

# SVM-based analysis and prediction on network traffic

Weidong Luo Xingwei Liu Jian Zhang

School of Mathematics & Computer Engineering, Xihua University, Chengdu 610039, P.R.China

## Abstract

With continuous scale-up of the network and increase of the kinds of the services on the network, more and more people pay attention to the modeling and prediction for network traffic. Recently, SVM (Support Vector Machine), a new machine learning method, is comprehensively used to solve the problem of non-linear classification and regression. A network traffic predictive method presented in this paper is based on the LS-SVM (Least Squares SVM). Using NS2 simulator, we simulate the process of the network running with Drop-tail and RED controller respectively, then collect the being predicted traffic data which is on the bottleneck router. The results on the precision of prediction is good and feasible.

**Keywords:** Network traffic, Predictive model, LS-SVM, NS2

## 1. Introduction

In the evolving process of the network, the research of modeling and prediction for the network traffic is all along focused on by people. The traffic model is the foundation of network performance analysis and designing. Good model and predictive methods are significant for network protocol designing, network management as well as diagnoses, high-powered network devices designing and network Qos enhancement of the next generation network.

With continuous scale-up of the network and increase of the kinds of the services on the network, network traffic in time series has been a non-linear, fast changed dynamical system. Network traffic in real environment is of self-similarity, outburst and fractal etc. Therefore, the traditional traffic model such as Markov, Poisson, AR model etc. couldn't depict those characteristics sufficiently. Hence, new model method of accurate prediction is now a focus on network traffic research [1]-[3].

The rest of the paper is organized as follows. Certain new network traffic predictive methods are discussed in Section 2. The network prediction is in section 3. Discussion and future work is in section 4.

## 2. Related work

Mathematic model of network traffic can be used to predict the future traffic, but there is also needs some AI (Artificial Intelligence) methods to make the predictive process more efficient and flexible.

Recently, AI methods are comprehensively used on traffic prediction such as experts system, fuzzy logic, fuzzy neural etc., especially the ANN (Artificial Neural Network)[4].

Fuzzy theory is available in time series prediction, furthermore it has two merits that parameters of the fuzzy system is explicit in physic meaning and control rules of the fuzzy system contains the intrinsical information of the unknown non-linear system. Pang [5] proposed a self-adaptive traffic predictor based on the fuzzy theory on the ATM network's bandwidth allocating and queue management.

Another method is wavelet analysis. Complex network traffic through wavelet transformation can be predicted by traditional methods. Besides, wavelet prediction is fast in convergence and small in predictive error [6].

SVM is a new machine learning method that is put forward by V. Vapnik et al. and based on SLT (Statistics Learning Theory) and SRM (structural risk minimization) is used comprehensively in many fields, including pattern recognition, data mining, text classification, IDS(Intrusion Detection System) and time series prediction[7]-[9]. Compared with other learning machine, SVM has some unique merits like small sample sets, high accuracy and strong generalization performance. Standard SVM has a distinct characteristic it usually reflects the low-dimension non-linear object into the linear object in the high-dimension space. Then the linear

object can be easily copied with whatever classification or regression. In the process of up-dimension, SVM use Kernel function that meets Mercer condition to guarantee the original problem can be transformed into a protruding double-optimized problem with in-equation limitation. In standard SVM, we marked matrix  $Q = \sum_{i,j=1}^n (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)K(x_i, x_j)$  and its scale is square of the training sample number. If the sample scale is very large, the matrix  $Q$  will be so tremendous that the normal computer can not store it or compute it quickly using general double-optimization method. Now, many training algorithms have been put forward, including decomposition, distortion, geometry and multi-kinds algorithm etc. LS-SVM [10,11] is belonged to distortion algorithm. Compared with standard SVM, LS-SVM use square of the error as its loss function, it adds some functions, variables and coefficients to distort the original equations and retain the equation restriction and delete the in-equation restriction. The linear equations through the distortion that can be solved by least square method and decreases the computing complex. There are three kernel functions usually used:

- 1) Polynomial kernel:  $K(x_i, x_j) = (x_i \cdot x_j + 1)^d$
- 2) Sigmoid kernel:  $K(x_i, x_j) = \tanh[c_1(x_i \cdot x_j) + c_2]$
- 3) RBF kernel:  $K(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / \sigma^2)^d$

In the real application, people usually take RBF kernel, for it is only need one parameter  $\sigma$  - the kernel width. The larger value of the  $\sigma$ , the faster speed of the convergence of the system. For the combination of RBF and LS-SVM, the main parameter are  $\gamma$  and  $\sigma$ . These two parameters make decision mainly of learning and generation ability of the LS-SVM. The chief work is to seek the optimum combination  $(\gamma, \sigma)$ . But now, there is no a uniform method to find it.

### 3. Network traffic prediction

#### 3.1. Network topology

We use the topology shown in Figure 1. with NS2 [12]-[14] to simulate a LAN environment. There are 100 TCP sources (hosts in the LAN) with FTP flow connect to a destination (the outsides). They all pass through a bottleneck router  $R1$  that is our measure object. The bottleneck link whose bandwidth is 20Mb/s and delay is 60ms, locates between the two routers  $R1$  and  $R2$ . And the links between sources and  $R1$  are all 10Mb/s bandwidth and 5ms delay. And that between  $R2$  and Destinations are 45Mb/s and delay 5ms. All FTP are start obey the

Pareto distribution to simulate a continuous process of communication, and the entire simulation time is 200 sec.

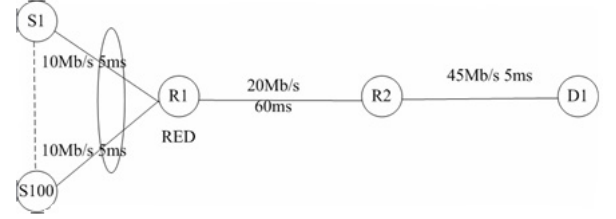


Fig. 1: Network topology.

We use RED controller and DropTail controller on  $R1$ , and they are prevalent used AQM(Active Queue Management) methods respectively used on high performance router and normal router in reality. In this way, we get two group data to make the experiment. The traffic data is collected by  $\Delta t$  second interval:

$$traffic_i = \Delta in / \Delta t_i \quad (5)$$

We set  $\Delta t = 0.5(\text{sec})$ , and  $\Delta in$  is the average of the received byte at the  $R1$  during the  $i$ th interval. The data is shown in Figure 2. We can see, the jitter of traffic is different because of the RED is inferior to DropTail by the way of marking/dropping the packet arrived in advance. And the traffic are non-linear and has been deemed that be of self-resemblant in nature [15] that makes it predictable.

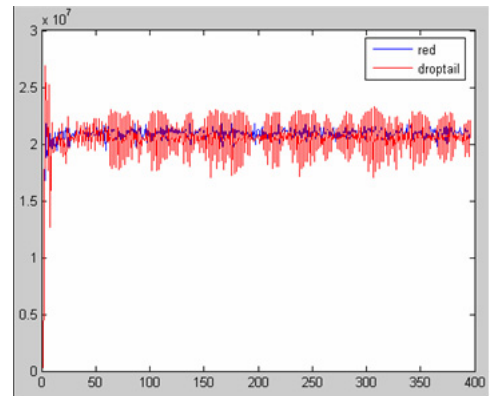


Fig. 2: Network traffic.

#### 3.2. Prediction process

Define  $y_k$  stands for the traffic at the time  $k$ . In order to get the stable time series so as to build conveniently the predictive model, we make difference operation for  $y_k$  and define  $\nabla y_k = y_{k+1} - y_k$ .

And we could make the second time difference operation on  $\nabla y$ , if the stabilization is not met.

In order to show the tendency of time series and build accurate model [16], we choose the SVM input vector:

$$x_k = [\nabla y_k, \nabla y_{k-1}, \nabla y_k - \nabla y_{k-1}]$$

in the input vector,  $\nabla y_k$  is the difference series in the current time and  $\nabla y_{k-1}$  is that in the last time. The SVM output vector is defined to:

$$y_k = \nabla y_{k+1}$$

that is the difference series in the next time. RBF kernel  $k(x_i, x) = \exp[-\|x - x_i\|^2 / 2\sigma^2]$  is chooser, and parameter  $\sigma$  should be confirmed reasonable. So the output of the predictive model is  $\nabla y_k = \sum_{i=1}^n \alpha k(x_i, x) + b$ . Prediction queue size is:

$$\tilde{y}_{k+1} = y_k + \nabla y_k.$$

The entire prediction process shown in Figure 3. is complex and repeated including:

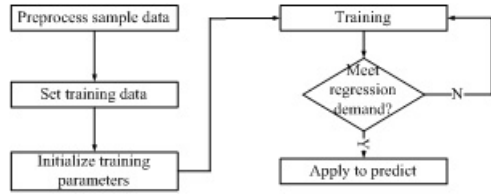


Fig. 3: Process of LS-SVM modeling prediction.

First, preprocessing the data (difference operation) which can promote the prediction precision and easy to be dealt with at the next step. Second, using predictive model we defined and setting initial parameters  $C = \infty$ ,  $\sigma^2 = 1$ . Third, setting the learning sample data  $\{x_k, y_k\}_{k=1}^l$ , it must be reasonable. If it too small the learning degree is not enough and leads to bad performance on generation. On the opposite, may be excess on regression. Fourth, defining prediction error  $\Delta^{k+1} = y(k+1) - \tilde{y}(k+1)$  and model estimation standard  $MAE = \sum_{i=1}^n |\Delta_i| / S$ , specify the sample number is  $S$ . Fifth, setting prediction error accuracy  $\eta_\Delta = |y - \tilde{y}| / y$ . The learning result  $\eta_\Delta\%$  which stands for the degree of regression must greater than 90% in requirement. Sixth, using LS-SVM algorithm to train the model until the  $MAE < \Delta$ , then validating the parameters  $\alpha$  and  $b$ . Finally, the model after training can be applied to predict the future output.

### 3.3. Results

According to the specified process mentioned above, we choose 300 traffic data as the experiment data. The front 100 is the training data

$\{x_k, \nabla y_{k+1}\}_{k=1}^{100}$  and the latter 200 is the prediction data  $\{x_k, \nabla y_k\}_{101}^{300}$ . After the model trained well, we use it to predict. For the data generated by DropTail, the predictive result is shown in Figure 4. The parameter pair  $(\gamma, \sigma) = (8, 30)$ , and predictive average error  $MAE = 6.1335e+5(byte)$ .

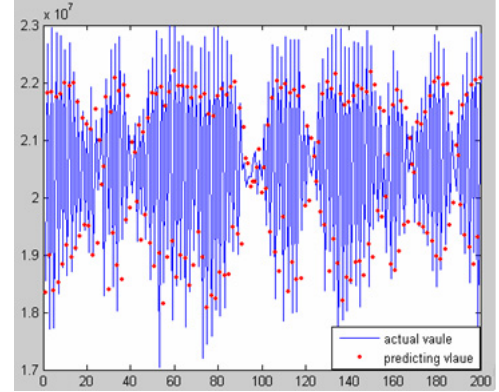


Fig. 4: DropTail prediction results.

The next result is for RED in Figure 5. The parameter pair  $(\gamma, \sigma) = (100, 8)$ , and predictive average error  $MAE = 2.9647e+5(byte)$ . Now, we can see the predictive effect are both good but RED superior to the DropTail because of its jitter is smaller than the latter.

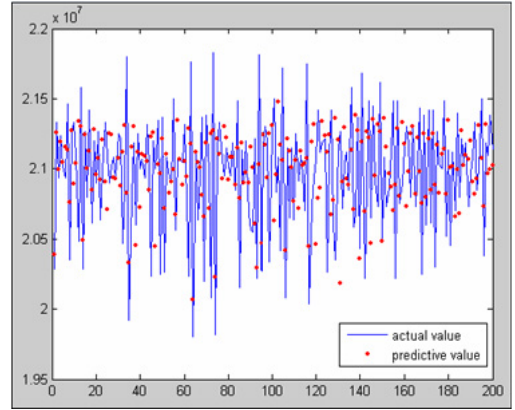


Fig. 5: RED prediction results.

In the Figure 4 and 5, the blue line stands for the actual value and red \* is the SVM output(predicting). The prediction error accuracy  $\eta_\Delta$  is shown in Table 1.

## 4. Discussion and future work

Network traffic engineering is significant for network designing, resource management, Qos and

sample type	DropTail	RED
< 5%	91.5%	98.5%
[5% ~ 10%)	6%	1.5%
[10% ~ 15%)	2.5%	0
≥ 15%	0	0

Table 1: Prediction error accuracy.

congestion control. The traditional traffic models just fit for the stable traffic prediction, but they are not good in non-linear process. From the experiment, our predictive model based on LS-SVM could obtain higher predictive accuracy.

However, the results would be bad if the traffic is time-jump seriously or the parameters are set unreasonable. So our future work is to optimize the predictive model and to design a well prediction and control scheme to realize the network congestion control in certain degree.

## References

- [1] A. Feldmann, A. C. Gigert, W. Willinger, et al, Looking behind and beyond self-similarity: Scaling phenomena in measured WAN traffic[A]. *Prof. the 35th Annual Allerton Conference on Communication, Control, and Computing*, 267-280, 1997.
- [2] W. E. Leland, M. S. Taqqu, W. Willinger, et al, On the self-similar nature of Ethernet traffic[J]. *IEEE/ACM Transaction on Network*, 2(1):1-15, 1994.
- [3] W. Willinger, M. S. Taqqu, R. Sherman, et al, Self-Similarity Through High-Variability: Statistical analysis of Ethernet LAN traffic at the source level[A]. *Prof. the ACM S ICCOMM'95*, 1(5):71-86, 1995.
- [4] H. S. Hippert, C. E. Pedreira, R.C. Souza, Neural networks for shortterm load forecasting: a review and evaluation[J]. *IEEE Transaction on Power System*, 16(1):44-55, 2001.
- [5] Q. X. Pang, S. D. Cheng, P. Zhang, Adaptive Fuzzy Traffic Predictor and Its Applications in ATM Networks[C]. *In:Prof. ICC'98*, pp.1759-1763, 1998.
- [6] X. Y. Wang, J. S. Meditch, Adaptive wavelet predictor to improve bandwidth allocation efficiently of VBR video traffic[J]. *Computer Communication*, (22):40-45, 1999.
- [7] V. Vapnik, Nature of Statistical Learning Theory[M]. *Springer*, 2000.
- [8] C. Nello, S. T. John, An Introduction to Support Vector Machines and Other kernel-based Learning Methods[M]. *Li Guo Zheng et el, Beijing, Publishing House of Electronics Industry*, 2004.
- [9] V. Vapnik, E. Levin, Y. Lecun, Measuring the VC-Dimension of a learning machine[J], *Neural Computation. MIT Press*, (6):851-876, 1994.
- [10] J. A. K. Suykens, J. K. Branbanter, L. Lukas, et al, Weighted least squares support vector machines: robustness and sparse approximation[J]. *Neurocomputing*, 48(1):85-105, 2002.
- [11] B. Liu, H. Y. SU, J. Chu, Predictive control algorithm based on least squares support vector machines[J]. *Control and Decision*, 19(12):1399-1402, 2004.
- [12] F. Kevin, V. Kannan, The ns Manual, <http://www.isi.edu/nsnam/ns/>. February 2005.
- [13] E. C. Park, H. Lim, K. J. Park, and C. H. Choi. Analysis and design of the virtual rate control algorithm for stabilization queues on TCP networks[J]. *Computer Networks, Elsevier Science*, 44:17-41, 2003.
- [14] Y. F. Fan, F. Y. Ren, and C. Lin, Design a PID controller for active queue management[A]. *Prof. the 8th IEEE International Symposium on Computers and Communication (ISCC 2003)*, pp. 985-990, Kemer-Antalya, June 2003.
- [15] X. F. Wang, J. G. Lu, Z. Q. Wang, Self-similarity in internet traffic-modeling,analysis and control[J]. *Decision-making and Control*, 17(1):1-5, 2002.
- [16] H. Z. An, Time series analysis[M]. *ShangHai, east of China normal school press*, 1992.