**Hierarchical Temporal Memory: A Summary**

Hierarchical Temporal Memory (HTM) is a new computing paradigm that replicates the structure and function of the human neocortex. HTMs can perform tasks which have been easy for people but hard for computers - for example, recognizing objects, making predictions, understanding speech, or navigating in a complex environment. HTMs promise to approach or exceed human performance in many cognitive tasks.

**HTM Networks Mirror the Hierarchical Structure of Real-World Objects**

The structure of an HTM Network mirrors the nested hierarchical structure of objects in the world. Most larger objects are made up of smaller objects. A neighborhood consists of houses, roads, and schools, etc.

The hierarchical structure of the world is temporal as well as spatial. Language, for example, is a hierarchical temporal sequence. A speaker expresses an idea over time by combining consonants and vowels to make syllables, syllables to make words, etc.

Most environments such as markets, traffic, and human interactions have both spatial and temporal hierarchical structure. Because HTM Networks have a hierarchical structure, they could potentially model such environments.

**The Structure of an HTM**

HTMs consist of a multi-level hierarchy of nodes in an inverted tree structure (see Figure 1, where each rectangle represents a memory node).   
  
  *Figure 1 - Symbolic Representation of an HTM:*

Information flows up and down. Nodes in Level 4 receive sensory data from the environment. Nodes in Levels 2 and 3 exchange information with a parent node above and with child nodes below.

**How an HTM Works**

An HTM receives direct sensory input from its world. It builds a model of its environment by discovering the causes of sensory data and observing how those causes behave over time. To build the model, it looks for correlations between data points that are adjacent in space and time.

Each node in the lowest level of the hierarchy receives raw sensory data related to some quantity in a limited region of the environment. For example, in a machine vision application, a node in the lowest level might receive signals from a few camera pixels within a large CCD array. If the purpose of the HTM is to predict earthquakes, a node in the lowest level might receive data from adjacent pressure sensors along a fault line.

An HTM Network can receive raw input data for any physical quantity that sensors can measure. It can also receive data from a computer file. For example, in an HTM designed to predict behavior of the stock market, a node in the lowest level might receive economic data about companies in a particular industry.

In most applications, the sensory data entering each lowest-level memory node changes rapidly. Each node receives updated information at regular intervals; the interval period depends on the application.

*Discovering Spatial and Temporal Patterns*

Suppose each node in Level 4 receives as its input the state of 16 (bright or dark) binary pixels in a 4 x 4 array within a larger array. The number of possible patterns of bright and dark pixels in a 4 x 4 array is about 64K. Some patterns tend to recur, such as those in Figure 2, which look like edges and corners.

  *Figure 2 - Commonly Occurring Spatial Patterns:*

The node identifies each commonly occurring pattern as a "cause;" i.e., it infers that an external event must be causing the pattern to occur. The node gives each cause a name and stores the names in memory. The node ignores non-repeating patterns.

Each node in Level 4 also searches its input data for recurring temporal patterns or sequences, such as those in Figure 3. Figure 3a shows an edge moving through space horizontally; 3b shows a corner moving diagonally from upper left to lower right. The node associates each temporal sequence with a cause, assigns a number identification (ID) to each cause, and stores the ID in memory.

  *Figure 3 - Commonly Occurring Temporal Sequences:*

*Establishing and Reporting Beliefs*

Every node establishes beliefs about what happens when the HTM Network receives new input data. For example, every node might create a two-column table in which the left column lists the recurring spatial patterns it has identified and the right column lists the probability that each spatial pattern is present. We call such table of spatial patterns a belief.

Each node also creates a second table in which the left column lists recurring temporal sequences and the right column lists the probability that each sequence is occurring. This table is also called a belief.

During each cycle, each child node reports its belief about temporal sequences in the environment to its parent node. Depending on the application, each parent node can also send downward to its child nodes its belief about temporal sequences in the environment.

*Nodes at Higher and Lower Levels of the Hierarchy*

All nodes at all levels in the hierarchy run the same learning algorithm: they search input data for recurring temporal sequences and spatial patterns. Nodes at higher levels do not receive sensory data, but rather the beliefs of their child nodes. For nodes at higher levels, spatial patterns consist of commonly occurring combinations of beliefs that their child nodes simultaneously report. The temporal sequences consist of recurring changes in those beliefs.

As a result, nodes at different levels deal with different phenomena. At lower levels of the HTM, nodes deal with simple events occurring in lower levels of the hierarchy of its environment. These phenomena change quickly and occur in small regions of space. At higher levels of HTM, these higher level nodes oversee the lower level nodes and so receive influences from a broader range of sensory data. They deal with more complex phenomena (sequences of sequences) that occur in higher levels of the hierarchy of its environment. These phenomena tend to change more slowly and to occur over larger regions of space.

*Basic Functions of an HTM*

The HTM Network as a whole performs two key functions:

* *Discover causes.* Causes are persistent and repeating structures in the world. An HTM discovers causes by examining its input data to identify recurring spatial and temporal patterns.
* *Infer novel causes in its environment.* An HTM classifies its input data relative to known causes that it has identified. When an HTM sees a novel input, it determines not only the most likely high-level cause, but also the hierarchy of sub-causes.

Depending on the application, an HTM could also perform these additional functions:

* *Make predictions.* An HTM compares changes in the incoming data with known sequences, looks for similarities, and predicts events based on the way sequences commonly evolve.
* *Direct behavior.* An HTM could also operate a broad range of systems.

An HTM Network resembles a Bayesian Probability Network, in which the nodes constantly share information. HTMs exploit a variation of Belief Propagation, an iterative mathematical technique, to force the entire network to settle quickly on a set of mutually consistent beliefs.

**Types of Problems Which HTMs Can Solve**

A suitable application for HTMs would satisfy these conditions:

* The domain you want to model has an inherent hierarchy.
* The goal of the HTM is to discover causes in the real world, infer causes of novel input, make predictions, or direct behavior.
* The problem domain generates data related to the causes you want to investigate. The data flow and patterns change continuously through time, while its underlying causes remain relatively stable.
* The data have some spatial and temporal correlation.
* The problem domain provides a large set of training examples organized in temporal sequences.

To read the complete version of this whitepaper, please see [HTM: Concepts, Theory and Terminology](http://web.archive.org/web/20100105020530/http:/numenta.com/Numenta_HTM_Concepts.pdf)

**Relevant Links**

* [Numenta Community Wiki](http://web.archive.org/web/20100105020530/http:/numenta.com/mediawiki/index.php/Community_Wiki)
* [Forum: HTM Theory](http://web.archive.org/web/20100105020530/http:/numenta.com/phpBB2/viewforum.php?f=6)
* [HTM Algorithms](http://web.archive.org/web/20100105020530/http:/numenta.com/for-developers/education/algorithms.php)