ISSN 1392 – 1215
ELECTRONICS AND ELECTRICAL ENGINEERING
ELEKTRONIKA IR ELEKTROTECHNIKA
2008. No. 3(83)

T180
TELECOMMUNICATIONS ENGINEERING
TELEKOMUNIKACIJŲ INŽINERIJA

Prediction Accuracy of Neural Network Models

G. Rutka
Faculty of Electronics and Telecommunication, Riga Technical University
Riga, Azenes str. 12-317, LV-1048, phone:+371 9627600, e-mail: gundegarutka@tvnet.lv

Introduction

The predictability of data networks and internet is assessed. Analysis of traffic data from networks and services such as Ethernet LANs [1], Variable Bit Rate (VBR) video [2], ISDN traffic [3] have demonstrated the presence of features such as self similarity, long range dependence, heavy tail distributions and fractal dimensions which are among the characteristics of fractal process.

The self-similar and non-linear nature of network traffic makes high accurate prediction difficult. The problem with self-similar models is that they are computationally complex. Their fitting procedure is time consuming while their parameters cannot be estimated based on the on-line measurements. The goal is to forecast future traffic variations as precisely as possible, based on the measured traffic history.

Self- Similarity

The process is self-similar if its statistical behavior is independent of the time-scale. This means that averaging over equal periods of time does not change the statistical characteristics of the process.

For a self similar time series:

$$\{X\}=\{X_1, X_2, \ldots, X_k\}. \qquad (1)$$

The m-aggregate $\{X_k^{(m)}\}$ with its k-th term:

$$X_k^{(m)} = \frac{X_{km-m+1} + \ldots + X_{km}}{m}, \text{ where } k=1,2,3,\ldots \qquad (2)$$

The Hurst parameter H in (1) is in the range 0.5<H<1 and it characterizes the process in terms of the degree of self-similarity and long time dependence. The degree of self-similarity and long-range dependence increases as $H\rightarrow 1$. In our experiments self-similarity will be estimated by the use of variance-time plot method. This is one of the easiest methods how to estimate Hurst's coefficient. In the process the variance of aggregate the self-similar process is defined:

$$VAR(X^{(m)})= VAR(X)/m^{\beta}. \qquad (3)$$

In the (3) β is calculated from the equation:

$$H=1-\beta/2. \qquad (4)$$

The (3) can be rewritten is the following form:

$$\log\{ VAR (X^{(m)})\} \sim \log\{ VAR(X) \}- \beta \log \{m\}. \qquad (5)$$

If VAR(X) and m are plotted on a log-log graph then by fitting a least square line through the resulting points we can obtain a straight line with the slope of $-\beta$ [4], [5], [6], [7].

The self- similar traffic prediction with neural networks

Neural networks are capable of learning complex nonlinear relationships and have been successfully applied to the problem of time series prediction.

In our research we use three types of neural networks- linear networks, multilayer perceptron (MLP) networks and radial basis function (RBF) networks. The training vector forming principles are shown in Fig.1.

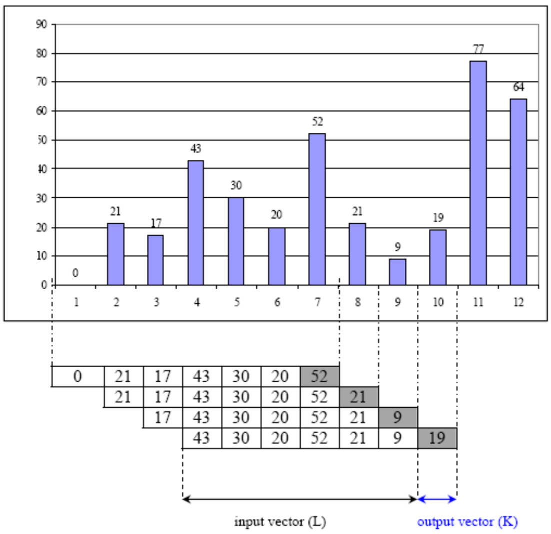

Fig. 1. Training vector forming principles

The input vector (L) is formed according to the aggregation level based on measurements of traffic self-similarity and autocorrelation function. In our experiments

L=12, 24, 48, 164, 168, 720 ect. The output vector (K) is our prediction vector. This vector is chosen according to the autocorrelation function period, if it is. We have chosen the output vector K=1, 10, 12, 24, 48, 72.

The scheme of MPL neural networks is shown in Fig. 2.

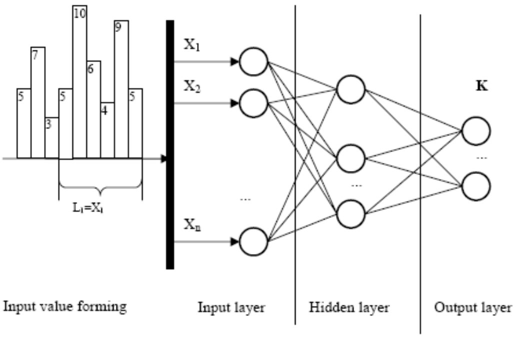

Fig. 2. The scheme of MLP neural networks

Estimation of prediction error

Many authors have applied different error estimation techniques- the mean error [8], [9], the absolute error function [10], the squared error [11], ect. In our experiments we have used the mean squared error:

$$MSE = \frac{1}{M}\sum_{t=1}^{M}\left(X_t - \hat{X}_t\right)^2, \tag{6}$$

where X_t – an output of the network at the time t, $\hat{X}_t$ – the predicted output of X_t, M – total predicted values.

As MSE grows, the accuracy of the made prediction reduces.

The description of analyzed traffic

Our research is emphasized to self- similar traffic prediction using neural networks. Traffic data is taken from website http://freestats.com/ , collected for different time periods. Another data trace is collected using website access statistics of local area network users using access to the site www.fotoblog.lv (these are real time traffic traces). As the third type of traffic data we analyze MMPP (simulated traffic traces).

Markov Modulated Poisson Process (MMPP) has been extensively used to model B-ISDN sources such as voice and video, as well as characterizing the superposed traffic. It captures the burstiness and correlation properties of the network traffic.

An MMPP is a doubly stochastic Poisson Process [12]. The arrivals occur in a Poisson manner with a rate that varies according to a k-state Markov chain, which is independent of the arrival process. Accordingly, an MMPP is characterized by the transition rate matrix of its underlying Markov chain and arrival rates. Let *i* be the state of the Markov chain, $i \in \{1,\ldots k\}$ with k maximum states and σ_{ij} - the transition rate from state I to state j, with $i \neq j$, and λ_i be the arrival rate when the Markov chain is in state i, with $\lambda_i > 0$, defining:

$$\sigma_i = -\sum_{j=1; i \neq j}^{k} \sigma_{ik}\,. \tag{7}$$

In matrix form we have:

$$Q = \begin{bmatrix} -\sigma_{11} & \sigma_{12} & \ldots & \sigma_{1k} \\ \sigma_{21} & \sigma_{22} & \ldots & \sigma_{2k} \\ \ldots & \ldots & \ldots & \ldots \\ \sigma_{k1} & \sigma_{k2} & \ldots & \sigma_{k} \end{bmatrix}, \tag{8}$$

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \ldots & 0 \\ 0 & \lambda_2 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \lambda_k \end{bmatrix}. \tag{9}$$

Assuming that Q does not depend on time t the steady state probability vector π of Q is the solution of the following system equations:

$$\pi Q = 0, \tag{10}$$

$$\sum_{i=1}^{k} \pi_j = 1. \tag{11}$$

One measure of the goodness of fit of the model is the quantile- quantile (Q-Q) plot shown in Fig.1. Here the quantiles of data trace and of a simulation of the fitted process are shown. If both sets of data were drawn from the same distribution we expect the plot to be linear. The fit, as shown, appears to be a very good (see Fig.3.).

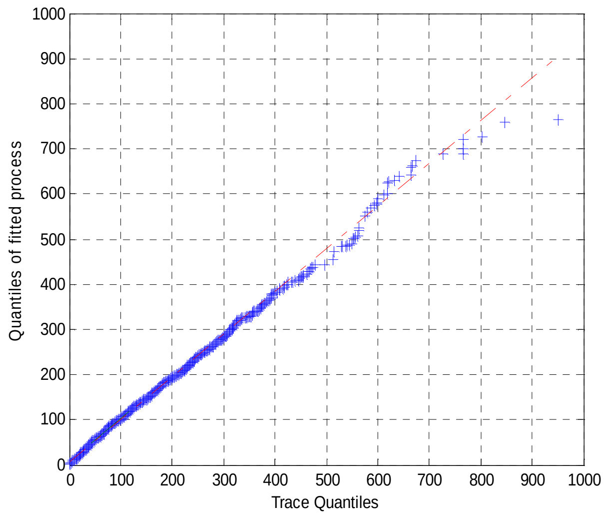

Fig. 3. Q-Q plot of Freestats and MMPP Freestats trace

The traffic sources analyzed in our experiments are summarized in Table 1 (in the brackets are indicated the day period - (7d) means 7days).

Table 1. Summary of the traffic data used in the study

No	Name	Observations	Step
1.	Freestats trace (7d)	168	1h
2.	Freestats trace (49d)	1176	1h
3.	Freestats trace (53d)	1272	1h
4.	Freestats trace (82d)	1986	1h
5.	Freestats trace (365d)	8760	1h
6.	MMPP Freestats trace (49d)	1176	1s
7.	MMPP Freestats trace (82d)	1986	1s
8.	MMPP Freestats trace (365d)	8760	1s

Estimation results of self similarity

The self similar traffic estimation was based on variance – time plot. The results are summarized in Table 2.

Table 2. Summary of the Hurst parameter

No.	**Traffic**	**Hurst parameter**
1.	Freestats trace (7d)	0.57
2.	Freestats trace (49d)	0.64
3.	Freestats trace (53d)	0.64
4.	Freestats trace (82d)	0.71
5.	Freestats trace (365d)	0.75
6.	MMPP Freestats trace (49d)	0.55
7.	MMPP Freestats trace (82d)	0.71
8.	MMPP Freestats trace (365d)	0.56

As we see in Table 2, that the real time traffic shows a trend of higher level statistical self similarity when the number of observations grows (Hurst parameter is closer to the value 1). That means the Hurst parameter H differs according to the total volume of the traffic observations (infinite is the best, but not the real case). Simulated traffic doesn't show such trend (MMPP traces).

In Fig.4. we can see the variance - time plot for Freestats traces. For Freestats trace (365d) the variance-time curve shows asymptotic slope that is easily estimated to be - 0.51, resulting in a practically identical estimate of the Hurst parameter H of about 0.745. In that case our trace is considered to be statistical self-similar with parameter H=0.745 (0.5<H<1).

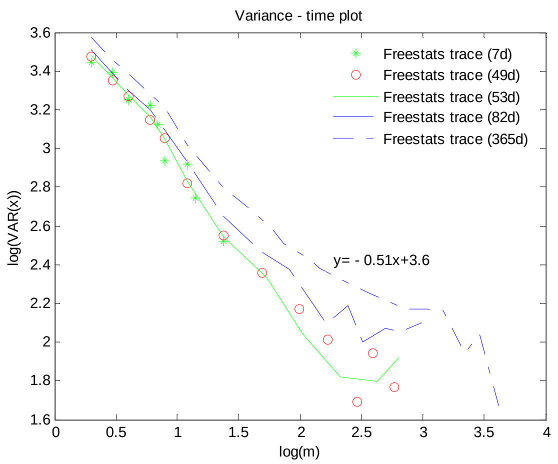

Fig. 4. Variance – time plot

Simulation results

For statistical analyses and neural network testing we use program package "MATLAB p6.5". The results are summarized in the Table 3- Table 5. There the best results for each trace are marked in "Bold", but the best result in the whole table is underlined. In the brackets is shown the input vector length L.

The best results of prediction using linear networks are summarized in Table 3.

As we see in Table 3, for Freestats trace (49d) and Freestats trace (82d) we would like to note that as the number of K increases, the prediction errors decrease. Contrary for MMPP Freestats trace (365d) we see that as the number of K increases, the prediction errors increase. In other cases we can't identify the trend. The prediction accuracy for Freestats trace (49d) we can see in Fig.5.

Table 3. The prediction error using linear networks

Traffic	**K=1**	**K=12**	**K=24**	**K=72**
Freestats trace (49d)	3964.6 (L=168)	35.73 (L=168)	0.02 (L=168)	**0.014 (L=720)**
Freestats trace (82d)	4046.5 (L=168)	1502 (L=720)	1192 (L=720)	**1.704 (L=720)**
Freestats trace (365d)	55592 (L=168)	**176.56 (L=168)**	6626.9 (L=24)	22754 (L=168)
MMPP Freestats trace (49d)	**64 (L=720)**	96.1 (L=720)	65.333 (L=720)	88.889 (L=720)
MMPP Freestats trace (82d)	64 (L=720)	96.1 (L=720)	25.546 (L=720)	**1.125 (L=1440)**
MMPP Freestats trace (365d)	**0.1301 (L=168)**	1492.8 (L=24)	3360.7 (L=24)	4325.8 (L=24)

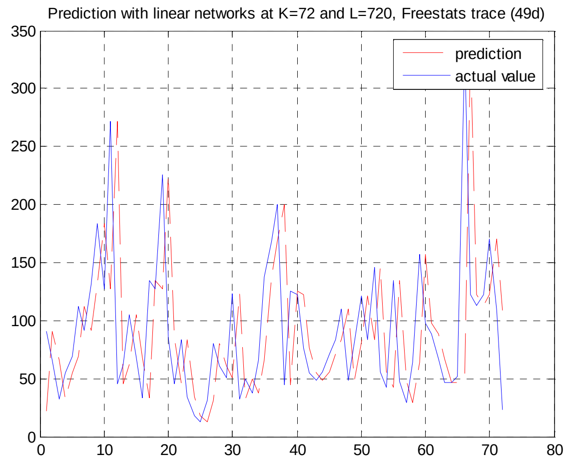

Fig. 5. The prediction accuracy with linear networks for Freestats trace (49d)

The best results of prediction using MLP networks are summarized in Table 4.

Table 4. The prediction error using MLP networks

Traffic	**K=1**	**K=10**	**K=12**	**K=24**
Freestats trace (49d)	3527.4 (L=98)	1701.1 (L=168)	**72.28 (L=98)**	634.26 (L=168)
Freestats trace (82d)	**1726.1 (L=168)**	11004 (L=720)	14099 (L=168)	6445.1 (L=168)
Freestats trace (365d)	54265 (L=24)	**72.47 (L=168)**	2243.4 (L=24)	13411 (L=24)
MMPP Freestats trace (49d)	**1726.1 (L=168)**	10434 (L=164)	5057.8 (L=164)	4552.5 (L=48)
MMPP Freestats trace (82d)	2660.8 (L=12)	22682 (L=168)	18058 (L=164)	**154.4 (L=48)**
MMPP Freestats trace (365d)	935.35 (L=168)	1061.7 (L=168)	897.93 (L=168)	**470.71 (L=168)**

In Table 4, for all traces we can't identify the trend of prediction accuracy, because it changes "bursty". The best case of the prediction was achieved for Freestats trace (365d) at L=168 and K=10.

The best results of prediction using RBF networks are summarized in Table 5.

Table 5. The prediction error using RBF networks

Traffic	**K=1**	**K=10**	**K=12**	**K=24**
Freestats trace(49d)	7396	193.6	468.75	**28.167**
Freestats trace (82d)	8836	62.5	602.08	**12.042**

The prediction result with RBF networks doesn't depend on input vector L. As we see in the Table 5, the best result is achieved at K=24 for Freestats trace (82d). This table shows the results only for two traces, because the computation time is very long and the server used for calculations wasn't able to calculate.

Conclusions

The prediction results show linear networks better predictors than MLP or RBF networks.

Neural networks exhibit long computation time. The data summarized in this paper was calculated for several months.

Prediction at the presence of self similarity is not easy. In some cases we can note that as the number of K increases, the prediction errors increase as well or vice versa. But in most cases of prediction with neural networks, the prediction errors have "bursty" characteristics. This could be explained by the self similar nature of analyzed traffic.

The prediction accuracy of simulated traffic and real time traffic differs. It is difficult to identify common trends as the prediction errors have "bursty" characteristics.

This work has been partly supported by the European Social Fund within the National Program "Support for the carrying out doctoral study program's and post- doctoral researches" project "Support for the development of doctoral studies at Riga Technical University".

References

1. **Leland, W., M. Taqqu, W. Willinger, D. Wilson.** On the Self-Similar Nature of Ethernet Traffic (Extended Version) // IEEE/ACM Transactions on Networking. – 1994. – No. 2(1). P. 1–13.
2. **Chong S., Li S., Ghosh J.** Predictive Dynamic Bandwidth Allocation For Efficient Transport Of Real Time VBR Video Over ATM // IEEE Journal of Selected Areas in Communications 13. – 1995. – P. 12–23.
3. **Hellstern K. M., Wirth P.** Traffic Models for ISDN Users: Office Automation Application // Proceedings ITC-12. – Denmark, 1991. – P. 167–172.
4. **Jon M. Peha.** Protocols can make traffic appear self-similar // IEEE/AMC/SCS Communication networks and distributed systems modeling and simulation conf. . – Jan. 1997. – P.47–52
5. **Robert S., Le Boudee J. Y.** A Markov modulated process for self-similar traffic. – Sept 23, 1995. – P. 1–14.
6. **Jonckheere E. A., Silvester J. A.** Neural network modelling of self-similar teletraffic patterns. – 2003. – P. 1–5.
7. **Rose O.** Estimation of the Hurst parameter of long range dependent time series // University of Wurzburg, Research report series. – Feb. 1996. – P. 1–15.
8. **Wrazen M., Bromirski M., Lysek K.** Technique for TCP over ATM traffic predictiom in the presence of traffic policing // Military Communications Conference MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. – 2001. – IEEE Volume 1. – P. 737–740
9. **Caines P. E., Ljung L.** Prediction error estimators: Asymptotic normality and accuracy // IEEE Conference on Decision and Control including the 15th Symposium on Adaptive Processes. – Dec. 1976. – Vol. 15, Part: 1. – P. 652–658.
10. **Jonckheere E. A., Silvester J. A.** Utilizing Neural Networks To Reduce Packet Loss In Self Similar Teletraffic Patterns. – 2003. – P. 1942–1946.
11. **Broersen P. M. T.** The prediction error of autoregressive small sample models // IEEE Transactions on Signal Processing, IEEE Transactions on Acoustics, Speech, and Signal Processing. – May 1990. – Vol. 38, No. 5. – P. 858–860.
12. **Soshant Bali, Victor S. Frost.** An algorithm for fitting MMPP to IP traffic traces // IEEE Communications Letters. – Feb. 2007. – Vol. 11, No. 2. – P. 207–209.

Submitted for publication 2007 11 28

G. Rutka. Prediction Accuracy of Neural Network Models // Electronics and Electrical Engineering. – Kaunas: Technologija, 2008. – No. 3(83). – P. 29–32.

A view of models used for Internet traffic prediction using neural network applications is presented. We look at the problem of traffic prediction in the presence of self-similarity. Our experiments inspect prediction error of linear, multilayer perceptron and radial basis function networks. Ill. 5, bibl. 12 (in English; summaries in English, Russian and Lithuanian).

Г. Рутка. Точность прогнозирования в нейронных сетях // Электроника и электротехника. – Каунас: Технология, 2008. – № 3(83). – С. 29–32.

Представлены модели, используемые для предсказания нагрузки в интернете применяя нейронные сети. Рассматривается проблема предсказания трафика в случае самопохожести. Наши эксперименты показывают точность прогнозирования в линейной сети и сетей, функционально основанных на использовании многослойных персептронов и радиальной базисной функции. Ил. 5, библ. 12 (на английском языке; рефераты на литовском, английском и русском яз.).

G. Rutka. Prognozavimo tikslumas naudojant neuroninius tinklus // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2008. –Nr. 3(83).– P. 25–28.

Pateikti apkrovoms internete prognozuoti naudojami neuroninių tinklų pagrindu sudaryti modeliai. Tiriama duomenų srauto prognozavimo savaiminio panašumo atveju problema. Eksperimentai rodo, jog prognozavimas tikslus tiesiniame tinkle ir tinkluose, kurie funkciniu požiūriu paremti daugiasluoksnių perceptronų ir radialinės bazinės funkcijos naudojimuю Il. 5, bibl. 12 (anglų kalba; santraukos anglų, rusų ir lietuvių k.).

DOI: 10.5755/j02.eie.11072