ISSN : 0974 - 7435 Volume 10 Issue 23

2014

BioTechnology
An Indian Journal

FULL PAPER
BTAIJ, 10(23), 2014 [14693-14703]

Network traffic forecasting of ideological & political websites based on EMD_CPSO modified LS-SVM

Xia Xin, Li Hao
Agricultural University of Hebei West Campus of modern science and Technology College, (CHIAN)

ABSTRACT

In order to enhance the accuracy of network traffic forecasting of ideological and political websites, in pertinence to the nonlinear and non-stable property of network traffic data, this paper introduces the chaos theory into PSO for optimization selection of the nuclear parameters and penalty coefficient of LS-SVM, and proposes an EMD_CPSO optimized LS-SVM (ECLS-SVM) network traffic forecasting model. It begins with EMD decomposition of the time sequence of network traffic data to extract detail features and trend features of network traffic data. Next, with the extracted network traffic features as the input into the CPSO optimized LS-SVM model, a network traffic forecasting model is built on the base of EMD_CPSO optimized LS-SVM. Finally, the network traffic data are utilized for a simulation experiment. By comparing the ECLS-SVM algorithm based single-step, 3-step, 5-step and 7-step forecasting results and the forecast time and mean square errors among different models, it is seen that the ECLS-SVM algorithm can effectively boost both accuracy and efficiency of network traffic forecasting, which attaches vital theoretical significance and practical value to instructing rational distribution and planning of network resources.

KEYWORDS

Empirical mode decomposition (EMD); Chaos theory; Particle swarm optimization (PSO); Least square SVM; Network traffic.

© Trade Science Inc.

INTRODUCTION

With the rapid development in internet technology, continuous increase in flow and bandwidth, as well as growing size and complexity of network, the onerousness of network administration is surging dramatically with frequent occurrences of network failure. In order to ensure reliable transmission of internet data and rational distribution of resources, high-quality network traffic forecasting attaches vital theoretical significance and practical value to network planning, administration and designing.

Lei Ting et al.[1] combined the wavelet transform theory and neural network technique together to put forward a wavelet neural network based network traffic forecasting model, which extracted the detail features and overall features of network traffic via wavelet transform. With the extracted features of network traffic as the input into, while the actual network traffic as the output from, the neural network, the simulation result evinces higher forecasting accuracy, except for its stability that remains to be improved further.

Against the local optimum problem of BP neural network, Wen Guo et al.[2] took the advantage of PSO in global optimization to modify the BP network and applied this algorithm to network traffic forecasting. The forecasting accuracy and convergence rate of the model after being modified were superior to traditional BP neural networks, except for the deficiency that PSO was open to the premature problem and prone to running into local optimum.

Liu Yuan et al.[3] implemented network modification by introducing the chaos theory into wavelet neural network. The simulation result illustrates the forecasting error of chaotic wavelet neural network is far smaller than the network traffic forecasting of RBF neural network, except for higher difficulty in selecting and determining wavelet basis function.

Yang Guang et al.[4] proposed a wavelet kernel LS-SVM based network traffic forecasting in combination with the multi-resolution property of wavelet kernel function. Through forecasting and verification of actual data, this method is attested with certain superiority except for the selection of parameters which need to be determined empirically.

Specific to the non-stable and nonlinear characteristics of network traffic data, this paper implements self-adaptive optimization to the kernel parameters and penalty coefficient in LS-SVM in combination with EMD and CPSO algorithms, proposing an ECLS-SVM algorithm based network traffic forecasting model. Via EMD, the detail features and trend features of network traffic are extracted, and both input and output of ECLS-SVM based network traffic forecasting model are constructed to implement the forecasting of network traffic, which lays a decision-making basis for rational configuration and reliable transmission of network resources.

CHAOTIC PSO

PSO algorithm

First put forward by Kennedy et al.[5-6], PSO Algorithm was used to simulate the flying and foraging behavior of bird flocks and achieve optimization of foraging path through the synergistic mutual help and competition between bird flocks. In the foraging process, the particles will trace two extrema of the particle swarm, namely the optimum solution $pbest$ pinpointed from the particle itself up to the current state, and the optimum solution $gbest$ pinpointed from the whole species up to the current state.

The velocity and position of the particle can be updated by Formula (1) and Formula (2)[7]:

$$v_{id}(t+1) = v_{id}(t) + c_1 rand_1 * \left(p_{pid} - x_{id}(t)\right) + c_2 rand_2 * \left(p_{gd} - x_{id}(t)\right) \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2)$$

In Formula (1) and Formula (2), x_i and v_i denote the current position and velocity of the particle, respectively ($i = 1,2,\ldots,m$; $d = 1,2,\ldots,n$); $rand_1$ and $rand_2$ are random numbers between 0 and 1; c_1 and c_2 are learning factors ($c_1, c_2 > 0$).

Chaotic PSO (CPSO)

Against the premature problem existing in traditional PSO Algorithm, the chaos theory is introduced into PSO for modification. The detailed process of the algorithm is presented as below[8]:

(1) Chaos initialization. Assume the optimization variable is D-dimensional. Generate a D-dimensional vector $z_1 = [z_{11}, z_{12}, \ldots, z_{1D}]$ at random, with all components ranging between 0 and 1. Derive M components, $z_1, z_2, \ldots, z_M$, according to Logistic Equation

$$z_{n+1} = \mu z_n (1 - z_n), n = 0,1,2,\ldots; \quad 0 < z_n < 1; \mu \in [0,4] \quad (3)$$

Employ Formula (4) to implement the mapping from chaos interval to variable range:

$$x_{ij} = a_j + (b_j - a_j)z_{ij} \quad (4)$$

In Formula (4), b_j and a_j denote the upper limit and lower limit of the optimization variable required, respectively.

(2) Evaluate the fitness function value of each particle according to fitness function. Select N out of M initial particle swarms as the initial solution. The velocity of particles is generated at random.

(3) Set the initial individual extremum and global extremum. Set the current position of each particle as individual extremum P_i. Evaluate the fitness function value of each individual extremum P_i according to fitness function. Select the position of the particle with the optimum value which is defined as the global extremum P_g.

(4) Implement update of flying velocity and position of particles according to Formula (1) and Formula (2).

(5) Chaotically optimizing the optimum position P_g. Begin by mapping the optimum position into the value range, $[0,1]$, of Logistic Equation. Next, generate m sequences of chaotic variables according to Logistic Equation. Finally, map the generated sequences of chaotic variables into optimization variables to get m particles. Evaluate the fitness function value of each particle to derive the optimum solution p'.

$$z_9 = \frac{P_g - a_i}{b_i - a_i} \quad (5)$$

(6) Use the optimum solution p' to update the position of a random particle among those in the current search swarm.

(7) Skip to Step (4). Where the terminal condition of the particle swarm is satisfied, output the optimum result; otherwise, continue the iteration.

EMPIRICAL MODE DECOMPOSITION (EMD)

EMD is a self-adaptive signal decomposition approach which absorbs the advantage of multi-resolution of wavelet transform while overcoming the deficiency that the wavelet basis is difficult to select and that the decomposition dimension is difficult to determine. Therefore EMD is highly adaptable for decomposition of nonlinear and non-stable signals.

The detailed EMD decomposition process of time sequence of network traffic is presented as below[9]:

(1) Identify all maximum point(s) among the network traffic data, meanwhile perform fitting calculation of the upper envelop line $e_{up}(t)$;

(2) Extract the minimum point(s) among the network traffic data, meanwhile perform fitting calculation of the lower envelop line $e_{low}(t)$. On the base of both upper and lower envelop lines, compute the mean, $m_1(t)$, of both;

$$m_1(t) = \frac{e_{up}(t) + e_{low}(t)}{2} \quad (6)$$

(3) Subtract $m_1(t)$ from the sequence of network traffic data, $x(t)$, to get $h_1(t)$. Treat $h_1(t)$ as a new sequence of network traffic data, $x(t)$. Repeat Step 1. Perform k times of screening until $h_1(t) = x(t) - m_1(t)$ satisfies the IMF condition. Let $c_1(t) = h_1(t)$, then $c_1(t)$ is an IMF1 component of network traffic data.

Iterate the above procedure. Ultimately the sequence of network traffic data, $x(t)$, can be decomposed as:

$$x(t) = \sum_{i=1}^{N} c_i(t) + r_N(t) \quad (7)$$

ECLS-SVM NETWORK TRAFFIC FORECASTING MODEL

LS-SVM SVM

The LS-SVM proposed by Suykens can be converted into[10]:

$$Min\ J(\omega, \xi) = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N}\xi_k^2 \quad s.t.\ y_k = \phi(x_k)\omega^T + b + \xi_k \quad (8)$$

In Formula (8) $\xi_k \geq 0$, $k = 1, 2, \cdots, N$, C is a penalty factor.

Adopt the Lagrangian method to compute Formula (8) and convert it into Formula (9):

$$L(\omega, b, \xi, \alpha) = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{N}\xi_k^2 - \sum_{i=1}^{N}\alpha_k[(\omega^T\phi(x_k) + b + \xi_k) - y_k] \quad (9)$$

In Formula (9), α_k ($k = 1, 2, \cdots, N$) denotes a Lagrangian multiplier.

Find the respective partial derivative of ω, b, ξ and α, and let it be zero, to get:

$$\omega = \sum_{k=1}^{N}\alpha_k\phi(x_k) \quad \sum_{k=1}^{N}\alpha_k = 0, \quad \alpha_k = C\xi_k \quad \omega^T\phi(x_k) + b + \xi_k - y_k = 0 \quad (10)$$

According to the $Mercer$ condition and Formula (11), the kernel function $k(x_i, x_j)$ can be expressed as:

$$k(x_i, x_j) = \phi(x_i)\phi(x_j) \quad (11)$$

Since the RBM kernel function outperforms typical kernel functions, this paper uses it for forecasting, by the following formula:

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \quad (12)$$

Thus the LS-SVM network forecasting model is:

$$f(x) = \sum_{i=1}^{m}\alpha_i \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) + b \quad (13)$$

Formula (13) reveals the performance of LS-SVM is mainly susceptible to γ and σ. In order to implement their self-adaptive selection, this paper uses the CPSO algorithm to implement self-adaptive optimization of γ and σ.

ECLS-SVM network forecasting model

Since the parameters required to be optimized for LS-SVM are γ and σ, the mathematical model to optimize them is:

$$F = \{\gamma\ ,\ \sigma\} \quad (14)$$

Through optimization, the self-adaptive selection of parameters γ and σ, whose fitness function can be defined, is implemented under the condition of ensuring optimization of forecasting accuracy. Assume the actual network traffic is $y(t)$, and the forecast network traffic is $\hat{y}(t)$, at time t. Then the difference $e(t)$ between the actual network traffic $y(t)$ and forecast network traffic $\hat{y}(t)$ can be expressed as the following formula:

$$e(t) = \hat{y}(t) - y(t) \quad (15)$$

Against the nonlinear problem in network traffic forecasting, with n as the sample size of actual network traffic data, CPSO is employed to optimize the kernel parameters and penalty coefficient of LS-SVM so that the quadratic sum of the difference of LS-SVM between the actual network traffic output and forecast network traffic achieves the minimum. The fitness function can be expressed as the following formula:

$$\min\ Fitness(t) = \frac{1}{2n}\sum_{t=1}^{T}[e(t)]^2 \quad (16)$$

Procedure of algorithm

The EMD-based CPSO-LS-SVM network traffic forecasting algorithm undergoes the following procedure, with the flowchart shown as Fig 1.

Step 1: Normalize the network traffic data;

Step 2: Employ EMD to decompose the normalized network traffic data, extract the detail features and trend features of network traffic data, and construct training samples and test samples;

Step 3: Set the population size $popsize$, learning factors c_1 and c_2, and maximum iterations $\max gen$ of CPSO algorithm;

Step 4: Input the constructed training samples into LS-SVM. Compute the fitness function value of particles according to fitness function Formula (19) to find the particle's individually and globally optimum position as well as the optimum value;

Step 5: Update the particle's velocity and position;

Step 6: Compute fitness while updating position and velocity;

Step 7: If $gen > \max gen$, save the optimum solution; else let $gen = gen + 1$ and skip to Step 4;

Step 8: Perform network traffic forecasting according to the optimum parameters γ and σ in correspondence to the optimum position.

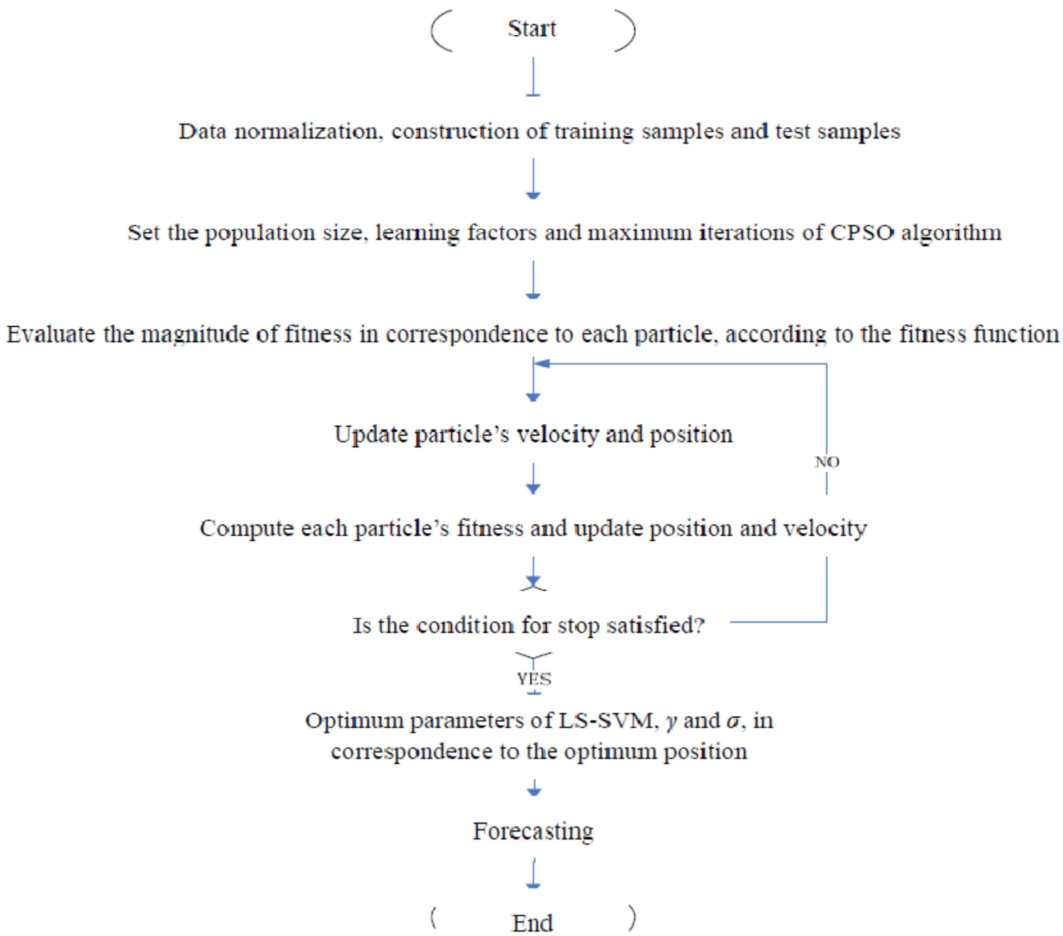
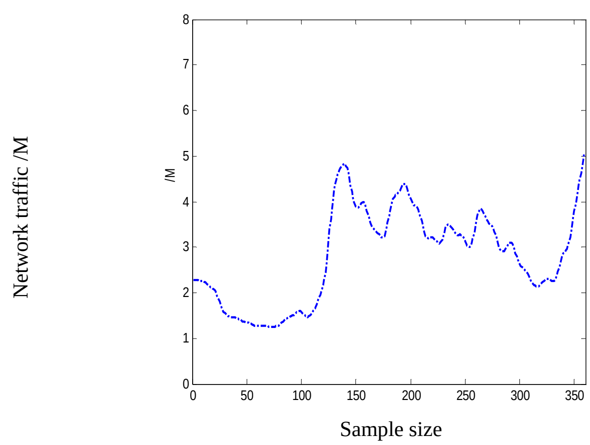


Fig 1 ECLS-SVM Based Forecasting Flowchart

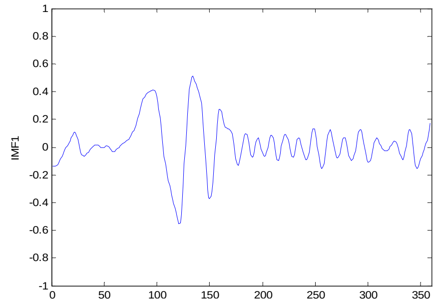
SIMULATION EXPERIMENT

Data source

The data in this paper originates from a library of flow. The data through 15 days in total from November 7th to 21st, 2014 are collected as the research object. The network access flow is collected on an hourly basis every day. A total number of 15*24=360 groups of data are collected. The primary data of network traffic are shown as Fig 2(a).

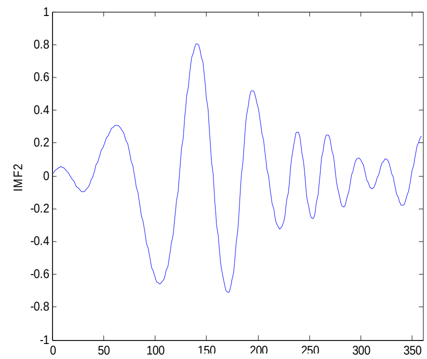


(a) Primitive Network Traffic



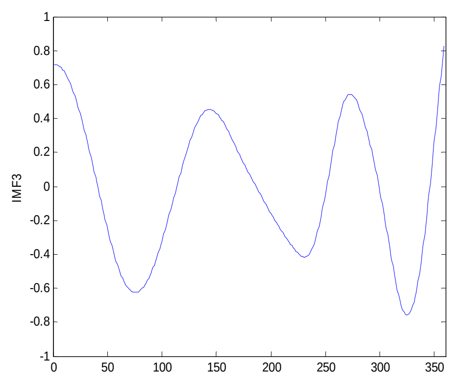
Sample size

(b) IMF1 Component



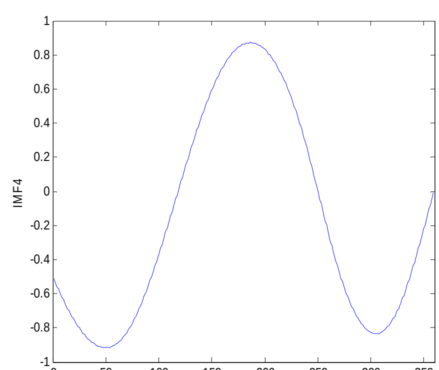
Sample size

(c) IMF2 Component



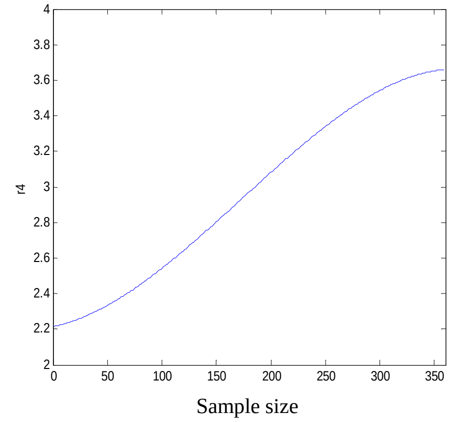
Sample size

(d) IMF3 Component



a Sample size

(e) IMF4 Component



(f) Trend Component

Fig 2 Primitive Network Traffic & Chart of EMD Decomposition Result

Data processing

Apply EMD decomposition to the primitive sequence of network traffic data to derive different IMF components in succession. Fig 2(b) through Fig 2(f) reveal the primitive network traffic data is decomposed into 4 components with lower fluctuations and 1 residual component. According to the analytic result of IMF components, CPSO is employed to optimize the model of kernel parameters and penalty coefficient of LS-SVM and implement network traffic forecasting.

Evaluation indicator

In order to verify the effectiveness of the algorithm in performing network traffic forecasting, this paper adopts mean square error as the indicator to evaluate the effectiveness of network traffic forecasting. The evaluation formula is shown as below:

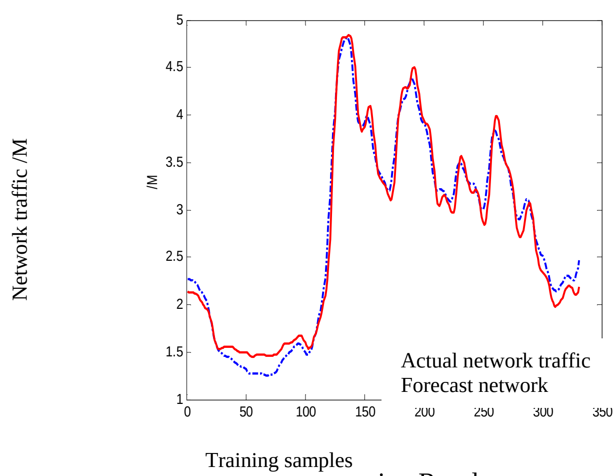
Mean square error:

$$MSE = \sqrt{\frac{1}{K}\sum_{i=1}^{K}(x_i - \hat{x}_i)^2} \qquad (17)$$

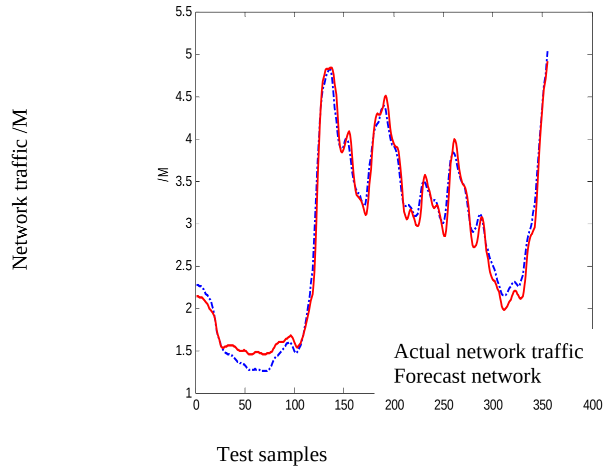
Where x_i and $\hat{x}_i$ denote the actual network traffic and forecast network traffic, respectively.

Experimental results

Separate the 360 groups of network traffic data into 336 groups of training samples and 24 groups of test samples, which are used to verify the results of forecasting. Set the maximum iterations of CPSO algorithm as 100, population size as 20, popmin=-5.12, popmax=5.12, vmax=1, and vmin=-1. The results of forecasting are shown in Fig 3, Fig 4, Fig 5 and Fig 6, which represent the single-step forecasting, 3-step forecasting, 5-step forecasting and 7-step forecasting, respectively.

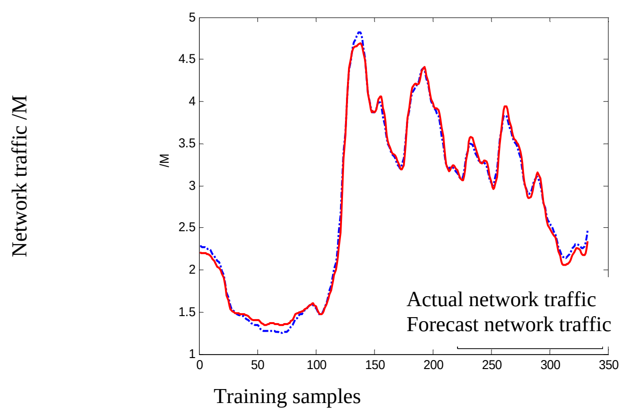


...ing Result

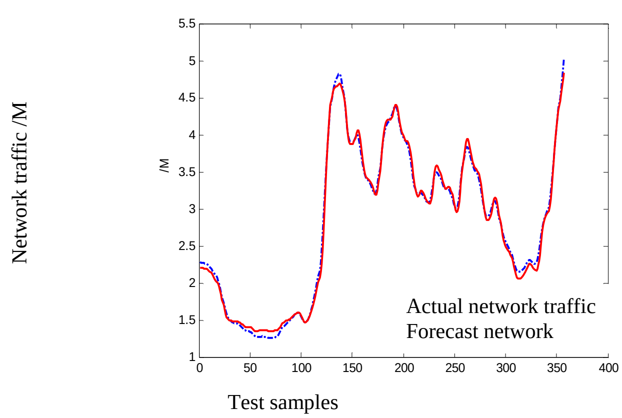


(b) Forecasting Result

Fig 3 ECLS-SVM Algorithm Based Single-step Network Traffic Forecasting

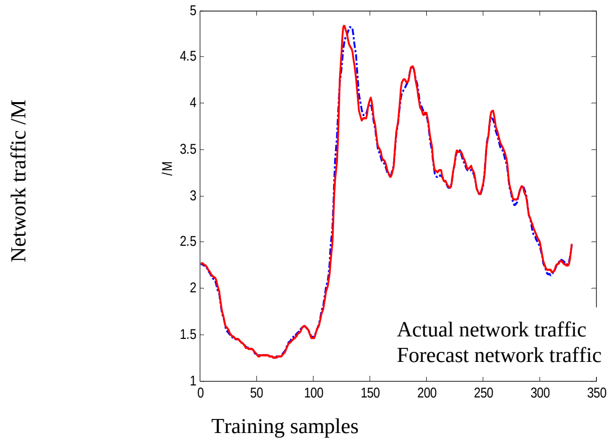


(a) Training Result

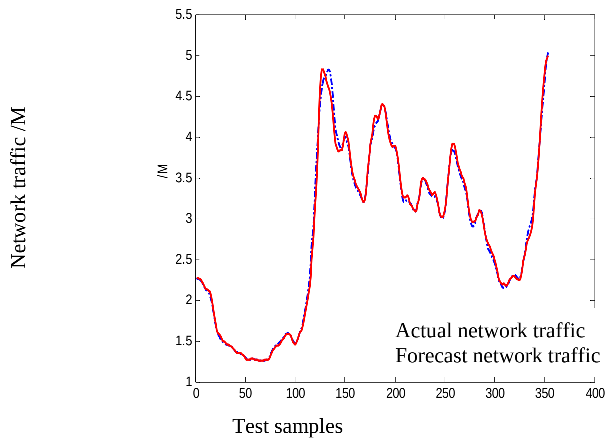


(b) Forecasting Result

Fig 4 ECLS-SVM Algorithm Based 3-step Network Traffic Forecasting

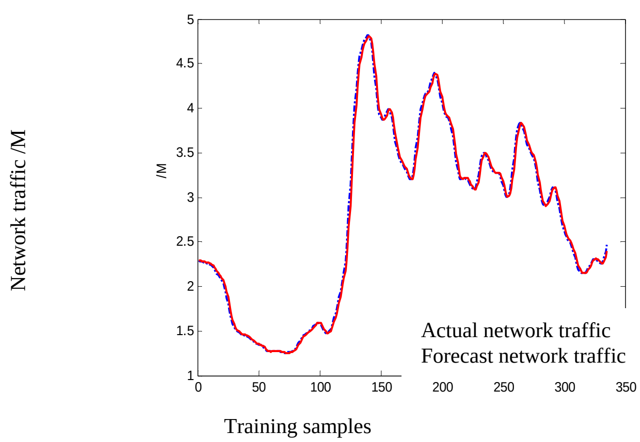


(a) Training Result

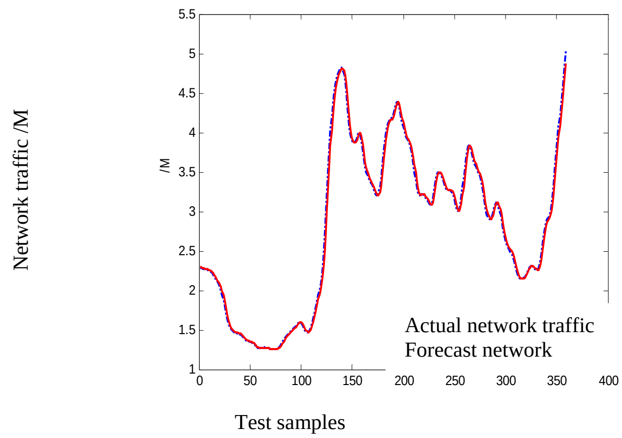


(b) Forecasting Result

Fig ECLS-SVM Algorithm Based 5-step Network Traffic Forecasting



(a) Training Result



(b) Forecasting Result

Fig 6 ECLS-SVM Algorithm Based 7-step Network Traffic Forecasting

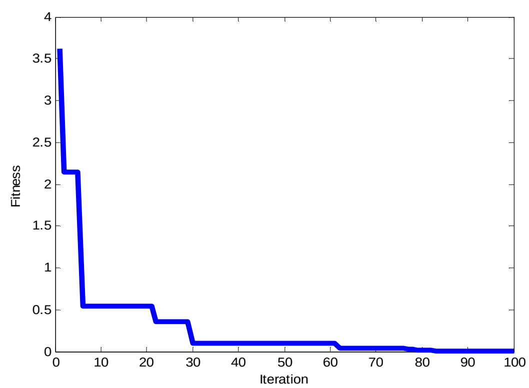


Fig 7 Fitness Curve Chart of CPSO Optimized LS-SVM

The single-step forecasting, 3-step forecasting, 5-step forecasting and 7-step forecasting result diagrams of ECLS-SVM reveal that, as the step size of forecasting increases, the forecasting accuracy of ECLS-SVM algorithm becomes higher and higher, achieving a good effect. Fig 7 shows the fitness curve of CPSO algorithm optimized LS-SVM.

To contrast the superiority of ECLS-SVM algorithm, the forecasting results of ECLS-SVM algorithm, CPSO-LS-SVM and LS-SVM are compared by running each 10 times. The result of comparison is shown as Table 1.

Table 1 Comparison of Forecast MSE among Three Models (ECLS-SVM Algorithm, CPSO-LS-SVM and LS-SVM)

Running Time(s)	LS-SVM	CPSO-LS-SVM	ECLS-SVM
1	0.0088	0.0060	0.0040
2	0.0072	0.0055	0.0046
3	0.0068	0.0050	0.0040
4	0.0084	0.0064	0.0042
5	0.0065	0.0056	0.0038
6	0.0074	0.0061	0.0043
7	0.0076	0.0056	0.0035
8	0.0064	0.0046	0.0042
9	0.0072	0.0076	0.0040
10	0.0067	0.0051	0.0047
Mean	0.0072	0.0058	0.0041

The comparative result of forecast MSE in Table 1 among the three models, ECLS-SVM algorithm, CPSO-LS-SVM and LS-SVM, reveals that the ECLS-SVM algorithm achieves the best effect of forecasting as superior to both CPSO-LS-SVM and LS-SVM models, and that CPSO-LS-SVM turns out superior to LS-SVM in forecasting effect.

Table 2 Comparison of Forecast Time among Three Models (ECLS-SVM Algorithm, CPSO-LS-SVM and LS-SVM; Unit/s)

Step Size of Forecasting	LS-SVM	CPSO-LS-SVM	ECLS-SVM
Single-step	111.40	97.36	37.21
3	97.60	89.22	34.25
5	86.33	74.22	32.18
7	79.45	65.80	31.27

The comparative result of forecast time in Table 2 among the three models, ECLS-SVM algorithm, CPSO-LS-SVM and LS-SVM, reveals that the shortest forecast time of ECLS-SVM algorithm is shorter than both CPSO-LS-SVM and LS-SVM models, whereas the forecast time of CPSO-LS-SVM is shorter than LS-SVM.

CONCLUSION

In pertinence to the randomness of selection of kernel parameters and penalty coefficient in LS-SVM, this paper employs the chaotic particle swarm optimization algorithm to optimize the kernel parameters and penalty coefficient in LS-SVM, meanwhile extracts the detail features and trend features of network traffic in combination with EMD, and constructs an ECLS-SVM based network traffic forecasting model to perform single-step, 3-step, 5-step and 7-step forecasting, respectively. From the forecast MSE and forecast time of different network traffic forecasting models, it is discovered that the ECLS-SVM algorithm is superior to the other models in both forecasting accuracy and efficiency, which thus lays a decision-making basis for rational distribution and reliable transmission of network resources.

REFERENCES

[1] Lei Ting, Yu Zhenwei. A Wavelet Neural Network Model for Network Traffic Prediction[J]. Computer Applications, 2012,26 (3): 526-528.

[2] Wen Guo, Yizheng Qiao, Haiyan Hou. BP Neural Network Optimized with PSO Algorithm and Its Application in Forecasting. Proceedings of the IEEE international Conference on Information Acquisition. 2011:617-621

[3] Liu Yuan, Dai Yue, Cao Jianhua. The Traffic Flow Chaotic Time Series Prediction Based on Wavelet Neural Network [J]. Computer Engineering, 2012,34 (16): 105-110

[4] Yang Guang, Zhang Guomei, Liu Xingyu. Prediction of Network Traffic Based on Wavelet Kernel LS-SVM [J]. Microcomputer Development, 2011,15 (12): 125-128

[5] Wang Xifeng, Gao Ling, Zhang Xiaoluan. Self Similarity Analysis and Research of Network Traffic Prediction [J]. Computer Technology and Development, 2010,17 (11): 42-45

[6] Feng Hailiang, Lin Qing, Chen Dideng. A Prediction Model of Combination of Network Traffic Based on Neural Network [J]. Computer Applications, 2010,26 (9): 2206-2208.

[7] Cristiano Cervellera, Aihong Wen, Victoria C.P. Chen. Neural network and regression spline value function approximations for stochastic dynamic programming[J]. Computer & Operations Research, 2012, 34(1): 70-90.

[8] Junfeng Li, Aiping Yang , Wenzhan Dai. Modeling Mechanism of Grey Neural Network and Its Application[C]. Proceedings of 2007 IEEE International Conference on Grey Systems and Intelligent Services, Nanjing, 2009: 404-408.

[9] Li Zongfu, Deng Qiongbo, Li Huan. Kohonen of SOFM Neural Network and Its Evolution Research. Computer Engineering and Design, 2013,25 (10): 1729-1830.

[10] H. Wang and Y. Liu. A Hybrid Particle Swarm Algorithm with Cauchy Mutation [C], in Proc IEEE Int. Conf. Swarm Intelligence Symposium (SIS), 2011, pp: 356-360

[11] H. Wang, Y. Liu. An improved particle swarm optimization with adaptive jumps [C]. In Proceedings of IEEE International Conference on Evolutionary Computation, 2012, 392-397