A COMPARISON OF THREE NEURAL NETWORK ARCHITECTURES FOR AUTOMATIC SPEECH RECOGNITION

BRAD A. HAWICKHORST AND STEPHEN A. ZAHORIAN
Department of Electrical and Computer Engineering
Old Dominion University, Norfolk, VA 23529

RAM RAJAGOPAL
Department of Electrical and Computer Engineering
Old Dominion University, Norfolk, VA 23529

ABSTRACT:
Neural networks are often used as a powerful discriminating classifier for tasks in automatic speech recognition. They have several advantages over parametric classifiers such as those based on a Mahalanbois distance measure, since no (possibly invalid) assumptions are made about data distributions. However there are disadvantages in terms of amount of training data required, and length of training time. In this paper, we compare three neural network architectures--two layer feedforward perceptrons trained with back propagation, Radial Basis Function (RBF) networks, and Learning Vector Quantization (LVQ) networks--with respect to these issues. These networks were also experimentally evaluated with vowel classification experiments using a Binary Pair Partitioned scheme, requiring a total of 120 pairwise sub-networks for a 16 category vowel classifier. Although the three network architectures give similar average classification performance (i.e., for the 120 sub-networks), there are large differences in terms of training time, performance of individual sub-networks, and amount of training data required. These results, and the implications for choosing network architectures in a larger speech recognition system are presented.

INTRODUCTION

Neural networks trained with backpropagation have been used to accurately classify vowel sounds in continuous speech recognition systems. However, the length of time required to train the networks can present problems, particularly when investigating a variety of feature sets to represent speech data. This paper compares the performance of multi-layer feedforward perceptrons trained with backpropagation versus that of two alternative neural network paradigms-- Radial Basis Function (RBF) networks, and Learning Vector Quantization (LVQ) networks. Classification accuracy, training speed and network evaluation speed are considered. The effect of reduced training data is also compared for each of the three networks.

DESCRIPTION OF DATA SET

All networks were evaluated on the same data set, which consisted of a ten feature cepstral coefficient representation of vowel sounds extracted from continuous speech (TIMIT database). All coefficients were scaled for a mean of 0.0 and a standard deviation of 0.2. For each of 16 vowel sounds, a training set and an evaluation set were used for training and testing. The relative sizes of these sets corresponded to the frequency of occurrence of the vowels in the speech samples. The number of vowel tokens per set ranged from approximately 500 to 2500 for training with about 1/10 as much data for testing.

For the tests conducted in this study, the task of each neural network is to classify an input vector as one of only two vowel sounds on which it was trained. This method has been found to have advantages over a single large network (Zahorian and Nossair, 1993. Since there are 16 vowels sounds, 120 networks are required to cover all possible vowel combinations. Performance of network paradigms was evaluated using average results over all 120 networks.

BACKPROPAGATION MODEL

Backpropagation uses a gradient descent approach to minimize output error in a feed-forward network. The algorithm involves presenting an input vector, comparing the network output to the desired output for that vector, and updating each weight by an amount corresponding to the derivative of the error with respect to that weight times some learning rate, and adjusted by a "momentum" factor.

The architecture used in these experiments consisted of five tan-sigmoid hidden units and one output tan-sigmoid output unit. Weights were updated after presentation of each training vector using a variable learning rate and momentum to speed training . After each epoch, i.e. all tokens from each vowel set processed, the learning rate was reduced by a factor of .89 and processing continued using random starting indexing for each set. Training continued until 160,000 training vectors had been processed. Classification results for backpropagation, for both training and test data, are presented along with the best results of the other network paradigms in table 3 as an average for all 120 networks.

The backpropagation networks displayed good classification, and more importantly, good generalization to the test set. The major drawback was the 77 million floating point operations required to train the network.

RADIAL BASIS FUNCTION MODEL

The radial basis function network uses hidden units with localized receptor fields. The RBF network hidden unit can be thought of as representing a point in N-dimensional space which responds to input vectors whose Cartesian coordinates are close to those of the hidden unit, where N is the number of features in the data representation. Many transfer functions may be used for the hidden units -- but the most common is Gaussian, which gives a response that drops off rapidly as the distance between the hidden unit and the input vector increases and is symmetrical about the radial axis-- hence the name Radial Basis Function. The rate with which the response drops is determined by the "spread" of the hidden unit. The output layer of an RBF network is linear. The challenge of designing an RBF network lies in properly placing hidden layer neurons and choosing an optimal value for the spread constant such that the entire input space of interest is covered with minimum overlap. These decisions are usually made empirically, rather than through automatic training methods.

Like feedforward perceptron networks trained with backpropagation, radial basis function networks are capable of approximating any continuous function with arbitrary accuracy given enough hidden units. A major advantage of the radial basis function network is usually considered to be its short training time in comparison to backpropagation, although the computation and storage requirements for classification of inputs after the network is trained is usually greater.

LEARNING VECTOR QUANTIZATION MODEL

Learning vector quantization employs a self-organizing network approach which uses the training vectors to recursively "tune" placement of competitive hidden units that represent categories of the inputs. Once the network is trained, an input vector is categorized as belonging to the class represented by the nearest hidden unit.

The hidden units may be thought of as having inhibitory connections between each other so that the unit with the largest input "wins" and inhibits all other units to such an extent that only the winning unit generates an output. In the computer simulation there are no actual inhibitory connections and the winner is simply the hidden unit "closest" to the input vector

As with the RBF network, each hidden unit can be thought of as representing a point in N-dimensional space. In both network types the output of the hidden units are based on the proximity of the input vector - the difference between the two is that in an RBF network, several Gaussian hidden units can have significant outputs, while in the LVQ network the output of all but one competitive unit is zero. The final output of both network types is determined by the weights of the linear output unit. These weights are computed by solving the linear network equation **W**o=**T**/**P**, where **W**o is the linear weight vector, **P** is a matrix of inputs into the output layer corresponding to all training vectors, and **T** is a vector containing target outputs associated with the training vectors.

Training an LVQ network is accomplished by presenting input vectors and adjusting the location of hidden units based on their proximity to the input vector. The nearest hidden unit is moved a distance proportional to the learning rate, toward the training vector if the class of the hidden unit and the training vector match, and away if they do not. The hidden layer weights are trained in this manner for an arbitrary number of iterations, usually with the learning rate decreasing as training progresses. The objective is to place the hidden units so as to cover the decision regions of the training set.

LVQ networks have been found to perform well in pattern classification (Kohonen, 1990). As with RBF networks, they tend to have shorter training time requirements than feedforward networks trained with backpropagation, but processing required for input classification may be larger since more hidden units are often required.

EXPERIMENTAL RESULTS

All networks were simulated using the neural network toolbox of the software package MATLAB, by Mathsoft Inc. The programming language is not compiled and therefore rather inefficient in execution. The intent of these experiments was to compare the relative performance of the three network types, not to produce efficient neural network code.

RBF

The method used to train the hidden layer of the RBF was to evaluate every input as a possible location for a hidden unit, then choose the location which resulted in the least number of misclassifications. The process was repeated until the desired number of hidden units were found. A Vector Quantization algorithm was used to reduce the training sets to 256 codewords per vowel, then the networks were trained on the codewords. The trained networks were evaluated on the original training sets as well as the evaluation sets.

The two primary variables which affected performance were the number of hidden units used and the Gaussian spread. The best value for the spread constant changed as the number of hidden units increased, since the density of hidden units in vector space increased. For networks of less than 20 hidden units, a spread value of .8 produced the best classification accuracy for most vowel pairs.

TABLE 1: EFFECT OF NUMBER OF HIDDEN UNITS ON RBF

Vowel Pair "OY-UH"	Classification Accuracy		Computation Required	
	Training Set	Evaluation Set	Training-MFLOPS	Evaluation-FLOPS
2 hidden units	77.3%	82.4%	16	94
8 hidden units	83.2%	80.7%	31	370
16 hidden units	81.5%	80.7%	52	738
100 hidden units	91.5%	79.0%	700	4602

The accuracy of classification of the training set increased with the number of hidden units; however, generalization degraded with large numbers of hidden units and the amount of computation required for both training and evaluation was also greatly increased. Eight hidden units produced the best compromise of classification accuracy and computation burden.

LVQ

The primary factors which controlled the behavior of the LVQ network were the number of hidden units, learning rate and training time. Two different schemes for initial placement of the hidden units were tried -- random placement and placement of all units at the mean value of the training set. The networks seemed to stabilize to the same solution regardless of the initialization of the hidden layer weights. The number of hidden units had a greater effect , as shown in table 2 for one vowel pair with a learning rate of .03 decreasing by .98 every 100th of 5000 cycles.

TABLE 2: EFFECT OF NUMBER OF HIDDEN UNITS ON LVQ NETWORK

Vowel Pair “OY-UH”	Classification Accuracy		Computation Required	
	Training Set	Evaluation Set	Training-MFLOPS	Evaluation-FLOPS
2 hidden units	78.8%	84.2%	.8	82
8 hidden units	77.7%	75.4%	2.7	322
16 hidden units	77.9%	75.4%	5.1	642
100 hidden units	77.9%	70.1%	28.9	4002

The generalization capacity of the networks seemed to decrease as hidden units were added. Experiments run over all 120 vowel pairs indicated that the best LVQ performance for this application was achieved with only 2 hidden units trained with 1000 input vector presentations.

COMPARISON OF BEST RESULTS FOR THE THREE NETWORKS

The best results for RBF and LVQ are summarized in table 3 along with the backpropagation results. Classification accuracy is presented as an average of all 120 vowel-pair networks.

TABLE 3: BEST PERFORMANCE OF THREE NETWORK TYPES

Average Performance for 120 Networks	Classification Accuracy		Computation Required	
	Training Set	Evaluation Set	Training-MFLOPS	Evaluation-FLOPS
RBF - 8 hidden units	89.9%	90.6%	32.9	370
LVQ - 2 hidden units	88.8%	89.7%	0.2	82
Backpropagation - 5 hidden units	91.9%	92.5%	77.4	152

An examination of the performance of individual networks revealed that in cases in which one vowel in a pair had a many more input vectors than the other, the backpropagation networks tended to favor the vowel with more inputs, classifying a much greater percentage of the smaller set incorrectly than the larger. Neither the RBF or the LVQ networks suffered from this problem.

The number of MFLOPS displayed above is somewhat misleading in terms of execution time. MATLAB reported roughly twice the number of floating point operations to train each network for backpropagation compared to RBF; however, the *time* required to train a network with backpropagation was about ten times that required for an RBF network. The cause of this discrepancy appears to be that the RBF code used large matrix operations which seemed to be the most efficient processing mode for MATLAB. The backpropagation code used a large amount of looping, which was apparently very inefficient in the uncompiled MATLAB code.

EFFECT OF SMALLER TRAINING SETS

The size of the available training set is often less than what is desired in automated speech recognition applications. In order to compare the effects of a smaller training set on classification accuracy, the best performing networks of the three types were trained on much smaller input sets, reduced by selecting every thirtieth input vector and discarding the rest. These tests represent the average of six vowel-pair networks chosen arbitrarily. Note that these six vowel pairs were more difficult to discriminate than the average of all 120 vowel pairs.

TABLE 4: EFFECT OF REDUCED TRAINING SET

Average Performance for 6 networks		Classification Accuracy	
		Training Set	Evaluation Set
Backpropagation	Full Set	84.3%	85.2%
	Reduced Set	87.3%	80.7%
LVQ	Full Set	80.4%	82.5%
	Reduced Set	83.9%	77.4%
RBF	Full Set	83.5%	83.2%
	Reduced Set	89.8 %	82.1%

In all three cases, the networks were able to learn the much smaller training sets with a greater degree of accuracy; however, the RBF networks appear to have maintained a higher degree of generalization than the other two networks.

CONCLUSIONS

Neither of the two network paradigms under consideration were superior in terms of classification accuracy to feedforward perceptrons trained with backpropagation; however, some interesting characteristics were observed.

The RBF network came very close to matching the accuracy of backpropagation with a much shorter training time, and seemed to retain its generalization capacity better than backpropagation when trained on a drastically reduced input set. These are both important considerations in speech processing applications.

The LVQ network was less accurate than feedforward multi-layer perceptrons trained with backpropagation and it was somewhat surprising that the best performance was achieved with only two hidden units.. Although it is not as accurate as the backpropagation or RBF networks, an LVQ network with two hidden units may be useful in evaluation of speech data feature sets since both training time and evaluation time are extremely short.

Neither RBF nor LVQ shared the backpropagation problem of favoring the larger set in vowel pairs with mismatched input set sizes. However, the backpropagation can also be modified to eliminate this discrepancy.

REFERENCES

Kohonen, T., (1990). Statistical Pattern Recognition Revisited, *Advanced Neural Computers* R. Eckmiller (editor), 137-144.

Norton, C. A., and Zahorian, S. A. (1995). Speaker Verification with Binary-Pair Partitioned Neural Networks, ANNIE-95.

Zahorian S. A., Nossair, Z. B., and Norton, C. A., (1993). A Partitioned Neural Network Approach for Vowel Classification using Smoothed Time/Frequency Features, *Eurospeech-93*, 1225-1228.