A novel minimax probability machine for network traffic prediction

Mu Xiangyang[1,2] , Zhang Taiyi[1]
1. Dept. of Information and Communication Engineering
Xi'an Jiaotong University
Xi'an, 710049
2. Xi'an Shiyou University
Xi'an, 710065
China
Muyou98@xsyu.edu.cn
Zhou Yatong[3]
3. School of Information Engineering,
Hebei University of Technology,
Tianjin ,300401
China
Zhouyatong_zw@126.com

Abstract: - Network traffic prediction is important to network planning, performance evaluation and network management directly. A variety of machine learning models such as artificial neural networks (ANN) and support vector machine (SVM) have been applied in traffic prediction. In this paper, a novel network traffic one-step-ahead prediction technique is proposed based on a state-of-the-art learning model called minimax probability machine (MPM). In the experiments, the predictive performance is tested on two different types of traffic data, Ethernet and MPEG4, at the same timescale. We find the predictions of MPM match the actual traffics accurately. Furthermore, we compare the MPM-based prediction technique to the SVM-based techniques. Results show that the predictive performance of MPM is competitive with SVM

Key-Words: - network traffic, minimax probability, support vector machine, prediction

1. Introduction

Network traffic prediction is of significant interest in many domains, including congestion control, admission control and network bandwidth allocation. In high-speed network such as asynchronous transfer mode (ATM), the bandwidth can be allocated based on the accurate traffic prediction, thus ensuring Qos of the users and accomplishing the preventive congestion control [1]. Traffic prediction requires accurate modeling techniques which can capture the statistical characteristics of actual traffic [2]. The traditional linear model cannot capture the property of uncertainty and time-variance about network traffic [3]. In order to improve prediction performance, nolinear models such as artificial neural networks (ANN) [4] and Markov modulated Poisson process (MMPP) models [5] are introduced to capture the real traffic characteristic. However, these models suffer from problems like the existence of local minima or the choice of model structure. Recently, support vector machines (SVM), a new modeling technique, has been applied for traffic prediction [1].

Within the machine learning community, there has been a good deal of excitement about the use of MPM model for regression and prediction in recent years [6][7]. Compared to SVM, MPM pays more attention to the typical rather than the boundary samples [8][13]. It is in some sense similar to the relevance vector machine proposed in Tipping [9]. Furthermore, the MPM is related to vicinal risk minimization [10], in which SVM were improved using the covariance of the classes to push the hyperplane away from the samples that belong to the class with the largest covariance matrix.

Since the MPM presents many merits, it has been suggested in various applications [11][14]. However, little attention has been paid to apply MPM to the prediction of network traffic. It is value for us to investigate the problem of whether a good performance could be obtained if we apply MPM to the traffic prediction. In this paper, a novel MPM-based network traffic one-step-ahead prediction technique is proposed and tested on two different types of traffic data, Ethernet and MPEG4. Experiments illustrated that the MPM can capture the uncertainty and time-variance about network traffic. Its predictive performance is comparable to SVM.

In rest of this paper, we formulate the problem of network traffic prediction in the next section,then a novel MPM model for network traffic prediction is introduced. Following that Experiments are reported . and some results are summarized in the end.

2 . Minimax probability machine for network traffic prediction

2.1 Network Traffic Prediction

Network traffic presents the number of packets per unit time. The traffic data can be seen as a time series $s(n)$ varied with the time n. We could predict the future traffic level by constructing a prediction model which takes into account the past observations. To be more specific, assume that exists a smooth map $f: R^d \to R$ such that

$$s(n) = f\left[s(n-1), s(n-2), \cdots, s(n-d)\right] \quad (1)$$

If the map f were known, the value of series s at n is uniquely determined by its d values in the past. So the prediction task can be achieved by estimating the map f.

For simplicity of notation, we define the scalar $t_n \equiv s(n)$ and the d-dimensional vector $\mathbf{x}_n \equiv \left(s(n-1), s(n-2), \cdots, s(n-d)\right)^T$ in such a way that Eq.(1) can be written simply as

$$t_n = f(\mathbf{x}_n) \ . \quad (2)$$

In order to estimate f, a training samples set D_N with capability N can be constructed as follows: $D_N = \left\{(\mathbf{x}_n, t_n) \in R^d \times R \mid n = 1, \cdots, N\right\}$

2.2 MPM for Network Traffic Prediction

Given the training samples set D_N, the MPM would like to estimate $f(\mathbf{x})$ by finding a model that maximizes the minimum probability of being $\pm\varepsilon$ accurate

$$\max\left[\min P\left(\left|f(\mathbf{x}) - t\right| \le \varepsilon\right)\right] \quad (3)$$

Assume the function $f(\mathbf{x})$ has the form of

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b_r = w^{(1)}x^{(1)} + w^{(2)}x^{(2)} + \cdots + w^{(d)}x^{(d)} + b_r \quad (4)$$

where $\mathbf{w} = \left(w^{(1)}, w^{(2)}, \cdots, w^{(d)}\right)^T$. MPM formulates the Eq. (3) as a binary classification problem to determine the parameters $\mathbf{w}$ and b_r.

The MPM turns each training sample $(\mathbf{x}_i, t_i)$ for $i = 1, \cdots, N$ into two class of $d+1$ dimensional vectors $\tilde{\mathbf{x}}_i$ or $\tilde{\mathbf{y}}_i$. The formal is labeled as class $\tilde{\mathbf{x}}$, and the latter one as class $\tilde{\mathbf{y}}$.

$$\tilde{\mathbf{x}}_i = (t_i + \varepsilon, \mathbf{x}_i)^T = \left(t_i + \varepsilon, x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}\right)^T, \quad i = 1, 2, \cdots, N \quad (5)$$

$$\tilde{\mathbf{y}}_i = (t_i - \varepsilon, \mathbf{x}_i)^T = \left(t_i - \varepsilon, x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}\right)^T, \quad i = 1, 2, \cdots, N \quad (6)$$

The above defined artificial classification problem could be solved by any binary classifier. In this paper, we focus on using MPM for classification (MPMC) [8] as the underlying classifier for the problem defined by (5) and (6). Assume the boundary obtained by the MPMC is

$$\mathbf{a}^T\mathbf{u} = b \quad (7)$$

where

$$\mathbf{u} = \left(f\left(\mathbf{x}\right), \mathbf{x}\right)^T = \left(f\left(\mathbf{x}\right), x^{(1)}, x^{(2)}, \cdots, x^{(d)}\right)^T \qquad (8)$$

The parameters $\mathbf{a} = \left(a^{(1)}, a^{(2)}, \cdots, a^{(d)}, a^{(d+1)}\right)^T$ and b in Eq. (7) can be determined by following constrained optimization problem

$$\min_{\mathbf{w}} \left(\left\| \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}^{1/2} \mathbf{a} \right\|_2 + \left\| \mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}^{1/2} \mathbf{a} \right\|_2 \right) \qquad (9)$$

$$s.t. \quad \mathbf{a}^T \left(\boldsymbol{\mu}_{\tilde{\mathbf{x}}} - \boldsymbol{\mu}_{\tilde{\mathbf{y}}} \right) = 1$$

where $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$, $\boldsymbol{\mu}_{\tilde{\mathbf{y}}}$, $\mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}$, and $\mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}$ satisfy $\tilde{\mathbf{x}} \sim \left(\boldsymbol{\mu}_{\tilde{\mathbf{x}}}, \mathbf{R}_{\tilde{\mathbf{x}}\tilde{\mathbf{x}}}\right)$,and $\tilde{\mathbf{y}} \sim \left(\boldsymbol{\mu}_{\tilde{\mathbf{y}}}, \mathbf{R}_{\tilde{\mathbf{y}}\tilde{\mathbf{y}}}\right)$. In addition, the Ref. [8] gave a kernelized version of the optimization problem (9) by mapping the samples into a high-dimensional feature space.

The boundary (7) obtained by the MPMC turns directly into the prediction function one wants to estimate. That is to say, once the parameters $\mathbf{a}$ and b have been determined, we use the classification boundary to predict the output $f(\mathbf{x})$ for a new input $\mathbf{x}$. When substituting expression (8) in (7), we obtain

$$a^{(1)} f\left(\mathbf{x}\right) + a^{(2)} x^{(1)} + a^{(3)} x^{(2)} + \cdots + a^{(d+1)} x^{(d)} = b \qquad (10)$$

The Eq. (10) can be reformulated as

$$f\left(\mathbf{x}\right) = -\frac{a^{(2)}}{a^{(1)}} x^{(1)} - \frac{a^{(3)}}{a^{(1)}} x^{(2)} - \cdots - \frac{a^{(d+1)}}{a^{(1)}} x^{(d)} + \frac{b}{a^{(1)}} \qquad (11)$$

Compared the Eq. (11) to Eq. (4), it is derived

$$w^{(i)} = -a^{(i+1)} / a^{(i)} \;, \; i = 1, 2, \cdots, d \qquad (12)$$
$$b_r = b / a^{(1)}$$

3. Experimental Results

In the experiments, we use the Ethernet and the MPEG4 network traffic as real traffic series for prediction. The MPEG4 traffic is available at [12], and the Ethernet is collected at Bellcore Morristown Research and Engineering Center. Two traffics present the number of packets per unit time. And they are aggregated at different timescales of 1 and 5 seconds. In this paper we only consider the timescales of 1 seconds.

Before prediction, two different types of traffic data are normalized to the interval [0, 1]. After that, we construct 120 samples for training and 100 for testing candidate models. Mean square error (MSE) and prediction error (PE) are used as performance measures in prediction. Their definitions are

$$MSE = \frac{1}{NT} \sum_{n=1}^{NT} \left[s\left(n\right) - f\left(\mathbf{x}_n\right) \right]^2 ,$$

$$PE\left(n\right) = s\left(n\right) - f\left(\mathbf{x}_n\right),$$

where NT is the number of test samples, $s(\cdot)$ is the actual series and $f(\cdot)$ is the prediction.

The kernel functions for MPM and SVM are both based on Gaussian kernel, that is

$$K\left(\mathbf{u}_n, \mathbf{u}\right) = \exp\left(-\left\| \mathbf{u}_n - \mathbf{u} \right\|^2 / \sigma^2 \right)$$

where the kernel parameter σ is determined by a simple cross-validation technique. For the MPM model, the parameter ε is set for 0.4. For the SVM, the regularization factor is set for 10^3 , and the insensitive loss parameter is set for 0.02. For simplicity, the dimension parameter d is fixed as 3.

Fig.1 and Fig.2 show samples of the prediction results in graphical form. The actual values of traffic (at timescale 1 seconds) are shown as solid line, the corresponding predicted values are superimposed on the actual values as dotted line and dashed line respectively. As it can be seen from the figures, we find the predictions of MPM match the actual traffics accurately.

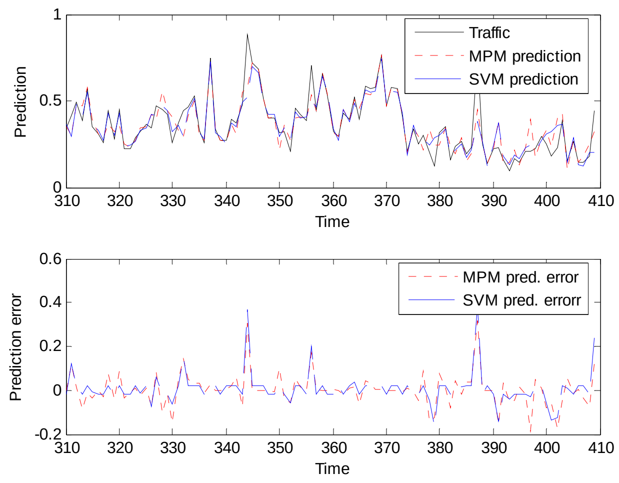

Fig.1 The actual and predicted values of Ethernet traffic. The upper graph shows traffic (solid line), the prediction using MPM (dotted line) and SVM (dashed line). The lower graph shows the prediction error curves of two models.

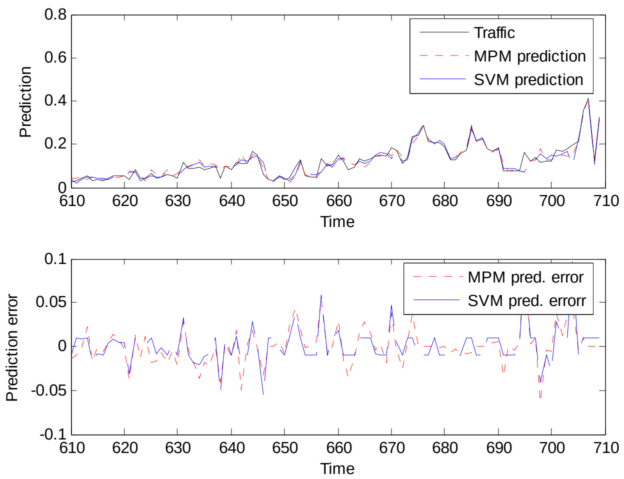

Fig.2 The actual and predicted values of MPEG4 traffic. The upper graph shows traffic (solid line), the prediction using MPM (dotted line) and SVM (dashed line). The lower graph shows the prediction error curves of two models.

To analyze the effect of the parameter d on the predictive performance, we change the value of d from 3 to 5 and 8. The predictive performance is quantified by the MSE and is shown in Table 1. We can see from the table that the performance of MPM and SVM is comparable. The results also show that the parameter d has a significant effect on the performance. In addition, we notice that for each model, using a larger d in prediction usually gives a smaller MSE. However, the prediction speed would reduce significantly when the parameter d is too large.

Table 1. Prediction performance on two traffics expressed in terms of mean square error (MSE).

Models	d	MSE ($\times 10^{-3}$) Ethernet	MPEG4
MPM	3	5.706	0.466
	5	0.392	0.045
	8	0.412	0.101
SVM	3	5.554	0.453
	5	0.380	0.096
	8	0.386	0.091

4. Conclusion

To overcome the drawbacks of traditional learning models for network traffic prediction, a novel MPM-based traffic one-step-ahead prediction technique is proposed in this paper. The prediction performance is tested on two different types of traffic data, Ethernet and MPEG4, at the same timescale. The experiments demonstrate that the proposed technique attains satisfactory performance in prediction accuracy. Therefore, the proposed technique can be used for congestion control in high-speed network, to meet the user QoS requirements.

The experiments also demonstrate that the predictive performance of MPM is competitive with SVM. Compared to the SVM, the MPM can obtain a direct estimate of a lower probability bound Ω , the future predicted outputs of the model will be within some $\pm\varepsilon$ bound of the true values .

References:

[1] Liu Zunxiong, Zhang Deyun, Liao Huichuan. Multi-scale combination prediction model with least square support vector machine for network traffic. Int. Sym. Neural Networks (ISNN), Chongqing, China. LNCS, Springer-verlag. 2005, Vol. 3498, pp385-390.

[2] A. Khotanzad, and N. Sadek. Multi-scale high-speed network traffic prediction using combination of neural networks. Proceedings of the International Joint Conference on Neural Networks. vol.2. pp. 1071- 1075, July 2003.

[3] B. S. Chen, S. C. Peng and K. C. Wang. Traffic Modeling, Prediction, and Congestion Contol for High-Speed Networks: A Fuzzy AR Approach. IEEE Trans. Fuzzy Systems, 2000, 8(5), pp. 491-508.

[4] J. Hall and P. Mars. Limitation of artificial neural network for traffic prediction in broad networks, IEEE Proc. Comm., vol.147, pp.114-118, April 2000.

[5] A. Sang and S.-Q. Li. A predictability analysis of network traffic. Computer Networks, 2002, 39(4), pp. 329-345.

[6] T. R. Strohmann and G. Z. Grudic. A Formulation for minimax probability machine regression. In Advances in Neural Information Processing Systems (NIPS) 14, T. G. Dietterich, S. Becker, and Z. Ghahramani editors, Cambridge, MA: MIT Press, 2002.

[7] T. R. Strohmann and G. Z. Grudic. Robust minimax probability machine regression. Submitted to Journal of Machine Learning Research. Available from http:// nago.cs.colorado.edu/~strohman/

[8] G. R. G. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan. A robust minimax approach to classification. *Journal of Machine Learning Research*, Vol.3, pp.555–582, 2002.

[9] M. E. Tipping. The relevance vector machine. In Solla, S., Leen, T. K., and Muller, K.-R., editors, Advances in Neural Information Processing Systems (NIPS) 12. Morgan Kaufmann.

[10] O. Chapelle, J. Weston, L. Bottou, and V. Vapnik. Vicinal risk minimization. In T. Leen, eds, Advances in Neural Information Processing Systems (NIPS) 13, pp416-422. MIT Press.

[11] C. H. Hoi and M. R. Lyu, Robust face recognition using minimax probability machine, Tech. Rep., Dep. Computer Science and Engineering, The Chinese University of Hong Kong, 2004.

[12] F. H. P. Fitzek and M. Reisslein. MPEG4 and H.263 video traces for network performance evaluation (extended version). Technical Report: TKN-00-06, TU Berlin Dept. Electrical Engineering, Telecommunication networks Groups, 2000.

[13] P. Tzekis, A. Papastergiou, A. Hatzigaida and A. Cheva, A Sophisticated Edge Detection Method for Muscle Biopsy Image Analysis, Procedings of 7th WSEAS International Conference on SIGNAL, SPEECH AND IMAGE PROCESSING, pp118-123.

[14] Seok-Han Lee , Kyung-Hyun Yoon , and Jong-Soo Choi, An Efficient Method for Robust Height and Position Estimation of Moving Objects. Proceedings of 6th WSEAS International Conference on CIRCUITS, SYSTEMS, ELECTRONICS, CONTROL & SIGNAL PROCESSING (CSECS'07),pp362-367