A novel learning algorithm on probability measure for intrusion detection*

FANG Xiang
School of computer science and technology
Shandong Institute of Business and Technology
Yantai, China

JIAYing
School of computer science and technology
Shandong Institute of Business and Technology
Yantai, China

***Abstract*—Attacks cyber-based have already seriously threaten the security of network environment and network application with the rapid development and wide application of network services. Intrusion detection plays a vital role in the network security. The machine learning methods have been utilized in Intrusion Detection. Because the network intrusion system has to deal with a huge amount of data, its consumption is too large in the space and time. We present an algorithm that learns from probability measures instead of the specific samples in the traditional support vector machine (SVM).The novel algorithm can increase efficiency by the scale down dataset. The simulation test results on the KDD cup99 dataset show that our method is faster than traditional SVM algorithm at the premise of recognition accuracy.**

Keywords—Intrusion detection; Support vector machine; Probability measures; Kernal fuction

I. INTRODUCTION

With the development of electronic technology, especially the level of integrated circuit technology progress, the computer has become an indispensable tool for human life. Services network-based have been extended to various fields including politics, military, economy, science and technology, become an indispensable part for modern society. It is no exaggeration to say, the level of computer network technology has been an important symbol to measure a nation's comprehensive strength. At the same time, the network security becomes a relevant and challenging area of research. Intrusion detection is one of the crucial problems in computer science.

Dorothy E. Denning [1] introduced the concept of detecting cyber-based attacks on computer networks in 1987. Many types of machine learning methods are used to find anomalies or improve the performance of intrusion detection system, such as artificial neural network (ANN), the self-organizing map (SOM), support vector machine (SVM), K nearest neighbor (KNN), the decision tree (DTs), the bayesian (Bayes theorem), extreme learning machine (ELM) and fuzzy logic, and the combination of two or more methods[2]. Especially the introduction of the SVM with better sparsity as well as stronger extensive ability performs better than other in network intrusion detection systems. However, SVM has some limitations, when the training set of the sample is very large it produces a lot of support vectors and the training time is longer, it is very difficult to deal with the high speed of the network.

Learning from the probability distribution instead of the sample to improve the efficiency of machine learning methods is not a new idea. T. Jebara et al [3] proposed the probability product kernel (PPK) as a generalized inner product between two inputs in order to build the positive definite kernel based on probability distribution in 2004. Matthias Hein et al [4] put forward the combination of support vector machine and probability measure, and use to digit recognition and image classification problems, and achieved good effect. Nishant A. Mehta [5] presented the GMMK (generative mean map kernel). By learning from samples, GMMK got probability estimation $\widehat{p_x}$ and $\widehat{p_y}$ of x and y, and were used as a surrogate to construct the kernel between those examples, and proved GMMK, PPK ($\rho = 1$), and linear kernels are equivalent. Using the expected kernels can enhance the robustness of the SVM when the input data is uncertain or incomplete. H.S. Anderson et al [6] improved the SVM by expected kernels for missing features, and proved the algorithm comparable to the standard SVM with expected kernels. Eskin et al [7] put forward that learning from the probability distribution of the sample and applied to anomaly detection. Rocha et al [8] calculated the probability density by the distance between the sample and K neighbor and achieved good results with intrusion detection.

We put forward a novel intrusion detection approach combining SVM and probability distribution, inspired by classic support vector. In the proposed methods, we attempt to make a connection to the kernels on corresponding input spaces, that is, our novel SVM uses probability distribution as training data to improve the efficiency of classification by the two level kernels. The rest of this paper is organized as follows. In Section 2, the classic SVM is presented. In Section 3, based on the concept and theory of regularization over the input space on which the distributions are defined, we proposed positive definite kernels on distributions and pointed the relations between SVM sample-based and SVM distribution-based. Then, the classic SVM, and our algorithm are implemented in KDD cup99 dataset, and experimental results are discussed in Section 4. Section 5 presents conclusion and future work.

Shandong province Programs for Science and Technology Development (2014GGX101044).

II. CLASSIC SVM

Support vector machine which is a machine learning method of analyzing data and recognizing patterns, is used for classification and regression analysis. The basic SVM takes a set of input data, maps to a high dimensional space, and finds an optimal hyper plane that maximize margin between two classes in the high dimensional feature space and predicts, so as to categorize the original data. In the classical SVM, for each given input, one of two possible classes is presented in the output.

For a pattern classification problem, given an IID training data set $\{\mathrm{x}_i, y_i\}_{i=1}^{N}$, where $\mathrm{x}_i \in R^n, y_i \in \{-1,+1\}$, N is the number of samples, n is the dimensions of the sample, the map from input space R^n into high-dimensional feature space H. The decision only relies on vector inner product operation in H, i.e. $\langle \varphi(X_i)\cdot\varphi(X_j)\rangle$ .If a kernel function $K(X_i, X_j)$ can make $K(X_i, X_j) = \langle \varphi(X_i)\cdot\varphi(X_j)\rangle$, then the corresponding classification function is: $f = \sum_{i=1}^{N} a_i y_i K(x_i, \cdot x)$.

In the process of training and decision of support vector machine, use the kernel function instead of inner product operation of the sample vector, there is no need to know the specific expression of ϕ.

Given a non-empty compact set X , H is the complete inner product spaces of function family *F* (i.e., Hilbert space), and $f : x \to R$ for any $f \in F$. If for all $x \in X$, there exists a continuous linear point mapping function $f \to f(x)$, H is called a reproducing kernel Hilbert space (RKHS). In RKHS, $f(x)$ can be expressed as an inner product: $f(x) = \langle f, \phi(x)\rangle_{\mathrm{H}}$, where $\phi: X \to \mathrm{H}$ is the map from X to the feature space H, and the inner product of two mapping characteristics is called kernel.

III. FRAMEWORK OF OPTIMIZED SVM

A. problem description and Definition

Suppose that we have l training data $T = \{(p_1, y_1), (p_2, y_2), ..., (p_l, y_l)\} \in (\mathrm{P} \times y)^n$, where $p_i \in \mathrm{P}, y_i \in Y = \{+1, -1\}$, $i = 1, 2, ..., n$, p_i is the probability measures on a measurable space (X, A) , where A is a σ-algebra of subsets of *X*, P is the set of p_i ,and y is the label of binary classification[9]. According to the basic idea of classic SVM, we replace X of classic SVM with the distribution P as a mean function in an RKHS, in order to find the best rule $I(x): \mathrm{P} \to \mathrm{Y}$ that show the relationship between the input probability p_i and output classification y_i.

Suppose the function $f : X \to R$ in a feature space H exist a reproducing kernel $k: \mathrm{X} \times \mathrm{X} \to \mathrm{R}$, then the mean map form P to H is:

$$\mu : \mathrm{P} \to \mathrm{H}, p \mapsto \int_x k(x, \cdot) dp(x) \qquad (1)$$

Suppose for all of $x \in \mathrm{X}$, $k(x, \cdot)$ is bounded. If k is characteristic, μ is injective [10]. We can write this map $\mu(p)$ as μ_p , and have the reproducing kernel of the following form [11]:

$$\mathrm{E}_p[f] = \langle \mu_p, f \rangle_{\mathrm{H}}, \forall f \in \mathrm{H} \qquad (2)$$

Replace inner product with kernel function, the kernel function of μ_p is the following form:

$K(p, q) = \langle \mu_p, \mu_q \rangle_{\mathrm{H}}$ where p,q is probability measures (3)

Because of $\sup_x \|k(x, \cdot)\|_{\mathrm{H}} < \infty$ and the reproducing property of H：

$$K(p, q) = \iint \langle k(x, \cdot), k(z, \cdot)\rangle_{\mathrm{H}} dp(x) dq(z)$$
$$= \iint k(x, z) dp(x) dq(z) \qquad (4)$$

From (3) and (4), we can know that the K is positive definite kernel on P, and have the following:

Suppose the train dataset $\{p_i, y_i\}_{i=1}^{n}$, where $p_i \in \mathrm{P}, y_i \in \mathrm{R}$ and a loss function $L: (\mathrm{P} \times \mathrm{R} \times \mathrm{R})^n \to \mathrm{R} \cup \{+\infty\}$, the empirical risk functional is expressed as:

$$R(\mathrm{E}_p[f]) = \frac{1}{n} \sum_{i=1}^{n} L(p_i, y_i, \mathrm{E}_{p_i}[f])$$
$$= L(p_1, y_1, \mathrm{E}_{p_1}[f], ..., p_n, y_n, \mathrm{E}_{p_n}[f])$$

Introducing a strictly monotone increasing functionΩ as the regularization item, the risk function becomes:

$$L(p_1, y_1, \mathrm{E}_{p_1}[f], ..., p_n, y_n, \mathrm{E}_{p_n}[f]) + \Omega(\|f\|_{\mathrm{H}}) \quad (5)$$

In RKHS, the function *f* that make (5) minimum can be represent as $f = \sum_{i=1}^{n} a_i \mu_{p_i}$, where a_i is weight that controls the contribution of the distributions.

The demonstration can be obtained from basic principle of support vector machine, and we won't reiterate them here in consideration of references [12]. Thus, classification problems based on the probability distribution can be expressed as a finite linear combination of μ_p .If we limit P to Dirac-measure on *X* and training set $\{\delta_{x_i}, y_i\}_{i=1}^{n}$, (5) is usually regularization function, and the corresponding results is $f = \sum_{i=1}^{n} a_i k(x_i, \cdot)$. Therefore, the solution is a variant of SVM.

B. The determination of kernel function

The classic SVM feature map $\phi(x)$ generally is non-linear, however μ_p is linear. So in the different stages of the modeling, we employ two kernel functions: the first level embedding kernel function k used to obtain measurement vectors, the second level kernel functions K use to allow for the nonlinear algorithm on distributions.

We can define the nonlinear kernel on P by imitating the definition of nonlinear kernel on *X*, and use the mean embedding μ_p for $p_i \in \mathrm{P}$ as its feature representation. Suppose that (1) is injective, $\langle \cdot,\cdot \rangle_p$ is an inner product on P, we can know $\langle p,q \rangle_{\mathrm{P}} = \langle \mu_p, \mu_q \rangle_{\mathrm{H}}$ by linearity [13]. The nonlinear kernel on P define as $K(p,q) = k(\mu_p, \mu_q) = \langle \psi(\mu_p), \psi(\mu_q) \rangle_{\mathrm{H}_k}$, where k is a positive definite kernel. So, provided that the kernel computation depends only on the inner product, we can use some standard nonlinear kernels on *X* to define nonlinear kernels on P. A. Christmann and I. Steinwart [14]had proved that the RBF kernel function $K(\mathrm{P},Q) = \exp\left(-\frac{\gamma}{2}\|\mu_{\mathrm{P}} - \mu_Q\|_{\mathrm{H}}^2\right)$ is universal kernels for any $p,q \in \mathrm{P}$, where *X* is compact set and μ is injective.

In general, we can use the expected kernel function $K(p,q) = \mathrm{E}_{x\sim p, z\sim q}\left[k(x,z)\right]$, where $\mathrm{E}_{x\sim\mathbb{P}}[\cdot]$ is the mathematical expectation that follow distribution P , to solve the SVM problem. When a given distribution and the embedding kernel function k, it is easy to calculate $K(\mathrm{P},Q)$. Such as, given Gaussian distribution $N(m,\Sigma)$ and the Polynomial degree 2 $k(x,y) = (\langle x,y \rangle + 1)^2$ as embedding kernel, and the second level kernel function $K(\mathrm{P}_i,\mathrm{P}_j) = \langle \mu_{\mathrm{P}_i}, \mu_{\mathrm{P}_j} \rangle_H$ $= (\langle m_i, m_j \rangle + 1)^2 + tr\Sigma_i\Sigma_j + m_i^T\Sigma_j m_i + m_j^T\Sigma_i m_j$; given Gaussian distribution $N(m,\Sigma)$ and RBF kernel $\exp\left(-\frac{\gamma}{2}\|x-y\|^2\right)$, and the second level kernel function $K(\mathrm{P}_i,\mathrm{P}_j) = \langle \mu_{\mathrm{P}_i}, \mu_{\mathrm{P}_j} \rangle =$ $\exp\left(-\frac{1}{2}(m_i - m_j)^T\left(\Sigma_i + \Sigma_j + \gamma^{-1}\mathrm{I}\right)^{-1}(m_i - m_j)\right) / |\gamma\Sigma_i + \gamma\Sigma_j + \mathrm{I}|^{\frac{1}{2}}$. Alternatively, if enough of examples are trained, a simple probability model can be chosen to approximate the true probability distribution.

IV. Experiments and analysis

A. data set

KDD cup99 intrusion detection data set is built by DARPA (Defense Advanced Research Projects Agency) for the Knowledge Discovery and Data Mining competition in 1999. The original KDD dataset consists of nearly 5,000,000 labeled records each having 41 attributes [15]. At present, the KDD cup99 dataset is still the most commonly used in evaluation on intrusion detection algorithm; many researchers use this data to validate their algorithms.

In the experiments, we evaluate the performance of the algorithm that learns from the probability distribution and the classic SVM on 10% KDD cup99 dataset in intrusion detection. The 10% KDD cup 99 data sets contains the original 494, 021 records, and 145, 585 samples after deleted duplicate data. The data record belongs to either the "Normal Class" or the "Attack" that is divided into four types (DOS, U2R, Probing and R2L). The sets are given in Table 1.

TABLE I. Distribution for the subset of the KDD Cup 99 NSL-KDD dataset tested.

NO.	types	Distinct samples	Percent of total
1	normal	87832	60.33%
2	DOS	54572	37.48%
3	Probe	2130	1.46%
4	R2L	999	0.69%
5	U2R	52	0.04%
Total	145585		

Although KDD cup99 data set has been pretreated, it still need to be handled in the following two aspects[16]:

1) Continuous: There are several characteristic values are discrete, and even the symbol model, in the KDD cup 99 data set. However our classifier can only accept numeric data, it is necessary to make the discrete data become continuous before training. In this article, we map the symbols into discrete numerical data, and then the discrete numerical data are transformed to [0, l] continuous data by dividing the value by the number of symbols in a row. For example, the character service has a total of 64 label values, convert the labels to integer m between l and 64, and divide m by 64 as the corresponding values of attributes.

2) Standardization: For the dimensions are different to each attribute in KDD cup 99, to standardize data is also required before training. On account of the nonnegativity of the network data, transform the data to the value [0, l]. For characteristic x_i, the transformation formula is $x_i^{'} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}$, where $x_{\max}$ is maximum value and $x_{\min}$ is the minimum value of x_i.

In order to test the ability to process large data processing ability and the scalability of the algorithm, we retain all the characteristics of the KDD cup99, and we have the 41 dimension continuous and standardized feature vector data.

B. Software and hardware equipment

We carried out experiments using MATLAB R2010b merging into libSVM3.20 on Windows XP SP3 platforms. The results reported here are from a machine with Intel Core i3,3.40GHz CPU,4GB RAM.

C. selection of parameters

In order to evaluate the algorithm comprehensively and accurately, we employ several metrics to characterize the performance. They are the time for train or predicts and the detection accuracy, detection precision and false positive rate.

We compare the performance of the classic SVM and our algorithm. For SVM [17], we employ a Gaussian RBF kernel with d^2-distance between the samples $d^2\left(x_i, x_j\right)=\sum_{k=1}^{m}\frac{\left(x_i^{\,k}-x_j^{\,k}\right)^2}{x_i^{\,k}+x_j^{\,k}}$, where x_i is the ith sample and x_i^k is the kth vector of the ith sample i.e., $K\left(x_i, x_j\right)=\exp\left(-\frac{\gamma}{2}d^2\left(x_i, x_j\right)\right)$, the algorithms are fixed by 5-CV over the parameters $C \in \gamma \in \{2^{-3},2^{-2},\ldots\ldots,2^{7}\}$, $\gamma \in \{10^{-3},10^{-2}, 10^{-1}, 10^{0},10^{1}\}$. For our algorithm, we adopt the empirical embedding kernel with Gaussian RBF base kernel k: $K\left(x_i, x_j\right)=\sum_{i=1}^{M}\sum_{j=1}^{M}x_i^k x_j^k k\left(x_i, x_j\right)$ and the Gaussian RBF kernel as the second level kernel. The algorithms are fixed by 5-CV over the parameters $C \in \{2^{-3},2^{-2},\ldots,2^{7}\}$, $\gamma \in \{10^{-3},10^{-2}, \ldots,10^{3}\}$. We computed the accuracy of every class in table 2 and calculated overall value of each metric on KDD cup99 in Table 3. Moreover, we present several algorithms of other publications.

TABLE II. THE ACCURACY OF SOME ALGORITHMS ON KDD CUP99

Classifier	Normal	DOS	Probe	R2L	U2R
Classic SVM	99.29%	83.6%	85.1%	87.3%	87.4%
Our algorithm	99.59%	94.96%	48.85%	32.98%	4.46%

TABLE III. OVERALL VALUE OF EACH METRIC ON KDD CUP99

Classifier	Time(s)	Accuracy	Precision	false positive rate
1R-RF[18]	222.1	99.98%	98.2%	4%
C4.5[18]	265.5	99.95%	99.5%	8.3%
RBF ELM[19]	30.22	89.34%	89.32%	9.9%
Naïve Bayes[20]	667.8	93.02%	99.6%	4%
Classic SVM	232.4	98.69%	92.3%	8.13%
Our algorithm	69.51	96.63%	98.13%	9.74%

Table 2 and table 3 indicate that the proposed algorithm can keep high generalization of SVM, and has high detection efficiency compared with all kinds of intrusion detection scheme, which is due to embedding distributions into RKHS. Our algorithm outperforms SVM in terms of training speed as the size of the training set increases, and has the optimal compromise in detection rate and detection efficiency.

V. CONCLUSIONS AND FUTURE RESEARCH

Although intrusion detection has been studied extensively and intensively, most algorithms rely on supervised classification. Support vector machine (SVM) has better generalization performance, so we put forward an improved learning algorithm based on probability distribution to speed up this algorithm. The simulation experiment on KDD Cup 99 data sets confirmed that our algorithm is superior to classical SVM algorithm in computational time cost at the premise of recognition accuracy. The proposed algorithm obtained the best balance in efficiency and effectiveness.

Future research will include multi-kernel combination, feature filtering, application to other intrusion detection data, online processing and combining with other intelligent algorithms.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their thoughtful and constructive comments which were very helpful in improving the paper. The authors thank Shandong province Programs for Science and Technology Development, China (2014GGX101044).

REFERENCES

[1] Dorothy E. Denning. An Intrusion Detection Model[J].IEEEtransaction on software engineering，1987，13(2):222-233

[2] FangXiang,Wang Lina. Survey of Intelligence Algorithm for Network Intrusion Detection[J]. Communications Technology.2015,12 :222-233

[3] Jebara T, Kondor R, Howard A. Probability Product Kernels[J]. Journal of Machine Learning Research, 2004, 5(5):819-844.

[4] Hein M, Bousquet O, Ghahramani Z, et al. Hilbertian metrics and positive definite kernels on probability measures[J]. Proceedings of Aistats, 2005(2005):136-143.

[5] Nishant A. Mehta and Alexander G. Gray. Generative and Latent Mean Map Kernels. CoRR abs/1005.0188,2010.

[6] H.S. Anderson and M.R. Gupta. Expected kernel for missing features in support vector machines. In Statistical Signal Processing Workshop (SSP), pages 285–288, 2011.

[7] Eleazar Eskin, Anomaly detection over noisy data using learned probability distributions[C]. Proceedings of the International Conference on Machine Learning, Morgan Kaufmann, 2000:255–262.

[8] Rocha L.M, Cappabianco F.A.M, Falcão A.X. Data clustering as an optimum-path forest problem with applications in image analysis[J]. International Journal of Imaging Systems & Technology, 1994, 31(19):50-68.

[9] TAO Jian-Wen,WANG Shi-Tong. Multiple Kernel Local Leaning-Based Domain Adaptation.Journal of Software.2012,23(9):2297-2310

[10] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Scholkopf, and Gert R. G. Lanckriet. Hilbert space embeddings and metrics on probability measures[J]. Journal of Machine Learning Research, 2010(99):1517-1561.

[11] Yang YH ,Speed T. Design issues for cDNA microarray experiments[J]. Nat. Rev. Genet. 2002(8):579–588.

[12] Shao Xigao. The Research and Application of Multiple Kernel Prediction Model Based on Statistical Learning Theory [D]. Central south university.2013

[13] A. Berlinet,Thomas C. Agnan. Reproducing kernel Hilbert spaces in probability and statistics[M]. Kluwer Academic Publishers, 2004.

[14] A. Christmann , I. Steinwart. Universal kernels on non-standard input spaces[J]. In Proceedings of Advances in Neural Information Processing Systems (NIPS), 2010:406–414.

[15] Ali Farzan, Naser Razavi, Mohammad Ali Balafar and Farshad Arvin. Intrusion Patterns Recognition in Computer Networks. Proceedings of the World Congress on Engineering and Computer Science 2011 Vol I.

[16] Sun gang. Research on intrusion detection system based on SVM[D]. Beijing University of Posts and Telecommunications, 2007

[17] Luo Min, Yin Xiaoguang,Wang Lina, Li Xiaohong. Instrusion detection research besed on kernel fuction.Application Research of Computers 24(12),2007：162-164

[18] Wang Xiang, Hu xuegang, YANG Jie. Research on improved intrusion detection model with random forest based on feature evaluation of One-R[J]. Journal of heifei university of technology.2015,38(5):627-630

[19] Fossaceca J M, Mazzuchi T A, Sarkani S. MARK-ELM: Application of a novel Multiple Kernel Learning framework for improving the robustness of Network Intrusion Detection[J].Expert Systems with Applications , 2015, 42: 4062–4080

[20] Farid D M, Harbi N, Rahman M Z. Combining Naïve Bayes and Decision Tree for Adaptive Intrusion Detection[J]. International Journal of Network Security & Its Applications, 2010, 2(2):52-58.