Anomaly detection of large scale distributed storage system based on machine learning

INSTITUTE OF HIGH ENERGY PHYSICS, CAS

WANG LU (LU.WANG@IHEP.AC.CN)

Agenda

Introduction

- Challenges and requirements of anomaly detection in large scale storage systems
- Definition and category of anomaly
- Category of anomaly detection methods

Two machine learning algorithms

- Isolation Forest
- Hierarchical Temporal Memory(HTM)
- Preliminary results with Ganglia monitoring data

Plan of future development

Introduction

In large scale storage system, anomaly detection is a crucial component

- reduces impacts of hardware and software failures
- prevents further lost from human mistakes or malicious intrusion

Current methods

- static threshold on performance metrics,
- active probing such as ping, touch, df etc.,

Challenges

- system scale,
- variety and changeability of continuous workload
- massive monitoring data coming from various sources (also opportunity for machine learning based detection)

Ideal Properties of Anomaly Detection Algorithms

Data driven, machine learning based, little dependency on human experience,

Runs in unsupervised, automated way,

Scale to multivariate massive monitoring data,

High detection efficiency,

Real time Detection,

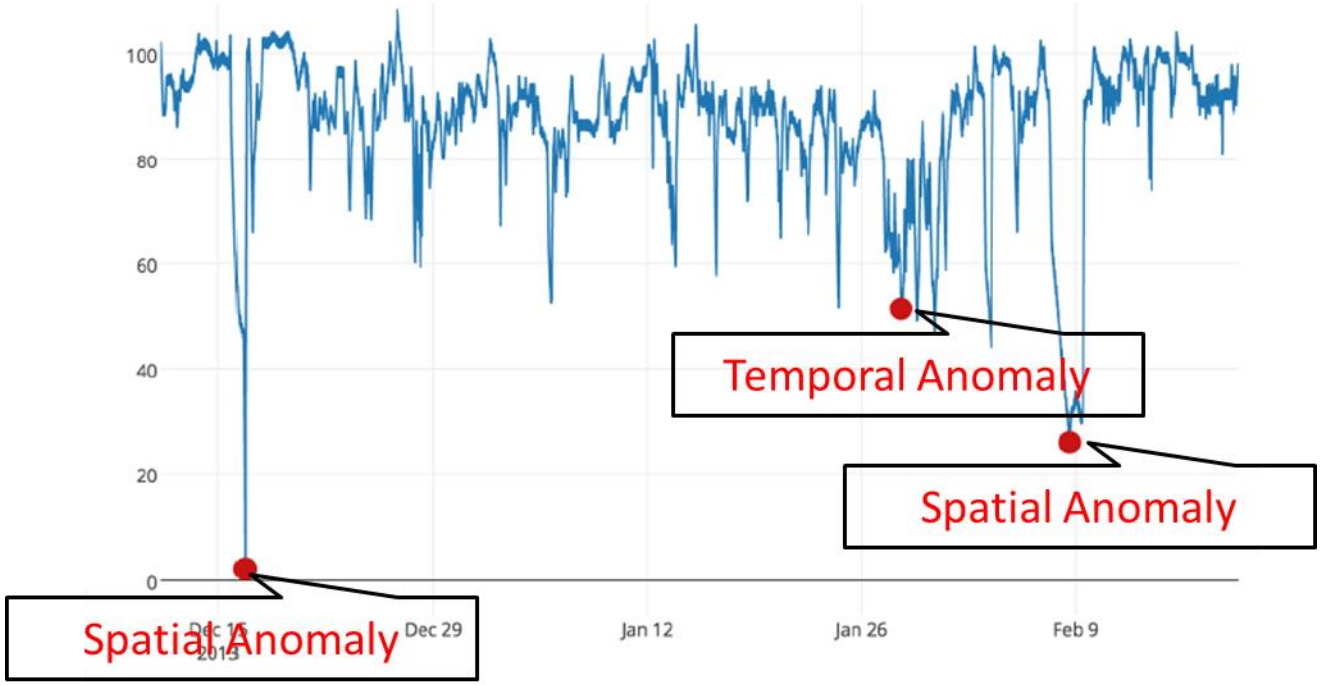
on-line training,

…

What is Anomaly in Machine Learning ?

Anomalies are data patterns that have different characteristics from normal instances,

- Spatial anomaly
- Temporal anomaly



Category of Anomaly Detection Methods

Statistics-based method

- for example, fit a Gaussian distribution and set a threshold,
- Dependency on human experience to set the threshold

Classification-based method

- train a binary classification model(Xgboosting, MLP, SVM …) by labeled data
- requires labeled dataset, offline training, cannot easily adapt to changes of workload and system configuration

Cluster-based method

- unsupervised learning , do not require labeled dataset
- computing intensive, cannot scale

Category of Anomaly Detection Methods

Cutting-based method

- anomalies are “minority and different”, much more susceptible to isolation than normal points
- unsupervised, scale to massive dataset
- requires offline training, human setting of anomaly ratio
- represented algorithm: Isolation Forest

“Prediction+ Deviation Measurement” based method

- trains prediction models with normal time series,
- Deviations between model prediction and real subsequent behavior represents degrees of abnormal
- unsupervised/semi-supervised way, do not necessarily need offline training,
- possible to exploit the advantages of deep learning algorithms,
- represented algorithms: HTM

Isolation Forest

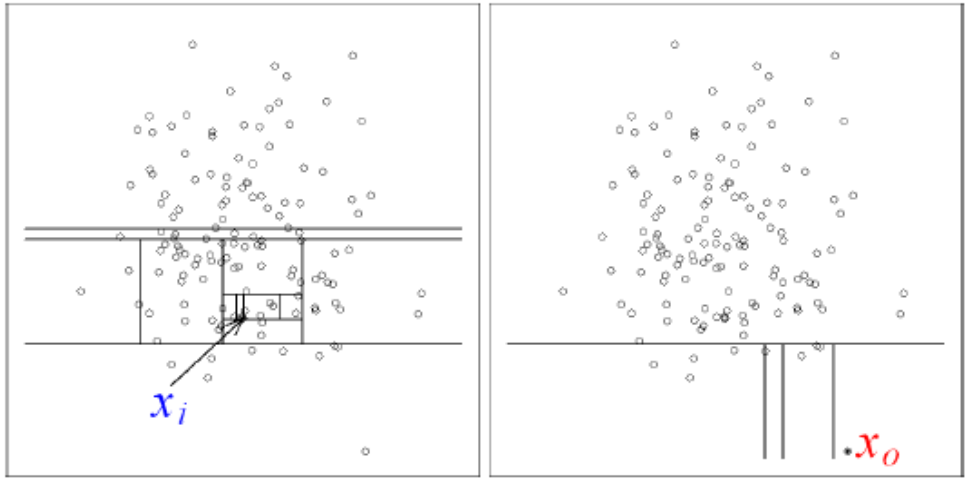
Isolation Forest[1]builds a set of binary trees from dataset: X = {x_1, ..., x_n} of n instances from a *d*-variate distribution.

- to build one isolation tree, recursively divide X by randomly selecting an attribute q and a split value p, until either: (i) the tree reaches a height limit, (ii) |X| = 1 or (iii) all data in X have the same values,
- for a new data point *x* , its path length h(*x*) of a iTree is the number of edges *x* traverses from the root node to an external node,
- its anomaly score is

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}},$$

$$c(n) = 2H(n-1) - (2(n-1)/n)$$

$$H(i) = \ln(i) + 0.5772156649$$



(a) Isolating x_i (b) Isolating x_o

Isolation Forest

Linear time complexity with a low constant,

training stage: $O(t\psi \log \psi)$

evaluating stage: $O(nt \log \psi)$

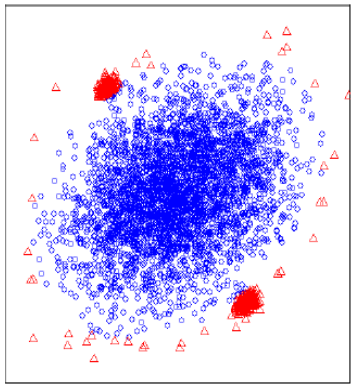
Empirical evaluation show that

- h(x) converges with a small number of trees(t)
- training with a sub sample gives comparable result with training on full dataset
  - Default setting of sub sampling (Ψ): 128

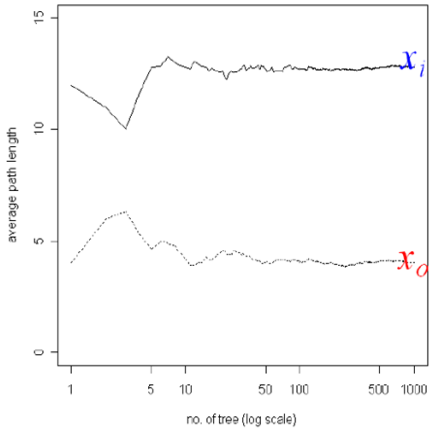
Very effective on large, high dimensional data

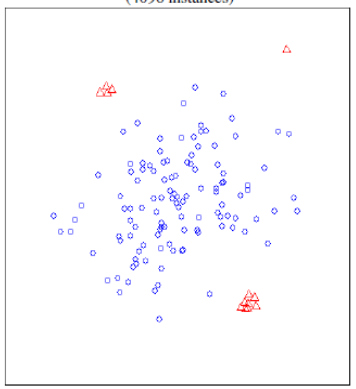
Favorable results comparing to other spatial AD algorithms

Widely used, included in scikit-learn package



(a) Original sample
(4096 instances)



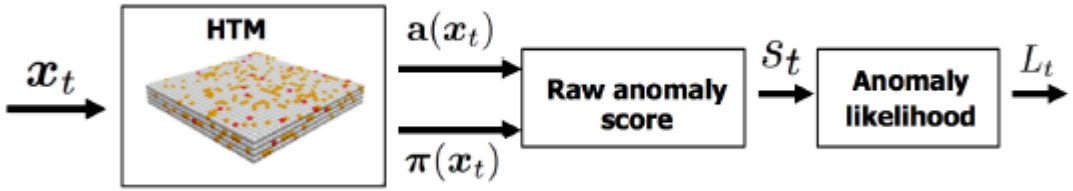


(b) Sub-sample
(128 instances)

HTM

Hierarchical Temporal Memory(HTM)[2] is a machine learning algorithm derived from neuroscience that models spatial and temporal patterns in streaming data.

HTM based anomaly detection[3]



Given an input x_t, the vector $\mathbf{a}(x_t)$ is a sparse binary code representing the current input.

vector $\pi(x_t)$ represents a prediction for $\mathbf{a}(x_{t+1})$

HTM

Raw Anomaly Score

$$s_t = 1 - \frac{\pi(x_{t-1}) \cdot \mathbf{a}(x_t)}{|\mathbf{a}(x_t)|}$$

Anomaly Likelihood

- assumes anomaly scores in a large time windows follows a Gaussian distribution
- use average value of small window to smooth impacts of noisy
- set a threshold on Gaussian tail probability to detect anomaly

This part can also be used with result of other prediction algorithm in time series scenarios!

$$\mu_t = \frac{\sum_{i=0}^{i=W-1} s_{t-i}}{k}$$

$$\sigma_t^2 = \frac{\sum_{i=0}^{i=W-1} (s_{t-i} - \mu_t)^2}{k-1}$$

$$L_t = 1 - Q\left(\frac{\tilde{\mu}_t - \mu_t}{\sigma_t}\right) \qquad W' \ll W$$

$$\tilde{\mu}_t = \frac{\sum_{i=0}^{i=W'-1} s_{t-i}}{j}$$

Typically set W=8000, W'=10, threshold of $L_t = 1 - 10^{-5}$

HTM

The HTM out perform peering algorithms on the benchmark NAB[4]

The Numenta Platform for Intelligent Computing (NuPIC) is a machine intelligence platform that implements the HTM learning algorithms,

With NuPIC, AD task runs in an unsupervised, on line detection way

- separate models for separate detection,
- need warming time, at the start of detection and training, model outputs high anomaly scores

Currently, we 100% reused the configuration of network in the demo example of NuPIC

Experiments

Dataset

- 44 days's ganglia monitoring data of 10 Lustre metadata servers
- 15 metrics:

  bytes_in_value, bytes_out_value, cpu_idle_value, disk_free_value, load_fifteen_value, load_five_value,load_one_value,mem_buffers_value,mem_cached_value,swap_free_value

  mem_free_value,pkts_in_value,pkts_out_value,proc_run_value,proc_total_value,

- Collecting Frequency:
  - 5mins/data point

Results of Isolation Forest

Training data

- monitoring data of 10 MDS in November,2018

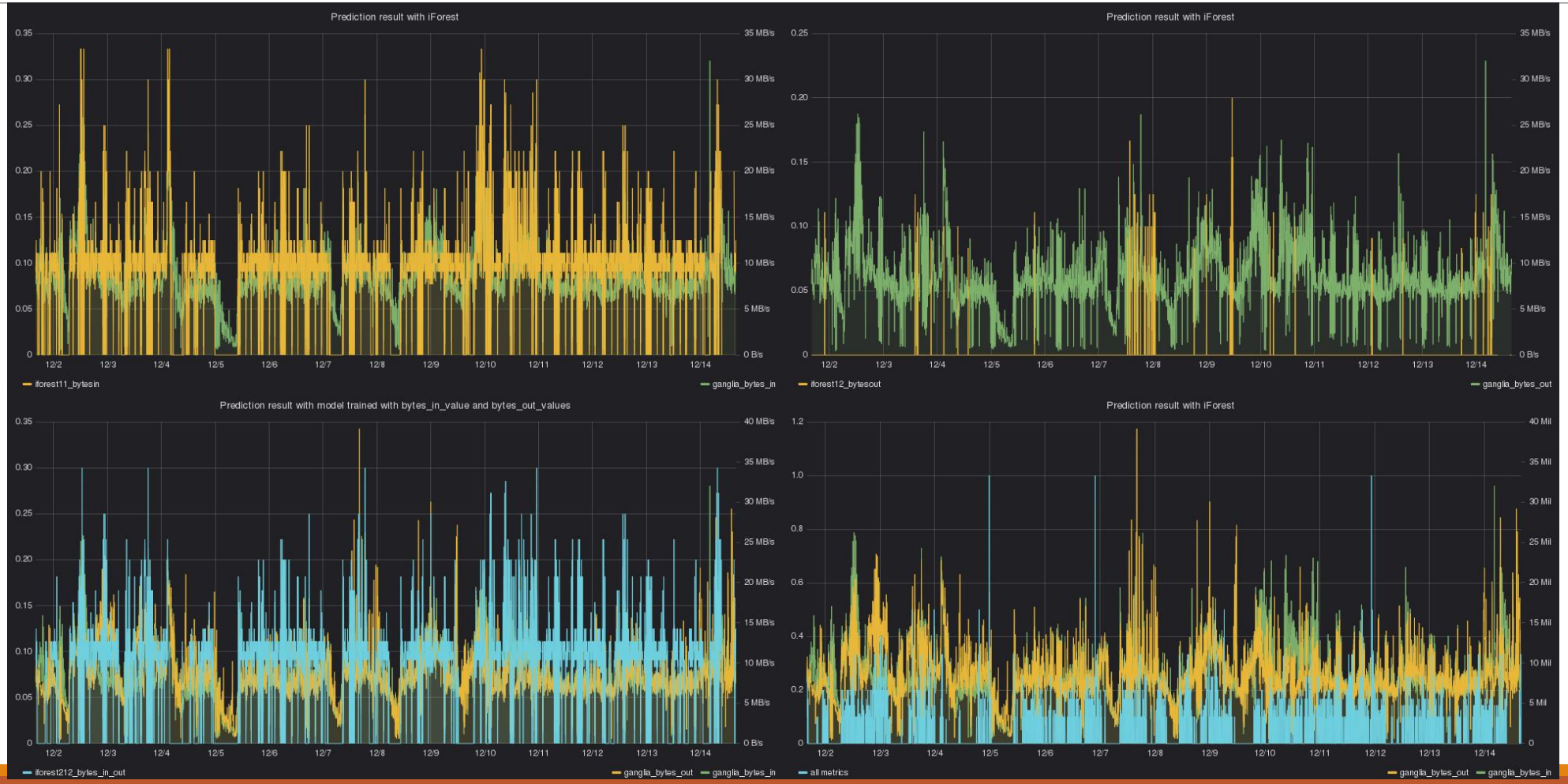
Testing data

- monitoring data of 1 MDS in two weeks(Dec.1-Dec.14, 2018)

Two most interesting metrics

- bytes_in_value and bytes_out_value
- threshold metrics of current anomaly detection method on production system

Result of Isolation Forest



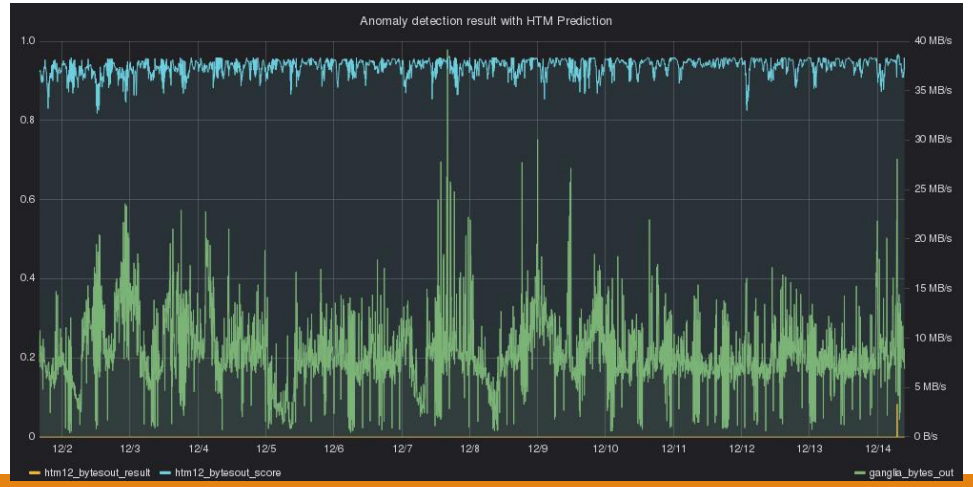
Result of Isolation Forest

Overlaps of ranked anomaly scores(totally 8211 entries)

Method1	Method2	Top 10	Top 50	Top 100	Top 200	Top 1000
IForest_bytes_in	IForest_bytes_out	0	0	3	61	693
IForest_bytes_in	IForest_bytes	9	25	31	96	759
IForest_bytes_in	IForest_all	4	23	42	120	708
IForest_bytes_out	IForest_bytes	0	17	71	165	896
IForest_bytes_out	IForest_all	0	11	50	137	855
IForest_bytes	IForest_all	4	26	61	156	909

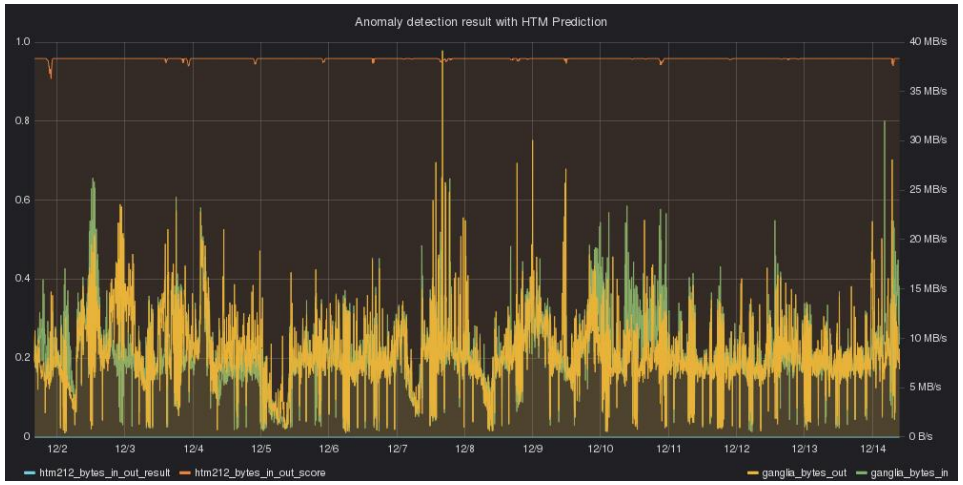
Results of HTM

- Online model trained with ganglia metric of a Lustre MDS
- Single metric:
  - "bytes_in_value"
- Training sample: *45 days, 5mins/point*
- Prediction result on left figure: *from Dec1 –Dec14*
- Average Anomaly likelihood: 0.951
- Detection result: 0 (since we set L_t =1-10^{-5})

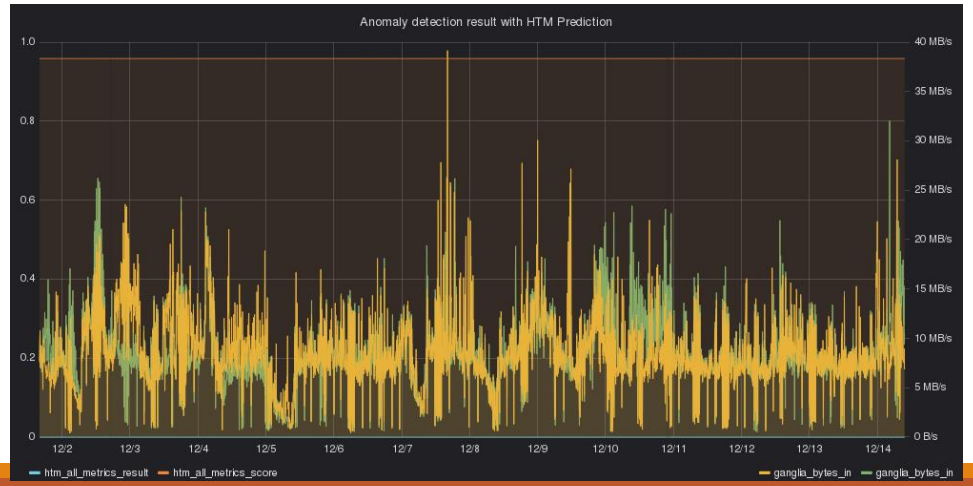


- Online model trained with ganglia metric of a Lustre MDS
- Single metric:
  - "bytes_out_value"
- Training sample: *45 days, 5mins/point*
- Prediction result on left figure: *from Dec1 –Dec14*
- Average Anomaly likelihood: 0.937
- Detection result: 0 (since we set L_t =1-10^{-5})

Results of HTM

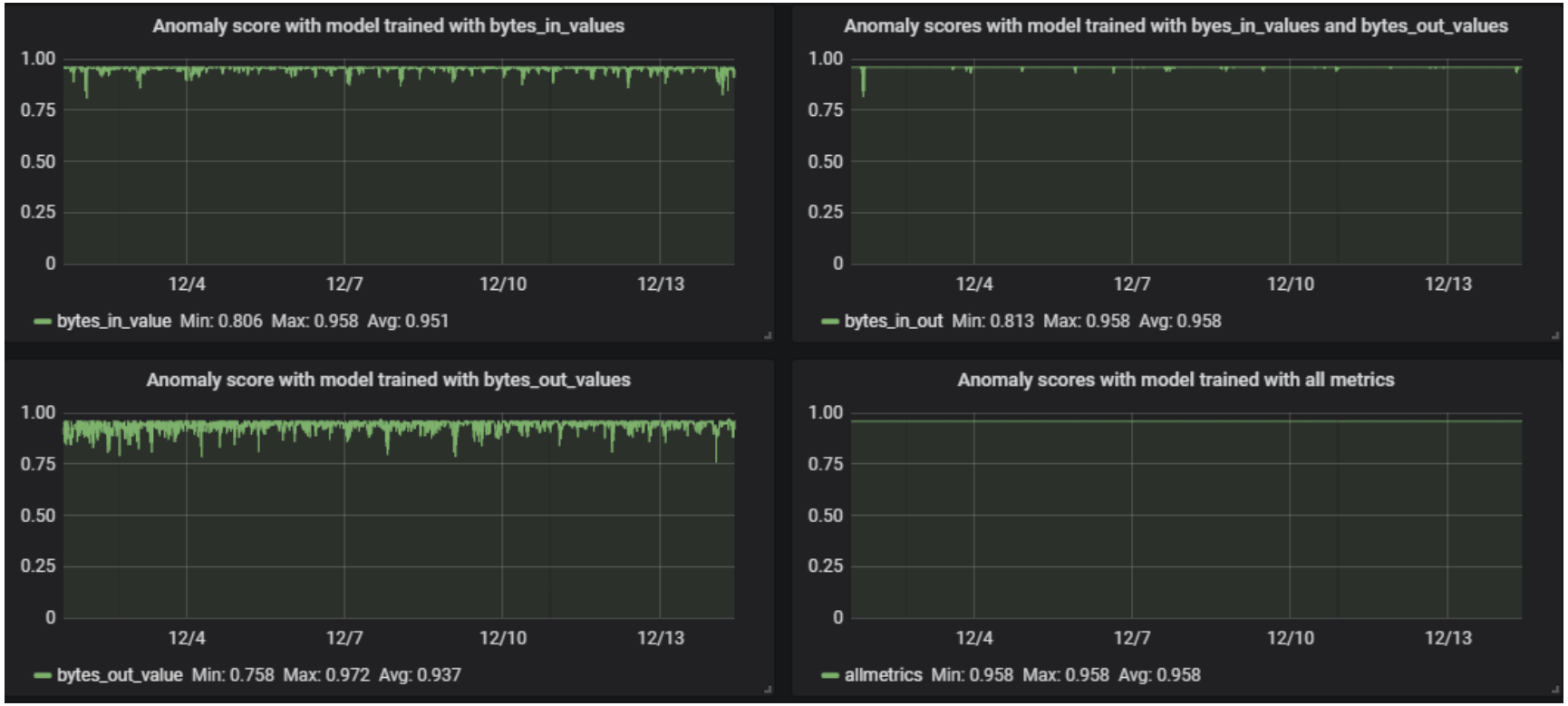


- Online model trained with ganglia metric of a Lustre MDS
- Two metrics:
  - "bytes_in_value" and "bytes_out_value"
- Training sample: *45 days, 5mins/point*
- Prediction result on left figure: *from Dec1 –Dec14*
- Average Anomaly likelihood: 0.958
- Detection result: 0 (since we set L_t =1-10^{-5})



- Online model trained with ganglia metric of a Lustre MDS
- Single metric:
  - All the ganglia metrics
- Training sample: *45 days, 5mins/point*
- Prediction result on left figure: *from Dec1–Dec14*
- Average Anomaly likelihood : 0.958
- Detection result: 0 (since we set L_t =1-10^{-5})

Results of HTM



- The high anomaly likelihood indicates lack of training sample ? programming error?

Next step

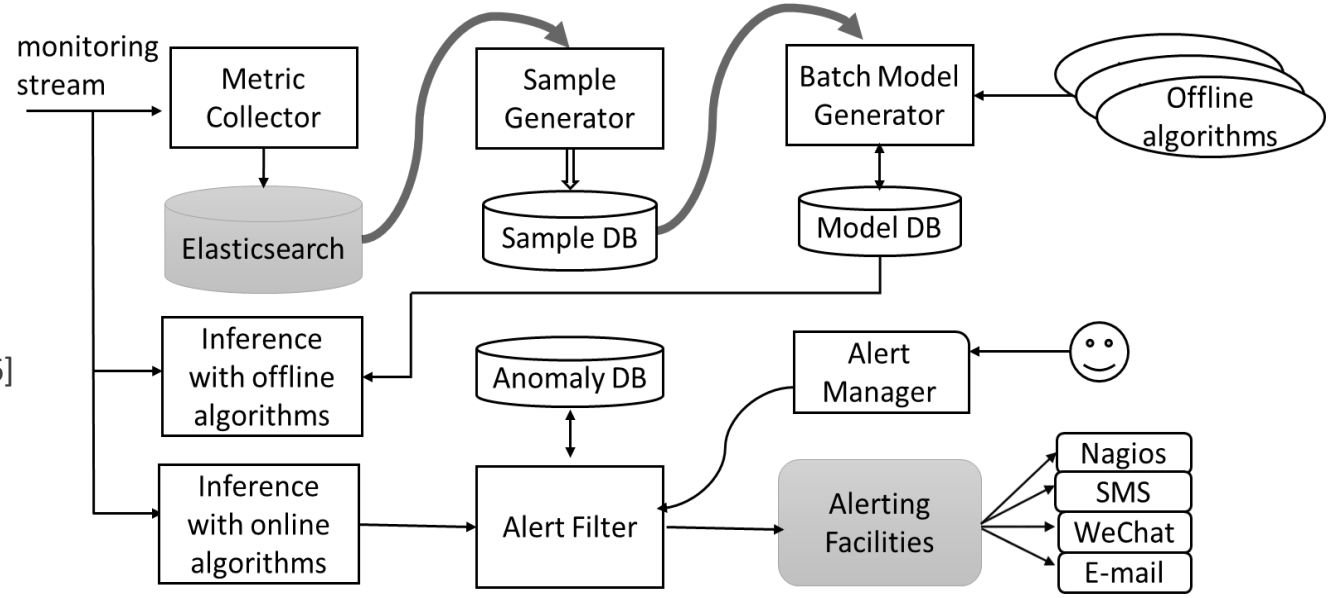
Better understand the mechanism of HTM

Use RNNs as prediction algorithm in time series anomaly detection

Implement a python based framework to facilitates AD tasks in IT OPS

- Sample cleaning and building,
- Training configurator and monitor,
- Alert filtering and threshold setting,
- Visualization and retagging of anomaly
- Statistic of anomaly detection

Example: Yahoo/EGADS[5], Tencent/Metis[6]



Summary

Anomaly detection is a crucial and challenging task in building a robust large scale storage system,

We have started studies on this task since last fall,

- Through studies of related paper, we found cut-based algorithms and “prediction + deviation measurement ” algorithms are two promising methods
- We made some experiments to drive our standing of the two methods, further research of inside mechanisms is still needed to explain some of the results

A generic, extensible framework will save efforts of development and speed up the implementation and evaluation cycle of a new algorithm(“ there is no one-fits-all method”)

A public dataset coming from everyday IT operations of data center will definitely be helpful!

- Like ImageNet in computer vision domain

Thank you!

References

1. Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest“. Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. IEEE, 2008.

2. Y. Cui, S. Ahmad, and J. Hawkins. “Continuous Online Sequence Learning with an Unsupervised Neural Network Model”.Neural Computation 28, 2474–2504 (2016)

3. Subutai Ahmad, Alexander Lavin, Scott Purdy, Zuha Agha, “Unsupervised real-time anomaly detection for streaming data”,Neurocomputing,Volume 262,2017,Pages 134-147,ISSN 0925-2312,https://doi.org/10.1016/j.neucom.2017.04.070.

4. Lavin, Alexander and Ahmad, Subutai. “Evaluating Realtime Anomaly Detection Algorithms the Numenta Anomaly Benchmark”. In 14th International Conference on Machine Learning and Applications (IEEE ICMLA’15), Miami, Florida, 2015. IEEE. doi: 10.1109/ICMLA.2015.141.

5. A Java package to automatically detect anomalies in large scale time-series data https://github.com/yahoo/egads

6. https://github.com/Tencent/Metis/blob/master/README.en.md