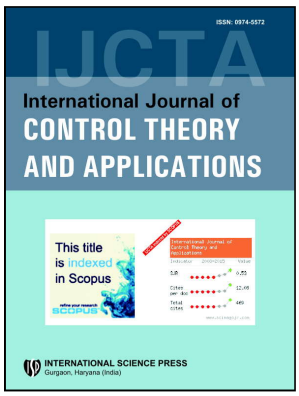

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 19 • 2017

Performance Analysis of Supervised Learning Based Intrusion Detection System

Shalini Chaurasia[1] and C. Rama Krishna[1]

[1] Computer Science & Engineering Department NITTTR, Chandigarh, India,
Emails: shalini.cs27@gmail.com , rkc_97@yahoo.com

Abstract: For network & Computer area Intrusion Detection System (IDS) has more and more turn out to be a central problem. The primary research problem of IDS from the research concerns is Optimizing its efficiency that receives increasingly attention. The chance from spammers, attackers & crook organizations has grown up with the enlargement of net, hence, IDS grew to be a core part of digital network for the reason that of incidence of such threats. We perform three arrangements of examinations. From the major investigation, the frameworks are ready using the entire 41 highlights. The second trial where we perform feature extraction through making use of Kernel Principal Component Analysis (KPCA) as to decide upon the satisfactory factors as opposed to utilizing all the 41 entails and play out the trial with Linear Support Vector Machine (SVM), Stochastic Gradient Descent (SGD) and Adaptive boost (Ada) and believe concerning the results. The third analysis where we perform feature selection by means of making use of correlation as to opt for the fine components as opposed to using all the 41 entails and play out the trial with Linear SVM, SGD and Adaptive Boost and examine the effects.

Keywords: *IDS, Machine learning, KDDcup99 Dataset, Feature Extraction, Feature Selection .*

1. INTRODUCTION

IDS is defined as a software utility that detects system activities for hazardous movements and generates experiences to management. IDS comprise various varieties of tactics with the target to observe site visitors in specific approaches [10]. The groups are using IDS with aim to identify problems with security policies and documenting existing threats.

Today, IDS has become the need of nearly every organization. It helps to record information associated with detected actions, and alert security administrator and produce reports. This system constantly monitors network for any abnormal activity.

IDS is mainly of two types i.e. Network Intrusion detection System (NIDS) and Host Intrusion Detection Systems (HIDS). In NIDS, anti-threat software is installed only at specific instance such as servers that provide communication between the outside environment and the network segment that is to be protected. In HIDS, anti-

threat application software such as antivirus software, firewall & spyware detection programme installed on each computer which is connected over a network that has two way access to the outside environment such as the internet. A snapshot of process documents is taken by it and compared it with the earlier taken snapshot.

If we when put next it with firewall, nonetheless they both establish with protection, IDS framework varies from firewall. Firewall constrains access between techniques to prevent interruption and do not flag an assault from throughout the procedure. An IDS, assesses a suspected interruption as soon as it has happened and flags a warning. A framework that ends associations is called an interruption counteractive action framework.

System attack is as a rule characterised as an interruption to your method base a good way to first break down your surroundings and acquire knowledge with a exact finish goal to abuse the present open ports or vulnerabilities - this may increasingly comprise unapproved access to your assets too [20].

Passive attacks are in nature of roof dropping on, or checking of transmission. Inactive assaults contain exercise examination, checking of unprotected correspondences, unscrambling feebly encoded action, and catching confirmation information, for example, passwords.

Active attack includes some alteration of the information stream or formation of the false stream.

2. PROPOSED ALGORITHM

The methodology of designing the proposed scheme is divided into three phases: Normal, Feature Selection and Feature Extraction.

2.1. Normal

In the primary analysis, the frameworks are prepared utilizing all the 41 features in this phase following steps takes place as shown in Figure 1.

Step1: KDD-99 dataset with 41 features.

Step2: Input the feature & labels into Linear SVM, SGD & Adaptive Boost & make three models.

Step3: Perform the test on these models and calculate the precision, recall & accuracy.

2.2. Feature Extraction

The third analysis where we perform feature selection by means of making use of correlation as to opt for the fine components as opposed to using all the 41 entails and play out the trial with straight SVM, SGD and Adaptive Boost and examine the effects. We use correlation method because it represents the distribution based similarity of features with reducing the imbalancing of features in form of distribution. In this feature extraction phase, the following steps take place as shown in the Figure 1.

Step1: KDD-99 dataset with 41 features.

Step2: Feature extraction with KPCA.

Step3: Input the feature & labels into Linear SVM, SGD & Adaptive Boost & make three models.

Step4: Perform the test on these models and calculate the precision, recall & accuracy.

2.3. Feature Selection

Feature selection is performed in the third analysis by utilizing correlation to choose some best components as opposed to utilizing all the 41 includes and play out the trial with Linear Support Vector Machine (SVM), Stochastic Gradient Descent (SGD) and Adaptive Boost and analyze the outcomes. We use correlation method

because it represents the distribution based similarity of features with reducing the imbalancing of features in form of distribution. In the feature selection phase the following steps takes place as shown in Figure 1.

Step1: KDD-99 dataset with 41 features.

Step2: Feature Selection by correlation method.

Step3: Input the feature & labels into Linear SVM, SGD & Adaptive Boost & make three models.

Step4: Perform the test on these models & calculate the precision, recall & accuracy.

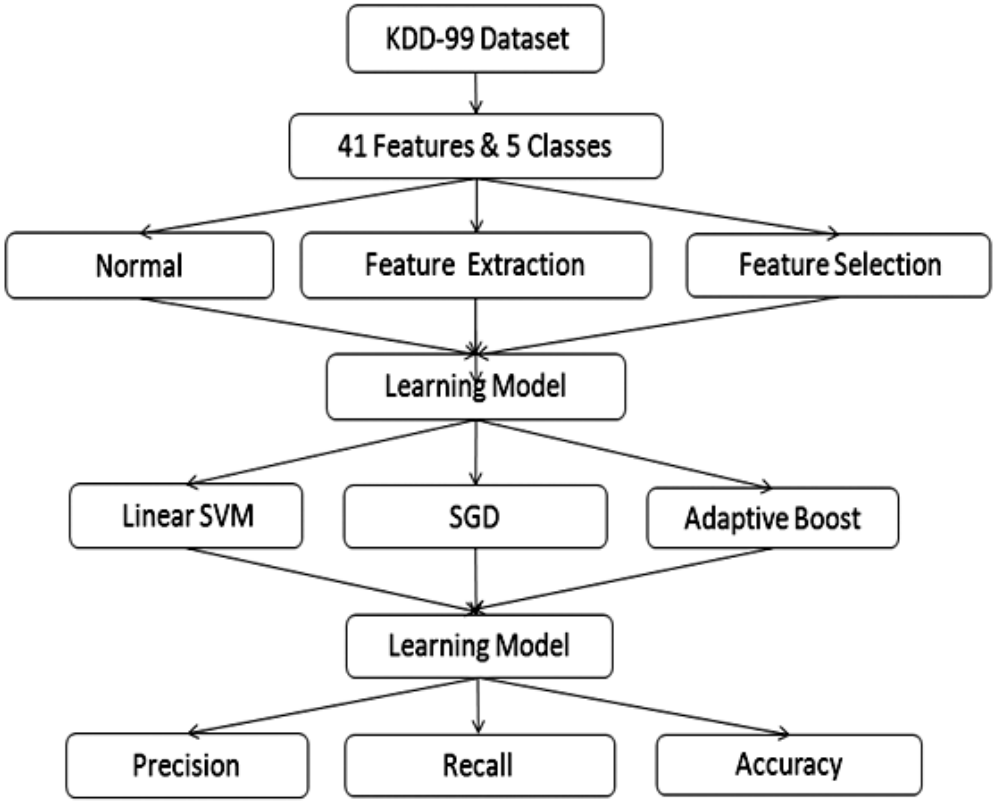

Figure 1: General Methodology

3. EXPERIMENT AND RESULT

3.1. Description of Dataset

Data set KDD 99 are used to carried out the experiment. It was once created headquartered on the Defence developed research undertaking company DARPA based on intrusion detection analysis software [27]. They simulated computer network operated as associate usual setting that used to be contaminated by using quite a lot of varieties of attacks. The uncooked facts set turned into processed into connection files. For every connection, forty one more than a few features had been extracted. Each and every connection was labelled as traditional or below exact kind of assault. There are 39 attacker forms that could be labelled into 4 important categories which summarized in Table 1. There are four important classes of assaults described below:-

3.1.1. Denial of Service (Dos)

DoS attack results by means of stopping legit requests to a community useful resource through consuming the bandwidth or by way of overloading computational assets. An attacker tries to restrict reliable customers from utilising a provider e.g. TCP SYN Flood, Smurf.

3.1.2. Probe

An attacker tries to search out information in regards to the target host. These types of attack gather understanding of goal procedure previous to initiating an attack. For example, scanning victims so as to get competencies about on hand offerings, making use of operating process.

Table 1
Attacks Type in KDD Dataset

DoS (391458)	*U2R (52)*	*Probe (4107)*	*R2L (1126)*
Back (2203)	Buffer-overflow (30)	Ipsweep (1247)	ftp_write (8)
Land (21)	Loadmodule (9)	Nmap (231)	Guess_passwd (53)
Neptune (107201)	Perl (3)	Portsweep (1040)	Imap (12)
Pod (264)	Rootkit (10)	Satan (1589)	Multihop (7)
Smurf (280790)			Spy (2)
Teardrop (979)			Phf (4)
			Warezclient (1020)
			Warezmaster (20)

3.1.3. User to Root (U2R)

On this case, an attacker starts out with access to a normal person account on the approach and is able to take advantage of the procedure vulnerabilities to obtain root entry to the process. An attacker has nearby account on sufferer's host and tries to attain the root privileges.

3.1.4. Remote to Local (R2L)

On this, an attacker who doesn't have an account on a far off computing device sends packet to that machine over a network and exploits some vulnerabilities to achieve neighbourhood entry as a user of that computing device. An attacker does no longer have regional account on the victim host and check out to receive it.

3.2. Performance Metrics

The important performance parameters chosen to analyze the results are:

1. Precision
2. Recall
3. Accuracy

3.2.1. Precision

Precision is the fact of being accurate and correct. Precision gives the idea of correctly predicted instances. It is measured as proportion of true positive from all positives and is calculated as:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

3.2.2. Recall

Recall measure how much relevant data is retrieved from any machine learning algorithm. It focuses on the valuable information.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

3.2.3. Accuracy

Accuracy is one of the primary measures for describing the performance of any algorithm. It represents the degree to which an algorithm can correctly predict the positive and negative instances and is calculated by following formula:

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{3}$$

Where True Positive (TP), True Negative (TN), False Positive (FP) & False Negative (FN) are numbers.

3.3. Experimental Setup

The experiment used to be performed by making use of the KDD99 information set [21]. Windows machine is used to carried out the experiment having configuration Intel® Core™ 2 Duo CPU 2310M @ 2.66 GHz having RAM 8GB and the operating system is MS windows 7. Now, we have used MATLAB R2013a that is an open source framework MATLAB R2013a [12].

3.4. Results

In this section, we reward a performance evaluation for some supervised learning methods. Here we used good identified KDD Cup99 data [21] to make important investigations for network anomaly. In this we perform three arrangements of trials. In the primary analysis, the frameworks are prepared utilizing all fourty one features. Feature extraction is performed in second investigation by utilizing Kernel Principal Component Analysis (KPCA) as to choose the best components as opposed to utilizing all the 41 includes and play out the trial with Linear SVM [22], SGD and Adaptive Boost and analyze the outcomes. Feature selection is performed in the third analysis by utilizing correlation to choose some best components as opposed to utilizing all the 41 includes and play out the trial with Linear Support Vector Machine (SVM), Stochastic Gradient Descent (SGD) and Adaptive Boost and analyze the outcomes.

3.4.1. Performance Measure with all 41 Features Results

We first compared the performance of three classification schemes, namely Linear SVM, SGD and adaptive Boost. Table 2 illustrates the performance of Linear SVM, SGD and Adaptive Boost algorithms for KDD99 data set for all 41 features without applying feature extraction & feature selection. The result showed by SGD has significant by 41 features. Also Linear SVM showed better results in comparison Adaptive Boost.

3.4.2. Performance Measure for Feature Extraction using KPCA

Outcome of feature extraction utilizing KPCA confirmed that Linear SVM & SGD are most efficient for detecting assaults than Adaptive Boost as proven in Table 3.

Table 2
Performance evaluation with all 41 Features for KDD99 data set [21]

Classifier	*Accuracy*	*Precision*	*Recall*
Linear SVM	90.91	84	86.18
SGD	96	96	96.18
Adaptive Boost	84	84	86.18

Table 3
Performance evaluation with applying Feature Extraction using KPCA for KDD99 data set [21]

Classifier	*Accuracy*	*Precision*	*Recall*
Linear SVM	90.91	76	66.18
SGD	85.9998	71	68.18
Adaptive Boost	81	76	91

3.4.3. Performance Measure for Feature Selection using Correlation

Outcome of feature selection utilizing correlation showed that Linear SVM & SGD has accuracy rate higher than adaptive boost as proven in Table 4.

Table 4
Efficiency analysis with making use of Feature Selection using Correlation for KDD99 data set [21]

Classifier	*Accuracy*	*Precision*	*Recall*
Linear SVM	90.91	73	62.182
SGD	86	84	76.182
Adaptive Boost	81	73	62.182

4. CONCLUSION & FUTURE WORK

This paper emphasis on make a optimize classifier model for two classes attack and not attack but problem is data imbalance, which is improved by SGD like classifier. We perform three arrangements of examinations. From the major investigation, the frameworks are ready using the entire 41 highlights. The second trial where we perform feature extraction through making use of KPCA as to decide upon the satisfactory factors as opposed to utilizing all the 41 entails and play out the trial with Linear SVM, SGD and Adaptive boost and believe concerning the results. The third analysis where we perform feature selection by means of making use of correlation as to opt for the fine components as opposed to using all the 41 entails and play out the trial with linear SVM, SGD and Adaptive Boost and examine the effects. These results conclude average performance of SGD better than other classifiers.

There is still scope of improvements to propose systems which are able to detect all types of attacks & can reduce the feature set by feature selection and feature extraction with the help of different classifier.

REFERENCE

[1] P. Jain, S. Raghuwanshi, and P. Rk, "New Mobile Agent-Based Intrusion Detection Systems for," International Journal of Wireless Communication 1, vol. 1, no. 1, pp. 1–5, 2011.

[2] G. Giacinto, F. Roli, and L. Didaci, "Fusion of multiple classifiers for intrusion detection in computer networks," Pattern Recognit. Lett., vol. 24, no. 12, pp. 1795–1803, 2003.

[3] W. Yu, C. Xiaohui, and W. Sheng, "Anomaly Network Detection Model Based on Mobile Agent," Third Int. Conf. Meas. Technol. Mechatronics Autom., pp. 504–507, 2011.

[4] S. T. Sarasamma, Q. A. Zhu, and J. Huff , "Hierarchical Kohonenen Net for Anomaly Detection in Network Security," Fourth Int. Conf. on Machine Learning and Cybernetics., vol. 35, no. 2, pp. 704–567, 2011.

[5] Y. Li and L. Guo, "An active learning based TCM-KNN algorithm for supervised network intrusion detection," Comput. Secur., vol. 26, no. 7–8, pp. 459–467, 2007.

[6] M. M. T. Jawhar and M. Mehrotra, "Anomaly Intrusion Detection System using Hamming Network Approach," International Journal of Computer Science & Communication 1, vol. 1, no. 1, pp. 165–169, 2010.

[7] N. Srivastav and C. Rama Krishna, "Novel intrusion detection system integrating layered framework with neural network," Proc. 2013 3rd IEEE Int. Adv. Comput. Conf. IACC 2013, no. FEBRUARY 2013, pp. 682–689, 2013.

[8] M. a. S. Heba Ezzat Ibrahim, Sherif M. Badr, "Adaptive Layered Approach using Machine Learning Techniques with Gain Ratio for Intrusion Detection Systems," Int. J. Comput. Appl. (0975 – 8887), vol. 56, no. 7, pp. 10–16, 2012.

[9] S. W. Lin, K. C. Ying, C. Y. Lee, and Z. J. Lee, "An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection," Appl. Soft Comput. J., vol. 12, no. 10, pp. 3285–3290, 2012.

[10] S. Mukherjee and N. Sharma, "Intrusion Detection using Naive Bayes Classifier with Feature Reduction," Procedia Technol., vol. 4, pp. 119–128, 2012.

[11] M. Panda, A. Abraham, and M. R. Patra, "A hybrid intelligent approach for network intrusion detection," Procedia Eng., vol. 30, no. 2011, pp. 1–9, 2012.

[12] A. P. Muniyandi, R. Rajeswari, and R. Rajaram, Network anomaly detection by cascading k-Means clustering and C4.5 decision tree algorithm," Procedia Eng., vol. 30, no. 2011, pp. 174–182, 2012.

[13] M. Stevanovic and J. M. Pedersen, "An efficient flow-based botnet detection using supervised machine learning," 2014 Int. Conf. Comput. Netw. Commun., pp. 797–801, 2014.

[14] A. Karim, R. Salleh, M. Shiraz, S. Shah, I. Awan, and N. Anuar, "Botnet detection techniques: review, future trends, and issues," Comput. Electron., vol. 15, no. 11, pp. 943–983, 2014.

[15] A. Feizollah, N. B. Anuar, R. Salleh, and A. W. A. Wahab, "A review on feature selection in mobile malware detection," Digit. Investig., vol. 13, pp. 22–37, 2015.

[16] A. Karim, S. Adeel, A. Shah, R. Bin Salleh, M. Arif, and R. Noor, "Mobile Botnet Attacks – an Emerging Threat/ : Classification , Review and Open Issues," TIIS 9, vol. 9, no. 4, pp. 1471–1492, 2015.

[17] S. Garasia, D. Rana, and R. Mehta, "Http Botnet Detection Using Frequent Patternset Mining," Intl. Journal of Engineering Science and Advanced Technology (IJESAT), no. 3, pp. 619–624, 2012.

[18] C. Livadas, R. Walsh, D. Lapsley, and W. T. Strayer, "Using Machine Learning Techniques to Identify Botnet Traffic," Local Comput. Networks, Proc. 2006 31st IEEE Conf., pp. 967–974, 2006.

[19] I. Mohammad, R. Pandey and A. Khatoon, "A Review of types of Security Attacks and Malicious Software in Network Security," International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), vol. 4, no. 5, pp. 413–415, 2014.

[20] KDD Cup 1999 Intrusion Detection Data. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html[Accessed on: Feburary 2016]

[21] M. Xu, N. Ye, "Probabilistic networks with undirected links for anomaly detection", In Proceedings of IEEESystems, Man, and Cybernetics Information Assurance and Security Workshop, pp. 175–179, 2000.

[22] E. B. Beigi, H. H. Jazi, N. Stakhanova, and A. A. Ghorbani, "Towards effective feature selection in machinelearning-based botnet detection approaches," *in IEEE Conference on Communications and Network Security (CNS)*, pp. 247–255, IEEE, 2014.

[23] K. Singh, S. C. Guntuku, A. Thakur, and C. Hota, "Big data analytics framework for peer-to-peer botnet detection using random forests," *Information Sciences,* vol. 278, pp. 488-497, 2014