A way of anomaly detection in engineering equipment characteristics of Symmetra at IHEP IT center

Mrs. Popova Ekaterina, Mr. Kotliar Viktor

NRC “Kurchatov Institute” - IHEP

Abstract

The information flow should be monitored on anomaly detection. It is important, because it allows you to see a possible problem in advance and prevent it from turning into a real one. A huge flow of diverse data within the modern computing center flows from everywhere. As a rule, these are time series - numerical characteristics that are consistently measured after some time intervals.

At this work there was developed the way of analysis for engineering equipment characteristics in centralized system of uninterrupted power supply (Symmetra) at IHEP IT center. When tracking time series, extracted from the data processing and storage system, anomalies are detected using the Twitter AnomalyDetection package. The information on problem is provided to the engineering and operational staff.

Introduction

Anomaly or outlier is an object that differs from most other objects. The percentage of anomalies in data stream may be small, but we should track them, otherwise it can lead to serious consequences. Main challenges when we try to find anomalous events or objects are:

- the boundary between normal and anomalous behavior is often not precise;
- defining normal behavior is not trivial;
- normal behavior is changeable in some knowledge areas and it requires to permanent tracing;
- anomaly is different for different application domains;
- degree to which class labels (anomaly or normal) are available for at least some of the data;
- data contains noise.

There is a broad spectrum of anomaly detection techniques. They are: classification based, clustering based, nearest neighbor based, statistical, information theoretic, spectral.

The use of this or that technique depends on the type of task for anomaly detection. To formulate the task we need to determine nature of the input data, the availability or unavailability of labels for data, the constraints and requirements for knowledge area. Input data can be sequential, spacial or graph. Classical example of a sequential data is a time series.

It is also important to understand the essence of the desired anomaly. There are three types of anomalies: point, contextual, collective.

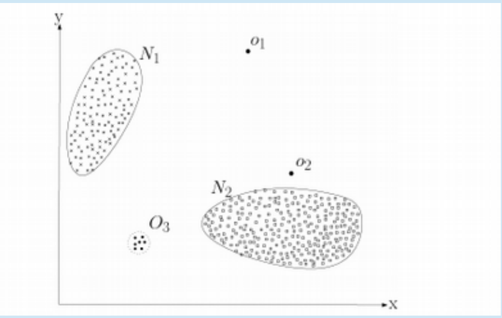


Fig.1 Point

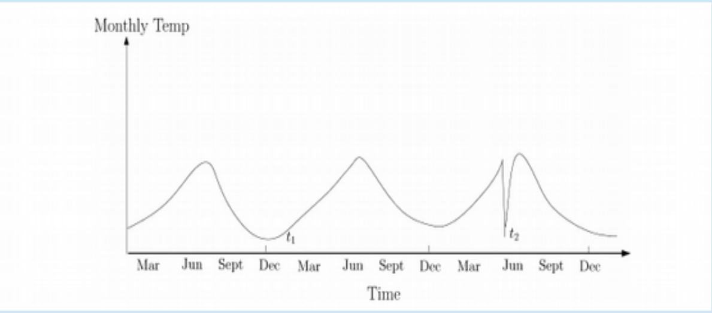


Fig.2 Contextual

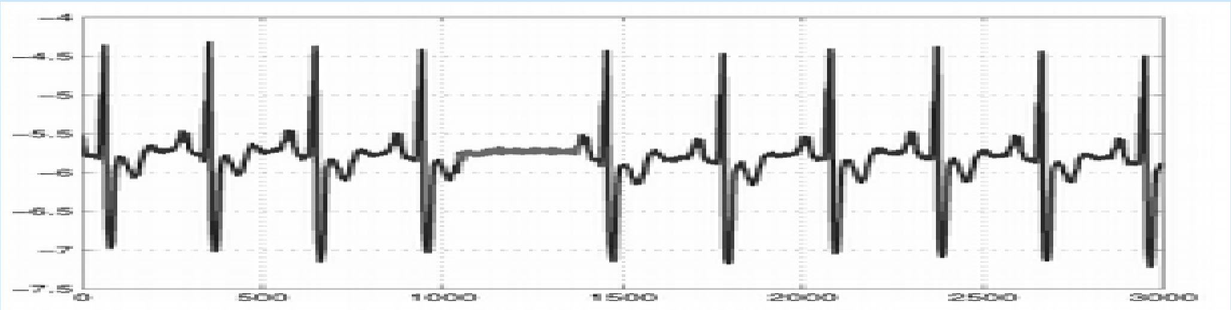


Fig.3 Collective

Based on the extent to which the labels are available, anomaly detection techniques can operate in one of the following three modes:

- Supervised anomaly detection
- Unsupervised anomaly detection
- Semi-supervised anomaly detection

Typically, the outputs produced by anomaly detection techniques can be scores or labels.

Statistical techniques

Key: “Normal data instances occur in high probability regions of a stochastic model, while anomalies occur in the low probability regions of the stochastic model.”

– *parametric*

normal data: f(x, ɵ) - the probability density function where x is an observation and ɵ are params;

abnormal data: the inverse of the f(x, ɵ)

– Gaussian Model (params are estimated on Maximum Likelihood Estimates (MLE))

1. 3σ rule - μ ± 3σ region contains 99.7% of the data instances
2. The box plot rule - [Q1 − 1.5IQR; Q3 + 1.5IQR] contains 99.3% of observations (Q1 – lower quartile, Q3 – upper quartile, IQR = Q3 – Q1 – Inter Quartile Range)
3. Grubb’s test
4. The student’s t-test
5. Hotelling t^2-test
6. χ^2 statistic, etc.

– Regression Model

1. Akaike Information Content (AIC)
2. Autoregressive Integrated Moving Average (ARIMA)
3. Autoregressive Moving Average (ARMA), etc.

– *nonparametric*

the model structure is not defined by default, but is instead determined from given data

Twitter AnomalyDetection

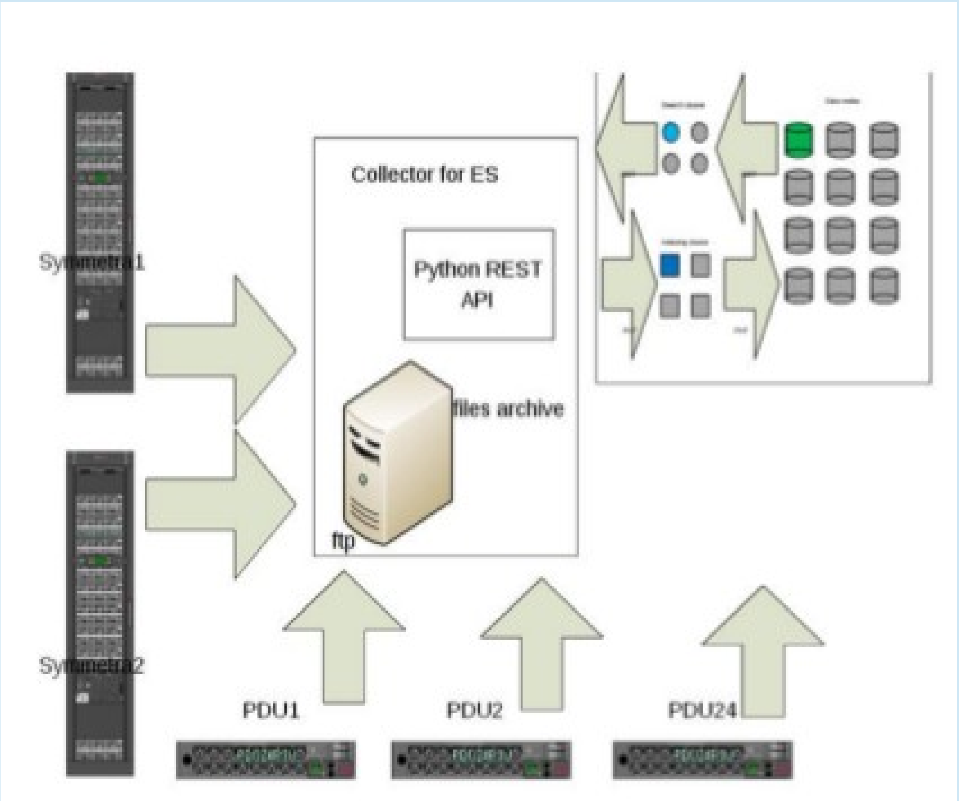
Algorithm 1 S-ESD Algorithm

```
Input:
X = A time series
n = number of observations in X
k = max anomalies (iterations in ESD)
Output:
$X_A$ = An anomaly vector wherein each element is a tuple
(timestamp, observed value)
Require:
$k \leq (n \times .49)$
1. Extract seasonal component $S_X$ using STL Variant
2. Compute median $\tilde{X}$
/* Compute residual */
3. $R_X = X - S_X - \tilde{X}$
/* Detect anomalies vector $X_A$ using ESD */
4. $X_A = ESD(R, k)$
return $X_A$
```

Seasonal Hybrid ESD (S-H-ESD) builds upon the S-ESD algorithm. S-H-ESD uses more robust statistical techniques and metrics such as median and MAD (Median Absolute Deviation). Run time of S-H-ESD is higher than that of S-ESD and in cases where the time series under consideration is large but with a relatively low anomaly count, it is advisable to use S-ESD.

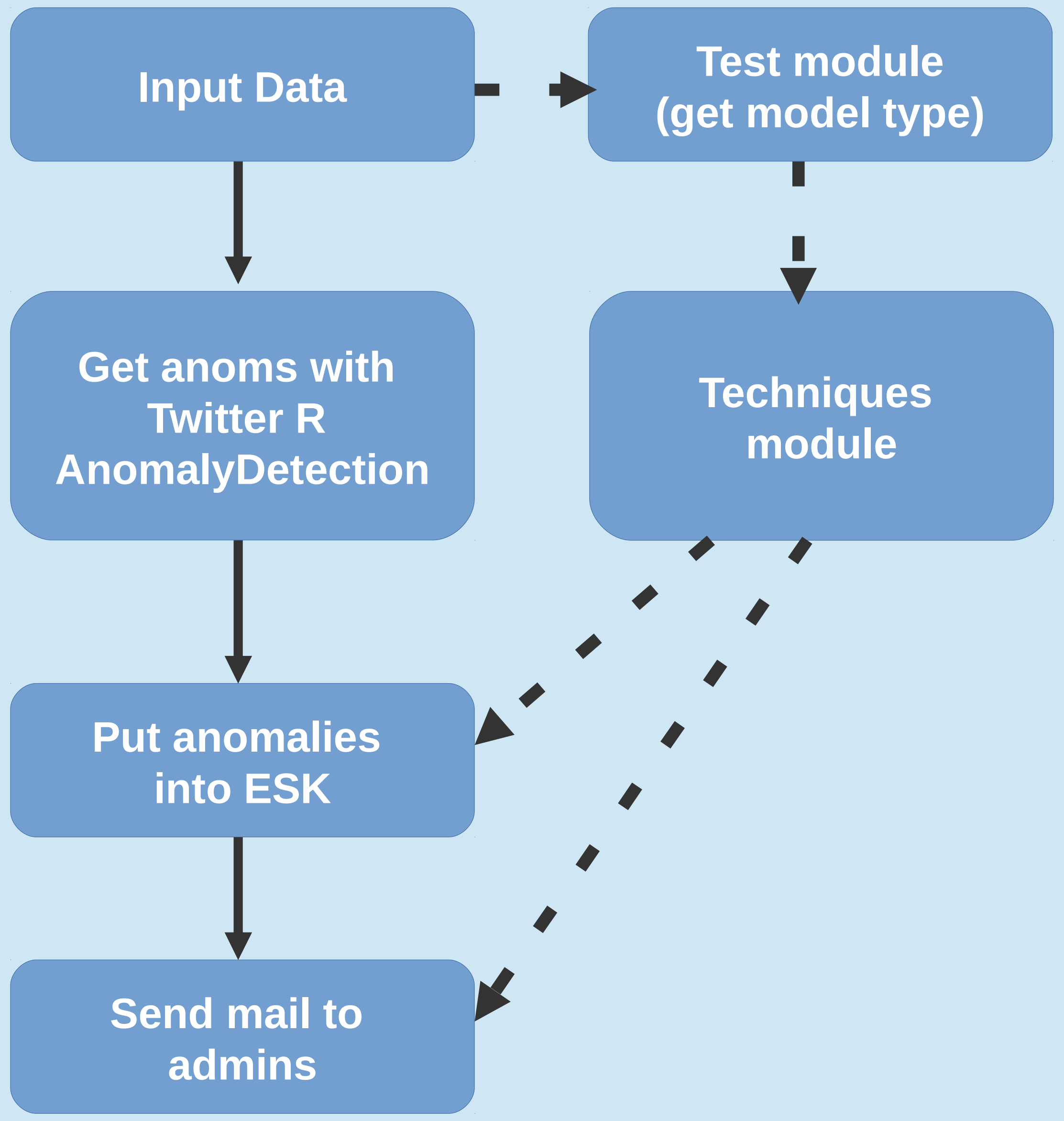
From Symmetra to ES

Symmetra is a high-efficiency 3-phase UPS that is scalable as data center grows up

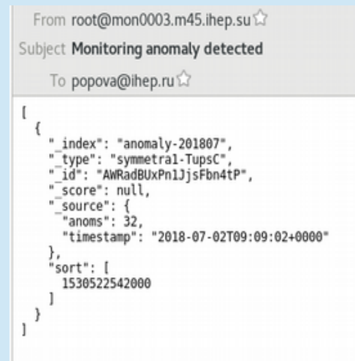


“Elasticsearch (ES) is an open-source, broadly-distributable, readily-scalable, enterprise-grade search engine. Accessible through an extensive and elaborate API, Elasticsearch can power extremely fast searches that support data discovery applications. “

It gathers UPS metrics from two APC Symmetra PX 160kW and more than 20 APC PDU through internal feature of APC to store data on a remote ftp server. Then all these data are parsed with python programs and are put through REST API to the ElasticSearch cluster. Next, by cron, we extract metrics (Ibat, temperature) from ES and try to catch anomalous events using Twitter R DetectAnomaly package. If anomalies exist, we put them into ES and send email to administrators.



Results

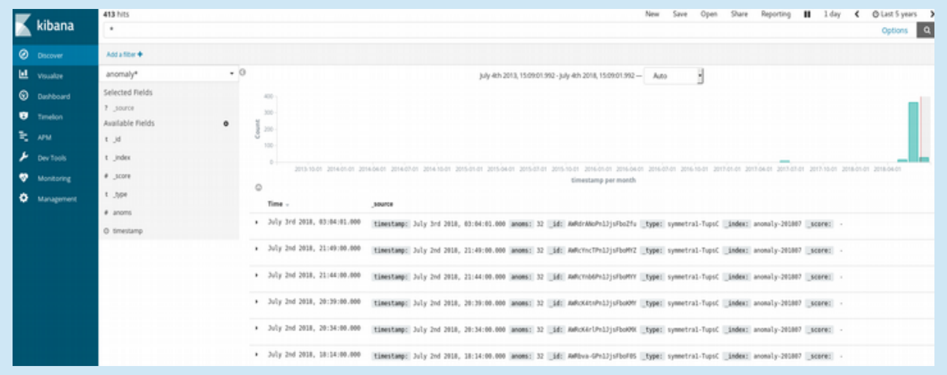


From root@mon0003.m45.ihep.su

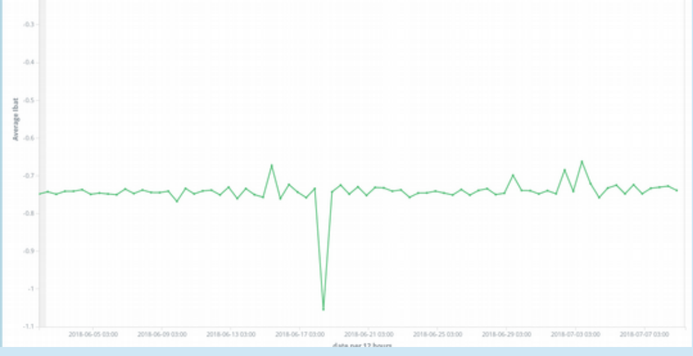
Subject Monitoring anomaly detected

To popova@ihep.ru

```
[
  {
    "_index": "anomaly-201807",
    "_type": "symmetra1-TupsC",
    "_id": "AWRadBUxPn1JjsFbn4tP",
    "_score": null,
    "_source": {
      "anoms": 32,
      "timestamp": "2018-07-02T09:09:02+0000"
    },
    "sort": [
      1530522542000
    ]
  }
]
```



```
July 3rd 2018, 03:04:01.000   timestamp: July 3rd 2018, 03:04:01.000 anoms: 32 _id: AWRdrANoPn1JjsFboZfu _type: symmetra1-TupsC _index: anomaly-201807 _score: -
July 2nd 2018, 21:49:00.000   timestamp: July 2nd 2018, 21:49:00.000 anoms: 32 _id: AWRcYncTPn1JjsFboMYZ _type: symmetra1-TupsC _index: anomaly-201807 _score: -
July 2nd 2018, 21:44:00.000   timestamp: July 2nd 2018, 21:44:00.000 anoms: 32 _id: AWRcYnb6Pn1JjsFboMYY _type: symmetra1-TupsC _index: anomaly-201807 _score: -
July 2nd 2018, 20:39:00.000   timestamp: July 2nd 2018, 20:39:00.000 anoms: 32 _id: AWRcK4tnPn1JjsFboKMY _type: symmetra1-TupsC _index: anomaly-201807 _score: -
July 2nd 2018, 20:34:00.000   timestamp: July 2nd 2018, 20:34:00.000 anoms: 32 _id: AWRcK4rlPn1JjsFboKMX _type: symmetra1-TupsC _index: anomaly-201807 _score: -
July 2nd 2018, 18:14:00.000   timestamp: July 2nd 2018, 18:14:00.000 anoms: 32 _id: AWRbva-GPn1JjsFboF0S _type: symmetra1-TupsC _index: anomaly-201807 _score: -
```



At this stage, we received a non-interactive script system for anomaly detection.

```
detectAnomaly_func <- function(index, type, source) {
  connect(es_host = "localhost")
  sourceList <- strsplit(source, ",")
  if (length(sourceList[[1]]) != 2) {
    stop("Errors in source list", call.=FALSE)
  }
  date <- sourceList[[1]][[2]]
  param <- sourceList[[1]][[1]]

  typeData <- Search(index = index, type = type, size = 10000,
    sort = paste(date, ":desc", sep=""),
    source = source)$hits$hits
  dateParam <- unique(sapply(typeData, '[[', 5),MARGIN = 2)
  tsData  <- as.POSIXct(strptime(dateParam[date,],
    "%Y-%m-%dT%H:%M:%S"))
  attributes(tsData)$tzone <- "UTC"
  tsFrame <- data.frame(tsData,unlist(dateParam[param,]))
  tsFrame <- tsFrame[nrow(tsFrame):1,]
  anomaly <- AnomalyDetectionTs(tsFrame, max_anoms=0.01,
alpha=0.05,
    direction='both', plot=FALSE, only_last='hr')
  return (anomaly[["anoms"]])
}
```

It allows us to track the anomalous events in electricity supply system (Symmetra) in realtime. Well-timed identified anomalies prevent expensive equipment from damage and reduce downtime in the computer center. But we got many incorrect anomalies and false positives make our work inaccurate. Furthermore we analyzed only two params (Ibat and Temp) separately.

Future work

In this work we considered the notion of anomaly, anomaly types, data processing methods. Statistical methods are described more detailed, so they are most suitable for working with time series. Further, using the set of real-time scripts, the Ibat, Temp metrics are extracted from the ESK stack. These data are checked for anomalies using the Twitter R AnomalyDetection package. The found anomalies were returned to the ESK, the letter was sent to system administrators. The obtained results help in tracking problems in the equipment maintenance and operation, but they are numerous and inaccurate. In addition, the parameters are hardly defined.

In future, we plan to test other statistical methods for data processing, consider indicators in terms of context and multidimensionality (for example, time as a context indicator, Ibat-Temp relationship), increase the number of metrics for analysis, make the analysis system interactive and more user-friendly for the end user.

GRID 2018, September 2018, DUBNA

clc.ihep.su