

# Background on HTM algorithm

August 2, 2022

## Informal History of HTM

HTM, or Hierarchical Temporal Memory is an artificial intelligence architecture rooted in neuroscience and justified by biological plausibility. It is the outgrowth of funding by Jeff Hawkins, the former CEO of Palm Pilot and techniques developed by research scientists Dileep George and Subatai Ahmad. In 2004, Jeff Hawkins published a book titled, "On Intelligence: How a New Understanding of the Brain Will Lead to the Creation of Truly Intelligent Machines". He founded the Redwood Institute for Neuroscience in 2005. The institute hired Dileep George, a computational scientist, and one year later in 2006, George gave a presentation titled "Hierarchical Temporal Memory: Theory and Applications" [6].

To give some artificial intelligence context to this timeline, Yan LeCun published successful work with LeNet on MNIST in 1998, but the breakthrough in deep learning for image classification did not occur until 2012 with Geoffrey Hinton's AlexNet which pioneered the use of a GPU. According to this *Quora answer*, before Hinton's success, AI researchers were questioning neural networks and were looking to Hawkins' book as a hopeful path forward. Hierarchical temporal memory promised to be a theory of brain computation that could be adapted to artificial intelligence. The argument of Hawkins' book is clear from the title: *Since the brain is an our example of a general intelligence, AI should seek to understand and mimic the brain.* A fairly sound argument in theory, but it runs into trouble with the fact that nobody understands how the brain works. (This can make interdisciplinary AI + neuroscience work an easy target for detractors, making it vulnerable both to critiques of its biological plausibility and its lack of performance compared to other purely computational AI approaches.) The above Quora answer claims that the lack of success by HTM may have been related to the way it was based on neuroscience data that was not well understood.

There is little doubt that the neuroscience data was not well understood. In 2018, Richard Axel, an Nobel Prize winning neuroscientist said, "we do not have a logic for the transformation of neural activity into thought and action. I view discerning [this] logic as the most important future direction of neuroscience" [1]. Given that this quote came 10 years after the advent of HTM suggests that

the neuroscience theory behind HTM was definitely lacking. However, we can't dismiss HTM on the basis of an incorrect neuroscience model precisely because no one understands how higher order cognitive functions emerge. We can only find better models by building and testing models and seeing where they break. That is the process of science. Did HTM make progress in understanding how the brain might work? I'm not sure, but it seems to be a locus of key ideas from neuroscience and therefore seems likely that they combined many theoretical concepts in novel ways.

When it comes to studying HTM now, there is a value question. If HTM is no longer making progress at understanding brain computation, *and* it is unsuccessful at AI problems, then we might decide to look for different tools. But does it have something to offer still? It remains to be seen.

In response to this *Quora question*, a researcher points out that HTM was slow originally, and another answer suggests that it is not so slow anymore.

Dileep George left Numenta in 2010. This *YCombinator thread* suggests that George left due to a difference in technical direction. . This *article* quotes George saying "HTM was an important effort, much like Poggio's foundational HMAX model, HTM had the right high level goals. But, when you dig into the algorithm level, you'll see that HTM implementations hadn't solved the problems of information representation in the hierarchy. This led to inefficient learning and scaling issues. The mathematical formulation of HTMs were made up of these "blocky" nodes with boundaries that restricted information transfer between adjacent nodes, which is not very effective in dealing with domains like vision and sound."

George went on to found Vicarious, which explored many other brain inspired algorithms but seems to have more of an application focus than Numenta. Vicarious focused on on robotics and became one of the leaders the field. Vicarious was recently acquired by Intrinsic, a Google X spinout, and Dileep George began working at Deep Mind.

To this day there still appears to be some overlap between the George's research and Numenta's work. Both have produced computational models of mapping in the hippocampus [7, 8].

I'm not sure what happened when George left Numenta. Subutai Ahmad was the VP of engineering from 2005-2014, and then the VP of research from 2014-Present. So it appears that Subutai was hired from the beginning and became the lead technical person at Numenta some time after George left.

I mention Subutai in particular because I read his 2017 spacial pooler paper very thoroughly and was impressed [4]. The spatial pooling algorithm was validated with the use of several metrics that studied the internal components and verified that each piece was behaving in an optimal manner. The algorithm included biologically plausible mechanisms for making sure that all neurons were used and that all neurons fired at equal rates. This came in contrast to most ML algorithms that I have studied where the actual function of the internal components is hypothesized by experts and becomes folklore, which sometimes becomes dogma. Perhaps this is why papers like the lottery ticket hypothesis come out, because there is a lack of understanding the truth of how the algorithm

works at the earlier stages. The spatial pooling paper on the other hand offered clear proof that the algorithm actually worked the way they thought it did and evidence for why this was optimal.

## Reproducing Results from the Numenta Spacial Pooling Paper

I started with the Julia package `HierarchicalTemporalMemory.jl`, and was able to instantiate a spacial pooler, and pass MNIST images as a bitmap, but after a single epoch, the visual fields of the mini-columns did not appear to change. I tried a number of different parameter settings but I wasn't able to get it to work. This is probably because I don't understand the algorithm.

Numenta has a lot of source code online, but unfortunately they don't maintain it. It is not compatible with Python3. It was very difficult to get it to run on my computer. Here are the steps.

- Install virtualenv *pip install virtualenv*
- Make a Python2 virtual environment. *virtualenv -p="path/to/python2/binary" nupic\_venv*
- Activate your virtual environment. *source nupic\_venv/bin/activate*
- Clone nupic and nupic.vision from github. To keep all of the packages in one place I made a directory inside of nupic\_venv called byhand for all the packages I would download by hand.
- Run *pip install -e .* inside of each one. It will probably fail. It failed for me because the dependancy files couldn't be downloaded from PyPI.
- Go to PyPI and download the .whl or .tar.gz for the version that is needed.
- Run *pip install path/to/package.whl/or/package.tar.gz* for each one.
- For `nupic.vision/src/nupic/vision/mnist/convertImages.py` I had a problem with pillow. I fixed it by running *pip uninstall pillow; pip install pillow -U* which installs a later version of pillow, and luckily this still works with the code.

The nupic code base is very well written. It's clear that it was made by software engineers rather than by research scientists. And it is extensive. This makes me think that they were aiming to use it in production. It makes me surprised that they dumped it and stopped maintaining it. Did they decide against HTM? I don't think that they have published any HTM papers recently. *This thread* confirms that Numenta has moved on from HTM. This is a red flag for me. The company that built the algorithm dumping it. Feels like they are the people who know it best so they should be the ones who know how to use it.

Instead they appear to have moved on to sparse deep neural nets. Which are neural nets that work with 5% of the parameters. Significant and useful work, but 5% of 175 billion (parameters in GPT-3) is 8.75 billion. And this sort of feels like selling out. Maybe they are using neuroscience to do this but I watched one of their research meetings where they read through a Gregory Hinton paper on sparsity and said that their work was similar.

## How does HTM compare to Deep Learning?

### Time Series Prediction

I only found a few papers that directly compare HTM to deep learning algorithms [11, 10, 5, 3]. Of these, [11] is the most comprehensive. It suggests that it is the first paper to make such a comparison, and it shows definitively that with proper hyper parameter optimization, LSTM outperforms HTM, both in terms of accuracy and speed. It goes further than this, by using the same datasets as the original Numenta papers and showing that with proper hyper parameters, LSTM actually outperforms HTM [5, 3]. The last section of the paper describes training the models from the previous dataset on a non stationary dataset with a global trend. HTM was able to adapt to this dataset without any hyperparameter adjustment. With no hyper parameter optimization, LSTM failed miserably on this dataset. Unfortunately the authors did not re-optimize parameters and so we don't know how the algorithms would compare.

### Anomaly Detection

Struye et. al. claims that HTM has proved itself at anomaly detection. Indeed Numenta has published a *repository* for benchmarking real time streaming anomaly detection. I think the adaptive nature of HTM makes it well suited for this sort of problem. At the same time, I wonder if Numenta was highlighting the best pieces of their algorithm and it hasn't been fully examined by others. Looking into this further I found a paper titled "Current Time Series Anomaly Detection Benchmarks are Flawed and are Creating the Illusion of Progress" [12]. The paper examines many anomaly detection time-series datasets including the Numenta benchmark and finds that they contain mislabeled data, many trivial problems, too many anomalies and a run to failure bias (most anomalies occur at the end). Basically, the benchmark isn't great. In the paper, you can see one of the Numenta benchmark time series, a spiking pattern, in the dataset and it looks pretty silly. The paper goes into depth on the use of anomalies in the Numenta taxi driver dataset and shows the existence of multiple false positives and false negatives in the labels.

Furthermore, within the Numenta anomaly benchmark, HTM was beaten by another brain inspired algorithm called Adaptive Resonance. Adaptive resonance was suggested to me by some of the contributors to the HTM community.

## Adaptive Resonance Theory

The main contributors to this field are wife and husband Gail Carpenter and Stephen Grossberg. I read an introductory paper by Grossberg and it made some sweeping claims. I searched around for a second opinion and found *this Quora answer*. The Quora answer gave the perspective of a neuroscientist who also highlighted the huge claims of Adaptive Resonance Theory but pointed out that it doesn't guide much of the thinking in neuroscience. They also pointed out that one of the ART papers has 13 self citations out of 17 total citations.

I found a recent review titled "A survey of adaptive resonance theory neural network models for engineering applications" [2]. It is very comprehensive. ART has many different kinds of algorithms that are adapted to different applications.

But I am concerned for two reasons. First, so far I can find no comparisons of ART algorithms to deep learning methods. Second, ART has been around for over 30 years and as far as I can tell, no cognitive scientist has picked it up. This seems like a red flag to me and lines up with the Quora answers' point that about the number of self citations. Did ART develop in isolation from the scientific community? And that is concerning. The way ART can make such broad claims without fearing reproach is odd. I need evidence from outsiders that ART is a good research direction.

Compare this to Numenta, who pitted their work head to head with deep learning. It makes HTM much more respectable. They had to make something good because they were comparing to others.

## Community of Brain Computation

That is the confusing thing. Wolfgang Maass claims that, "there is at present no community of computer theorists studying the brain" [9]. There is no central community that I know of, but there are several small groups. There is the Olhausen group, the Carpenter group, the Papadimitriou group, the Maass group, the Numenta group, the Schoner group, and maybe the Tenenbaum group counts but they are more focused on Bayesian inference.

## Thoughts Going Forward

What brain computation needs is theoretical advances. Advances that take breakthrough neuroscience experiments and put them into math at the right resolution. The right resolution is one that sheds light on the way neurons collectively process information.

I think that HTM is just building an architecture around the sparse representation. And who is to say that it wouldn't work without more development. Neural networks took years of development before they worked. They had tons of problems. Maybe we can advance HTM some as well.

*This comment* in the HTM forum covers some problems with the HTM spacial pooler. It was written in July 2017 which places it at about the same

time at the spacial pooler paper was published.

I talked to Eric Weisman a former PhD student at the Redwood institute and he was extremely helpful. He told me that Jeff Hawkins started Numenta because the academics he hired were going too slow. And it is because they were trying to solve the theory of brain computation problem, and he wanted an architecture, something he could sell.

My current perspective is that HTM is a architecture built around the theory of a sparse distributed representation, the most recent advance in the theory of brain computation. It is more competitive than neural networks when they were invented but it is less competitive than deep neural nets.

So what happened with neural nets? Someone invented a faster way to train them and then someone made them really big and put them on a GPU.

## References

- [1] Richard axel. *Neuron*, 99(6):1110–1112, 2018.
- [2] Leonardo Enzo Brito da Silva, Islam Elnabarawy, and Donald C. Wunsch. A survey of adaptive resonance theory neural network models for engineering applications. *Neural Networks*, 120:167–203, 2019. special Issue in Honor of the 80th Birthday of Stephen Grossberg.
- [3] Yuwei Cui, Subutai Ahmad, and Jeff Hawkins. Continuous online sequence learning with an unsupervised neural network model. *Neural Computation*, 28(11):2474–2504, nov 2016.
- [4] Yuwei Cui, Subutai Ahmad, and Jeff Hawkins. The htm spatial pooler—a neocortical algorithm for online sparse distributed coding. *Frontiers in Computational Neuroscience*, 11, 2017.
- [5] Yuwei Cui, Chetan Surpur, Subutai Ahmad, and Jeff Hawkins. A comparative study of htm and other neural network models for online sequence learning with streaming data. In *2016 International Joint Conference on Neural Networks*, pages 1530–1538, 07 2016.
- [6] Dileep George. Hierarchical temporal memory: Theory and applications. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*, volume Supplement, pages 6643–6643, 2006.
- [7] Dileep George, Rajeev V. Rikhye, Nishad Gothoskar, J. Swaroop Guntupalli, Antoine Dedieu, and Miguel Lázaro-Gredilla. Clone-structured graph representations enable flexible learning and vicarious evaluation of cognitive maps. *Nature Communications*, 12(1):2392, 2021.
- [8] Marcus Lewis. Hippocampal spatial mapping as fast graph learning, 2021.
- [9] Wolfgang Maass, Christos H. Papadimitriou, Santosh Vempala, and Robert Legenstein. *Brain Computation: A Computer Science Perspective*, pages 184–199. Springer International Publishing, Cham, 2019.

- [10] Jonathan Mackenzie, John F. Roddick, and Rocco Zito. An evaluation of htm and lstm for short-term arterial traffic flow prediction. *IEEE Transactions on Intelligent Transportation Systems*, 20(5):1847–1857, 2019.
- [11] Jakob Struye and Steven Latré. Hierarchical temporal memory and recurrent neural networks for time series prediction: An empirical validation and reduction to multilayer perceptrons. *Neurocomputing*, 396:291–301, 2020.
- [12] Renjie Wu and Eamonn Keogh. Current time series anomaly detection benchmarks are flawed and are creating the illusion of progress. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2021.