In *Applications and Science in Soft Computing*, A. Lotfi & J.Garibaldi (eds.), *Advances in Soft Computing* Series, Springer 2003, pp 3–8.

A Recurrent Self-Organizing Map for Temporal Sequence Processing

T. A. McQueen, A. A. Hopgood, J. A. Tepper and T. J. Allen

Department of Computing & Mathematics, The Nottingham Trent University, Burton Street, Nottingham, NG1 4BU, United Kingdom
e-mail:{thomas.mcqueen,adrian.hopgood,jonathan.tepper,tony.allen}@ntu.ac.uk

Abstract. We present a novel approach to unsupervised temporal sequence processing in the form of an unsupervised, recurrent neural network based on a self-organizing map (SOM). A standard SOM clusters each input vector irrespective of context, whereas the recurrent SOM presented here clusters each input based on an input vector and a context vector. The latter acts as a recurrent conduit feeding back a 2-D representation of the previous winning neuron. This recurrency allows the network to operate on temporal sequence processing tasks. The network has been applied to the difficult natural language processing problem of position variant recognition, e.g. recognising a noun phrase regardless of its position within a sentence.

1 Introduction

Temporal sequence processing (TSP) is an increasingly important field for neural networks, with applications ranging from weather forecasting to speech recognition [1]. TSP involves the processing of signals that vary over time. Problems such as predicting the weather generally cannot be solved by just examining a set of current inputs from the dynamic system in question, e.g. a satellite image showing today's cloud cover. Rather, any prediction must be based on the current input in the context of a number of previous inputs, e.g. a satellite image for today along with satellite images from the previous five days, showing how the weather has changed so far over the week.

Neural network models for TSP outperform alternative methods, such as NARMAX [9], mainly due to their ability to learn and generalize when operating on large amounts of data [9]. Supervised learning is usually used to solve TSP problems, i.e. the recurrent neural network must be explicitly trained by providing a desired target signal for each training exemplar. Current supervised learning methods are computationally inefficient [8] and are unable to solve certain types of problems [6].

A number of unsupervised neural networks for TSP have been proposed [6], mostly based on the self-organizing map (SOM) [5]. These models use a variety of external and internal memory mechanisms to capture information concerning past inputs, e.g. tapped delay lines and leaky integrators. Unsupervised learning has advantages over equivalent supervised techniques in that it makes fewer assumptions about the data it processes, being driven solely by the principles of self-organization, as opposed to an external target signal.

We present a novel, unsupervised, recurrent neural network based on a SOM to identify temporal sequences that occur in natural language, such as syntactic groupings. The network uses both an input vector and a context vector, the latter of which provides a 2-D representation of the previous winning neuron. The proposed network is applied to the difficult natural language processing (NLP) problem of position variant recognition, e.g. recognizing a noun phrase regardless of its position within a sentence.

2 Architecture and algorithm

The network has a 28-bit input vector that provides a binary representation of the input tag being processed. In addition to this input vector, the network also uses a second context vector. The size of this context vector can be varied depending on the size of the network, but in experiments detailed below the context vector was set to 10 bits (Fig. 1). Both the input and the context vector are used in the Euclidean distance calculation to determine the winning neuron in a similar manner to a standard SOM.

The context vector represents the previous winning neuron using a 10-bit coordinate vector. The most significant five bits of this vector (i.e. the five bits on the left) represent the binary number of the winning neuron's column, while the least significant five bits (i.e. five bits on the right) represent the binary number of the winning neuron's row.

This approach is an efficient method of coordinate representation that provides the network with a 2-D view of spatial context. It is an improvement over an initial approach, which represented the previous winning neuron using only a binary representation of its number within the SOM. Such a representation prevented the network from seeing similarities between neighboring neurons in adjacent columns. For example, neuron 20 and neuron 40 are neighbors on the SOM shown above and will therefore be representative of similar patterns. However, the binary representation of the numbers 20 (i.e. 010100) and 40 (i.e. 101000) are dissimilar. Thus similar input patterns may result in dissimilar context causing similar sequences to be clustered to significantly different regions of the SOM. It is envisaged that this would reduce the network's ability to generalize.

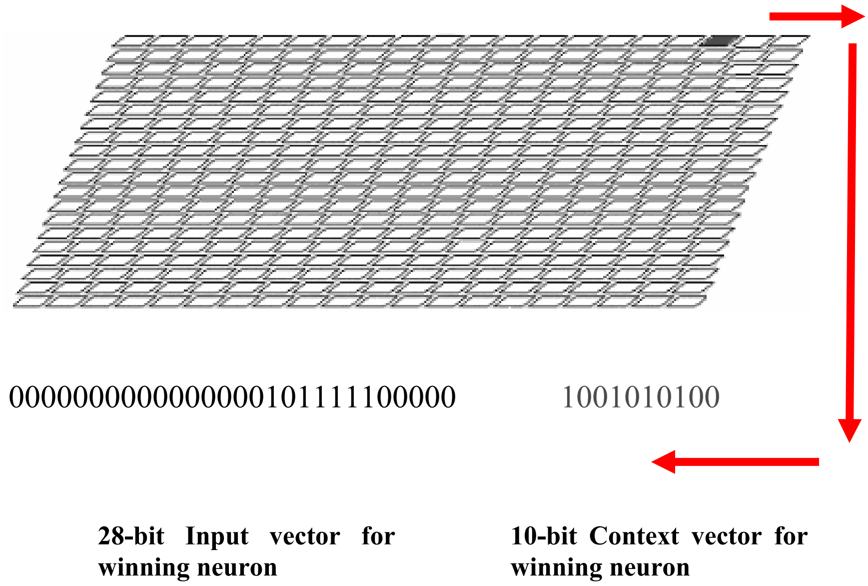

Fig. 1 Network showing recurrent feedback

The coordinate system of context representation solves this problem by effectively providing the network with a 2-D view of winning neurons. In the example given above, neuron 20 would be represented as 0000110100, while neuron 40 would be represented as 0001010100. (Note that only two bits are different in this example as opposed to four bits in the example above).

As with the standard SOM, the recurrent SOM presented here uses a neighborhood function to update the weights of neurons in a region around the winning neuron. Both the weight vector and the context vector of neighboring neurons are moved towards those of the respective input and context vectors. The network uses a Gaussian neighborhood function to calculate the learning rate that will be applied to these neurons. This function allows immediately neighboring neurons to experience similar weight changes to those of the winning neuron, while distant neurons experience minimal weight changes. However, in order to improve computational efficiency, the neighborhood function uses a cut-off value, beyond which neurons do not take part in weight updates at all.

3 Experiments

Initially, the new network is being applied to a corpus-based natural language task (Fig. 2) using the Lancaster Parsed Corpus (LPC) [7]. At present, the main objective of the research is to identify coarse phrase boundaries (e.g. noun phrases or verb phrases with little or no embedding) that may emerge on the topological map from exposure to linear sequences of words (sentences) that have been pretagged with symbols denoting the word's part-of-speech (e.g. noun, adjective, verb etc) [2].

A network with an output layer of 20 × 20 neurons was trained in two phases, following Kohonen's research on training SOMs [3]. The first convergence phase consisted of 1000 epochs, in which the learning rate was linearly reduced from an initial value of 0.1, but was not allowed to fall below 0.01. This was followed by a second fine-tuning phase in which a learning rate of 0.01 was applied for 2500 epochs. While the number of epochs in the first phase conforms with Kohonen's research [3], the number of epochs in phase two is considerably smaller than the number suggested. At this initial stage in the research, this reduction is necessary due to time and computational constraints. However, experimental analysis has not shown a significant reduction in the quality of results when training times in phase two are reduced.

A sample of 654 sentences from the LPC [7] were presented to the network. Presentation occurred in random order to improve training efficiency and to prevent the weights from becoming stuck during the low neighborhood value in phase two. The context vector is set to zero between each sentence to prevent contextual information from previous sentences interfering with subsequent sentences.

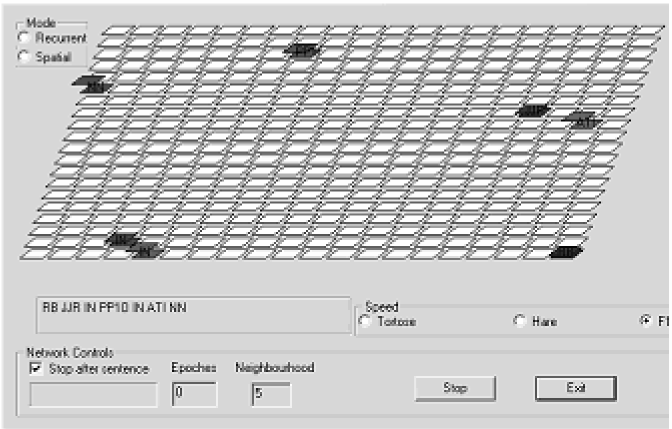

Fig. 2 – Screenshot from the current network. The raised, coloured polygons represent winning neurons for the sentence of tags presented to the network.

4 Results

The preliminary results are encouraging, as they show that word tags are being clustered in locations consistent with their context. The results in Figs. 3–5 show three simple artificially constructed sentences of varying tense. Despite these variations in tense, each exhibits a similar trace pattern over the map. We refer to these traces as signatures.

Fig. 6 shows two simple noun phrases with and without a preposition. While both sentences show similar signatures for the noun phrase, the effect of the preposition can clearly be seen to alter the signature of the second phrase.

It is hoped that further analysis will reveal the extent to which the network can exploit the context and show what kind of temporal syntactic patterns the network can find in input sequences. A major benefit of finding such patterns in an unsupervised manner is that, unlike supervised techniques, there is no dependency on manually annotated corpora, which are not widely available due to the high costs associated with manually annotating raw language data. In fact it is envisaged that, should the unsupervised system prove successful in extracting syntactic structure, it would serve as an automatic syntactic annotation system thus reducing the need and cost of manual annotation.

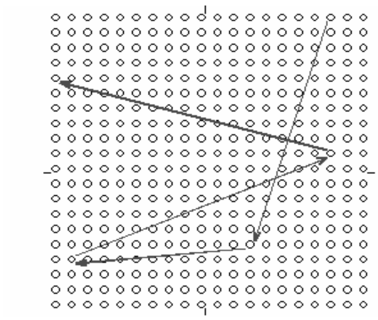

Fig. 3 – *Signature for sentence:"she goes down the stairs"*

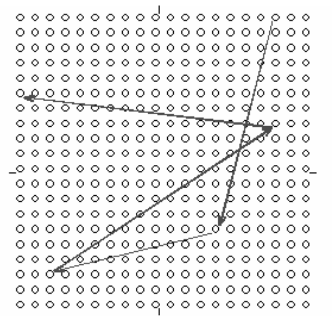

Fig. 4 – *Signature for sentence: "she went down the stairs"*

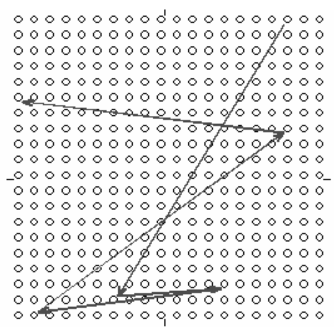

Fig. 5 – *Signature for sentence:"she is going down the stairs"*

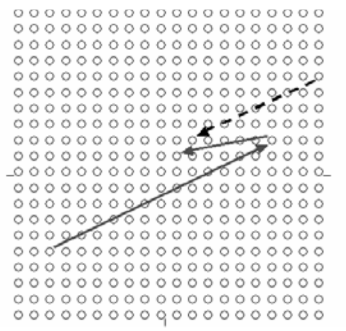

Fig. 6 – *Noun phrase with and without preposition*

The home

In the home

5 Conclusions and future work

We have presented a novel recurrent SOM and applied it to the problem of position-variant recognition. We have shown that the network forms signatures in response to temporal sequences present in the inputs.

In addition to the natural language task, research is also being conducted into enhancing the recurrent SOM using lateral connections and a temporal Hebbian learning [4] mechanism. The purpose of such a mechanism is to attempt to control the recurrency, allowing feedback to occur only when the winning neurons, whose representations are to be fed-back, are stable. This temporal Hebbian learning mechanism has been used in a previous experimental neural network and it is hoped that it will reduce the SOM's training time.

In the next phase of this investigation, hierarchical clustering methods based on temporal SOMs will be developed to obtain finer-grained syntactic groupings. Future work will focus on the context representation that is fed back. The representation may be enlarged to give more emphasis to the context vector than the input vector, and it may also be optimized using genetic algorithms. Further experiments will be performed in the domain of natural language processing; specifically the network will be used to attempt to detect phrase boundaries. Additionally, if the network proves successful, it may also be used in a number of other areas including computer virus detection, speech recognition and image analysis.

On a wider scale, the recurrent SOM could be used as the core of a temporal neural processing system. For example, the recurrent SOM clusters patterns based on input featural similarities whilst a supervised neural network uses these reduced representations to perform a mapping to a corresponding set of desired outputs.

References

[1] Barreto G and Arajo A (2001) Time in self-organizing maps: An overview of models. Int. J of Computer Research, 10(2):139-179

[2] Garside R, Leech G and Varadi T (1987) Manual of information to accompany the Lancaster Parsed Corpus. Department of English, University of Oslo

[3] Haykin S (1999) Neural Networks: A Comprehensive Foundation, Prentice Hall

[4] Hebb D (1949) The Organization of behaviour, John Wiley

[5] Kohonen T (1984) Self-Organization and Associative Memory, Springer-Verlag

[6] Mozer M (1994) Neural net architectures for temporal sequence processing, in.Weigend A and Gershenfeld N (eds), Time Series Prediction, pp 243–264

[7] Tepper J, Powell H and Palmer-Brown D (2002) A corpus-based connectionist architecture for large-scale natural language parsing. Connection Science, 14 (2)

[8] Schmidhuber J (1991) Adaptive history compression for learning to divide and conquer, Int. Joint Conf. on Neural Networks, Vol 2, pp 1130–1135

[9] Varsta M and Heikkonen J (1997) Context learning with the self-organizing map, Proc. Workshop on Self-Organizing Maps, pp 197–202