Validation in Distributed Representations

S H Srinivasan M Narasimha Murty
Department of Computer Science and Automation
Indian Institute of Science
Bangalore 560 012, INDIA

Abstract

In distributed representations, an object is represented as a pattern of activation over a number of units. Not all patterns of activity are valid in a given context. In case of the Hopfield model of content addressable memory only the patterns of activation corresponding to the stored memories are valid. The problem of validating a given pattern of activity is called the validity problem. In this paper it is shown that using complex activations, it is possible to solve the validity problem. The proposed model can also act as a content addressable memory though its dynamics is intractable because of asymmetric connections.

There are two major representation schemes in connectionist networks: local and distributed. In localist schemes each concept, feature, or goal is represented by a single unit. Activity of the unit stands for the presence of the concept (or feature or goal). In distributed representations, a concept is represented as a pattern of activity over a number of units and each unit participates in the representation of many concepts. In this case we say that units stand for microfeatures. Microfeatures may have no direct interpretation in terms of the environment; these are used to capture the underlying similarities of the objects represented.

Distributed schemes have a number of advantages over localist schemes: fault tolerance, generalization, learnability etc [2]. But there are many problems with distributed representations. The major problems that have been addressed in the literature are: binding problem, cross talk, optimal number of units assigned to a concept etc. One of the issues that is missed in the studies on distributed representations is the validity of a given pattern of activity.

Before commenting on the nature of the problem, we will have to make some further assumptions about the mode of computation. One of the dominant modes of computation in connectionist networks is relaxation and often the problem to be solved is posed as a *completion* or a *constraint satisfaction* task [3]. In this setting, suppose we start with an incomplete pattern and the network ends up in a stable equilibrium state after computation. Is there a mechanism by which we can ensure the validity of the final state? In other words, how can we be sure that the final state is one of the stored states? We call this the validity problem. This is one way of formalizing the notion of "knowing what one knows". In other words, after recalling something based on partial input cues, the system should have a way of checking whether it has come across that before. There have been some solutions to this problem. In Kanerva's model of memory [5], the recall process diverges if the distance between the input cue and each one of the stored items is greater than a critical value; otherwise it converges to the correct pattern. Thus divergence on a given cue is interpreted as ignorance. In Moopenn et al. [6], the spurious memories are mapped to one of the nearest memory points so that there is a path form every spurious memory to a genuine one. But this needs the prior knowledge of which memories are spurious attractors. There is another solution based on localist representation. Carpenter and Grossberg [1] discuss the "stability–plasticity" dilemma. The problem is to recognize when to stop learning so that the previously learned information is not lost/corrupted. This involves the ability to discriminate between previously encountered inputs and new inputs. The solution is to assign one unit for a class of similar input patterns. When an input does not "trigger" any of these units, it is classified as a novel input. In this paper we propose a solution to the validity problem in distributed representations using complex activations.

The model considered here is based on the Hopfield model [4]. We modify the real Hopfield model by having an output function, i.e., the output of a unit is not its activation as in the usual model but some function of it.[1] We make this a *complex* function. This will make some of the equilibrium points complex. We

[1] In the Hopfield model, the output function can be taken to be the identity function

CH 3065-0/91/0000-0036 $1.00 © IEEE

define genuine memories as the real stable equilibrium points. If we have a way of making the memories to be stored stable (i.e., solve the learning problem), we can expect that on other inputs, the system will reach a complex equilibrium point. By probabilistic analysis and actual simulations, we show that this indeed is the case. Hence this gives a method to distinguish genuine memories from spurious ones. We will show that the Hebb rule can be used for learning. Because of the use of output function, Hebb rule leads to asymmetric synapses. Since there is no general way to analyze the dynamics of asymmetric nets we do only equilibrium analysis. Simulation studies have indicated that the model has error correcting properties and hence it can be used as a CAM even though its dynamics is yet to be fully understood.

Finally we consider a more complicated learning algorithm which can be used to increase the capacity of the model.

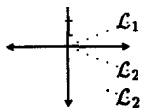
1 The model

$$\Theta(\rho e^{j\theta}) = \begin{cases} 1, & \frac{-\pi}{8} \le \theta < \frac{\pi}{8} \\ e^{j\frac{\pi}{4}}, & \frac{\pi}{8} \le \theta < \frac{3\pi}{8} \\ j, & \frac{3\pi}{8} \le \theta < \frac{5\pi}{8} \\ e^{j3\frac{\pi}{4}}, & \frac{5\pi}{8} \le \theta < \frac{7\pi}{8} \\ -1, & \frac{7\pi}{8} \le \theta < \frac{9\pi}{8} \\ e^{j\frac{5\pi}{4}}, & \frac{9\pi}{8} \le \theta < \frac{11\pi}{8} \\ -j, & \frac{11\pi}{8} \le \theta < \frac{13\pi}{8} \\ e^{-j\frac{\pi}{4}}, & \frac{-3\pi}{8} \le \theta < \frac{-\pi}{8} \end{cases}$$

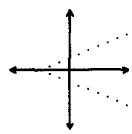
$-1\}^N$.

$$w_{ik} = \sum_{\mu=1}^{\mu=M} S_i^\mu \overline{f(S_k^\mu)}$$

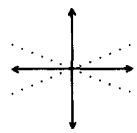
where $\overline{\ }$ indicates complex conjugation operation. Since $f(\cdot)$ is a complex function, w_{ik} are complex even for real memories. Also the weights are asymmetric for most choices of $f(\cdot)$.



(A) Receptive field for $+1$ in the complex plane. The angle is 45 degrees.



(B) If the input is one of the stored patterns, noise should be in this region for $+1$ to be valid. Displacement to the left is $(N-1)$.



(C) For an arbitrary input pattern, noise in this region gives rise to real output.

Figure 1: Receptive fields in the complex plane for various cases.

2 Statistical analysis

In this section we define and calculate following two measures of performance.

1 probability of a valid memory being recognized as a spurious memory, called the **fidelity** probability.

2 probability of a spurious memory being recognized as a valid memory, called the **selectivity** probability. (For the memory to have good selectivity the selectivity probability should be small.)

If the input to a unit is +1, then the net input to the unit should be in the region shown in the figure (1A) for the activation in the next iteration to be +1. If we decompose the net input to signal and noise then the noise should be in the region shown in figure (1B) for this to happen; in this figure, the region is shifted to the left by the magnitude of the signal term. The fidelity probability is the probability that the noise is in the region shown in figure (1B). For an unknown input pattern, the signal term is zero. If the noise lies in the region of figure(1C) the output is real. For a random input, the selectivity probability is half the probability that the noise lies in the above mentioned region. We calculate these probabilities with the following assumptions.

Let the bits in a given pattern be random with correlation coefficient r and the patterns themselves be independent.

Let S^ν be the input to the network and let $f(S^\nu)$ denote the vector obtained by applying $f(\cdot)$ to each component of S^ν. Then the net input at unit i is

$$\begin{aligned} net_i^\nu &= \sum_{k=1,k\neq i}^{k=N} w_{ik} f(S_k^\nu) \\ &= S_i^\nu \|f(S^\nu)\|^2 + \sum_{\mu\neq\nu} (f(S^\nu), f(S^\mu)) S_i^\mu \end{aligned} \tag{1}$$

The first term is the signal at unit i and the second term is the noise. It should be noted that the noise term is complex. Let P_i^ν and Q_i^ν be the real and imaginary parts of the noise (at the input of unit i with pattern ν as input). Let $\mu_{P_i}, \mu_{Q_i}, \sigma^2_{P_i}$, and $\sigma^2_{Q_i}$ be the means and variances of these two terms and p the correlation coefficient. (The superscript ν is omitted for simplicity.)

If the input is not one of the stored patterns, then we have only noise and let μ_P, μ_Q, σ_P^2, and σ_Q^2 be the respective means and variances and ρ' being the correlation coefficient.

Since noise is the sum of $M-1$ (or M in the second case) independent random variables, we can do a normal approximation for P_i and Q_i. For a pattern S to be an equilibrium *point* of the network, the angle between S_i and net_i should be between $\frac{-\pi}{8}$ and $\frac{\pi}{8}$ for all the units i. Let m = $\tan(\pi/8)$ and let $\mathcal{L}_1$ and $\mathcal{L}_2$ be the lines $y = mx$ and $y = -mx$ (figure 1A). Let $S_i = 1$. Then for the output to be 1, neti should lie in the first quadrant in the region enclosed by $\mathcal{L}_1$ and $\mathcal{L}_2$.

Let the input pattern S be one of the stored vectors and let $S_i = 1$. Then the signal strength at i, denoted μ, is $N-1$ and $net_i = \mu + \text{noise}$. For the output to be 1, the noise should lie in the region as indicated in

the above paragraph but shifted to the left by μ. (see figure 1B.) In other words, the noise should be in the region enclosed by the lines y = $m(x+\mu)$ and $y = -m(x+\mu)$ for $S_i = 1$. The probability of this is

$$\int_{x=-\mu}^{x=\infty} \int_{y=-m(x+\mu)}^{y=m(x+\mu)} f(x,y)dydx$$

where $f(x,y)$ is the bivariate gaussian density function with appropriate parameter values supplied depending on the context. For any pattern which is not one of the stored patterns, the signal strength is 0 and the probability of a spurious memory giving rise to a $+1$ or -1 is

$$2\int_{x=0}^{x=\infty} \int_{y=-mx}^{y=mx} f(x,y)dy\,dx$$

There is no closed form expression for these probabilities and they have to be evaluated numerically for different parameter values. With the proper tuning of r and M for a given N, we can get good selectivity.

2.1 Probability calculations

The probability calculations were undertaken for five different values of the output function which are listed below.[2]

$f_1(1) = e^{\frac{j\pi}{4}}$ and $f_1(-1) = e^{\frac{j\pi}{4}}$; $f_2(1) = 1$ and $f_2(-1) = e^{\frac{j4\pi}{3}}$

$f_3(1) = 1$ and $f_3(-1) = e^{\frac{j3\pi}{2}}$; $f_4(1) = 1$ and $f_4(-1) = e^{\frac{j5\pi}{3}}$

$f_5(1) = 1$ and $f_5(-1) = e^{\frac{j175\pi}{180}}$

For N = 125 and for each of these functions, the above probabilities were calculated for M = 2, 4, 6, 8, 10, 12 and $r = -0.004, -0.002, 0.0, 0.1,$ and, 0.2. The results of the calculations are partly summarized in the following graph.

- fidelity and selectivity probabilities *vs.* angle for for various loading values (M) for $r = 0$. (see figure (2) for fidelity and figure (3) for selectivity.)

3 Simulation studies

The network was simulated for N = 125 for various values of M for randomly generated patterns. The following performance measures were studied for each value of M.

- the number of patterns recalled correctly. For each angle, the maximum number of patterns recalled correctly is shown in figure **(4)**.
- out of 1000 random real patterns, the number of patterns that gave rise to real patterns after one synchronous update. Figure (5) gives the results.
- out of 1000 random real patterns, the number of patterns that were the equilibrium points of the network i.e., remained unchanged after one synchronous update. In all the cases, this was 0.

[2] For equilibrium analysis of the model for real inputs, the complete definition of $f(\cdot)$ is not necessary. Hence we have given the definitions at ± 1 only. Also for equilibrium analysis, only the angle between $f(+1)$ and $f(-1)$ matters. Hence the results are summarized in terms of angle in figures 2,3, and 4

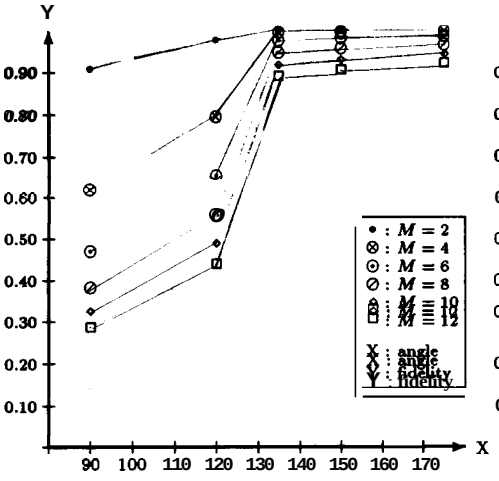


Figure 2: fidelity probabilities for $r = 0$

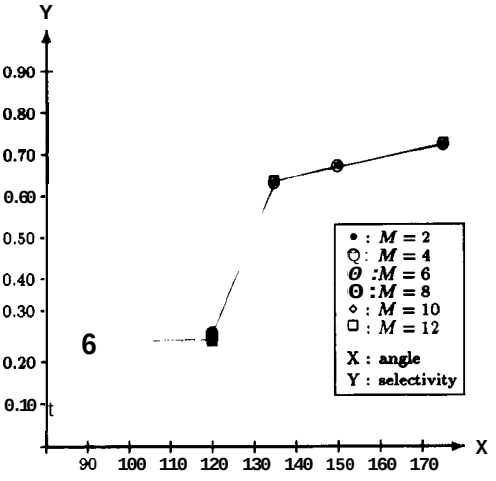
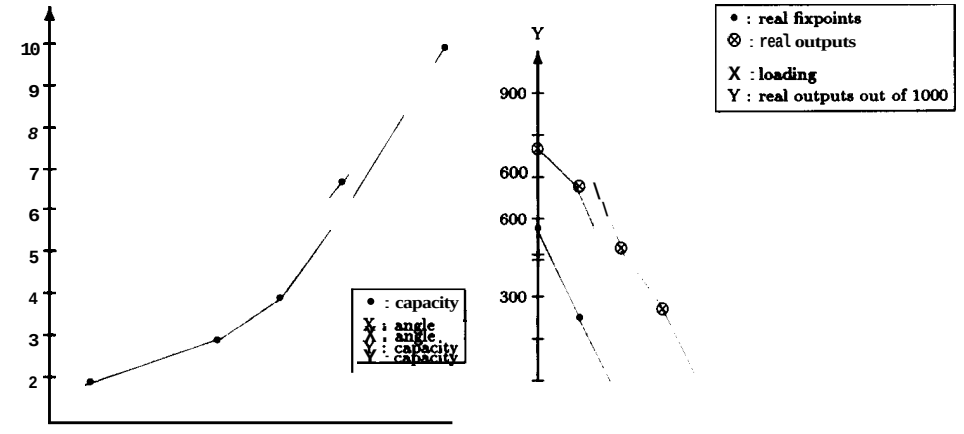


Figure 3: selectivity probabilities for $r = 0$



4 An analytic learning algorithm

It can be seen from the above sections that the performance of the memory is poor for the output function $f_5(\cdot)$. We tried the following algorithm in this case. As already mentioned, for a pattern S to be an equilibrium point of the network, the angle between S_i and $net_i = P_i + jQ_i$ should be in the range $(-\pi/8, \pi/8]$. Hence we pose the learning problem as that of maximizing the cosine of the angle between the activation (S_i^μ) and the net input (net_i^μ) for all units (i) and for all patterns (μ). Thus the function to be maximized is

$$\sum_\mu \sum_i \frac{S_i^\mu \Re[net_i^\mu]}{|net_i^\mu|}$$

This function was maximized for $N = 25$ and $r = 0$ (i.e., randomly generated patterns) for various values of M. In each case, the number of patterns, out of 1000 randomly generated patterns, that were the equilibrium points of the network were also noted. The model could retrieve 11 patterns before making error on recall; the number of spurious equilibrium points were 0 in all the cases.

5 Discussion

It can be seen from figure (4) that the capacity of the memory depends greatly on the output function used. We explain this based on the notion of "orthogonality in the range". Considering equation 1, we see that if for a given set of input patterns, the function $f(\cdot)$ is such that the transformed patterns are mutually orthogonal then the recall is perfect. It can be shown that for the function $f_1(\cdot)$ none of the pairs of vectors in the input space are orthogonal; and for $f_2(\cdot)$, exactly three vectors in the input space are mutually orthogonal (but there are many such triads.). The number of vectors in the input space that give rise to orthogonal vectors increases for other functions and it can be proved that if the output function preserves orthogonality, then it is just a "constant multiple" function and hence all real input vectors give rise to real output vectors and becoming equivalent to the real Hopfield model. Hence for good capacity, the output function should be as orthogonality preserving as possible.

As already mentioned, the synapses are complex and asymmetric. Conjugate symmetric models possess a energy function and it is possible to formulate the dynamics the energy reduces on each update. But in the present model the dynamics cannot be easily studied because of asymmetry. But small scale simulations indicate that the model does possess error correcting property. For $N = 125$ and $M = 5$ with the output function having an angle 150°, error correction of up to 15 bits could be obtained on all the 5 memories. Further studies are required on the complete dynamic properties of the model.

References

[1] G. A. Carpenter and S. Grossberg. *Computer vision, graphics, and image processing,* 37(1):54, 1987.

[2] G. E. Hinton, J. L. McClelland, and D. E. Rumelhart. Distributed representations. In [7].

[3] G. E. Hinton and T. J. Sejnowski. Learning and relearning in boltzmann machines. In [7]

[4] J. J. Hopfield. *Proc Natl Acad Sci USA,* 79:2554, 1982.

[5] P. Kanerva. *Sparse distributed memory.* MIT Press, Mass, 1988

[6] A. Moopenn, J. Lambe, and A. P. Thakoor. IEEE *Trans SMC,* 17(2):325, 1987.

[7] D. E. Rumelhart and J. L. McClelland, editors. *Parallel distributed processing: Explorations into the microstructure of cognition (Vol I).* MIT Press, Cambridge, MA, 1986.