Cortical Computation

Christos Papadimitriou*
University of California at Berkeley
christos@cs.berkeley.edu

Santosh Vempala†
GeorgiaTech
vempala@cs.gatech.edu

ABSTRACT

A computational theory of cortex necessitates a novel paradigm of exquisitely distributed computation. Here we review recent work on a primitive called *Predictive Join*, or *PJoin*, which is both plausible and useful in regards to cortical computation, and which enables a spontaneous form of unsupervised learning exhibiting many of the characteristics of brain activity. We also outline several immediate goals of a computational research program on the brain.

Categories and Subject Descriptors

F.1.1 [**Computation by Abstract Devices**]: Models of Computation; J.3 [**Life and Medical Sciences**]: Biology

Keywords

Brain; Neuroidal model; PJoin; Synchrony; Control

Introduction

Understanding the brain is the greatest scientific problem of our time — arguably of all times. Despite the current deluge of scientific discoveries regarding the anatomy and function of the brain of all kinds of animals, no overarching theory for the brain's high-level operation and the emergence of the mind appears in the horizon. As the brain is by far the most blatantly computational of all scientific objects, this quest presents a unique opportunity for key research contributions by computer scientists — and in fact, as we shall see, by students of distributed computation. The study of the brain from the computational point of view must deal with the complete disconnect between the remarkable accomplishments of brains on one side, and the triumphs of computers on the other. For example, we are not aware of a single nontrivial computational problem which has been solved in a manner that is neurally plausible, by any stretch. This disjointness suggests that the nature of the computation taking place in the brain must differ dramatically from all extant computational paradigms.

*Research supported by NSF grants CCF-0964033 and CCF-1408635, and by Templeton Foundation grant 3966.

†Research Supported by NSF awards CCF-1217793 and EAGER-1415498.

PODC'15, July 21–23, 2015, Donostia-San Sebastián, Spain.
ACM 978-1-4503-3617-8 /15/07.
http://dx.doi.org/10.1145/2767386.2767452.

The Neuroidal Model

The *neuroidal* model was proposed in 1994 by Leslie Valiant as a formal model of low-level cortical computation [2, 3]. Computational nodes called *neuroids* are connected by directed edges called *synapses* in a random directed graph. Each neuroid has a threshold as well as a state, and each synapse has a strength and a state. Communication between neuroids is extremely restricted: a neuroid i can communicate with neuroid j only if (1) there is a synapse ij; (2) i has just *fired* (see below); and (3) the sum of the strength of synapse ij together with all other strengths of synapses kj, such that k too has just fired, happens to exceed j's threshold. If this is the case, then j fires, and this is the only sense in which it can receive a communication from i. In addition, at each step the thresholds, strengths, and states of all neuroids and synapses are updated through a universal update rule, taking into account only local information — for example the new state of synapse ij can depend on the state of neuroid j but not that of i, as it is natural to think that synapse ij lives close to j. By design, the neuroidal model is a deliberately conservative model of what circuits of neurons can accomplish. It can be also seen as an extremely restricted model of distributed computation. Computation in the brain, we believe, must take place within the confines of such a model.

Asynchronous and Spontaneous Computation

How does one evaluate neuroidal algorithms — algorithms programmed in this model? What are the neuroidal analogues of time and space? Parallel time is of course an important criterion here as well; if anything, it is far more stringently crucial: Since neurons take a few milliseconds to fire, and cognitive phenomena happen within roughly a few hundreds of milliseconds, brain algorithms should take only a few dozen steps. The total number of neurons involved should not be excessive. In addition, we propose *synchrony* and *control* as quantitative measures of complexity. Synchrony could be the maximum total number of pairs of neuroids that need to fire in lockstep, while control is the longest chain of synchronized steps required for the algorithm to succeed. These are important resources to be used sparingly, since neurons are independent distant entities, and thus perfect synchrony that is widespread, or total control that goes on for too long, is unrealistic. Furthermore — and importantly — the nature and style of the algorithm must be, in a very specific and subtle sense, appropriate. This is a computational model to be programmed in modesty and restraint, without a trace of extravagance and abuse, and with minimal outside control. Ideally, neuroidal algorithms should be simple and reactive (in the dictionary, not the technical, sense: they should just respond to arising circumstances), essentially self-assembling data structures. Sophistication, cleverness, and shrewd efficiency of the kind admired and lauded in our field are no virtues in this domain.

PJoin

In [1] such a neuroidal algorithm is presented for the task of unsupervised learning. Given a pattern in $\{0,1\}^n$, with each bit represented as a set of neurons, a tree-like structure is spontaneously created which entails a representation of the pattern and includes a special "top element". If — and only if — the same pattern is presented again, the same top element will eventually fire. If another pattern is presented, it will also be learned in the same sense and manner, and with a different top element. Incidentally, all these specifications are met with high probability, not certainty, and the randomness comes from both coin flips by the algorithm and the randomness of the underlying graph. New patterns are learned by appropriately sharing substructures with preexisting ones while also adding their own, thus creating a dag. Importantly, much of the firing by neuroids is transmitted *downwards* — meaning, towards the inputs and away from the top elements. This is in agreement with many experimental findings in neuroscience noting that many more synapses travel downward in the cortical hierarchy — e.g., for vision — than up.

The main idea in this algorithm is a novel operation that we call *PJoin*, or *predictive Join*, the building block of the dag described above, and a primitive which we believe may be a productive paradigm for cortical computation in general. It extends Valiant's Join(A,B) operation [2]. Join(A,B) combines two items A and B — two sets of neurons representing two real-world ideas, such as *almond* and *butter* — to create a new item representing the combined idea. If the two original items A and B happen to fire simultaneously, then Join(A,B) will fire next. In the case of PJoin(A,B), if only one of the constituent items fires — say A fires — then PJoin(A,B) will not quite fire, but will proceed to "predict B." This entails mobilizing B (notice the downward step) which, if it is itself a PJoin of two other items C and D further below, will in turn predict those. Eventually, if an item representing a bit of the pattern being presented is predicted correctly, it will fire. If its prediction of B is eventually verified, PJoin(A,B) will fire, otherwise it will not. We show that PJoin(A,B) can be created by a neuroidal algorithm extending Valiant's algorithm for Join(A,B). We also prove that PJoin enables a very simple and natural algorithm for unsupervised learning with favorable performance, operating as described in the previous paragraph. It is this predictive ability of PJoin that enables the unsupervised learning of patterns; with ordinary Joins, a combinatorial explosion of the data structure would ensue. Furthermore, experiments with the algorithm [1] show that ensembles of patterns of length 100 can be learned almost always within fewer than 80 parallel steps per pattern presentation.

Some Research Questions

Invariants. Can PJoin help make progress on the problem of how the brain learns *invariants?* The brain is capable of a very effective and sophisticated clustering-cum-interpolation operation, whereby several views of the same real-world object — say, the face of a friend — together with the friend's voice or name, are all collected into a single higher-order representation. The mechanism of invariants has been for decades an elusive mystery in brain science.

Prediction. PJoin should be seen not as a concrete scientific prediction of something that actually happens in the brain, but as a possible stance in brain science: is it possible that the brain's success is the result not of elaborate and specialized algorithms, but of very simple and rough primitives available and applied at an astronomical scale, and, through sensors and actuators, reacting to and interacting with an interesting world?

Grunge. Even if primitives like PJoin are a large part of the brain's computational story, they are unlikely to exist as precise and clean as an algorithms designer would have wished them. Can there be an algorithmic theory of amorphous and chaotic computation that will better approximate what happens?

Organic environments. Speaking of which, there is little doubt that human brains would accomplish much less on a boring planet. Animal brains perform as they do by sensing complex environments and affecting them in response. And yet, we evaluate our Learning Theory algorithms on i.i.d. samples, often from uniform distributions. What would a mathematical theory be, of "organic distributions," ones that generate the kind of environments in which brains thrive and evolve?

Language. Among all organic environments, language is unique in that it was created by us less than 100,000 years ago — when our brain was essentially the same as it is now. Presumably, language evolved by exploiting our brain's capabilities, it has adapted so that baby brains can learn it and adult brains can communicate their thoughts effectively through it. What can language teach us about our brains? Incidentally, PJoin, together with another similar "clustering" primitive that could be called PLink, seem apt primitives for learning and representing grammar.

Motor control. The most basic task accomplished by the brian is movement in the physical environment — walking, balancing, self-defense etc. — much of which is learned behavior via complex coordination and continuous feedback from the external world, and with significant variations from one brain to another. What is a plausible mechanism to explain this amazing and ubiquitous ability?

1. REFERENCES

[1] C. H. Papadimitriou and S. S. Vempala. Cortical learning via prediction. In *Proc. of COLT*, 2015.

[2] L. G. Valiant. *Circuits of the mind*. Oxford University Press, 1994.

[3] L. G. Valiant. Memorization and association on a realistic neural model. *Neural Computation*, 17(3):527–555, 2005.