TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.4, April 2014, pp. 2833 ~ 2842
DOI: http://dx.doi.org/10.11591/telkomnika.v12i4.4736

A P2P Traffic Identification Approach Based on SVM and BFA

Chunzhi Wang, Zeqi Wang, Zhiwei Ye*, Hongwei Chen
Hubei University of Technology / School of Computer Science, Wuhan, China
*Corresponding author, e-mail: weizhiye121@163.com

Abstract

Nowadays new peer to peer (P2P) traffic with dynamic port and encrypted technology makes the identification of P2P traffic become more and more difficult. As one of the optimal classifiers, support vector machine (SVM) has special advantages with avoiding local optimum, overcoming dimension disaster, resolving small samples and high dimension for P2P classification problems. However, to employ SVM, the parameters selection of SVM should be considered and thus some optimization methods have been put forward to deal with it, still, it is not fully solved. Hence, in the paper, a peer to peer traffic identification approach based on support vector machine and bacterial foraging algorithm is proposed for better identification of P2P network traffic. First, the best parameters for SVM are tuned with bacterial foraging algorithm. Subsequently, SVM set with the best parameters is used to identify P2P traffic. Finally, experimental results show the proposed approach can effectively improve the accuracy of P2P network traffic identification.

Keywords: *P2P traffic identification, bacterial foraging algorithm, support vector machine*

Copyright © 2014 Institute of Advanced Engineering and Science. All rights reserved.

1. Introduction

Peer-to-peer (abbreviated to P2P) computer network is a distributed application architecture that partitions tasks or workloads among peers. Peers are equally privileged participants in the application. Each computer in the network is referred to as a node. The owner of each computer on a P2P network would set aside a portion of its resources - such as processing power, disk storage or network bandwidth -to be made directly available to other network participant, without the need for central coordination by servers or stable hosts [1]. In P2P networks, clients provide resources, which may include bandwidth, storage space, and computing power [1-3]. This property is one of the major advantages of using P2P networks because it makes the setup and running costs very small for the original content distributor. Another characteristic of a P2P network is its capability in terms of fault-tolerance. When a peer goes down or is disconnected from the network, the P2P application will continue by using other peers. With the widespread use of peer-to-peer (P2P) technologies, it has occupied the majority of the total Internet traffic in applications, such as communication, entertainment and sharing, for the advantages of convenience, high-speed, rich resources and no-center.

However, it brings challenges for content supervision and a lot of problems arise at the same time. Firstly, the decentralization and anonymity of P2P streaming network made informational resource more dispersion and concealment; thus some violent and blue movies or video were transmitted arbitrarily, which brought bad effects to people, juveniles especially; in addition, decentralized networks introduce new security issues because they are designed so that each user is responsible for controlling their data and resources. Secondly, harmful data can also be distributed on P2P networks by modifying files that are already being distributed on the network. Consequently, it is of great importance for accurate identification of traffic that is generated by P2P applications [4-5].

Traditionally, network traffic can be easily identified by detecting the port numbers of that traffic, as most of the traffic in the Internet uses standard port numbers [6]. Yet, as many newly-emerged P2P applications using dynamic port numbers, masquerading techniques, and payload encryption to avoid detection, the classical approaches based on port mapping and payload analysis are ineffective. Thus the task of identifying P2P traffic is becoming more

Received September 3, 2013; Revised October 25, 2013; Accepted November 20, 2013

challenging and a lot of work has been done to effectively identify P2P data traffic, which is mainly divided into four categories.

(1) Port number based approaches, which were the first techniques to detect P2P traffic. In the early emergence of P2P, the port-based method is successful because many well-known applications have specific port numbers (assigned by IANA). However, in order to circumvent detection, fixed port numbers are unreliable. Therefore, more and more P2P applications never use port-based methods any more [7].

(2) Deep Packet Inspection (DPI) method, which tries to judge whether it has found known P2P characteristics through byte-by-byte scan of message contents for data traffics. DPI technologies have a high degree of exactness and they can judge specific application types, but they execute in a low speed, can do nothing to encrypted data and new P2P applications with unknown characteristics, and have a high maintain cost. Because most P2P protocols are proprietary and reverse engineering is needed, and the method is not able to handle with brand-new applications that use unknown P2P protocols [8]. With the development of anti-identify technologies for P2P software, single use of DPI technologies may not satisfy the requirement.

(3) Deep Traffic Inspection (DFI) method, which tries to judge whether it has satisfied P2P traffic features through statistical analysis of data traffic [9]. Compared with DPI technologies, DFI execute in a high speed, are effective to encrypted data and new P2P applications with unknown characteristics, and have a relatively low maintain cost, but their exactness is lower than that of DPI, and cannot judge specific application types. Currently, studies and products based on DPI technologies are comparably more.

(4) Machine learning based identification. As a matter of fact, to identify P2P traffic is a problem belongs to pattern recognition, naturally all kinds of classification method could be applied to P2P traffic identification problem, such as Bayesian decision, C4.5 decision tree, Neural Network etc [10-12]. These methods use training data to establish a classification model, which is used to generate a classifier to classify the unknown data set. However, many machine learning methods have their inherent weaknesses. Especially in this real-time traffic identification, the accuracy and efficiency of the methods are not well meeting the demand.

Support Vector Machine (SVM) is one of the most promising and powerful machine learning methods for classification and regression problems of small samples and high dimensions. It was initially presented by Vapnik in the last decade of the 20th century based on statistical learning theory and structural risk minimization principle [13]. It has been proved to be very successful in many applications such as handwritten digit recognition, image classification, face detection, object detection, text classification [14]. Since SVM has an excellent ability to solve binary classification problems and the main purpose of P2P traffic identification is accurately classifying two classes: P2P and non-P2P traffic, there are some studies which have proposed P2P traffic identification based on SVM [15]. While it is noteworthy that the parameters have influence on the generalization performance of SVM, some work has done to deal with the parameter problems of SVM [16-17].

However, the problem has not been fully resolved. The Bacterial Foraging Algorithm (abbreviated to BG) is a recently developed swarm intelligence algorithm based on the foraging behavior of E. coli bacteria. It is a simple stochastic global optimization technique. The optimization problem search space could be modeled as a social foraging environment where groups of parameters communicate cooperatively for finding solutions to difficult problems. This idea could be applied to optimal parameters of SVM. So we can use BG algorithm to find the optimal parameters of SVM for the identification of P2P traffic.

The rest of the paper is organized as follows. In section 2 we simply describe the basic principle of support vector machine. The idea of bacterial foraging algorithm is explained in section 3. How to employ BG algorithm to find the optimal parameters of SVM is illustrated in section 4. In section 5, the performance of proposed approach is evaluated on the real P2P traffic data and is compared with existing techniques. Some conclusions are also provided towards the concluding section.

2. The Basic Principle of Support Vector Machine

Support Vector Machine is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding overfitting to the data, which is an active part of the machine learning research around the world.

Classifying data is a common task in machine learning. A support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite dimensional space, which can be used for classification and other tasks.

Supposed that some given data points each belong to one of two classes, and the goal is to decide which class a new sample point will be in. There are many hyperplanes that might classify the data. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training data point of any class, since in general the larger the margin the lower the generalization error of the classifier. If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier; or equivalently, the perception of optimal stability.

Let us have a data set $\{x_i, y_i\}, i = 1, \cdots, l$ of examples where $y_i \in \{-1, 1\}$ and $x_i \in R^d$ where x_i is an arbitrary data point and y_i its corresponding bipolar label. Let us also define a linear decision surface by the equation $f(x) = \omega \cdot x + b = 0$ .The original formulation of the SVM algorithm seeks a linear decision surface which maximizes the margin between the closest positive and. negative examples. This may be achieved through the minimization of the penalty term $\| \omega \|^2 / 2$. It yields $\omega = \sum_i \alpha_i y_i x_i$ with the constraint $\sum_i \alpha_i y_i = 0$, and $0 \le \alpha_i \le C, \forall i = 1, \cdots, l$ where C is a regularization variable called trade-off parameter or Penalty factor. The parameters α_i can be found after the following quadratic optimization problem is maximized:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j y_i y_j x_i x_j \tag{1}$$

The data examples whose corresponding α_i values are not zero are called support vectors.

Instead of considering the input space, we may consider a given augmented space by replacing the inner product of Equation (1) by the dot product $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ which yields:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{2}$$

Where function $K(x, y) = \phi(x) \cdot \phi(y)$ represents semi-definite kernel. Some of the classical SVM kernels are Linear Kernel, Polynomial Kernel and RBF Kernel.

The resulting decision frontier is $f(x) = \sum_i \alpha_i y_i K(x_i, x) + b$ where x_i, α_i and b represent respectively the ith support vector, its corresponding multiplier and the hyperplane bias. For nonseparable classes, the L2-SVM variant minimizes the function $(1/C)(\| \omega \|^2 / 2) + \frac{1}{2} \sum_i \xi_i^2$ which leads to the maximization of:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \tilde{K}(x_i, x_j) \tag{3}$$

With,

$$\tilde{K}(x_i, x_j) = K(x_i, x_j) + \frac{1}{C} \quad \text{if } i = j,$$

$$\tilde{K}\left(x_i, x_j\right) = K\left(x_i, x_j\right) \qquad \text{if } i \neq j,$$

Under the constraints $\sum_i \alpha_i y_i = 0$ and $\alpha_i \geq 0, \forall i = 1, \cdots, l$.

The effectiveness of SVM depends on the selection of kernel, the kernel's parameters, and soft margin parameter.

The most common kernel functions used in SVM are as follows:

Linear kernel function:

$$K(x_i, x_j) = x_i^T \cdot x_j \tag{4}$$

Polynomial kernel function:

$$K(x_i, x_j) = (x_i^T x_j + 1)^{\sigma} \tag{5}$$

RBF kernel function:

$$K(x_i, x_j) = \exp\left(-\| x_i - x_j \|^2 \big/ \sigma^2 \right) \tag{6}$$

The parameters σ in kernel function reflect the characteristic of training data, and great effect on the performance of the SVM. Penalty factor C determines the trade-off cost between minimizing the training error and minimizing the model's complexity. Whether the value is too big or small can reduce the generalization of SVM. In real applications, most of parameters are selected empirically by trying a finite number of parameter values and selecting those that get the least test error. Except for consuming enormous time, such trial and error procedures for selecting the parameters of SVM often fail to obtain the best performance as it is imprecise and the result is unreliable. In practice, grid search is a rather efficient for good parameters of SVM, however, it is only suitable for adjustment of very few parameters and does not perform well in practice because it is complex in computation and time consuming. It is a vital step to optimize the parameters of SVM for a good performance in handling a learning task as the performance of SVM will be weakened if these parameters are not properly chosen. The paper proposes a BG algorithm to optimize parameters C and σ automatically, the principle of BG will be illustrated in the next section.

3. Bacterial Foraging Algorithm

Recently, researches of optimal foraging of bacteria have been used for solving optimization problems. The foraging behavior of Escherichia coli, which is a common type of bacteria, is considered in the research [18]. The E. coli bacteria that are present in the intestines have a foraging strategy governed by four processes, namely, chemotaxis, swarming, reproduction, and elimination and dispersal [19].

Chemotaxis: This process achieves through swimming and tumbling as shown in Figure 1. Depending upon the rotation of the flagella in each bacterium, it decides whether it should move in a predefined direction (swimming) or the bacterium. To represent a tumble, a unit length random direction, $\phi(j)$ say, is generated; this will be used to define the direction of movement after a tumble. In particular,

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\phi(j) \tag{7}$$

Where $\theta^i\left(j,k,l\right)$ represents the ith bacterium at jth chemotactic kth reproductive, and lth elimination and dispersal step. $C\left(i\right)$ is the size of the step taken in the random direction specified by the tumble. “C” is termed as the “run length unit”.

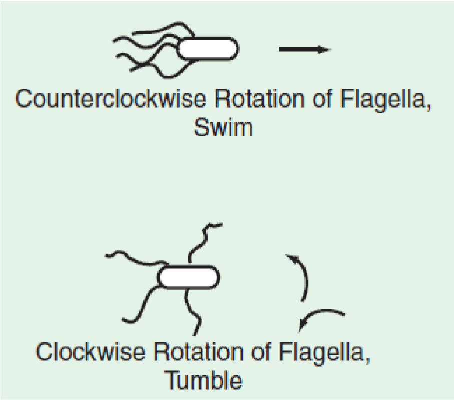

Figure 1. Swimming and Tumbling of E.coli

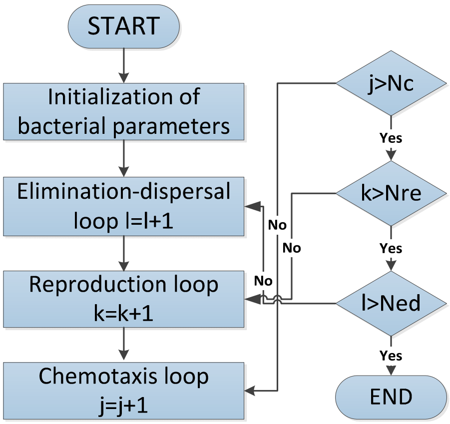

Figure 2. Flowchart of Bacterial Foraging Algorithm

It is always desired that the bacterium that has searched the optimum path of food should try to attract other bacteria so that they reach the desired place more rapidly. Swarming makes the bacteria congregate into groups and hence move as concentric patterns of groups with high bacterial density. Mathematically, swarming can be represented by:

$$J_{CC} = \sum_{i=1}^{S} J_{CC}^{i}\left(\theta, \theta^{i}\left(j,k,l\right)\right) \tag{8}$$

$$= \sum_{i=1}^{S}\left[-d_{attract}\exp\left(-\omega_{attract}\sum_{m=1}^{p}\left(\theta_m-\theta_m^i\right)^2\right)\right]+\sum_{i=1}^{S}\left[-h_{repellent}\exp\left(-\omega_{repellent}\sum_{m=1}^{p}\left(\theta_m-\theta_m^i\right)^2\right)\right]$$

Where $J_{CC}\left(\theta, P\left(j,k,l\right)\right)$ is the cost function value to be added to the actual cost function to be minimized to present a time varying cost function. “S” is the total number of bacteria. “p” is the number parameters to be optimized that are present in each bacterium. $d_{attract}, \omega_{attract}, h_{repelent}$ and $\omega_{repelent}$ are different coefficients that are to be chosen judiciously.

Reproduction: The least healthy bacteria die, and the other healthiest bacteria each split into two bacteria, which are placed in the same location. This makes the population of bacteria constant.

Elimination and Dispersal: It is possible that in the local environment, the life of a population of bacteria changes either gradually by consumption of nutrients or suddenly due to some other influence. Events can kill or disperse all the bacteria in a region. They have the effect of possibly destroying the chemotactis progress, but in contrast, they also assist it, since dispersal may place bacteria near good food sources. Elimination and dispersal helps in reducing the behavior of stagnation (being trapped in a premature solution point or local optima).

The process of bacteria foraging algorithm for solving the optimization problem includes: (1) Encode the solutions for the problem; (2) Design evaluation function; (3) Generate initial solution population; (4) Optimize parameters by using the interaction between groups.

Supposed Nc, Nre and Ned denote maximum chemotaxis, maximum reproduction, and maximum elimination-dispersal respectively. The optimization procedure of the bacterial foraging algorithm is described briefly as following Figure 2.

4. SVM Parameters Optimization Based on BG

4.1. The principle of optimal parameters of SVM

The performance of SVM mainly referred to the generalization ability. As stated in section 2, penalty factor C and kernel function parameters σ exert a considerable influence on the generalization ability of SVM. The kernel function parameters determine the mapping of the original space to high dimensional space and the value of penalty factor can adjust the error and complexity. As the value of each parameter, too big or too small, all hampers the generalization of SVM, the optimization of parameters is important to achieving good generalization ability in practice. This study proposes to employ BG algorithm to optimize parameters C and σ automatically. The main idea of applying BG to search the best parameters pair (C and σ) of SVM is as below.

Each position vector of the bacteria stands a candidate parameters pair for SVM. The initial population is generated with N number of solutions and each solution is a D-dimension vector, here D is set to 2 that each solution represents 2-D candidate parameters. X_i represents the i-th bacteria position in the population which denotes a candidate parameter pair and its fitness can be measured by fitness function, with defined movement rules, the virtual bacterium moves in the search place and update their position with predefined rules, till the virtual bacterium meets their end condition, the algorithm will terminate and output the best position as the optimal parameters for SVM.

4.2. The Implementation of the Proposed Method

The RBF kernel function is taken as the kernel function. The parameters needs to be optimized are *C* and σ . The basic steps are stated as follows:

Step 1: Read data S from file, then divide it into two groups S1 (training set) and S2 (testing set). Supposed the size of bacterium as N, and randomly generated N groups set of {C, σ } to initial location of the position of bacterium;

Step 2: Design fitness evaluation function $fitness = f(C,\sigma)$. According to the value of *C* and σ , train the SVM model with group S1, then consider opposite value of testing accuracy with group S2 as the fitness.

Step 3: Execute the loop of chemotaxis, reproduction and elimination-dispersal.

Step 4: Encode, define the position of bacterium with best fitness as the optimal C*or σ *.

The procedure for describing proposed BG-SVM is as follows:

Step 1: Initialize BG with population size. Set the number of bacterium and its dimension and other (l=0, k=0, j=0). Evaluate the fitness value of each bacteria. Take the cross validation error of the SVM training set as fitness value.

Step.2 Elimination-dispersal loop: l=l+1

Step.3 Reproduction loop: k=k+1

Step.4 Chemotaxis loop: j=j+1

Step.5: Executive chemotaxis operation

Step.6 If j<Nc, turn to step 4

Step.7 Executive reproduction operation

Step.8 If k<Nre, turn to step 3

Step.9 Executive elimination-dispersal operation

Step.10 If l<Ned, turn to step 2, else finish

Step.11 Repeat the step 4-7 until a value of the fitness function converges or the number of iteration reached.

After converging, the global best object is fed in to SVM classifier for testing.

5. Experiment and Discussion

To test the effectiveness of the proposed method, some real campus P2P traffic data are used to evaluate the performance of the proposed BG-SVM model for the parameter optimization of SVM and traffic identifation of P2P; moreover, the performance is compared with some commonly used algorithms presented in the literature, which are GA, PSO and GA-PSO algorithm. The data of P2P network traffic is collected from network laboratory of Hubei University of Technology. We collected more than 300 sample data and chose 11 features which showed in Table 1 for our experiments.

Table 1. The Description of Features

Features	**Explanation**
Duration	The duration of traffic
Type	IP or IP-Port
TCP-IO	The ratio of sending and receiving TCP packets
UDP-IO	The ratio of sending and receiving UDP packets
All-IO	The ratio of sending and receiving all packets
Avg-speed	The average speed of traffic
Avg-packets	The average packets size of traffic
Avg-TCP/UDP	The byte ratio of TCP and UDP average packets size
TCP/UDP	The byte ratio of TCP and UDP traffic
TCP-pro	The proportion of TCP in traffic
UDP-pro	The proportion of UDP in traffic

The collected data of network traffic is divided into training set and testing set, and put into SVM to classify for P2P identification. We chose nine-tenths for training data and one-tenth for testing data to get better experimental results. The ranged of parameters C and σ is form 2^{-10} to 2^{10} on the dataset. The main parameters used for these approaches are: the initial population for three algorithms is the same, that is 20, and all these algorithms will terminate after being executed 100 times. Moreover, the crossover rate for GA is 0.4 and mutation rate for GA is 0.01. And for PSO, C1=2.0 and C2=2.0, the value of inertia weight is set as 1. For BG, Ped=0.25.

Figure 3-Figure 5 show average classification accuracy and best classification accuracy respectively by three approaches: GA-SVM, PSO-SVM and BG-SVM.

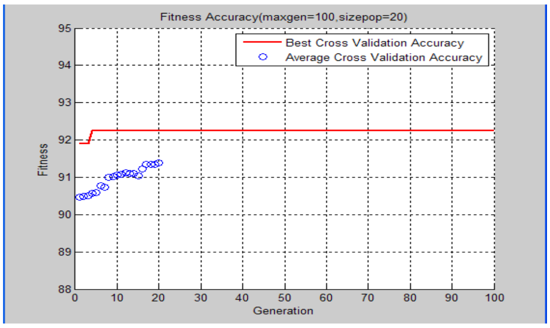

Figure 3. Classification by GA Algorithm (100, 20)

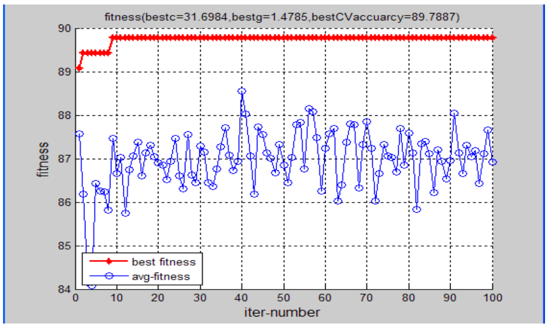

Figure 4. Classification by PSO Algorithm (100, 20)

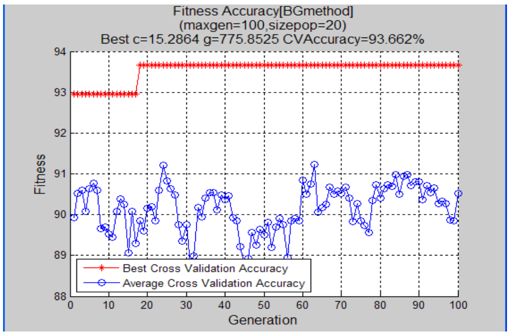

Figure.5. Classification by BG Algorithm (200,50)

From experiment one, the best classification accuracy and average classification accuracy obtained by the proposed BG-SVM algorithm are higher compared with the GA-SVM approach and PSO-SVM approach, which are illustrated in Figs.3-5: the best cross validation accuracy (BCVA) of GA is 92.3069% and the average cross validation accuracy (ACVA) of GA is a range from 90.5% to 91.5%; the BCVA of PSO is 89.7887% and the ACVA of PSO fluctuates markedly above and below 87%; while the BCVA of BG is 93.662% and the ACVA of BG fluctuates narrowly between 89% to 91%. Moreover, as we can see in Figure.3, the GA method stops evolution after 20th generation. So we changed population size from 20 to 50, maximum generation from 100 to 200, and showed results as following Figure.6-Figure8.

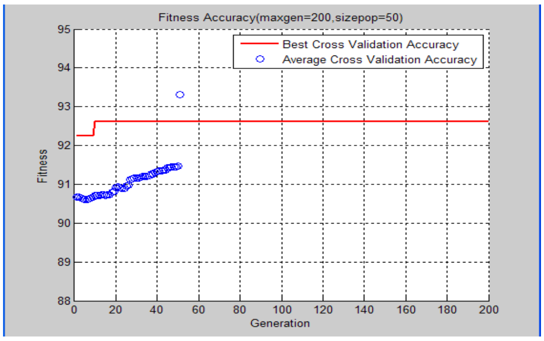

Figure 6. Classification by GA Algorithm (100, 20)

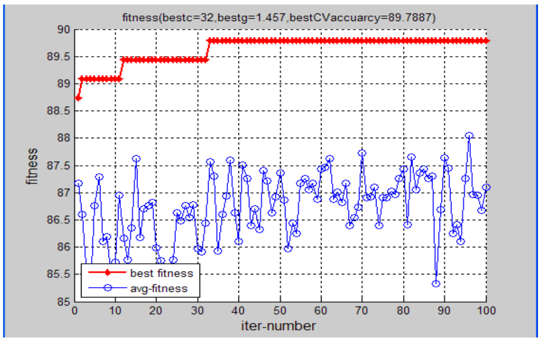

Figure 7. Classification by PSO Algorithm (200, 50)

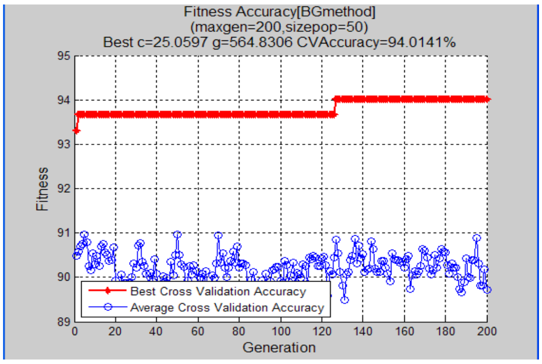

Figure 8. Classification by BG Algorithm (200, 50)

From experiment two, the best classification accuracy and average classification accuracy obtained by the proposed BG algorithm are also better than PSO algorithm and BG algorithm after changing population size and maximum generation. From Figure 6 to Figure 8, the BCVA of GA increases to 92.6804% and the ACVA of GA still ranges from 90.5% to 91.5%; two indexes of PSO are nearly the same with experiment one; while the BCVA of BG grows to 94.0141% and the ACVA of BG is more stable. Finally, the superior of proposed BG method can be proved by further experiment compared with GA-PSO algorithm [20-21] which combines good points of GA algorithm and PSO algorithm, and showed in Figure 9-10.

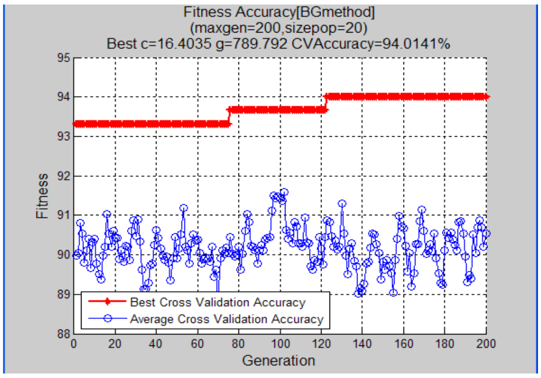

Figure 9. Classification by BG Algorithm (200, 20)

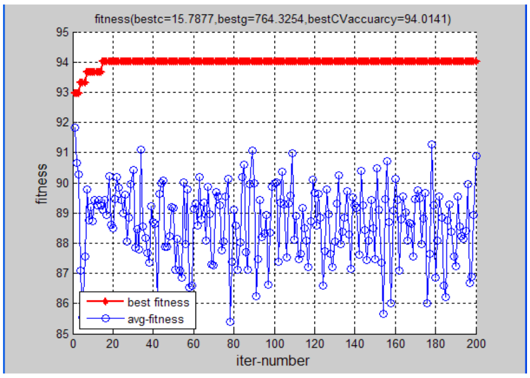

Figure 10. Classification by GA-PSO Algorithm (200, 20)

As we can seen in Figure 9 and Figure 10, we find even the best cross validation accuracy of BG algorithm is the same as GA-PSO algorithm, moreover, the average cross validation accuracy of the former is better than the latter, which shows that the BG algorithm is more stable that the GA-PSO algorithm. Based on the above experiments, we may conclude that BG-SVM method has the potential to be useful in classification for P2P traffic identification.

6. Conclusion

In this paper, a P2P traffic identification approach is developed based on SVM and BG algorithm. And we have tested the proposed method on real P2P datasets and compared them with several existing techniques. The experimental results indicate that the proposed BG algorithm is feasible to optimize the parameters for SVM, which testifies that the novel BG-SVM model can yield promising results. In all, ideally, the proposed method has the high accuracy of identifying P2P traffics. However, because of the limitation of the experimental environment, the P2P traffic captured in this paper is from campus network, the testing data used in the process still cannot cover all the factors that can affect test results. Moreover, in general, different traffic feature of P2P has different effect on classification of the flow, in the paper, all the features are set with the same weight value; how to set proper weight value for these features is worth further studying.

Acknowledgements

This work is supported by Natural Science Foundation of China (61170135 61202287) and Natural Science Foundation of Hubei Province of China (2011CDB075), the Key Project for Scientific and Technological Research of Education Department of Hubei Province in China (No. D20111409, No. D20121409) and the Key Project for Scientific and Technological Research of Wuhan City in China (201210421134).

References

[1] Yin Chunyong, Sun Ruxia, Yang Lei. Node-based sampling P2P bot detection. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10(5): 1117-1122.
[2] Yu Fanpeng, Zhang Sidong, Zhou Xin. Incentive Mechanism Algorithm for P2P File System in Campus Networks. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2012; 10(6): 1477-1484.
[3] Shen Xuemin, Yu Heather, Buford John, Akon Mursalin. Handbook of Peer-to-Peer Networking (1st ed.). New York: Springer. 2009; 118.
[4] DS Wallach. *A Survey of Peer-to-Peer Security Issues.* Proceedings of International Symposium on Software Security. 2002; 42-57.
[5] Yan Hua, Dah-Ming Chiu, John CS Lui. Profiling and identification of P2P traffic. *Computer Networks.* 2009; 53: 849-863.

[6] Hongwei Chen, Xin Zhou, Fangping You, Hui Xu, Chunzhi Wang, Zhiwei Ye. A SVM Method for P2P Traffic Identification based on Multiple Traffic Mode. *Journal of Networks.* 2010; 5(11): 1381-1388.

[7] Li Jun, Zhang Shunyi, Liu Shidong, Xuan Ye. *Active P2P Traffic Identification Technique.* International Conference on Computational Intelligence and Security. 2007; 37-42.

[8] Jingang Zheng, Yabin Xu. Identification of network traffic based on support vector machine. *Advanced Computer Theory and Engineering (ICACTE).* 2010; 3: 286-290.

[9] Yabin Xu and Jingang Zheng. Identification of Network Traffic Based on Radial Basis Function Neural Network Intelligent Computing and Information Science Communications. *Computer and Information Science*. 2011; 134: 173-179.

[10] Rui Wang, Yang Liu, Yue-xiang Yang, Hai-long Wang. *A New Method for P2P Traffic Identification Based on Support Vector Machine.* International Conference on Artificial Intelligence & Machine Learning Conference (AIML06). Sharm El Sheikh, Egypt. 2006; 58-63.

[11] Shen Fuke, Change Pan, Ren Xiaoli. Research of P2P Traffic Identification Based on BP Neural Network. *Intelligent Information Hiding and Multimedia Signal Processing*. 2007; 75-78.

[12] Gu Yiran, Wang Suoping. Traffic Identification Method for Specific P2P Based on Multilayer Tree Combination Classification by BP-LVQ Neural-Network. *Information Technology and Applications (IFITA)*. 2010; 34-38.

[13] Dong Seong Kim, Ha-Nam Nguyen, Jong Sou Park. Genetic algorithm to improve SVM based network intrusion detection system. *Advanced Information Networking and Applications (AINA).* 2005; 155-158.

[14] Mathias M Adankon, Mohamed Cheriet. Model selection for the LS-SVM. Application to handwriting recognition. *Pattern Recognition.* 2009; 42: 3264-3270.

[15] Xiao Li Zhang, Xue Feng Chen, Zheng Jia He. An ACO-based algorithm for parameter optimization of support vector machines. *Expert Systems with Applications.* 2010; 37: 6618-6628.

[16] Xuanyu Liu, Cheng Shao, Haifeng Ma, Renxi Liu. Optimal earth pressure balance control for shield tunneling based on LS-SVM and PSO. *Automation in Construction.* 2011; 20: 321-327.

[17] Zhenyuan Jia, Jianwei Ma, Fuji Wang, Wei Liu. Hybrid of simulated annealing and SVM for hydraulic valve characteristics prediction. *Expert Systems with Applications.* 2011; 38: 8030-8036.

[18] Nasir ANK, Tokhi MO, Ghani NM, Ahmad MA. A novel hybrid spiral-dynamics bacterial-foraging algorithm for global optimization with application to control design. *Computational Intelligence (UKCI)*. 2012; 1-7.

[19] Das S, Dasgupta S, Biswas A, Abraham A, Konar A. On Stability of the Chemotactic Dynamics in Bacterial-Foraging Optimization Algorithm. Systems, Man and Cybernetics. Part A: Systems and Humans. 2009; 670-679.

[20] Qian YuLiang, Zhang Hao. *Fault diagnosis for generator unit based on RBF neural network optimized by GA-PSO.* 8th International Conference on Natural Computation. 2012; 233-236.

[21] Kim Dong Hwa, Park Jin. *Loss minimization control of induction motor using GA-PSO*. 9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems. 2005; 222-227.