ASSOCIATIVE MEMORY MODELS WITH STRUCTURED CONNECTIVITY

Simon Turvey, Steve Hunt, Ray Frank, Neil Davey
Department of Computer Science, University of Hertfordshire,
Hatfield, AL10 9AB.
UK
{S.P.Turvey, S.P.Hunt, R.J.Frank, N.Davey} @herts.ac.uk

Abstract

High capacity associative memory models with dilute structured connectivity are trained using naturalistic bitmap patterns. The connectivity of the model is chosen to reflect the local spatial continuity of the data. The results show that the structured connectivity gives the networks a higher effective capacity than equivalent randomly diluted networks. Moreover the locally connected networks have a much lower mean connection length than the randomly connected network.

Key Words
Associative Memory, Neural Network, Dilution, Capacity.

1. Introduction

High capacity associative memory models can be constructed from networks of perceptrons, trained using the normal perceptron training procedure. Such networks have a capacity much higher than that of the standard Hopfield network, and in fact their capacity is related to the capacity of a single perceptron. A perceptron with N inputs can learn 2N random unbiased patterns, but this capacity (α = 2) is increased if the training set is correlated [1]. Moreover Lopez et al [2] showed that, even for unbiased patterns the capacity of the perceptron could be much higher than 2, if certain conditions were met in the training data. These improvements in capacity performance are matched by improvements in the performance in networks as associative memories: the attractor basin size of trained patterns can also be increased. In this paper we are interested in networks with diluted connectivity, where an individual perceptron is connected to only a fraction of the other perceptrons in the network. Diluting these networks on a random basis causes the capacity to fall in a roughly linear way with the fraction of connections removed [3]. However when connections are removed the training set of each perceptron in the network is changed, since the training set is then determined by the specific values of the perceptrons to which each unit is connected. We are interested in whether characteristics of certain types of training data can be exploited by diluting the network connectivity in a definite, structured way. In particular we investigate whether networks with a specific pattern of reduced connectivity can give enhanced performance with naturalistic, bitmap training patterns with inherent spatial continuity.

2. Network Dynamics

All the high capacity models studied here are modifications to the standard Hopfield network. The net input, or *local field*, of a unit, is given by: $h_i = \sum_{j \neq i} w_{ij} S_j$ where S is the current state and w_{ij} is the weight on the connection from unit *j* to unit *i*. The dynamics of the network is given by the standard update: $S'_i = \Theta(h_i)$, where Θ is the heaviside function. Unit states may be updated synchronously or asynchronously. Here we use asynchronous, random order updates. A *symmetric* weight matrix and asynchronous updates ensures that the network will evolve to a fixed point. If a training pattern ξ^{μ} is one of these fixed points then it is successfully stored, and said to be a *fundamental memory*. A network state is stable if, and only if, all the local fields are of the same sign as their corresponding unit, equivalently the aligned local fields, $h_i S_i$, should be positive.

3. Network Topology

Associative memory models based on the Hopfield architecture are usually fully connected, so that any spatial relationship between the units in the network is irrelevant. Here, however, we arrange the units in the network into a two dimensional grid, as in a two dimensional SOM [4]. This therefore introduces a topology on the units in the network, and we can use this topology to define a distance between any two units in the network. We use the square neighbourhoods (as is normally the case in a SOM), so that the 8-units in the immediate square around a unit are at distance one from that unit, as shown in Figure 1. We say that the network has structured connectivity with d =1 if every unit is connected to every other unit at distance 1 and no others, d =2 if every unit is connected to every other unit at distance of not more than 2, and so on. Note that this is a symmetric connection strategy. Wraparound on the grid is not used, so that the edge units have fewer connections than the inner units.

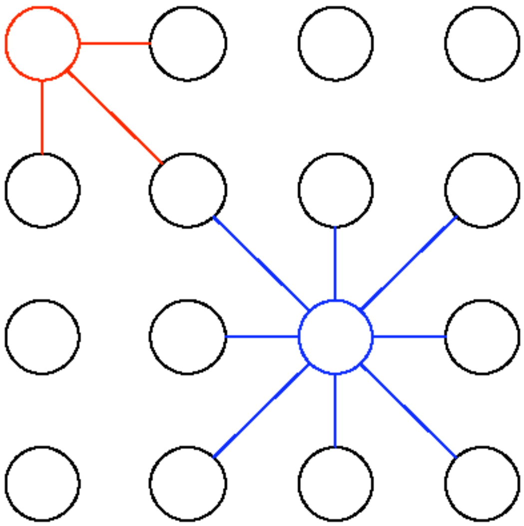

Figure 1: A small network in which neighbourhood connectivity has been established at a distance (d) of 1. Connections are shown for two neurons as an example.

4. Training

The networks are trained using a modification of the normal perceptron training rule that ensures symmetric weights. The algorithm is:

`Begin with zero weights`
`Repeat until all local fields are correct`
  `Set state of network to one of the` ξ^p
  `For each unit,` i`, in turn:`
    `Calculate` $h_i^p \xi_i^p$`. If this is less than T then change the weights to unit` i `according to:`

$$\forall j \neq i \quad w'_{ij} = w_{ij} + \frac{\xi_i^p \xi_j^p}{N} \quad w'_{ji} = w_{ji} + \frac{\xi_i^p \xi_j^p}{N}$$

Where ξ^p denotes the training patterns, and T is the learning threshold which here has the value of 0. All weights on removed connections are fixed at zero.

5. Capacity results for Perceptron Networks

A perceptron with N inputs can learn up to 2N random patterns, and as the correlation in the training set increases so does the capacity of the perceptron. Imposing symmetry on the weights, in a network of perceptrons, does not affect this maximum capacity [5]. Even with uncorrelated training sets capacity may be greater than 2N. This occurs when correlated subsets of the training set have correlated outputs [2]. So, for example, if pairs of the training set are correlated and have the same output then the training set is more likely to be learnable. Put simply, if similar patterns have the same label then a perceptron is more likely to be able to learn the classification. The increasing capacity is shown in Figure 2.

6. Training Sets Used

Two sets of training patterns, representing reasonably naturalistic images were created. All the generated patterns were 400 bits, 20 by 20 bitmap images. The *geometric* data uses solid geometric shapes placed at random within the 2-dimensional grid. Each image is formed by four shapes: triangles, squares or circles. The choice of shape is random and they may overlap but are clipped if they overrun an edge. The *character* data consists of 20 by 20 character bitmaps. Examples from these data sets are shown in Figure 3. The geometric data set is roughly unbiased (bias, the proportion of +1's is 0.52), whereas the character data has a bias of 0.2. Both sets have the desired characteristic of within pattern spatial continuity. This can be seen in *the mean local correlation* of the data set, for different neighbourhood sizes, Tables 1 and 2. For both data sets the correlation of individual bits with their neighbours decreases as that neighbourhood is increased.

Neighbourhood Size, d	Mean Local Correlation
1	0.89
2	0.83
3	0.77
4	0.72
5	0.68
Full Grid	0.5161

Table 1: Mean Local Correlation for the Geometric Data

Neighbourhood Size, d	Mean Local Correlation
1	0.87
2	0.78
3	0.74
4	0.71
5	0.70
Full Grid	0.68

Table 2: Mean Local Correlation for the Character Data

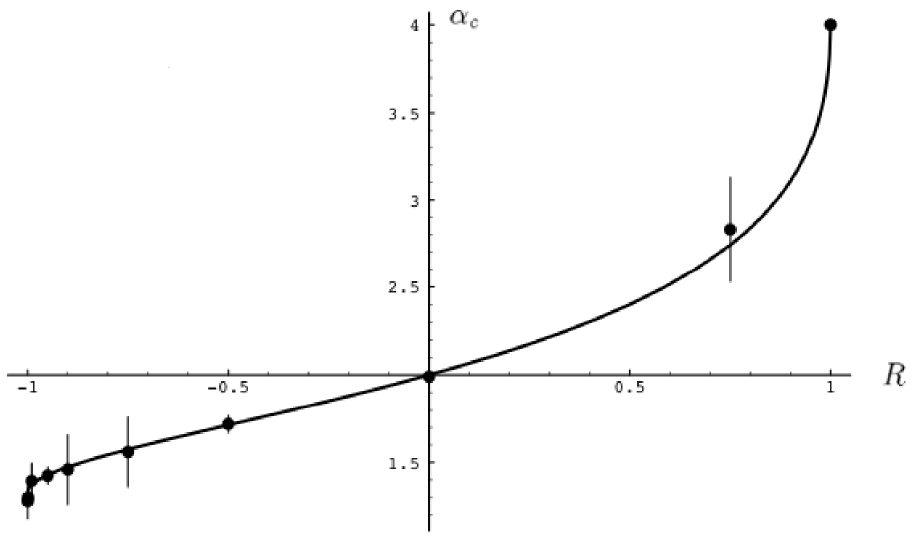

Figure 2: The capacity of a perceptron as the pair wise overlap of training patterns with the same output, R, is varied. With R = 0 the normal capacity of 2 is shown, but as R increases so does the capacity, approaching a limiting value of 4. Taken from Lopez et al [2].

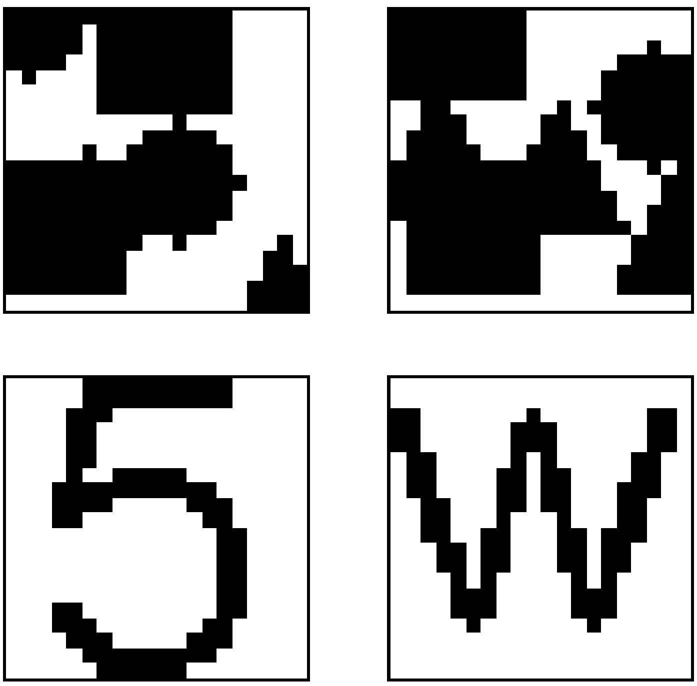

Figure 3: Example bitmaps from the geometric (above) and character training sets.

7. Results

The networks used here are highly diluted, for example in networks with d = 1 (units connected to those in an immediate square neighbourhood) each unit is connected to no more than 8 other units, and corner units are connected to only 3 other units. So with any training set it is very likely that some units will fail to train. We therefore report the number of units that fail to train at a given loading and expect this figure to be lower for networks with structured connectivity than for those with the same level of random connectivity. The network is trained for 1000 epochs, well beyond the number of epochs normally required for convergence at the kind of loadings we use here. The number of units that have failed to converge at this point is counted.

6.1 Geometric Data

Figure 4 shows how the number of neurons that fail to train increases with the loading on the network. For comparison the results for networks with equivalent levels of random connectivity are also shown in Figure 5. The randomly connected networks show the expected pattern. The capacity of such networks should be about 2N where N is the number of inputs for each perceptron. So that for the random network with a mean connection per neuron of just under 8 (equivalent to the d = 1 structured network) most units should fail with about 16 patterns – a loading of 16/400 or 0.04. However the structured network shows a very different pattern at this level of connectivity with a roughly linear increase in failed neurons as the loading increases, but no sudden jump in the failure rate. Remarkably the d = 3 network (each unit having roughly40 inputs) has a very low failure rate throughout the loading range – up to 100 patterns. The equivalent randomly connected network has more than half the units failing to train with 75 patterns in the training set.

6.2 Character Data

The character data is biased and so the capacity of an individual perceptron should here be higher than for the unbiased geometric data. However the dramatic benefit of structured local connectivity is even more apparent here, see Figures 6 and 7. Once again the d =3 network shows very low failure rate across all loadings and even the d = 2 network has less than 25% failures at the top loading

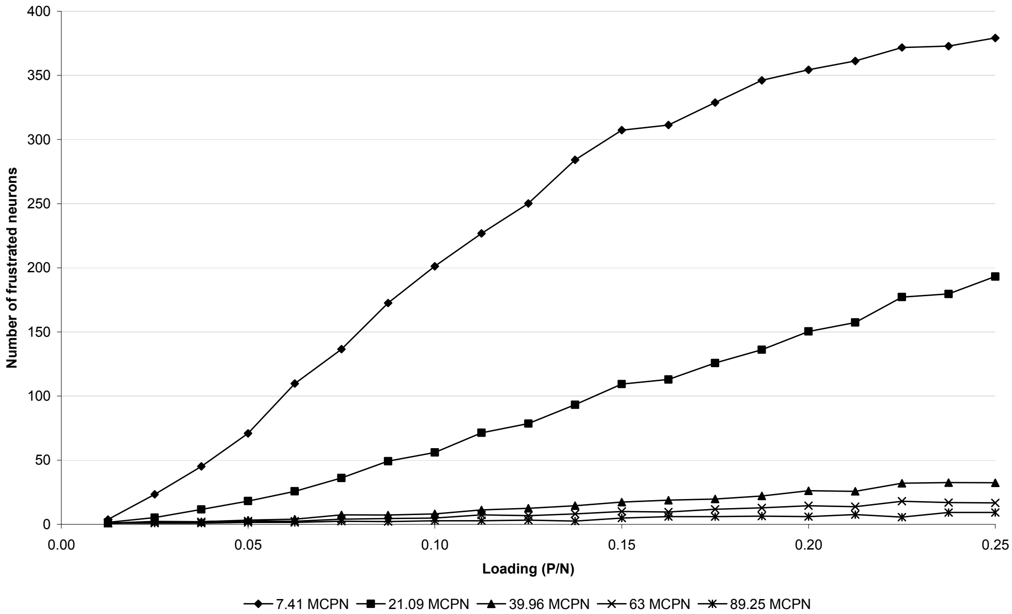

Figure 4 Failed neuron count against increasing pattern load for networks constructed with *structured* connectivity at levels of 7.41, 21.09, 39.96, 63, and 89.25 mean connections per neuron (corresponding to d = 1, d =2 etc) and trained using *geometric* data. Mean values over 5 runs at each loading are given.

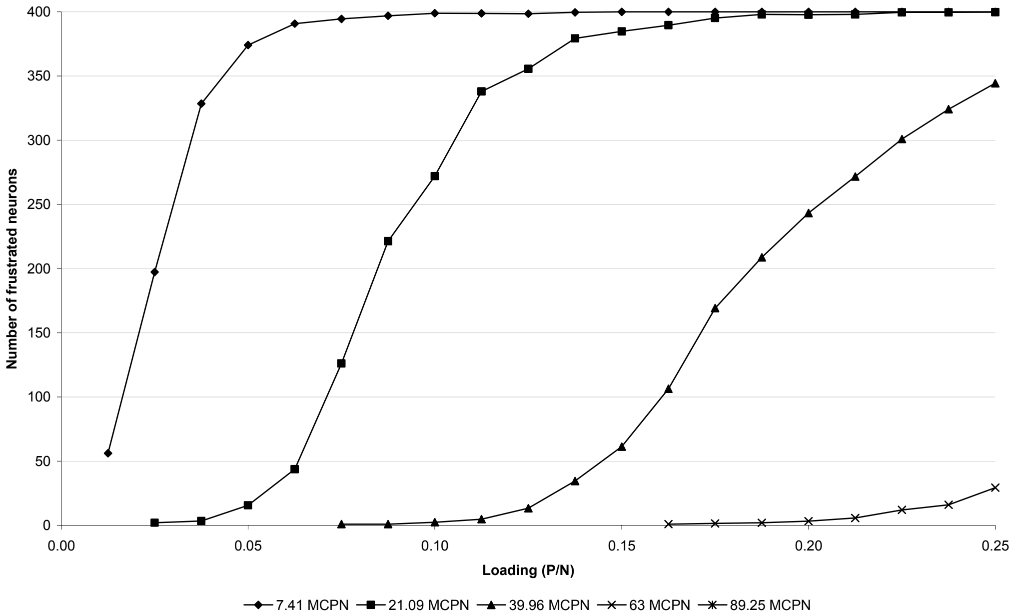

Figure 5 Failed neuron count against increasing pattern load for networks constructed with *random* connectivity trained using *geometric* data. Mean values over 5 runs.

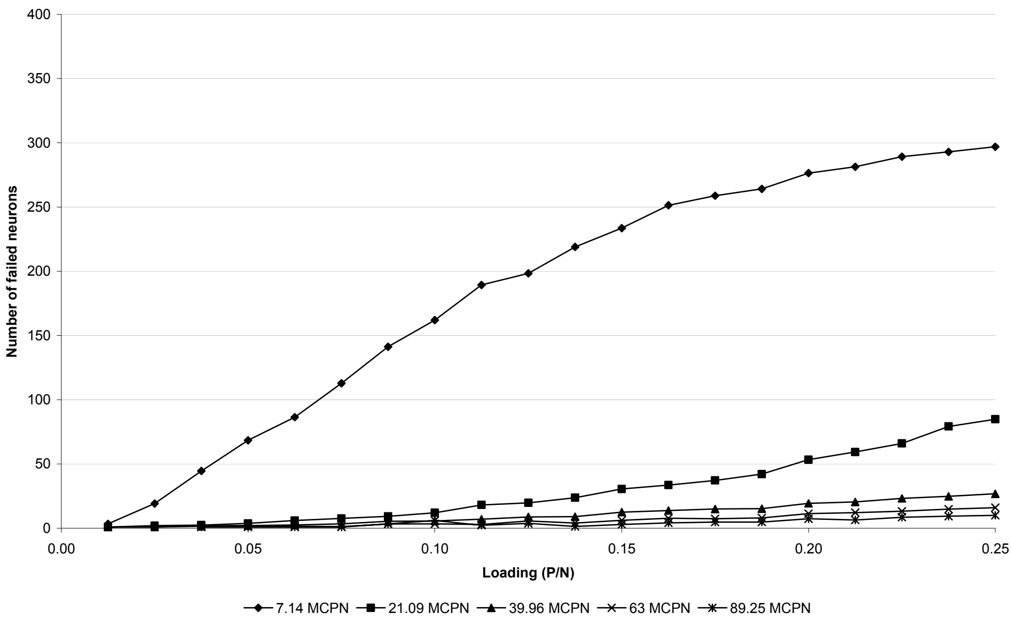

Figure 6: Failed neuron count against increasing pattern load for networks constructed with *structured* connectivity trained using *character* data. Mean values over 5 runs.

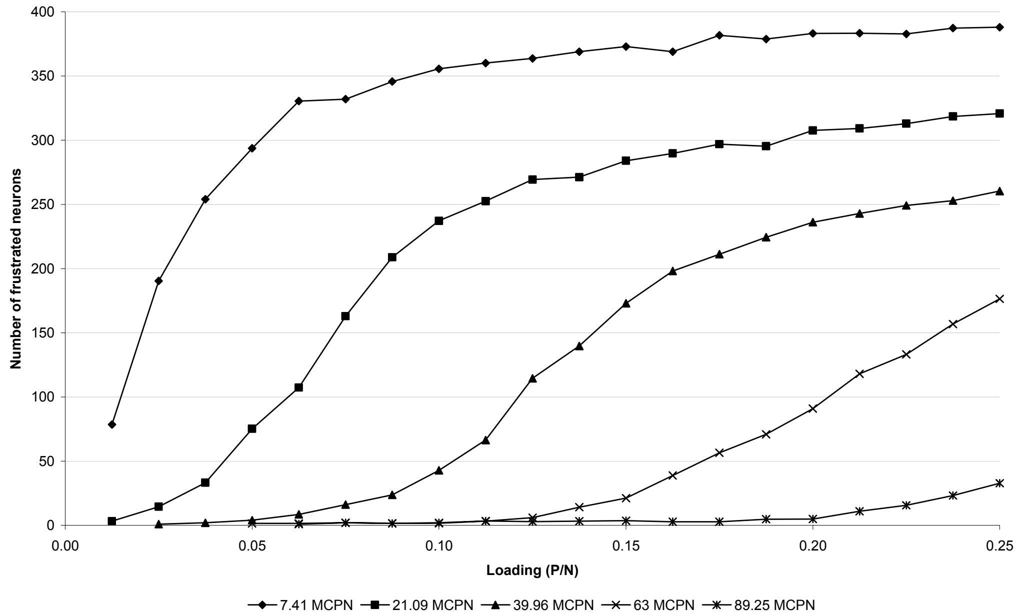

Figure 7: Failed neuron count against increasing pattern load for networks constructed with *random* connectivity trained using *character* data. Mean values over 5 runs.

8. Discussion

Much natural data shows spatial and/or temporal continuity and this aspect of the data could be exploited by an engineered or evolved system – artefactual or natural. Here we have shown that a simple associative memory model, a network of perceptrons, can exploit the local correlation present in simple bitmap images. The effective capacity (tolerating a small number of failed units) of the networks with structured connectivity is much better than those with an equivalent number of random connections.

A significant further benefit of the locally connected networks should also be noted. The mean connection length is obviously much lower in these networks. For example the d = 1 network has mean connection length of 1, whereas the randomly connected network in a 20 by 20 grid has a mean connection length of about 9.3. This has significance for any physical instantiation of these networks.

A possible criticism of the models used here is that a small number of neurons that fail to train has to be tolerated. However it is straightforward to overcome this limitation. The failed units may simply be given additional connectivity until they are able to learn their training set. As long as the number of failed units is small this will not have a significant impact on the overall connectivity pattern. The exploration of this idea is described in [6].

A more important problem with the idea of locally structured connectivity is that the pattern correction/completion behaviour of the network can be adversely affected. The recall process may get stuck in patterns with large subdomains of errors [7]. The subdomain may not have enough distal input to overcome its locally stable configuration. This issue may be addressed by introducing additional random connectivity and the results of doing this are promising [6, 8].

In summary this paper has described how structured local connectivity can increase the effective capacity of an associative memory and can produce large savings in the length of connections required.

References

1. Gardner, E., *The space of interactions in neural network models.* Journal of Physics A, 1988. **21**: p. 257-270.
2. Lopez, B., M. Schroder, and M. Opper, *Storage of correlated patterns in a perceptron.* Journal of Physics A: Mathematical and General, 1995. **28**: p. L447-L452.
3. Davey, N., R. Adams, and S.P. Hunt. *High Performance Associative Memory and Weight Dilution*. in *The 8th International Conference on Neural Information Processing (ICONIP 2001)*. 2001. Shanghai.
4. Hertz, J., A. Krogh, and R.G. Palmer, *Introduction to the Theory of Neural Computation*. 1991, Redwood City, CA: Addison-Wesley Publishing Company. 327.
5. Nardulli, G. and G. Pasquariello, *Domains of attraction of neural networks at finite temperature.* Journal of Physics A: Mathematical and General, 1991. **24**(5): p. 1103.
6. Turvey, S.P., *Analysing and Enhancing the Performance of Associative Memory Architectures*, in *Computer Science*. 2003, University of Hertfordshire: Hatfield.
7. Noest, A.J., *Domains in Neural Networks with Restricted-Range Interactions.* Physical Review Letters, 1989. **63**(16).
8. Bohland, J. and A. Minai *Efficient Associative Memory Using Small-World Architecture.* Neurocomputing, 2001. **38-40**: p. 489-496.