www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 2 Issue 6 June, 2013 Page No. 1909-1913

NEUROCOMPUTATIONAL MODEL USING LEABRA FRAMEWORK FOR INFORMATION STORAGE

Singh S.
Dept. of CSE&IT, ITM University, Gurgaon – 122017, India
sandeepsingh@itmindia.edu

Abstract

Local, error-driven and associative, biologically realistic algorithm (LEABRA) is a widely used framework to design neurocomputational models for cognitive processes. The complex structure of brain layers and interconnected neuronal units form a pattern to store specific information. In an object the information content is high at edges, corners and angles formed in between two planes. It is quoted in various research journals that the neuronal weight computation is based the high information content parts than the less variation in colour in the image. In this work we have proposed a neurocomputational model to store and retrieve the information of an object. After training the model is tested on various similar objects and it can recognise the object with some error. The model can also recognise the objects having similar in terms of number of sides and number of angles.

Keywords

LEABRA, Artificial Neural Network, Object Recognition, Hebbian Learning, neurocomputational model

1. Introduction

Electrophysiological studies on the brain of monkey and fMRI studies of human brain showed active part of the brain to recognise the learned objects. After receiving the object characteristics from visual stimuli present on retina, the signal goes to primary visual cortex (V1) which project the image to middle temporal area (MT) for initial processing like orientation, motion and selectivity [1,2,3]. Extra striate visual cortex area V2 and V4 are connected with V1 projection and also for feedback. Physiological studies proved that how the objects are being stored and later on recognised in the primate brain and particularly in V4 area [4]. In the proposed model, we considered the layers architecture of visual cortex. Number of neuronal units in each layers are suitably calculated based upon required number of points to store the object and train the network. To represent the visual cortex area V4, the model inherited the properties of "hierarchical model for object recognition" proposed by Maximilian Riesenhuber and Tomaso Poggio in 1999 [5].

An object is recognised by its stored characteristics like its size, curves, bends, edges and edge-angles due to the more information contents in these regions [6]. In the proposed neurocomputational model, during the training cycles, the network is trained for these information of the object in a layer of neuronal units. In the testing period, the network tries to recognise the previously learned object characteristics and show the output in terms of the weights of the units.

2. LEABRA

It is a collection of computational formalisms for developing cognitive models that make contact with both observable behavior and detailed biological mechanisms. LEABRA models are constrained by our knowledge of processes at the level of membrane channels and individual neural functioning and also by our knowledge of gross brain anatomy and the role of various neurotransmitter systems. LEABRA is of particular interest because it incorporates many of the mechanisms that have appeared in the history of connectionist research. Its recurrent activation dynamics allow it to exhibit pattern completion and soft constraint satisfaction performance akin to that seen in Hopfield networks, other attractor networks, and spreading activation models.

Synaptic weight learning in LEABRA includes a Hebbian learning algorithm, allowing for self-organization learning, and an error-correction learning algorithm formally related to the back-propagation of error technique. Hebbian learning is performed using conditional principal components analysis (CPCA) algorithm [7] with correction factor for sparse expected activity levels. Error driven learning is performed using GeneRec, which is a generalization of the Recirculation algorithm, and approximates Almeida-Pineda recurrent back propagation. The symmetric, midpoint version of GeneRec [8] is used, which is equivalent to the contrastive Hebbian learning algorithm (CHL). LEABRA networks can also make use of a reinforcement learning algorithm based on the role of the neurotransmitter system in learning [9]. By bringing all of these mechanisms together, LEABRA provides a single focal framework through which a wide variety of connectionist concepts. LEABRA is fully supported in PDP++.

3. Model

The characteristics of the objects like length of edges in case of cube, height and cone angle in case of cone etc. are extracted and stored in input file to pass the file as an input to the model. The function of retina is considered to construct the model more biologically realistic while collecting the properties of an object. Visual stimuli available on retina are actually capture the image properties in terms of on and off signals. Input layer passes the signals to V1 layer where the object image is enlarged or shrink as per the size of the input so that it can use the maximum number of units on V1 layer. V1 layer passes the signal to V4 layer where the actual learning takes place. In the model, the units of V4 layer are divided in 16 groups. Number of groups of units depends upon the properties of an object to memorize. In our experiment, we usually find the basic geometrical objects have not more than 16 properties to store. In the model, a group of units represent individual properties of an object. In training cycle, the events are designed to describe the object to the network. More objects are described in more than one event setup.

The objective is to select correct response in case of an object, depending on the task and sensory input. The input layer is directly connected to V1 visual cortex but the connections are not so strong to trigger the response. The V1 area units of visual cortex also need some bottom-up support from the V2 layer. The job of the V2 layer is to integrate stimulus input with the correct response selected by the V1 area units of visual cortex and on the basis of what it has learned in previous experience

.

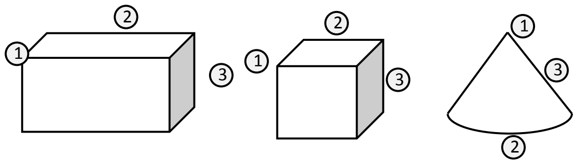

Fig 1 Objects and its' properties to learn

In the model, there are three layers, separately storing the object properties and accordingly modulating the two responses. By this the selection of correct and influential potential of units suggests the correct response. In learning phase, the objects properties, as shown in fig. 1, are stored in the form of unit weights. Properties like the angle on which three edges are connected (in fig. 1). After that it identify the edges (and in fig. 1 for cuboid and cube) and compare its' lengths. Likewise it stores other properties of the objects. The columns of V4 layer is divided into two parts, the left columns are representing the "correct" units with separate columns for response 1 and response 2 and the two right columns are representing "incorrect" units with separate columns for response 1 and response 2. The correct response columns of V4 area only projected to the layer 'internal segment of MT layer' form a direct pathway. The 'incorrect' column to the layer 'external segment of MT layer' forms an indirect pathway. GPe columns inhibit the associated column in GPi, so that striatal Go and No-Go activity have opposing effects on the GPi. At last, each column in the GPi tonically inhibits the associated column of the thalamus, which is reciprocally connected to the Premotor cortex.

The network architecture simply supports the existence of connections, but how these ultimately influence behavior depends on their relative strengths? The network starts off with random weights and representations in both the Visual cortex area V1 and V4 layers are learned. Distributed activity within each striatal column enables correct and incorrect representations to develop for various stimulus configurations during the course of training.

4. Details of the model

The model is based on Leabra framework [10] in which the units are "point-neuron" function using rate-coded output activation. There are separate excitatory and inhibitory synaptic input channels. Synaptic connection weights were trained using reinforcement learning version of Leabra. The learning algorithm includes two phases, allowing simulation of feedback effects and it is more close to behavioral phenomena than the standard error backpropagation. In the "minus phase," the network settles into activity states on the basis of input stimuli and its synaptic weights, ultimately resettles a response. The "plus phase" describes the network when it resettles. It resettles in the same way but with the change in potential of the units in layers. An increase potential level for correctly recognise and a dip for incorrect identification is applied by the V4 layer. The Connection weights are then adjusted to learn on the difference between activity states in the minus and plus phases.

PDP++ software also provides to write scripts and execute it at run time during the simulation. Since the behavior of satiation layer is somewhat dynamic i.e. we only have to increase units' weight when the model choose the correct response. To include this behavior, we added some scripts to manipulate the satiation layer units' weight. An object is defined in model which contains a current value of unit weights of V1 and V4 layer. During one trial, if the model chooses a correct response, this object weight is updated and assigned to the V4 layer units so that next time the unit weights will update from its previous value.

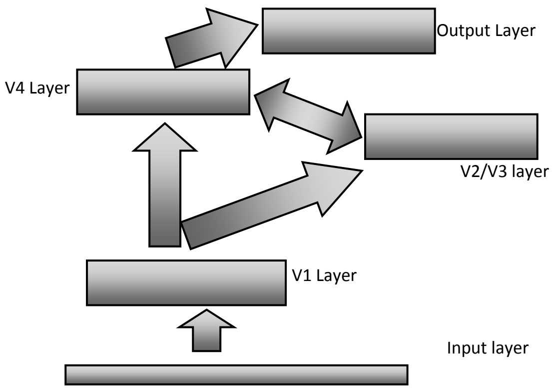

Fig. 2 Blok-diagram of the model. Rectangles represent the different layers used in the model. Arrows show the signal propagation to different layers of the model.

At the start of each trial in the learning phase, a potential level of neuronal units is maintained by setting the V1 layer units to be semi-active with activation value 0.5. At the initial stage of training, the network selects a random response, dictated by random initial weights in visual cortex. If the response is correct a potential level of V4 neuronal units set to have an activation value of 1.0 i.e. high firing rate. This potential shift causes a more logical correct representation in the V1 to associate with the correct response that was just selected. In case of incorrect response, a dip in potential of units occurs with all V4 units set to zero activation results the network to learn incorrect response behavior.

5. Experiment

The main objective of the experiment is to establish the proper connection between the visual cortex layers so that the model can depict the realistic storage and retrieval of the impression of objects. In the human brain there are about 10^{11} neurons where each one is connected to roughly 10^3 to 10^4 other neurons, i.e., there are more than 10^{14} interconnections called synapses [11]. In cognitive neuroscience, it is too difficult to model more than one brain regions if one region is affected by modulating functions in other brain regions. This will affect the activities of one layer to other in visual cortex which is very important for many aspects of cognition [12]. Some researchers consider that the function to encode one image object leads to the mappings of similar looks like objects (e.g., [13]). Others believe that different modulatory role of the visual cortex to facilitate or suppress stimulus-response like associations that are represented in the visual cortex layers [14]. Layer architecture gives the role of filtering the object impression to recognise later-on. We restrict the functioning of the model to recognize and retrieval of the objects. It does may affect the other functioning of the brain but it is not included in the model. A model proposed by Serre T. et. al. [15] gives the clear idea about the functional dependency of the bran processes in case of unsupervised learning. In the proposed model, the output is simply the finding of matching recognized pattern as per the provided input. For experimental purpose, the number of units in input layer and output layer are being taken to explain the objects in terms of angle, edge length and comparison of edge length. Likewise, the units in V4 layer is being taken to learn the object properties.

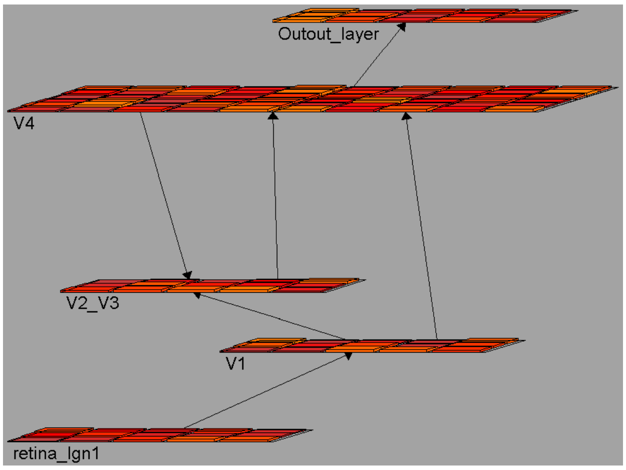

Fig. 3 Neurocomputational Model designed with the help of pdp++.

During learning phase, we restrict the number of properties of an object to pass as an input to the model. Simulation is executed for the number of inputs 1, 2, 3, 4, 5 and 6. As shown in the fig. 3 , for lower inputs, the number of correct response selection is less because the model get could not differentiate two objects having common object properties like the angles of the edges if a cubes and cuboids. For each correct response selection of a learned object, the network show errors in initial trials and it learns quickly. For higher number of inputs, the number of correct response increases because the network can easily distinguish the objects based on the different properties like the angles of a pyramid and cubes. When the number of inputs increases, it has been observed the slow learning rate due to large settlement of large number of neuronal units and their weights.

6. The Results

The model generated results are being stored in the file and compared with the input patterns. The model gives the comparative results at the output layer as per inputs. The error decreases while increasing the number of properties to learn. For example, if we try to learn the network based on only the edge length of the cube then during the test cycle, the model gives the same result in case of cude and cuboid. We compared the error rate i.e. the number of errors occurred while learn the network with different number of properties of an object.

$$E_{network} \; 1 = \left[\sqrt{\sum_{i=1}^{n}(I_i - O_i)^2}\right]/n$$

Where:

$E_{network}$ *is the RMS difference between input and output units*

I_i *is the* i^{th} *units weight of input layer*

O_i *is the* i^{th} *units weight of output layer*

n *is the number of units compared between input and output layer*

Equation 1 calculates the error of the network ($E_{network}$) for a particular number of properties. It is based on the RMS of the differences between input and output unit weights. n denotes the number of units to compare in input and output layer. Error rate is the number of error calculated as above per training for a particular number of properties. Fig. 4 shows a graph comparing the number of errors increases when the number of properties increases, the network gives low error rate.

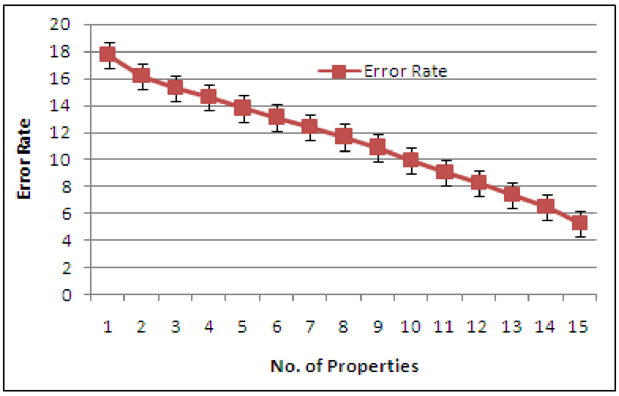

Fig. 4 Comparison of No. of properties to learn and error rate.

After successfully train the network, which is detected by mentoring the error rate, the network is ready to recognize the objects. In the first step the number of properties is to be chosen to test the network. Then a new event has to be maintained which defines the input pattern. This event is then associated with the environment of the network. This gives us the new activation values for the units of output layer.

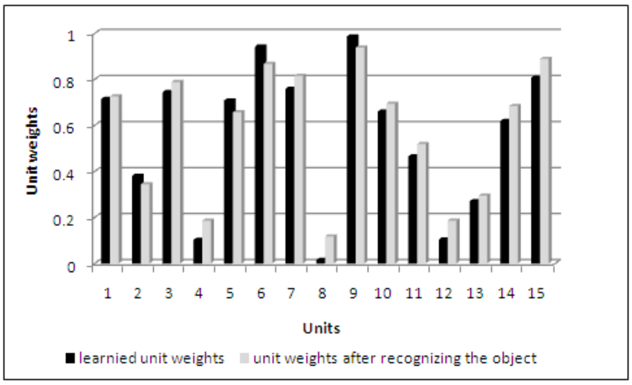

Fig. 5 Comparison of output layer unit weights of the learned network and weights after recognizing the object

By experimenting with the model with different number of properties and different number of epoch for the basic geometrical objects, the model gives us similar unit weights as it gave at after learning the same object. Fig. 5 shows the relation between the weights of the units at the time of learning the cube object and at the time of recognizing the same cube object. It clearly shows that the object properties are being identified by the network. One more experiment was performed on this model. If we try to give the properties of an object which was not learned by the network then network try to match the properties with the closely related object which the network has learned previously. Like in our network, if we train the network to recognize the object. Fig. 6 shows the comparative graph between the learned properties of a cuboid and corresponding recognized properties of an unknown object cube. Graph clearly shows that the elevated errors for the properties for which the network does not have any learned information. Network searches the maximum number of matched properties at the time of recognizing the object. In case of cube and cuboid, the mismatched property is the comparison of edge length. Since the learned property of the cuboid is the dissimilar lengths of edges.

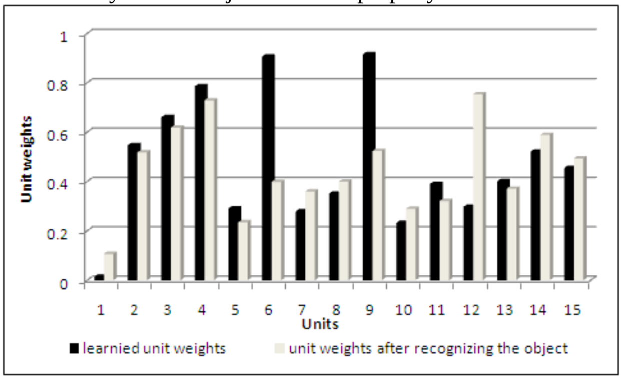

Fig 6 Comparison of output layer unit weights of the learned network for various object but not cube and weights after recognizing the similar cuboid object.

When a cube is given to the network, it does not recognize the edge length properties and responsible units give the low weights of the units. Unit number 1, 6, 9 and 12 were responsible to store the property of dissimilar edge length but due to cube which has the same edge length, these units gives the dissimilar weights as compared to the learned property.

7. Conclusion

There are some limitations of this model. Since we only have used the 15 units in each layer of input, V1, V2/V3 and output layer, the model can only recognize very limited number of basic geometrical objects. Objects should have clearly distinguishable properties otherwise the model recognizes the closely related similar object. While defining test event i.e. the properties to recognize, it has to be defined for the predefined units for corresponding object properties.

To improve the model capabilities to recognize a large number of objects, the dimensions of the network has to be increased. In that case to train the network a higher epoch is required. To test the network, the event generation should also be more precise so that the network can give low error results. Further expansion of this model would lead to understand other behavioral consequences while recognizing old memories.

8. References

[1] Yukiyasu Kamitani & Frank Tong, "Decoding the visual and subjective contents of the human brain," Nature Neuroscience Vol 8, pp 679-685, May 2005.

[2] Born, Richard T., and David C. Bradley. "Structure and function of visual area MT." Annu. Rev. Neurosci. 28 (2005): 157-189.

[3] Pack C.C., Born R.T. and Livngstone M.S., "Two Dimensional Substructure of Streo and Motion Interaction in Macaque Visual Cortex," Neuron, Vol 37, pp 525-535, February 6, 2003.

[4] Serre T, Wolf L, Poggio T, "Object recognition with features inspired by visual cortex," Proc IEEE Conf Comput Vision Pattern Recognition 2:994–1000: 2005.

[5] Riesenhuber, Maximilian, and Tomaso Poggio. "Hierarchical models of object recognition in cortex." Nature neuroscience vol 2 pp 1019-1025: 1999.

[6] Gauvrit, Nicolas, Hector Zenil, and Jean-Paul Delahaye. "Assessing Cognitive Randomness: A Kolmogorov Complexity Approach." arXiv preprint arXiv: pp 1106.3059, 2011.

[7] Olshausen BA, Field DJ. Energence of simple-cell receptive field properies by learning a sparse code for natural images. Nature. 381:607–609: 1996.

[8] O'Reilly, R. C. "The Leabra model of neural interactions and learning in the neocortex." PhD thesis, Carnegie Mellon University, Pittsburgh, PA. (1996)

[9] O'Reilly, R. C. "Biologically plausible error-driven learning using local activation differences: The generalized recirculation algorithm." Neural Computation, vol 8, pp 895–938 (1996)

[10] O'Reilly, R. C., & Munakata, Y. "Computational explorations in cognitive neuroscience." Cambridge, MA: MIT Press: (2000)

[11] Flanagan, John G. "Neural map specification by gradients." Current Opinion in Neurobiology 16 (2006): 59-66.

[12] Nieoullon, A., "Dopamine and the regulation of cognition and attention." Prog. Neurobiol. vol 67 (1), pp 53–83: 2002

[13] Packard, Mark G., and Barbara J. Knowlton. "Learning and memory functions of the basal ganglia." Annual review of neuroscience 25.1, pp 563-593.(2002)

[14] Riesenhuber, Maximilian, and Tomaso Poggio. "Models of object recognition." Nature neuroscience, vol 3, pp 1199-1204: (2000)

[15] Serre, Thomas, Aude Oliva, and Tomaso Poggio. "A feedforward architecture accounts for rapid categorization." Proceedings of the National Academy of Sciences 104.15 pp 6424-6429, 2007.