Progress Report
October 18 to November 12, 2010

Raj K. Kumar, Amir Hesam Salavati
E-mail: raj.kumar@epfl.ch, hesam.salavati@epfl.ch

Supervisor: Prof. Amin Shokrollahi
E-mail: amin.shokrollahi@epfl.ch

Algorithmics Laboratory (ALGO)
Ecole Polytechnique Federale de Lausanne (EPFL)

December 13, 2010

1 Introduction

In this report we will give a summary of our activities in past three weeks. We start by describing a short summary of our meetings and discussion with a number of professors at EPFL and their groups. Later, we are going to explain our recent advances on increasing the storage capacity of Hopfield networks. We are currently considering applications of low correlation sequences to reduce the interference term and increase the storage capacity of Hopfield networks.

2 Meetings and Presentations

In the past three weeks, we had a number of meetings with other groups and professors the summary of which is given below, in chronological order.

2.1 Meeting with Prof. Volkan Cevher

As the instructor of the course "Graphical Models" and an expert on topics related to machine learning, we had a discussion with Prof. Volkan Cevher on November 5, 2010. We exlpained our problem and asked if there are other similar problems in machine learning that use Hopfield network or any other similar neural structure. He suggested to view the problem as a learning process over a complete graph in which the graph should *learn* some patterns by adjusting its weights. In this way we are not limited to the Hopfield weighting rule (see equation (2) and can find our own waiting scheme. Raj and I considered this approach as well and has some progress in that regard although we have not yet been able to develop a good algorithm which satisfies the stability condition and does error correction as well.s

2.2 LCN Group Meeting

On 11 November 2010, we presented our ideas for the members Prof. Gerstner's lab, Laboratory of Computational Neuroscience (LCN)[1]. We first described basics of coding theory in order to develop a mutual language for deeper collaborations between members of LCN and ALGO. The presentation proceeded with explanation of our ideas on increasing the storage ca-

[1] The presentation can be found here: https://prezi.com/secure/0ef0fba57232de7893d81909b8d43c4c342c493a

pacity of Hopfield networks, some of which is going to be explained in more details in what follows. In the end, we had a lively discussion with Prof. Gerstner and other members of LCN about the ideas and how to explore them further.

Two of the LCN members were more interested and we met them later on next week. Raj and I explained how codes on graphs and low correlation sequences work and they also suggested some topics in neuroscience literature to us that seemed relevant, including Boltzmann machines.

One particular idea which seemed really interesting was to introduce *low activity* constraint into our model, i.e. consider the model that most of neurons are silent at any given time. This is equivalent to having an encoder for which the codewords contain more zeros than ones. This assumption is more biologically meaningful and its application might be interesting.

2.3 Meeting with Prof. Oliver Leveque

As a result of our discussion with Prof. Cevher, we developed some ideas and formulations about how to pick weights to ensure stability of the network. It was basically a set of linear inequalities that involve both desired patterns and weights. In other words, given the set of desired patterns which we would like to memorize, one should find weights that satisfy the following set of inequalities:

$$< \underline{w}_i, \underline{x}^\mu > x_i^\mu > 1, \ \forall \ i = 1, \dots, n, \ \forall \mu = 1, \dots, M \qquad (1)$$

In which x_i^μ is the i^{th} bit of pattern μ, $\underline{x}^\mu$ is the $1 \times n$ vector corresponding to pattern μ, $\underline{w}_i$ is the i^{th} row of the weight matrix W, $< ., . >$ is the inner product of two vectors, M is the number of desired patterns and n is the length of those patterns.

If $x_i^\mu > 0$, then one can transform equation (1) to $< \underline{w}_i, \underline{x}^\mu >> x_i^\mu$, which relates everything to the eigenvalues of the weight matrix W. That was the reason we talked to prof. Leveque. He had a number of suggestions about this problem but since x_i^μ is not necessarily positive for all patterns and all bits, there is not much that one can do in terms of using eigenvalues.

3 Brief Review of Hopfield Networks

In this section, we briefly review the principles of Hopfield networks as a reminder. Hopfield networks are artificial neural networks that mimic associative memory mechanism in brain, i.e. they memorize a number of patterns during the training phase and then recall the closest pattern to any input given to the network. In that regard, Hopfield networks are kind of like an error correcting decoder.

A Hopfield network is a *complete* graph with n nodes (neurons) in which links are weighted and each node can have a binary state (± 1) [1]. If weights are carefully chosen, Hopfield networks are able to "memorize" a number of patterns with length n. Here, memorizing means if we do a training phase, memorized patterns are stable states of the network, i.e. if we feed them as inputs the network does not evolve. Furthermore, in certain types of Hopfield networks, we can achieve some degree of error correction. In these cases, the network converges to the closest memorize pattern (stable state) for a given input. We will get back to these types later on.

In traditional Hopfield networks, the weights between nodes i and j, w_{ij}, is determined as follows:

$$w_{ij} = \frac{1}{n} \sum_{m=1}^{M} x_i^m x_j^m \tag{2}$$

where x_i^m is the i^{th} bit of the m^{th} memorized pattern and M is the total number of such patterns.

Given the weights, if we initialize the network with some pattern, neurons update their state according to equation (3), in which the state of neuron k is denoted by s_k. In words, each neuron calculates the weighted sum over its input links and if the sum was larger than a threshold θ (which can vary over time in general), neuron fires, i.e. its state is changed to $+1$, and remains silent otherwise.

$$s_i = \begin{cases} 1, & \sum_{j=1}^{n} w_{ij} s_j > \theta \\ -1, & \text{Otherwise} \end{cases} \tag{3}$$

4 Low Correlation Sequences

In previous reports, we saw that if we try to memorize codewords of a linear code, we will have trouble since all the weights would become zero if one uses the traditional Hopfield weighting scheme as given by equation (2). To overcome this issue, we are going to use sequences with low correlation properties. The motivation comes from the expansion of equation (3): suppose we have selected the weights according to the Hopfield rule and would like to recall one of the memorized pattern, say pattern q. Then, if we compute the input weighted sum of neuron i we will get:

$$\begin{aligned} h_i &= \sum_{j=1}^{n} w_{ij} x_j^p = \frac{1}{n} \sum_{j=1}^{n} \sum_{\mu=1}^{M} x_i^m x_j^m x_j^p \\ &= \frac{1}{n} \sum_{\mu=1}^{M} x_i^m \sum_{j=1}^{n} x_j^m x_j^p = x_i^p + \frac{1}{n} \sum_{\mu=1,\neq p}^{M} x_i^m < \underline{x}^m, \underline{x}^p > \end{aligned} \tag{4}$$

In which $< \underline{x}^m, \underline{x}^p >$ is the inner product of patterns p and m. In the above equation, the first term, i.e. x_i^p is the desired one because we would like the sign of h_i to be equal to the sign of x_i^p if our goal is to recall pattern p (we assume that the firing threshold θ is zero for the moment). The second term is the interference term and it depends on the correlation between pattern p and all the other patterns. Ideally, we would like the interference term be as small as possible such that it does not affect the sign of x_i^s.

One way to obtain small interference terms is to consider low correlation sequence for which the inner product between any of the patterns are reasonably small. There is a well-known Welch bound on the minimum correlation of a set binary patterns which determines the performance of optimal low correlation sequences. A family of low correlation patterns, called Gold sequences, has been introduced for CDMA communications in 1970s [2] and are optimal in terms of Welch bound, i.e. the *maximum* correlation between any two pattern is equal to the lower bound enforced by the Welch bound. It can be shown that the correlation between any two members of Gold sequences is at most $O(\sqrt{n})$.

5 Future Works

In the next weeks, we are going to apply low correlation sequences to Hopfield networks and assess their performance in terms of storage capacity and noise tolerance.

Pursuing integration of low activity idea into our model, as mentioned in our meeting with LCN group members, would be another topic we will address in (longer) future.

References

[1] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities", National Acad Science, 1982.

[2] R. Gold, "Optimal binary sequences for spread spectrum multiplexing", IEEE Transactions on Information Theory, Vol. 13, No. 4, 1967, pp. 619 −621.