Applied Soft Computing 18 (2014) 178–184

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A novel hybrid KPCA and SVM with GA model for intrusion detection

Fangjun Kuang[a,b], Weihong Xu[a,c,*], Siyang Zhang[b]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210018, China
[b] Department of Electronic and Information Engineering, Hunan Vocational Institute of Safety & Technology, Changsha 410151, China
[c] College of Computer and Communications Engineering, Changsha University of Science and Technology, Changsha 410077, China

ARTICLE INFO

Article history:
Received 11 October 2012
Received in revised form
17 November 2013
Accepted 15 January 2014
Available online 30 January 2014

Keywords:
Intrusion detection
Kernel principal component analysis
Kernel function
Support vector machines
Genetic algorithm

ABSTRACT

A novel support vector machine (SVM) model combining kernel principal component analysis (KPCA) with genetic algorithm (GA) is proposed for intrusion detection. In the proposed model, a multi-layer SVM classifier is adopted to estimate whether the action is an attack, KPCA is used as a preprocessor of SVM to reduce the dimension of feature vectors and shorten training time. In order to reduce the noise caused by feature differences and improve the performance of SVM, an improved kernel function (N-RBF) is proposed by embedding the mean value and the mean square difference values of feature attributes in RBF kernel function. GA is employed to optimize the punishment factor C, kernel parameters σ and the tube size ε of SVM. By comparison with other detection algorithms, the experimental results show that the proposed model performs higher predictive accuracy, faster convergence speed and better generalization.

1. Introduction

Intrusion detection is one of the most essential things for security infrastructures in network environments, and it is widely used in detecting, identifying and tracking the intruders [1]. Capabilities of intrusion detection technologies have great importance with the performance of intrusion detection system (IDS). Researches always want to find an intrusion detection technology with better detection accuracy and less training time.

However, there are many problems in the traditional IDS, such as the low detection capability against the unknown network attack, high false alarm rate, and insufficient analysis capability and so on. In nature, intrusion detection can be seen as classification problem, to distinguish between the normal activities and the malicious activities. The concerned problems of machine learning are how the systems automatically improve the performance with the increase of experience, which is consistent with that of the IDS. Therefore, various machine learning methods are developed for intrusion detection, such as decision tree [1], genetic algorithm (GA) [2], neural network [3], principal component analysis (PCA) [4], fuzzy logic [5], K-nearest neighbor [6], rough set theory [7] and support vector machine (SVM) [8].

Among the methods mentioned above, SVM is an effective one, the main reason is that the distribution of different types of attacks is imbalanced, where the learning sample size of the low-frequent attacks is too small compared to the high-frequent attack. SVM is a margin-based classifier based on small sample learning with good generalization capabilities, which is frequently used in real world applications of classification [9]. It realizes the theory of VC dimension and principle of structural risk minimum, thus it does not have the over-fitting problem that artificial neural network cannot overcome. SVM has manifested its robustness and efficiency in the network action classification, and it is widely used in IDS as a popular method [10]. Eskin [11] addressed an unsupervised anomaly detection framework, and applied it in three unsupervised learning algorithms, including clustering method, K-nearest neighbor and SVM. Shon et al. [12] employed genetic algorithm (GA) for feature selection, and used SVM for intrusion detection. Srinoy [13] proposed an intrusion detection model using SVM and particle swarm optimization (PSO) which used PSO to extract intrusion features and SVM to classify. Fei et al. [14] proposed an incremental clustering method based on the density. Horng et al. [15] used the hierarchical clustering algorithm to provide the SVM with fewer, abstracted, and higher qualified training instances. To overcome the problem of uncertainty in IDS, Kavitha et al. [16] adopted a new technique known as neutrosophic logic (NL). Wu and Banzhaf [17] referred to the review of computational intelligence in intrusion

* Corresponding author at: School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210018, China.
Tel.: +86 18073101198.
E-mail addresses: kfjztb@126.com (F. Kuang), xwhxdcs@126.com (W. Xu), ztb731021@126.com (S. Zhang).

1568-4946/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2014.01.028

detection and applied numerical evaluation measures to quantify the performance of IDS. Kolias and Kambourakis [18] gave the survey of swarm intelligence in intrusion detection. Kuang et al. [19] proposed a SVM model based on kernel principal component analysis (KPCA) and GA, which used KPCA to extract intrusion features, and GA to optimize the parameter of SVM. Li et al. [20] put forward pipeline of data preprocess and data mining in IDS, and used gradually feature removal method to feature reduction and SVM to classify.

However, standard SVM still has some limitations, the performance depends on its parameters selection, and when the differences between the attributes of the sample are very big, using RBF in the training process will produce a large number of support vectors and the training time will be longer too. And two main parts should be conducted which are detection model set-up and intrusion feature extraction to get better performance.

To solve the above mentioned problems, we present a novel intrusion detection approach combining SVM and KPCA to enhance the detection precision for low-frequent attacks and detection stability. In the proposed method, KPCA maps the high dimension features in the input space to a new lower dimension eigenspace and extracts the principal features of the normalized data, and multi-layer SVM classifier is employed to estimate whether the action is an attack. In order to shorten the training time and improve the performance of SVM classification model, an improved radial basis kernel function (N-RBF) based on Gaussian kernel function is developed, and GA is used to optimize the parameters of SVM.

The rest of this paper is organized as follows. In Section 2, the proposed SVM classification model is described in detail. The classification procedure is presented to illustrate how to use the proposed SVM model for intrusion detection in Section 3.The experimental results are discussed in Section 4. Section 5 presents conclusion and future work.

2. Novel KPCA SVM classification model

2.1. Kernel principal component analysis

Principal component analysis (PCA) [21] is a common method applied to dimensionality reduction and feature extraction. PCA method can only extract the linear structure information in the data set, however, it cannot extract this nonlinear structure information. KPCA is an improved PCA, which extracts the principal components by adopting a nonlinear kernel method [22,23]. 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 $x_1, x_2, \ldots, x_n \in R^d$ be the n training samples for KPCA learning [19]. The ith KPCA-transformed feature t_i can be obtained by

$$t_i = \frac{1}{\sqrt{\lambda_i}} \gamma_i{}^T [k(x_1, x_{new}), k(x_2, x_{new}), \ldots, k(x_n, x_{new})]^T, \quad i = 1, 2, \ldots, p \tag{1}$$

Here, Column vectors $\gamma_i (i = 1, 2, \ldots, p;\ 0 < p \le n)$ is the orthonormal eigenvectors to the p largest positive eigenvalues $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_p$, $k(x_i, x_j)$ is the calculation of the inner product of two vectors in the hyper-dimensional feature space F with a kernel function.

By using Eq. (1), the KPCA-transformed feature vector of a new sample vector can be obtained.

2.2. SVM classification model

After feature extraction using KPCA, the training data points can be expressed as $(t_1, y_1), (t_2, y_2), \ldots, (t_p, y_p)$, $t_i \in R^k$ $(k < d)$ is the transformed input vector, y_i is the target value [19]. In the ε-SVM classification [24], 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 denotes as follows:

$$e(f(t) - y) = \begin{cases} 0, & |f(t) - y| \le \varepsilon \\ |f(t) - y| - \varepsilon, & \text{otherwise} \end{cases} \tag{2}$$

Formally the optimization problem by requiring the follows:

$$\begin{aligned} &\text{minimize} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{p}(\xi_i + \xi_i{}^*) \\ &\text{subject to } y_i - (w' \Phi(t_i) + b) \le \varepsilon - \xi_i \\ &(w' \Phi(t_i) + b) - y_i \le \varepsilon - \xi_i{}^* \\ &\xi_i, \xi_i{}^* \ge 0, i = 1, 2, \ldots, p; C > 0 \end{aligned} \tag{3}$$

where ξ_i and $\xi_i{}^*$ are slack variables, the punishment factor C is regularization constant, ε denotes the tube size of SVM. C and ε are both determined by users empirically, 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.

At the optimal solution, the decision function takes the following form:

$$f(t) = sgn\left(\sum_{i=1}^{p}(\alpha_i - \alpha_i{}^*)K(t_i, t_j) + b\right) \tag{4}$$

where α_i and $\alpha_i{}^*$ are the Lagrange multiplier coefficients for the ith training sample, and obtained by solving the dual optimization problem in support vector learning [24]. The training sample for which $\alpha_i \ne \alpha_i{}^*$ is corresponded to the support vectors, $K(k_i, k_j)$ is a kernel function, b is found by the Karush–Kuhn–Tucker conditions at optimality.

2.3. N-RBF kernel function for SVM model

In the SVM, there are some common kernels, shown as follows, and any of those can be chosen to achieve the boundary function. Their detailed usages and descriptions, including parameters definitions, can be found in [25,26].

(1) Gaussian RBF kernel: $K(t_i, t_j) = \exp\left(\frac{-\left\|t_i - t_j\right\|^2}{\sigma^2}\right), \sigma \in R$

(2) Polynomial kernel: $K(t_i, t_j) = (a(t_i \cdot t_j) + b)^d, a \in R, b \in R, d \in N$

(3) Sigmoid kernel: $K(t_i, t_j) = \tanh(a(t_i \cdot t_j) + b), a \in R, b \in R$

(4) Inverse multi-quadric kernel: $K(t_i, t_j) = \frac{1}{\sqrt{\left\|t_i - t_j\right\|^2 + \sigma^2}}, \sigma \in R$

SVM always has good performance in classification when using RBF, which is an effective kernel function for fewer parameters set and an excellent overall performance. A network record contains dozens of attributes, and there may be significant differences between them. Therefore, when the differences between the attributes are very big, using RBF in the training process will produce a larger number of support vectors and the training time will be longer too.

In order to shorten the training time and improve the performance of SVM, an improved kernel function N-RBF is developed by embedding the mean value and the mean square difference values