Convergence time characteristics of an associative memory for natural language processing

Nigel Collier*
Department of Language Engineering
UMIST
PO Box 88, Manchester M60 1QD, United Kingdom
E-mail: nigelc@ccl.umist.ac.uk

Abstract

We take a new look at one of the fundamental properties of discrete time associative memory and show how it can be adapted for natural language processing (NLP). Many tasks in NLP could benefit from such associative functionality particularly those which are traditionally regarded as being context driven such as word sense disambiguation.

The results describe the typical time to convergence of a Hopfield network when trained on patterns representing sentences from a large corpus. Through numerical simulation we estimate the time order of convergence and compare this to previous findings for randomly generated, unbiased and uncorrected patterns.

1 Introduction

Linear associative memories have been well studied in fields such as statistical physics [Bruce *et* a/., 1987][Amit, 1989][Tanaka and Yamada, 1993], and biophysics [Hopfield, 1982; 1984] for their ability to store and recall a set of patterns robustly, even when the patterns to be recalled can only be presented to the networks in a highly corrupted form. It is this robustness in the face of noise and an ability to recall stored patterns based on associations that makes these memories so potentially valuable for natural language processing (NLP), in particular tasks which have traditionally been regarded as context driven such as word sense disambiguation.

In this paper we take a new look at one of the fundamental properties of a discrete time, discrete state Hopfield network. We explore its time to convergence which has previously been investigated for the storage and recall of randomly generated patterns. Clearly, any patterns derived from natural language sentences will not be random, but rather will reflect a complex underlying linguistic distribution. Evidence from using *random,* correlated and biased patterns suggests that the network's storage limit will change. We want to investigate how other properties of the network might also be effected.

This will be a worthwhile undertaking if we can adapt associative memory for NLP and decouple the time order of complexity from the complexity of the linguistic structures in the stored data we have gone some way to forming a more efficient device for language processing than is currently available.

•Currently Visiting Fellow at Research & Development Center, Toshiba Corporation, 1 Komukai Toshiba-cho, Kawasaki 210 Japan. E-mail: nigel@eel.rdc.toshiba.co.jp

2 The Model

The model we are looking at in this paper is a simple one-layered discrete state Hopfield network [Hopfield, 1982]. The network is essentially autoassociative in nature, but can be coupled into layers (e.g. [Tanaka and Yamada, 1993]) to simulate more complex functions, or adapted into a Boltzmann machine for better generalisation performance. The results presented in this paper should act as a basis for trying to understand these more complex systems when presented with non-random biased pattern sets.

The Hopfield network consists of a set of N units V_i ($i = 1,.., N$) each of which may take either of two values, 1 or 0. The activation of each unit is defined by a simple threshold function

$$V_i = \begin{cases} 0 & if \quad H_i < U_i \\ 1 & if \quad H_i \geq U_i \end{cases} \tag{1}$$

where U represents a set of thresholds and H is defined as

$$H_i = \sum_{j,j \neq i}^{N} T_{ij} V_j + I_i \tag{2}$$

Inputs to unit i come internally from all the other units in the network and externally from /, which is a constant input in our network set at the start of processing.

Processing takes place by randomly and asynchronously updating units according to Eqn. (1) until Eqn. (3) converges, indicating that a stable state has been reached.

$$E(\{V\}) = -1/2 \sum_{i,j}^{N} T_{ij} V_i V_j + \sum_{i}^{N} U_i V_i - \sum_{i}^{N} I_i V_i \quad (3)$$

The knowledge which the network has about the patterns to be stored is contained in the symmetric matrix of weights T. Storage of the set of patterns $\xi^{(\mu)}$ $(\mu = 1, .., n)$ is according to the Hebb rule

$$T_{ij} = 1/N \sum_{\mu=1}^{n} \xi_i^{\mu} \xi_j^{\mu} \quad (4)$$

which embeds the patterns as stable attractors in the network dynamics. For storage to take place two of the necessary conditions are that T should be symmetrical, thus $T_{ij} = T_{ji}$, and that units should have zero self-interaction, so $T_{ii} = 0$.

From early numerical studies by [Hopfield, 1982] as well as analytic studies by [Amit *et al.*, 1987a; 1987b] we know that the number of fixed points in the network's dynamics which can be associated with the patterns to be stored is limited to the storage ratio

$$\alpha = n/N \quad (5)$$

and that a critical value exists for the storage of randomly generated patterns with zero bias at $\alpha_c = 0.14$. This has been observed (e.g. [Amit *et al.*, 1987a]) to rise to 0.18 for randomly generated biased patterns. After the critical storage value is exceeded it is known that storage and recall degrade discontinuously in the large N limit [Gardner, 1986], but continuously for small N [Grensing *et al.*, 1987] up to some point α_0 $(\alpha_c < \alpha_0)$.

In [Collier, 1996] it was shown that evidence existed for the shift in α_c from 0.14 to 0.18 to be extended to non-random patterns which were biased and correlated. Clearly the storage limits of the network are one fundamental property of this associative memory. A second fundamental property is the convergence time which we define as the time it takes to retrieve a pattern from a corrupted version of that same pattern.

This is one of Koiran *et a*l's [Koiran, 1994] 'interesting questions' for future investigation, and any evidence for an upper bound on the convergence time should be interesting across a number of fields.

3 Previous Work

Given the importance of the class of associative network models it is surprising to find that most emphasis has been placed on the storage capacity and less notice has been taken of the convergence time. Consequently there seem to be many unanswered questions which merit investigation.

Previous analysis of the time order complexity of the discrete Hopfield network for randomly generated patterns has most recently ([Tanaka and Yamada, 1993]) given us an upper bound for successful recall of *0(log(N)),* where *N* is the number of units in the network. There are at least two remaining unknowns: what is the time order complexity for a failed recall, and is recall time the same for correlated, biased, non-random patterns? A secondary question relates to the factor of scale in Tanaka and Yamada's simulations which were carried out using a range of TV from 100 to 10000. It would be nice to confirm their results over a range of networks to see if there are any finite size effects, as well as for those effects which come from different pattern types.

Let us first define what we mean by recti// *time.* A recall will be called successful if the network converges to a stable attractor which has a vector representation V^{μ} which is the same as a pattern in the training set $\xi_i^{(\mu)}$ for all bits in the pattern. A partial recall or a failed recall will be recorded otherwise. Although other researchers have used measures such as mean cpu. time to convergence, we have defined recall time to be the number of bit flips required for the network to settle into a stable state. This should provide a hardware independent measure.

In the case of the single layer Hopfield network convergence to a single stable state is guaranteed, whereas in the case of multi-layered networks we would expect the network to settle into a limit cycle of length 1 or 2. In this paper we will confine our investigation to the single layered network, but we are interested in measuring what may be termed the 'transient period' from initial network state to stability.

The great advantage of knowing that a failed recall will take much longer than a successful recall is of course a saving in processing time and a reduction in pattern recognition error. We can simply abandon a recall attempt as a failure if its processing time exceeds our expectation for a network of a given size.

In an early paper [Bruce *et a*l., 1987] report that flow times to convergence were found to depend on a number of factors the most important of which was the storage ratio *a.* Of secondary importance was the system size *N* and the particular updating schedule.

4 Training

We train the network using the localist Hebb rule of Eqn. (4) to obtain the weight matrix *T* which encodes the knowledge about the patterns to be stored. Since artificial pattern construction would not give us the rich characteristics of natural language, the training patterns are generated from a corpus of 'real' texts. In our simulations this was a corpus of English and Japanese texts

taken from newspaper editorials. Each of the English content words has been sense tagged with a unique label denoting its sense. This was undertaken by finding the Japanese lexical translation of the English words in the corresponding Japanese sentence in the corpus. The resulting corpus can be considered to be sense tagged.

From this sense tagged corpus of 16000 sentences we extracted a number of subcorpora and used these as our training sets. Sentence-to-pattern construction takes place as follows: a lexicon of length N is generated for each corpus from the constituent words giving each word sense a unique index. For each sentence we generate a bit vector of length N where an occurrence of a word in the sentence is shown by the bit being set to 1 at the position the word occupies in the lexicon. The set of n patterns thus generated, $\xi^{(\mu)}$, is given to Eqn. (4) to generate the weight matrix T. We see that each node in the network therefore corresponds to one word in the corpus lexicon - a localist representation. Looked at another way we are generating a high dimensional symbolic feature space in which the words which constitute the sentences are the features.

This encoding, while not optimal in terms of storage capacity, does allow us to compare our results with a wealth of results from fields such as statistical physics. In terms of representational adequacy for NLP, we see that the encoding does not capture either constituency or sequential relations between words. What it does capture is simple cooccurrence in sentences between word pairs which is a minimum required in our view for the modelling of contextual knowledge of language. If this is found to be inadequate the representation could be bought closer to the ideal language model by, for example, having a non-symmetrical weight matrix in which T_{ij} would be interpreted as "i followed by j" and hence capturing sequence knowledge.

All of the basic parameters which define the matrices generated from each of the subcorpora TR1 to TR7 are shown in Table 1. We see that N ranges from 260 to 7045, which should show us any finite size effects and covers substantially the same range used in the simulations of [Tanaka and Yamada, 1993]. We also see that the minimum value for the storage ratio a in TR1 is above the critical level α_c of 0.14 predicted for unbiased random patterns. It was seen in [Collier, 1996] that this is not necessarily a problem because N is finite, thus giving us continuous degradation in storage rather than the catastrophic discontinuous effect predicted for infinite N. Moreover, we also know that for biased systems of training patterns the value of the critical storage ratio a_c has been observed to shift from 0.14 upwards to 0.18.

The bias constant a for each corpus of patterns is calculated from the probability that a bit in a pattern will be set to 1 as follows

$$Prob(\xi_i^{(\mu)} = 1) = \frac{\sum_\mu^n \sum_i^N \xi_i^\mu}{n \times N} \tag{6}$$

and then bias becomes

$$a = 2Prob(\xi_i^{(\mu)} = 1) - 1 \tag{7}$$

Giving us a value $-1 \leq a \leq +1$. In fact, since variation between word distributions in the training corpus is quite large it is probably erroneous to try and characterize patterns simply in terms of bias since individual distributions differ widely from this norm. In this respect our work differs from other studies of randomly biased patterns such as [Treves and Amit, 1988][Perez-Vicente and Amit, 1989]. For this reason we also looked at several other macro-statistics which characterise the pattern sets. These are now described and values are given in Table 1.

We calculated a simple measure of pattern correlation between patterns from Eqn. (7) as

$$\langle\langle \xi^\mu \xi^\nu \rangle\rangle \equiv a^2 \tag{8}$$

$\bar{c}^{(\mu)}$ is the mean connectivity of the weight matrix T which is generated from training using the localist Hebb rule of Eqn. (4). $s^{(\mu)}$ is the fraction of the total storage space N^2 which is actually used to store the word association statistics encoded through training. As the subcorpora become larger the fraction of space used becomes progressively smaller. We note that it shows natural language patterns generate very sparse matrices when trained with Eqn. (4).

Usually, storage of patterns which are highly biased in such an associative memory as we have described above would be difficult because of the influence of noise which leads to the destabilizing of attractor states. To compensate for this noise we have introduced a global inhibitor so that where $T_{ij} = 0$ according to Eqn. (4) we replace this with a small negative constant designed to provide the extra amount of inhibition to ensure that nominated states are attractors. The value for the inhibitor which we found best was $10/N$ which may well correspond to the mean number of content words in a sentence and hence the mean number of bits set to 1 in a training pattern ξ^μ.

Table 1 confirms that the patterns we are looking at are biased, giving us low activation networks with sparse coding. The probability of any bit being set to 1 in a training pattern is $Pr(\xi_i^{(\mu)} = 1)$ which starts at 0.037 and decreases almost to zero in TR7 giving us a very strong bias in a throughout. We see here the influence of storing patterns representing natural language sentences because in any sentence pattern ξ^μ, the mean number of content words remains fixed at about 10 irrespective of the size of the corpus. The consequence is that bias, a,

is close to its limit of —1, whereas in an unbiased system we would expect it to be close to 0.

	TR1	TR2	TR3	TR4	TR5	TR6	TR7
N	260	673	921	1357	4131	4935	7045
n	41	121	167	273	1412	2000	4000
α	0.16	0.18	0.18	0.20	0.34	0.41	0.57
$-a$	0.93	0.97	0.98	0.98	0.99	0.99	0.99
a^2	0.86	0.95	0.96	0.97	0.99	0.99	0.99
$\bar{c}^{(\mu)}$	16.0	18.8	24.9	28.9	44.5	48.8	63.4
$s^{(\mu)}$	0.06	0.03	0.03	0.02	0.01	0.01	0.01

Table 1: Subcorpora Characteristics

5 Simulations

In our simulations we follow the work of [Amit, 1989] and try to relate the mean perfect recall of patterns from the training set to the mean convergence rate of the network. To make the results more interesting we randomly corrupt the test patterns using a level of noise m_0 which ranges from 0.0 to 1.0 in increments of 0.1, so that the probability of a bit in the test pattern being flipped from 1 to 0 is equal to the noise level. Noise was applied only to the 1 bits as most of the information is contained here. We can expect convergence times to increase with mo as the number of bits which need to be reset increase.

To calculate the fraction of error in recall we have looked at the distance between the actual stable state chosen by the network $\tilde{V}^{\mu}$ for a pattern $<\xi^{\mu}$, and the nominated stable state V^{μ} in autoassociation tests. We can measure the fractional Hamming distance between $\tilde{V}^{\mu}$ and V^{μ} as

$$D = 1/N \sum_{i}^{N} |\tilde{V}_i^{\mu} - V_i^{\mu}|$$

The mean error free fraction $\overline{F}_I$ is defined as

$$\overline{F}_I \equiv Prob(D = 0) \qquad (10)$$

In order to calculate *Fj* we therefore measure the fraction of error free recalls for an ensamble of test patterns taken from the training set $\xi^{(\mu)}$ over a large number of trials.

Convergence times were estimated from the mean number of bit flips required for the network to reach a stable state. The number of bit flips is the number of network updates which result in an output unit V_i changing state from 0 to 1 or from 1 to 0. This quantity was also calculated over the same trials as noise m_0 was increased from 0.0 to 1.0 in 0.1 increments.

F_1 and mean convergence times were estimated numerically for 50 test patterns taken from the training sets TR1 to TR7 (shown in Table 1) over 10 trials. This was then repeated for each level of noise mo- Unlike previous simulations we cannot easily 'manufacture' training patterns, so we are limited to the values of *a* which the corpora give us.

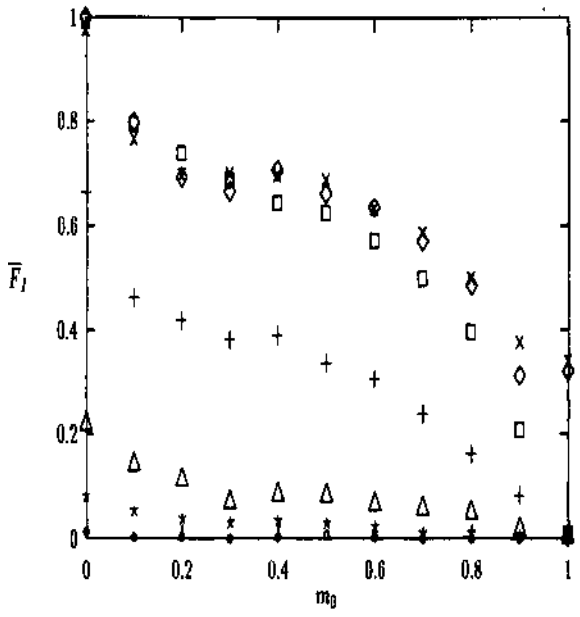

Figure 1: Mean fraction of patterns recalled with no error *Fj* against initial pattern noise mo for 1 bits. **TR1:×, TR2:◇, TR3:□, TR4:+, TR5:△, TR6:⋆, TR7:*o*.**

Results for the mean error free fraction *Fj* given in Figure 1 show us that pattern sets with storage ratios less than 0.20 have good recall while those above this level show continuous degrading recall. As [Collier, 1996] reported, this leads us to put the critical storage ratio a_c somewhere around 0.18. In this respect TR1 to TR3 have similar perforance and TR4 to TR7 show degrading performance as *a* increases.

Looking now at mean spin flip to convergence results in Figure 2 we see that the training sets which performed well in recall (TRI to TR4) have similar convergence times. Clearly TR5 to TR7 take much longer on average to converge to a stable state. If we take a closer look at TRI to TR4 in Figure 3 we see that in fact TRI to TR3 have almost the same convergence times and TR4 is slightly greater, confirming again that TR4 has degraded performance.

This result seems to agree with [Bruce *et* a/., 1987]'s conclusion that a is a factor in convergence times. We can say that our simulations have shown that where $\alpha \leq \alpha_c$ then convergence times will be governed by the amount of induced noise, mo, and the number of 1 bits

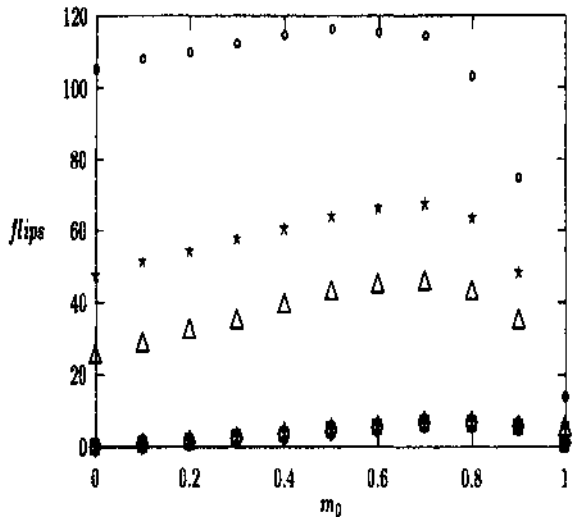

Figure 2: Mean number of bit flips to network convergence *flips* against initial pattern noise m_0. TR1:x, TR2:◇, TR3:□, TR4:+, TR5:△, TR6:⋆, TR7:*o*.

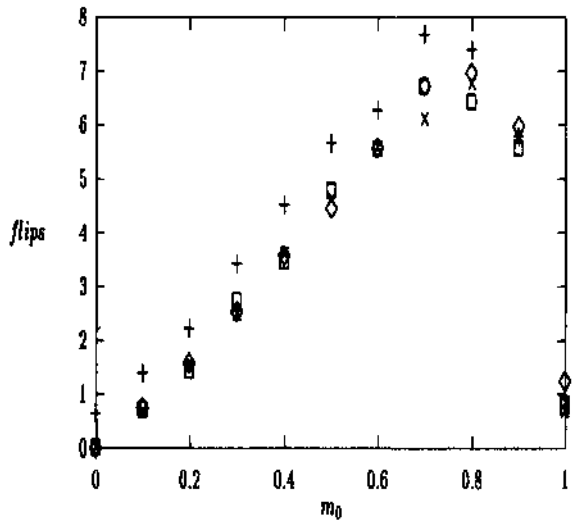

Figure 3: Mean number of bit flips to network convergence *flips* against initial pattern noise m_0. TR1:x, TR2:◇, TR3:□, TR4:+.

in the pattern, shown by $(1-|a|)N$. When $\alpha > \alpha_c$ convergence will be much greater than this, as observed by [Tanaka and Yamada, 1993], and is related to both the size of the system N, the difference of the storage ratio from the critical value $(\alpha_c - \alpha)$ and the amount of induced noise mo.

6 Conclusion

Estimating the convergence time order for successful and failed recall using single spin flips is an approach which is clearly limited in scope. The results need confirming analytically and also for other natural language training sets. Nevertheless our results do have something important to say about convergence times for linear associative networks of the Hopfield type.

We have seen that the predicted convergence rate of *0(log(N))* was not observed in our training sets, possibly because of correlations between patterns and bias, despite using a similar range of system sizes for *N* as [Tanaka and Yamada, 1993]. The earlier observation by [Bruce *et* a/., 1987] of the convergence rate being linked to the storage ratio a was observed and appears to be the major factor. When recall fails we have more complex behaviour with convergence times being governed by a number of factors of which system size and the storage ratio are clearly dominant.

Acknowledgements

I would like to thank Professor J. Tsujii for his many helpful and critical suggestions on the computational linguistic aspects of this work. I also gratefully acknowledge the kind permission of Asahi Newspapers of Japan to use their editorial corpus. Funding was provided by the Economics and Social Research Council in the UK award no. R00429434065.

References

[Amit *et* a/., 1987a] D. Amit, H. Gutfeund, and H. Sompolinsky. Information storage in neural networks with low levels of activity. *Phys. Rev. A,* 35(5):2293+, 1987.

[Amit *et* a/., 1987b] D. Amit, H. Gutfeund, and H. Sompolinsky. Statistical mechanics of neural networks near saturation. *Ann. Phys.,* 173:30+, 1987.

[Amit, 1989] D. Amit. *Modeling Brain Function - The world of attractor neural networks.* Cambridge, England: Cambridge University Press, 1989.

[Bruce *et* a/., 1987] A. Bruce, E. Gardner, and D. Wallace. Dynamics and statistical mechanics of the Hopfield model. *Journal of Physics A,* 20:2909-2934, 1987.

[Collier, 1996] N. Collier. Storage of natural language sentences in a Hopfield network. In *International Conference on New Methods in Language Processing*

(NeMLaP-II), Bilkent University, Ankara, Turkey, September II-13th 1996.

[Gardner, 1986] E. Gardner. Structure of metastable states in the Hopfield model. *Journal of Physics A,* 19:L1047-L1052, 1986.

[Grensing *et ai,* 1987] D. Grensing, R. Kiihn, and J. van Hammen. Storing patterns in a spin-glass model of neural networks near saturation. *Journal of Physics A,* 20:2935-2947, 1987.

[Hopfield, 1982] J.J. Hopfield. Neural networks and physical systems with emergent selective computational abilities. *Proceedings of the National Academy of Science, USA,* 79:2554+, 1982.

[Hopfield, 1984] J.J. Hopfield. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Science, USA,* 81:3088-3092, May 1984.

[Koiran, 1994] P. Koiran. Dynamics of discrete time, continuous state Hopfield networks. *Neural Computation,* 6(3):459-468, May 1994.

[Perez-Vicente and Amit, 1989] C. Perez-Vicente and D. Amit. Optimised network for sparsely coded patterns. *Journal of Physics A,* 22:559-569, 1989.

[Tanaka and Yamada, 1993] T. Tanakaand M. Yamada. The characteristics of the convergence time of associative naural networks. *Neural Computation,* 5(3):463-472, May 1993.

[Treves and Amit, 1988] A. Treves and D. Amit. Metastable states in asymmetrically diluted hopfield networks. *Journal of Physics A,* 21:3155-3169, 1988.

NEURAL NETWORKS

Neural Nets 3:
Neurobiologically Inspired Computation