Hierarchical Temporal Memory

Comparison with Existing Models

Version 1.01

The purpose of this document is to compare HTMs with several existing technologies for modeling data. HTMs use a unique combination of the following ideas:

- A hierarchy in space and time to share and transfer learning
- Slowness of time, which, combined with the hierarchy, enables efficient learning of intermediate levels of the hierarchy
- Learning of causes by using time continuity and actions
- Models of attention and specific memories
- A probabilistic model specified in terms of relations between a hierarchy of causes
- Belief Propagation in the hierarchy to use temporal and spatial context for inference

Many of these ideas existed before HTMs and have been part of some of the models we describe below. The power of HTM comes from a unique synthesis of these ideas.

The No Free Lunch Theorem [6] asserts that no learning algorithm has an inherent advantage over another learning algorithm for all classes of problems. What matters is the set of assumptions an algorithm exploits in order to learn the world it is trying to model. It is important to keep this distinction in mind while evaluating other technologies. A model that is more general does not necessarily make a better learning system: more general models are typically harder to learn. Specific models, on the other hand, risk being too finely tailored to one application, at the expense of other applications. In order to construct a learning system that can be applied to a variety of situations we need a set of assumptions that are general enough to apply to a large class of problems but are specific enough to match characteristics of real-world data and thereby allow efficient learning. Through evolution, biology has discovered such a set of assumptions. This set of assumptions can be summarized as hierarchy in time and space. By assuming that a hierarchy in time and space exists in this world, efficient algorithms can be built to exploit that structure to learn about the world. This is what neocortex does. This is what HTMs do, too.

In the following sections we will compare HTMs with many existing technologies. We will use this comparison to highlight what we have learned from these models. We divide the discussion into three parts, to organize three of the major ways that existing models differ from HTMs. However, these categories should not be taken as mutually-exclusive, since most algorithms differ from HTMs on multiple fronts.

- General-purpose probabilistic models
- Non-generative models
- Empirical neurobiological models

General-purpose probabilistic models

There are many classes of probablistic models which are used to analyze statistical relationships among a set of variables. They neither specify nor require any special meaning on the variables – which can represent anything from disease states to words in a spam email. Similarly, they neither specify nor require a particular statistical relationship among the variables, though they work best when they can exploit conditional independencies among the variables. These models are merely efficient ways to represent and perform computations on potentially complicated probability distributions.

Two such classes of models are Bayesian networks and energy-based models.

Bayesian networks [7,8] employ a directed acyclic graph as the basic structure on which the independence assumptions about probability distributions are specified. Many other statistical models (for example, Hierarchical Hidden Markov Models) can be expressed as Bayesian networks.

Energy-based models [12] (EBMs) are not technically probabilistic models, but are a very close cousin. There is generally a rough equivalence mapping between an EBM and a probabilistic model, but EBMs lend themselves to some convenient forms of hand-waving for a large class of problems.

While these models are excellent tools for probabilistic analysis, they are, alone, far too broad to solve difficult real-world problems. HTMs do not violate the fundamental principles of these models, but rather they take these models further, by making additional assumptions about the nature of the world – exactly the unique list of ideas listed in the introduction to this paper.

They should be viewed not as rivals to HTMs, but *tools in our toolbox* as we try to solve difficult statistical problems.

Many researchers have sharpened general-purpose probabilistic models to make them more useful for a class of problems. A few deserve special consideration, because they share some aspects with HTMs.

Hierarchical Hidden Markov Model [4] comes closest to the way HTMs model time, modelling the nested structure of time in a hierachy. However, the hierarchy that is exploited in HHMMs is only in one dimension (usually time).

HTMs have a hierarchy in space and time. This gives HTMs several unique advantages while learning about the world. Moreover, the theory of HTM includes provisions for using action and attention to learn the world.

Boltzmann Machine and Helhmholtz Machine [3,5] are abstract energy-based models with a neural instantiation. They typically use stochastic sampling to learn a probability distribution. They seek to solve a very non-convex optimization problem, and have historically struggled with the problem of getting "stuck" in mediocre solutions.

These machines do not include the temporal aspects of data in the model and do not make any assumptions about hierarchy. They typically work with the assumption that learning a probability distribution of the world is the essence of learning and therefore do not include discovering causes, models of action and models of attention.

Non-generative models

Many learning algorithms do not concern themselves with building a model that can *describe* the input, but but rather try to "skip" directly to mapping inputs to the correct answer, where the correct answer is provided by some external source. Such models are called *discriminitive* models. They are typically supervised, whereas HTMs are fundamentally unsupervised. (Though we frequently apply supervision at the top of a hierarchy, this is not required, and almost all learning is performed unsupervised). Furthermore, it is the ability of the HTM to *generate* data that allows it to do things like predict forward in time.

Support Vector Machines [1] (SVMs) are an efficient way to find boundaries in a high-dimensional space that separate the various examples into their labelled categories. They do not make any assumptions about the hierarchical or temporal organization of the world and hence cannot exploit these properties for efficient learning. Since the underlying model of SVMs are discriminative and not generative, they cannot be used to predict forward in time.

Like Bayesian networks, SVMs should be viewed as another tool in our toolbox – not a viable replacemnt for an HTM. For example, SVMs are an excellent candidate for the last stage of classification in a top-level HTM node.

"Classic" Neural Networks [10], eg. multi-layer perceptrons, are supervised learning models that are typically trained with an algorithm known as "back-propagation". (We use "classic" to differentiate from its newer forms, like the Boltzmann Machine, which have stronger generative semantics.) Classic neural networks are generally not thought of as generative models. Although some instantiations of neural networks use space and time, they do not exploit temporal coherence as HTMs do. Neural networks generally require a large amount of data to train and often struggle with over-fitting.

Other models are not discriminitive, but also lack generative semantics. One such example is Slow Feature Analysis.

Slow Feature Analysis (SFA) [11] is one of the first models to demonstrate that it is possible to learn invariant features in a hierarchy using temporal slowness as the underlying principle. HTMs use the same principle for learning in a hierarchy and hence share many properties with SFA. However, SFA does not offer a way to generate data and could not be used to predict forward in time. In addition, the SFA algorithm was not designed to be highly scalable like HTMs.

SFA should be viewed as a motivating concept, which is similar to that which guides and inspires HTM.

Empirical neurobiological models

Researchers have tried to directly implement observed neurobiological behavior into application-specific model. One example is the HMAX model. The major difference is that *learning* plays a secondary role, if any, in the model's performance: instead, hand-crafted features are directly provided by the researcher.

HMAX model [9] is a biologically realistic model of vision proposed by Riesenhuber and Poggio. This model has many similarities with HTMs. The HMAX model uses a hierarchical structure that is similar to HTMs, and the feature set that the HMAX model uses at different levels resembles the coincidences and temporal groups an HTM would learn when exposed to sequences of moving object images. However in the present HMAX model, the basic features (equivalent to coincidences in HTMs), and the feature groups (equivalent to temporal groups in HTMs) are obtained using human intervention.

Moreover, the HMAX model does not have a generative model behind it. The inference mechanism in the HMAX model is not probabilistic as it is in the HTMs. The HMAX model does not have feedback connections and cannot predict forward in time. All of these things are possible with an HTM.

Since an HTM ends up learning similar properties to those put into the HMAX model, the HMAX model should be viewed as evidence supporting the applicability of HTMs.

References

[1] Burges, C. J. C. (1998), 'A Tutorial on Support Vector Machines for Pattern Recognition', *Data Mining and Knowledge Discovery* **2**(2), 121-167.

[2] Charniak, E. (1991), 'Bayesian Networks without Tears', *AI Magazine* **Winter**, 51-63.

[3] Dayan, P.; Hinton, G.; Neal, R. & Zemel, R. (1995), 'The Helmholtz Machine', *Neural Computation* **7**(8), 889-904.

[4] Fine, S.; Singer, Y. & Tishby, N. (1998), 'The Hierarchical Hidden Markov Model: Analysis and Applications', *Machine Learning* **32**(1), 41-62.

[5] Hinton, G. & Sejnowski, T. (1986),Learning and relearning in Boltzmann machines'Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume I: Foundations', MIT Press, Cambridge, Massachusetts, pp. 282-317.

[6] Ho, Y. C. & Pepyne, D. L. (2002), 'Simple Explanation of the No-Free-Lunch Theorem and Its

Implications', *Journal of Optimization Theory and Applications* **V115**(3), 549--570.

[7] Murphy, K. P. (2002),'Dynamic Bayesian Networks: Representation, Inference and Learning', PhD thesis, University of California, Berkeley, Computer Science Division.

[8] Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, San Francisco, California.

[9] Riesenhuber, M. & Poggio, T. (1999), 'Hierarchical models of object recognition in cortex', *Nature Neuroscience* **2**(11), 1019-1025.

[10] Ripley, B. D. (1996), *Pattern Recognition and Neural Networks*, Cambridge University Press, New York.

[11] Wiskott, L. & Sejnowski, T. (2002), 'Slow Feature Analysis: Unsupervised Learning of Invariances', *Neural Computation* **14**(4), 715-770.

[12] LeCun, Y.; S. Chopra; R. Hadsell; M. Ranzato, F. Huang, “A Tutorial on Energy-Based Learning”, in *Predicting Structured Outputs*, Bakir et al. (eds), MIT Press 2006.