[Overview of the HTM](https://www.taoeffect.com/other/nupic/" \l "overview-of-the-htm)

In these sections we’ll give an overview of the essential concepts necessary to have a basic understanding of the HTM. We’ll also consider a few ways in which they compare with other types of artificial neural networks (ANNs). Some of the topics discussed in Numenta’s [white paper](http://numenta.org/resources/HTM_CorticalLearningAlgorithms.pdf), including reasons for various design decisions, performance considerations, etc., won’t be covered. Prior exposure to basic ANN concepts is assumed.

[‘H’ is for ‘Hierarchy’](https://www.taoeffect.com/other/nupic/#h-is-for-hierarchy)

The HTM has a hierarchical topology on account of hierarchies observed in some naturally occurring neural networks, such as those observed in the brain.

A hierarchical topology is also useful because it allows for the mapping of highly detailed (and possibly noisy) data, to progressively more stable and abstract concepts in the context of limited computer memory. The graphic below was taken from Numenta’s white paper and illustrates four HTM *regions* stacked on top of one another in a hierarchy. The aforementioned “noisy data” (pixels from a webcam, bits and bytes from an mp3, etc.) are fed into the bottom region of the hierarchy:

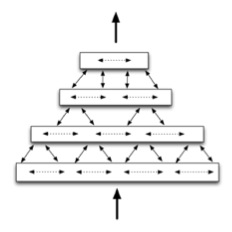


Figure 1.1: “Simplified diagram of four HTM regions arranged in a four-level hierarchy, communicating information within levels, between levels, and to/from outside the hierarchy”

**Cells**

An HTM can be composed of just one region, or several regions with data being fed through them. The region itself is comprised of interconnected *cells* that can be in one of three states:

1. Active from feed-forward input.
2. Active from lateral input (a prediction)
3. Inactive.

The cells are stacked on top of each other in vertical columns forming a three dimensional grid:

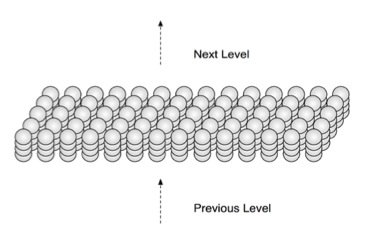


Figure 1.2: “This HTM region, including its columnar structure, is equivalent to one layer of neurons in a neocortical region.”

**Sparse Distributed Representations**

Recall that the purpose of a region is to map (potentially noisy) input-data to more stable abstract concepts. These “abstract concepts” actually have very non-abstract representations, namely the on or off state of cells within the region.

HTM regions have “sparse distributed representations”. Numenta’s whitepaper defines this term as follows:

“Sparse” means that only a small percentage of neurons are active at one time. “Distributed” means that the activations of many neurons are required in order to represent something. A single active neuron conveys some meaning but it must be interpreted within the context of a population of neurons to convey the full meaning. HTM regions also use sparse distributed representations.

I think the paper did a fine job explaining how this works in practice with the following example:

For example, a region might receive 20,000 input bits. The percentage of input bits that are “1” and “0” might vary significantly over time. One time there might be 5,000 “1” bits and another time there might be 9,000 “1” bits. The HTM region could convert this input into an internal representation of 10,000 bits of which 2%, or 200, are active at once, regardless of how many of the input bits are “1”. As the input to the HTM region varies over time, the internal representation also will change, but there always will be about 200 bits out of 10,000 active.

[‘T’ is for ‘Temporal’](https://www.taoeffect.com/other/nupic/#t-is-for-temporal)

Numenta’s white paper states that, “Time plays a crucial role in learning, inference, and prediction.” We’ll see the truth in this [when we compare](https://www.taoeffect.com/other/nupic/#ANNs) the HTM with purely feed-forward ANNs.

HTM networks are trained with, and form predictions about, time-varying input-data. We should have good reason to believe that the data contains some type of temporal pattern, otherwise it is simply noise (like static on the radio). Too many unrelated temporal patterns in the data-stream can also lead to poor results (students with ADHD and poor study habits know this very well).

[‘M’ is for ‘Memory’](https://www.taoeffect.com/other/nupic/#m-is-for-memory)

We’ve been dancing around the technical details for a while now, so let’s have a closer look at how HTMs create memories and form predictions.

Recall that each HTM region contains a three dimension grid of cells, and each cell can be either on or off (for various reasons). The cells are grouped in vertical columns, and all the cells in a column get the same feed-forward input (either directly from the data, or the output from a lower HTM region). The input, as mentioned previously, is converted to a sparse distributed representation, and then each column gets a unique subset of that input (usually overlapping with other columns, but never the exact same subset). Cells also receive lateral input from other columns of cells.

Which cells get activated within a column depends on a variety of factors, and the use of columns allows the HTM to create temporal contexts. The white paper explains:

By selecting different active cells in each active column, we can represent the exact same input differently in different contexts. A specific example might help. Say every column has 4 cells and the representation of every input consists of 100 active columns. If only one cell per column is active at a time, we have 4^100 ways of representing the exact same input. The same input will always result in the same 100 columns being active, but in different contexts different cells in those columns will be active.

Which cells in a column become active depends on the current activation state of those cells:

When a column becomes active, it looks at all the cells in the column. If one or more cells in the column are already in the predictive state, only those cells become active. If no cells in the column are in the predictive state, then all the cells become active. You can think of it this way, if an input pattern is expected then the system confirms that expectation by activating only the cells in the predictive state. If the input pattern is unexpected then the system activates all cells in the column as if to say “the input occurred unexpectedly so all possible interpretations are valid”.

Real neural networks contain inhibitory signals as well as excitatory ones. The same concept is applied in HTM regions as part of the lateral input that cells receive from their neighboring columns:

The columns with the strongest activation inhibit, or deactivate, the columns with weaker activation. (The inhibition occurs within a radius that can span from very local to the entire region.) The sparse representation of the input is encoded by which columns are active and which are inactive after inhibition. The inhibition function is defined to achieve a relatively constant percentage of columns to be active, even when the number of input bits that are active varies significantly.

Lateral input can inhibit as well as excite cells, and this is how HTM regions form predictions:

When input patterns change over time, different sets of columns and cells become active in sequence. When a cell becomes active, it forms connections to a subset of the cells nearby that were active immediately prior. These connections can be formed quickly or slowly depending on the learning rate required by the application. Later, all a cell needs to do is to look at these connections for coincident activity. If the connections become active, the cell can expect that it might become active shortly and enters a predictive state. Thus the feed-forward activation of a set of cells will lead to the predictive activation of other sets of cells that typically follow. Think of this as the moment when you recognize a song and start predicting the next notes.

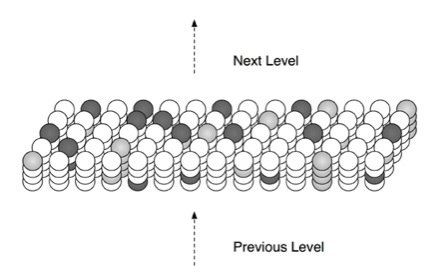


Figure 1.3: At any point in time, some cells in an HTM region will be active due to feed-forward input (shown in light gray). Other cells that receive lateral input from active cells will be in a predictive state (shown in dark gray).

The influence one cell has on another is moderated by the strength of the a virtual synapse between, which can have a decimal value between 0 and 1. For additional details, please see the white paper.

[Comparing Intelligences](https://www.taoeffect.com/other/nupic/#comparing-intelligences)

[HTM vs. Primitive feed-forward ANNs](https://www.taoeffect.com/other/nupic/#htm-vs.primitive-feed-forward-anns)

As mentioned to previously, HTM regions are used to map noisy data to more stable concepts. Another way to state that is: regions extract *temporal patterns* out of their input.

In this section I will use the phrase “primitive ANN” to refer to feed-forward networks that do not have contain cycles between cells.

Primitive ANNs *do not* take into account time, and this severely limits the types of patterns they are capable of recognizing. They do terrible job of anticipating future events because they don’t support a contextual temporal model.

*What does that \_\_\_\_?*

Precisely! I see that you understand perfectly. ☺

Those fluent in modern English can probably fill in the blank without much trouble, or at least are capable of narrowing down the possibilities to a handful of words.

To illustrate the difference between temporal patterns and those handled by primitive ANNs, consider how you might have reacted had I instead written:

*That \_\_\_\_?*

Now it is much more difficult to predict what belongs in the blank. This is due to the lack of prior context. Human senses such as touch, vision, and hearing, absorb and react to symbols (input) in a sequential manner. The sequence these symbols present themselves in usually results in predictable patterns. We can predict the word that belongs in the first blank simply because our brain has heard it stated so many times after the words “What”, “does”, and “that”.[9](https://www.taoeffect.com/other/nupic/#fn9)

Primitive ANNs do not use any sort of temporal context to make predictions. Instead, they treat input as chunks of raw data that is then mapped directly to an output without consideration for the order in which the data arrived. The output depends entirely on the ANN’s training data and error-correction algorithms.

Therefore, a primitive ANN processes “What does that” in a single gulp, treating it no differently than it would a single word.

Another consideration is efficiency. Numenta’s white paper points out that hierarchies result in less training time for the network “because patterns learned at each level of the hierarchy are reused when combined in novel ways at higher levels.” While I’m not 100% certain (as I did not have [the time](https://www.taoeffect.com/other/nupic/#time) to test and compare implementations), I suspect that hierarchy accounts for only part of the improvement in efficiency. I think a hierarchy of HTMs would outperform an “equivalent” hierarchy of primitive ANNs because of the memory-efficiencies that temporal contexts provide.

[HTM vs. Long short term memory](https://www.taoeffect.com/other/nupic/#htm-vs.long-short-term-memory)

[Long short term memory](https://en.wikipedia.org/wiki/Long_short_term_memory) (LSTM) refers to a type of [recurrent neural network](https://en.wikipedia.org/wiki/Recurrent_neural_network) (RNN) first described in 1997 by Sepp Hochreiter and Jürgen Schmidhuber.[10](https://www.taoeffect.com/other/nupic/#fn10)

RNNs have the capability of processing and interpreting temporal sequences unlike the feed-forward networks [we just discussed](https://www.taoeffect.com/other/nupic/#ANNs). They accomplish this through loop-back connections among the cells within the network, giving their memory the capacity to process arbitrary sequences of data.

Of the various types of RNNs out there, the LSTM produces some of most the impressive results. Most other types of RNNs are not able to handle long pauses between events in the input data (for example, a period of silence when someone is talking). The LSTM does not have this limitation. Felix Gers thesis explains: [11](https://www.taoeffect.com/other/nupic/#fn11)

The [LSTM] algorithm overcomes this and related problems by enforcing *constant* error flow. Using gradient descent, LSTM explicitly learns when to store information and when to access it.

In *Spatio–Temporal Memories for Machine Learning: A Long-Term Memory Organization*, researchers Starzyk and He took inspiration from both the LSTM and HTM to create their own hierarchical neural net based on interconnected long-term memory (LTM) cells: [12](https://www.taoeffect.com/other/nupic/#fn12)

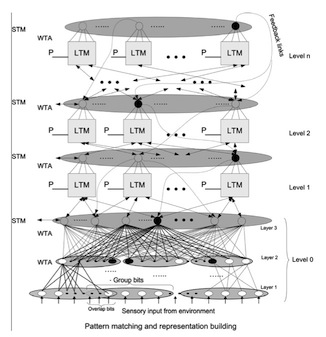


Figure 2.1: Overall LTM architecture.

I think a comparison of these two approaches deserves more attention and is something that I’d like to research further. I am also interested in reading additional material on this topic, so please, [don’t be a stranger](https://twitter.com/taoeffect) and feel free to send relevant material my way, I’d greatly appreciate it!

[Academia vs. Numenta](https://www.taoeffect.com/other/nupic/#academia-vs.numenta)

Surprisingly, the first major difference to jump out at me between the HTM and LSTM had nothing to do with algorithms or their respective topologies. Instead, it was the potent stench of dry, lifeless academic language wafting out of all of the documentation I could find on the LSTM that left a most profound impression on me. Indeed, the time I spent reading Ph.D. theses and papers from IEEE journals represented one of the least enjoyable parts of my research purely on account of the terrible prose. Regardless of what some might believe, Universities do not bundle academic degrees with a license to carry a rotting right-hemisphere. The guilty tend to cling to [argumentum ad populum](https://en.wikipedia.org/wiki/Argumentum_ad_populum) as their sole defense. In a just universe, they’d still be working through secondary school.

On the other hand, I have to commend academics for being team players, a concept that appears somewhat foreign to Numenta. Whereas all of the academic papers I read contained countless[13](https://www.taoeffect.com/other/nupic/#fn13) references and citations to other work in the field, the same could not be said of any of Numenta’s. None of the three Numenta papers on the HTM [14](https://www.taoeffect.com/other/nupic/#fn14) [15](https://www.taoeffect.com/other/nupic/#fn15) [16](https://www.taoeffect.com/other/nupic/#fn16) contain anything resembling a references section acknowledging the authors of prior work upon which their ideas are based.

The academics, for their part, cited Hawkins’ work where appropriate.[17](https://www.taoeffect.com/other/nupic/#fn17)

[Conclusion](https://www.taoeffect.com/other/nupic/#conclusion)

From my research I’ve concluded that, at least for me, the HTM and its corresponding learning algorithm(s) warrant further research, as they appear to be very useful and powerful tools for prediction, modeling real-world data, and creating artificial intelligences.

Also, software should not be patentable.[18](https://www.taoeffect.com/other/nupic/#fn18) [19](https://www.taoeffect.com/other/nupic/#fn19) [20](https://www.taoeffect.com/other/nupic/#fn20) [21](https://www.taoeffect.com/other/nupic/#fn21) [22](https://www.taoeffect.com/other/nupic/#fn22) The practice should be abolished immediately for the good of mankind and whomsoever disagrees with me on this owes me $1 billion in royalties because I patented the concept of being an asshole.

Comments on [reddit](https://pay.reddit.com/r/MachineLearning/comments/1mds16/hierarchical_temporal_memory_nupic_and_numentas/) or [my blog](https://www.taoeffect.com/blog/2013/09/the-apache-contributors-license-agreement-is-very-dangerous/).

[About the Author](https://www.taoeffect.com/other/nupic/#about-the-author)

[Greg Slepak](https://twitter.com/taoeffect) refers to a sometimes-conscious entity whose primary public activities some recognize as ‘entrepreneurship’, ‘software engineering’, and ‘writing’. He taught himself C at age 12, and continued to learn other programming languages, with present favorites Clojure and newLISP. During his freshman year as an undergraduate at UF, he joined [CIMAR](http://cimar.mae.ufl.edu/CIMAR/) and wrote the [High Level Planner](https://www.youtube.com/watch?v=hI2gRYXfpfg) for UF’s entry to the 2007 DARPA Urban Challenge. In 2008 he founded [Tao Effect LLC](http://www.taoeffect.com/) as part of an effort to make [data encryption](http://www.espionageapp.com/) more user-friendly.

[Document History](https://www.taoeffect.com/other/nupic/#document-history)

This is a living document and future updates are a significant possibility. Notifications of updates will be posted to my [twitter account](https://twitter.com/taoeffect), and possibly to [my blog](https://www.taoeffect.com/blog/) as well.

| **Version** | **Date** | **Comment** |
| --- | --- | --- |
| 2.0.0 | September 14, 2013 | Changed title of paper. Replaced “Numenta’s Unethical Behavior” with “Numenta’s Commendable Behavior” because Numenta updated the agreement. Updated many related sections. |
| 1.0.1 | August 28, 2013 | Minor edits. |
| 1.0.0 | August 26, 2013 | Document first published online. |

[Acknowledgments & References](https://www.taoeffect.com/other/nupic/#acknowledgments-references)

I’m very much aware of the horrific state of this section, and hope to clean it up a bit in a future update. If you’d like to help me de-wikifiy the citations you are most welcome to [contact me](https://twitter.com/taoeffect) with links to better sources.

I would like to thank Andrea Devers for helping me edit and proofread this document, and Bob Jesse for his feedback.

1. <https://en.wikipedia.org/w/index.php?title=Hierarchical_temporal_memory&oldid=567124697>[↩](https://www.taoeffect.com/other/nupic/#fnref1)
2. <https://en.wikipedia.org/w/index.php?title=Hierarchical_temporal_memory&oldid=567124697#Similarity_to_other_models>[↩](https://www.taoeffect.com/other/nupic/#fnref2)
3. [*To a Mouse* by Robert Burns](http://www.quotationspage.com/quote/32287.html)[↩](https://www.taoeffect.com/other/nupic/#fnref3)
4. <http://www.kinostudios.com/mandelbulb.html>[↩](https://www.taoeffect.com/other/nupic/#fnref4)
5. <https://en.wikipedia.org/w/index.php?title=Linus_Torvalds&oldid=569095816>[↩](https://www.taoeffect.com/other/nupic/#fnref5)
6. <http://docs.python.org/release/2.3.4/inst/search-path.html>[↩](https://www.taoeffect.com/other/nupic/#fnref6)
7. <http://docs.python.org/2/install/>[↩](https://www.taoeffect.com/other/nupic/#fnref7)
8. <http://docs.python.org/2/install/>[↩](https://www.taoeffect.com/other/nupic/#fnref8)
9. Requiring commas and periods within quotes (when they were not originally there) is an archaic and illogical practice that appears to have been invented as a hack to get around [ancient printing problems](http://grammar.ccc.commnet.edu/grammar/marks/quotation.htm#footnote). Its use in modern English is therefore an idiotic anachronism and should be abolished. If you see anyone doing it please be sure to forward [this link](http://grammar.ccc.commnet.edu/grammar/marks/quotation.htm#footnote) to them. Raising hell about the matter is entirely up to you. I personally encourage it.[↩](https://www.taoeffect.com/other/nupic/#fnref9)
10. S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural Computation, 9(8):1735–1780, 1997.[↩](https://www.taoeffect.com/other/nupic/#fnref10)
11. [Long Short-Term Memory in Recurrent Neural Networks](http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.6677)[↩](https://www.taoeffect.com/other/nupic/#fnref11)
12. Starzyk, He [Spatio-temporal memories for machine learning: a long-term memory organization.](http://www.ncbi.nlm.nih.gov/pubmed/19336289). See page 769 in “IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 20, NO. 5, MAY 2009”.[↩](https://www.taoeffect.com/other/nupic/#fnref12)
13. Actually, there were counted.[↩](https://www.taoeffect.com/other/nupic/#fnref13)
14. [HTM\_CorticalLearningAlgorithms.pdf](http://cl.ly/3G092O1H150C)[↩](https://www.taoeffect.com/other/nupic/#fnref14)
15. [Numenta HTM Concepts.pdf](http://cl.ly/1z1W2X0V0y0c)[↩](https://www.taoeffect.com/other/nupic/#fnref15)
16. [Numenta HTM Learning\_Algos.pdf](http://cl.ly/2y1G322H2E3G)[↩](https://www.taoeffect.com/other/nupic/#fnref16)
17. Starzyk, He [Spatio-temporal memories for machine learning: a long-term memory organization.](http://www.ncbi.nlm.nih.gov/pubmed/19336289). See page 769 in “IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 20, NO. 5, MAY 2009”.[↩](https://www.taoeffect.com/other/nupic/#fnref17)
18. <http://www.forbes.com/sites/reuvencohen/2013/05/08/new-zealand-government-announces-that-software-will-no-longer-be-patentable/>[↩](https://www.taoeffect.com/other/nupic/#fnref18)
19. <http://www.burgess.co.nz/law/why-software-should-not-be-patentable-the-academic-approach/>[↩](https://www.taoeffect.com/other/nupic/#fnref19)
20. <http://endsoftpatents.org/pages/resources-for-lawyers/>[↩](https://www.taoeffect.com/other/nupic/#fnref20)
21. <http://www.tonymarston.net/php-mysql/software-patents-are-evil.html>[↩](https://www.taoeffect.com/other/nupic/#fnref21)
22. <http://www.javaworld.com/javaworld/jw-03-2012/120316-open-sources.html>[↩](https://www.taoeffect.com/other/nupic/#fnref22)