A Boosting-Based Framework for Self-Similar and Non-linear Internet Traffic Prediction

Hanghang Tong[1], Chongrong Li[2], and Jingrui He[1]

[1] Department of Automation, Tsinghua University, Beijing 100084, China
{walkstar98, hejingrui98}@mails.tsinghua.edu.cn

[2] Network Research Center of Tsinghua University, Beijing 100084, China
licr@cernet.edu.cn

Abstract. Internet traffic prediction plays a fundamental role in network design, management, control, and optimization. The self-similar and non-linear nature of network traffic makes highly accurate prediction difficult. In this paper, a boosting-based framework is proposed for self-similar and non-linear traffic prediction by considering it as a classical regression problem. The framework is based on Ada-Boost on the whole. It adopts Principle Component Analysis as an optional step to take advantage of self-similar nature of traffic while avoiding the disadvantage of self-similarity. Feed-forward neural network is used as the basic regressor to capture the non-linear relationship within the traffic. Experimental results on real network traffic validate the effectiveness of the proposed framework.

1 Introduction

Internet traffic prediction plays a fundamental role in network design, management, control, and optimization [11, 12, 14]. Prediction on large time scale is the base of long term planning of Internet topology. Dynamic traffic prediction is a must for predictive congestion control and buffer management. Highly accurate traffic prediction helps to make the maximum use of bandwidth while guaranteeing the quality of service (QoS) for Variable-Bite-Rate video which has a stringent requirement on delay and cell rate loss (CRL). Traffic prediction on different links can also be used as a reference to route. Layered-encoded video applications can determine whether to transmit the enhanced layer according to prediction results.

Essentially, the statistics of network traffic itself determines the predictability of network traffic [2, 9, 11, 12, 14]. Two of the most important discoveries of the statistics of Internet traffic over the last ten years are that Internet traffic exhibits self-similarity (in many situations, also referred as long-range dependence) and non-linearity. Since Will E. Leland's initiative work in 1993, many researchers have dedicated themselves to proving that Internet traffic is self-similar [8, 12]. On the other hand, Hansegawa et al in [5] demonstrated that Internet traffic is non-linear by using

F. Yin, J. Wang, and C. Guo (Eds.): ISNN 2004, LNCS 3174, pp. 931–936, 2004.

surrogate method [16]. The discovery of self-similarity and non-linearity of network traffic has brought challenges to traffic prediction [9, 11, 12, 14].

In the past several decades, many methods have been proposed for network traffic prediction. To deal with the self-similar nature of network traffic, the authors in [15] proposed using FARIMA since FARIMA is a behavior model for self-similar time series [3]; the authors in [17] proposed predicting in wavelet domain since wavelet is a natural way to describe the multi-scale characteristic of self-similarity. While these methods do improve the performance of prediction for self-similar time series, however, they are both time-consuming. To deal with the non-linear nature of network traffic, Artificial Neural Network (ANN) is probably the most popular method. While ANN can capture any kind of relationship between the output and the input theoretically [5, 7, 10], however, it might suffer from over-fitting [4]. Another kind of prediction method for non-linear time series is support vector regression (SVR) [10] which is based on structural risk minimization. However, the selection of suitable kernel functions and optimal parameters is very difficult [5].

In this paper, we propose a boosting-based framework for predicting traffic which exhibits both self-similarity and non-linearity (BBF-PT), by considering traffic prediction as a classical regression problem. On the whole, BBF-PT is based on AdaBoost [13] and follows R. Bone's work [1]. To take advantage of self-similarity while avoiding its disadvantage, Principle Component Analysis (PCA) is adopted as an optional step. BBF-PT takes feed-forward neural network (FFNN) as the basic learner to capture the non-linear characteristic of network traffic and makes use of the boosting scheme to avoid over-fitting. BBF-PT itself is straightforward and can be easily incorporated with other kind of basic regressors. Experimental results on real network traffic which exhibits both self-similarity and non-linearity demonstrate the effectiveness of the proposed framework.

The rest of this paper is organized as follows: in section 2, we present our framework (BBF-PT) in detail; experimental results are given in section 3; finally, we conclude the paper in section 4.

2 Boosting-Based Framework for Predicting Traffic (BBF-PT)

2.1 Problem Definition

By denoting network traffic as a time series: $X = (x_i : i = 0, 1, 2, \cdots)$. The prediction problem can be defined as follows [2]: given the current and past observed values $X_i = (x_{i-p+1}, \cdots, x_{i-1}, x_i)$, predict the future value x_{i+q}, where p is the length of history data used for prediction and q is the prediction step.

In practice, the traffic prediction problem can be considered as a classical regression problem under the assumption that $x_i \in \zeta_2$ $(i = 1, 2, \cdots)$ [2]. That is, the prediction of x_{i+q} can be written as:

$$\hat{x}_{t+q} = \underset{y \in \xi_2}{\operatorname{argmin}} E\{(y - x_{t+q})^2\}. \quad (2)$$

2.2 Flowchart of BBF-PT

As a general method for improving the accuracy of some weak learning algorithms, boosting has been shown to be very effective for classification [1, 6, 13]. When applied to a regression problem, there are mainly two classes of methods: 1) modifying some specific steps of the existing boosting algorithms for classification so that they are suitable for a regression problem [1, 6]; 2) converting a regression problem to a classification problem by introducing an additional variable [13]. In this paper, we adopt the first method and leave the latter for future work.

How to exploit the correlation structure is the key problem in traffic prediction [11]. For self-similar traffic, it means: 1) containing more past observed values will be helpful to improve the prediction accuracy, however, it also requires more processing time; 2) traffic exhibits sustained burstiness, generally resulting in bigger peak prediction error [17]. To address these problems, we propose using Principle Component Analysis (PCA) as an optional step so that: 1) the same length of history data can contain more "information" about the past observed values and thus help to improve the prediction accuracy while not increasing the processing time drastically (here, the additional processing time is for PCA); 2) De-correlation on the input data might help to reduce the peak prediction error. Based on Ada-Boost [13] and following R. Bone's work [1], the flowchart of BBF-PT can be given as follows:

The flowchart of BBF-PT

1. Prepare a training set for regression. For every example in the training set, the input is $X_i = (x_{i-p+1}, \cdots, x_{i-1}, x_i)$ ($X_i \in R^p$), and the output is $Y_i = x_{i+q}$ ($Y_i \in R$), where $i = 1, 2, \cdots, N$ and N is the total number of examples.
2. Initialize weight distribution for all examples in the training set: $D_1(i) = 1/N$.
3. Iterate until the stopping criterion is satisfied. In every iteration $t\ (t = 1, 2, \cdots, T)$:
   - ♦ Train a weak regressor h_t using distribution D_t on the training set;
   - ♦ Get a weak regressor $h_t : R^{N_p} \to R$;
   - ♦ Evaluate the error $l_t(i)$ for every example i in the training set using the weak regressor h_t ;
   - ♦ Compute the training error ε_t of h_t ;
   - ♦ Choose $\alpha_t \in R$ to measure the importance of h_t ;
   - ♦ Update the weight distribution of the examples in the training set: $D_t \to D_{t+1}$;
   - ♦ Judge whether the stopping criterion is satisfied.
4. Combine the weak regressors: $\{h_t, t\text{=}1, 2, \cdots, T\} \to H$.

Optional: Perform PCA on the input data in the training set; store the corresponding principle axis $u_i (i = 1, 2, \cdots, N_p)$; and take the projection of X_i on $u_i (i = 1, 2, \cdots, N_p)$ as the actual input for both training and testing.

2.3 The Details of BBF-PT

The necessary modifications to make Ada-Boost for classification suitable for a regression problem include: 1) the form of the basic learner (here the basic learner is the basic regressor); 2) the way to evaluate the error information for every example; 3) the scheme of combining weak learners.

The basic regressor: essentially, BBF-PT can use any kind of basic regressors. In this paper, we adopt feed-forward neural network (FFNN) [5, 7, 17] to capture the non-linearity within the network traffic.

Evaluating the error information: There are three basic forms to evaluate the error information for examples in the training set, namely linear, squared and saturated [1, 6]. We adopt the linear form in this paper.

Computing ε_t: the training error ε_t of h_t can be computed as $\varepsilon_t = \sum_{i=1}^{N} D_t(i) \cdot L_t(i)$.

Choosing α_t and updating weight distribution: BBF-PT takes the same scheme as Ada-Boost [13] to measure the importance h_t and update weight distribution.

Stopping criterion: In Ada-Boost, the iteration will stop when $\varepsilon_t \geq 0.5$. In BBF-PT, we set a maximum iteration number $T_{\max}$ to avoid slow convergence. That is, if $\varepsilon_t \geq 0.5$ or $t > T_{\max}$, the iteration process will stop.

Combining weak regressors: There are mainly two methods to combine the weak regressors: weighted mean and weighted median. Since the weighted median method is less sensitive than the weighted mean method [1, 6], it is adopted in BBF-PT.

3 Experimental Results

The network traffic that we use is the JPEG and MPEG version of the "Star Wars" video which is widely used to examine the performance of the network traffic prediction algorithms [7]. In our experiment, we divide the original traffic into some traces with equal length 1000. Then we make use of variance-time [8] and surrogate method [16] to test self-similarity and non-linearity of a given trace, respectively. Those exhibiting both self-similarity and non-linearity are selected. Each trace is normalized to $[0,1]$ for comparison simplicity.

There are a set of parameters and operations that need to be set in BBF-PT:

- If PCA is not performed, the length of history data p used for predicting is set to be 10; else p is set to be 20 and N_p is 10;
- At the current stage, we are concerned with one-step prediction so q is 1;
- We used the first 50% data in each trace to compose the training set;
- The FFNN contains three layers and composes of 10 hidden neurons and 1 extra neuron for the output layer;
- The transfer functions for the hidden layer and the output layer in FFNN are sigmoid and linear respectively;

- The weights of FFNN are initialized randomly;
- FFNN is trained by standard back-propagation algorithm;
- The maximum iteration number $T_{\max}$ is set to be 30.

The mean-square-error (MSE) of the prediction results obtained by BBF-PT with and without PCA is listed in figure 1. It is compared with the results obtained by Feed-Forward Neural Network (mean results over 30 runs) without boosting. From figure 1, it can be seen that: 1) boosting does improve the prediction accuracy; and 2) PCA further improves the prediction performance.

We use the maximum absolute prediction error on the testing set of a given trace to measure the peak prediction error. The results obtained by BBF-PT with and without PCA are listed in figure 2. It can be seen that in the most cases, the PCA step can help to reduce the peak prediction error.

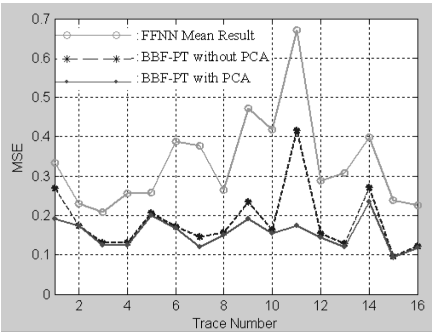

Fig. 1. MSE of prediction results

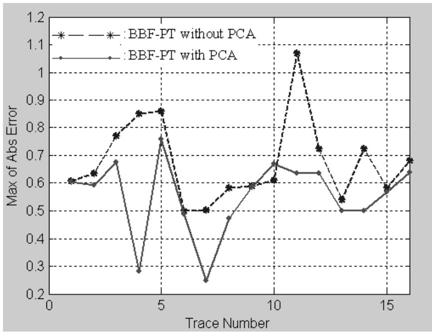

Fig. 2. Max. error of prediction results

4 Conclusion

In this paper, we have proposed a boosting-based framework for self-similar and non-linear network traffic prediction by considering it as a classical regression problem. The framework is based on Ada-Boost on the whole and makes use of FFNN to capture the non-linearity within traffic at the current stage. We also propose using PCA as an optional step to take advantage of self-similarity while trying to reduce the peak prediction error. Experimental results demonstrate the effectiveness of the proposed scheme. Future work includes: 1) incorporating other kinds of basic regressors into our framework; 2) extending the current method to multi-step prediction; 3) considering network traffic prediction from a classification point of view.

Acknowledgements. This paper is supported by National Fundamental Research Develop (973) under the contract 2003CB314805.

References

1. Bone, R., Assaad, M., Crucianu, M.: Boosting Recurrent Neural Networks for Time Series Prediction. Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms (2003)
2. Brockwell, P., Davis, R.: Time Series: Theory and Methods. Springer-Verlag, New York, 2nd edition (1991)
3. Gripenberg, G., Norros, I.: On the Prediction of Fractional Brownian Motion. Journal of Applied Probability (1996) 400-410
4. Hall, J., Mars, P.: The Limitations of Artificial Neural Networks for Traffic Prediction. IEE Proceedings on Communications (2000) 114-118
5. Hansegawa, M., Wu, G., Mizuno, M.: Applications of Nonlinear Prediction Methods to the Internet Traffic. The 2001 IEEE International Symposium on Circuits and Systems (2001) 169-172
6. Kegl, B.: Robust Regression by Boosting the Median. COLT/Kernel (2003) 258-272
7. Khotanzad, P., Sadek, N.: Multi-Scale High-Speed Network Traffic Prediction Using Combination of Neural Network. IJCNN (2003) 1071-1075
8. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.: On the Self-Similar Nature of Ethernet Traffic. IEEE/ACM Tran. on Networking (1994) 1-15
9. Lopez-Guerrero, M., Gallardo, J.R., Makrakis, D., Orozco-Barbosa, L.: Sensitivity of a Network Traffic Prediction Algorithm. PACRIM (2001) 615-618
10. Muller, K.: Predicting Time Series with Support Vector Machines. Proceedings of the International Conference on Artificial Neural Network (1997) 999-1004
11. Ostring, S., Sirisena, H.: The Influence of Long-rang Dependence on Traffic Prediction. IEEE ICC (2001) 1000-1005
12. Sang, B., Li, S.: A Predictability Analysis of Network Traffic. Proceeding of INFOCOM (2000) 343-351
13. Schapire, R.E.: The Boosting Approach to Machine Learning: an Overview. MSRI Workshop on Nonliear Estimation and Classification (2002)
14. Shah, K., Bohacek, S., Jonckheere, E.: On the Predictability of Data Network Traffic. Proceedings of the 2003 American Control Conference (2003) 1619-1624
15. Shu, Y., Jin, Z., Zhang, L., Wang, L.: Traffic Prediction Using FARIMA Models. IEEE ICC (1999) 891-895
16. Small, M., Yu, D., Harrison, R.G.: Surrogate Test for Pseudoperiodic Time Series Data. Physical Review Letter (2001)
17. Wang, X., Shan, X.: A Wavelet-based Method to Predict Internet traffic. IEEE International Conference on Communications, Circuits and Systems and West Sino Expositions (2002) 690-694