T. Homma, L.E. Atlas and R.J. Marks II, "A neural network model for vowel classification," Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 1987. Published as a reprint in Proceedings of the 1988 Connectionist Models Summer School, (Morgan Kaufman Publishers, San Mateo, CA. 1988) pp.380-387.

A Neural Network Model for Vowel Classification

Les E. Atlas
Toshiteru Homma
Robert J. Marks II

Interactive Systems Design Lab
Department of Electrical Engineering, FT-10
University of Washington
Seattle, WA 98195

ABSTRACT

The notion of a global computation performed by a large number of simple and identical logic elements (neural network) has recently stirred great interest in the field of pattern recognition. The appeal lies in the simulation of human neural processing and the high potential for fast VLSI and optical architectures. In order to gain a preliminary understanding of the efficacy of this approach for speech analysis applications, we have applied several simulated neural networks to the problem of speaker-independent vowel normalization. The results of this normalization were compared with human perceptual data and it was found that an orthogonal projection neural network matched the human data fairly well. It was also found that a conventional matched filter classifier performed better than the neural networks. A neural network version of this best classifier was derived and tested.

INTRODUCTION

There has recently been a rapid growth of interest in artificial neural network research. In 1982, Hopfield published a paper [1] which has inspired a number of researchers in various disciplines to investigate neural network models. The original model presented was a programmable, auto-associative memory for static binary patterns. Its operation was asynchronous and stochastic. This model had the ability to recall stored memories when a partial and noisy version of the memory was input. McEliece et al. [2] analyzed the model to estimate its information storage capacity using statistical approximations. Several researchers reported optical implementation of the model with some simplification and modification (e.g., Psaltis and Farhat [3]; Athale et al. [4]). Typically, these models operated synchronously with uniformly zero thresholds and no external inputs.

Instead of using the outer product matrix in the Hopfield model, Personnaz et al. [5] suggested an orthogonal projection matrix which projects an input vector onto the subspace spanned by the stored memory vectors. Their model operated synchronously with zero thresholds, bipolar (+1 and -1) states, and no external inputs. On the condition that all the stored vectors are linearly independent, the model showed more reliable auto-associative capability than the original Hopfield model.

Based on these results, we simulated various neural network formulations, applied these to recognize vowels whose spectra are time invariant, and evaluated performance by a sequence of test inputs which scan through the F_1 - F_2 feature plane. One simulated neural network model was the modified Hopfield model with the orthogonal projection matrix, which operated synchronously with zero thresholds and no external inputs. The original model using the outer product matrix was also simulated for comparison. A conventional thresholded matched filter classifier was used as a control and a neural network approximation of the control was also simulated.

Vowel spectra were quantized as binary input and storage patterns. The test input sequence was created by specifying the first two formant frequencies, F_1 and F_2 scanning through the F_1 - F_2 plane. The prototypes were obtained using the averaged formant frequencies found in the classic study by Peterson and Barney [6] and the performance of the network formulation was assessed by comparisons with human perceptual performance. The goal of this study was to check the validity of these network types for the limited task of speaker independent vowel recognition.

BACKGROUND AND NETWORK FORMULATIONS

Hopfield Model

The basic idea behind Hopfield's net is to have L identical neurons, each with an initial stored binary value. The ith neuron is assumed to be connected to the jth neuron with a transmittance of T_{ij}. If the sum of the inputs to a neuron exceeds a specified threshold, the neuron fires (sets its binary value). If the sum is less than the threshold, the neuron turns off (resets its binary value). This process continues for all neurons until a stable state is reached.

The choice of T_{ij}'s is the programmed part of the neural network. Hopfield designed these interconnection values as a representation of stored binary vectors or "library elements." Given a portion of one of the library element vectors, the entire vector can sometimes be regenerated by iterative operations described above. Furthermore, the algorithm is highly tolerant of faults. Elements (neurons) can be destroyed or interconnect values can be grossly quantized [7], and the processor will still be functional.

Synchronous Hopfield Model (Outer Product)

In Hopfield's derivation of the choice of interconnects, an energy function was formulated and minimized. Also, it was assumed that neurons were not synchronous. In order to utilize conventional signal space formulations, a synchronous version of the Hopfield model can be derived:

Let $\{\mathbf{f}_n \mid 1 \le n \le N\}$ denote a set of library elements, each of length L. The ith element of $\mathbf{f}_n$ is f_{ni}. Each f_{ni} is either 1 or −1. We form the $L \times L$ matrix T with elements

$$T_{ij} = \sum_{n=1}^{N} f_{ni} f_{nj}; \; i,j = 1,2,\ldots L$$

Presented at International Conf. Acoust., Speech, and Sig. Proc., Dallas, TX, April 6-10, 1987

We define the $L \times N$ library matrix by

$$\mathbf{F} = [\, \mathbf{f}_1 \; \mathbf{f}_2 \; \cdots \; \mathbf{f}_N \,]$$

resulting in

$$\mathbf{T} = \mathbf{F}\mathbf{F}^T$$

Hence, we refer to T as the *outer product matrix*. Note that this model assumes symmetric interconnects ($T_{ij} = T_{ji}$) and autointerconnects ($T_{ii} \neq 0$).

Consider a given initialization or input vector, $\mathbf{g}$, composed of ± 1's. We can form the iteration

$$\mathbf{g}_{M+1} = \mathrm{sgn}[\mathbf{T}\mathbf{g}_M]$$

where $\mathbf{g}_0 = \mathbf{g}$ and the vector operator, sgn, examines each element of $\mathbf{T}\mathbf{g}_M$ and, if the element is positive, sets it to 1. Otherwise, the element is reset to -1.

Each iteration can be described from a signal space standpoint. $\mathbf{F}^T\mathbf{g}_M = \alpha$ where each element of α is proportional to the magnitude of a projection of $\mathbf{g}_M$ onto $\mathbf{f}_i$. These α_i actually correspond to the outputs of a matched filtering operation where the input is $\mathbf{g}_M$ and the filter is $\mathbf{f}_i$. $\mathbf{F}\alpha = \mathbf{F}\mathbf{F}^T\mathbf{g}_M$ will therefore be equivalent to a linear combination of the $\mathbf{f}_i$'s weighted by the projection components. The last part of each iteration, which is the sgn, corresponds to a nonlinear thresholding operation. This is equivalent to finding the nearest corner of a hypercube in the L-dimensional signal space.

This overall iterative process is therefore algorithmically equivalent to the block diagram shown in figure 1. If the behavior of the synchronous Hopfield network is the same as the asynchronous net, the above iteration would ideally converge to the library element $\mathbf{f}_m$ that is closest to $\mathbf{g}$ in the Hamming distance sense.

Orthogonal Projection Model

This model involved a change in only the formulation for T, *i.e.* only the interconnection weights were modified in the neural net. As first suggested in [5], we used an orthogonal projection to formulate an alternative to T which will be referred to as $\hat{\mathbf{T}}$. This alternative will project $\mathbf{g}_M$ onto the *subspace* spanned by $\mathbf{f}_i$ where $i = 1,2,\ldots,N$ and $L>N$. The basic idea behind the orthogonal projection is to make sure that the neural network is always able to converge to any given library element when that library element is used as the input. Namely,

$$\mathbf{f}_i = \hat{\mathbf{T}}\mathbf{f}_i$$

for $i = 1,2,\ldots,N$. These equalities are the same as

$$\mathbf{F} = \hat{\mathbf{T}}\mathbf{F}$$

If all $\mathbf{f}_i$ are linearly independent, then rank[F] = N and a pseudo-inverse of F can be defined as

$$\mathbf{F}^+ = [\mathbf{F}^T\mathbf{F}]^{-1}\mathbf{F}^T$$

and the desired equalities imply that

$$\hat{\mathbf{T}} = \mathbf{F}\mathbf{F}^+ = \mathbf{F}[\mathbf{F}^T\mathbf{F}]^{-1}\mathbf{F}^T$$

The iteration for the orthogonal projection network is therefore

$$\mathbf{g}_{M+1} = \mathrm{sgn}[\hat{\mathbf{T}}\mathbf{g}_M]$$

Matched Filter Classifier

This formulation, which does not correspond to a neural network-type design, was used as the control. The goal is to use a conventional technique to find the library element $\mathbf{f}_m$ that is the closest to the input $\mathbf{g}$ in the Hamming distance sense. The first step is the same as the bank of matched filters described in the synchronous Hopfield model, *i.e.* find

$$\alpha = \mathbf{F}^T\mathbf{g}$$

The second step consists of searching through the elements of α to find the maximum, say α_m. The index m then points to $\mathbf{f}_m$ as the closest library element. The rank-ordered search through the elements of α does not fit the usual definitions of a neural network architecture. Furthermore, this control classifier is non-iterative.

Lateral Inhibition Model

This last formulation was driven by the need to put an approximate version of the above matched filter classifier into a neural network architecture. The second step of the matched filter classifier was replaced with an iterative neural network which consisted of positive and identical autointerconnects and negative and identical connections between neurons. Each neuron could take on an integer (and not just binary) value. The effect of this structure was intended to enhance large initial values while suppressing smaller values. In the limit, the processor with the largest initial value will increase infinitely while the value stored at all other processors will continue to decrease. Thus, the largest matched filter output eventually becomes much larger than all other outputs.

In order to make this structure practical, a clipping nonlinearity η was used at each neural output. η had a linear input-output relationship between two empirically chosen thresholds. The maximum matched filter output was indicated whenever a neural output reached the larger of the two thresholds. This neural network-based classified can be expressed as two steps. The first step is identical to the control classifier, *i.e.* find

$$\alpha = \mathbf{F}^T\mathbf{g}$$

Note that this non-iterative step could be accomplished by a neural network with interconnection weights corresponding to the elements of all $\mathbf{f}_i$. The second step is the lateral inhibition iteration

$$\alpha_{M+1} = \eta[\mathbf{H}\alpha_M]$$

where

$$\mathbf{H} = (1 + 1/N)\,\mathbf{I} - (1/N)\vec{1}\,\vec{1}^T$$

where I is the identity matrix, $\vec{1}$ is a vector of length N which consists of all 1's, and H thereby specifies the network interconnections for lateral inhibition.

METHODS

The data base used for this study was the average vowel formant frequencies found for 76 speakers (male, female, and children) and the human identification of vowels produced by listening tests with 70 subjects. Ten vowels were used and all of this data comes from the study by Peterson and Barney [6].

For all four network formulations, a bipolar vector was used to represent the vowel spectra. This vector was constructed by dividing the entire natural logarithm scaled frequency range (100 - 4 kHz) into L sections. The spectra was quantized by assigning +1 to areas in the neighborhood of a formant and −1 to all other frequency bins. This is shown schematically in figure 2. Note that only the first 2 formants were used; the excursions of F_3 outside the chosen frequency range eliminated the usefulness of this higher peak.

The bandwidth of the quantized vowel (which was not determined by Peterson and Barney) was chosen to have a fixed width of 100 Hz for formants below 500 Hz and to have a width equal to 0.2 times the formant frequency for frequencies above 500 Hz.

Using the notation of the previous section, the vowel spectral coding corresponds to $L = 100$. The choice of N is based on the number of stored patterns. In order not to exceed the capacity of the various network formulations, N was chosen to be 10, *i.e.* one average female vowel was chosen as the library element for each vowel classification region. Therefore, in order to generate the various network interconnections, f_i where $i = 1,2,...,10$ were derived by quantizing the 10 average female vowels.

Each network was tested by inputs sets which exhaustively covered all possible choices of F_1 and F_2 where $F_2 > F_1$. The final, stable, vector was determined for each input. These output vectors then were used to determine the classification regions described in the results.

RESULTS

All four vowel classification methods described were tested. The assessment of performance can be made by comparing the automatically classified regions with the average regions found by the human listeners in the Peterson and Barney study. Figure 3 depicts these desired regions within the feature space. The areas which listeners found ambiguous (*i.e.* inconsistent vowel labelings) are not included. These ideal regions are included as dashed lines in all subsequent figures for reference.

Figure 4 shows the classification results for the Synchronous Hopfield Model. It is fairly obvious that this model failed to to perform adequately. Only one out of all 10 vowels was crudely classified. The rest of the vowels, even those that were identical to the stored average vowels, were not identified at all.

The orthogonal projection model's result is shown in figure 5. As intended, all vowels which closely resemble the stored average were classified. However, several problems exist. The first is that a spurious /i/ region exists for vowels with very low formant frequencies. Another problem is the small size of the classification regions. It was not possible to make this network formulation accurately capture the variation in articulation across male, female, and children.

Figure 6 illustrates the regions for the control matched filter classifier. Note that the regions are a better match to the perceptual data than figure 5. Also, the variable detection threshold implicit in this classifier allowed the regions to be larger.

The same experiments with the lateral inhibition model produces the results shown in figure 7. The spurious /i/ region is again seen. Nevertheless, the classification performance is somewhat better than that seen in figure 5.

CONCLUSIONS AND FUTURE WORK

There is a large possible variety of neural network type structures which could be fit to a signal classification problem. We have demonstrated that a synchronous type of Hopfield model is of little use for the problem and parameters studied. An orthogonal projection modification improved the performance dramatically, but did not equal the results seen for a conventional matched filter classifier. It was also shown that a neural network version of the best performing classifier was almost as good.

It still remains to be seen whether the performance of a neural network can be any better than the more conventional techniques that have applied to problems of speaker independence. However, the formulations discussed may be appropriate for certain architectures which are needed for very large vocabularies. We feel that this study has helped to provide a firmer theoretical foundation for more extensive studies of novel architectures for speech analysis. Several needed extensions are the use of neural networks for dynamic patterns in speech, the addition of learning to update the interconnection weights, and the performance of neural networks which include more inspiration from the biology, *e.g.* cascaded networks and temporal synchrony. Other issues important to the speech analysis problem include the potential for neural networks to distinguish regions which are not linearly separable and for learning networks to automatically cluster distinct features [8].

ACKNOWLEDGMENTS

This research was supported by a National Science Foundation Presidential Young Investigator Award and by a contract with the Boeing High Technology Center.

REFERENCES

[1] J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci. USA* 79, pp. 2554-2558, 1982.

[2] McEliece, R., E. Posner, E. Rodemich, and S. Venkatesh, "The Capacity of the Hopfield Associative Memory." (Submitted to IEEE Trans. Inf. Theory)

[3] Psaltis, D. and N. Farhat, "Optical Information Processing Based on an Associative Memory Model of Neural Nets," *Optics Lett.* 10, pp. 98-100, 1985.

[4] Athale, R., H. Szu, and C. Friedlander, "Optical Implementation of Associative Memory with Controlled Nonlinearity in the Correlation Domain," *Optics Lett.* 11, pp. 482-484, 1986.

[5] Personnaz, L., I. Guyon, and G. Dreyfus, "Information Storage and Retrieval in Spin-Glass like Neural Networks," *J. Physique Lett.* 46, pp. 359-365, 1985.

[6] Peterson, G. and H. Barney, "Control Methods Used in a Study of the Vowels," *J. Acoust. Soc. Am.* 24, pp. 175-184, 1952.

[7] Farhat, N., D. Psaltis, A. Prata, and E. Paek, "Optical Implementation of the Hopfield Model," *Applied Optics* 24, pp. 1469-1475, 1975.

[8] Rumelhart, D. and D. Zipser, "Feature Discovery by Competitive Learning," *Cognitive Science* 9, pp. 75-112, 1985.

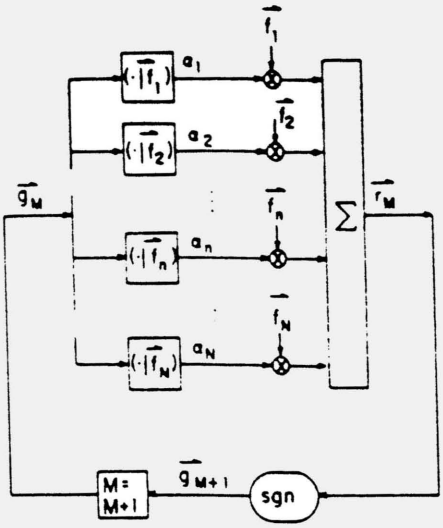

Figure 1. Representation of synchronous Hopfield model.

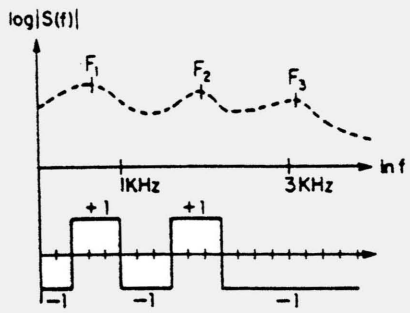

Figure 2. Schematic representation of bipolar coding of vowel spectra.

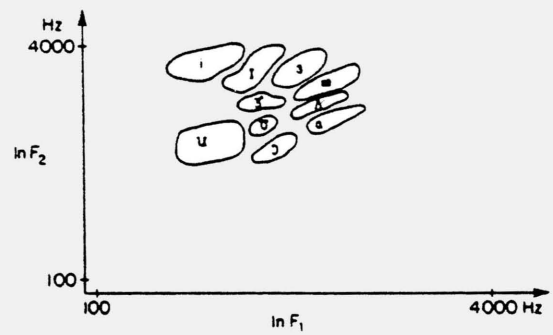

Figure 3. Human listener vowel classification results. Data from Peterson and Barney [6].

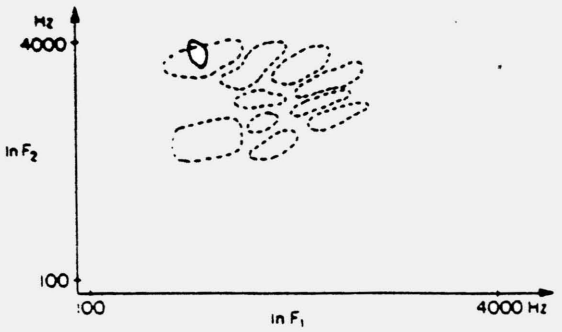

Figure 4. Vowel classification result for synchronous Hopfield model.

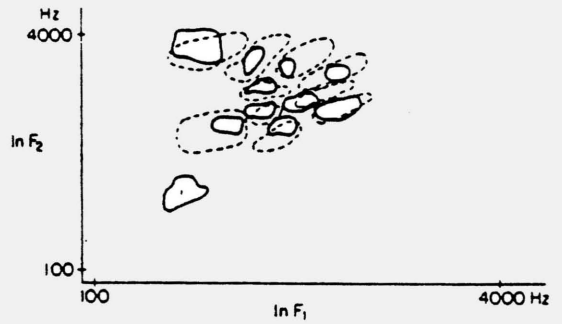

Figure 5. Vowel classification result for orthogonal projection model.

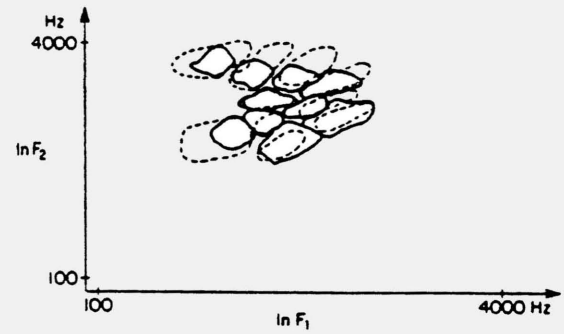

Figure 6. Vowel classification result for matched filter classifier (control).

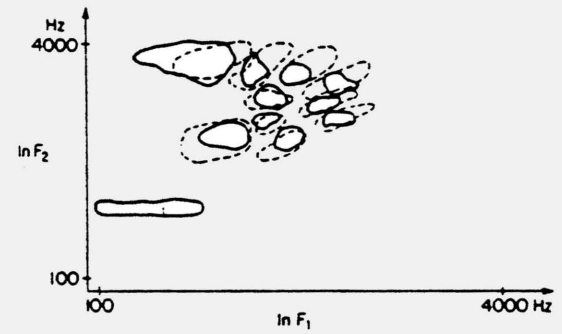

Figure 7. Vowel classification result for lateral inhibition model.