A Hypothesis on How the Neocortex Extracts Information for Prediction in Sequence Learning

Weiyu Wang

Department of Biology, Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong
bo_wwx@ust.hk

Abstract. From the biological view, each component of a temporal sequence is represented by neural code in cortical areas of different orders. In whatever order areas, minicolumns divide a component into sub-components and parallel process them. Thus a minicolumn is a functional unit. Its layer IV neurons form a network where cell assemblies for sub-components form. Then layer III neurons are triggered and feed back to layer IV. Considering the delay, through Hebbian learning the connections from layer III to layer IV can associate a sub-component to the next. One sub-component may link multiple following sub-components plus itself, but the prediction is deterministic by a mechanism involving competition and threshold dynamic. So instead of learning the whole sequence, minicolumns selectively extract information. Information for complex concepts are distributed in multiple minicolumns, and long time thinking are in the form of integrated dynamics in the whole cortex, including recurrent activity.

Keywords: Sequence prediction; Columnar architecture; Neocortex; Connectionism; Associative memory.

1 Introduction

Most human and animal learning processes can be viewed as sequence learning. Sun and Giles summarize problems related to sequence learning into four categories: sequence prediction, sequence generation, sequence recognition, and sequential decision making [1]. The four categories are closely related [1], and sequence prediction is arguably the foundation of the other three. Sequence learning can be touched by various disciplines, while typically it deals with sequences of symbols and is applied to language processing. In this problem, a temporal pattern is defined as a temporal sequence and each static pattern constituting it is defined as a component (Wang and Arbib, [2]). Because of the intrinsic complexity of language, a component usually cannot be determined solely by the previous component, but by a previous sequence segment defined as context [2]. To learn complex sequences, a short-term memory (STM) at least of the maximum degree of these sequences is inevitable. And at least one context detector is assigned to each context. So according to the model proposed by Wang and Yuwono in 1995 [3][4], a neural network with 2m+(n+1)r neurons (m context sensors, m modulators, n terminals each with a STM of length r) can learn an arbitrary sequence at most of length m and degree r, with at most n symbols. Starzyk

F. Sun et al. (Eds.): ISNN 2008, Part I, LNCS 5263, pp. 21–29, 2008.

and He proposed a more complex model with hierarchical structure in 2007 [5]. To learn a sequence of length l with n symbols, the primary level network requires 3nl+2n+2l+m neurons, where m is for the number of output neurons for the next hierarchical level network (equals the number of symbols in the next level), and the total number of neurons should include all hierarchical levels [5]. Such expensive cost makes the application of sequence learning undesirable.

Another problem is if we expand the discipline from language to others, for example vision, the input is nearly a continuous time temporal sequence with continuous value components, as the time interval is in milliseconds and thousands of neurons are involved for the primary visual representation. This leads to an extremely large symbol set, and extremely long sequence to be learn even within a few minutes. So it is obviously impossible to take the traditional sequence learning method aiming at remembering the whole sequence and the relationships from each context to its corresponding component.

So we have to think out other methods to solve these problems. And there is surely an answer, as the guarantee is just the existence of us ourselves. We do read piles of articles and indeed learn something from them. We receive tremendous amount of information from our sense organs throughout our lives, and even at the last moment of our life, we can recall some scenes in our earliest life stage. Obviously, what's important is not only how to learn, but also what to learn. This article touches sequence learning from a different viewpoint—how to pick up useful information from the input sequences and store it in an organized way. This is defined as "information extraction". Our idea is to solve this problem by studying the biological architecture of the nervous system. A mechanism for information extraction is hypothesized based on the hierarchical and columnar organization of the cerebral cortex in part 2. A neural network is built to simulate the function of a single minicolumn according to this hypothesis in part 3. Part 4 gives the conclusion and summarizes the significance of this model.

2 The Mechanism for Information Extraction

2.1 The Hierarchical Structure of Neocortex and Abstraction

Take the visual pathway as example. Light enters the eyes and is transduced to electrical signal in retina. The neural signal is transferred to primary visual areas via thalamus. Then information is submitted to secondary visual areas. For forming declarative memory, further transfer is to medial temporal lobe memory system and back to higher order cortex areas [6][7][8]. Though the mechanism of declarative memory formation is not completely clear yet, it is widely accepted that forming abstract concepts requires high level integration of information. Along this pathway the integration level rises, so is the abstraction level. If we describe this pathway mathematically as a vector series V_1, V_2,…,V_n, where V_i is a N_i elements 0-1 vector, then a component of a temporal sequence is represented by a assignment to each vector in this vector series, instead of only one vector. Notice the higher footnote i is, the higher abstraction level V_i has. And V_i depends onV_{i-1} (i=2, 3,…n). This structure is somewhat an analog of Starzyk and He's hierarchical model [5], in the difference that

it deals with real neural code instead of symbols, and much more complex integration (computation) is applied between two hierarchical levels.

2.2 The Columnar Organization of Neocortex and PDP

The neocortex is horizontally divided into 6 layers. Layer IV contains different types of stellate and pyramidal cells, and is the main target of thalamocortical and intrahemispheric corticocortical afferents. Layers I through III are the main target of interhemispheric corticocortical afferents. Layer III contains predominantly pyramidal cells and is the principal source of corticocortical efferents. Layer V and VI are efferents to motor-related subcortical structures and thalamus separately [9]. Vertically neocortex is columnar organized with elementary module minicolumn. The minicolumn is a discrete module at the layers IV, II, and VI, but connected to others for most neurons of layer III [10].

Considering the vector series V_1, V_2,…,V_n, vector V_i is divided into sub-vectors in corresponding minicolumns for any i. Each sub-vector represents a sub-component, and is processed independently in its minicolumn. Minicolumns transmit processed information to the next hierarchical level minicolumns. This accords with the idea of "Parallel Distributed Processing" (PDP) proposed by Rumelhart and McClelland [11][12].

2.3 Minicolumn Architecture

A model for the structure of a minicolumn is shown in figure 1. In this model, all pyramidal cells and stellate cells in layer IV of the minicolumn form a symmetrical Hebbian network. As neurons involved are limited and closely packed, we can assume any neuron is connected to all other neurons through short axons, whose transmission delay can be omitted. If the pyramidal cells connect to pyramidal cells directly, the connections are excitatory. If the pyramidal cells connect to other pyramidal cells through stellate cells, the connections are inhibitory. Thus this network contains both excitatory and inhibitory connections. Typical Hebbian learning in this network will form cell assemblies [13]. Each cell assembly stands for a sub-component.

Signals are transmitted from layer IV pyramidal cells to layer III pyramidal cells through long axons. As layer III contains predominantly pyramidal cells, the connections are mainly excitatory. Thus layer III is not an idea place for forming cell assemblies, as without inhibitory connections two cell assemblies will intermingle with each other and become one if only they have very small overlapping. The representations in layer III are just corresponding to the cell assemblies in layer IV, and we can assume no overlapping in layer III, as this can be automatically achieved through a winner-take-all (WTA) mechanism also used in Wang and Arbib's model [2]. Signals are transmitted from layer III through long axons either to other minicolumns, or back to layer IV.

2.4 Association in Minicolumns during Learning

What's important is the transmission from layer III back to layer IV (the feedback). Typically the function of a feedback is thought for refinement or synchronization,

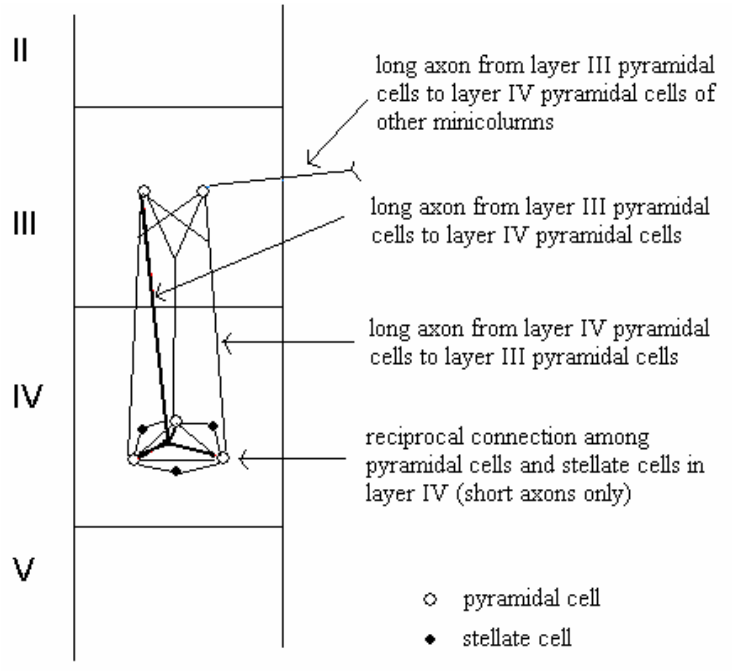

Fig 1. Structure of a minicolumn. Focusing on layer IV (afferent) and layer III (efferent). Pyramidal cells and stellate cells in layer IV connect with each other through short axons, forming a network with both excitatory and inhibitory synapses. Layer IV pyramidal cells transmit signals to layer III pyramidal cells through long axons. Layer III pyramidal cells may transmit signals to layer IV pyramidal cells of other minicolumns through long axons, or transmit signals back to its own layer IV pyramidal cells through long axons, forming feedback loop (indicated by the thick lines).

for example in the model proposed by Korner etc. [14]. But in our view, the feedback loop along with the transmission delay is the base for associating a sub-component to the next sub-component. Notice the involved two sub-components are not input at the same time, but Hebbian learning based on the synapse plasticity requires the two involved neurons exciting at the same time [13][15-17]. This is solved by the transmission delay of this feedback loop. The synapse modification can only happen in the synaptic junction, by the changes of the amount of neurotransmitter released by the presynaptic neuron, or the number of postsynaptic receptors [15-17]. Suppose the delay from the excitation of layer IV pyramidal cell bodies (dendrites) to the excitation of layer III pyramidal cell axon terminals is Δt, and the lasting time of sub-component A and sub-component B are t_1 and t_2 respectively (t_1, $t_2 >> \Delta t$), B follows A. Then from time 0 to Δt, no Hebbian learning happens at the synaptic junctions between layer III pyramidal cell axons and layer IV pyramidal cell bodies (dendrites), for only the later is exciting. From Δt to t_1, the Hebbian learning associates sub-component A with itself, denoted as learning the ordered pair (A,A). From t_1 to $t_1+\Delta t$, the layer III pyramidal cell axon terminals still represent sub-component A, while the

layer IV pyramidal cell bodies (dendrites) already code for sub-component B. Hence the association is (A,B). From $t_1+\Delta t$ to t_2, the association will be (B,B).

2.5 Competition and Threshold Dynamic during Retrieval

Suppose A, B, C, B, D, E, A, B, F, D, E, C, A, B denotes a sequence composed of sub-components of a temporal sequence in a minicolumn. Then after learning (A,B), (B,C), (B,D), (B,F), (C,B), (C,A), (D,E), (E,A), (E,C), (F,D) plus (A,A), (B,B), (C,C), (D,D), (E,E), (F,F) are learned. Now input A (lasting time $t> \Delta t$). The cell assembly for A in layer IV is evoked. From Δt to t, the feedback from layer III try to evoke both A and B. But A is exciting, supported by the exterior input. It will inhibit the exciting of cell assembly for B. Until the exterior input ceases at time t, the only remaining stimulation is from layer III, and this stimulation will last exactly Δt. Because cell assembly for A has excited, the threshold of its neurons raises, thus it cannot be evoked again for quite a while (at least Δt). Thus cell assembly for B finally gets its chance to excite. After another Δt cell assembly for B ceases exciting and cannot be evoked again, and layer III feedback try to evoke three cell assemblies for C, D, F separately. They all want to excite and inhibit the other two, the competition leads to nothing excited (more accurately, the three may excite as a "flash" for inhibition is triggered by exciting, but this "flash" is so short compared with Δt and disappears without further effect). Hence from the exterior performance of the minicolumn, only (A,B), (D,E), (F,D) are learned.

2.6 Summary

By the mechanism described above, an input temporal sequence is understood at different abstraction level in different hierarchical levels of the neocortex. In each hierarchical level, the components (temporal sequences) are divided into sub-components (sub-temporal sequences) by minicolumns. Each minicolumn only extracts the deterministic feature of the sub-temporal sequences: if sub-component A is always followed by sub-component B and no other sub-components, the minicolumn learns A predicts B.

3 The Neural Network Simulation of the Minicolumn

We only built a small network containing 10 layer IV pyramidal cells and 10 layer III pyramidal cells for demonstration. Of course the network can be expanded to hundreds of neurons to simulate the real minicolumn. Let binary arrays F[10] and T[10] denote the layer IV neurons and layer III neurons separately. For simplicity, we let T[i] = F[i], i=0, 1,…,9, though in real case the representations in layer III for cell assemblies in layer IV can be quite different and involve different numbers of neurons. Thus a cell assembly 1111100000 is also represented 1111100000 in layer III in our network. Array Thresh[10] denotes the thresholds of the layer IV neurons, whose value is 1 initially and 21 after exciting, but returns to 1 after Δt. Intra[10][10] is the learning matrix for association among layer IV neurons, whose value is in [-300, 30]. (Negative means inhibitory. As one pyramidal cell can inhibit another through numerous stellate cells, the inhibitory connection is thought to be much stronger.)

Inter[10][10] is the learning matrix for association from layer III to layer IV, whose value is in [0,2] (only excitatory, thus the effect of layer III pyramidal cells on layer IV pyramidal cells are not as strong as it of layer IV pyramidal cells on themselves).

The sequence learning process takes discrete steps, and set $\Delta t = 1$ step (the delay in a minicolumn cannot be very long). An input sequence is noted as A[10](a), B[10](b), C[10](c),…where A[10], B[10], C[10] are 10 element 0-1 vectors, and a, b, c are integers for the number of steps which the state lasts. At one step when the input is I[10](n) (n>0 is the remaining time this state lasts), learning starts with setting F[i] = I[i]. The learning rule for updating intra[i][j] is

$$\text{intra}[i][j]=(\text{intra}[i][j]>=0)\times(F[i]F[j]\times 0.5\times(30-\text{intra}[i][j])-F[i]\overline{F[j]}\times 3)+$$
$$(\text{intra}[i][j]<=0)\times(F[i]F[j]\times 30-F[i]\overline{F[j]}\times 0.5\times(300+\text{intra}[i][j]))+\text{intra}[i][j]$$

The learning rule for updating inter[i][j] is

$$\text{inter}[i][j]=T[i]F[j]\times 0.5\times(2-\text{inter}[i][j])+\text{inter}[i][j]$$

At each step, after updating both learning matrixes, the signal transmission from layer IV to layer III is denoted by T[i] = F[i]. The threshold refreshing rule is Thresh[i] = 1+20F[i] . Finally set I[10](n) = I[10](n-1), and continue (when n = 1, the next state is loaded in).

During retrieval, still suppose the stimulation is I[10](n). Retrieving starts with setting F[i]=I[i]. The evoked neurons is determined by

$$F[i]=1\times(\sum_{j\neq i}F[j]\text{intra}[j][i]+\sum_{j}T[j]\text{inter}[j][i]>\text{thresh}[i])$$

Notice in each step we need to repeat the above calculation until F[i] no longer changes (as newly evoked neurons can in turn evoke others). Then the result is the final evoked cell assembly. And let T[i] = F[i] simulating the information transmission. Refresh threshold by Thresh[i] = 1+20F[i]. Finally set I[10](n) = I[10](n-1) and continue (when n= 1, set I[i] = 0 and n = 1).

Now look at an example. The temporal sequence 1111000000(16), 0000000001(24), 0000111000(13), 1111000000(7), 0000000110(19) is input for 10 or more times (enough repeating times are necessary as the inter-state association can only happen once when one state changes to another). After learning, intra[10][10] approximates

$$\begin{pmatrix}
30 & 30 & 30 & 30 & -300 & -300 & -300 & -300 & -300 & -300\\
30 & 30 & 30 & 30 & -300 & -300 & -300 & -300 & -300 & -300\\
30 & 30 & 30 & 30 & -300 & -300 & -300 & -300 & -300 & -300\\
30 & 30 & 30 & 30 & -300 & -300 & -300 & -300 & -300 & -300\\
-300 & -300 & -300 & -300 & 30 & 30 & 30 & -300 & -300 & -300\\
-300 & -300 & -300 & -300 & 30 & 30 & 30 & -300 & -300 & -300\\
-300 & -300 & -300 & -300 & 30 & 30 & 30 & -300 & -300 & -300\\
-300 & -300 & -300 & -300 & -300 & -300 & -300 & 30 & 30 & -300\\
-300 & -300 & -300 & -300 & -300 & -300 & -300 & 30 & 30 & -300\\
-300 & -300 & -300 & -300 & -300 & -300 & -300 & -300 & -300 & 30
\end{pmatrix}$$

Inter[10][10] approximates

$$\begin{pmatrix} 2 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 2 & 2 & 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 2 \end{pmatrix}$$

Four cell assemblies 1111000000, 0000111000, 0000000110, 0000000001 are formed. The extracted information is (0000000001, 0000111000), and (0000111000, 1111000000), thus input 0000000001 will return sequence 0000111000, 1111000000. 0000000110 retrieves nothing as it is associated to nothing. 1111000000 retrieves nothing either, but it's because it is associated to both 0000000110 and 0000000001.

In this neural network, it is required that cell assemblies do not overlap. IF two cell assemblies in layer IV overlap, their representations in layer III also share a common part. This common part will try to evoke both cell assemblies no matter which of them causes this, leading to undesired results. This can be solved if another feed forward learning is added for constructing the representations for cell assemblies in layer III, ensuring no overlapping (for example, the WTA mechanism used in Wang and Arbib's model [2]).

Rarely oscillation may happen during retrieval. This requires the input sequence itself ends with repeating circles, like the sequence A, B, C, D, C, D, C. Thus after learning this sequence, input C or D will lead to the oscillation with C and D alternatively. But this situation is really rare as if the above sequence doesn't end with C or D, for example A, B, C, D, C, D, C, A. Then C will not retrieve D (as it is associated to both D and A), and no oscillation can happen.

4 Conclusion and Significance

The model proposed in this article deals with the sequence learning problem from a different viewpoint: extracting information. Adopting this idea, what's important is what information to extract rather than how to remember all information. The most significant advantage of this idea is that the memory capacity required is not proportional to the sequence length and degree, but to the useful information (knowledge) contained in the sequence. Multiple different sequences may contain common knowledge. The common knowledge appears as the same sub-sequences in the certain minicolumns of certain hierarchical levels. For example, a stone, a tire, or a basket ball rolling down a hill appear to be quite different scenes if considering every detail, but all of them are abstracted as the process of an round object rolling down a slope in physics. This is because the essence of abstraction is the process of extracting important common

features while omitting the other unimportant details. This process is fulfilled in our model by the complex connections among minicolumns of different hierarchical levels, which lead to complicated neural computation. Naturally, along with the increase of abstraction level, the knowledge is more and more general and the amount of information is reduced, represented by decrease of the variation of sequences. It is arguable that in the high enough hierarchical levels, only few sequences repeat frequently.

The learning is by forming associative memory in minicolumns. Each minicolumn associates a sub-component to itself and its immediate follower. But through competition and threshold dynamic, A evokes B if and only if B is the only possible follower of A. This means a minicolumn doesn't consider a temporal sequence's degree. Every temporal sequence is treated as a simple sequence. Thus a minicolumn can remember only a small portion of the sequence by itself, seemingly useless compared with Wang and Yuwono's model [3][4] and Stazyk and He's model [5]. But the advantage is that the neural network for a minicolumn is extremely simple, as described in part 3, with much less cost than Wang's or Stazyk's models. Thus it is very proper for being a functional unit. The complex tasks are hoped to be accomplished by the whole network composed of millions of such functional units.

Typically a sub-component in a minicolumn can only retrieve one or two following sub-components, and then this minicolumn ceases. But the retrieved sub-components are submitted to higher level minicolumns, and may trigger the retrieving in them. Repeating this activity, and by possible crosses or loops (recurrent activity), a sub-component might trigger unlimited retrieving. This process must be consciously controlled by concentration (a mysterious cognitive function not discussed in this article).

Finally, the model has the following important features:

1. higher hierarchical level minicolumns tend to learn more than lower hierarchical level minicolumns, as in high abstraction level sequence variation is reduced.
2. two seemingly completely different objects may retrieve the same thing, if only they share some common feature and the concentration is on this common feature. For example, an elephant and the glacier both may retrieve the concept of "huge".

Acknowledgments. Thank Bertram E. SHI of dept of electronic & computer engineering HKUST for offering illuminating advice and inspiring discussion.

References

1. Sun, R., Giles, L.C.: Sequence Learning: From Recognition and Prediction to Sequential Decision Making. IEEE Intell. Syst. 16, 67–70 (2001)
2. Wang, D., Arbib, A.M.: Complex Temporal Sequence Learning Based on Short-term Memory. Proc. IEEE 78, 1536–1543 (1990)
3. Wang, D., Yuwono, B.: Anticipation-Based Temporal Pattern Generation. IEEE Trans. Syst. Man Cybern. 25, 615–628 (1995)
4. Wang, D., Yuwono, B.: Incremental Learning of Complex Temporal Patterns. IEEE Trans. Neural Networks 7, 1465–1481 (1996)
5. Starzyk, A.J., He, H.: Anticipation-Based Temporal Sequences Learning in Hierarchical Structure. IEEE Trans. Neural Networks 18, 344–358 (2007)

6. Squire, R.L., Zola, M.S.: The Medial Temporal Lobe Memory System. Science 253, 1380–1386 (1991)
7. Thompson, F.R., Kim, J.J.: Memory systems in the brain and localization of a memory. PNAS 93, 13438–13444 (1996)
8. Mayes, A., Montaldi, D., Migo, E.: Associative Memory and the Medial Temporal Lobes. Trends Cogn. Sci. 11, 126–135 (2007)
9. Creutzfeldt, D.O.: Cortex Cerebri: Performance, Structural and Functional Organization of the Cortex. Oxford University Press, USA (1995)
10. Mountcastle, B.V.: The Columnar Organization of the Neocortex. Brain 120, 701–722 (1997)
11. Rumelhart, D.E., McClelland, J.L.: The PDP Research Group: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Foundations, vol. 1. MIT Press, Cambridge (1986)
12. McClelland, J.L., Rumelhart, D.E.: The PDP Research Group: Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Psychological and Biological Models, vol. 2. MIT Press, Cambridge (1986)
13. Hebb, D.O.: The Organization of Behavior. Wiley, New York (1949)
14. Korner, E., Gewaltig, O.M., Korner, U., Richter, A., Rodemann, T.: A model of computation in neocortical architecture. Neural Networks 12, 989–1005 (1999)
15. Bliss, P.V.T., Collingridge, L.G.: A synaptic model of memory: long-term potentiation in the hippocampus. Nature 361, 31–39 (1993)
16. Bear, F.M.: A synaptic basis for memory storage in the cerebral cortex. PNAS 93, 13453–13459 (1996)
17. Chen, R.W., Lee, S., Kato, K., Spencer, D.D., Shepherd, M.G., Williamson, A.: Long-term modifications of synaptic efficacy in the human inferior and middle temporal cortex. PNAS 93, 8011–8015 (1996)