ANOMALY DETECTION WITH CORTICAL LEARNING ALGORITHM FOR SMART HOMES

Marek Otahal[1], prof. Olga Stepankova[2]

[1]otahama2@fel.cvut.cz, NIT at CTU FEE, Prague, CZ

[2]step@labe.felk.cvut.cz, head of NIT group at CTU FEE, Prague, CZ

Abstract

After a brief introduction to principles of Cortical Learning Algorithm (CLA) and Hierarchical Temporal Memory (HTM) which frame the theory for NuPIC – a human neocortex-inspired neural network implementation, we focus on practical use-cases in context of Smart Homes: 1) processing bio-signals (ECG classification, prediction, anomaly detection) for the purpose of mobile assisted technologies. 2) prediction of sensory and location data in smart homes. Preliminary results, realized and in-progress projects are presented in this study as well as obstacles and some comparison to other solution for these ML tasks.

Keywords

anomaly detection in ECG/position, cortical learning, bio-sensors, monitoring, HTM/CLA, NuPIC

Brief Theory of Cortical Learning and Hierarchical Temporal Memory

Hierarchical temporal memory (HTM)

HTM is a machine learning technique inspired closely by the structural and algorithmic properties of the mammalian neocortex. The observation is that most of the high level cognitive tasks – vision, movements, planning or language are performed by the neocortex and on the contrary to their wide functionality, all the neocortex exhibits a remarkably uniform neural structure [1] - organization into columns & layers, as shown in Fig 1.

HTM is a memory based learning system which performs unsupervised online-learning and continuous predictions where the role of time is implicit (expressed by sequential order).

The structure of HTM memory system models: neurons (called cells in HTM/CLA context), synapses & dendrites, which are organized into columns, layers and regions at a higher level.

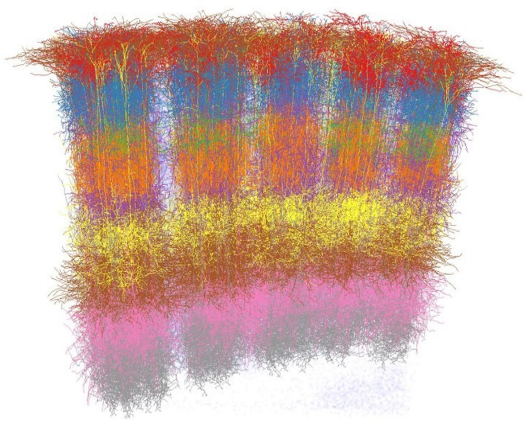

Fig. 1: Cell-type-specific 3D reconstruction of five neighboring barrel columns in rat vibrissal cortex (credit: Marcel Oberlaender et al.)

The following list is a short summary of core HTM memory features:

- *Hierarchy*

Is the organizational arrangement of regions (higher level units, "small networks"), performs abstraction (similar to hidden layers in deeplearning) as well as combination of different senses (like small neural networks [2])

- *Sparse Distributed Representations (SDRs)*

A neural representation (embedding) is the state of (output) layer after being presented with (a sequence of) patterns.

"*Sparse*" means each pattern is represented only by a very small percentage of active columns (~2% typically [3]).

"*Distributed*" is that a single column is used to participate in different patterns, and that a distribution of a pattern is locally distributed (not a single cluster).

"*Decomposable*" SDRs are (context aware) semantic embeddings. For example, Fig. 2 shows semantic similarity of SDRs comparing "apples and oranges".

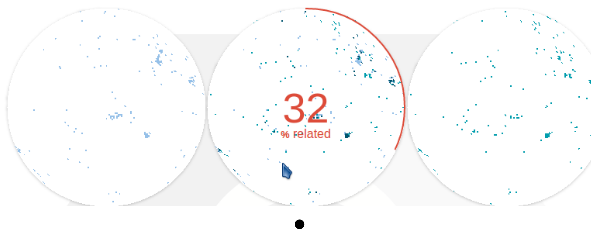

Fig. 2: A representation and semantic similarity of words "Apples" and "Oranges" produced by cortical.io in a form of SDR that can be used in HTM/CLA.

- *Role of time*

There are two ways how the notion of time is implemented in HTM, the *implicit time* which assumes patterns from our sensory system enter the memory at the same intervals (eg. from eyes), where the distance (in time) between two events is expressed by the number of samples separating them.

In some use-cases it is common to model time as a new (artificial) variable field, rather than expressing it by number of samples. Experiment to compare these two approaches is being worked on.

Time itself is crutial to the theory and how we perceive through our senses – most (with vision being a partial exception) of our senses require creating a conceptual model based on time-series of sensory inputs (like word "hello" is formed from phenomes h-e-ll-ou).

Cortical learning algorithms

Cortical learning algorithm (CLA) is the learning method in HTMs, similar to Hebbian learning rule – applied at several levels: *synapses* – tend to map the input space in the most effective way; *cells* – learn to represent (different) context for the same concept; and *columns* – groupings of cells with same feed-forward input, with inhibition – columns form the output SDR.

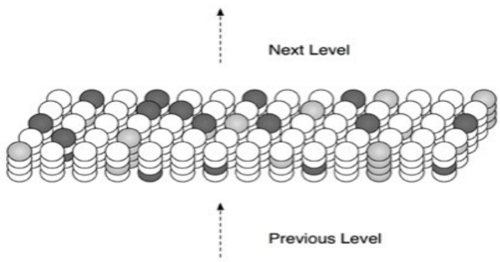

Fig. 3: At any point in time, some cells in an HTM region will be active due to feed-forward input (shown in light gray). Other cells that receive lateral input from active cells will be in a predictive state (shown in dark gray). (credit: CLA Whitepaper, [5])

A CLA region finds common patterns in the input, then puts them in context with the historical observations and finally makes predictions about the next step.

For a nice and easy explanation of the principles behind HTM/CLA see "On Intelligence" [4], or "the CLA Whitepaper" for a more detailed description [5].

Processing bio-signals with HTM/CLA

Thanks to its biological motivation when processing bio signals HTM seems to be a good choice as it is responsible for processing them in our brains, which we are trying to mimic with this theory.

HTM has been experimentally applied to domains of vision (image recognition [6], object detection in videos [7]), hearing (song composer) [8], natural language processing [9], planning or motion control [10].

Task 1: Classification of ECG

A task in collaboration with J.Vitku (CTU) and E.Volna (UO, Ostrava), our goal was to classify given ECG samples [11] (1 electrode, each sample 100 measurements, 70 patients – 50 training, 20 evaluation) into "healthy" and "sick" categories and highlight the discriminating subsequences in the signal.

Methodology

For CLA, no preprocessing or complex parameter optimization was required, I've designed a simple CLA region that was set to perform 1-step ahead prediction.

Because CLA is online-learning algorithm, and we lacked “big data“, 2 separate models were trained, each on different class of training data only. And classification was done by running the test sequence through both models and choosing the best estimator. (1)

Another approaches which I'm about to try are creating a region performing anomaly detection, training it only on “healthy” training data, then disabling learning and exposing it to the training “unhealthy” dataset to estimate a threshold on overall anomaly score over the whole sequence. In this case, classification means computing anomaly score for the test sample and assigning class according to the threshold.

Third option would be to run a sort of a supervised online learning, where an artificial label is assigned to each measurement from the training samples ({ECG value at time T, class of this sample})

Results

In our experiments, CLA model described in (1) slightly outperformed the multi-layer NN. The biggest advantage we attribute to the fact that no preprocessing and fewer training time were needed.
As the other subtask for CLA, we used it to automatically select discriminative subsections of the sequence (a technique used to speed-up the NN method and where the interesting data were pre-selected by a human supervisor, the CLA could replace this supervision).

Task 2: Anomaly detection and predictions

Given real-life ECG signal measurements (16 electrodes, 30Hz sampling, approx. 1 hour of data) our aim is to detect (and potentially predict) anomalies.

Methodology for anomaly detection is creating a simple “anomaly detection” model and performing parameter optimization (called “swarming” in NuPIC). We are trying models with separate electrodes and with combining all electrodes as multiple inputs. As for online-learning, the training/evaluation process is done in one go through the date, where a selected part at the beginning is considered burn-in (training time).

Anomaly prediction will be more complicated, as from the essence “anomalies” are impossible to predict (at least not within the same model). What we hope can be done is learning “emerging symptoms” - a set of features that are still “normal”, but are known to cause an anomaly N steps later. We would like to achieve this for variable N which suggests CLA is a good choice with its variable length contextual memory.

The idea is to use 2 models with different dynamics (one more stable, serving as an etalon) and use them to predict the same values from which we can compute anomalies “in future”.

During this task, I have implemented new methods for computing anomaly [12] which we are testing currently. Selecting correct parameters proved to be challenging – to achieve balance on the details the model captures vs. certain simplicity of the results. Another observed issue was computation speed per sample, which kept slightly increasing up to approx. 10 samples/s on my machine. This fact is known, can be mitigated by some parameter options (experiments and optimization metric that takes time into account underway) as well as with speed optimizations which are being worked on (reimplementation in C++) we are also considering rewrite to GPGPU (in python).

Sensory data from Smart Homes

In the following section, we would like to highlight some advantages to using CLA for processing sensory data from smart homes, and offer examples of several existing projects or new ideas that could be implemented.

Motivation for Smart Homes and Internet of Things

We believe the concept of smart homes is coming from future to the reality – with the miniaturization of various electronic gadgets, wearable electronic, omnipresent wifi signal and connections to social media – there already are, and the number will be growing, devices that are more or less stand alone helpers in our lives, and these generate lots of (meta)data. Privacy considerations aside, these data can be very useful for modeling our everyday lives or functions of your environment. And in majority of the cases the produced data is streaming data from (multiple) sensors where we are interested in predictions and/or detecting anomalies.

Ideas & Applications

From our experiments, biological signals in combination with wearable electronics could be a very interesting area for predictions/anomaly detection. We have experience with processing ECG signal, next step would be adding other sources (eg. oxidation levels, body temperature, EEG signal). My current work in progress is a novel “electronic baby-sitter” – a pulse anomaly detector in combination with an open-source webcam pulse detection library – exampe how it works show in Fig. 4.

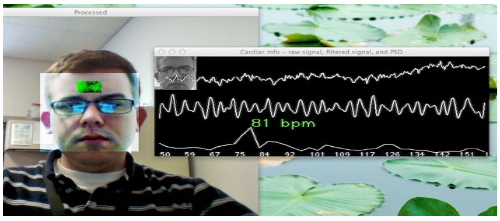

Fig. 4: Pulse detection from nearly invisible changes to skin color with camera. (credit: Webcam-pulse-detector [13])

Typical application would be applying optimization based on predictions or alarm set by anomaly detector to various appliances on our homes – eg. Heat predictions in the building, or server load anomaly detections [14].

A relatively new option is encoding and processing positional data (GPS location), there is even a rewarded competition underway for interesting ideas and applications with this data. A simple but useful application would be a location sensor placed on your dustbin "that has to be moved from shelter to the street every Tuesday to be emptied." An interesting project, with an advantage of having data and alternative implementations for benchmarking already in place, is traffic prediction in cities obtained from time-GPS data from cars.

Conclusion

Hierarchical cortical memory, closely derived from biological inspiration from the mammalian neocortex, is very well suited for processing various online streaming data, unsupervised learning, dealing with noise and unaligned sequential data – typical for many sensors found in Smart homes and Internet of things. We have carried out experiments with biological signals (namely ECG) and presented some ideas for new applications in context of Smart Homes or improvements to HTM/CLA algorithm.

Acknowledgement

The work has been supported by research grant No. SGS14/144/OHK3/2T/13.
Computational resources were provided by the MetaCentrum under the program LM2010005 and the CERIT-SC under the program Centre CERIT Scientific Cloud, part of the Operational Program Research and Development for Innovations, Reg. no. CZ.1.05/3.2.00/08.0144.

References

[1] Rockel, A. J., Robert W. Hiorns, and Thomas PS Powell. "The basic uniformity in structure of the neocortex." *Brain* 103, no. 2 (1980): 221-244.

[2] Sporns, Olaf, Christopher J. Honey, and Rolf Kötter. "Identification and classification of hubs in brain networks." *PloS one* 2, no. 10 (2007): e1049.

[3] Quiroga, R. Quian, Gabriel Kreiman, Christof Koch, and Itzhak Fried. "Sparse but not 'grandmother-cell'coding in the medial temporal lobe." *Trends in cognitive sciences* 12, no. 3 (2008): 87-91.

[4] Hawkins, Jeff, and Sandra Blakeslee. "*On intelligence*." Macmillan, (2007).

[5] Hawkins, Jeff, Ahmad, Subutai. And Dubinsky, Donna. " Hierarchical temporal memory including Cortical learning algorithms", (2011).

[6] Škoviera R.- "Image Classification Based on Hierarchical Temporal Memory and Color Features", Slovak Academy of Sciences, MEASUREMENT 2013, Smolenice (2013).

[7] Arel, Itamar, Derek C. Rose, and Thomas P. Karnowski. "Deep machine learning-a new frontier in artificial intelligence research [research frontier]."*Computational Intelligence Magazine, IEEE* 5, no. 4 (2010): 13-18.

[8] Jin Pan, Danny Bankman, Stanislav Nikolov, "Song learning", https://github.com/snikolov/nupic/tree/master/examples/opf/experiments/multistep/midi (2013)

[9] NuPIC et al., "NLP articles with HTM/CLA", community wiki, https://github.com/numenta/nupic/wiki/NLP-Projects (2014).

[10] Keithcom, "CLA controlled car", github, https://github.com/keithcom/nta_car (2014).

[11] Volna, E., Vitku, J., Otahal, M., Kotyrba, M., "ECG prediction based on classification via neural networks", (due 2015, paper in progress).

[12] Otahal, M., "new anomaly methods", github, source code PR, (2014)

[13] Thearn, "Webcam Pulse detector", github, https://github.com/thearn/webcam-pulse-detector (2014).

[14] Subutai Ahmad, "Anomaly detection using the cortical learning algorithm", Numenta hackathon slides, http://www.slideshare.net/numenta/anomaly-detection-subutai-may-2014-hackathon (2014).

Ing.Marek Otáhal
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague
Karlovo nám. 13, 12135 Praha 2, CZ

E-mail: otahama2@fel.cvut.cz