Poster: Anomaly Detection to Improve Security of Big Data Analytics

Tom Slooff[1], Francesco Regazzoni[2,1], Fabien Brocheton[3], Antonio Parodi[4], Radim Cmar[5*]
[1]Università della Svizzera italiana, Lugano, Switzerland, tom.slooff@usi.ch, francesco.regazzoni@usi.ch
[2]University of Amsterdam, Amsterdam, The Netherlands, f.regazzoni@uva.nl
[3]NUMTECH, Aubière, France, fabien.brocheton@numtech.fr
[4]Centro Internazionale di Monitoraggio Ambientale, Savona, Italy, antonio.parodi@cimafoundation.org
[5]Sygic, Bratislava, Slovakia, rcmar@sygic.com

ABSTRACT

Big data analytics largely rely on data. Because of their central role, it is fundamental to ensure the security and correctness of data used in these applications. Anomaly detection could help to increase the security of big data analytics applications. However, these applications are very diverse both for the properties of the data analyzed and for the computations to be carried out on them. As a result, the selection of the most appropriate anomaly detection method is a challenging and time consuming task for designers. Hierarchical Temporal Memory (HTM) is as an anomaly detection technique sufficiently generic to achieve satisfactory performance on a wide range of applications, thus suitable to ease the burden of selecting the anomaly detection method. To confirm this, in this paper we explore the performance of HTM on a dataset used for air quality prediction. Our preliminary results show that HTM achieves excellent performance when compared to other popular anomaly detection methods.

1 INTRODUCTION

Big data analytics applies advanced analytic techniques on extremely large data sets, commonly called big data. These data are typically characterized by high volume or high variety of data. As the name states, big data analytics largely rely on data. It is thus of crucial importance to ensure the correctness, the security and the integrity of the data themselves. Protection of data can be ensured by the use of appropriate cryptographic primitives, that, if correctly used and implemented in a robust way, can provide confidentiality, authentication, and integrity. A pivotal role in ensuring the integrity of the data computed by a system or exchanged between different computational nodes can also be played by anomaly detection. In a nutshell, anomaly detection learns the normal patterns of data and detects data points which fall outside of the norm. When used for security purposes, anomaly detection allows to identify anomalies in the data themselves, possibly introduced by malicious actors. To this end, anomaly detection has been used for intrusion detection or to detect malware activities. In big data analytics, anomaly detection nicely complements security techniques based on cryptographic algorithms, since it can be used to identify an adversary that, having legitimate access to the system, exploits this access to inject data altering the behaviour of the data analytics models.

Big data analytics techniques are often computational intensive applications whose workflow often requires a distributed and heterogeneous computation infrastructure composed of several nodes. To provide security, anomaly detection should be applicable at any needed point within the workflow. As such, anomaly detection is faced with the challenge of handling many types of time-series data presented to it. A workflow may require processing of batched data or real-time detection in streamed datasets. Since heterogeneous hardware is often used, the selected anomaly detection method should be sufficiently computationally efficient to be usable on any a large variety of platform, also those with constrained capabilities.

There is a plethora of anomaly detection techniques to choose from. One possibility is to let the designers tailor the technique exactly to the requirements of the target application. This approach is very likely to produce the best results. However, the burden of the selection of the technique, its implementation, and its training are completely on the shoulders of the designer. Another approach consists in providing a library of anomaly detection techniques that are applicable to any workflow. This approach could lead to a less optimal solution compared to the previous approach, but, if it reaches sufficient performance to fulfill the need of the target application, would make the task of designers much simpler. In this paper we explore Hierarchical Temporal Memory as such generic technique on a dataset used for air quality prediction. In the rest of this paper we introduce Hierarchical Temporal Memory, we present the initial results we obtained, and we highlight the next steps.

2 HIERARCHICAL TEMPORAL MEMORY

Hierarchical Temporal Memory (HTM) is a biologically-inspired algorithm modeled after the neuron activation of the neocortex [2]. The pyramidal neuron is modeled with proximal dendritic segments which receive feedforward input data, and with distal connections which receive lateral input from other neurons. The internal data structure of HTM is the sparse distributed representation (SDR), which represents the firing of neurons at a particular point in time. The pipeline of HTM starts with an encoder. Encoders transform raw input data to binary arrays, maintaining semantic information from the original data through overlap properties on these arrays. Next, the spatial pooler transforms the binary arrays to SDR representing mini-column activations. Through Hebbian learning, the spatial pooler becomes sensitive to common input patterns. Mini-columns contain multiple neurons. Where the activation of the mini-columns represents the input data, the activation of specific neurons within the mini-columns encodes the temporal context. In

[*]This work is partially supported by the European Union Horizon 2020 Program under the EVEREST project (grant agreement No 957269)

the last step of the pipeline, the temporal memory learns temporal patterns through distal connections, and encodes temporal context with neuron activations within mini-columns of the SDR.

HTM performs learning online [1]. It is an efficient algorithm which allows for real-time detection of anomalies in streamed data. The encoders used for HTM provide a key benefit of the algorithm, as they allow the algorithm to work with any data for which an encoder is available. Furthermore, previous work found HTM is not sensitive in the hyperparameters [1]. These important features enable HTM to be used as a universal algorithm, where the human effort lies mainly in choosing encoders, and minimal tuning of the hyperparameters is required. For the above reasons, we selected HTM over other popular techniques such as Long Short Term Memory models, AutoEncoders, Autoregressive Integrated Moving Average, Isolation Forests, and Support Vector Machines.

3 PRELIMINARY RESULTS

Our goal is to compare the behaviour of HTM against against other popular anomaly detection techniques and assess its suitability for increasing the security of big data analitcs applications. In this section we present the preliminary results we obtained using a dataset containing climate temperature data from sensors located at an industrial site, which are used for air quality prediction [3]. Specifically, the model is compared against an Autoregressive Integrated Moving Average (ARIMA) model, an Error, Trend, Seasonality (ETS) model, and lastly the Long Short Term Memory (LSTM) model.

Since the available dataset does not inherently contain anomalies, these had to be synthetically generated. We did so adapting original data points with values sampled from a normal distribution centred on 0 with standard deviation equal to that of the first order difference of the dataset multiplied with 3. This method is chosen because it does not require modeling the underlying distribution and it is generally applicable, also to multi-variate datasets we will test in the future. The multiplication factor of 3 has been empirically determined through experimentation.

Due to the sampling from the normal distribution, most anomalies are relatively close to the original values. This indeed mimics anomalies that could be inserted by a clever adversary. A minority of anomalies will be very different from the original values, making these easier to detect. The performance of the models on running time and anomaly detection can be seen in figures 1 and 2 respectively. On both aspects, the HTM shows excellent performance: achieving a running time faster than the second-best technique with a factor 9, and an area under curve at least 0.1 higher than all other techniques.

4 CONCLUSIONS

Anomaly detection is an interesting tool to enhance security in big data analytics. The diverse nature of big data analytics applications makes the selection of the exact technique a challenging task for application designers. To ease this task, we explored the use of anomaly detection techniques that can be applied universally. Among them, we focused on HTM. The preliminary results show excellent performance of this technique compared to other anomaly detection methods. The next challenge will be the automatic selection of the encoders that maintain semantic information, as well as the automatic tuning of the model. Furthermore, HTM will be validated in other use cases.

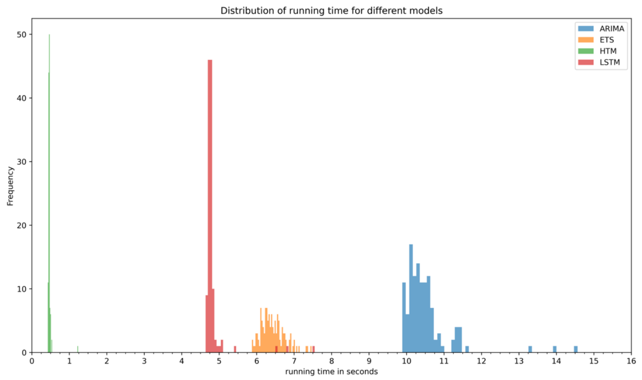

Figure 1: Histogram of the running time

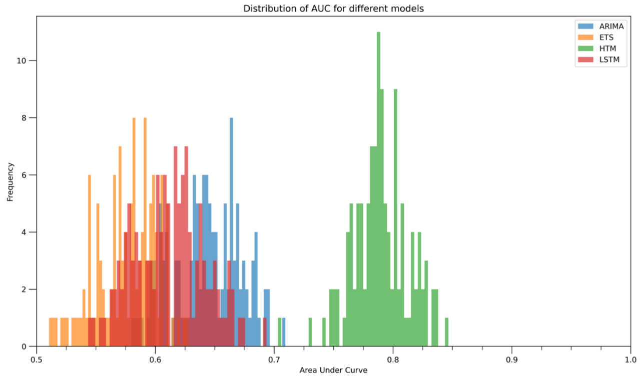

Figure 2: Histogram of the area under curve of the receiver operating characteristic curve

REFERENCES

[1] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. 2017. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* 262 (2017), 134–147. https://doi.org/10.1016/j.neucom.2017.04.070 Online Real-Time Learning Strategies for Data Streams.

[2] Jeff Hawkins and Subutai Ahmad. 2016. Why Neurons Have Thousands of Synapses, a Theory of Sequence Memory in Neocortex. *Frontiers in Neural Circuits* 10 (2016). https://doi.org/10.3389/fncir.2016.00023

[3] Christian Pilato, Stanislav Bohm, Fabien Brocheton, Jeronimo Castrillon, Riccardo Cevasco, Vojtech Cima, Radim Cmar, Dionysios Diamantopoulos, Fabrizio Ferrandi, Jan Martinovic, Gianluca Palermo, Michele Paolino, Antonio Parodi, Lorenzo Pittaluga, Daniel Raho, Francesco Regazzoni, Katerina Slaninova, and Christoph Hagleitner. 2021. EVEREST: A design environment for extreme-scale big data analytics on heterogeneous platforms. In *2021 Design, Automation Test in Europe Conference Exhibition (DATE)*. 1320–1325. https://doi.org/10.23919/DATE51398.2021.9473940