Journal of Software Engineering and Applications, 2014, 7, 1031-1038
Published Online November 2014 in SciRes. http://www.scirp.org/journal/jsea
http://dx.doi.org/10.4236/jsea.2014.712090

Scientific Research

An Integrated Intrusion Detection System by Combining SVM with AdaBoost

Yu Ren

College of Computer, Communication University of China, Beijing, China
Email: ryu.asak@gmail.com

Received 15 September 2014; revised 20 October 2014; accepted 12 November 2014

Academic Editor: Kwokwing Chau, Hong Kong Polytechnic University, Hong Kong

Abstract

In the Internet, computers and network equipments are threatened by malicious intrusion, which seriously affects the security of the network. Intrusion behavior has the characteristics of fast upgrade, strong concealment and randomness, so that traditional methods of intrusion detection system (IDS) are difficult to prevent the attacks effectively. In this paper, an integrated network intrusion detection algorithm by combining support vector machine (SVM) with AdaBoost was presented. The SVM is used to construct base classifiers, and the AdaBoost is used for training these learning modules and generating the final intrusion detection model by iterating to update the weight of samples and detection model, until the number of iterations or the accuracy of detection model achieves target setting. The effectiveness of the proposed IDS is evaluated using DARPA99 datasets. Accuracy, a criterion, is used to evaluate the detection performance of the proposed IDS. Experimental results show that it achieves better performance when compared with two state-of-the-art IDS.

Keywords

Intrusion Detection, Integrated Learning, Support Vector Machine, AdaBoost

1. Introduction

With the continuous development of network technology and the social economy, people enjoy the convenience that the Internet and computer technology bring, also experiencing the threat of malicious intrusion at the same time. Firewall, as the traditional network security technology, is difficult to form an effective defense against the upgrading of network intrusion means [1] [2]. In recent years, many experts and scholars pay increasingly atten-

How to cite this paper: Ren, Y. (2014) An Integrated Intrusion Detection System by Combining SVM with AdaBoost. *Journal of Software Engineering and Applications*, **7**, 1031-1038. http://dx.doi.org/10.4236/jsea.2014.712090

tion to intrusion detection system (IDS), which is the initiative protection technology. In view of the existing security risks in the Internet, an effective intrusion detection algorithm has important significance for the sustainable development of economy.

Traditional intrusion detection methods are mainly divided into anomaly detection and misuse detection. Anomaly detection mostly uses the expert experience and inference method. Statistical method [3] and Bayesian inference [4] are representative algorithms. This kind of intrusion detection method has better detection effect against intrusions with relatively stronger regularity, but it cannot resist the intrusion method with escalating high technology in today's Internet. The misuse detection can usually combine expert system with predicting algorithm. The expert system and the conditional probability are typical misuse detection algorithms; however, because the rules in expert system are difficult to be quickly updated, its performance is not good. The conditional probability method of misuse detection often depends on the temporal correlation, so the scope of application is small. In recent years, the technology of data mining and artificial intelligent algorithms has been introduced into IDS to enhance the detection accuracy, but it requires a lot of complete audit data as support. It is difficult to deal with the present situation where the network intrusion technology has strong concealment and updates quickly.

Ensemble learning is a machine learning method based on statistical learning theory, which can greatly improve the generalization ability of the learning algorithm. Under the condition of the limited number of training samples, it can ensure the relatively independence of test data and keep a smaller error. When ensemble learning method is introduced in IDS, in spite of the lack of prior knowledge, it will ensure that there is better classification accuracy, so that it has the better detection performance. Therefore, an intrusion detection ensemble learning algorithm based on the support vector machine (SVM) is proposed in the paper, which combines the SVM with the ensemble learning algorithm AdaBoost.

This paper is organized as follows. Section 2 introduces the algorithm principle and system structure. Section 3 proposes an intrusion detection model based on SVM. Section 4 proposes an intrusion detection ensemble learning algorithm based on AdaBoost. Experiment results are shown in Section 5. Finally we conclude in Section 6.

2. The Algorithm Principle and System Structure

The basic goal of IDS is through the collection and analysis of network data, detecting the behaviors of the breach of security strategy and the signs of attack existed possibly in target system. Through the active safety protection, the IDS will intercept the malicious intrusion and give the alarm to administrator before the network is endangered.

For this goal, an integrated IDS by combining SVM with AdaBoost is proposed in the paper and its structure is shown in **Figure 1**. Data acquisition module is responsible for collecting real-time network data flow from various information sources. In order to ensure the detecting speed and accuracy, the collected data were dealt with by the detection engine module, which consists of two parts: feature selection and detection model. Feature selection is used to extract features and reduce the data space dimension. The detection model is established through the learning feature data.

Classification algorithm is the focus of this paper and its idea is that combining the SVM with the AdaBoost, first, using SVM to learn the network feature data, then intrusion detection model is obtained as base classifiers, at the same time, in order to solve the problem that the accuracy of SVM is not high for small sample, ensemble learning algorithm AdaBoost is introduced, then which iteratively optimizes the base classifiers based on SVM, and improve accuracy.

3. Intrusion Detection Model Based on SVM

The SVM algorithm proposed by Vapnik and other scholars can effectively deal with the nonlinear data and limit the overlearning. It has both rigorous theoretical foundation and mathematical foundation. It does not exist the problem of local minima. It has strong generalization ability for this kind of small sample learning application such as network intrusion detection, and has weak dependence on the number of samples [5] [6].

The standard SVM algorithm is a convex quadratic optimization problem. The global optimum always can be found in the above problem. But when training samples increase, due to the too many constraints, it will greatly increase the training time and the memory requirements, which becomes the bottleneck in practical applications.

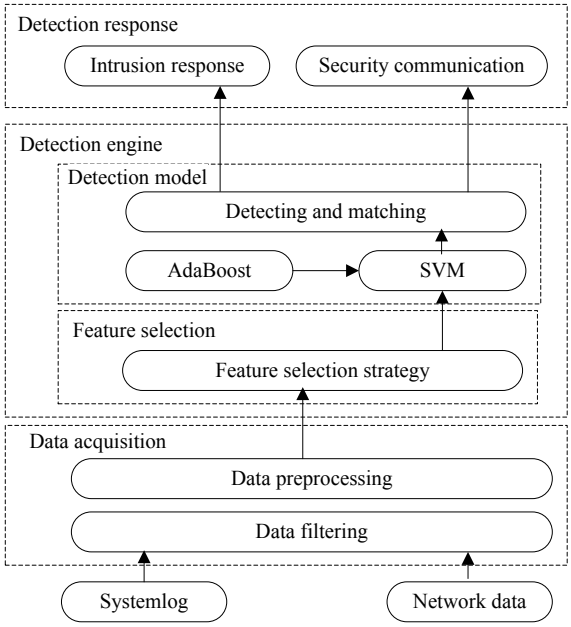

Figure 1. The principle of intrusion detection algorithm.

In order to improve the training efficiency of SVM, Suyken changed the constraints and the risk function of standard SVM, and then proposed the least square support vector machine (LS-SVM) [7]. LS-SVM only needs to solve linear equations, and makes the SVM easier to be implemented, and greatly improves SVM's training efficiency. Therefore, the LS-SVM is used for base classifier in the paper.

In the actual network environment, each network node will receive a mass of network data. In these data, only a small part of information represents the intrusion behavior. In order to reduce the useless data, feature selection strategy of KDDCUP99 (Data Mining and Knowledge Discovery Cup in 1999) is improved in this paper. $w_n(d)$ is a function of feature selection, which is defined as Equation (1):

$$w_n(d)=\frac{P_{ik}\times tf_{ik}\times\log\left(N/n_k+0.001\right)}{\sqrt{\sum_{k=1}^{n}\left(P_{ik}\times tf_{ik}\right)^2\times\log\left(N/n_k+0.001\right)}}\times\left(1-1/L\right) \tag{1}$$

where P_{ik} is a weight factor of a network data t_k from dataset d. tf_{ik} is the frequency that t_k appears in d. N is the number of data in d. n_k is the frequency that t_k containing a specific port appears in d. L is the length of t_k. The packets with greater weight are selected as the training samples.

Given the training sample set (x_i, y_i), where $i=1,2\cdots,n$; $x_i\in R^n$ is input variable; $y_i\in R^n$ is output variable. Sample contains 9 basic features, 13 content features, 9 flow features within two seconds and 10 host flow features. The basic idea of SVM regression theory is to find a nonlinear mapping φ from an input space to an output space. By the nonlinear mapping [8] [9], mapping data x into a high-dimensional feature space F. In the feature space F, an estimation function $f(x)$ shown in Equation (2) is used to complete linear regression.

$$f(x)=\left[\omega\times\varphi(x)\right]+b,\ \phi: R^n\to F,\ \omega\in F \tag{2}$$

where b is the threshold. Function approximation problem is equivalent to Equation (3).

$$R_{\text{reg}}(f)=R_{\text{emp}}(f)+\lambda\|\omega\|^2=\sum_{i=1}^{n}C(e_i)+\lambda\|\omega\|^2 \tag{3}$$

where $R_{\text{reg}}(f)$ is the objective function; $R_{\text{emp}}(f)$ is the empirical risk function; n is the number of samples; λ is an adjustable constant; C is error penalty factor; $\|\omega\|^2$ reflects the complexity of f in high-dimensional space.

Considering the linear ε-insensitive loss function has better sparsity, the loss function is shown as Equation

(4):

$$\left|y-f(x)\right|_{\varepsilon}=\max\left\{0,\left|y-f(x)\right|-\varepsilon\right\} \tag{4}$$

Empirical risk function is shown as Equation (5):

$$R_{\text{emp}}^{\varepsilon}(f)=\frac{1}{n}\sum_{i=1}^{n}\left|y-f(x)\right|_{\varepsilon} \tag{5}$$

According to the statistical theory, a regression function is determined by the following objective function minimization, which is shown as Equation (6):

$$\begin{cases}\min\left\{\dfrac{1}{2}\|\omega\|^2+C\sum\limits_{i=1}^{n}\left(\xi_i^*+\xi_i\right)\right\}\\ y_i-\omega\cdot\phi(x)-b\le\varepsilon+\xi_i^*\\ \omega\cdot\phi(x)+b-y_i\le\varepsilon+\xi_i\\ \xi_i\ge 0,\quad \xi_i^*\ge 0\end{cases} \tag{6}$$

where C is the weight parameter that is used to balance the model complex item and the training errors item; ξ_i, ξ_i^* are slack factors; ε is insensitive loss function. Equation (6) can be converted to its dual problem Equation (7) which can be easily solved and shown as below:

$$\begin{cases}\max=-\dfrac{1}{2}\sum\limits_{i,j=1}^{n}\left(a_i^*-a_i\right)\left(a_j^*-a_j\right)K\left(x_i,y_j\right)x-\dfrac{1}{2}\sum\limits_{i,j=1}^{n}\left(a_i^*-a_i\right)\left(a_j^*-a_j\right)K\left(x_i,y_j\right)+\sum\limits_{i}^{n}a_i^*\left(y_i-\varepsilon\right)-\sum\limits_{i=1}^{n}a_i\left(y_j-\varepsilon\right)\\ \sum\limits_{i=1}^{n}a_i=\sum\limits_{i=1}^{n}a_i^*\\ 0\le a_i^*\le C\\ 0\le a_i\le C\end{cases} \tag{7}$$

By using the Lagrange multiplier method and kernel technology, LS-SVM can be converted to Equation (8) which is shown as below:

$$\begin{bmatrix}0 & \mathbf{1}^{\mathrm{T}}\\ \mathbf{1} & \Omega+C^{-1}I\end{bmatrix}\begin{bmatrix}b\\ a\end{bmatrix}=\begin{bmatrix}0\\ Y\end{bmatrix} \tag{8}$$

where $\Omega_{i,j}=k\left(x_i,x_j\right)=\psi\left(x_i\right)\psi^{\mathrm{T}}\left(x_j\right)$, $i,j=1,2,\cdots,n$, $a=\left[a_1,a_2,\cdots,a_n\right]$, $b=\left[b_1,b_2,\cdots,b_n\right]$, $\mathbf{1}=\left[1_1,1_2,\cdots,1_n\right]$.

In order to satisfy the any symmetric function in the Mercer condition, the selection of kernel function $k\left(x_i,x_j\right)=\psi\left(x_i\right)\psi^{\mathrm{T}}\left(x_j\right)$ requires some prior knowledge, but at present there is no general conclusion. Scholkopf discussed the selection and construction of kernel function. To construct a black box model of SVM, the most important is the selection of kernel function. Linear function, polynomial function, radial basis function, multilayer perceptron function are some typical kernel function. The radial basis function is used in this paper, which is shown in Equation (9).

$$K\left(x_i,x_j\right)=\exp\left[\frac{\left\|x_i-x_j\right\|}{\sigma^2}\right] \tag{9}$$

SVM regression functions $f(x)$ can be obtained by solving the above problems, which is shown as Equation (10):

$$f(x)=\sum_{i}^{n}\left(a_i-a_i^*\right)K\left(x_i,x_j\right)+b \tag{10}$$

Thus, the base classifier of the network intrusion detection system is obtained, and a new network data vector x can be classified by linear decision function which is shown as Equation (11):

$$g(x)=\operatorname{sgn}\left(\sum_{i}^{n}\left(a_i-a_i^*\right)K\left(x_i,x_j\right)+b\right) \tag{11}$$

4. Intrusion Detection Ensemble Learning Algorithm Based on AdaBoost

In the network, many factors can make nodes to be under the threat of intrusion, which result in the intrusion time and feature information presenting a certain weak randomness. Single SVM algorithm has certain generalization ability for small sample, but for the problem of intrusion detection, its accuracy is still not high. AdaBoost is a typical ensemble learning method, and it can synthetically optimize multiple weak base classifiers with relatively low accuracy [10] [11] and then we can get a strong classifier with high accuracy, which can improve forecast accuracy. In this paper, AdaBoost is used in the classifier based on SVM. The principle is: using the SVM algorithm to generate a series of base classifiers, training of each base classifier depends on the outcome of last base classifier [12] [13]. The probability distribution of training samples is adjusted by the error rate of base classifier on the training set, and the final intrusion detection model is built by weighting each base classifiers.

The model based on ensemble learning algorithm AdaBoost is shown in **Figure 2**. Firstly, calculating sample weight according to the prediction error of base classifier (initialization, the weight of each sample in subset is the same); secondly, using weighted samples to train every base classifiers, and getting the intrusion detection model. At the same time, calculating and updating the model weight. Then, iteration, until the number of iterations or model precision achieves target setting. If iteration is over, the final intrusion detection model is generated by weighting every intrusion detection model.

Training algorithm

Step 1: given feature sample set $S=\left\{\left(x_1,y_1\right),\left(x_2,y_2\right),\cdots,\left(x_n,y_n\right)\right\}$, where n is the number of samples; x_n is a input vector, and represents training sample; y_n represents class label. The initial weight of each sample d_1, d_2, ···, d_n is set to $1/n$. Maximum iteration of algorithm is set to T.

Step 2: using algorithm to optimize connection weight of SVM, and getting optimal weight.

Step 3: using sample set to train the optimized SVM, to getting tth intrusion detection model h_t.

Step 4: recording the intrusion detection model h_t, and calculating and saving its weight ω_t. Then using the samples to train h_t, and calculate the sum of the absolute values of the prediction error δ. If δ is less than the set value, or the number of iterations achieves maximum iterations, the iteration is over and enter into Step 6, or else, entering into Step 5.

Step 5: updating the weight $d_1,d_2,\cdots,d_m$, according to the δ, returning Step 2.

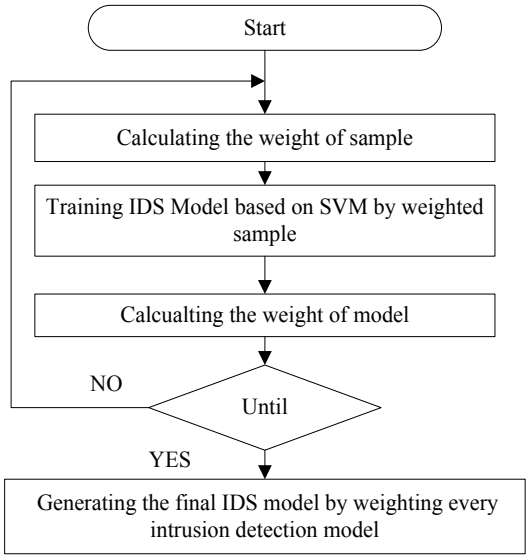

Figure 2. The principle of ensemble learning algorithm based on AdaBoost.

Step 6: getting the final prediction model $h = \sum_{t=1}^{T} \omega_t h_t$.

There are two main factors affecting the AdaBoost ensemble learning effect: the one is how to distribute sample weight in each round of cycle; the two is how to integrate many rules into an effective prediction rule. These two points are respectively reflected by the sample weights and model weights.

4.1. The Calculation of Sample Weight

Through adjusting the sample weights, the effect of the error samples for intrusion detection model can be effectively reduced, and the contribution of the correct sample can be promoted. The acquisition of sample weight is divided into two steps: computation and normalization. The weight is measured by using the absolute value of prediction error; the method is defined as Equation (12):

$$\begin{cases} E_t = \sum_{k=1}^{n} \left(d_t(k) \left(h_t(k) - y_t(k) \right) \right)^2 \\ \beta_t = \dfrac{E_t}{1 - E_t} \\ d'_{t+1}(k) = d_t(k) \cdot \beta_t \exp\left(1 - \dfrac{h_t(k) - y_t(k)}{\max\limits_{t=1}^{n} \left| h_t(k) - y_t(k) \right|} \right) \end{cases} \tag{12}$$

where E_t represents the sum of the weighted variance of training sample on the tth intrusion detection model h_t. β_t is adjustment coefficient; there is a variety of ways about the selection of adjustment coefficient, and in order to ensure the final prediction model is stable, this paper adopts the above way. $d'_{t+1}(k)$ is the new weight of sample, which is use to update $d_t(k)$ at the next iteration.

The sum of all sample weights must be 1, so the weights must be normalized; the method is defined as Equation (13):

$$d_{t+1}(k) = \frac{d'_{t+1}(k)}{\sum_{k=1}^{n} d'_{t+1}(k)} \tag{13}$$

4.2. The Calculation of Model Weight

The weight of intrusion detection model directly influences the output of the final prediction model. In order to enhance the contribution of intrusion detection model with the smaller errors in the final model, we use the absolute value of prediction error to measure the model weight ω_t; the method is defined as Equation (14):

$$\begin{cases} E_t = \sum_{k=1}^{n} \left(d_t(k) \left(h_t(k) - y_t(k) \right) \right)^2 \\ \beta_t = \dfrac{E_t}{1 - E_t} \\ \omega_t = \dfrac{1}{2} \cdot \ln\left(\dfrac{1}{\beta_t} \right) \end{cases} \tag{14}$$

where E_t represents the sum of the weighted variance of training sample on the tth intrusion detection model h_t. β_t is adjustment coefficient. ω_t is the effect weight of the tth intrusion detection model h_t for final intrusion detection model.

5. Experiment Results

In order to verify the effectiveness of the algorithm, computer simulation is carried out in accordance with our

proposed intrusion detection algorithm in this paper. All the algorithms are implemented in MATLAB 7.0 environment on a PC (Personal Computer) with Intel P4 processor (2.9 GHz) with 2 GB RAM. We investigate its classification accuracy.

For the evaluation of the performance of IDS, the majority of experts and scholars generally use DARPA99 data. In order to ensure the authority of the simulation, this paper uses the same dataset to evaluate the algorithm. The dataset was divided into training set (comprising 5 million connection data) and test set (comprising 311029 connection data). The test set includes some attacks that have not appeared in the training set.

This paper extracts 29313 sample data of 41 dimensional from the training set, which contains 6059 "Normal", 3866 "Neptune", 516 "Portsweep", 177 "SatanJ", 11 "Buffer_overflow" and 2183 "Guess-password", and extracts 124970 sample data from test set, which is divided into 5 test sets. In experiments, we focus on the comparison between our algorithm and two state-of-the-art algorithms, including BP (Back Propagation) neural network and SVM. The "Accuracy" is used to evaluate methods, which is defined as Accuracy = (TP + TN)/(TP + FP + TN + FN), where TP, TN, FP and FN are the number of true positive, true negative, false positive and false negative, respectively. The test results are shown in **Tables 1-3**.

It can be seen from **Table 1**, test accuracy on "Satan" and "Buffer_overflow" is worse than other test set using BP neural network algorithm for intrusion detection, and their error is relatively large, because the number of sample in the two training set is relatively few and the network data has certain randomness. If the sample is larger, the effect is slightly promoted, but accuracy is still low.

Compared with **Table 1**, **Table 2** shows that test accuracy on "Satan" and "Buffer_overflow" is better than **Table 1**, because the small sample generalization ability of SVM is slightly better than the BP neural network.

Table 1. Accuracy of IDS based on BP neural network.

%	Normal	Neptune	Portsweep	Satan	Buffer_overflow	Guess-password
Test set 1	85.2	78.5	71.3	65.6	61.2	75.4
Test set 2	81.5	74.6	72.5	68.3	53.5	73.6
Test set 3	81.2	75.3	75.3	72.2	48.4	75.1
Test set 4	84.3	74.3	74.6	64.6	58.7	74.2
Test set 5	85.6	76.8	72.3	66.6	57.3	72.3

Table 2. Accuracy of IDS based on SVM.

%	Normal	Neptune	Portsweep	Satan	Buffer_overflow	Guess-password
Test set 1	88.4	77.8	73.5	69.7	72.2	74.9
Test set 2	82.5	77.6	75.5	75.3	75.5	78.6
Test set 3	83.2	76.3	74.3	78.2	62.4	76.1
Test set 4	85.3	72.3	72.6	72.6	68.7	72.2
Test set 5	88.6	75.8	75.3	75.6	69.3	73.3

Table 3. Accuracy of IDS based on SVM and AdaBoost.

%	Normal	Neptune	Portsweep	Satan	Buffer_overflow	Guess-password
Test set 1	98.2	95.5	96.4	89.6	86.3	96.4
Test set 2	95.5	93.6	95.4	92.3	85.5	95.6
Test set 3	97.2	96.3	93.2	87.2	87.8	97.1
Test set 4	98.3	95.3	92.3	85.6	83.6	93.2
Test set 5	97.6	93.8	93.1	88.6	89.2	93.3

For the other test sets, there's not much difference between the two tables. Detection accuracy overall increase slightly in **Table 2**, but it's still low.

As can be seen from **Table 3**, due to the adoption of the AdaBoost algorithm for iterative correction of SVM model, the influence of random sample to the model is greatly reduced, the generalization ability is greatly enhanced compared with the SVM and BP neural network algorithm. The final intrusion detection model is more close to the real-world scenarios of network intrusion, and reduces the problem that the small sample causes the accuracy sharp decreasing. For the small sample such as "Satan" and "Buffer_overflow", accuracy still has been guaranteed, at the same time, the overall detection accuracy of the model is also a big promotion.

6. Conclusions

In this paper, we have proposed an efficient intrusion detection system by combining SVM with AdaBoost algorithms to detect attacks with the characteristics of fast variation, strong concealment and random. The IDS uses our proposed algorithm that is an integrated learning algorithm. Firstly, the feature of higher weight packets is learnt by using SVM. Through the training for the SVM an intrusion detection base classifier is established. Secondly, SVM base classifiers are iteratively trained by using the ensemble learning algorithm AdaBoost. Finally, the final intrusion detection model is generated. The experiment results show that our proposed algorithm is effective in detecting attacks with high detection accuracy, even if detect objects have the characteristics of small sample and randomness. Compared with IDS based on SVM or BP neural network, our proposed IDS greatly improves detection accuracy.

However, the weight setting is important for our algorithm. Our future works include constructing better weighting function and improving the generalization ability further. It is also interesting to use our proposed IDS to real-world scenario.

References

[1] Xiao, L., Chen, Y. and Chang, C.K. (2014) Bayesian Model Averaging of Bayesian Network Classifiers for Intrusion Detection. *Proceedings of the* 2014 *IEEE* 38*th Annual International Computers*, *Software* and *Applications Conference Workshops*, Vasteras, 2014, 128-133.

[2] Panja, B., Ogunyanwo, O. and Meharia, P. (2014) Training of Intelligent Intrusion Detection System using Neuro Fuzzy. *Proceedings of* 2014 15*th IEEE/ACIS International Conference on Software Engineering*, *Artificial Intelligence*, *Networking and Parallel/Distributed Computing* (*SNPD*), Las Vegas, 2014, 1-6.

[3] Fuchsberger, A. (2005) Intrusion Detection System and Intrusion Prevention Systems. *Information Security Technical Report*, **34**, 134-139. http://dx.doi.org/10.1016/j.istr.2005.08.001

[4] Larrañaga, P., Karshenas, H. and Bielza, C. (2013) A Review on Evolutionary Algorithms in Bayesian Network Learning and Inference Tasks. *Information Sciences*, **233**, 109-125 http://dx.doi.org/10.1016/j.ins.2012.12.051

[5] Wang, Z., Li, L. and Niu, L. (2009) Load Modeling Based on Support Vector Machine Based on Bayesian Evidence Framework. *Transactions of China Electrotechnical Society*, **24**, 83-86.

[6] Davy, M., Desobry, F. and Arthur, G. (2006) An Online Support Vector Machine for Abnormal Events Detection. *Signal Processing*, **86**, 2009-2025. http://dx.doi.org/10.1016/j.sigpro.2005.09.027

[7] Wang, Z. and Sun, X. (2011) Document Classification Algorithm Based on MMP and LS-SVM. *Procedia Engineering*, **15**, 1565-1569. http://dx.doi.org/10.1016/j.proeng.2011.08.291

[8] Li, M., Wen, G. and Wang, S. (2008) Parameters Selection of Support Vector Regression Based on Genetic Algorithm. *Computer Engineering and Applications*, **44**, 23-26.

[9] Yang, G., Li, C. and Ma, G. (2008) Parameter Selection for Support Vector Machines Based on Hybrid Genetic Algorithm. *Journal of Harbin Institute of Technology*, **40**, 134-138.

[10] Chen, A., Xia, L. and Zhao, G. (2004) Face Detection Based on Boosting Algorithm. *Computer Engineering and Applications*, **27**, 48-52.

[11] Fei, W., Zhuang, Y. and Pan, H. (2003) Recognition of Multiple Audio Clip Classes Based on FBM and AdaBoosting. *Journal of Computer Research and Development*, **18**, 62-65.

[12] Zhao, J. (2013) Asymptotic Convergence of Dimension Reduction Based Boosting in Classification. *Journal of Statistical Planning and Inference*, **143**, 651-662. http://dx.doi.org/10.1016/j.jspi.2012.10.007

[13] Xia, L. and Dai, R. (2003) Fault Testing on Rolling Bearing Based on Boosting Fuzzy Classification. *Pattern Recognition and Artificial Intelligence*, **18**, 72-75.

Scientific Research Publishing (SCIRP) is one of the largest Open Access journal publishers. It is currently publishing more than 200 open access, online, peer-reviewed journals covering a wide range of academic disciplines. SCIRP serves the worldwide academic communities and contributes to the progress and application of science with its publication.

Other selected journals from SCIRP are listed as below. Submit your manuscript to us via either submit@scirp.org or Online Submission Portal.