Journal of Computational Information Systems 8: 8 (2012) 3237–3244
Available at http://www.Jofcis.com

A Novel Approach of KPCA and SVM for Intrusion Detection

Fangjun KUANG [1,2], Weihong XU [1,3,*], Siyang ZHANG [2], Yanhua WANG [3], Ke LIU [2]

[1] *School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China*

[2] *Department of Computer Science and Technology, Hunan Vocational Institute of Safety & Technology, Changsha 410151, China*

[3] *College of Computer and Communications Engineering, Changsha University of Science and Technology, Changsha 410077, China*

Abstract

A novel hybrid approach of kernel principal component analysis (KPCA) and improved support vector machine (SVM) using genetic optimization algorithm (GA) is proposed for intrusion detection. The original data is normalized preprocessing, KPCA is used as a preprocessor of SVM to extract the principal features of the normalized data, SVM is used to classification forecasting by finding the most appropriate kernel function and the optimal parameters with GA. Additionally, the proposed KPCA SVM with GA that can automatically determine the optimal parameters was tested on intrusion detection. Compared with other detection algorithms, Experimental results demonstrated that the proposed KPCA SVM model performed higher predictive accuracy, faster convergence speed and better generalization, implying that the proposed model is successful for intrusion detection.

Keywords: Kernel Principal Component Analysis; Support Vector Machines; Genetic Algorithms; Intrusion Detection

1 Introduction

With the development of computer and communication technologies, network security has been a challenge for both the researchers and enterprises. Intrusion detection system (IDS) is one of the key methods to protect network security. Capabilities of intrusion detection technologies have great importance with the performance of IDS. Researches always want to find an intrusion detection technology with better detection accuracy and less training time.

In nature, the intrusion detection can be seen as a classification problem, to distinguish between the normal activities and the malicious activities. In order to solve the problem, two main parts

*Corresponding author.
Email address: xwhxdcs@126.com (Weihong XU).

April 2012

should be conducted which are detection model setup and intrusion feature extraction. Using machine learning theory such as genetic algorithm [1], neural network [2], clustering methods [3], and support vector machine (SVM) [4] to setup the intrusion detection model had got better performance than the traditional intrusion detection technologies. Among the methods mentioned above, SVM is an effective one, which is a well-known classifier tool based on small sample learning. SVM has manifested its robustness and efficiency in the network action classification, it therefore becomes a popular method widely used in IDS [5].

At the same time, there are also some progresses made in the feature extraction field. Andrew H. Sung ranked the importance of the 41 features for the five categories in KDD CUP99 datasets by deleting one feature at a time when adopting the SVM and neural network as the classification methods [6]. Melanie J. Middlemiss used genetic algorithm to do the feature extraction for intrusion detection [7]. The researches about combination of feature analysis technologies and classification algorithms for IDS also have been working on. Taeshik Shon proposed a machine learning framework for intrusion detection using SVM and GA which used GA to extract intrusion features and SVM to classify [8].

This paper presents a new approach for network intrusion detection. In the proposed method, we use the KPCA maps the high dimension features in the input space to a new lower dimension eigenspace, and then use the GA for parameter selection of KPCA SVM model to estimate whether the action is an attack. The remainder of this paper is structured as the following. In section 2, the KPCA SVM with GA model for intrusion detection is discussed in detail. An experimental result to evaluate the proposed model is displayed in section 3, and a comparison among single SVM, PCA SVM, KPCA SVM and KPCA SVM with GA is shown. Section 4 provides the conclusion, discussion of proposed IDS and future work are described.

2 KPCA SVM with GA Model

2.1 Kernel principal component analysis

Principal Component Analysis (PCA) [9] is a common method applied to dimensionality reduction and feature extraction. PCA method only can extract the linear structure information in the data set but can not extract this nonlinear structure information. Kernel Principal Component Analysis (KPCA) is an improved PCA, which extracts principal components by adopting a nonlinear kernel method [10], [11], and [12]. A key insight behind KPCA is to transform the input data into a high dimensional feature space F in which PCA is carried out, and in implementation, the implicit feature vector in F does not need to be computed explicitly, while it is just done by computing the inner product of two vectors in F with a kernel function.

Let $x_1, x_2, \cdots, x_n \in R^m$ be the n training samples for KPCA learning. By the nonlinear mapping function Φ, the measured inputs are extended into the hyper-dimensional feature space F as follows

$$\Phi : x \in R^m \to \Phi(x_i) \in F^h \tag{1}$$

Where $\Phi(x_i)$ is sample of F and $\sum_{i=1}^{m} \Phi(x_i) = 0$.

The mapping of x_i is simply noted as $\Phi(x_i) = \Phi_i$. The covariance matrix of the sample in the

feature space F can be constructed by

$$C_\Phi = \frac{1}{n}\sum_{i=1}^{n}(\Phi_i - m_\Phi)(\Phi_i - m_\Phi)^T \tag{2}$$

Here, column vector $m_\Phi = \sum_{i=1}^{n}\frac{\Phi_i}{n}$. Nonzero eigenvalues of covariance matrix C_Φ are positive, and Schölkopf [13] et al. has suggested the following way to find these positive eigenvalues.

It is easy to see that every eigenvector v of C_Φ can be linearly expanded by

$$v = \sum_{i=1}^{n}\alpha_i\Phi_i \tag{3}$$

To obtain the coefficients $\alpha_i(i = 1, 2, \cdots, n)$, a kernel matrix K with size $n \times n$ is defined, and its elements are determined by virtue of kernel tricks.

$$K_{ij} = {\Phi_i}^T\Phi_j = (\Phi_i \bullet \Phi_j) = k(x_i, x_j) \tag{4}$$

Here, $k(x_i, x_j)$ is the calculation of the inner product of two vectors in F with a kernel function. Centralize K by

$$\overline{K} = K \cdot 1_n K \cdot K 1_n + 1_n K 1_n \tag{5}$$

Here, matrix $1_n = (1/n)_{n\times n}$.

Let column vectors $\gamma_i(i = 1, 2, \cdots, p; 0 < p \le n)$ be the orthonormal eigenvectors of $\overline{K}$ corresponding to the p largest positive eigenvalues $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_p$. The orthonormal eigenvectors β_i of C_Φ corresponding to the p largest positive eigenvalues $\lambda_1, \lambda_2, \cdots \lambda_p$ are

$$\beta_i = \frac{1}{\sqrt{\lambda_i}}(\Phi_1, \Phi_2, \cdots, \Phi_n)\gamma_i, i = 1, 2, \cdots, p \tag{6}$$

To a new column vector sample x_{new}, the mapping to the feature space is $\Phi(x_{new})$. The projection of x_{new} onto the eigenvectors $[\beta_i(i = 1, 2, \cdots, p)$ is $t = (t_1, t_2, \cdots, t_p)^T$, and it is also called KPCA transformed feature vector.

$$t = (\beta_1, \beta_2, \cdots, \beta_p)^T\varphi(x_{new}) \tag{7}$$

The ith KPCA-transformed feature t_i can be obtained by

$$\begin{gathered} t_i = {\beta_i}^T\varphi(x_{new}) = \tfrac{1}{\sqrt{\lambda_i}}{\gamma_i}^T(\Phi_1, \Phi_2, \cdots, \Phi_n)^T\varphi(x_{new}) \\ = \tfrac{1}{\sqrt{\lambda_i}}{\gamma_i}^T[k(x_1, x_{new}), k(x_2, x_{new}), \cdots, k(x_n, x_{new})]^T, i = 1, 2, \cdots, p \end{gathered} \tag{8}$$

By using Eq. (8), the KPCA-transformed feature vector of a new sample vector can be obtained.

2.2 KPCA SVM model

After feature extraction using KPCA, the training data points can be expressed as (t_1, y_1), (t_2, y_2), $\cdots$, (t_p, y_p), $t_i \in R^n$ $(n < m)$ is the transformed input vector, $y_i \in R^n$ is the target value. In the ε-SVM classification [4], [14], the goal is to find a function $f(t)$ that has at most ε deviation from the actually obtained targets y_i for all the training data, and at the same time, is as flat as possible. The ε-insensitive loss function reads as follows

$$e(f(t)-y) = \begin{cases} 0, |f(t)-y| \le \varepsilon \\ |f(t)-y| - \varepsilon, otherwise \end{cases} \tag{9}$$

To make the SVM regression nonlinear, this could be achieved by simply mapping the training patterns by $\Phi : t \in R^n \to F$ into some high dimensional feature space F. Suppose $f(t)$ takes the following form:

$$f(t) = w \cdot t + b \tag{10}$$

A best fitting function $f(t)$ is estimated in feature space F as follows

$$f(t) = (w' \bullet \Phi(t)) + b \tag{11}$$

Where, "$\bullet$" denotes the dot product in the feature space F. In the case of (11), flatness means that one can seek small w'. Formally this problem can be written as a convex optimization problem by requiring the follows:

$$\begin{aligned} \min imize \quad & \tfrac{1}{2}\|w\|^2 + C\sum_{i=1}^{p}(\xi_i + {\xi_i}^*) \\ subject\ to \quad & y_i - (w'\Phi(t_i) + b) \le \varepsilon - \xi_i \\ & (w'\Phi(t_i) + b) - y_i \le \varepsilon - {\xi_i}^* \\ & \xi_i, {\xi_i}^* \ge 0, i = 1, 2, \cdots, p; C > 0 \end{aligned} \tag{12}$$

Where ξ_i and ${\xi_i}^*$ are slack variables, the constant C determines the trade-off between the flatness of $f(t)$ and the amount up to which deviations large than ε are tolerated. By constructing the Lagrangian function, the dual problem can be given as follows:

$$\begin{aligned} \min imize \quad & -\tfrac{1}{2}\sum_{i,j=1}^{p}(\alpha_i - {\alpha_i}^*)(\alpha_j - {\alpha_j}^*)(x_i \cdot x_j) \\ & + \sum_{i=1}^{p} y_i(\alpha_i - {\alpha_i}^*) - \varepsilon\sum_{i=1}^{p}(\alpha_i + {\alpha_i}^*) \\ subject\ to \quad & \sum_{i=1}^{p}(\alpha_i - {\alpha_i}^*) = 0 \\ & 0 \le \alpha_i, {\alpha_i}^* \le C \end{aligned} \tag{13}$$

Where α_i and α_i^* are the Lagrange multiplier coefficients for the ith training example of regression, and obtained by solving the dual optimization problem in support vector learning [4], [14]. The non-negative coefficients α_i and α_i^* are bounded by a user-specified constant C. The training example for which $\alpha_i \ne \alpha_i^*$ is corresponded to the support vectors. At the optimal solution from (13), the regression function takes the following form:

$$f(t) = \sum_{i=1}^{p}(\alpha_i - {\alpha_i}^*)K(t_i, t_j) + b \tag{14}$$

Where $K(.,.)$ a kernel function, b is found by the Karush-Kuhn-Tucker conditions at optimality.

According to [4], any symmetric positive semi-definite function, which satisfies Mercer's conditions, can be used as a kernel function in the SVM context. Mercer's conditions can be written as follows:

$$\begin{aligned} &\iint K(s,z)g(s)g(z)dsdz > 0, \quad \int g^2(s)ds < \infty \\ &where \quad K(s,z) = \sum_{i=1}^{\infty} \alpha_i \psi(s)\psi(z) \\ &\qquad \alpha_i \geq 0 \end{aligned} \tag{15}$$

In this paper, the radial basis kernel function (RBF) used in the SVM classification method is as follows:

$$K(s,z) = \exp(\frac{-\|s-z\|^2}{\sigma^2}) \tag{16}$$

2.3 GA for parameter selection of KPCA SVM model

GA has been considered with increasing interest in a wide variety of applications [15]. These algorithms are used to search the solution space through simulated evolution of "survival of the fittest". These are used to solve linear and nonlinear problems by exploring all regions of state space and exploiting potential areas through selection, crossover and mutation operations applied to individuals in the population [15].

In this paper, the selection of the three positive parameters, σ, ε and C of KPCA SVM model is important to the accuracy of the classification for intrusion detection. Therefore, genetic algorithms are used in the proposed KPCA SVM model to optimize the parameter selection. A negative mean absolute percentage error (MAPE) is used as the fitness function for evaluating fitness [16]. The MAPE is as follows:

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{a_i - f_i}{a_i}\right| \times 100\% \tag{17}$$

Where a_i and f_i represent the actual and forecast values and N is the number of classification forecasting periods. GA is used to yield a smaller MAPE by searching for better combinations of three parameters in KPCA SVM, which is described below:

Step 1: The creation of an initial population of chromosomes. The three free parameters σ, ε and C are encoded in a binary format, and represented by a chromosome.

Step 2: The fitness of each chromosome is evaluated by the cross-validated predictive accuracy of the SVM model. Based on fitness functions, chromosomes with higher fitness values are more likely to yield offspring in the next generation. The roulette wheel selection principle is applied to choose chromosomes for reproduction.

Step 3: Crossover and mutation. Mutations are performed randomly by converting a '1' bit into a '0' bit or a '0' bit in to a '1' bit. The single-point crossover principle is employed. Segments of paired chromosomes between two determined break-points are swapped. The rates of crossover and mutation are probabilistically determined. In this study, the probabilities of crossover and mutation are set to 0.5 and 0.1, respectively.

Step 4: A new population is created for the next generation.

Step 5: If the number of generations equals a given scale, then stop; else go to step2.

Step 6: Obtain the optimal parameters σ, ε and C of the KPCA SVM model.

3 Experiments

The intrusion detection belongs to classification problems in essence, it discriminates abnormal data from anomaly data. Since the tested intrusion data is of a high dimension and contains many noise attributes, in order to improve the accuracy and speed of the system, the original data is normalized preprocessing, then we extract the principal components, which are easy for classification, from the normalized data using KPCA, finally, SVM is used to classification forecasting by finding the most appropriate kernel function and the optimal parameters by GA. The framework of a new intrusion detection model based on KPCA and SVM is shown as figure 1.

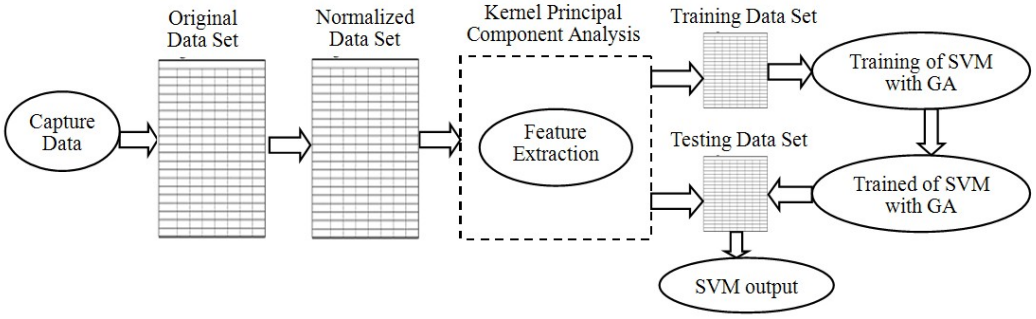

Fig. 1: The framework of intrusion detection model based on KPCA SVM with GA

In order to evaluate the performance of KPCA SVM with GAs model for intrusion detection, the KDD CUP99 datasets were selected in our simulation. The datasets can be downloaded at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. The original datasets consist of the training datasets and testing datasets. The datasets have five categories which are normal, denial of service (DoS), unauthorized access from a remote machine (Remote to Local, R2L), unauthorized access to local supervisor privileges (User to Root, U2R) and Probe. Each connection record in the datasets has 41 various features, of which 34 are continuous attributes and 7 are discrete ones. In experiments, select samples randomly from the original training and testing datasets, use 6000 samples in our experiments, 4000 for training and 2000 for testing. Before the experiments, we changed the symbolic field of the data points into numeric values and transformed them into the normalized format.

The experiment was processed within a MATLAB R2009b environment, which was running on a PC powered by Pentium IV 3.0 GHz CPU and 3.0 GB RAM. In the experiments by the KPCA and SVM with GA method, RBF kernels were used for KPCA and RBF kernels were also adopted for SVM, GA method was used to select the optimal parameter of SVM and KPCA. KPCA was applied to feature extraction, this method aimed to map the high dimensional original input data to a lower dimensional eigensapce, which hold the principal features and abandons the subordinate and noise data. In this paper, we choose p eigenvectors, which correspond to the

first p biggest eigenvalues, to form the sub-eigenspace, satisfying:

$$\sum_{i=1}^{p} \lambda_i \Big/ \sum_{i=1}^{n} \lambda_i > 90\% \tag{18}$$

In order to find out the KPCA SVM with GA for intrusion detection performance, we compared the single SVM, PCA SVM, KPCA SVM and KPCA SVM with GA, the comparisons of experimental results were given in table 1. All the SVM methods adopt RBF as their kernel function, in normal SVM detection model, the parameters σ, ε and C are randomly selected. In proposed SVM detection model, the optimal parameters σ, ε and C are obtained by genetic algorithm.

Table 1: Comparison of different algorithms

Methods	Parameters			Train Accuracy (%)	Test Accuracy (%)	Runtime(s)	The number of features
	σ	ε	C				
single SVM	0.6451	0.0253	5.2780	99.725 (3989/4000)	89.2 (1784/2000)	926.922238	41
PCA SVM	0.7269	0.0047	7.4185	99.825 (3993/4000)	95.3 (1906/2000)	650.779577	8
KPCA SVM	0.3695	0.0032	4.8576	99.975 (3999/4000)	98.9 (1978/2000)	389.433005	4
KPCA SVM with GA	0.1088	0.0018	1.3520	99.975 (3999/4000)	99.2 (1984/2000)	407.918466	4

Table 1 showed that SVM for intrusion detection by feature extraction using PCA, KPCA had a good performance in accuracy and runtime than that without feature extraction. From this table it can also be worked out that when principal components extracted by using PCA were used as the inputs of SVM to perform the classification, the best accuracy was 95.3% while the number of features chosen was 8. However, when those extracted by KPCA were adopted to train the SVM, the maximum classification accuracy was 99.2% while the number of features chosen was 4. This result demonstrated that the features extracted by KPCA could provide more additional discriminatory information for improving classification performance than PCA. And, dimension reduction can improve the generalization performance of SVM. It can also see that the overall performance of KPCA SVM with GA is better than other detection methods.

4 Conclusion

In this paper, a localized KPCA to SVM with GAs for intrusion detection was proposed. These methods can remove time-shift sensitivity of SVM when extracting features by KPCA and optimizing the parameters, σ, ε and C with GA. The intrusion detection data was examined in the experiment. The simulation shows that SVM for intrusion detection by feature extraction using PCA, KPCA can achieve better generalization performance than that without feature extraction. This demonstrates the fact that dimension reduction can improve the generalization performance of SVM and SVMs parameters can be optimized using GAs. Furthermore, the experiment also shows that on intrusion detection data, KPCA perform is better than PCA. The reason lies in the fact that KPCA can explore higher order information of the original inputs than PCA. By using the kernel method to generalize PCA into nonlinear, KPCA also implicitly takes into account higher order information of the original inputs. More number of principal components could also be extracted in KPCA, eventually resulting in better generalization performance. For future work, we want to develop more algorithms of combining KPCA with some other classification methods for intrusion detection.

Acknowledgement

This research was partially supported by Science Foundation of Ministry of Education of China under Grant No. 208098, and Scientific Research Fund of Hunan Provincial Education Department of China under Grant No. 11C0209 and No. 09C1105. And the authors are grateful to the referees for their suggestions and comments.

References

[1] M. Ludovic. Genetic algorithm, a biologically inspired approach for security audit trails analysis. In Proceedings of the 12th International Conference on Computer Safety, 1993.

[2] J. Ryan, M. J. Lin and R. Miikkulainen. Intrusion detection with neural networks. Advances in Neural Information Processing Systems 10, Cambridge, MA: MIT Press, 1998.

[3] H. Shah, J. Undercoffer, A. Joshi. Fuzzy clustering for intrusion detection. In Proceedings of IEEE International Conference on Fuzzy Systems, 2003.

[4] C. Batur, L. Zhou, C. C. Chan. Support vector machines for fault detection. Proceedings of the 41st IEEE Conference on Detection and Control, Las Vegas, USA, 2002.

[5] C. F. Tsai, Y. F. Hsu, C. Y. Lin, W. Y. Lin, Intrusion Detection by Machine Learning: A Review. Expert Systems with Applications, vol.36, pages 11994 – 12000, 2009.

[6] A. H. Sung. Identify important features for intrusion detection using support vector machines and neural networks. In Proceedings of the 2003 Symposium on Applications and the Internet, 2003.

[7] M. J. Middlemiss, G. Dick. Weighted feature extraction using a genetic algorithm for intrusion detection. Evolutionary Computation, CEC'03, 3, pages 1669 – 1675, 2003.

[8] T. Shon, Y. Kim, C. Lee, J. Moon. A machine learning framework for network anomaly detection using SVM and GA. Proceedings of the 2005 IEEE Workshop on Information Assurance and Security, New York, USA, 2005.

[9] I. T. Jolliffe. Principle component analysis, Springer-Verlag, New York, 1986.

[10] Z. G. Chen, H. D. Ren, X J Du. Minimax probability machine classifier with feature extraction by kernel PCA for intrusion detection, Wireless Communications, Networking and Mobile Computing, pages 1 – 4, 2008.

[11] W. Z. Liao, J. S. Jiang. Image feature extraction based on kernel ICA, Image and Signal Processing, Proceedings, 2: pages 763 – 767, 2008.

[12] M. Ding, Z. Tian, and H. Xu, Adaptive kernel principal analysis for online feature extraction, Proc. World Acad. Sci., Eng. Technol., 59: pages 288 – 293, 2009.

[13] B. Schölkopf, A. Smola, and K. R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation, 10 (5): pages 1299 – 1319, 1998.

[14] D. Srivastava and L. Bhambhu, Data classification using support vector machine, J. Theoret. Appl. Inf. Technol., 12 (1): pages 1 – 7, 2010.

[15] K. S. Tang, K. F. Man, S. Kwong, Q. He. Genetic algorithms and their applications. IEEE Signal Processing Magazine, 13: pages 22 – 37, 1996.

[16] P. F. Pai, W. C. Hong. Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms, Electric Power Systems Research, 74: pages 417 – 425, 2005.