International Journal of Mobile Communication & Networking.
ISSN 2231-1203 Volume 11, Number 1 (2020), pp. 1-11

http://www.ripublication.com

Study of Packet Loss Prediction using Machine Learning

Dr. Kalpana Saha (Roy) [1], Tune Ghosh [2]
Computer Science & Engineering Department
Govt. College of. Engg. & Ceramic Technology
Kolkata, West Bengal, India

Abstract

The importance of providing guaranteed Quality of Service (QoS) cannot be overemphasised in the Next Generation Network (NGN) environment. NGN supports converged services on a common IP transport network. Call Admission Control (CAC) mechanism provides QoS to class-based services in a proactive manner. We use Machine Learning (ML) techniques for providing autonomous CAC due to the factors of complexity, scale and dynamicity of NGN. Packet loss prediction is one of the metrics for provisioning QoS. This paper is an effort to measure the packet loss prediction using machine learning approach. Two ML based model are used to predict packet loss, Decision Tree model and Logistics Regression model. Performance measure is based on experiments and observations. The outcome of the comparative study states that Decision Tree model gives better result compared to the Logistics Regression model for prediction of packet loss.

Keyword: QoS, NGN,CAC, Decision Tree, Logistics Regression, prediction, packet loss, machine learning

I. INTRODUCTION

QoS is a prime concern for both the service providers and subscribers. The problems of guaranteed QoS arise due to the advances in the network architectures, the demand for multimedia services and applications. IP-based NGN promises guaranteed QoS [1][4][5]. CAC is a major preventive technique to provide guaranteed QoS to various class-based services as recommended by the ITU-T for NGN [2]. Guaranteed QoS was first witnessed in the ATM networks supporting class-based services [3]. CAC

mechanism plays a proactive role in providing QoS by limiting the entry of traffic at the edges of the network. Its job is to make a decision to admit or deny a new call into the network based on the condition that the QoS of the existing calls and also the new calls are satisfied. CAC approach becomes difficult and intractable to solve through conventional analytical methods due to growing of number of services, their classes and size of the network [6][7]. We use Machine Learning (ML) approach to solve traditional CAC approach. ML helps the system behavior through the process of learning which is based on observation of performance data over a period of times [8]. Once appropriately trained, they are able to estimate and predict future system behavior and subsequently make admission decisions with high accuracy and speed. ML approaches are applied in the telecommunications domain to solve various network management problems [9]. Neural network (NN) offers approach learning for traffic and service quality [10]. Combination of Particle Swarm Optimization and Fuzzy logic for next generation mobile communication networks provide better CAC scheme [12]. Support Vector Machine (SVM) based CAC algorithm utilizes service vector and network vector to predict admission state for admission decisions [13]. This scheme accelerates calculation speed with lower call delay. Lower call blocking probability and call dropping probability is also achieved here. Bayesian network (BN) based CAC framework implements delay prediction based on call admission decisions [14]. There occurs a comparison between NN and BN for response time modeling in service-oriented systems [15]. Packet loss prediction is one of the metrics for provisioning QoS. Our paper is an effort to measure the packet loss prediction using machine learning. Two ML based model are used to predict packet loss, Decision Tree model and Logistics Regression model. Performance measure is based on experiments and observations. The outcome of the comparative study provides some interesting insights into the behaviour of Decision Tree model and Logistics Regression model for prediction of packet loss. The rest of this paper is structured as follows. In section II we present the details of machine learning model. Section III provides software used with variables. Simulated results with graphical representations are provided in section IV. Section V concludes the paper by suggesting possible future works.

II. MACHINE LEARNING MODEL

CAC system generally resides on an edge router, whose function is to allow controlled traffic into the core network. A generic CAC framework based on the ML approach is shown in figure 1. The input to such a system consists of customer call requests and the output is a decision to either admit or deny the call. The call request consists of traffic descriptors and desired QoS in the network. Traffic descriptors include parameters like peak rate, average rate, maximum burst duration and type of application which are supplied by the caller. QoS requirements include some measure of metrics like packet loss, average delay or delay variation (jitter). Available link bandwidth and buffer occupancy are also inputs to CAC. Based on the choice of inputs and outputs, the ML module is trained offline with a set of data which is observed in the system over a period of times. The training data set consists of cases

where both the inputs and outputs are known. However, when the trained model is in the online mode, it provides the estimate of the output for a particular input combination. It is clear that the overall CAC performance is dependent on the prediction accuracy of the model. Prediction accuracy depends on how well the model estimates the unknown output when presented with unseen cases not present in the training set. In our paper, we use two ML based model to predict packet loss, Decision Tree model and Logistics Regression model. A decision tree is a decision support tool which provides decisions and their possible consequences. Logistic regression is the appropriate regression analysis to conduct when the dependent variable is binary. The logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variable.

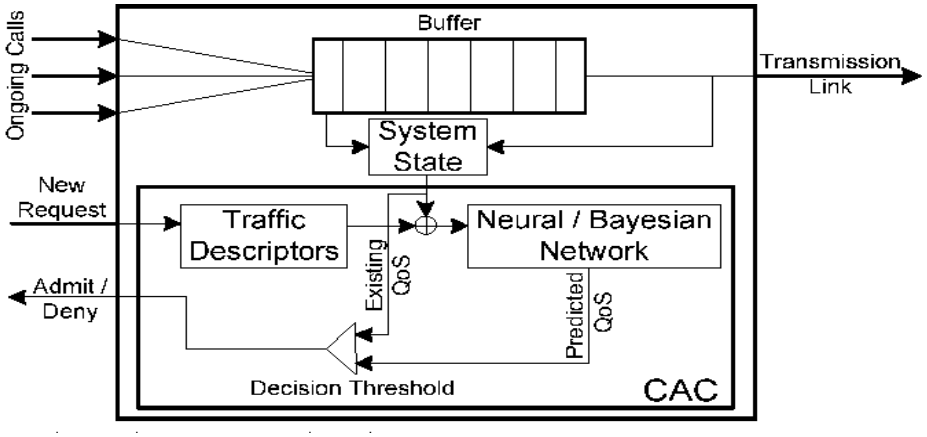


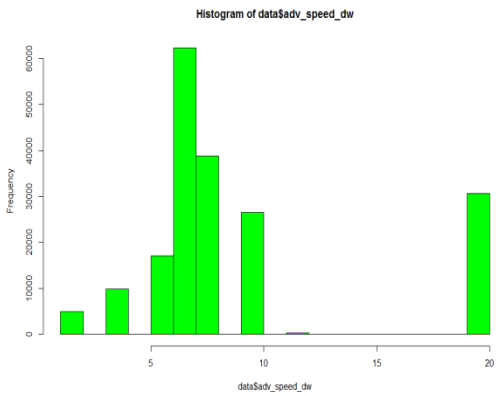
Figure 1: Machine Learning based CAC

III. SOFTWARE USED WITH VARIABLES

Here, the environment is R Studio. We use R language. We collect the data set from internet. We get total number of observations 190166 from 12 variables. Continuous variables are ad_speed_dw, adv_speed_up, latency, jitter, downstream_throughput, upstream_throughput. Categorical variables are packet_loss, technology, isp, id, postcode, timestamp.

IV. SIMULATED RESULTS WITH GRAPHICAL REPRESENTATIONS

It is clear from the figure 2, figure 3 and figure 4 that the medians of "adv_speed_dw" in two categories of "packet_loss" are significantly different. It means the variable "adv_speed_dw" has a role in deciding "packet_loss".



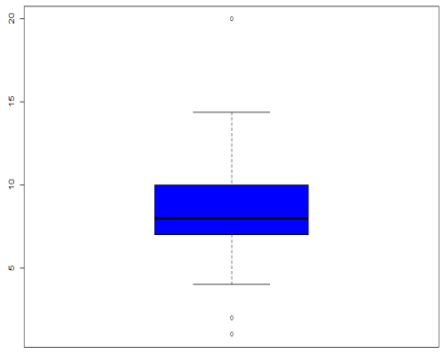


Fig 2. hist(data$adv_speed_dw, col='green') ***Fig 3. boxplot(data$adv_speed_dw, col = "blue")***

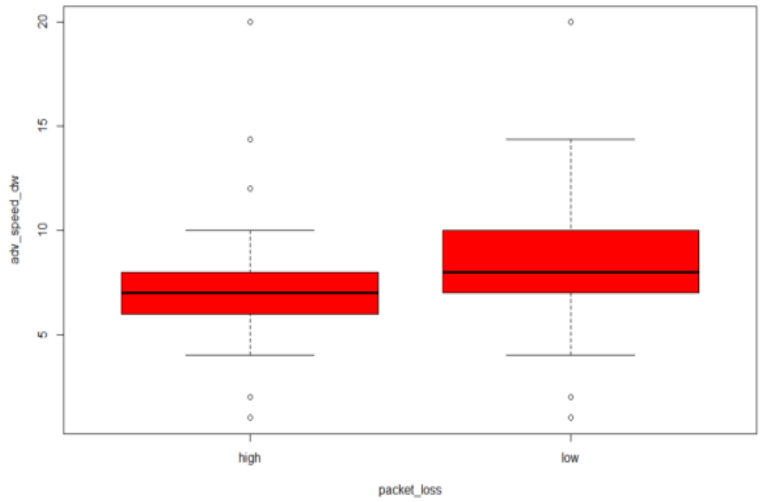


Fig 4. boxplot(adv_speed_dw~packet_loss, data = data, col= "red")

It is clear from the figure 5, figure 6 and figure 7 that the medians of "adv_speed_up" in two categories of "packet_loss" are not different. It means the variable "adv_speed_up" has no role in deciding "packet_loss".

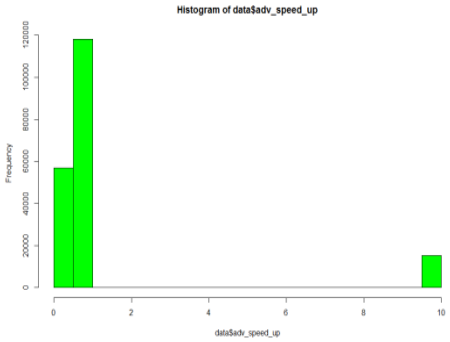


Fig 5. hist(data$adv_speed_up, col='green')

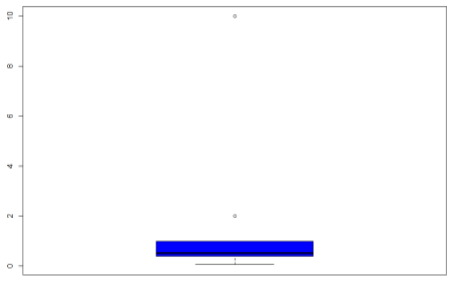


Fig 6. boxplot(data$adv_speed_up, col = "blue")

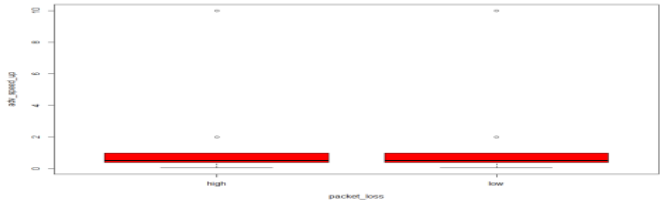


Fig 7. boxplot(adv_speed_up~packet_loss, data = data, col="red"

For the variable latency, it is difficult to visualize difference in median in two categories due to presence of extreme values which are shown in figure 8, figure 9 and figure 10.

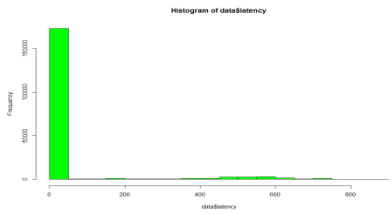


Fig 8. hist(data$latency, col='green')

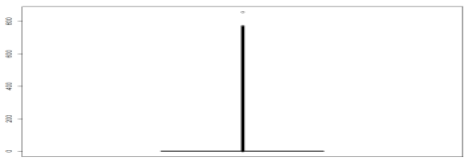


Fig 9.boxplot(data$latency, col = "blue")

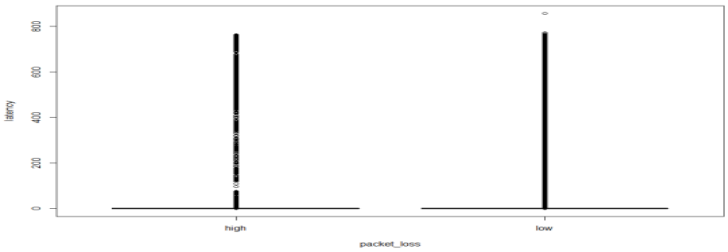


Fig 10. boxplot(latency~packet_loss, data = data, col= "blue")

For the variable jitter, it is difficult to visualize difference in median in two categories due to presence of extreme values which are shown in figure 11, figure 12 and figure 13. We observe that variable 'jitter' has a significance value more than 0.05. So we can confirm that this variable does not have any effect in deciding the 'packet_loss'.

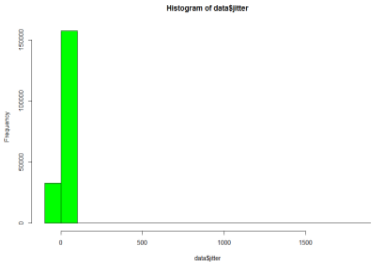


Fig 11.hist(data$jitter, col='green')

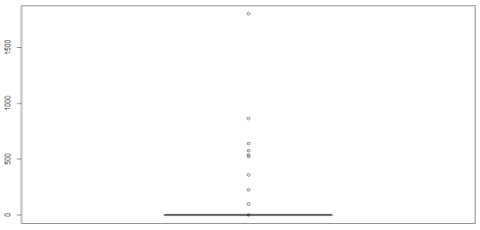


Fig12. boxplot(data$jitter,col="blue")

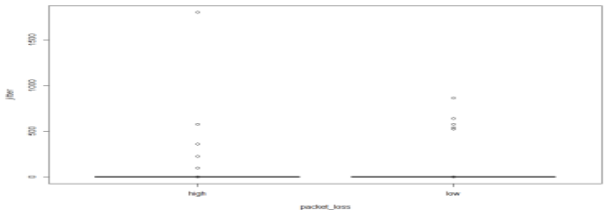


Fig 13. boxplot(jitter~packet_loss, data = data, col= "blue")

It is clear from the figure 14, figure 15 and figure 16 that the medians of "downstream_throughput" in two categories of "packet_loss" are significantly different. It means the variable "downstream_throughput" has a role in deciding "packet_loss".

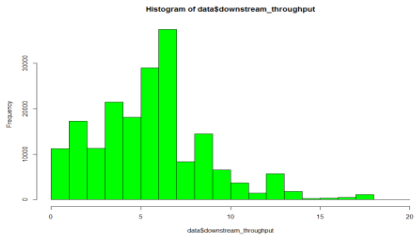


Fig 14. hist(data$downstream_throughput col='green')

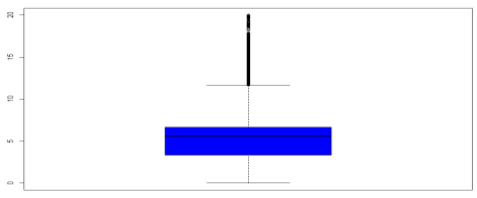


Fig 15.boxplot(data$downstream_throughput, col = "blue")

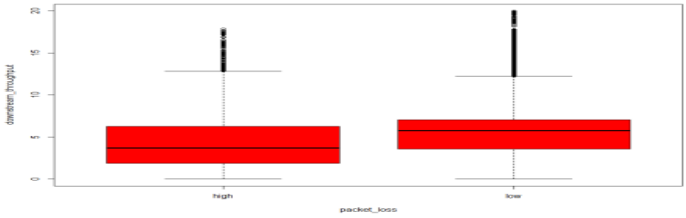


Fig 16.boxplot(downstream_throughput~packet_loss,data = data, col= "red")

It is clear from the figure 17, figure 18 and figure 19 that the medians of "upstream_throughput" in two categories of "packet_loss" are slightly different. It means the variable "upstream_throughput" has some role in deciding "packet_loss

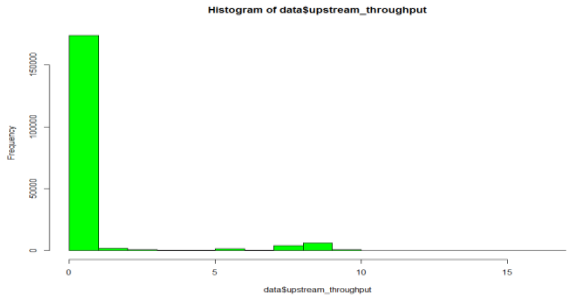


Fig 17. hist(data$ upstream _throughput col='green')

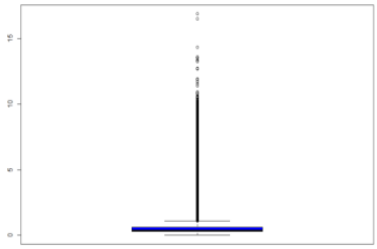


Fig 18.boxplot(data$ upstream _throughput, col = "blue")

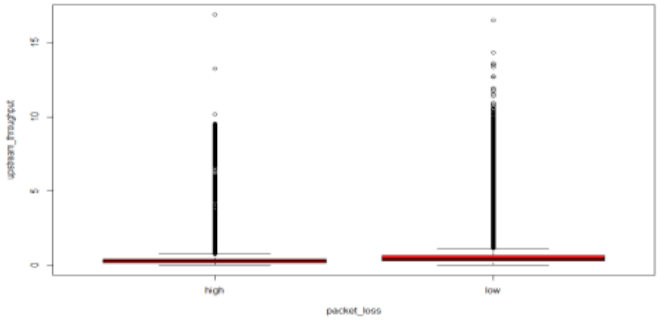


Fig 19. boxplot(upstream_throughput~packet_loss, data = data, col= "red")

Categorical variables packet_loss is shown in figure 20.

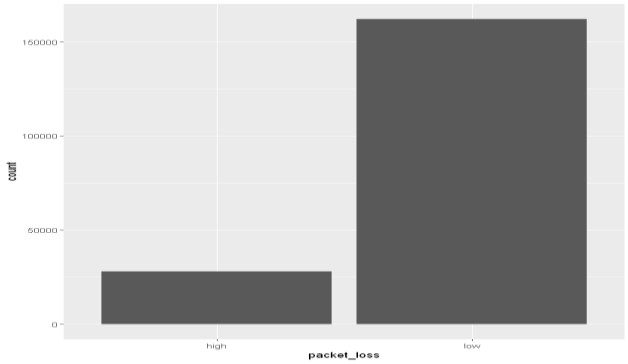


Fig 20.variable "packet_loss" ggplot(data, aes(x = packet_loss)) + geom_bar()

We consider the variable "technology" having highest number of occurrence among all. They are depicted in figure 21 and figure 22.

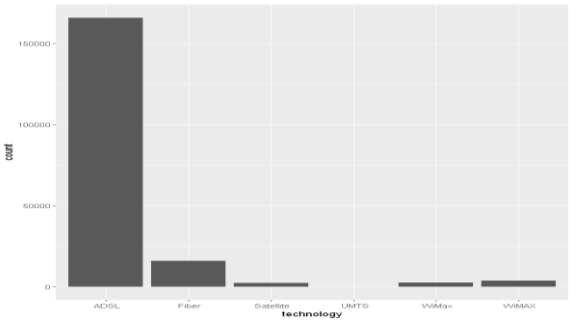


Fig 21. ggplot(data, aes(x = technology)) + geom_bar()

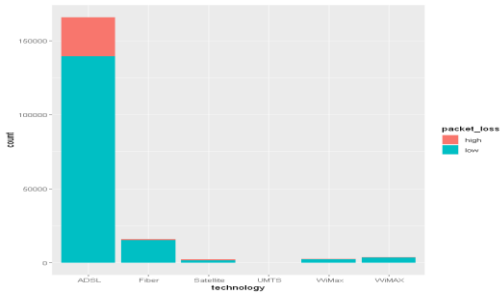


Fig 22. ggplot(data, aes(x = technology, fill = packet_loss)) + geom_bar()

We consider the variable "isp" having highest number of occurrence among all. They are depicted in figure 23 and figure 24.

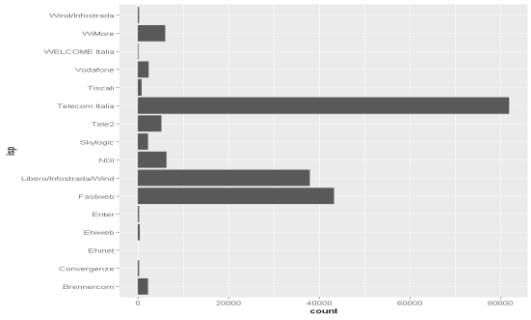


Fig 23. ggplot(data, aes(x = isp)) + geom_bar()+coord_flip()

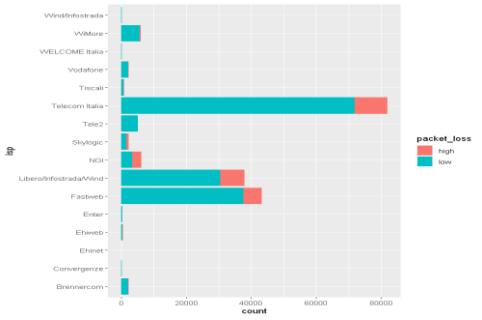


Fig 24. ggplot(data, aes(x = isp, fill = packet_loss)) + geom_bar()+coord_flip()

We drop 3 categorical variables "postcode", "timestamp" and "id" because postcode have missing values and timestamp have not appropriate format for taking data. id are not important variable for measuring packet loss.

We split the dataset in ratio 80:20 as train and test sets and check missing values. We get from observations over a period of times that there is no missing value. These results are obtained using the process of 10-fold cross validation, where the collected data is partitioned into training and test sets and the prediction accuracy averaged over 10 such iterations. Table 1 shows the parameters and their values. Table 2 shows the observations from prediction of LR model and table 3 shows the observation from Decision Trees model. We observe that LR model gives 85% accuracy on packet loss prediction and mis-classification error is 0.15. We also observe that DT model gives 89 % accuracy on packet loss prediction.

Table-1: Parameters and their values

Parameter	Value
Flow generation rate (sec)	5
Average duration (sec)	2.0
Packet generation rate (packets/sec)	Exponential (4)
Packet size (bits)	Exponential (1024)
Type of service	Best Effort

Table 2: Observations from prediction of LR model

Observations get from prediction of LR model		Confusion matrix	
Total dataset dimensions	190166 9	high	291 5264
Training dataset dimensions	152346 9	low	316 31949
Test dataset dimensions	37820 9		
High	0. 7784285		
Low	0.8653945		

Table 3: Observation from Decision Trees model

Observations get from Decision Trees model	
High	3166 1512
Low	2389 30753
Output: 0.8968535	

V. CONCLUSIONS WITH FUTURE WORKS

In our paper we go through all the experiments and observations. We conclude that Decision Tree model gives 89% accuracy on packet loss prediction and Logistic Regression (LR) model gives 85% accuracy on packet loss prediction. So it is clear from comparative study that Decision Tree model gives better result compared to the Logistics Regression model. In future works we will try to get 100% accuracy on packet loss prediction using Machine Learning.

REFERENCES

[1] General overview of NGN, ITU-T Recommendation Y.2001, Dec 2004.

[2] Resource and admission control functions in next generation networks, ITU-T Recommendation Y.2111, Nov 2008.

[3] A. Hiramatsu, "ATM communications network control by neural networks", IEEE Transactions on Neural Networks, vol. 1, no. 1, pp. 122-130, March 1990.

[4] Chen, X., Wang, C., Xuan, D., Li, Z., Min, Y., Zhao, W. "Survey on QoS Management of VoIP", In Proc. of the 2003 International Conference on Computer Networks and Mobile Computing, IEEE Computer Society, Los Alamitos (2003).

[5] Wang, S., Mai, Z., Xuan, D., Zhao, "Design and implementation of QoS-provisioning system for voice over IP", IEEE Transactions on Parallel and Distributed Systems 17(3), 276–288 (2006).

[6] H. G. Perros, K. M. Elsayed, "Call admission control schemes", IEEE Communications Magazine, pp. 82-91, Nov 1996.

[7] Yu, J., Al-Ajarmeh. I, "Call Admission Control and Traffic Engineering of VoIP", in Second International Conference on Digital Telecommunications, IEEE ICDT, 2007.

[8] E. Alpaydin, Introduction to Machine Learning, MIT Press, 2004.

[9] J. Qi et. al, "Artificial intelligence applications in the telecommunication industry", in Expert Systems, vol. 24, pp. 271-291, September 2007.

[10] T.T.T. Nguyen, G. Armitage, "A survey of techniques for internet traffic classification using machine learning", IEEE Communications Surveys & Tutorials, vol.10, no.4, pp.56- 76, Fourth Quarter 2008.

[11] S. Haykin, Neural networks and learning machines, 3rd Ed., Pearson Co., NJ, USA, 2009.

[12] C. Huang, Y. Chuang, and D. Yang, "Implementation of call admission control scheme in next generation mobile communication networks using particle swarm optimization and fuzzy logic systems", Expert Systems with Applications, vol. 3, pp. 1246-1251, Oct 2008.

[13] Ping Guo, Yinghua Jiang, Jingan Ren, "Policy-based QoS Control Using Call Admission Control and SVM 2nd International Conference on Pervasive Computing and Applications, 26-27 July 2007

[14] A. Bashar, G. Parr, S. McClean, B. Scotney, and D. Nauck, "Knowledge discovery using Bayesian network framework for intelligent telecommunication network management", in Proc. of 4th International Conference on Knowledge Science, Engineering and Management, (KSEM 2010), Springer LNAI, vol. 6291, pp. 518-529, Sep 2010.

[15] R. Zhang, and A. Bivens, "Comparing the use of Bayesian networks and neural networks in response time modeling for service-oriented systems," in Proc. of ACM Workshop on Service-Oriented Computing Performance, SOCP 2007, pp. 67-74, June 2007.