TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.4, April 2014, pp. 2548 ~ 2558
DOI: http://dx.doi.org/10.11591/telkomnika.v12i4.4802

Network Traffic Prediction Algorithm Based on Improved Chaos Particle Swarm SVM

Bai Shuzhen
Nanning College for Vocational, Guangxi Nanning, China
email: shenghuazhongqin@163.com

Abstract

Because network traffic is complex and the existing prediction models have various limitations, a new network traffic prediction model based on wavelet transform and optimized support vector machine (ChOSVM) is proposed. Firstly, the network traffic is decomposed to the scale coefficients and wavelet coefficients by non-decimated wavelet transform based on suitable wavelet base and decomposition level. Then they are sent individually into different SVM with suitable kernel function for prediction. The parameters of SVM are selected by chaos particle swarm optimization. Finally predictions are combined into the final result by wavelet reconstruction. Experiments on network traffic of different time granularity show that compared with other network traffic prediction models, ChOSVM has better performance.

Keywords: index terms-network traffic prediction, wavelet transform, chaos quantum particle swarm optimization, SVM

1. Introduction

With the rapid development of the network communication technology, the network is carrying more and more application service [1-5]. It requires higher quality of network service, traffic control and network management. Network traffic analysis and prediction have significant meanings for large-scale network capacity planning, network equipment design, network resource management and user behavior regulation. Traffic prediction with high quality is getting more and more important [6, 7].

Indeed, Traffic modeling is fundamental to the network performance evaluation and the design of network control scheme that is crucial for the success of high-speed networks [8]. This is because network traffic capacity will help each webmaster to optimize their website, maximize online marketing conversions and result in campaign tracking [9, 10]. Furthermore, detecting the efficiency and performance of IP networks based on accurate and advanced traffic measurements is an important topic in which research needs to explore a new scheme for monitoring network traffic and then find out its proper approach to predict the actual traffic accurately [11, 12]. Many models have been developed to study complex traffic phenomena [13-18] and the demand for accurate traffic parameter prediction has long been recognized in the international scientific literature [20-22].

The main purpose here is to obtain a better understanding of the characteristics of the network traffic. One of the methods used for the preventive control is to predict the near future traffic in the network and then take appropriate measures such as controlling buffer sizes. Several works developed in the literature are interested to resolve the problem of improving the efficiency and effectiveness of network traffic monitoring by forecasting data packet flow in advance. Therefore, an accurate network traffic prediction model should have the ability to capture the prominent traffic characteristics, e.g. short and long range dependence, self-similarity in large time scale and multi-fractal in small-time scale. Several traffic prediction schemes have been proposed.

Using quantum particle swarm optimization[26-28] to handle complex problems with lots of extremum has the problem of relapsing into local extremum, slow convergence velocity and low convergence precision. A quantum particle swarm optimization based on chaotic searching is proposed. Extremum disturbance can help particles quickly break away from the local optimum, and chaotic searching can improve the local searching ability. The experiment results show that the proposed algorithm is better than traditional quantum particle swarm optimization

Received September 4, 2013; Revised October 12, 2013; Accepted November 18, 2013

in ability of breaking away from the local optimum, converging speed and precision. Then we use chaos particle swarm optimization algorithm to optimize the parameters of SVM [23-25]. Simulations and comparisons demonstrate the effectiveness an efficiency of SVM parameter optimization using chaos particle swarm optimization algorithm. Since network traffic is complex and the existing prediction models have various limitations, a new network traffic prediction model based on wavelet transform and optimized support vector machine is proposed. Firstly, the network is decomposed to the scaling coefficients and wavelet coefficients by non-decimated wavelet transform based on suitable wavelet basis and decomposition level. Then they are sent individually into different SVM with suitable kernel function for prediction. The parameters of SVM are selected by chaos particle swarm optimization algorithm. Finally predictions are combined into the final result by wavelet reconstruction. Experiments on network traffic of different time granularity show that compared with other network traffic prediction models, our proposed method has better performance.

In the next section, we introduce an improved quantum particle swarm optimization. In Section 3 we propose a new network traffic prediction based on chaos particle swarm optimization SVM. In Section 4, we test the performance of different network traffic prediction model. In Section 5 we conclude the paper and give some remarks.

2. An Improved Quantum Particle Swarm Optimization

2.1. Quantum Particle Swarm Optimization

In quantum particle swarm optimization, particle can searches for global optimal solution in the feasible solution space. The algorithm has small parameters and is easy to control. The state of particle is represented by position vector and each particle must converge to its own random point PP_i, $PP_i = (PP_{i1}, PP_{i2}, \ldots, PP_{id})$. Particles move according to the following three equation.

$$\begin{aligned} mbest(k+1) &= \frac{1}{N}\sum_{i=1}^{N} P_i(k) \\ &= (\frac{1}{N}\sum_{i=1}^{N} P_{i1}(k), \frac{1}{N}\sum_{i=1}^{N} P_{i2}(k), \ldots, \frac{1}{N}\sum_{i=1}^{N} P_{id}(k)) \end{aligned} \quad (1)$$

$$PP_{ij}(k+1) = r \cdot P_{ij}(k) + (1-r) \cdot P_{gj}(k). \quad (2)$$

$$X_{ij}(k+1) = \begin{cases} PP_{ij}(k+1) + \beta(k) \cdot \left| mbest(k+1) - X_{ij}(k) \right| \cdot \ln\frac{1}{u}, & rand(0,1) \le 0.5 \\ PP_{ij}(k+1) - \beta(k) \cdot \left| mbest(k+1) - X_{ij}(k) \right| \cdot \ln\frac{1}{u}, & rand(0,1) > 0.5 \end{cases}. \quad (3)$$

$j = 1, 2, \cdots, d$, $r = rand(0,1)$, $u = rand(0,1)$. N is the number of particle and d is dimension of particle. $mbest(k+1)$ is average position of particle individual optimal position $pbest(k)$ in the k-th iteration. $P_{ij}(k)$ is the j-th dimension position of the i-th particle in the k-th iteration. $P_{gj}(k)$ is the j-th dimension global optimal position in the k-th iteration. $PP_{ij}(k+1)$ is random point between $P_{ij}(k)$ and $P_{gj}(k)$. $\beta(k)$ is contraction expansion coefficient, which can control the convergence speed of the algorithm.

$$\beta(k) = \begin{cases} 0.5 + 0.5 \cdot \frac{f_i(k) - f_{\min}(k)}{f_{avg}(k) - f_{\min}(k)}, & f_i(k) \le f_{avg}(k) \\ 1, & f_i(k) > f_{avg}(k) \end{cases}. \quad (4)$$

$f_i(k)$ is the fitness value of the i-th particle of the k-th iteration, $f_{\min}(k)$ is the optimal fitness of the k-th population, $f_{avg}(k)$ is the average fitness of the k-th population. The process of quantum particle swarm optimization is as follows.

Step 1. Iteration time $k = 0$ .Initialize the position vector of each particle in the swarm.

Step 2. Calculate fitness value f_i of each particle according to objective function.

Step 3. Update individual optimal fitness $Pbestfitness$ of each particle and individual optimal position P_i .

Step 4. Update global optimal fitness $Gbestfitness$ of each particle and individual optimal position P_g .

Step 5. Calculate the new position of each particle according to (1), (2) and (3).

Step 6. $k = k + 1$ and return to step2 to recalculate, until meet the stopping condition.

2.2. An Improved Particle Swarm Optimization Based on Chaos

In QPSO evolution process, each particle go on next searching by studying its individual optimal location and the current global optimal location. When each particle traps in local optimum, it can use other particles to jump out of local optimal. But when most of the particles are trapped in local optimal, algorithm stagnation phenomenon will occur. In multi-start PSO after each iteration for several times, it reserves the current particle swarm optimal position and all particles are initialized, in order to improve the diversity of population to expand the search space. But all particle swarm initialization will completely destroy the structure of the particle swarm, which will greatly slow down the rate of convergence of the algorithm. Huwang proposed an improved algorithm, which adjusted individual optimal value and global optimal value and make particle converge to the new position. It can experience new search path and area to find the new solution. In continuous populations, if it can not find a optimal solution, it begin to disturb individual optimal position of particle and the global optimal position and forces to change individual history optimal fitness and global optimal fitness of particle.

If $PIterCount > T_p$, then do individual extreme disturbance to reset each dimension of individual optimal position of particle.

$$P_{ij} = rand(0,1) \cdot (Xup(j) - Xdown(j)) + Xdown(j) \tag{5}$$

$PIterCount$ is stagnation steps of particle individual. T_p is threshold of stagnation steps of particle individual. $Xup(j)$ is upper limit of the j-th dimension of particle and $Xdown(j)$ is lower limit of the j-th dimension of particle. Then update history optimal fitness of particle $Pbestfitness(i) = f(P_{i1}, P_{i2}, \cdots, P_{id})$. If $GItercount > T_g$, then do global extreme disturbance to reset each dimension of global optimal position of particle.

$$P_{gj} = rand(0,1) \cdot (Xup(j) - Xdown(j)) + Xdown(j) \cdot \tag{6}$$

$GItercount$ is global optimal stagnation steps, and T_p is threshold of global optimal stagnation steps. Then update global optimal fitness $Gbestfitness(i) = f(P_{g1}, P_{g2}, \cdots, P_{gd})$.Group fitness variance is defined as (7).

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{f_i - f_{avg}}{f} \right)^2 . \tag{7}$$

N is the number of population and f is normalized scaling factor, which can limit the size of σ^2.

$$f = \begin{cases} \max\limits_{1\le i\le N}\{|f_i - f_{avg}|\}, & \max\limits_{1\le i\le N}\{|f_i - f_{avg}|\} > 1 \\ 1, & others \end{cases}. \tag{8}$$

σ^2 reflects convergence degree of all particles in the particle swarm. The smaller the σ^2, the particle swarm tend to be converge. Otherwise particle swarm is in random searching stage. With the iteration times increasing, individual fitness of particle swarm is more and more close, so σ^2 will be smaller and smaller. When $\sigma^2 < T$, algorithm will do local searching intensively.

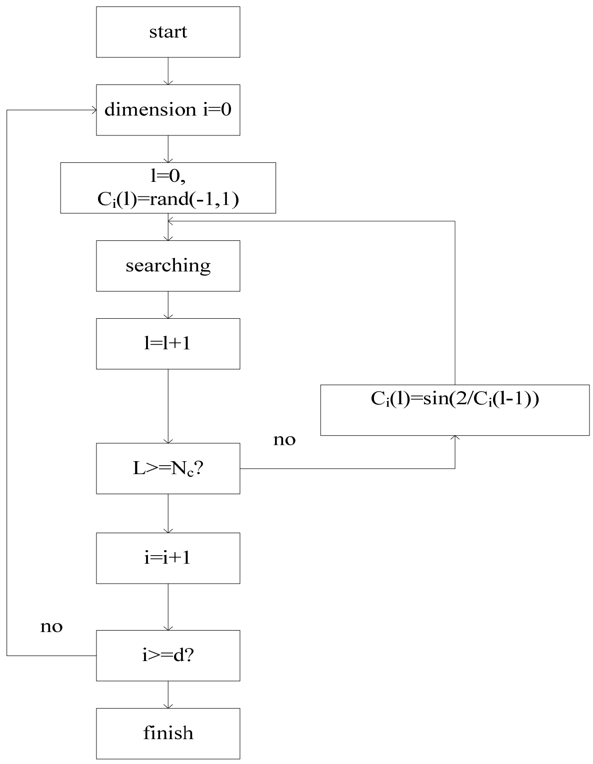

Figure 1. Chaos Searching Flow Chart

Using ergodicity, regularity, and randomness of chaos variable can optimize the searching process. If the particles find solution near global optimal solution, the chaos search can greatly enhance the local refined search ability of particle swarm. If particles trapped in local optimum, the chaos search can also help particles out of local optimum to a certain extent. In this paper, the chaotic search algorithm is aimed at each dimension of global optimal location P_g of quantum particle swarm optimization. The process of chaos searching is as follows and its flow chart is Figure 1.

Step 1. $i = 0$ and i is label of chaos searching of particle swarm.

Step 2. Do chaos searching to the i-th dimension of P_g.

(1) Iteration time $l = 0$ and chaos variable $C_i(l)$ belonging to $[-1,1]$ is generated randomly which does not include chaos fixed point.

(2) If $C_i(l) > 0$, (9) sets up. Otherwise (10) sets up.

$$P_i = \begin{cases} P_{gi} + P_{gi} \cdot C_i(l) & P_{gi} \cdot 2 < Xup(i) \\ P_{gi} + (Xup(i) - P_{gi}) \cdot C_i(l) & other \end{cases}. \tag{9}$$

$$P_i = P_{gi} + (P_{gi} - Xdown(i)) \cdot C_i(l). \tag{10}$$

If $f(P_i) < Gbsetfitness$, $Gbsetfitness = f(P_i)$, $P_{gi} = P_i$.

(3) $l = l + 1$, $C_i(l) = \sin(2 / C_i(l-1))$.

(4) Repeat (2) and (3) until given maximum chaos iteration time N_c comes.

Step 3. $i = i + 1$

Step 4. Return to step 2 to recalculate until all dimensions of particle experience chaos searching.

The proposed new particle swarm algorithm based on chaos searching and extreme disturbance is as follows and its flow chart is Figure 2.

Step 1. Iteration time $k = 0$ and maximum chaos iteration time N_c is established. Position of each particle in the swarm is initialized.

Step 2. Calculate fitness of each particle according to objective function f_i.

Step 3. Update individual optimal fitness $Pbestfitness$ and individual optimal position P_i.

Step 4. According to $PIterCount$ determine whether the particle stops. If it stops, then do individual extreme disturbance. Otherwise turn to step 5.

Step 5. Update global optimal fitness $Gbestfitness$ and global optimal position P_g.

Step 6. According to $GItercount$ determine whether the swarm stops. If it stops, then do global extreme disturbance and turn to step 7.

Step 7. Calculate swarm fitness variance σ^2. If $\sigma^2 < T$, do chaos searching and turn to step 8.

Step 8. Calculate new position of each particle according to (1), (2) and (3).

Step 9. $k = k + 1$ and return to step 2 to recalculate until the terminal condition is met.

3. A New Network Traffic Prediction Scheme Based on Chaos Particle Swarm SVM

Parameters selection of SVM is a kind of combinatorial optimization problem and is the search for an optimal solution in search space. Parameter optimization process of SVM based on intelligent algorithm is as follows and its flow chart is Figure 3.

Step 1. Within given parameter range, produce N number of particles randomly, which is N groups of SVM parameters (ε, C, γ). Use real coding so that we need randomly initialize within the area of solution.

Step 2. Calculate fitness of each particle.

(1) For each particle training set (x_i, y_i), with n number of samples is divided into k number of subset $G_1, G_2, \cdots, G_k$, $i = 1, 2, \cdots, n$.

(2) G_i groups of samples are used to check and other subsets are used to train SVM. Error is calculated by (11). y_j is the actual value and $\hat{y}_j$ is output value.

$$E_{G_i} = \sum_{y_j \in G_i} [y_j - \hat{y}_j]^2. \tag{11}$$

(3) repeat (2) from G_1 until k groups of data are checked.

(4) Calculate fitness of particle using (12).

$$fitness = \frac{1}{n}\sum_{i=1}^{k} E_{G_i} \ . \qquad (12)$$

Step 3. According to the process of improved particle swarm optimization based on chaos, go on the iteration until meeting the termination condition.

Step 4. output optimal solution (ε, C, γ).

Real network traffic shows not only short correlation and long correlation, but also has a periodic feature. To forecast the network traffic, the key point is to extract and separate different ingredients of the network traffic and set up model according to different characteristics to simplify complex issues. Considering sequence after the α Trous wavelet transform of the sequence can establish direct contact at the time point of each time scale, which has the time shift invariance and better generalization ability of support vector machine (SVM). This paper proposes a network traffic prediction model based on wavelet transformation and optimized SVM.

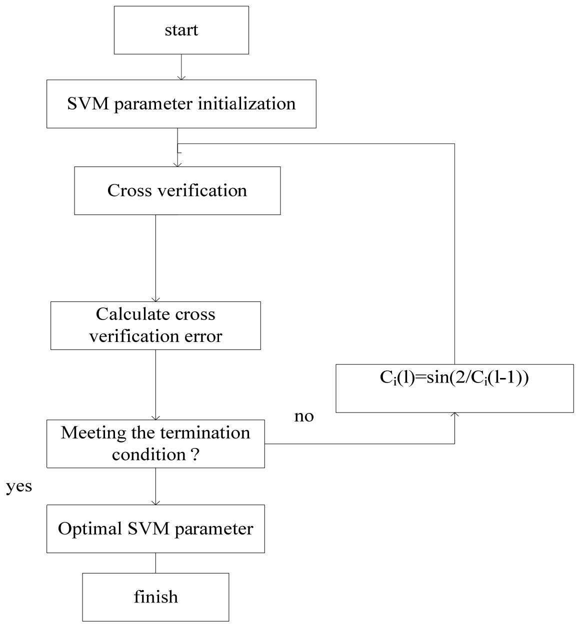

Figure 2. Parameter Optimization Process of Proposed Scheme

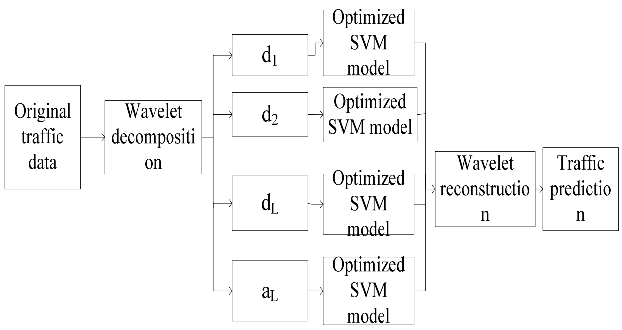

Figure 3. Architecture of Network Traffic Prediction

Architecture of network traffic prediction is shown in Figure 3. The detailed prediction process is as follows.

Step 1. Wavelet decomposition and reconstruction. The choice of wavelet base and wavelet decomposition series have an impact on forecasting accuracy. Wavelet decomposition series L is neither too small nor too big. Too small L can't effectively isolate different frequency characteristics of the network traffic. Too big L can result in model prediction error accumulated to the final forecasting result, which lower prediction accuracy and also can increase the computational complexity. So we should choose suitable wavelet base and decomposition series to decompose network traffic data into wavelet coefficients $d_1, d_2, \cdots, d_L$ and scale coefficient a_L.

Step 2. Data processing. Due to the bigger change range of the data, in order to improve the prediction accuracy, each signal component is normalized.

$$\hat{x} = \frac{x - \min(x_i)}{\max(x_i) - \min(x_i)}. \tag{13}$$

After normalization $\hat{x} \in [0,1]$.

Step 3. Model initialization. Determine the training set and test set, according to minimum cross validation error criterion choose a suitable embedding dimension m for each coefficient component. Input vector and output vector of SVM prediction model are obtained according to embedding dimension m.

Step 4. Determination of SVM model. The processed signal through wavelet transform is approximately smooth in the high frequency part. This portion of the signal can be predicted using the traditional linear model. But the complexity of network traffic requires constantly adjusting the model parameters in order to adapt to the changing of the flow condition. Parameter optimization of SVM is based on improved chaos particle swarm.

Step 5. SVM training and prediction. We adopt minimum sequence algorithm.

Step 6. Each prediction result is normalized inversely according to (14).

$$x = \hat{x} \cdot [\max(x_i) - \min(x_i)] + \min(x_i). \tag{14}$$

Then do wavelet reconstruction to get the final prediction of network traffic.

4. Experiment Results and Analysis

4.1. Rough Time Granularity Network Traffic Prediction

Experimental platform is matlab7.0. In order to evaluate prediction performance, this paper we choose root mean square error as performance index.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{15}$$

y_i is the real value and $\hat{y}_i$ is prediction value. The smaller the $RMSE$, the better the prediction performance.

Rough time granularity network traffic data comes from http://newsfeed.ntcu.net/~news/2006, which collected a total of 43 days network traffic per hour of primary node router. There are 1032 rough time granularity data. 240 data of the first 10 days is taken as training set and 792 data of the last 33days is taken as testing set. Basic parameter of chaos particle swarm optimization SVM is shown in Table 1 and Table 2.

Table 1. Parameter of Chaos Particle Swarm Optimization SVM

Coefficient component	Population size	Maxiter	N_c	T_p	T_g
Scale coefficient	20	12	10	4	5
Wavelet coefficient	20	12	10	4	5

Table 2. Parameter of Chaos Particle Swarm Optimization SVM

RBF core parameter γ	Sigmoid core parameter k	Sigmoid and Polynomial core parameter v	Polynomial core parameter d	c	ε
[0.01,3]	[0.001,1]	[0,1]	2	[0.1,1000]	[0.001,0.1]
[0.01,3]	[0.001,1]	[0,1]	2	[0.1,1000]	[0.006,0.1]

Table 3. Model Prediction Performance under Different Wavelet Decomposition Series

Decomposition serial	1	2	3	4	5
RMSE	10.3625	5.8964	3.5903	3.2386	3.1537

Table 4. Model Prediction Performance under Different Wavelet Base

Wavelet	RMSE				
base	a3	d1	d2	d3	combination
haar	2.4075	7.5996	4.5394	3.0675	6.6557
db3	1.3444	3.6425	3.1063	1.4526	3.5903
db4	1.0255	3.3561	2.8812	1.2347	3.2265
db6	0.6569	2.9786	2.5573	0.8569	2.5379
db8	0.1163	2.5255	1.7973	0.5846	2.1383

Table 5. Embedding Dimension of Each Signal Component

Signal component	a3	d1	d2	d3
Embedding dimension m	8	10	8	6

Table 3 is model prediction performance under different wavelet decomposition serials. When decomposition becomes from 1 level to 3 level, RMSE decreases quickly and later RMSE decreases slowly. Considering the computation complexity and prediction accuracy, the traffic data is decomposed into 3 levels. Table 4 gives the model prediction performance under different wavelet base. We can see that prediction performance of db8 wavelet base is better than other wavelet base. So in this paper, we adopt db8 wavelet base to decompose network traffic data into 3 levels. Then we obtain corresponding wavelet coefficient d1, d2, d3 and scale coefficient a3. Table 5 is embedding dimension of each signal component. Table 6 is prediction performance contrast under different core function after db8 wavelet decomposition. We can see that Linear core SVM is better than other core SVM and in scale coefficient layer, RBF core SVM has better performance. We compare the performance of proposed algorithm with other prediction models including SVM model, model based on wavelet transformation and BP naming WaBPNN, WFIRNN and WaSVM using standard particle swarm optimization to optimize its parameters. Input node of BP of each signal component is the same with embedding dimension m. Number of hidden layer node is 12. Order of FIR is 12×2 and number of output node is 1. We do 20 times experiment to acquire mean value of RMSE. Table 7 is SVM model optimization parameter of each layer. Table 8 is prediction performance contrast of different model.

Table 6. Model Prediction Performance under Different Core Function

Kernel	RMSE			
	d1	d2	d3	d4
Linear	2.2594	1.2693	0.5605	0.2185
RBF	2.5255	1.7973	0.5846	0.1163
Sigmoid	2.4742	1.8569	0.5911	0.2531
Polynomial	2.6367	1.7281	0.6125	0.2593

Table 7. SVM Model Optimization Parameter of Each Layer

Coefficient component	Chaos searching threshold T	SVM parameter		
		γ	C	ε
a3	2e-7	0.321	544.949	0.0013
d1	1e-7	×	837.359	0.0176
d2	3e-7	×	279.916	0.0169
d3	5e-7	×	292.234	0.0131

Table 8. Prediction Performance Comparison of Different Model

Model	RMSE				
	a3	d1	d2	d3	combination
SVM	×	×	×	×	14.1358
WaBPNN	0.7639	6.9036	2.3227	0.9216	8.2274
WFIRNN	0.3512	3.8963	1.8624	0.7282	3.6329
WaSVM	0.1532	2.7195	1.5328	0.6571	2.1915
ChOSVM	0.1163	2.2594	1.2693	0.5605	1.6954

It can be seen that prediction of SVM model is the largest. Wavelet neural network is easy to trap into local optimization. In a word, prediction accuracy of ChOSVM is better than other models.

4.2. Fine Time Granularity Network Traffic Prediction

Table 9. Embedding Dimension of Each Signal Component

Signal component	a2	d1	d2
Embedding dimension m	7	10	8

Table 10. Prediction Performance Comparison of Different Model

Kernel	RMSE		
	d1	d2	a2
Linear	1.9535	1.0368	0.8601
RBF	2.4641	1.2084	0.8517
Sigmoid	2.5394	1.2668	0.8823
Polynomial	2.3898	1.3391	0.9065

Table 11. SVM Model Optimization Parameter of Each Layer

Coefficient component	Chaos searching threshold T	SVM parameter		
		γ	C	ε
a2	5e-8	0.689	570.035	0.0021
d1	3e-7	×	540.869	0.0172
d2	5e-8	×	634.083	0.0151

Table 12. Prediction Performance Comparison of Different Model

Model	RMSE			
	A2	d1	d2	combination
SVM	×	×	×	16.4367
WaBPNN	2.1861	8,8132	3.7541	7.2794
WFIRNN	1.1861	5.9354	2.1962	4.1478
WaSVM	0.9065	2.4612	1.2967	1.9763
ChOSVM	0.8586	1.9535	1.0368	1.5958

Fine time granularity network traffic data comes from http://ita.ee.lbl.gov/html/contrib/BC.html. We choose BC-Oct89Ext data set. Traffic data is transformed into 500 traffic data. The first 200 data is taken as training set and the last 300 data is taken as testing set. We adopt db8 wavelet base to decompose traffic data to 2 level. Embedding dimension of each signal component is shown in Table 9. Table 10 is prediction performance contrast under different core function after db8 wavelet decomposition. We can see that Linear core SVM is better than other core SVM and in scale coefficient layer, RBF core SVM has better performance. SVM model optimization parameter is shown in Table 11 and prediction performance comparison of different model is shown in Table 12. We compare the performance of proposed algorithm with other prediction model including SVM model. It can be concluded that prediction accuracy of ChOSVM is better than other models.

5. Conclusion

Network traffic shows obvious multi-scale characteristic, which is composed of different signals and different components have different inherent law. Based on the complex characteristics of network traffic, this paper puts forward a network prediction model based on wavelet decomposition and scale coefficient. The results show that the optimized SVM has better generalization ability, ChOSVM can achieve ideal prediction accuracy only using a small number of training samples and its performance is obviously better than the single SVM model and some existing combination forecasting model. It has good robustness and strong generalization ability and high prediction accuracy.

References

[1] Xiaoling Tan, Weijian Fang, Yong Qu. Network Traffic Prediction Algorithm based on Wavelet Transform. International Journal of Advancements in Computing Technology. 2013; 5(5): 183-190.

[2] A Scherrer, N Larrieu, P Borgnat, P Owezarski, P Abry. *Non-gaussian and long-memory statistical mod-eling of Internet traffic*. 4th International Workshop on Internet Performance, Simulation, Monitoring and Mea-surements. 2006.

[3] Y Gao, G He, JC Hou. *On leveraging trafficpredictability in active queue management.* Proc. IEEE INFOCOM. 2002.

[4] G He, Y Gao, JC Hou, K Park. A *case for ex-ploiting self-similarity of Internet traffic in TCP congestion control*. Proc. IEEE ICNP. 2002.

[5] H Tong, C Li, J He. *A boosting-based frameworkfor self-similar and non-linear Internet traffic prediction*. Proc. of International Symposium on Neural Networks (ISNN). 2004.

[6] Y Xinyu, Z Ming, Z Rui, S Yi. *A novel LMS method for real-time network traffic prediction*. Proc. of Computational Science and Its Applications (ICCSA). 2004.

[7] F Rouai, M Ahmed. A *new approach for fuzzy neural network weight initialization*. INNS-IEEE International Joint Conference on Neural Network (IJCNN). 2001; 2.

[8] N Sadek, A Khotanzad. *Dynamic bandwidth al- location using a two-stage fuzzy neural network based traffic predictor*. Proceedings of IEEE International Joint Conference on Neural Networks. 2004; 3: 2407–2412.

[9] AM Yang, XM Sun, CY Li, P Liu. *A neuro-fuzzy method of forecasting the network traffic of ac-cessing web server*. 2nd International Conference on Fuzzy Systems and Knowledge Discovery, Lecture Notes in Computer Science. 2005; 3613

[10] M Çınar, M Engin, EZ Engin, YZ Ateşçi. Early Prostate Cancer Diagnosis by Using Artificial Neural Networks and Support Vector Machines. *Expert Sys-tems with Applications*. 2009; 36(3): 6357- 6361.

[11] DC Park. Structure Optimization of Bi-Linear Recur-rent Neural Networks and its Application to Ethernet Network Traffic Prediction. *Information Sciences*. 2009.

[12] BR Chang, HF Tsai. Novel Hybrid Approach to Data-Packet-Flow Prediction for Improving Network Traffic Analysis. *Applied Soft Computing*. 2009; 9(3): 1177-1183.

[13] R Yunhua. Evaluation and Estimation of Second-Order Self-Similar Network Traffic. *Computer Communications*. 2004; 27(9): 898-904.

[14] BC Lu, ML Huang. *Traffic Flow Prediction Based on Wavelet Analysis, Genetic Algorithm and Artificial Neural Network*. Proc. of the International Conference on Information Engineering and Computer Science (ICIECS), Wuhan.2009: 1–4.

[15] XY Yang, SS Yang, J Li. A flooding-based DDoS detection algorithm based on non-linear preprocessing network traffic predicted method. *Chinese journal of computers*. 2011; 34(2): 395–405.

[16] M Jiang, et al. Research on the comparison of time series models for network traffic prediction. *Acta Electronic Sinica*. 2009; 37(11): 2353–2358.

[17] XT Chen, JX Liu. Network traffic prediction based on wavelet transformation and FARIMA. *Journal on communications*. 2011; 32(4): 153–157.

[18] HL Sun, YH Jin, YD Cui, SD Cheng. Large-time scale network traffic short-term prediction by grey model. *Journal of Beijing University of Posts and Telecommunications*. 2010; 33(1): 7–11.

[19] L Zhu, L Qin, KY Xue, XY Zhang. *A Novel BP Neural Network Model for Traffic Prediction of Next Generation Network.* Proc. of the Fifth International Conference on Natural Computation (ICNC 09), Tianjian. 2009; 32–38.
[20] QF Yao, CF Li, HL Ma, S Zhang. Novel network traffic forecasting algorithm based on grey model and Markov chain. *Journal of Zhejiang University (science ed.).* 2007; 34(4): 396–400.
[21] P Wang, SY Zhang, XJ Chen. *SFARIMA: A New Network Traffic Prediction Algorithm*. Proc. of the 1st International Conference on Information Science and Engineering (ICISE), Nanjing. 2009; 1859–1863.
[22] YM Mao, SY Shi. *Research on Method of the Subsection Learning of Double-Layers BP Neural Network in Prediction of Traffic Volume*. Proc. of the International Conference on Measuring Technology and Mechatronics Automation (ICMTMA 09), Zhangjiajie. 2009; 294–297.
[23] M Dusi, A Este, F Gringoli, L Salgarelli. *Using GMM and SVM-Based Techniques for the Classification of SSH-Encrypted Traffic.* IEEE International Conference on Communications, Dresden. 2009: 1– 6.
[24] CJ Gu, SY Zhang. Network traffic classification based on improved support vector machine. *Chinese Journal of Scientific Instrument*. 2011; 32(7): 1507-1513.
[25] E Alice, G Francesco, S Luca. Support Vector Machines for TCP traffic classification. *The International Journal of Computer and Telecommunications Networking*. 2009; 53(14): 2476-2490.
[26] Shu Fan, Fengchun Tian, Qinghua He, Pengfei Jia, Jingwei Feng, Yue Shen. Wound Infection Recognition Based on Quantum-Behaved Particle Swarm Optimization. *Journal of Convergence Information Technology*. 2012; 8(5): 1209-1219.
[27] Youxin LUO, Xiaoyi CHE, Lingfang LI. Chaotic Artificial Bee Colony Hybrid Discrete Constrained Optimization Algorithm. *Journal of Convergence Information Technology.* 2012; 7(22): 484-491.
[28] J Sun, B Feng, WB Xu. *Particle Swarm Optimization with Particles Having Quantum Behavior.* Proc. Congress on Evolutionary Computation. 2004; 325-331.