International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8, Issue-1, May 2019

Strategies for Network Intrusion Detection using Machine Learning Algorithms

N Radhika Amareshwari, S Ramanjaneyulu, G Swapna

Abstract: ***Spying identification is an emerging field of software development and research and networking with increasing Internet use in daily life. There have been many issues with classical IDS systems, along with other low network attack identification, high false alarm rate, and inadequate analytical capacity. The primary potential of research on the subject then is to establish a model for intrusion detection with increased performance and decreased preparation time. Machine learning is an efficient tool for analyzing any abnormal events taking place in the network activity flow. This paper proposes a mixture of two methodologies to determine any abnormal conduct in internet traffic. This paper recommends to use the principal component analysis (PCA) and Rough-set Support Vector Machine (SVM) as the hybrid intrusion detection models.***

Index Terms: Intrusion Detection; Machine Learning; Support Vector Machine

I. INTRODUCTION

As computer power, storage capacities and data collection increase, machine learning and artificial intelligence are more widely linked across industry and applications than ever before in recent memory. There are so many types of hazards, such as malware and DDOS attacks, on the Internet. An intrusion detection system can protect a network against such attacks. An IDS system can detect intrusions and degenerate an alert when an intrusion is detected. This intrusion detection system analyzes all traffic in a network. This is a difficult task for large datacenters. There is a huge amount of data across the data center network. Therefore, standard intrusion systems cannot complete all traffic. An intrusion detection system can alert malicious behavior administrators. Most intrusion detection systems need a lot of manual maintenance to deliver good performance. This thesis attempts to determine whether an intrusion detection system is capable of performing in an acceptable way. This is done by using algorithms for machine learning.

Revised Manuscript Received on 30 May 2019.
***** Correspondence Author
N. Radhika Amareshwari*, Working as Assistant Professor, Department of CSE,Geethanjali College of Engineering and Technology, Hyderabad, Telangana, India.
S.Ramanjaneyulu, Working as Assistant Professor, Department of CSE,Geethanjali College of Engineering and Technology, Hyderabad, Telangana, India.
G.Swapna, Working as Assistant Professor, Department of CSE,Geethanjali College of Engineering and Technology, Hyderabad, Telangana, India.

These are algorithms that can learn and find input patterns. Automatic intrusion detection problem promises machine learning algorithms. The intent of an access control structure should not be to stop the attack, however merely to identify, detect or press release attacks, scheme and network security issues [1]. Intrusion detection systems are usually supplemented by firewalls [2]. The other feature of the IDS is to detect anomalies outside network traffic and report them to the administrator or to prevent suspected contacts [3]. IDS is able to detect both internally and externally attacks. Several methods for network intrusion detection are proposed. The intrusion detection of anomalies in the network is a major element in network security [4],[5].It seems very peculiar in case of behavior of data usage has discussed by the authors in [6], thereby creating a classification issue for detection systems more tuff job**,** namely the effectiveness and efficiency of the distinction between normal and abnormal activities.Machine learning has now been expanded to implement an effective system of intrusion detection. Machine learning methods in current intrusion detection are extremely functional and improved.In certain cases the authors in [7] presented various SVM methods and NN in [8] for making the detection of attacker in the traffic of the network for upgrading the classification rate better in the anomaly detection systems. This study contends that new methods, like machine learning, offer a different way to close the cyber gap by reducing the number of cyber security staff necessary to investigate, evaluate and share information on malware detection. This paper proposes a new algorithm using a combination of the two machine-learning methods, Main component analysis & Rough set SVM technique, which allows the detection of anomalous network behaviors and the classification of normal and abnormal behaviors.
The main contributions and organization of this paper are summarized as follows: In section 2 we describe background details of machine learning schemes. The section 3 proposed work.The section 4 results and discussion. Finally in section 5 we concluded the paper.

II. BACKGROUND WORKS

Principal component analysis:
PCA is one of the widely employed data mining analytical methods to decrease degrees of freedom and to recognize data points with a maximum variance. In this technique, the emphasis is on the identification of the volume-based abnormalities in source-destination flow aggregated in backbone networks. The PCA method defines anomalous volume of traffic on a particular connection by comparing it with previous values. PCA thus separates the measurement of traffic into sub-regions that represent normal and abnormal traffic.

The PCA results in the projection of a feature space on a smaller subspace, representing data by reducing feature space dimensions.
This reduces calculation costs and the parameter estimation error.
The standard PCA approach can be summarized in six simple steps:

(i) Determine the covariance matrix of the normalized d-dimensional dataset.

(ii) Determine the eigenvectors and eigenvalues of the covariance matrix.

(iii) Sort the eigenvalues in descending order.

(iv) Select the k eigenvectors that correspond to the k largest eigenvalues where k is the number of dimensions of the new feature subspace.

(v) Construct the projection matrix from the k selected eigenvectors.

(vi) Transform the original dataset to build a new k-dimensional feature space.

In PCA, we expressed the data before computing the covariance matrix for kernel PCA, we need to do the same.

$$\tilde{\phi}\left(X_n\right) = \phi\left(X_n\right) - \frac{1}{N}\sum_{l=1}^{N}\phi\left(X_l\right) \qquad (1)$$

$$\tilde{K}_{nm} = \tilde{\varphi}\left(X_n\right)^T \tilde{\varphi}\left(X_m\right) \qquad (2)$$

$$= \varphi\left(X_n\right)^T \varphi\left(X_m\right) - \frac{1}{N}\sum_{l=1}^{N}\varphi\left(X_n\right)^T \varphi\left(X_l\right) - \frac{1}{N}\sum_{l=1}^{N}\varphi\left(X_l\right)^T \varphi\left(X_m\right) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{l=1}^{N}\varphi\left(X_j\right)^T \varphi\left(X_l\right)$$

$$k\left(X_n, X_m\right) - \frac{1}{N}\sum_{l=1}^{N}k\left(x_n, x_l\right) - \frac{1}{N}\sum_{l=1}^{N}k\left(x_l, x_m\right) + \frac{1}{N^2}\sum_{j=1}^{N}\sum_{l=1}^{N}k\left(x_j, x_l\right)$$

In matrix notation, the expression is given as

$$\tilde{K} = K - 1_N K - K 1_N + 1_N K 1_N \qquad (3)$$

Eigen-decomposition is then done for the centered kernel matrix *K*. Suppose $\{a_1. \ldots a_K\}$ are the top *K* eigenvectors of kernel matrix *K*. The *K*-dimensional KPCA projection $z = [z_1. \ldots z_K]$ of a point *x*: $Z_i = \varphi\left(X\right)^T V_i \qquad (4)$

Recall the definition of v_i

$$V_i = \sum_{n=1}^{N} a_{in}\varphi\left(X_n\right) \qquad (5)$$

Thus

$$Z_i = \varphi\left(X\right)^T V_i = \sum_{n=1}^{N} a_{in}k\left(X, X_n\right) \qquad (6)$$

Support Vector Machine Classification Model:

The SVM classification device is designed to solve a binary classification issue by identifying the class frontier of the hyper plane to maximize the margin in the given training data. SVM has been used to address Kernel functions on linear boundary problems. The optimal hyper plane is linear and nonlinear to patterns. The kernel function separates the transformation of original data into new space. Recently, several improved SVMs have grown quickly, including some of the most prevalent and most efficient kernel SVMs. The advantages of kernel SVM are:

(1) Working well in practice and were remarkably successful in such diverse fields as the categorization of natural languages, bioinformatics and computer vision;

(2) Having few tunable parameters; and

(3) Training often involves convex quadrature optimization.

A kernel K effectively computes dot products in a higher-dimensional space R^Mwhile remaining in R^N. In symbols:

For the following, let $\vec{x}_i, \vec{x}_j \in R^N$ be rows from the dataset.

1. Polynomial Kernel: $\left(\gamma.\left(\vec{x}_i, \vec{x}_j\right) + r\right)$

2. Radial Basis Function (RBF) Kernel: $\exp\left(-\gamma.\left|\vec{x}_i - \vec{x}_j\right|^2\right)$

3. Sigmoid Kernel: $\tanh\left(\left\langle \vec{x}_i, , \vec{x}_j \right\rangle + r\right)$

III. PROPOSED MODEL

A new hybrid model for intrusion verification is proposed in this section. The proposed model incorporates PCA with SVM to classify network traffic anomalies. In order to achieve higher levels of precision the SVM parameters such as the punishment factor (C) and kernel parameter (Gamma) are optimized. Figure.1 shows the block diagram of the system proposed. During the first level PCA considers an optimal subset of all attributes through the removal of noisy information from the attributes that encompass it. The variance threshold is generally set to a higher value, so that the cumulative variance of the various principal components is higher than the threshold and the feature vectors can be selected. The second phase utilizes the optimal subset obtained from PCA for the classification of the training data and the test data set for SVM.

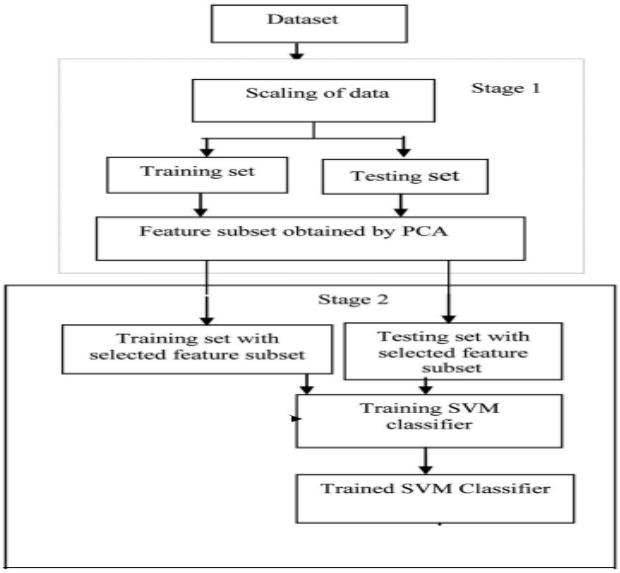

Fig.1: Proposed intrusion detection model

The framework used to acquire an optimal subset of PCA and to optimize the optimal SVM kernel parameters for categorization is discussed in the following section.

Pre-processing of Dataset:

In this work, we used the KDDcup'99 dataset for real world testing. The data from the 1998 DARPA Intrusion Detection Evaluation [7] is the origin of the KDD Cup 1999. Several system calls can be made up of a process. Some ineffective data are filtered and modified in this phase.

Some document items, for instance, must be converted into numbers. Each database mechanism has 41 attributes in Table 1.

Principal Component Analysis:
Figure.2 shows the step-by-step evaluation of the generation of PCA feature vectors. After the pre-processing a normalized function matrix is acquired and fed as an input for the mean and covariance of individual components. Eigenvectors are produced for all features and the highest own values are retained, and the respective vectors are maintained (in a vector set) in order to acquire the optimally functioning subset. Very last, data pre-processing and data discretion is used to arrange data. The RST will then be used to find relevant features. In conclusion, the system includes the SVM to identify the data as follows:

Table 1: KDD cup'99 features

No	Features	No	Features
1	duration	22	is_guest_login
2	protocol_type	23	count
3	service	24	srv_count
4	flag	25	serror_rate
5	src_bytes	26	srv_serror_rate
6	dst_bytes	27	rerror_rate
7	land	28	srv_rerror_rate
8	wrong_fragment	29	same_srv_rate
9	urgent	30	diff_srv_rate
10	hot	31	srv_diff_host_rate
11	num_failed_logins	32	dst_host_count
12	logged_in	33	dst_host_srv_count
13	num_compromised	34	dst_host_same_srv_rat
14	root_shell	35	dst_host_diff_srv_rate
15	su_attempted	36	dst_host_same_src_port_rate
16	num_root	37	dst_host_srv_diff_host_rate
17	num_file_creations	38	dst_host_serror_rate
18	num_shells	39	dst_host_srv_serror_ra
19	num_access_files	40	dst_host_rerror_rate
20	num_outbound_cmd	41	dst_host_srv_rerror_rat
21	is_host_login		

The flowchart is our three-step intrusion detection technique that can be seen in figure 3.

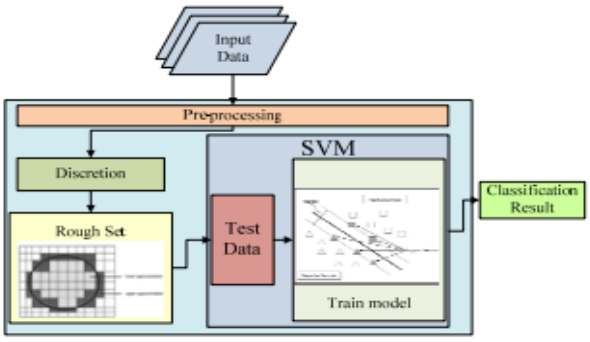

Fig.3: Flow process of working prototype system
Initially, data pre-processing and data discretion are used to arrange data. The RST can then be used to find interesting characteristics. Eventually, the device uses the SVM to classify the data mentioned as continues to follow: This dataset is being used for the Third International Competition for Data Mining Tools and Knowledge Discovery. This competition was aimed at creating a network detector to explore "bad" links and "good" links. The assault has "bad" link categories as seen in Table 2.

Table 2: classification of variou attack types

Attack types	class	Attack types	class
Normal	Normal	guess_pswd	R2L
apacha2	DoS	Imap	
back	multihop		
land	named		
mailbomb	phf		
netune	sendmail		
pod	Snmpget attack		
processtable	snmpguess		
smurf	spy		
teardrop	warezclient		
udpstorm	warezmaster		
buffer_overflow	U2R	worm	
httprunnel	xlock		
loadmodule	xsnoop		
perl	Ipsweep	Probe	
ps	mscan		
rootkit	nmap		
sqlattack	portsweep		
xterm	saint		
ftp_write	R2L	satan	

IV. RESULTS AND DISCUSSION

As KDDCUP99 dataset contains total 41 features, we need to train our data using all of them. The experimental results shows that by reducing the feature set before performing the experiment, we can observe improvement in the detection rate of the system. Also more time is required to process more features. So we can extract the useful feature which helps in improving the performance of our system.

Response time: Reason for more response time is mentioned as following. First reason is the use of SVM classifier in sequential manner [1].

Solution to the first problem is implementing SVM classifier in parallel. But parallel implementation requires more hardware resources. As a solution to second problem, we can use PCA feature reduction technique to minimize feature set [3] which decreases the computational time and hence can improve response time.

Detection rate: Reason for low detection rate is selection of improper method for preprocessing the data [4]. As a solution to the problem stated above, results in the literatures show that improvement in detection rate is observed when PCA feature subset [3] is used instead of using fuzzy membership function[4] for preprocessing the dataset.

False alarm: Reason for false alarm generation in the system is either due to over training or lack of training provided the data points during preprocessing step.The accuracy of our proposed methodology might be the healthiest, but its false position and assault rate is lower than that of Entropy to SVM. Table 3 shows the results.

Table 3. Comparison of 3techniques using SVM approach

Algorithm type	Attack Detection Rate	False Positive Rate	Accuracy
41 features to SVM	70.03%	29.97%	86.79%
Entropy to SVM	92.44%	7.56%	73.83%
Rough Set of SVM	86.72%	13.27%	89.13%

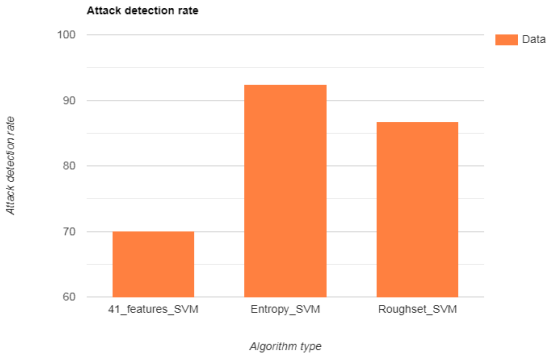

Fig.4:comparison of attack detection rate for various algorithms of SVM

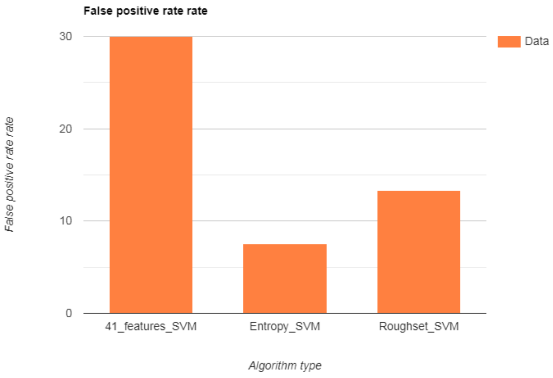

Fig.5:comparison of false positive rate for various algorithms of SVM

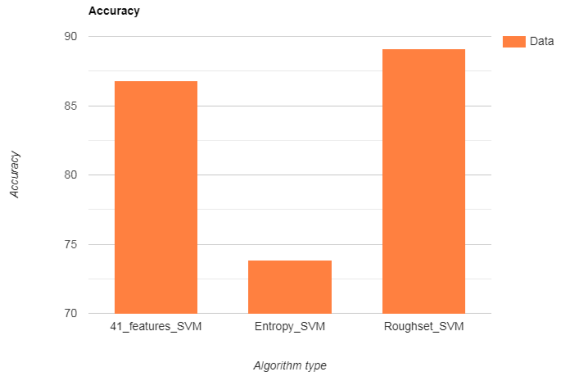

Fig.6:comparison of accuracy for various algorithms of SVM

V. CONCLUSION

In this paper, two machine learning techniques have been used for intrusion detection in the network. This paper recommends a model for intrusion detection, which integrates the primary component analysis (PCA) and RBF kernel support vector machines (SVMs). Reducing dimensionality by PCA eliminates noisy attributes and maintains the optimal subset of attributes. SVMs build classification strategies based on PCA training data. A mixture of a quite low range of features can lead to further research work that is having the capable of overall decrement in the amount of time for best detection of attacker in the traffic of concerned network type.

REFERENCES

1. S. A. Hofmeyr, S. Forrest, A. Somayaji, J. Computer Security - JCS, 6(3), p151 (1998).
2. E. Lundin, E. Jonsson, J. Computer Networks, 3(4), p623 (2000).
3. X. D. Hoang, J. Hu, P. Bertok, "A multi-layer model for anomaly intrusion detection using program sequences of system calls", Proceeding of the IEEE International Conference on Networks (ICON), 531-536(2003) .3.
4. G. Xiaoqing, G. Hebin, and C. Luyi, "Network intrusion detection method based on Agent and SVM," 2010 2nd IEEE Int. Conf. Inf. Manag. Eng., pp. 399–402, 2010.
5. R. P. Lippmann and R. K. Cunningham, "Improving intrusion detection performance using keyword selection and neural networks," Comput. Networks, vol. 34, pp. 597–603, 2000.
6. E. Denning, R. Ave, and M. Park, "Attempted break-in," pp. 118–131.
7. W. Hu, Y. Liao, and V. R. Vemuri, "Robust anomaly detection using support vector machines," Proc. Int. Conf. Mach. Learn., pp. 282–289, 2003.
8. Z. Zhang, J. Li, C. N. Manikopoulos, J. Jorgenson, and J. Ucles, "HIDE: a Hierarchical Network Intrusion Detection System Using Statistical Pre-processing and Neural Network Classification," Proc. IEEE Work. Inf.