ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

Specialization within cortical models: An application to causality learning

Hervé Frezza-Buet, Frédéric Alexandre

LORIA/INRIA-Lorraine, France
Herve.Frezza@loria.fr, Frederic.Alexandre@loria.fr

Abstract. In this paper we present the principle of learning by specialization within a cortically-inspired framework. Specialization of neurons in the cortex has been observed, and many models are using such "cortical-like" learning mechanisms, adapted for computational efficiency. Adaptations will be discussed, in light of experiments with our cortical model addressing causality learning from perceptive sequences.

1. Introduction

Learning through specialization as been reported as a characteristic of cortical plasticity. Indeed, biological data show that circuits of cortical neurons receiving the same information can be tuned by learning, in order to perform different functions on this information. Thus, cortical computation combines both the distributed representation through population coding and local representation through specialization mechanisms.

In this paper, neural models inspired from biological data and grounded in the specialization paradigm will be discussed. Subsequently, a mechanism detecting on-line causality sequences from perceptive events will be presented, which illustrates the convenience of specialization for sequence management with neural networks.

2. Learning through specialization within cortically-inspired models

Some models involving cortical learning through specialization are presented in this section. Models using a fixed number of neurons, each of them specializing through learning, are called *static models*. On the other hand, some other models, the *incremental models*, start learning with few units, learning involving a recursive splitting of units into more and more specialized units. In those models, a unit represents an assembly of cortical cells and, accordingly, it can split (the split unit representing the specialization of a wide subset of cells in the assembly).

ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

2.1. Static models

Static models are more plausible with regard to the physiology of the cortex than incremental ones. In a cortical model proposed by Burnod [1], the cortex is viewed as a bidimensional surface tiled with columns. Lateral connections between columns allow adjacent columns to couple or uncouple so that they either fire respectively synchronously, or not. Within a cortical area, columns receive the same kind of information. If all columns are half excited, learning involves splitting the area into excited columns (those specialized to detect the current pattern of information) and non excited columns (uncoupled from the previous ones due to lateral inhibition), so that learning increases the contrast of activities [1] in the cortical area. The same idea can be found in Kohonen's model of low level cortical processing for perception [9]. This model, the Self Organizing Map (SOM), consists of a set of neurons all receiving the same input. According to the weights of the connections, neurons fire preferentially for a particular state of the input space. Learning in the SOM consists of tuning (specializing) the neurons so that they reflect the variability of the input. After learning, a given input activates only one tuned neuron, activations within the map are highly contrasted. The topological properties of the model are supported by observations concerning the detection, by cortical neurons, of visual stimulus orientation. Neurons in the visual cortex of monkeys have been found to be locally organized according to the angle of the stimulus [8]. Other biologically plausible static models can be found in the literature, like the cortical model of Guigon [6]. This deals with associative properties of the posterior cortex, as well as the temporal processing of action within the frontal cortex.

Generally, these models are grounded in a distributed computation. The specialization is provided first by random weights and/or random activation that initially favor one of the neurons to fire. Subsequently lateral inhibition prevents other neurons from firing for the same input pattern, thus enhancing contrast, and providing a higher representation capability.

2.2. Incremental models

Although biologically plausible, static models are often difficult to implement on a computer. First, the high number of required units is memory and time consuming. Second, distributed inhibitory and activation mechanisms are often difficult to design and programmers end up using tricks (for SOM, an explicit winner-take-all mechanism and decreasing Mexican-hat-shaped lateral inhibition simplify the mechanism given in [9], see [7]).

Using incremental models is therefore convenient, optimizing the number of units to manage and simplifying inhibition management. For example, Growing Neural Gas [5] can be viewed as an incremental adaptation of SOM, and an incremental model inspired from Burnod's model has successfully been applied to phoneme recognition [2], and to word recognition [3].

ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

3. Application to causality chain learning

3.1. Framework and purpose

In this section we present an incremental cortical model for the detection of causal sequences of events. The units of the model are automata, having several state variables and executing functioning cycles synchronously. During a cycle, each automaton computes new values of its state variables, according to the state of other automata and its inputs. Each unit is connected to every other (which is locally realistic for columns inside cortical areas [1]) and is tuned on a given configuration of the perceptive input. When event e occurs with intensity $i_e \in [0..1]$, the unit E tuned on that event is said to be *excited*, and it stores i_e in a state variable E^{exci}. Another state variable E^{rec} is set to 1, coding that the event is occuring. At each cycle, the recency E^{rec} is decreased by a constant σ, until e occurs again or E^{rec} reaches 0 (then E^{exci} is reset to 0). Within a cortical framework, units can be *called* [1], meaning that the event they are tuned on is needed. In our model, this call is coded with a boolean state variable E^{call}, which is set to 1 for the unit initially needed. If a unit E is called ($E^{\text{call}} = 1$) and then excited due to the external world ($E^{\text{rec}} = 1$), the unit is said to be *satisfied* since the event the cortex was asking for occurs. Call propagation generally aims at triggering actions when a call reaches the motor units [4]; however, we only focus here on the propagation of calls through units having causal relationships.

3.2. Causality detection

Let E_i be a unit tuned on event e_i and let E_i be called ($E_i^{\text{call}} = 1$); as such, it can be viewed as a *goal unit*. Learning aims at finding, among all other units E_j (that have a lateral connection with E_i), the ones that fire ($E_j^{\text{rec}} = 1$) "often" before the satisfaction of E_i ($E_i^{\text{rec}} = E_i^{\text{call}} = 1$). The corresponding event e_j is then considered as a possible cause of the occurrence of e_i, the most recent e_j being used for step by step causality chain construction. If a connection between E_i and E_j exists, it is weighted with w_{ij} (initially 0, and kept in $[0..1]$), and associated with a flag δ_{ij}. The following processing is performed at each cycle, for each connected couple of units $(E_i, E_j)_{\exists w_{ij}}$. Parameters τ and τ' are fixed learning rates, $\leftarrow$ is the affectation and $M^i = \max_{\exists w_{ij}} E_j^{\text{rec}}$:

if $E_j^{\text{rec}} = 1$, $\delta_{ij} \leftarrow 1$.
if $E_i^{\text{rec}} = 1$ and $E_i^{\text{call}} = 1$,

case $E_j^{\text{rec}} = M^i$ and $M^i > 0$	:	w_{ij}	$\leftarrow$	$w_{ij} + \tau.E_j^{\text{rec}}.E_j^{\text{exci}}$
case $0 < E_j^{\text{rec}} < M^i$	:	w_{ij}	$\leftarrow$	$w_{ij} + \tau.(E_j^{\text{rec}} - M^i).E_j^{\text{exci}}$
case $E_j^{\text{rec}} = 0$	:	w_{ij}	$\leftarrow$	$w_{ij} - \theta$
in all cases	:	δ_{ij}	$\leftarrow$	0

else if $E_i^{\text{call}} = 1$ and E_j^{rec} reaches 0,

$$w_{ij} \leftarrow w_{ij} - \tau'.E_j^{\text{exci}}.\delta_{ij}$$

else nothing to compute.

ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

Let us detail the above algorithm for goal unit E_i. Learning occurs only when E_i is called. When an event e_j occurs ($E_j^{\text{rec}} = 1$), the flags δ_{ij} for all j are raised. If the goal E_i is satisfied ($E_i^{\text{rec}} = 1$ and $E_i^{\text{call}} = 1$), the weight w_{iJ} to unit E_J that has been excited the most recently ($E_J^{\text{rec}} = M^i$) increases, proportional to both the recency and the stored intensity of the event e_J. The weights w_{ij} connected to each other E_j are decreased if the corresponding event e_j has occurred before E_J ($E_j^{\text{rec}} < M^i$), proportionally to their incapacity to win the recency competition ($E_j^{\text{rec}} - M^i$). If a unit E_j has not been excited before the satisfaction of the goal E_i, w_{ij} decays via a leaky coefficient θ. In all cases, when the goal E_i is satisfied, flags δ_{ij} for all j are set to 0. Last, if E_i is called and the recency of a previously excited E_j reaches 0, w_{ij} is decreased, meaning that E_j has not permitted satisfaction of E_i. Using flags ensures that no satisfaction of E_i has occurred (and so removed the flag) between the time of E_j excitation and the reset of E_j^{rec}.

3.3. Making assumptions

In our model, causality detection aims at deriving assumptions. Let E_g be a goal unit and E_i a unit often excited just before E_g is satisfied. When the weight w_{gi} reaches 1, it is allowed to *assume* that the event e_i is *sufficient* to provide the occurrence of the goal event e_g. This assumption takes the form of a splitting of E_i, one of the split units being E_i itself (the *mother* unit), and the other, E_i' (a *specialized* unit). As with E_i, E_i' detects (by being excited) the event e_i. A specialized unit cannot itself split (it cannot specialize anymore), but the mother unit can split again if needed. The only restriction is that a mother unit cannot split twice because of a goal unit E_g (but it can split for E_g, and then for another E_g').

When a specialized unit E_i' is created, it is connected with weights $w_{i'j}$, to all units E_j that E_i is connected with, *except* units that split from E_i. As E_i' cannot split anymore, no weights $w_{ji'}$ are created. A specialized link is created between E_i' and E_g, so that the call activity of E_g can spread to E_i' (the value E_g^{call} at cycle t is effectively copied to E'^{call}_i at cycle $t+1$). Subsequently, calling E_g leads to E_i' being called : E_i' is the subgoal of E_g. The called E_i' can now play the role of a goal concerning other units, in order to eventually extend the causality chain further from E_i'.

The advantages of specialization (splitting) are twofold: First, the specialized link between E_i' and E_g can be weighted (let $\omega_{gi'}$ be the weight), in order to evaluate whether the satisfaction of E_i' is often followed by the satisfaction of E_g. No particular mechanism is given in this paper, we just mention here that this evaluation of the subgoal event efficiency $\omega_{gi'}$ may be different from the evaluation of the causality (weight w_{gi}). The former may not be competitive with efficiency evaluations for other units (simply measuring correlation) whereas the latter is obtained from the competition for recency (see equation 3.2.). Second, splitting prevents causality sequences that share the same event from merging. Section 3.4. is an illustration of the mechanism.

ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

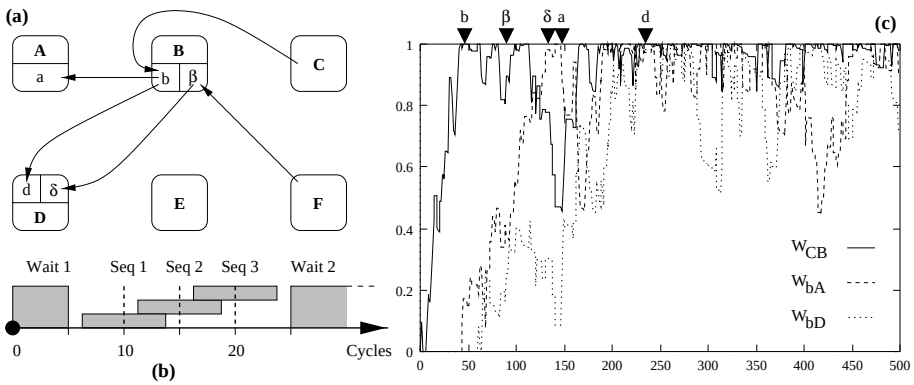

Figure 1: Experimental results. See section 3.4. for comments.

3.4. Example

To illustrate the behavior of the model, let us consider six mother units A, B, C, D, E, F (see figure 1-(a)). Training consists of many trials, in each of which a sequence of three events is presented. A trial lasts 55 cycles (see figure 1-(b): First, a unit is called (the call is maintained over the following cycles) and no event occurs for 5 cycles (`Wait 1` period). Then, the first event of the sequence is presented, during the `Seq 1` period, exciting the corresponding unit at a random cycle. The same is done for the second and third events respectively during the `Seq 2` and `Seq 3` periods (*distortion*). As the `Seq i` periods last 9 cycles and overlap by 4 cycles (see figure 1-(b)), two successive events of the sequence may occur in the wrong order (*permutation*). The call that was initially made stops 2 cycles after the presentation of the last event. The trial ends with 30 cycles where no events are presented (`Wait 2` period), before another trial begins. Moreover, each unit has a probability $p_{\text{exc}} = 0.01$ to be excited (*insertions*) at each cycle.

In each trial, one of the five sequences $S_1 = C^{\text{call}}ABC$, $S_2 = C^{\text{call}}DBC$, $S_3 = C^{\text{call}}nnC$, $S_4 = F^{\text{call}}DBF$, $S_5 = F^{\text{call}}nnF$ (n meaning "no event") is used, with respective probabilities 0.225, 0.225, 0.05, 0.45, 0,05, for each trial. After 500 trials, units in the system are as shown in figure 1-(a). Arrows represent specialized links ω_{ij} that carry out calls from goals to subgoals. The significant point here is that calls from C and calls from F *do not merge* in B, due to the specialization (splitting) of units. The figure 1-(c) shows the change of weights according to the number of trial. The times of split unit creation are indexed at the top of the graph. The parameter values are $\sigma = 0.02$, $\tau = \tau' = 0.1$ and $\theta = 0.01$. In figure 1-(c), the increase of the weight w_{CB} in order to create b is illustrated. After b is created, the call propagates from C to b, b being a subgoal. Both weights w_{bD} and w_{bA} then increase, because both events A and D are *exclusive in time* (thus they do not compete for recency), and are candidates for extending the causality chain from b (see 1-(c)).

ESANN'1999 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 21-23 April 1999, D-Facto public., ISBN 2-600049-9-X, pp. 263-268

4. Discussion

Incremental models for cortical specialization can be efficient for cortically inspired computation without simulating large neuron assemblies. The mechanism presented here provides robust *on-line* sequence management, with no separate learning and application stages. The learned regularities are used immediately by propagating calls, whilst the sequence is still able to extend. A further application of the algorithm to multi-modal sequence learning for robot control is presented in [4].

The existence of robust incremental models raises the question of the computational utility of static models. Using the causality learning example presented in this paper, we answer the question the following way: if the computational objective is clearly defined (causality learning), an incremental model may be able to focus on the property to implement, providing robust learning capabilities. However none of the present models are able to provide efficient sequence management (including robustness, sequence similarity detection, online learning and use, sequence evaluation, etc.) as the real cortex does. The challenge for designers is therefore to better understand the relationships between a static model and an analogous incremental one, in order to determine whether the computational power of a huge assembly of interconnected elementary units is attainable with models where large sets of neurons are simulated by a single unit.

References

[1] Y. Burnod. *An adaptive neural network : the cerebral cortex.* 1989.

[2] C. Dingeon, F. Alexandre, F. Guyot, and J.-P. Haton. Un autre apprentissage cortical : différencier pour généraliser. In *Proc. Neural Networks and their applications*, 305–315, 1989. In French.

[3] S. Durand and F. Alexandre. A Neural Network based on Sequence Learning; Application to Spoken Digits Recognition. In *Proc. Neural Networks and their applications*, 1994.

[4] H. Frezza-Buet and F. Alexandre. Selection of action with a cortically-inspired model. In *7th E.W.L.R*, 13–21, 1998.

[5] B. Fritzke. A growing neural gas network learns topologies. In *Advances in Neural Information Processing Systems 7*, 625–632. MIT Press,1995.

[6] E. Guigon. *Modelisation des proprietés du cortex cerebral.* PhD thesis, École centrale de Paris, 1993. in English.

[7] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the theory of neural computation.* Addison Wesley, 1991.

[8] D. H. Hubel and T. N. Wiesel. Functional architecture of macaque monkey visual cortex. *Ferrier Lecture Proc. Roy. Soc. London*, 1–59, 1977.

[9] Teuvo Kohonen. *Self-Organization and Associative Memory.* 1988.