SEMI-SUPERVISED NEURAL NET APPROACH TO FORECAST UNPREDICTABLE CPU LOAD IN AN ENTERPRISE APPLICATIONS ENVIRONMENT

Nitin Khosla[1] and Dharmendra Sharma[2]

[1]Assistant Director - Performance Engineering, Ictcapm,
Department of Home Affairs, Canberra, Australia
[2]Professor - Computer Science, University of Canberra, Australia

ABSTRACT

The aim of semi-supervised learning approach in this paper is to improve the supervised classifiers to investigate a model for forecasting unpredictable load on system and to predict CPU utilization in a big enterprise applications environment. This model forecasts the likelihood of a burst in web traffic to the IT systems and predicts the CPU utilization under stress conditions. The enterprise IT infrastructure consists of many enterprise applications running in a real time system. Load features are extracted while analyzing the patterns of work- load demand which are hidden in the transactional data of applications. This approach generates synthetic workload patterns, execute use-case scenarios in the test environment and use our model to predict the excessive utilization of the CPU behavior under peak load and stress conditions for the validation purpose. Expectation Maximization method with co-learning, attempts to extract and analyze the parameters that maximize the likelihood of the model after subsiding the unknown labels. As a result of this approach, likelihood of excessive CPU utilization can be predicted in few hours as compared to few days. Workload profiling and prediction has enormous potential to optimize the usages of IT resources with low risk.

KEYWORDS

Semi-supervised learning, Performance Load and stress testing, Co- learning, Machine learning applications.

1. INTRODUCTION

In the dynamic cloud based environment, web traffic or number of hits to an application increases exponentially in a short span of time (burst in traffic) and it drastically slows down the enterprise application system and on many occasions the system crashes as it cannot sustain the excessive load under stress conditions. Some crucial applications providing services to the public, e.g. benefits payments, custom clearances at airports, etc., halt suddenly. At many occasions the systems crash randomly due to unpredictable load because of excessive web traffic and this results in loss of efficiency and productivity. Many at times, the system crashes without any alerts and practically it is not feasible to take any remedial actions e.g. load balancing, etc. The high transaction-rate (excessive number of hits / second) at some moment of time or for a very short duration can drag the applications or computer systems to be very sluggish as the system becomes irresponsive and unable to process large number of transactions requests simultaneously at the servers.

Natarajan Meghanathan et al. (Eds) : ACITY, AIAA, DPPR, CNDC, WIMNET, WEST, ICSS - 2019
pp. 89-95, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91709

Our reliance on cloud internet computing is increasing every day and it has become unavoidable. Now it has become utmost important for big enterprises to keep the important applications running 24/7 to acceptable efficiency levels during the whole year. Managing large number of applications, running at a high efficiency level in the big enterprise system and developing new functionality / applications at the same time is a constant challenge between functionality and resource management [8]. Lot of time it is observed that the system has arrived to a situation when practically very little memory is available for the critical applications to run in an enterprise set-up and it can lead to a system crash. It becomes even more complex when transactions are generated in wide area distributed networks where the network traffic, latency and bandwidth are key factors impacting the performance of applications.

The primary aim of this research paper is to design and implement a practical approach to predict the unpredictable burst in traffic, using semi-supervised neural nets, by analyzing the work-load patterns hidden in of the key transactions and to observe CPU utilization under stress conditions (high volume of web traffic) using data mining techniques.

2. RESEARCH PROBLEM

In this research work, the load profiles were studied to identify patterns at different time periods during last one year. These patterns are used to develop test scenarios in the test environment. We also analyzed the big data and extracted load patterns from the raw transactional data [2]. This enabled us to identify issues related to load estimation in testing and the real world (production) environments and to develop a performance predictive model to forecast the CPU performance of the IT infrastructure under extreme stress conditions.

2.1. Performance Issues in the Current Practices

Most of the computer applications are developed upon business specifications and these specifications are primarily depend upon the user requirements [1]. We have noticed in our department that there are some critical limitations in measuring the performance of applications, in the current practices such as

- Not Reliable: System behaviour predicting e.g. response time, performance, under a short burst traffic (hits) situation is not reliable
- Not Robust: Lack of a practical robust approach due to the volatile and un- predictable web traffic
- Risk Based Approach: Performance testing (load and stress testing) is mainly done on key transaction or high-risk use cases only in complex environments as testing for each and every scenario is extremely time consuming and costly [2]. So, load tests are generally performed on -
  - Key transactions with critical impact
  - Critical functions which could impact people or important services

3. LARGE ENTERPRISE ENVIRONMENT

Enterprise environment of big public departments or corporates comprises of different types of system architectures (latest and legacy) *e.g.* mobile applications, cloud computing, etc. [1]. We have performed our experiments in a large and complex environment with more than 350 servers and large number of which were distributed across multiple sites (countries). This test environment (called as “pre-production” environment) is a subset of the whole enterprise set-up with all applications of the most recent releases but with limited data set. This test environment also represents a system simulating all the applications working in more than 52 overseas posts across the world.

4. DATA ANALYSIS AND FEATURE EXTRACTION

Raw data is captured from data logs / files on periodic intervals. To perform the data collection, profile points are configured in the IT system which collects the data continuously on pre-defined time intervals [1]. Different type of transactional data is captured e.g. CPU utilization, response times, bandwidth used, memory usages, etc. and used for analysis and debugging purposes.

Load and validation experiments are done in the IT test environment (called as pre- production environment) which represents the production environment. This test environment emulates the real-world type of scenarios or behavior. Profile points are used to monitor the transactions, responses times and other key parameters during the full path both ways (server to client and client to server). The analysis of data, recognition of patterns are used and extremely important for optimization [1]. This also helps to continuously improve the models to predict reaching critical load while meeting the dynamic needs and variability of the dynamic load patterns [12].

5. IDENTIFYING WORKLOAD PATTERNS

Workload patterns are quite typical and different from normal behavior of a CPU under stress load conditions [1]. Signs of high CPU utilization can be predicted while simulating the virtual traffic in test environment. The test environment also executes large number of applications as like real word scenario.

We have developed a profile trace-based approach to identify patterns of the CPU utilization of servers over a period. We have captured transactions and relevant data for the last one year with the help of profile data points. These profile capture points were configured at different threads and nodes of the applications path flows and then we have studied the patterns and respective behavior.

Assumption: it is assumed that CPU usages follow a cyclic behavior for some types of transactions, and it follows a periodic pattern. These patterns can be represented by a time series consisting of a pattern and/or a cyclical component [1][2].

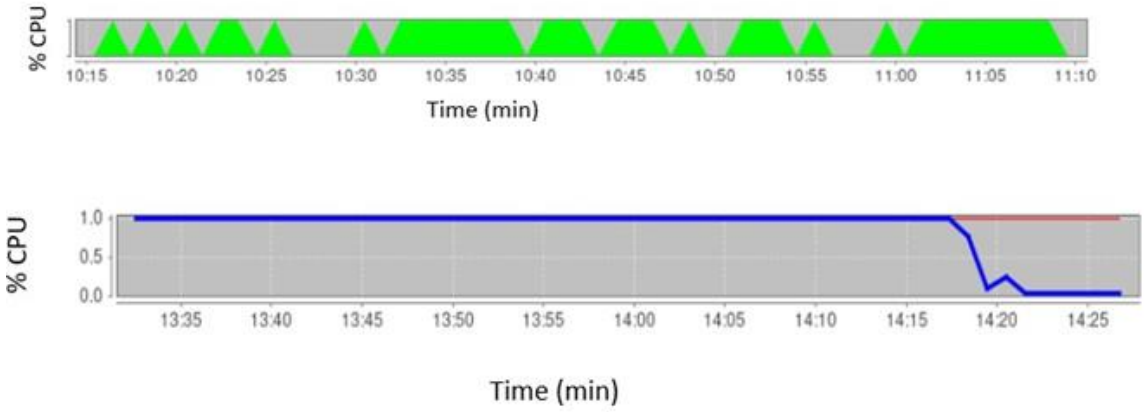

Figure 1. Different Load pattern of CPU Utilization, approx. 3000 – 4000 hits per minute.

Figure 1 shows load patterns which follow a cyclic sequence. In the second the % CPU suddenly drops from high usages (95%) to about 12% average. When it was about 95%, the transaction response times were very high and the system was showing sluggishness during the peak spikes. Data mining is done to capture two peak intervals where there a pattern, reaching a higher CPU utilization for a longer duration of time, this clearly shows abnormal behaviour of CPU utilization. We have also collected some data and did analytics on other parameters e.g.

hard disk usages, database hits, etc. and these can also provide insights from these patterns for predictive modelling. In this paper, it is out of scope and will be investigated as an extension to current work.

6. SEMI SUPERVISED LEARNING (SSL) FOR TRAINING

We have used a semi-supervised learning method to train our model. In this approach we use labelled data with some amount of unlabeled data. This is used in conjunction with a small amount of data can produce considerable improvement in learning accuracy over unsupervised learning [5]. Some of the advantages, in context to this research work, are –

a) A scalable probabilistic approach
b) Can generate a model which can simulate analogies of patterns on different profile data sets in a complex enterprise applications environment
c) Can achieve optimisation in terms of time and accuracy by predicting results

Considering some assumptions for the semi-supervised learning to work e.g. if two distinct points d_1, d_2 are close enough, then there might be respective outputs b_1, b_2. If we do not consider these assumptions, it would be hard to develop a practical model for a known number of training data sets to predict a set of infinitely possible test-cases which are mainly unseen [16]. We also have used other parameters in using labelled

data points such as - effort, time, tools and resources. Based upon the nature and potential implementation of this research, semi-supervised learning with forced-training [3][7] may provide some useful outcomes as it is based upon

i) Learning (through training) of data set with both labelled and unlabelled data
ii)Results are obtained in less time
ii) Assumptions of forced-training can reduce the training time

7. EXPERIMENT AND VALIDATION PROCESS

In our experiment and validation process, we simulate the load pattern showing burst in traffic in complex enterprise test environment as we observed after collecting the data. This represents the real-world scenario type patterns respective to different trans- actions. The test environment has a sub-set of data which proportionally represents large data sets associated with the integrated applications. More than 129 live applications fully functional as the real applications environment. The process included –

i) Data extraction and analysis of the of workload demand patterns over a long period of time – during last one year,
ii) Data collection using profile points and analysis. Data logs were created and extracted for a very short intervals of 5 minutes,
iii) Generate synthetic workloads patterns,
iv) Run stress tests in test environment with large number of virtual users using system applications as in real world scenario,
v) Validate the results by extracting data from different profile points of the application threads and nodes.
vi) Training the model using semi-supervised learning approach (deep learning paradigm) [7][11],
vi) Forecast the likelihood of the traffic burst (excessive CPU usages) using the trained model [4][6].

8. EXPERIMENT SET-UP

We designed and configured the following experiment set-up to perform our experiments.

i) Virtual User Generator: used to simulate key end-user business processes and transactions
ii) Controller: to manage, control and monitors the execution of load tests
iii) Load Generators: to generate virtual load and simulate work-load patterns while large number of virtual users generating web-traffic and exhibiting load patterns simulating web-traffic bursts.

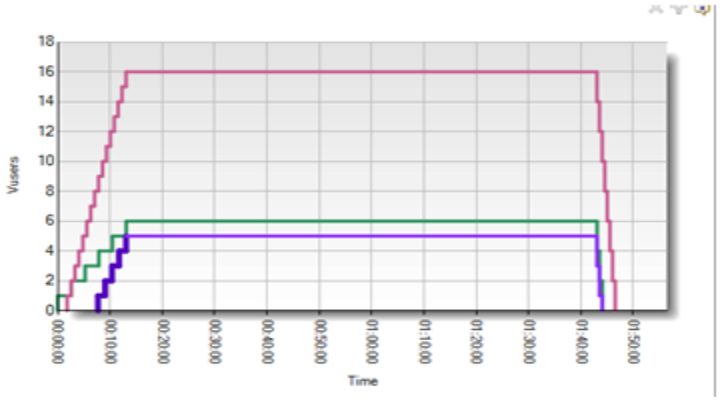

Figure 2. Load profile with virtual users

Figure 2 shows user work-load profile (stress conditions) with different ramp up times. This set up is used for validation of our results.

9. FORECASTING TRENDS

To study and analyse a trend in the load patterns we have worked out the aggregate demand difference of each occurrence of the pattern from the original workload [15]. We used a modified ETS (exponential smoothing) algorithm with ETS point predicts are equal to the medians of the predict distributions.

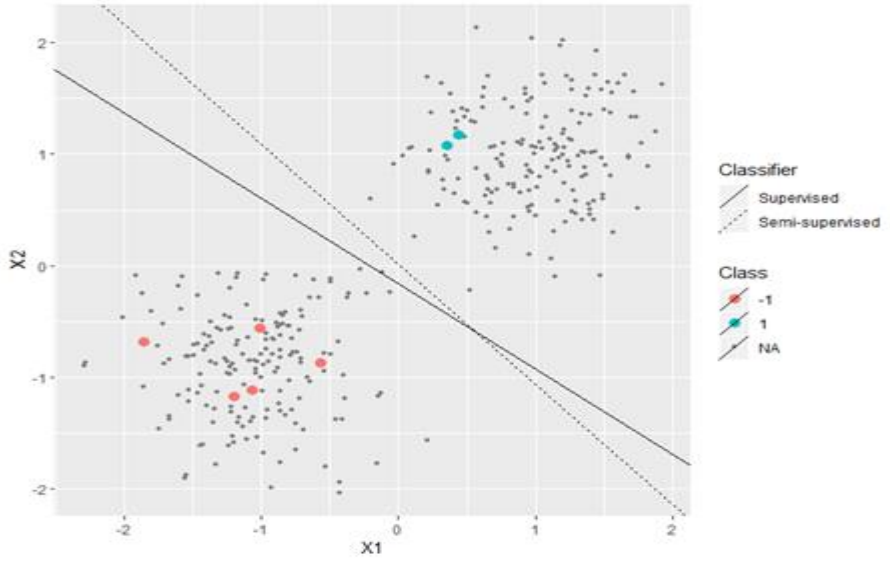

Figure 3 (a) Semi-supervised v/s Supervised learning (1year data set)

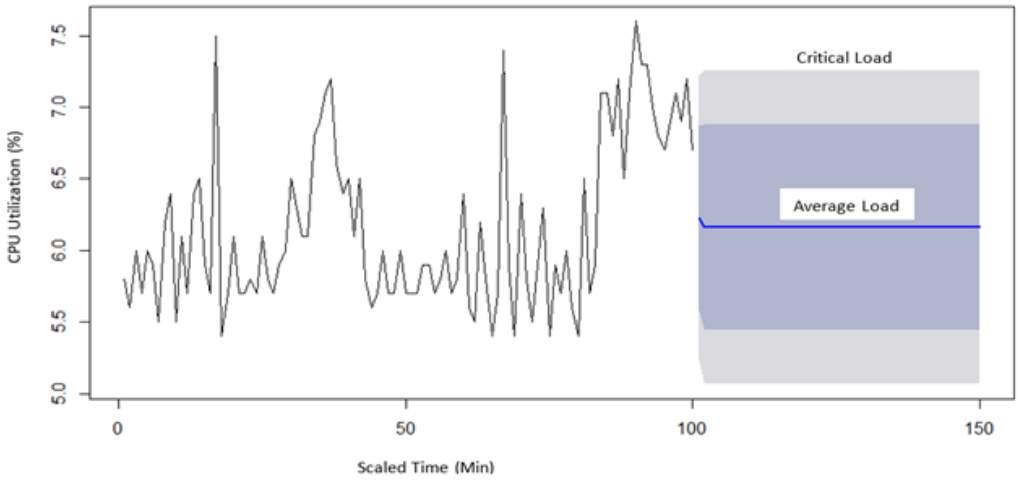

Figure 3(b) CPU Utilization under burst of traffic load conditions

Figure 3(a) shows the results of a modified semi-supervised neural network model, which is used to predict burst of traffic [9][10]. This model is now a part of our monitoring process in continuous evaluation of the demand patterns, as shown in Figure 3(b). This model predicts the burst of traffic behaviour and sets alarm for the system architects to take remedial actions e.g. re-allocation of IT resources to avoid a system crash or failure.

10. CONCLUSION

We have designed a novel practical approach to predict burst in traffic behavior in a complex and highly integrated environment (test or pre-production) where more than 130 IT applications were live and thousands virtual users generate user-load under stress conditions. Our integrated enterprise environment had a distributed system with more than 300 servers serving more than 440 clients simultaneously. Using semi-supervised neural network, the proposed approach predicts and identifies the burst in traffic in a complex enterprise IT infrastructure. Data analytics enabled the system architects and system capacity planners to distribute the work-load appropriately.

The proposed practical approach helped the IT architects to mitigate the risk of an un- expected failure of the IT systems, due to burst of traffic patterns, within a very short duration of time (3 to 4 hours) compared to 1 - 2 weeks as in the current practice. Validation of our results were done in an integrated test environment where alerts are activated as soon as the collective CPU utilization of the server's crosses 70% threshold critical limit. Experiments performed in test environment validated that our approach to predict potential burst of traffic worked effectively. In addition, we have found that this approach has benefited our department in efficient management of IT resources and helped to plan IT capacity for future demand predictions. This resulted in saving cost due to the optimum resource allocation in our IT enterprise IT environment.

As further work, we are working on investigating the impact of different parameters e.g. hard -disk failures, network latency [14], different types of transactions [13] and trying to develop a hierarchical semi-supervised learning model to extract patterns and to design an accelerated semi-supervised learning for predictive modelling.

REFERENCES

1. Daniel Gmach, Jerry Rolia, Ludmila Cherkasova, Alfons Kemper, (2007) "Workload Analysis and Demand Prediction of Enterprise Data Center Applications", IEEE 10th International Symposium on Workload Characterization, Boston, USA.

2. Jia Li, Andrew W. Moore, (2008) "Forecasting Web Page Views: Methods and Observations", Journal of Machine Learning Research.

3. Adams, R. P. and Ghahramani, Z. (2009) "Archipelago: nonparametric Bayesian semi-supervised learning", In Proceedings of the International Conference on Machine Learning (ICML).

4. H. Zhao, N. Ansari, (2012) "Wavelet Transform Based Network Traffic Prediction: A Fast Online Approach", Journal of Computing and Information Technology, 20(1).

5. Yuzong Liu, Katrin Krichhoff, (2013), "Graph Based Semi-supervised Learning for Phone and Segment Classification", France.

6. Danilo J Rezende, Shakir Mohamed, Daan Wierstra, (2014) "Stochastic Backpropagation and Approximate Inference in Deep Generative Models", Proceedings of the 31st International Conference on Machine Learning, Beijing, China.

7. Diederik P. Kingma, Danilo J Rezende, Shakir Mohamad, Max Welling, (2014) "Semi supervised Learning with Deep Generative Models", Proceedings of Neural Information Processing Systems (NIPS), Cornell University, USA.

8. Pitelis, N., Russell, C., and Agapito, L. (2014) "Semi-supervised learning using an unsupervised atlas". In Proceedings of the European Conference on Machine Learning (ECML), volume LNCS 8725, pages 565 –580.

9. Kingma Diederik, Rezende Danilo, Mohamed Shakir, Welling M, (2014) "Semi-supervised Learning with Deep Generative Models", Proceedings of Neural Information Processing Systems (NIPS).

10. L. Nie, D. Jiang, S. Yu, H. Song, (2017) "Network Traffic Prediction Based on Deep Belief Network in Wireless Mesh Backbone Networks", IEEE Wireless Communication and Net- working Conference, USA.

11. Chao Yu, Dongxu Wang, Tianpei Yang, et., (2018) "Adaptive Shaping Reinforcement Learning Agents vis Human Reward", PRICAI Proceedings Part-1, Springer.

12. Xishun Wang, Minjie Zhang, Fenghui Ren, (2018) "Deep RSD: A Deep Regression Method for Sequential Data", PRICAI Proceedings Part-1, Springer.

13. Avital Oliver, Augustus Odena, Colin Raffel, Ekin D Cubuk, et. (2018) "Realistic Evaluation of Semi-supervised Learning Algorithms", 6th International Conference on Learning Representations, ICLR, Vancouver, BC, Canada.

14. Kenndy John, Satran Michael, (2018) "Preventing memory leaks in Windows Applications", Microsoft Windows Documents.

15. M.F. Iqbal, M.Z. Zahid, D. Habib, K. John, (2019) "Efficient Prediction of Network Traffic for Real Time Applications", Journal of Computer Networks and Communications.

16. Verma. V, Lamb. A, Kannala. J, Bengio. Y, Paz DL, (2019) "Interpolation Consistency Training for Semi Supervised Learning", Proceedings of 28th International Joint Conference on Artificial Intelligence IJCAI Macao, China.