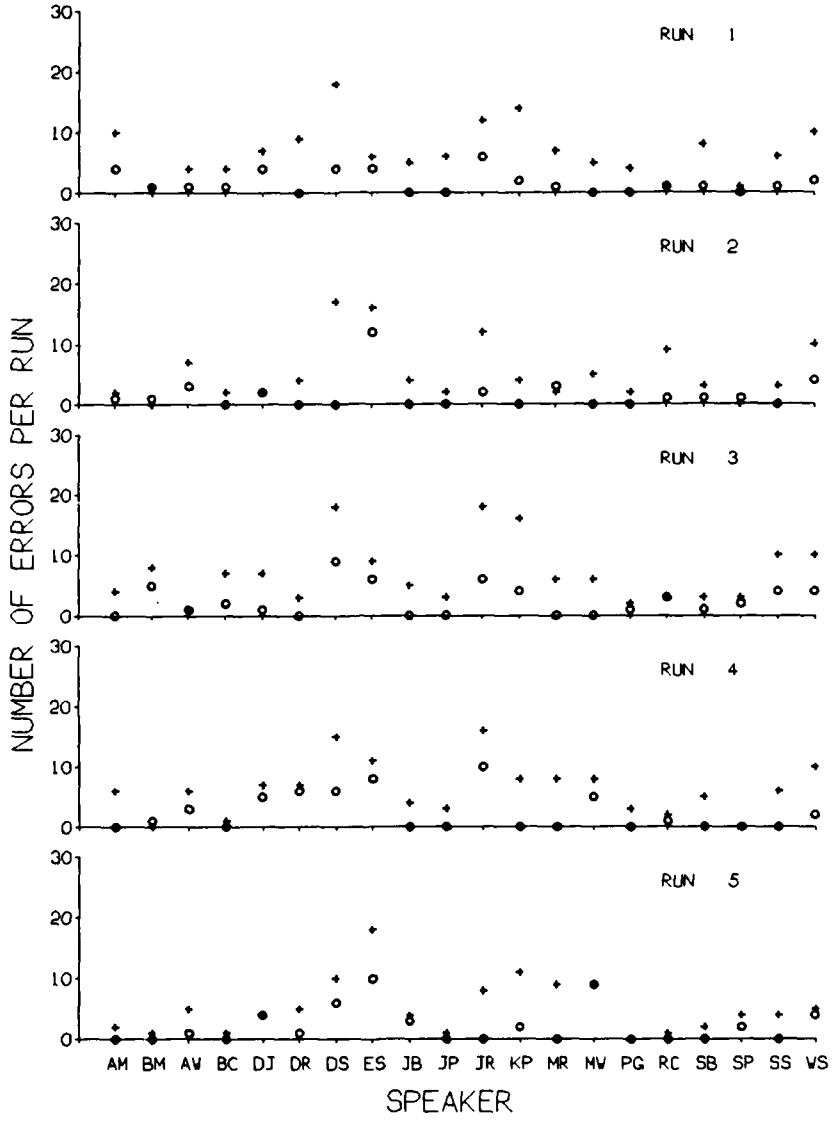


Figure 8: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.25 and α=0.75. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

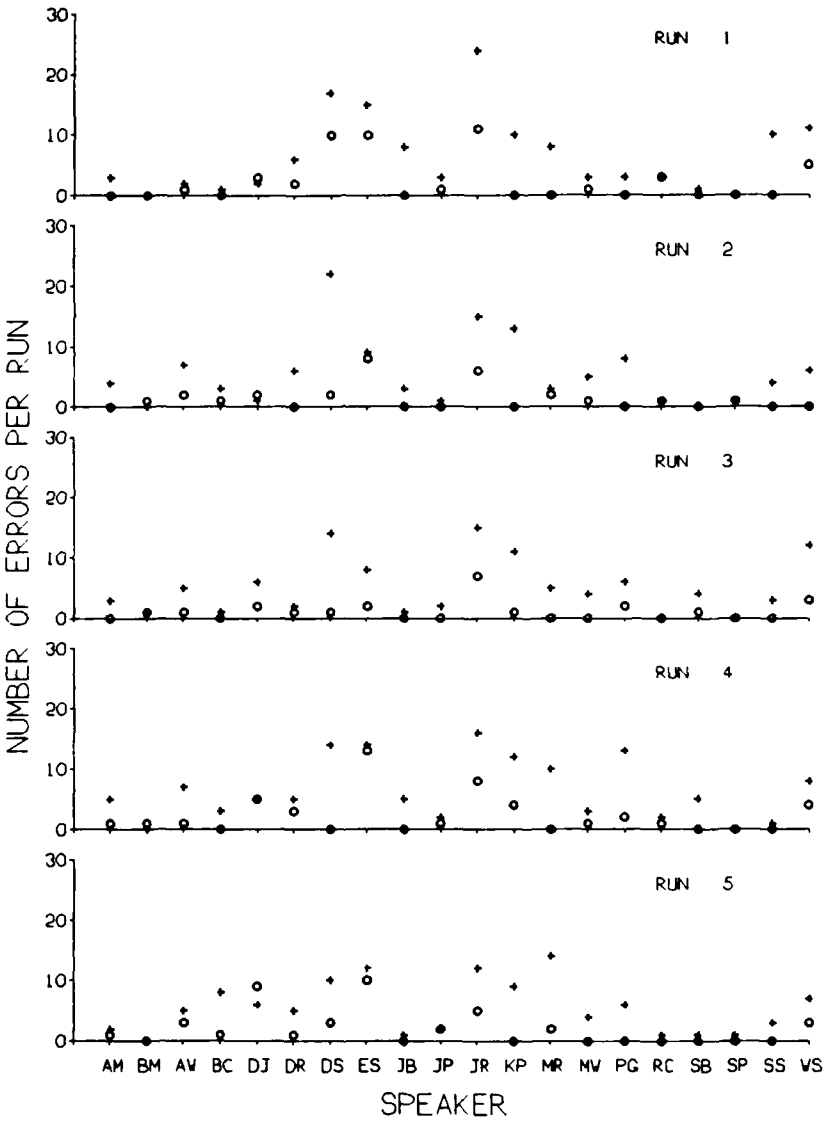


Figure 9: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.25 and α=0.25. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

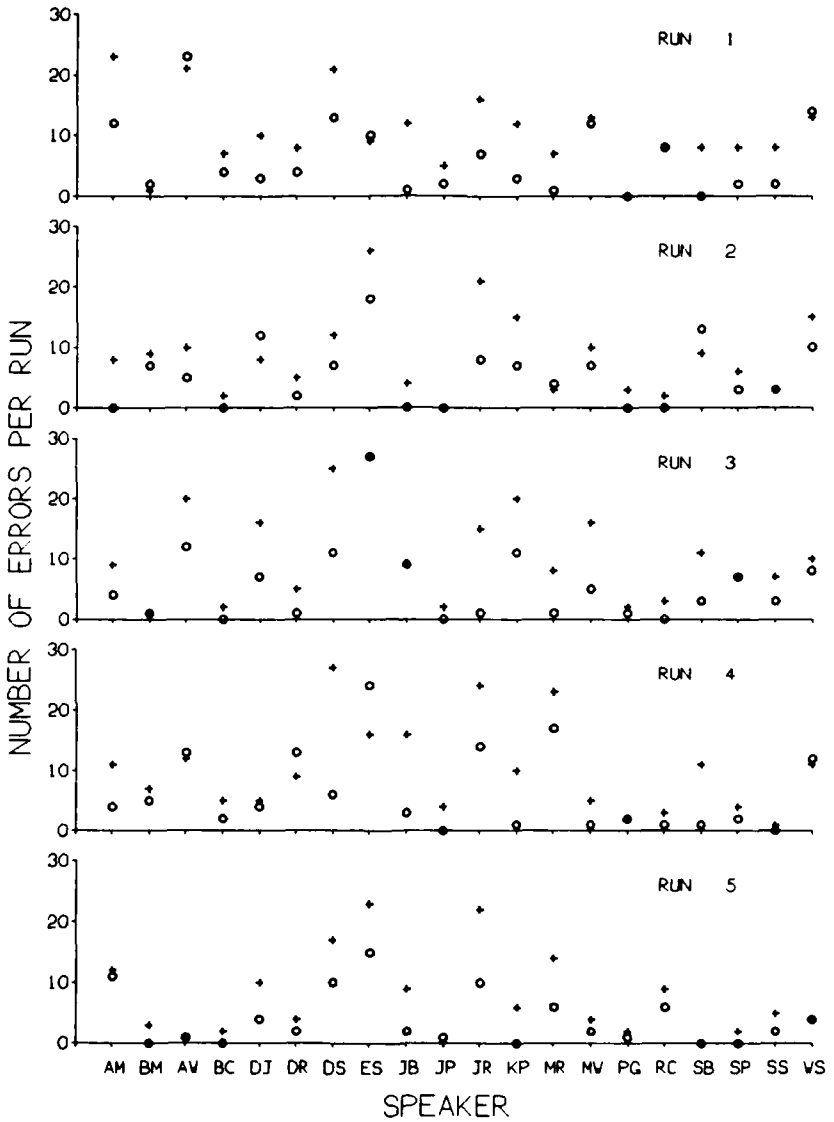


Figure 10: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.75 and α=0.5. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

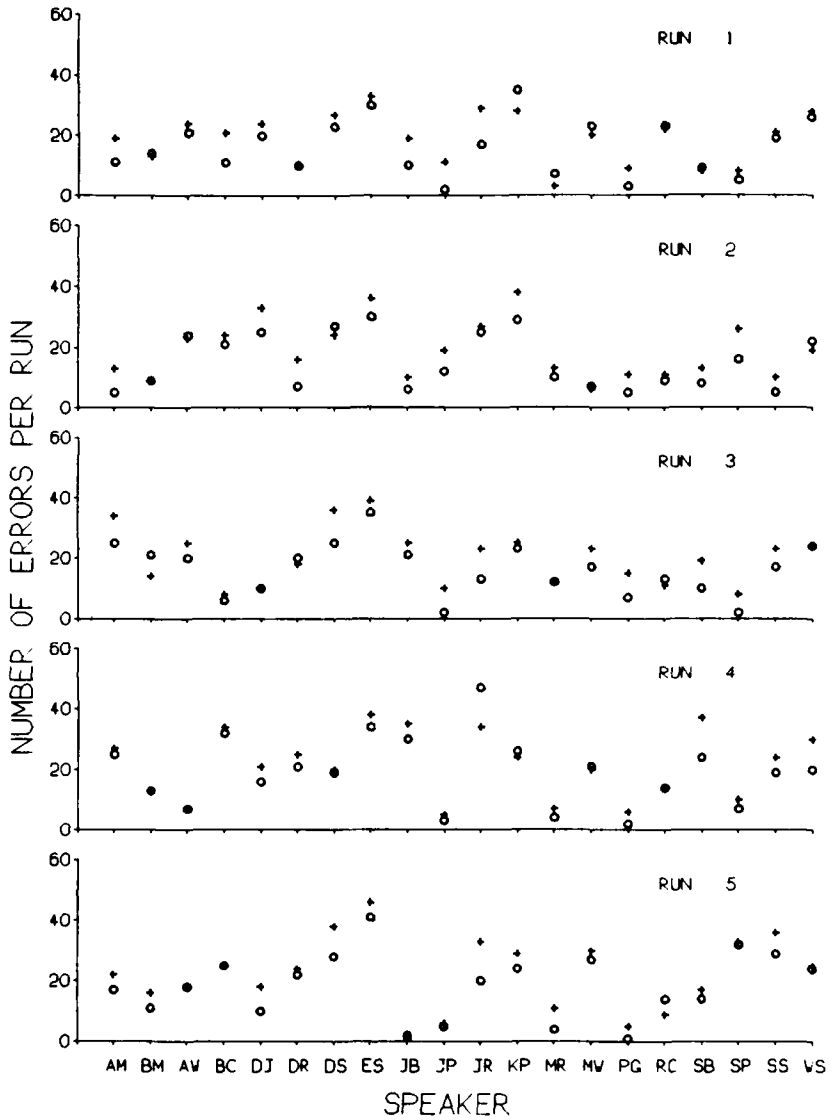


Figure 11: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.5 and α=0.75. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

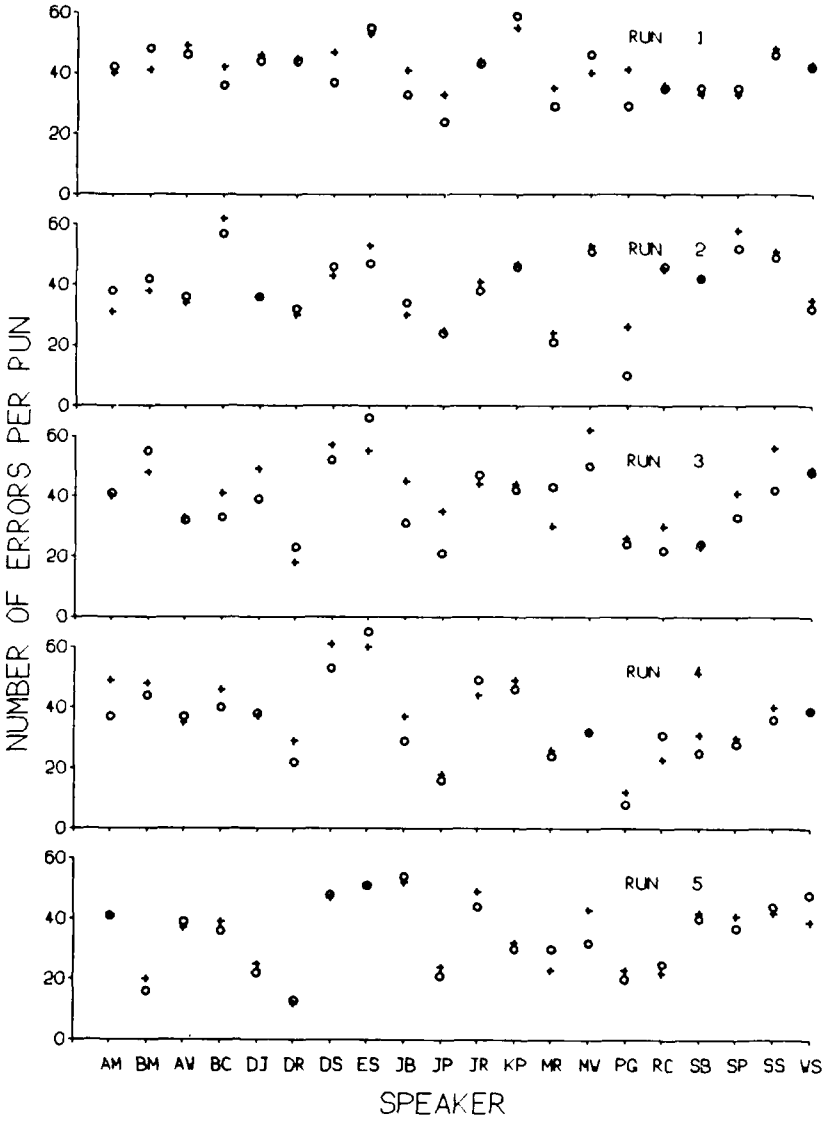


Figure 12: Errors after 1000 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.75 and α=0.75. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

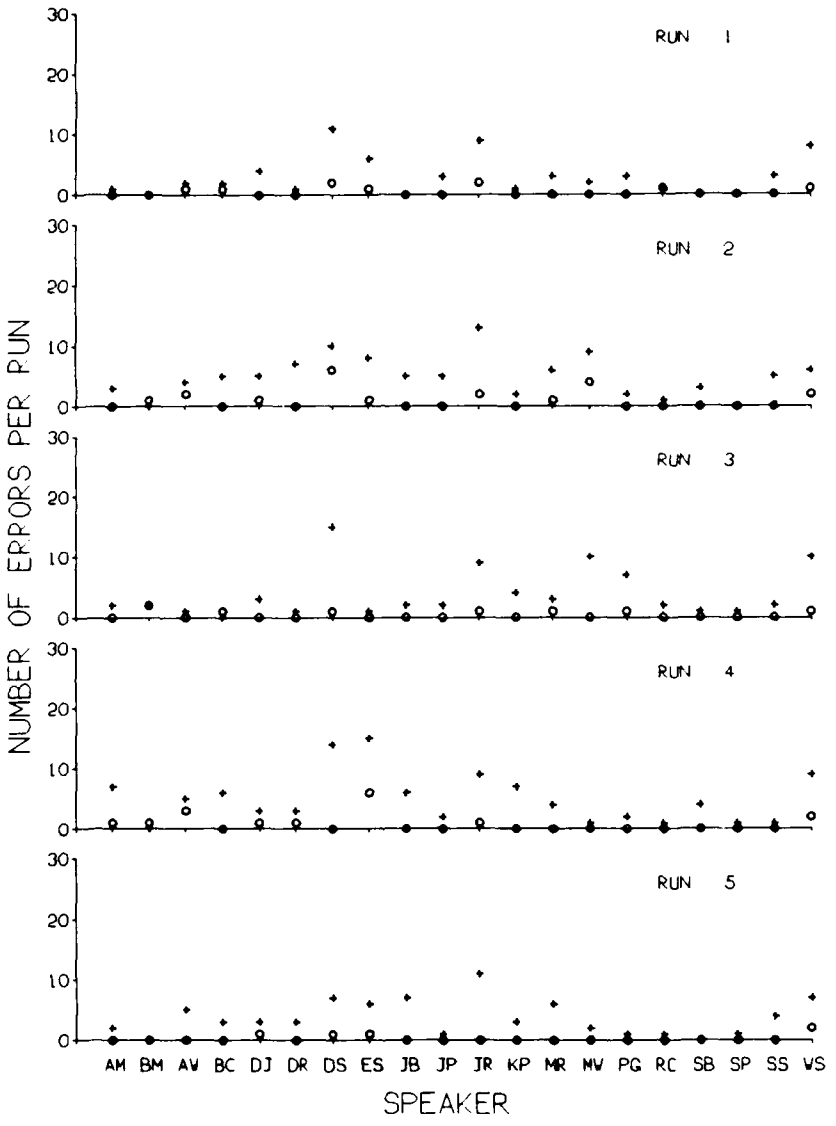
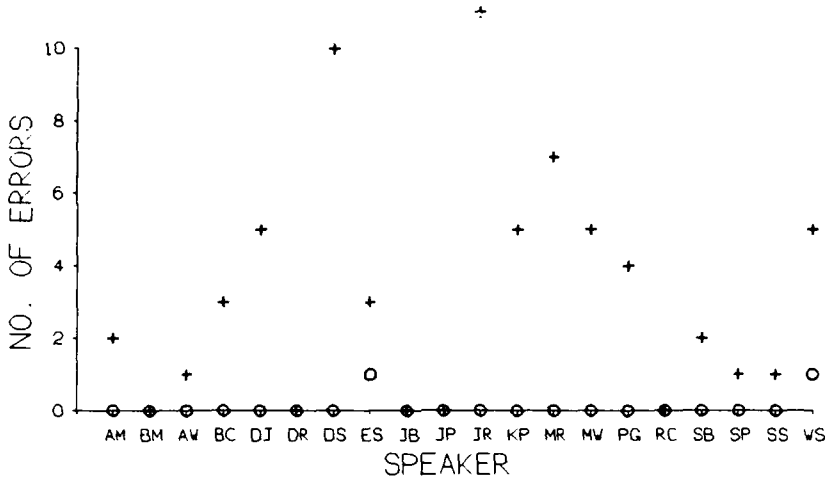


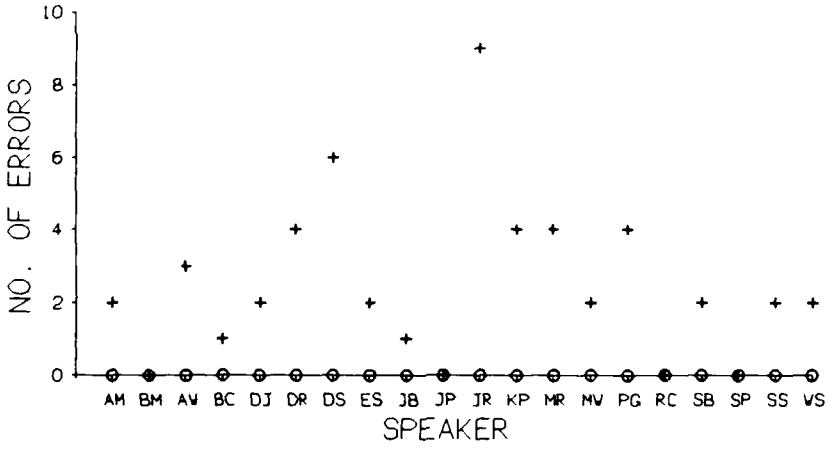
Figure 13: Errors after 1500 pattern presentations from MLP's with 1 x 8 hidden units, ϵ=0.05 and α=0.9. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

Appendix B Effect of Using Different *Termination* Criteria

The graphs here show the training and testing errors obtained using an MLP with a single layer of eight hidden units. Two pairs of learning rate/ momentum scaling term combinations are used, ϵ=0.25, α=0.5 and ϵ=0.05, α=0.9. Two different termination criteria are used for the learning phase. The upper graph shows the errors from terminating the learning phase after the MLP has seen 1000 or 1500 examples of each word. The lower graph shows the errors from terminating when the average error from the output units is less than 0.001 per word.

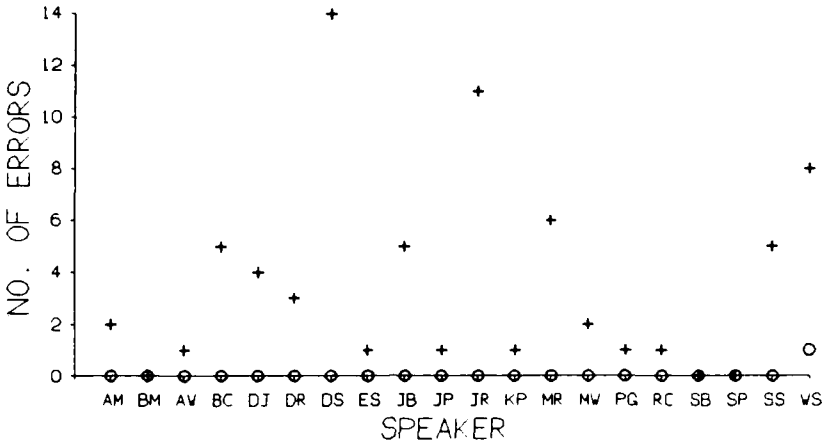


(a) 1000 presentations

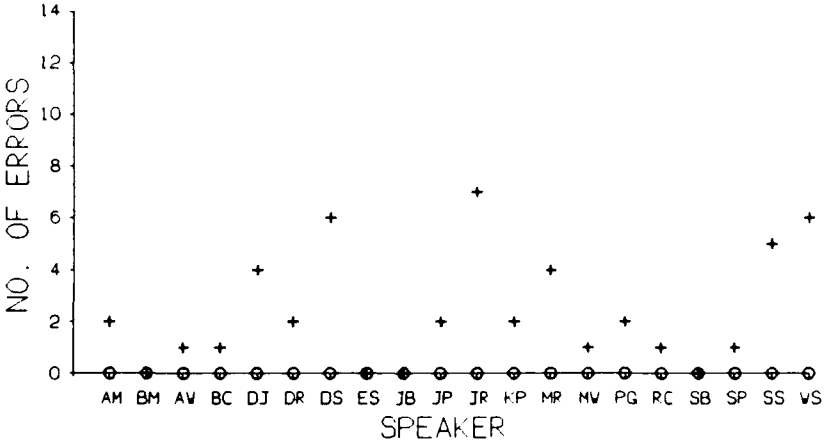


(b) Error < 0.001

Figure 14: Errors for MLP with 1 x 8 hidden units, ϵ=0.25 and α=0.5 with different termination criteria. Circles represent training set errors and crosses represent test set errors, both over 100 digits.



(a) 1500 presentations



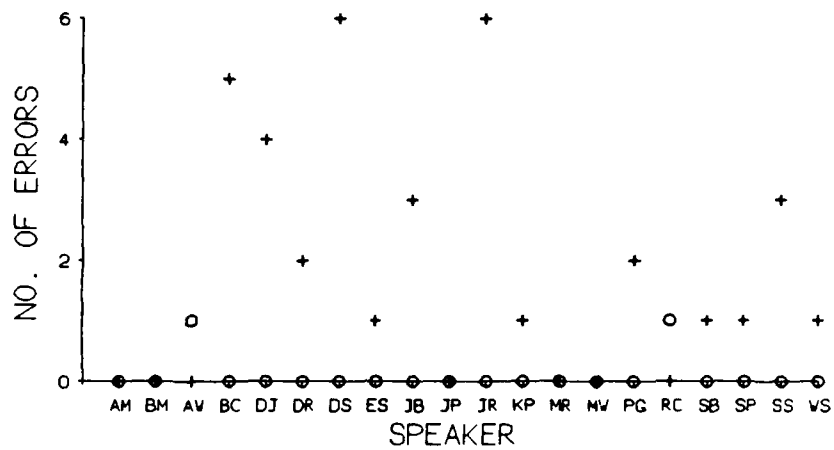
(b) Error < 0.001

Figure 15: Errors for MLP with 1 x 8 hidden units, ϵ=0.05 and α=0.9 with different termination criteria. Circles represent training set errors and crosses represent test set errors, both over 100 digits.

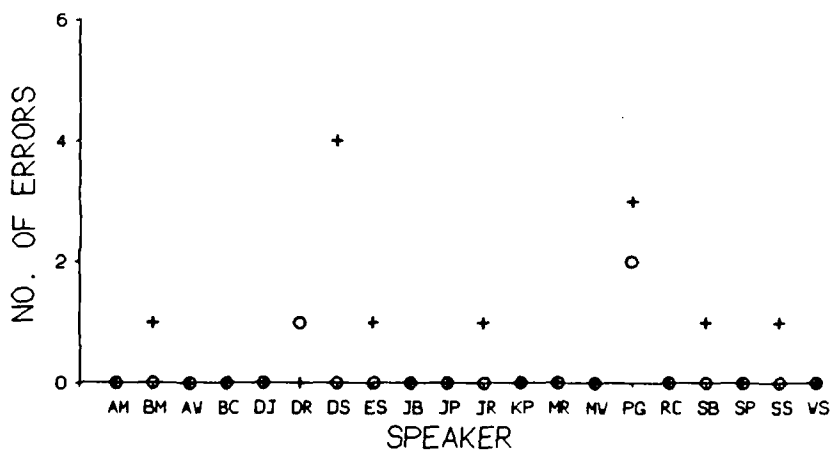
Appendix C Effect of Using Different Numbers of Training Words

The graphs here show the effect of using either 10 or 30 examples of each digit during the training phase of the speaker dependent experiments. Graphs are shown for 0, 1 x 8, 2 x 8, 1 x 50 and 2 x 50 hidden units with different values of ϵ and α. Except in the case of 0 hidden units, the training phase continued until the error was less than 0.001.

For 0 hidden units, the training phase was done in two parts. In the first part, the MLP was trained until the error was less than 0.01 (typically about 8000 presentations) then these weights were used as start-up weights for the next part. In practice, it was never possible to get the error below 0.001 so the MLP learnt over an extra 15000 pattern presentations. At the end of this second stage the error was typically about 0.005. Hence, there are two pairs of graphs for the MLP with 0 hidden units: these show the errors at the end of the first and second stages.

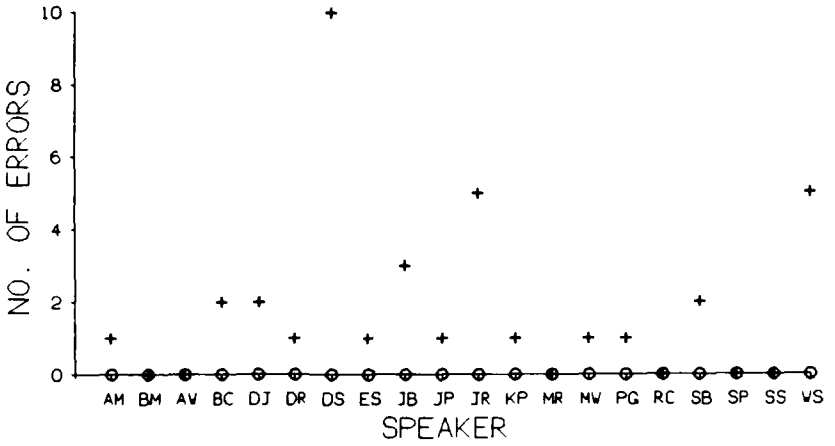


(a) 1 Training File

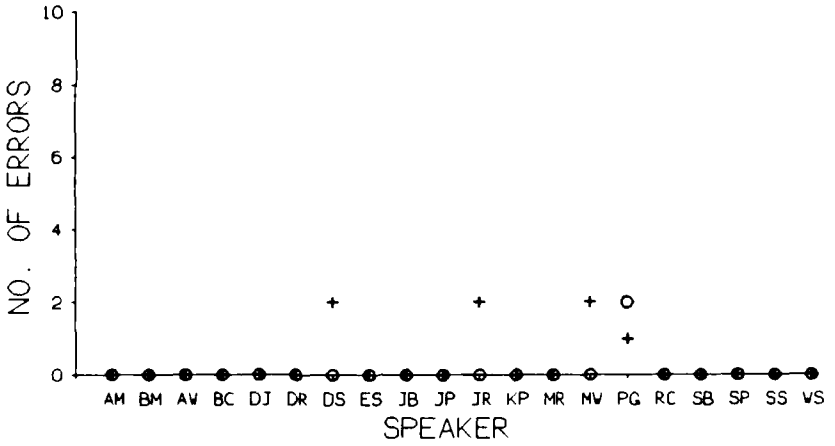


(b) 3 Training Files

Figure 16: Errors for MLP with 0 hidden units, ϵ=0.07 and α=0.5 trained until the error was less than 0.01. Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.

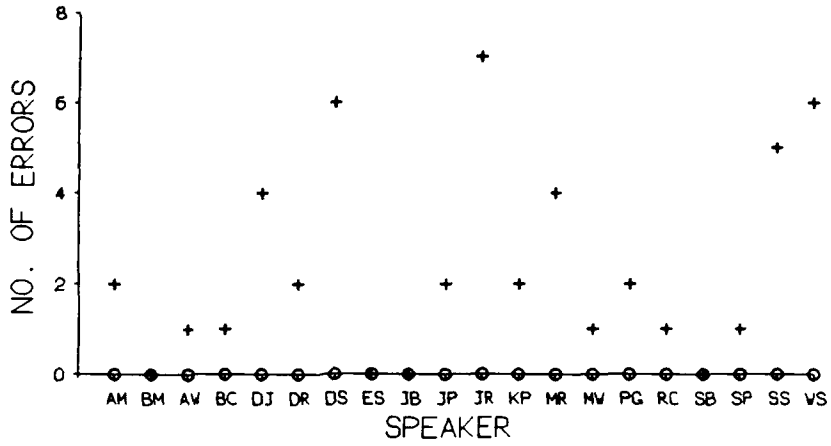


(a) 1 Training File

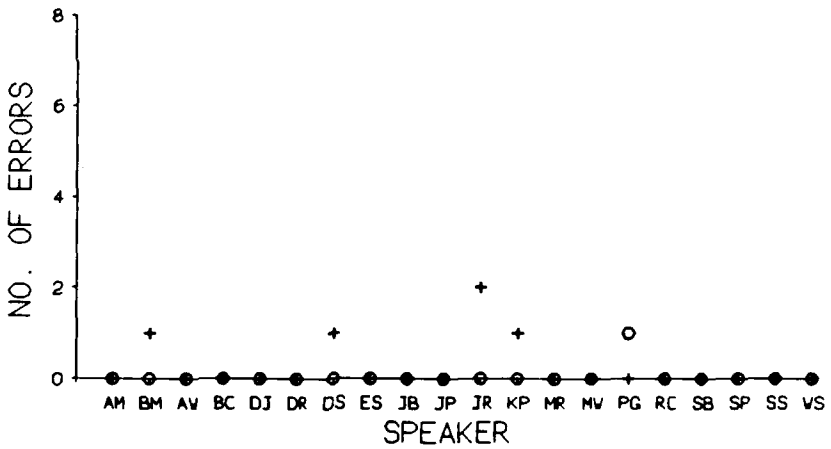


(b) 3 Training Files

Figure 17: Errors for MLP with 0 hidden units, ϵ=0.07 and α=0.5 trained until the error was about 0.005 (see explanation at start of this Appendix). Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.

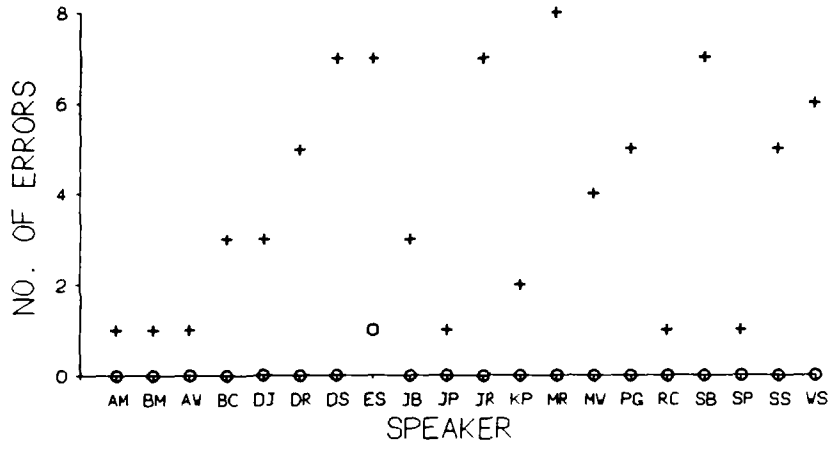


(a) 1 Training File

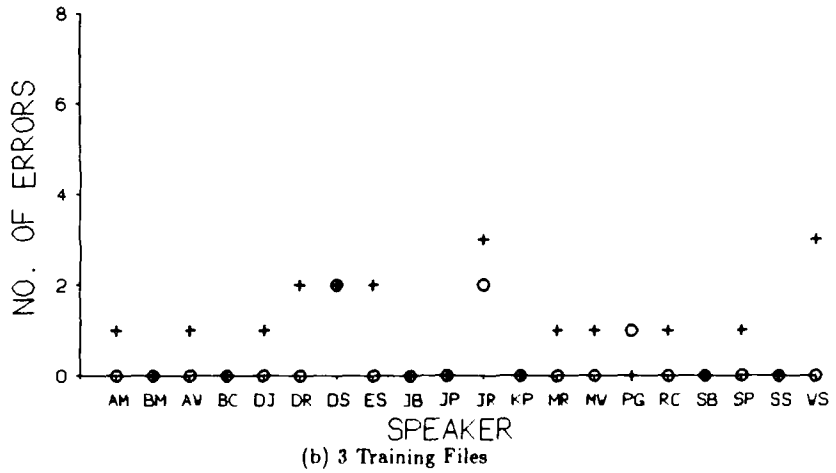


(b) 3 Training Files

Figure 18: Errors for MLP with 1 x 8 hidden units, ϵ=0.05 and α=0.9 trained until the error was less than 0.001. Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.

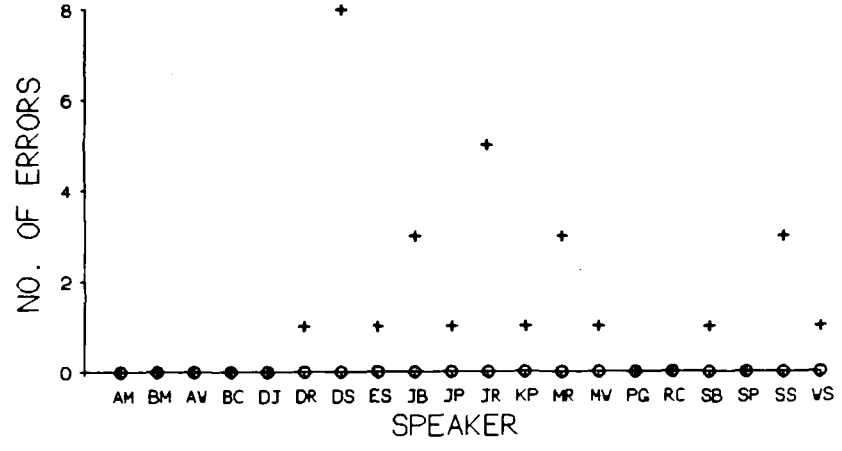


(a) 1 Training File

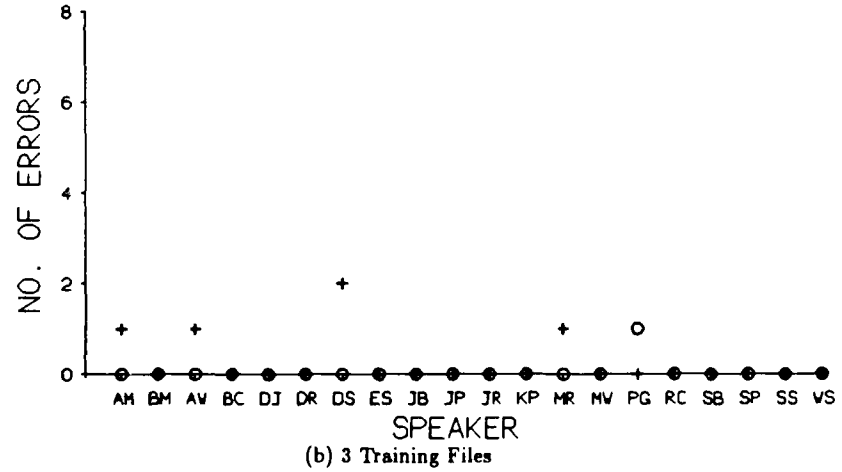


(b) 3 Training Files

Figure 19: Errors for MLP with 2 x 8 hidden units, ϵ=0.05 and α=0.9 trained until the error was less than 0.001. Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.

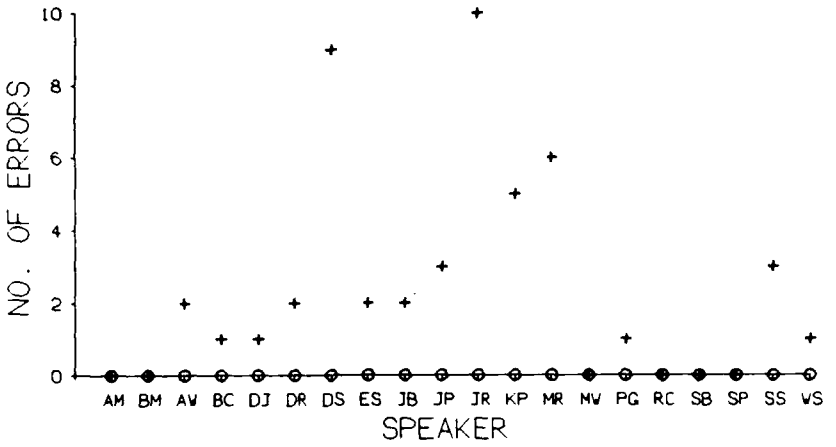


(a) 1 Training File

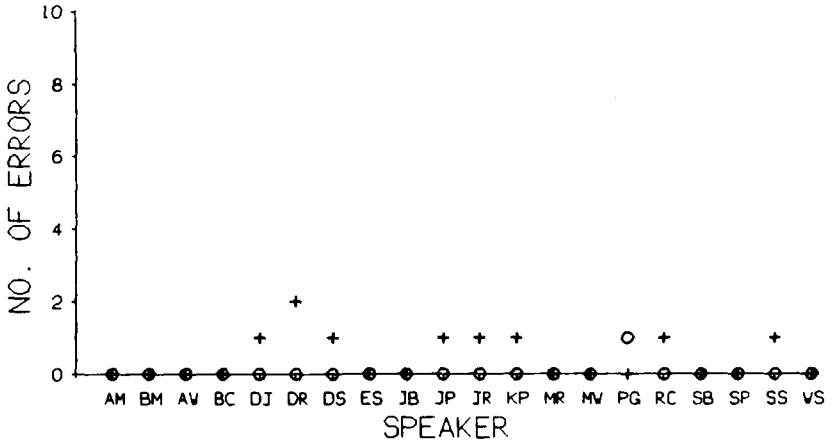


(b) 3 Training Files

Figure 20: Errors for MLP with 1 x 50 hidden units, ϵ=0.15 and α=0.75 trained until the error was less than 0.001. Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.



(a) 1 Training File



(b) 3 Training Files

Figure 21: Errors for MLP with 2 x 50 hidden units, ϵ=0.25 and α=0.5 trained until the error was less than 0.001. Circles represent training set errors, over 100 digits in (a) and 300 digits in (b). Crosses represent test set errors, over 100 digits in both graphs.

DOCUMENT CONTROL SHEET

Overall security classification of sheet UNCLASSIFIED

(As far as possible this sheet should contain only unclassified information. If it is necessary to enter classified information, the box concerned must be marked to indicate the classification eg (R) (C) or (S))

1. DRIC Reference (if known)	2. Originator's Reference	3. Agency Reference	4. Report Security Classification
	Memo 4073		U/C

5. Originator's Code (if known)	6. Originator (Corporate Author) Name and Location
778400	RSRE St Andrews Road, Malvern, Worcs. WR14 3PS

5a. Sponsoring Agency's Code (if known)	6a. Sponsoring Agency (Contract Authority) Name and Location

7. Title
EXPERIMENTS IN ISOLATED DIGIT RECOGNITION USING THE MULTI-LAYER PERCEPTRON

7a. Title in Foreign Language (in the case of translations)

-

7b. Presented at (for conference papers) Title, place and date of conference

8. Author 1 Surname, initials	9(a) Author 2	9(b) Authors 3,4...	10. Date	pp. ref.
PEELING, S.M.	MOORE, R.K.		1987.12	39

11. Contract Number	12. Period	13. Project	14. Other Reference

15. Distribution statement

Descriptors (or keywords)

continue on separate piece of paper

Abstract The multi-layer perceptron is investigated as a new approach to the automatic recognition of spoken isolated digits. The choice of the parameters for the multi-layer perceptron is discussed and experimental results are reported. A comparison is made with established techniques such as dynamic time-warping and hidden Markov modelling applied to the same data. The results, for this particular task, show that the recognition accuracy obtained using the multi-layer perceptron is comparable with that from using hidden Markov modelling.

END

DATE
FILMED

588