International Journal of Security and Its Applications
Vol.7, No.5 (2013), pp.193-204
http://dx.doi.org/10.14257/ijsia.2013.7.5.18

Intrusion Detection Ensemble Algorithm based on Bagging and Neighborhood Rough Set

Hui Zhao

School of Mathematics and Computer Science, Shaanxi University of Technology, Hanzhong , Shaanxi 723000 , china
zh911@sina.com

Abstract

Intrusion detection data often have some characteristics such as nonlinearity, higher dimension, much redundancy and noise, and partial continuous-attribute. This paper presents a new ensemble algorithm to improve intrusion detection precision. Firstly, it generates multiple training subsets in difference by using bootstrap technology. Then using neighborhood rough sets with different radiuses to make attribute reduction in these subsets, obtained the training subsets with greater difference, while Particle Swarm Optimization is used to optimize parameters of support vector machine in order to get base classifiers with greater difference and higher precision. Finally, the above base classifiers were integrdinedd by weighted synthesis method. The result of the emulation experiment in KDD99 data set indicates that this algorithm can effectively improve intrusion detection precision ,and it has higher generalization and stability.

Keywords: *Intrusion Detection, Bagging, Neighborhood Rough Set, Support Vector Machine, Particle Swarm Optimization, Ensemble Learning*

1. Introduction

Intrusion detection is an active network security technology that discover intrusion by collecting and analyzing information in the protected system, mainly monitors real-time network and computer system to discover and identify the intrusion activities and send the intrusion alarms. Intrusion detection is generally regarded as two-category problem of system status is normal or abnormal.

Obtained data from the field of intrusion detection often have characteristics of nonlinearity and higher dimension, and usually don't comply with some known distribution, so it is difficult for traditional statistics method to effectively detect it. Machine learning (*e.g.*, neural network, decision tree, bayesian network, support vector machine and k-nearest neighbor) is used in the field of intrusion detection. In above methods, support vector machine (SVM) is an intelligent machine learning method based on statistical learning theory [1]. SVM has some advantages, such as simple structure, global optimization, short training time and good generalization, it can better solve problems of higher dimension, nonlinearity, small sample and so on. Chen *et al.*, [2-3] conduct intrusion detection by using SVM, and it gained better effect, it indicated that SVM was better than other classification algorithms. But classification performance of SVM is often influenced by parameters [4-5], different parameters have different classification performance, and results of single classifier are easy to fall into local optimum, so single classifier may lead to problems of stability and reliability, ensemble technology is introduced to solve this problem [6-10]. Ensemble learning technology is a method of machine learning that has better learning effect than single classifier, it use series of classifiers to learning and combines various classification results by some rules. In the ensemble process, each base classifiers may search different local areas of solution space, and final combination results usually tend to certain actual goal in common. This

ISSN: 1738-9976 IJSIA

technology can improve stability and reliability of algorithm, as well as strengthen its generalization.

Krogh *et al.*, [11] point out that diversity and generalization of base classifiers are key factors for impacting ensemble effect. To enhance ensemble generalization, there are two aspects: firstly, improve diversity of base classifiers. Bagging is a kind of ensemble algorithm by directly perturbing training sample set [12], the gained training subsets by bootstrap technology have obvious diversity (about 37.7% distinctive samples) for obtaining base classifiers with diversity. Secondly, decrease generalization error of base classifiers. Not all attributes of intrusion detection data are effective or crucial in intrusion detection and often have much noise, meanwhile, it exists some redundancy in attributes, all these problems will reduce classifiers precision, especially SVM, it is very sensitive to singular points and noise data, therefore, effective attribute reduction is an important method to improve classifiers precision. Rough set is a math tool [13], which is established according to equivalence relation in discrete space and specializing in inaccuracy and uncertainty, and its core is attribute reduction and its generative rules include conditional attribute that is indispensable for classification. Cai *et al.*, [14-15] introduced rough set into intrusion detection to improve detection effect. But traditional rough set can't directly process continuous data, so data of continuous attributes must be discretized first, but information are loss in the discretization process, therefore, detection results are influenced highly by discretization technology, and detection precision is reduced. Neighborhood Rough Set is a method based on classical rough set [16-18], it can directly process continuous data. Therefore, information are not loss before reduction, making the attribute subsets have stronger classification capacity.

In addition, SVM performance is closely related to its kernel function type, kernel function parameter, penalty parameter and so on, these parameters will impact classification accuracy of SVM. Currently, we obtain the optimal parameter through our experience and a large number of repeatable experiments, but this method is time-consuming and obtained parameters is not always most optimal. In recently years, many scholars presented other methods of parameter optimization, the literature [19] used gradient descent algorithm to optimize parameters, the algorithm can shorten searching time of parameters, but it has higher requirement to the initial point and is a kind of linear searching method, so the results are easy to fall into local optimum; The literature [20-21] used genetic algorithm while the literature [22-23] used ant colony algorithm to optimize parameters, though these intelligent methods can reduce dependence on initial point , principles and ideas of these algorithms are complex, and different optimization problems need to be designed in different ways, such as crossover, mutation and selection, however, the results are also easy to fall into local optimum. Particle swarm optimization (PSO) is a effective global optimization algorithm [24-25], which guides optimization searching process to find optimal solution through the competition and cooperation between particles in a population of swarm. With the features of above mentioned methods, this method has simple concept, high efficiency and easy to implement features.

On the basis of the above analysis and characteristics of intrusion detection data, firstly, this paper used bootstrap technology to produce multiple training subsets with diversity, then attributes were reduced by neighborhood rough set in each training subset, while SVM parameters were optimized based on PSO. Finally, the base classifiers were ensembled by weighted average method. This algorithm combines sample perturbation of bootstrap with characteristic perturbation of neighborhood rough set to obtain training subsets with greater diversity. Meanwhile, attribute reduction based on neighborhood rough set eliminates invalid attributes, noise and optimized SVM parameters by PSO reduces generalization error of classifiers, so ensemble classifiers can effectively improve generalization performance. Finally, we do emulation experiments through KDD99 data set to verify validity and superiority of the algorithm.

2. Ensemble Algorithm

2.1. Attribute Reduction based on Neighborhood Rough Set

Intrusion detection data have characteristics of higher dimension and large sample, which causes two problems during the process of machine learning: firstly, there are much noise and redundancy attribute in intrusion detection data set, which seriously impacts classification accuracy of classifiers; secondly, training and classification time of machine learning increase as data dimensions increase, which will decrease efficiency of classification. Large researches demonstrate that attribute reduction is an effective method to improve precision of classifiers and efficiency of the algorithm.

Rough set was a math analysis tool and was brought up by Pawlak in 1982 to effectively process incomplete and inaccurate information [13]. This theory don't need any prior information and can only rely on internal information of data themselves to discover tacit knowledge within them, reveal potential rules and effectively process incomplete and inaccurate data. In traditional rough set theory, continuous data must be first discretized, which will result in original information loss, therefore the results of calculation and process are highly decided by discretization effect. Neighborhood rough set that was brought up by Qinghua Hu is a method that develops from the theory of classical rough set and can directly process continuous data [16-18]. It needn't discretize continuous data in advance, and can be directly used for problems of knowledge reduction *etc*. As a result, this paper adopts neighborhood rough set to reduce attributes, and subsequently implements neighborhood granulation of sample space of intrusion detection information, directly calculates distance of between-sample and determines neighborhood relation among samples.

Neighborhood decision system $NDT =< S, A = G \cup D, V, f >$, here, $S = \{s_1, s_2, \ldots s_m\}$ is a sample set, called a sample space. $G = \{g_1, g_2, \ldots g_n\}$ is an attribute subset, namely conditional attribute. $D = \{L\}$ is an output characteristic variable, called decision attribute, L refers to labels of sample. V_a refers to attribute range $a \in G \cup D$, f is an information function, can be expressed as $f : S \times (G \cup D) \to V$, here $V = \bigcup_{a \in G \cup D} V_a$.

If $\forall s_i \in S$ and $B \in G$, neighborhood of sample s_i in sub-attribute space B is labeled as $\delta_B(s_i)$, so $\delta_B(s_i) = \{s_j \mid s_j \in S, \Delta_B(s_i, s_j) \le \delta\}$, here δ is a presetting threshold value, and $\Delta_B(s_i, s_j)$ is a measure function in sub-attribute space B. Usually used measure functions include Manhattan distance, Euclidean distance and Chebychev distance. When s_1 and s_2 represent two samples in n-dimensional attribute space $G = \{g_1, \ldots, g_n\}$, and $f(s, g_i)$ represents value of attribute g_i of sample s in *i* dimension, Minkowsky distance can be defined as $\Delta_p(s_1, s_2) = (\sum_{i=1}^{n} \mid f(s_1, g_i) - f(s_2, g_i) \mid^p)^{1/p}$, here if $p = 1$, call Manhattan distance Δ_1; if $p = 2$, call Euclidean distance Δ_2; if $p = \infty$, call Chebychev distance Δ_∞.

Given that a neighborhood decision table $NDT, X_1, X_2, \ldots X_c$ is a sample subset having decision attribute values from 1 to c, so $X_i \cap X_j = \varnothing$, $i, j \in [1, c]$, $i \ne j, \bigcup_{i=1}^{c} X_i = S$, therefore $X_1, X_2, \ldots X_c$ is a partition of S, $\delta_B(x_i)$ refers to neighborhood information granularity (including sample x_i) created by attribute subset $B \subset G$, and then decision attribute D about upper approximation and lower approximation of attribute subset B can be represented as follows respectively: $Lower(D, B) = \bigcup_{i=1}^{c} Lower(X_i, B)$ $Upper(D, B) = \bigcup_{i=1}^{c} Upper(X_i, B)$.

Supposed that $a \in B$, importance degree definition of attribute is:

$$SIG(a, D, B) = \gamma(D, B) - \gamma(D, B - a)$$

Attribute reduction algorithm based on neighborhood rough set
Input: $NDT =< S, A = G \cup D, V, f >$ *and neighbourhood* δ
// δ is the threshold to control the size of the neighbourhood
Output:red; //attribute subset, namely reduction of G
(1) $\forall a \in G$ *:compute neighbourhood relation* N
(2) $red = \varnothing$ *;*
(3) for each $a_i \in G - red$ *Computing* $SIG(a_i, D, red) = \gamma(D, red \cup a_i) - \gamma(D, red)$
(4)Selecting a_k *satisfying* $SIG(a_k, D, red) = \max(SIG(a_i, D, red))$ *;*
(5)if $SIG(a_k, D, red) > 0$
Red=red $\cup a_k$ *;*
Go to (3)
Else
Return red;
End

2.2. Parameter Selection of SVM based on PSO

2.2.1. Support Vector Machine: SVM is a machine learning method based on statistical learning theory, and can solve higher dimension, nonlinearity, small sample problems using the principles of structural risk minimization [1]. Based on Mercer theorem, it transforms input space into a higher dimension characteristic space through proper nonlinear transformation, which makes searching linear regression optimal hyperplane in this characteristic space boil down to solving convex programming problem, and obtains global optimal solution.

For given sample points: $(x_1, y_1), \ldots, (x_l, y_l), x_i \in R^n, y_i \in \{-1, +1\}$ (2.1)

(1) Linear separable problem. The problem comes down to the following optimization problem.

$$F(w) = \frac{1}{2}\|w\|^2 \tag{2.2}$$

$$\begin{cases} (w \cdot x_i) + b \geq 1 & y_i = 1 \\ (w \cdot x_i) + b \leq -1 & yi = -1 \end{cases} \Leftrightarrow y_i[(w \cdot x_i) + b] \geq 1 \quad (i = 1, 2, \ldots, l)$$

Discriminant function: $f(x) = w_0 \cdot x + b_0$ (2.3)

To solve (2.2), transform the above-mentioned problem into its dual problem according to Wolfe theorem:

$$\begin{aligned} &MaxW(a) = \sum_i \alpha_i - \frac{1}{2} w(\alpha) \cdot w(\alpha) \\ &subject\ to\ \alpha_i \geq 0, \sum_i a_i y_i = 0 \end{aligned} \tag{2.4}$$

(2) Linear non-separable problem. Introduce Slack Variable x_i , transform (2.2) to the following problem.

$$\begin{aligned} &Min(\frac{1}{2}\|w\|^2 + C\sum_i \xi_i) \\ &Subject\ to\ y_i(w \cdot x_i + b) \geq 1 - x_i, x_i \geq 0 \end{aligned} \tag{2.5}$$

Similarly, attain corresponding dual problem:

$$\begin{aligned} &MaxW(a) = \sum_i \alpha_i - \frac{1}{2} w(\alpha) \times w(\alpha) \\ &subject\ to\ 0 \le \alpha_i \le C, \sum_i a_i y_i = 0 \end{aligned} \tag{2.6}$$

Solving (2.4) and (2.6) is a classical and constrained quadratic optimization problem and there have been many mature solving algorithms.

2.2.2. Particle Swarm Optimization: Particle Swarm Optimization was brought up by Eberhart in 1995 according to foraging behavior of bird flocking [24], which is an efficient optimization algorithm because of simple concept, convenient implementation, fast convergence rate and fewer parameter setting. In PSO algorithm, each individual is called a "particle" and actually each particle represents a potential solution. In a d-dimension target search space, each particle is regarded as a point in the space. Suppose that the swarm consists of m particles. m is also called swarm scale and its value will impact operational speed and convergence of the algorithm.

The mathematical description of PSO algorithm is: there is swarm $X = (x_1, ..., x_i, ..., x_m)$ consisting of m particles in a d–dimension space, where the location of i particle is $x_i = (x_{i1}, x_{i2}, ..., x_{iD})^T$ and its speed is $V_i = (v_{i1}, v_{i2}, ..., v_{id}, ..., v_{iD})^T$. Its individual extreme value is $p_{ibest} = (p_{i1}, p_{i2}, ..., p_{iD})^T$ and swarm extreme value is $p_{gbest} = (p_{g1}, p_{g2}, ..., p_{gD})^T$, according to the principle of setting current values as the best Particle x_i will change its rate and location in light of (2.7) and (2.8).

$$v_{ij}(t+1) = w \times v_{ij}(t) + c_1 r_1(t)(p_{ibest}(t) - x_{ij}(t)) + c_2 r_2(t)(p_{gbest}(t) - x_{ij}(t)) \tag{2.7}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{2.8}$$

Here $j = 1, 2, \cdots D$ and $i = 1, 2, \cdots m$, m is swarm scale, t is current evolving algebra and r_1, r_2 are random numbers located among $[0,1]$; c_1, c_2 are accelerating factor and w is weight vector.

2.2.3. SVM Parameter Optimization based on PSO: Many researches demonstrate that kernel function, kernel parameter and penalty parameter are essential factors for impacting its classification performance. Here, penalty coefficient C reflects penalty degree of the algorithm for outlier sample data and its value affects complexity and stability of the model. If C is too small, its penalty for outlier sample points is small while training error become bigger. However, if C is too big, learning accuracy will be improved correspondingly, but generalization capacity of the model become worse. C value impacts "outlier points" processing (abnormal data points under the influence of noise) in samples, and selecting proper C can improve anti-interference capacity to ensure stability of the model.

Radial basis function (RBF) $K(x, y) = \exp(-\frac{\|x - y\|^2}{\sigma^2})$ is used most extensively as a kernel function with wide convergence domain, which is an ideal kernel function. Width coefficient σ of kernel function reflects correlation degree among support vectors. If σ is very small, correlation among support vectors is relaxation, learning machine is more complex and generalization ability can't be ensured; If σ is too big, influence among support vectors is so strong that it is difficult to realize enough accuracy.

Hence, penalty parameter C and kernel parameter σ are important factors impacting classification accuracy of SVM. This paper will use PSO to search a kind of penalty parameters and kernel parameters $\{C, s\}$ to gain highest classification accuracy of SVM. Cross-validation error of SVM in training set is defined fitness function in the algorithm. Specific steps of this algorithm can be listed as follows:

Step1 Swarm initialization and parameter setting: swarm particle initialization $\{C,s\}$, swarm scale m, iteration number T, accelerating factor c_1,c_2 and weight factor w;

Step2 Calculate fitness values of initialization particles, then compare them and select particles with the best fitness (namely particles with the smallest fitness) and regard corresponding individual extreme value as initial global extreme value p_{gbest};

Step3 Implement iteration calculation and update location and speed of particles according to (2.7) and (2.8);

Step4 Calculate fitness of current particles;

Step5 Compare fitness value of each particle with corresponding value of its p_{ibest}, if better, update p_{ibest}, otherwise reserve old value;

Step6 Compare p_{ibest} of updated each particle with global extreme value p_{gbest}, if better, update p_{ibest}, otherwise reserve old value;

Step7 Determine if end condition is matched, when it reaches the largest iteration number T or obtained optimal particle no longer changes, terminate iteration, otherwise return to step3.

2.3. Idea and Framework of this Algorithm

2.3.1. Production of base Classifiers: Multiple classifiers ensemble is an effective way to improve classification accuracy. In order to obtain the better ensemble effect, each base classifier has enough diversity and can form complementary, so the diversity among the base classifiers is ensembled key. Through strengthening diversity among base classifiers, producing training subset with bigger diversity, it can be an effective method. Bagging is an ensemble algorithm based on bootstrap technology—randomly extract a certain amount of samples from original training set to make up training subset, scale of training subset is equivalent to original training set and training samples can be selected repeatedly [12]. By this way, some samples in original training set may occur in new training subset repeatedly, while some other samples may never occur, therefore, training subset with greater diversity can be generated.

For redundancy attribute and noise in intrusion detection data, neighborhood rough set with different radiuses was used to reduce attribute in each above produced bootstrap training subset. On the one hand, attribute reduction can eliminate redundancy attribute and noise to obtain base classifiers with higher precision; On the other hand, reduction for bootstrap training subset based on neighborhood rough set with different radiuses is equal to mapping training subset to different characteristic space, by this way, it can increase diversity of training subset to finally obtain base classifiers with higher accuracy and greater diversity.

For the above training subset after reduction, PSO was used to optimize penalty coefficient C and kernel parameter σ to obtain classifiers with higher accuracy and better stability.

2.3.2. Combination of Base Classifiers: Combination of base classifiers is also an important factor for impacting ensemble performance. Voting method is a combination method that is commonly used and easy to understand, it mainly includes majority voting and weighed voting.

Majority voting can be viewed as a special case that in weighed voting weighed values of all base classifiers are equal. For majority voting method, when two or more than two class labels obtain the biggest voting number at the same time, decision conflict occurs, therefore some methods are needed to solve conflict *e.g.*, plurality voting method based

on threshold value. Many researches demonstrate that majority voting is commonly ensemble strategy, but not the best because its result is impossibly better than any base classifier.

Weighed voting method can obtain better discrimination compared to majority voting method. Weight distribution of base classifiers is related to their prediction precision in training set, set i th classifier weight as b_i, when $b_i = \frac{\log p_i / (1- p_i)}{\sum_{i=1}^{L} \log p_i / (1- p_i)}$, its final decision is the sample belonging to class k. Here, p_i is prediction precision of i th classifier in training set, adopt b_i and adequately use prior information of all classifiers to maximize discrimination accuracy of the ensemble system.

2.3.3. The Step of this Algorithm

Input: training set $s_1 = \{(x_1, y_1) \ldots (x_m, y_m)\}$, test set $s_2 = \{(x_1, y_1) \ldots (x_n, y_n)\}$, base classifiers number T, radiuses of neighborhood rough set d_i ($i = 1,2,\cdots, T$)

Output: ensemble classifier f

Step1: for $i = 1:T$

(1) Resample from the training set s_1 to generate the bootstrap training subset s_1^i;

(2) Reduce attributes in training subset s_1^i using neighborhood rough set with radiuses d_i to produce reduced training subset s_1^{i*};

(3) PSO was used to optimize SVM parameters in s_1^{i*} to get optimal parameter combination $\{C_i, s_i\}$;

(4) Set optimal parameter $\{C_i, s_i\}$ and use SVM to train training set s_1^{i*} to produce base classifier f_i;

End

Step2: T base classifiers were ensemble by weighted averaging method.

3. Emulation Experiment

3.1. Experiment Data

We chose 10% data set of KDD CUP 99 data set as experiment data set. 10% data set contained training set and test set. 1000 samples were randomly extracted from training set of 10% data set as training data, which including 195 intrusion data, including 4 kinds of attacks (11 types in all)—Dos attack: 40 ueptne attacks, 90 smurf attacks and 6 back attacks; probing attack: 10 ipsweep attacks, 10 portsweep attacks and 15 satan attacks; U2R attack: 4 buffer_overflow attacks and 4 rootkit attacks; R2L attack: 4 waremster attacks, 6 Guess_passwd attacks and 6 warezclient attacks. 800 samples were randomly extracted from training set of 10% data set as test data, which had 100 known attacks, and 100 unknown attacks were added to verify detection effect of the algorithm for unknown attacks

Kinds of attacks that didn't exist in training set were added to test data set to verify detection capacity of the algorithm of this paper for unknown attacks, including 4 kinds (10 types in all)—Dos attack: 10 teardrop attacks, 20 pod attacks and 5 land attacks; probing attack: 10 nmap attacks; U2R attack: 2 loadmodule attacks and 2 perl attacks; R2L attack: 2 spy attacks, 2 phf attacks, 5 imap attacks, 2 multihop attacks and 2 ftp_write attacks, while test set data contained 21 types of attacks totally.

3.2. Standard of Evaluating Algorithm

Intrusion detection system mainly has 2 indexes for evaluating its performance:

$$\text{Detection Rate} = \frac{\text{attack sample number detected correctly}}{\text{total attack sample number}}$$

$$\text{False Alarm Rate} = \frac{\text{normal sample number judged attack wrongly}}{\text{total normal sample number}}$$

3.3. Experiment Methods

The experiments were done repeatedly for 50 times to research stability of the algorithm ,and the average value was taken as experiment result.

Algorithm 1: SVM (SVM parameters were generated randomly in algorithm 1)

Algorithm 2: Bagging+SVM (SVM parameter were generated randomly in algorithm 2)

Algorithm3:Bagging+Neighborhood-RS+ SVM (SVM parameter were generated randomly in algorithm 3)

Algorithm 4 (the algorithm of this paper): Bagging+Neighborhood-RS +PSO+SVM

3.4. Result and Analysis of the Experiment

3.4.1. The Relation of Neighborhood Rough Set Radiuses and Intrusion Detection Precision: In algorithm 3 and the algorithm 4, radiuses of neighborhood rough set must be set and difference of radiuses will result in diversity of classification accuracy, here radiuses of neighborhood were set between 0.01and 1, with step length as 0.01, 100 attribute subsets were obtained using 100 different radiuses of neighborhood.

Figure1 and Figure 2 visually show relation between detection rate and radiuses, as well as false alarm rate and radiuses respectively. Figure 1 indicates when radiuses are [0.01,0.24], its detection rate is always about 81%, while radiuses increase to [0.27,0.48], detection rate will increase to about 91%,it demonstrates that radiuses of neighborhood influence detection rate greatly. In Figure 2, when neighborhood radiuses are [0.01, 0.48], its false alarm rate is always about 2%, but for [0.48, 0.6], false alarm rate increases by a large scale.

According to the above analysis, when neighborhood radiuses are [0.3,0.48] and [0.6,0.66], its detection rate is higher while false alarm rate reaches to a lower status, this conclusion can be used as a source of reference for using neighborhood rough set.

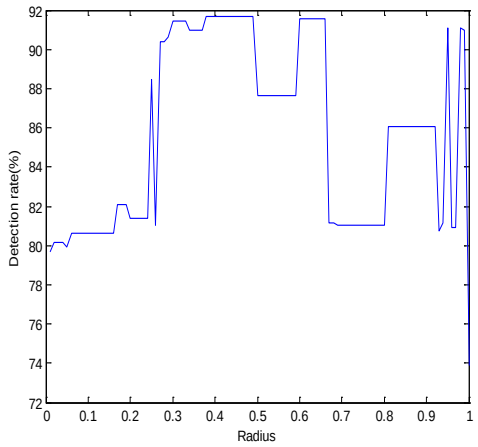

Figure 1. Relation between Radiuses of Neighborhood Rough Set and Intrusion Detection Rate

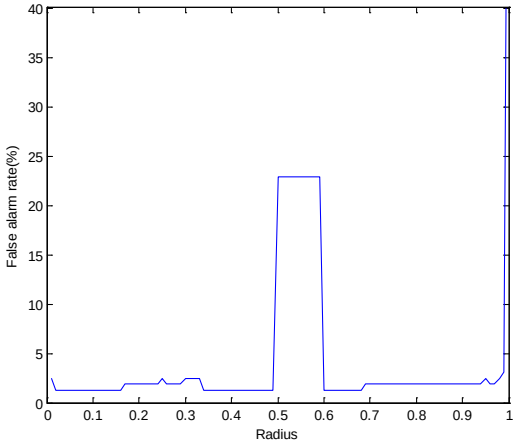

Figure 2. Relation between Radiuses of Neighborhood Rough Set and Intrusion False Alarm Rate

3.4.2. Compare Detection Precision of Different Algorithms: The parameters setting in PSO can be listed as follows: c_1 was 1.5, c_2 was 1.7, max evolving algebra was

200, swarm scale was 20, k was 0.6, elasticity coefficient in front of rate in rate update formula was 1,elasticity coefficient in front of rate in swarm update formula was 1,Cross Validation parameter of SVM was 3, range of parameter C was [0.1,100] and s was [0.01,1000].

Table 1 gives experiment results of 4 different algorithms. Result shows that both average value and optimal value of detection rate by the algorithm of this paper are highest in 4 algorithms, while its false alarm rate is lowest. Compared to algorithm 1, detection rate of algorithm 2 increases by about 3%, which indicates that bagging is effective and can improve intrusion detection effect; Compared to Algorithm 2, detection rate of Algorithm 3 increases by about 4%, which demonstrates that using neighborhood rough set to reduce attributes can not only eliminate redundancy attribute and noise, but also through neighborhood rough set directly processing continuous data avoid information loss during the process of traditional rough set discretization to improve intrusion detection effect; detection rate of algorithm 4 is about 2.5% higher than that of algorithm 3, and its detection performance is better than that of others, mainly because optimized SVM parameter by PSO strengthens generalization performance of classifiers greatly and finally results in large improvement of detection performance. Especially, when SVM parameter $C = 1.47, s = 4.09$ and neighborhood rough set radiuses is 0.31, optimal value of its detection rate reaches 93.17% and false alarm rate reaches 0.81%.

Table 1. Experiment Results of 4 algorithms (%)

	Algorithm 1			Algorithm 2			Algorithm 3			Algorithm 4		
	Average	Optimal	Ensemble	Average	Optimal	Ensemble	Average	Optimal	Ensemble	Average	Optimal	Average
Detection Rate（%）	80.12	87.36	86.89	83.15	90.04	91.14	87.01	92.01	93.43	89.21	93.17	80.12
False Alarm Rate（%）	2.12	1.64	1.94	1.97	1.29	1.62	1.65	1.01	0.92	1.19	0.81	2.12

Note:

(1) In Algorithm 1, "Average" refers to average result of 50 experiments (by Algorithm 1) , "Optimal" refers to optimal result of 50 experiments (by Algorithm 1);

(2) In Algorithm 2 (or 3 or 4), "Average" demonstrates that: first do an experiment by Algorithm 2 (or 3 or 4), calculate average value of all base classifiers and finally do 50 experiments and calculate average value again; "Ensemble" refers to average result of 50 experiments by Algorithm 2 (or 3 or 4); "Optimal" refers to optimal result of 50 experiments by Algorithm 2 (or 3 or 4).

Figure 3 visually shows comparison between ensemble value and average value of three ensemble algorithms (Algorithm 2, 3 and 4). It is clear that the ensemble method always gets better classification result than single classifier, while results of Algorithm 4 are obviously superior to those of Algorithm 2 and 3, which demonstrates that the algorithm of this paper improves detection effect.

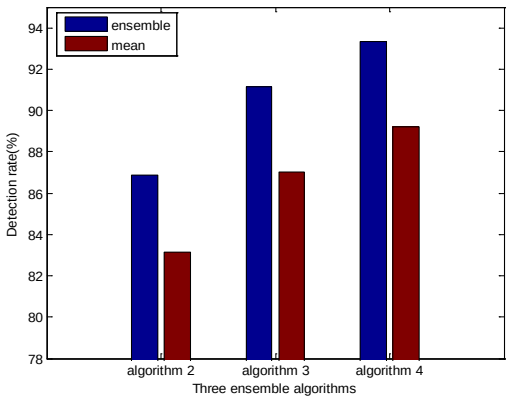

Figure 3. Compare Ensemble Result and Average Result of three Ensemble Algorithms

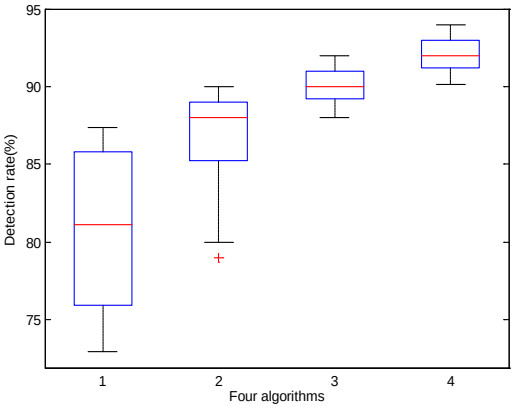

Figure 4. Boxplot of 50 Times of Experiment Results from 4 Algorithms

Table 2 shows detection rate of four algorithms for known and unknown attacks. We find that the algorithm of this paper shows large improvement for known and unknown attacks, it demonstrates this algorithm is effective in detecting unknown attacks and has higher generalization performance and robustness.

Table 2. Average Detection Rate of 4 Algorithms for Known and Unknown Attacks

Attack Kind	Known Attack (%)				Uknown Attack (%)			
	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4
DoS	80.02	85.19	90.83	93.24	79.93	84.98	90.69	93.17
Probing	81.53	84.98	91.18	93.91	81.21	85.51	91.62	94.26
U2R	80.79	84.78	90.74	92.47	81.03	84.79	90.29	92.30
R2L	82.66	86.10	91.99	94.89	82.50	85.93	92.12	95.03

3.4.3. Stability Analysis: Figure4 displays stability of four different algorithms. We can find that: detection accuracy of Algorithm 2 is improved largely compared to Algorithm 1,but its stability is still poor; detection accuracy and stability of Algorithm 3 far exceeds those of Algorithm 2, which shows that using attribute reduction of neighborhood rough set can improve detection accuracy as well as strengthen stability of the algorithm; detection accuracy of Algorithm 4 is improved compared to Algorithm 3 and its stability is a little better than Algorithm 3, because optimizing SVM parameter by PSO can produce base classifiers with smaller generalization error and finally improve ensemble performance and stability.

According to the above analysis, it is clear that detection effect and stability of the algorithm of this paper are superior to those of other algorithms, and it has better generalization performance and robustness.

4. Conclusions

Intrusion detection data often have noise and redundancy attribute and some attributes data are continuous, in order to solve the problem of information loss during continuous attribute discretization, this paper used the model of neighborhood rough set to reduce attributes ,while kernel function parameter and penalty parameter of SVM are optimized by PSO. Through emulation experiments of KDD99 data set, the result indicates that the algorithm of this paper can improve intrusion detection rate while reduce false alarm rate, and has higher generalization performance and robustness.

Acknowledgements

This paper was supported by Scientific Research Program Funded by Shaanxi Provincial Education Department (No.12JK0864) and Scientific Research Program Funded by Shaanxi University of Technology(No.SLGKY12-01、SLGKY13-41).

References

[1] D. Sanchez, "Advanced support vector machines and kernel methods",Neuro Computing, vol. 1, no. 55, **(2003)**.
[2] G. Y. Chen, Q. L. Zhang and X. Li, "SVM classification-based intrusion detection system", Journal of China Institute of Communications, vol. 5, no. 23, **(2002)**.
[3] X. Rao, C. X. Dong and S. Q. Yang, "An intrusion detection system based on support vector machine", Journal of Software, vol. 4, no. 14, **(2003)**.
[4] T. Chen, "Parameters optimization of Support vector regression based on differential evolution", Computer Simulation, vol. 6, no. 28, **(2011)**.
[5] T. Chen, "Parameters selection of support vector machine based on differential evolution", Computer Engineering and Applications, vol. 4, no. 54, **(2010)**.
[6] T. Chen, "Algorithm of selective SVM ensemble", Computer Engineering and Design, vol. 5, no. 32, **(2011)**.
[7] T. Chen and Z. L. Hong, "A combined svm ensemble algorihm based on KICA and KFCM", Advances in Intelligent and Soft Computing, vol. 4, no. 12, **(2012)**.
[8] T. Chen, "Selective SVM ensemble based on accelerating genetic algorithm", Application Research of Computers, vol. 2, no. 32, **(2011)**.
[9] T. Chen and Z. L. Hong, "SVM ensemble based on boosting and KFCM", Advances in Intelligent and Soft Computing, vol. 6, no. 12, **(2012)**.
[10] T. Chen, "Selective SVM ensemble based on accelerating genetic algorithm", Application Research of Computers, vol. 2, no. 32, **(2011)**.
[11] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation and active learning", Advances in Neural Information Processing Systems, vol. 7, **(1995)**.
[12] L. Breiman, "Bagging predictors", Machine Leaming, vol. 2, no. 24, **(1996)**.
[13] Z. Pawlak, "Rough sets theoretical aspects of reasoning about data", Kluwer Academic Publishers, London, **(1991)**.
[14] Y. R. Zhang, M. Xuan and S. P. Xiao, "An anomaly intrusion detection technique of support vector machine based on rough set attribute reduction", Computer Science, vol. 6, no. 33, **(2006)**.
[15] X. B. Zhao, R. Z. Jin and M. Gu, "Adaptive intrusion detection algorithm based on rough sets", Journal of Tsinghua University, vol. 7, no. 48, **(2008)**.
[16] Q. H. Hu, D. R. Yu and Z. X. Xie, "Numerical attribute reduction based on neighborhood granulation and rough approximation", Journal of Software, vol. 3, no. 15, **(2008)**.
[17] Q. H. Hu, H. Zhao and D. R. Yu, "Efficient symbolic and numerical attribute reduction with neighborhood rough sets", Pattern Recognition and Artificial Intelligence, vol. 6, no. 21, **(2008)**.
[18] Q. H. Hu, D. R. Yu and J. F. Liu, "Neighborhood rough set based heterogeneous feature subset selection", Information Sciences, vol. 18, no. 178, **(2008)**.
[19] O. Chappelle, "Choosing multiple parameters for support vector machines", Machine Learning, vol. 1, no. 46, **(2002)**.
[20] C. H. Zhang and L. C. Jiao, "Automatic parameters selection for SVM based on GA", Proceedings of the 5th World Congress on Intelligent Control and Automation. Piscataway.NJ, **(2004)** June 11-13.
[21] J. Yang, "Optimization of features with weight and model parameters of SVM based on genetic algorithm", Computer Simulation, vol. 9, no. 21, **(2008)**.
[22] L. Qi, "Parameters selection of support vector machine based on ant colony algorithm", System Simulation Technology, vol. 11, no. 34, **(2008)**.
[23] L. J. Qi, S. H. Ni and C. Su, "A study of optimizing SVM with ant colony algorithm and its application", Computer Simulation, vol. 11, no. 57, **(2009)**.
[24] Y. Fukuyama, "Fundamentals of particle swarm techniques", Modern Heuristic Optimization Techniques with Applications to Power Systems, IEEE Power Engineering Society, **(2002)**.
[25] X. G. Shao, H. Z. Yang and G. Chen, "Parameters selection and application of support vector machines based on particle swarm optimization algorithm", Control Theory and Applications, vol. 5, no. 23, **(2006)**.

Author

Hui Zhao received his M.S. in Computer sciences (2011) from Northwest University. Now he is full instructor of informatics at school of mathematics and computer sciences department, Shaanxi University of Technology. His current research interests include different aspects of network security and data mining.