MACHINE LEARNING TECHNIQUES FOR DATA CENTER ANOMALIES IDENTIFICATION

PhD. Student, Bogdan DINU, ***Department of Electronic Devices, Circuits and Apparatus, University "Politehnica" of Bucharest, ROMANIA***
PhD. Student, Octavian ARSENE, ***Department of Electronic Devices, Circuits and Apparatus, University "Politehnica" of Bucharest, ROMANIA***

Abstract: *One of the most important tasks within a Data Center is to monitoring computers. The present paper introduces an improved data mining process flow in order to speed it up and to avoid anomalous data to be part of the presentation layer. The collection process is using a parallel approach implemented in a PL/SQL package. The anomaly detection phase is using two different mechanisms, a probabilistic model in Octave implementation and a Support Vector Machine algorithm using Oracle Data Mining product.*

Key words*: clustering, classification, database anomalies, SVM*

1. PROBLEM DESCRIPTION

We know that the global volume of data is estimated for 2012 at Zettabyte's level. Accordingly with an International Data Corporation (IDC) estimation at the end of this year the total amount of space will be 2.7 Zettabyte's that means a rate of growth that is greater then 40%.
In addition to the issues with the amount of data another problem is related to the speed requested to deal with the info's in real time. These aspects includes financial component's, data quality, customer behavior, product monitoring.
Our surround digital world size already has Exabyte's as measure terms and a new language is prepared such as Zettabyte and Yottabyte. The speed and real time access along with the huge size of data became major factors in the data warehouse design, implementation and maintenance. In theory in a data center we have glued together a huge number of server's and hosts each of them having different activities like: file server, middle layer machine, OLTP application dedicated server, DW (data warehouse) environment and the list can be extended. Adding even more complexity to this observation the speed of data generation is comparable with an OLTP environment.
We will add also the observation that the OLTP server applications that feed the databases have a substantial quota from the total number of server's and during the last period all these applications become more complex and can cover vary segment industries. To conclude we have here combination of three important factors: volume, speed and complexity of data. This mix can raise many difficulties in a data center in order to obtain data protection related to the new attack techniques.
Also, from another perspective we want to protect all these hosts against viruses, data hacking activities but in the same time we want to obtain more information about the hardware/software components behavior that work into a real production environment. This collection will deliver important information about the reliability and functionality of all the components. Finally this kind of hierarchy will lead to costs decreasing.
To resume in a data center we can have a huge combination of hosts, (also each of this servers can have different destination). If only a fraction of metrics are collected from this kind of environment we will be able to obtain a volume of data able to feed a huge data warehouse.
The next step is to transform the data to fit into the model selected.

This set of features describes a new kind of environment called Real Time Data Warehousing. Some features come from OLTP (the speed, the model of data) some other come from a DW environment like the amount of data, (also a difference is that we don't have any kind of batch process that typically exists in DW).

At the end we want to obtain finally a set of reports based on the metric collected that will detect and emphasize anomaly related to the good functionality of any of the hosts.

In addition let's say that one of the goals will be to see all this stuff and results on hour mobile phone, iPod or other similar devices. All these devices don't have a strong computational power so the reports must be delivered in a fully computed shape. To complete also the last detail the complexity let's say that a basic collection of data is not good enough we want also to add a complex calculation between some metrics fields. This article is proposed as a solution for this particular combination of problems.

After the problems were described we assume that the correct answer here is to choose some data mining algorithm that will solve the presented issues. However, we do not have a golden rule in order to choose a machine learning algorithm, for a given classification problems. We can take into consideration some particularities of the problems and the final selection will be based on vary performance criteria like: speed of learning with respect to number of attributes, speed of classification, tolerance to noise, model parameter handling, tolerance to missing values.

To deliver more value here one of the aspects is to obtain a better speed at each iteration, for all these reports. Despite the volume of data is big we want also to obtain a fast answer for some set of rows. We assume that we have chosen one of the DM algorithms as the most suitable for a given problem. In general term after if we have made this selection the next step will be to made a data preparation, basically is a pre-processing phase where some set (or subsets) of data are processed to become inputs that will match with those required by the algorithm selected.

The data set is divided in three sets: (1) training ***x(mtraining)***, (2) cross-validation ***x(mcv)*** and (3) test ***x(mtest)*** accordingly to the 60% - 20% - 20% of data set proportions.

One of the methods of Data Mining (DM) is anomaly detection. This method detects patterns in a set of data that does not fit with a normal behavior into the dataset. This is applied in different situations like: share market, fraud detection and intrusion - detection in a network, monitoring machines in a data center and manufacturing.

2. DATA PREPARATION

First observation is related to the situations when data are collected but the set obtained cannot be used directly as a valid source for the data mining algorithm. All these set must be processed taking into account the restrictions for all the inputs and for all the DM techniques used. For example usually some fields cannot be used directly and need to be transformed. In other situations to highlight the potential outliers or anomalous record we need to apply "*winsorize*" or "*trimming*" techniques before to feed the main algorithm. In another vein in order to speed up the collection process we use a form of parallel update that was three times faster than the default approach.

3. ANOMALIES DETECTION

We used two data mining techniques: (1) using a model for anomaly probabilities and (2) Support Vector Machines (SVM-s). A secondary goal of the current study was to compare the two different approaches. The probability based model is implemented in Octave and the latter approach is deployed within Oracle Data Miner (ODM) providing powerful data mining functionality as native SQL functions within the Oracle Database.

3.1 Anomalies probabilities model

More formally in the anomaly detection problem, are given some data sets, ***x(1)*** through ***x(m)*** of ***m*** examples, assuming that these examples are normal (non-anomalous) examples, and the goal is to have an algorithm able to label if a new example ***xtest*** is anomalous. For each feature ***i = 1:::n***, the algorithm ***rst*** step needs to ***nd*** parameters $_i$ and $_i^2$ that ***t*** the data in the ***i***-th dimension $x^{(1)}_i$***; ::: ;*** $x^{(m)}_i$. The Gaussian distribution is given by:

The considered approach is the following, a model for the probability ***P*** of ***xi*** is built based on the unlabeled training set, where ***xi*** (***i = 1:::n***) are the features of the analyzed problem. Having built a model of the probability of ***xi***, if ***p*** of ***xtest*** is less than a threshold value then this test example is considered an anomaly (***y = 0*** if example is normal and ***y = 1*** for anomalous).

Using accuracy as evaluation metric does not show very clear if a classification algorithm is improved in case of skewed classes. The solution of this case is to use two different evaluation metrics: precision (***P***) and recall (***R***) which give a better sense this algorithm is doing:

Precision = TRUE+ / (TRUE+ + FALSE+) (1)

The concepts ***TRUE / FALSE*** positives or negatives are defined as:

1. ***TRUE+*** is the number of true positives: the ground truth label says it is an anomaly and the algorithm correctly classified it as an anomaly;
2. ***FALSE+*** is the number of false positives: the ground truth label says it is not an anomaly, but the algorithm incorrectly classified it as an anomaly;
3. ***FALSE-*** is the number of false negatives: the ground truth label says *t* is an anomaly, but the algorithm incorrectly classified it as not being anomalous.

Recall = TRUE+ / (TRUE+ + FALSE-) (2)

In order to compare the above two metrics and to simplify the decision process is used ***F1 score***:

F1 score = 2 x (P x R) / (P +R) (3)

The algorithm is the following:

4. fit probability model ***p(x)*** on training set $x^{(1)}; ::: ; x^{(mtraining)}$ (where $x^{(i)}$ is from R^n);
5. choose threshold ε on cross validation set $(x^{(1)}_{cv}; y_{cv}^{(1)}); ::: ; (x^{(mcv)}_{cv}; y_{cv}^{(mcv)})$ using ***F_1 score***;
6. Find anomalies on test set $x^{(1)}_{test}; ::: ; x^{(mtest)}_{test}$.

The model has a Gaussian (Normal) distribution for each of the features $_i$.

$$P(x, \mu; \sigma^2) = 1/\sqrt{2\pi\sigma^2} * e^{-(x-\mu)/2\sigma^2} \quad (4)$$

Where μ is the mean and σ^2 controls the variance. The parameters, (μ_i, σ^2_i), estimation of the ***i***-th feature use the following equations:

$$\mu_i = 1/m \sum_{j-1}^{m} x_i^{(j)} \quad (5)$$

$$\sigma^2_i = 1/m \sum_{j-1}^{m} (x_i^{(j)} - \mu_i)^2 \quad (6)$$

The computation is based on a vectorized implementation in Octave for data which is faster than a for-loop over every feature and every training example.

The second steps, selecting the threshold ε, the examples which have a very high or low probability are investigated.

The low probability examples are more likely to be anomalies.

The used way to determine which examples are anomalies is to select a threshold based on the cross validation set using ***F_1 score***. For each example is computed $p(x^{(i)}_{cv})$, if an example *x* has a low probability $p(x) < \varepsilon$, then is considered to be an anomaly. The ***F_1 score*** value tells how well the algorithm is doing on finding the ground truth anomalies given a certain threshold. The maximum value of ε is selected.

3.2 Support Vector Machine

Support Vector Machines (SVM) is a classification and regression analysis algorithm used in machine learning. SVM tries to predict for a given input an output class. In the present paper there are considered binary linear classification (this is the default acceptance but the situation can be more complex if the number of classes is greater than 2 - multiclass). First as training examples a primary set of data using the SVM model will be computed.

So for any input will be calculated in what class will belong the output and after that each new set of data will be mapped into the same space.

The test data will be classified by the computed SVM model.

The output in a SVM is given by the following formula:

$$f_i = b + \sum^m \alpha_j y_j K(xj ; xi) \quad (7)$$

Where f_i is called margin, it is the distance of each point (or data record) to the decision hyper plane defined by $f_i = 0$; α_j is the Lagrange multipliers for the ***j***-th training data record; both x_j and y_j correspond to the target value ***+/- 1***.

The division between the two classes must be in a bi-dimensional situation, a line defined by the largest separation which is the maximized distance between the nearest data point to the line of separation. From another perspective SVM can do a non-linear classification if the kernel function is modified as the model mapping will transform the inputs into a high-dimensional feature space.

This algorithm is embedded into the Oracle Data Miner but is improved in terms of numerical processing and offer many advantages: data security and integrity are maintain during the fully mining process, centralized view of data can be used in many applications, the system is highly available and flexible.

In another vein as general perspective SVM as algorithm: is insensitive to outliers, have a good accuracy from prediction perspective, the speed of classification is very high, also the tolerance for the redundant attributes is good, can be adapted for vary type of data. The SVM algorithm avoids the overtraining and is able to work with a huge number of attributes without to affect the overall performance.

The disadvantages of this algorithm could be found in some situations: if the extreme values of the outliers are very high, data set is not linear; SVM requires having data normalized.

4. RESULTS & CONCLUSIONS

There were collected data representing five important metrics from different servers for the purpose of paper:

7. swap utilization, as metric 1 (M1);
8. central processing unit utilization (CPU), as M2;
9. running queue length, represents the number of active processes in a CPU run queue, as M3;
10. memory utilization, as M4;
11. free memory, as M5.

These metrics were collected by the enterprise manager (EM) application module of the database server. The data set has 1610406 records. Thus the data matrix has 6 columns, host name and five metrics (metric M_i index represents the column in the matrix) and 1610406 rows; each row represents data for a specific database server (host).

After both algorithms were run against the same set of data we saw that the results converge in a high proportion (about 90%). The record detected as anomalous have the following technical explanation: (after data was processed and we have reported the results):

a) CPU utilization by definition is a critical threshold value, when the utilization exceeds 97% value it is an indicator of potential problems. If the Run Queue length indicator has a high value, over 85%, this might represent a critical state of the host. Too many active processing waiting in the CPU queue and the processor is heavily loaded, (potential problems applications without resources or other potential data protection issues etc).
b) Run Queue length exceeding 100%, this metric by definition represents the average number of processes in running queue, it emphasizes the following anomalous situation, a number of processes use the CPU time in an excessive manner and a part of them must be killed.
c) The default warning threshold value is 10% and the critical is 20%. When this value is bypassed with 100% and values greater than 40 are collected it means that the server encounters problems in order to deliver normal functionalities;
d) A high value for both Swap utilization and Free memory can be an intensive data processing (e.g. potential virus attack, an application failure);
e) Both metrics Run Queue length and Memory utilization must be synchronized. A high memory metric value should be related with a positive Run Queue Length value. This means that a significant number of processes need to wait in order to finish their tasks (again potential problems with the concurrent application or other issues related);

f)A high value for Swap utilization and a small positive value for free memory can represent a hardware failure.

A record is considered an anomaly when both methods recommend it. This cross validation assures that some hosts must be analyzed by the system engineers.

To conclude there are 2 main benefits when we used this 2 DM algorithm: one related to the anomaly detection (used to increase data protection) and the second one that is related to how to cut some production costs. Here the idea will be to add all the information into a global report that will count the problem against all the devices that work into a data center.

REFERENCES

[1] **ORACLE,** (2013) *Oracle performance guide*, http://cdn.ttgtmedia.com/searchSystemsChannel/downloads/Oracle_Performance_Survival_Guide_ch13.pdf ;

[2] **Eaton, J., Bateman, D. and Hauberg, S.** (2008) *Gnu Octave Manual*, Network Theory Limited;

3] **ORACLE**, (2013) *Oracle data miner*, http://www.oracle.com/technetwork/database/options/advanced-analytics/odm/index.html ;

[4] **Cortes, C. and Vapnik, V.** (1995) *Support-vector networks*, Machine Learning, pag. 273-297;

[5] **ORACLE**, (2013) *Oracle database administrator's guide*, http://docs.oracle.com/cd/E11882_01/server.112/e25494/toc.htm ;

[6] **ORACLE**, (2013) *Oracle database high availability*, http://docs.oracle.com/cd/E11882_01/server.112/e10803/toc.htm ;

[7] **ORACLE,** (2013) *Oracle database performance tuning*, http://docs.oracle.com/cd/E11882_01/server.112/e41573/toc.htm ;

[8] https://www.statsoft.com/textbook/support-vector-machines

[9] **Ralph Kimball and Margy Ross**, *The data warehouse toolkit: the complete guide to dimensional modeling*, The Data Warehouse Toolkit, Second Edition**,** ISBN 0-471-20024-7

[10] **Claudia Imhoff, Nicholas Galemmo and Jonathan G. Geiger**, (2003) *Mastering Data Warehouse Design Relational and Dimensional techniques*, ISBN: 0-471-32421-3