International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206431

Multilevel Intrusion Detection System with Affinity Clustering and Ensemble SVM

Sadhana Patidar[*1], Priyanka Parihar[2], Chetan Agrawal[3]
[*1] M.Tech Scholar, Department of CSE RITS, Bhopal, Madhya Pradesh, India
[2,3]Assistant Professor, Department of CSE RITS, Bhopal, Madhya Pradesh, India

Article Info
Volume 6, Issue 4
Page Number: 1-10
Publication Issue :
July-August-2020

Article History
Accepted : 01 Aug 2020
Published : 27 Aug 2020

ABSTRACT

Now-a-days with growing applications over internet increases the security issues over network. Many security applications are designed to cope with such security concerns but still it required more attention to improve speed as well accuracy. With advancement of technologies there is also evolution of new threats or attacks in network. So, it is required to design such detection system that can handle new threats in network. One of the network security tools is intrusion detection system which is used to detect malicious data packets. Machine learning tool is also used to improve efficiency of network-based intrusion detection system. In this paper, an intrusion detection system is proposed with an application of machine learning tools. The proposed model integrates feature reduction, affinity clustering and multilevel Ensemble Support Vector Machine. The proposed model performance is analyzed over two datasets i.e. NSL-KDD and UNSW-NB 15 dataset and achieved approx. 12% of efficiency over other existing work.

Keywords : Intrusion Detection System, Affinity Clustering, Ensemble Support Vector Machine, NSL-KDD, UNSW-NB 15, Detection Rate

I. INTRODUCTION

Intrusion Detection System (IDS) are implemented over host computer or network asa security tool or application to avoid malicious attacks over them. IDS is implemented as individual host based or network based. The function of host-based IDS is to detect attack over a single computer or host computer. But when IDS is applied over multiple systems, connected in a network, then it is termed as Network Intrusion Detection System (NIDS). In NIDS, the intrusions are detected and analyzed over network traffic and it is installed over network gateway to capture data packets and analyze its behavior. Host based IDS is categorized in four types [1][2]:

- File System Monitors
- Log file analyzers
- Connection analyzers
- Kernel-based IDS

Similarly, NIDS is mainly categorized into two types:

- Signature-based
- Anomaly based

In signature-based NIDS, the data packets are analyzed and their behavior or signature is stored in database. In such type of NIDS, only known attacks are detected as their signature is stored in database. But in anomaly-based NIDS system, the behavior of data packets is analyzed as how much it is deviated from normal behavior and are capable to detecting new attacks [3].

II. RELATED WORK

Ryan et al. [3] proposed anomaly detection model by using back-propagation 3-layer Multi-Layer Perceptron (MLP) to detect possible attack in network. This model analyzed each session logs and analyzed the behavior of data packets. The MLP model analyzed 22/24 anomaly cases correctly.

Ghosh et al. [4] proposed a similar model as [3]. In this model, the prediction of coming data packets are analyzed by generalized study of previous known packets. For analysis this model is designed by applying artificial neural networks (ANNs) in order to detect malicious behavior of coming network traffic. Similar approach was applied in [5] and [6] using Self-Organizing Maps (SOM). These models are trained on the basis of previously recognized packets and tested over real-time data packets or network traffic.

Meng et al. [8] analyzed network anomaly by using artificial neural network, support vector machine and decision treemachine learning approach. The performance of decision tree gives better result. Decision tree also detects U2R and R2L attacks with high efficiency as such attacks occurs with low frequency.

Feng et al. [9] integrated SVM and Self-Organized Ant Colony Network for intrusion detection. This model is hybrid by merging classification and clustering techniques.

Manjula et al. [10] proposed a predictive intrusion detection tool using machine learning classification algorithms such as Logistic Regression, Gaussian Naive Bayes, Support Vector Machine and Random Forest. Out of all random forest gives highest accuracy rate.

Saad Mohamed et al. [11] presented a hybrid approach to anomaly detection using of K-means clustering and Sequential Minimal Optimization (SMO) classification.

Shoneet al. [12] proposes network-based intrusion detection system that used the nonsymmetric deep autoencoder (NDAE) for feature learning and anomaly detection. The proposed model was analyzed over KDD-99 and NSL-KDD datasets. The proposed model gives more accurate result on KDD-99 dataset. But in this paper, false alarm rate of U2R is 50 which is very high and requires more training time and samples for improved detection accuracy.

III. PROPOSED MODEL

Intrusion or anomaly detection is one of the biggest challenges over today's such fast growing communication system and internet services. As applications over internet increases, it increases the security threats. So, to detect such malicious activities over internet, Intrusion Detection Systems (IDS) considered to be as effective security tool.According to literature review it is seen that machine learning techniques is considered to be more widely used application to design more efficient intrusion detection tool for achieving high accuracy rate as well as low false alarm rate [13]-[15].

More accurate predictive model can be built for such large network dataset by applying supervised machine learning approach which is quite impossible with traditional security tools.

The machine learning based IDS learns the pattern of data packets captured in the network to determine its behavior to be as anomaly or normal. But traditional IDS system will only identify such attacks which are known to machine. It cannot determine the unknown attack and its behavior.

In this section, the proposed model is described which is hybrid in nature and combines a set of algorithms to improves its efficiency to determine the nature of data packets and to predict type of attack. The NSL-KDD and UNSW-NB 15 datasets are used as a benchmark to evaluate the performance of the proposed model.

The algorithm flow of the proposed method is described as follows:

All the symbolic attributes of the dataset is converted into numerical form. For example, protocol type TCP, UDP and ICMP is converted into 1, 2, 3 respectively.
Normalization of dataset using statistical normalization. This makes the dataset values in a particular range and thus reduces the computational complexity for further classification algorithm.

Feature co-relation are evaluated using f_score correlation which finds related features. This also reduces the dimension of dataset for further processing.

Data Clustering is performed over corelated features. For this affinity clustering is applied.

Further clusters are considered to be as training dataset to learn classifier and testing dataset to evaluate the performance of system.

Training multi-level ensemble SVM with randomly selected sets from training samples.

Test dataset is applied to trained ensemble SVM to evaluate its attack type and performance of proposed model.

The proposed algorithm flow diagram of intrusion detection model is illustrated in fig. 1. The proposed framework consists of following stages:

Pre-processing

This stage purpose is to preprocess the database file in which there is conversion of symbolic attributes in numerical is done.

Post-Processing Phase
After pre-processing, the feature vector matrix ofdatasets (NSL-KDD and UNSW-NB 15)are further send to F-Score correlation algorithm to determine co-related features.F-Score correlation is an algorithm which is used to determine the direct or indirect relation among data values. Suppose the dataset contained n features then F_score determines n*n relation among them.

Let's considers:
Input dataset (xn), where n=1, 2… k
Number of classes = c, where (c>=2)
Then F-Score correlation of ithfeature with jthfeature is determined as in eqn (i):

$$F_i = \frac{\sum_{j=1}^{c}(\bar{x}_i^j - \bar{x}_i)^2}{\sum_{j=1}^{c}\frac{1}{n_j - 1}\sum_{n=1}^{n_j}(x_{n,i}^j - x_i^j)^2} \qquad \text{(i)}$$

Where $\bar{x}_i$= mean of ithfeature of entire dataset.
$\bar{x}_i^j$=mean of ith feature of the jth dataset.
$\overline{x_{n,i}}^j$= ith feature of the nth instance in the jth dataset.
The numerator of above-mentioned equation, eqn(i), represents the discrimination among classes in the dataset. Whereas the denominator represents the discrimination within each of the classes in the

dataset. If this F-Score valueis smaller among feature set then that features are not related to each other whereas if the value is higher, then that feature is highly related and can be added to the feature subset.

A. Clustering

The reduced feature dataset is clustered in this phase. For clustering the data, affinity clustering is used in this paper. Table I shows the number of instances before and after applying affinity clustering. Affinity clustering contains a set of "exemplars"that is used to cluster data on the basis of their similarity. A pair of similar data values are input in affinity clustering such that, sim(p, q) where p, q = 1, 2,...,N. The algorithm finds such exemplars which have maximum similarity. The maximum similarity is calculated as the sum total of similarities among all data values in all exemplars.For determining the similarity matrix among all data values, two messages are pass as an argument among them:

- Responsibility, r[p,q]
- Availability, a[p,q]

Responsibility message is passed to show that data value q is well suited to be considered as exemplar for data value p.Similarly, availabilitymessage is passed from data value q to data value p to show that q is suitable to be as an exemplar for p. Initially, r[p,q] and a[p,q] is set to zero and updated accordingly eqn (ii) and eqn (iii):

$$r[p,q] = (1 - \lambda)\rho[p,q] + \lambda r[p,q] \quad \text{(ii)}$$

$$a[p,q] = (1 - \lambda)\alpha[p,q] + \lambda a[p,q] \quad \text{(iii)}$$

Where.

λ = damping factor to avoid numerical oscillations
$\rho[p,q]$ =propagating responsibility
$\alpha[p,q]$ = propagating availability.

Category	No.of Instances	Frequency Normalization	Statistical Normalization
Normal	6690	1242	1207
Dos	4660	1034	943
Probe	1136	137	128
R2L	106	28	29
U2R	5	2	2
Total	12597	2443	2309

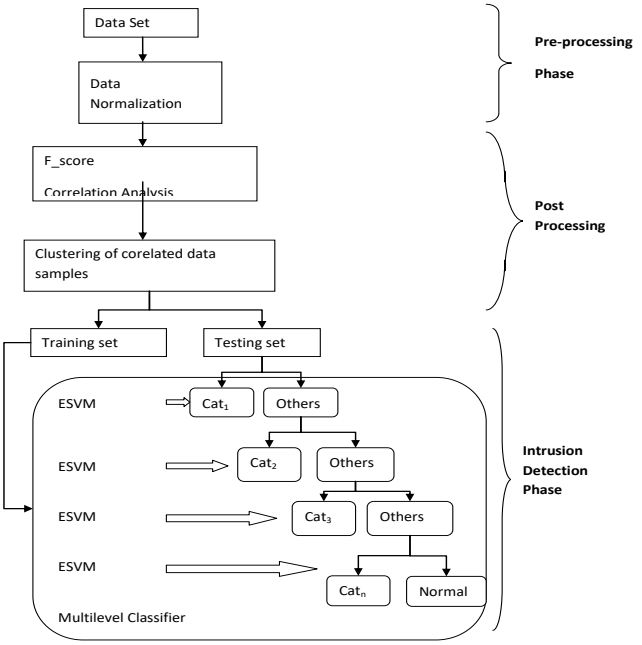

Figure 1 : Proposed Flow Diagram of Intrusion Detection System

B. Intrusion Detection

For intrusion detection or classification dataset multilevel classifier is used. In this research work three multilevel classifier performance is analyzed i.e. Multilevel ensemble support vector machineis applied on attack and Normal packets and are classified using ESVM classifier algorithm (as shown in figure 2).

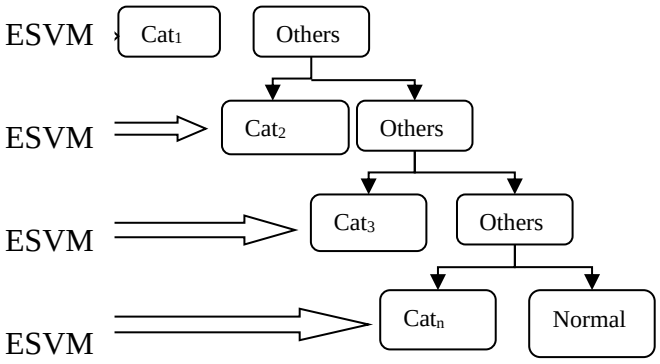

Figure 2 : Multilevel ESVM Classifier

Ensemble Support Vector Machine

The support vector machine (SVM) algorithm has some drawbacks. One of the problems associated with SVM is that it was designed and well suited for binary-class classification problems. For multi-class classification, SVM has to be combined multiple times. After combining, SVM multiple times, its performance degrades and as per study ensemble techniques gives more efficient result as compared to traditional SVM method. Another drawback of SVM is that during learning process with large feature sets containing dataset, it consumes much processing time to converge. In order to reduce such time complexity approximation algorithms are needed to be applied. The use of any approximation algorithms would degrade the performance efficiency of SVM classification. In order to overcome the drawbacks of traditional SVM machine learning algorithm, this paper proposed and ensemble SVM classification algorithm for multilevel classification of intrusion detection system as discussed in above section. The proposed ensemble SVMimproves the efficiency and accuracy rate of theclassificationproblem.

In ensemble SVM, each SVM module is trained independently with random training samples and correctly classified the data samples of each SVM. Similarly, all other data values are trained independently on individual SVM module and finally integrated as an ensemble or combination of several SVMs which will expand the correctly classified area incrementally. This proposed ensemble SVM will performs better in case of multi-class classification problems.In fig. 3, a generalized architecture of proposed SVM is given and explained.

During training phase of proposed ensemble SVM architecture, each SVM module is trained individually with randomly selected training samples from the dataset.This makes each trained SVM module be different from each other. Each SVM module can be trained with different training sets and rules. Bagging, random selection and boosting selection strategies can be used to select training samples. In this proposed architecture bagging rules are taken as a base for ensemble SVM in which each SVM modules are trained individually and further they are aggregated by applying combination method. During testing phase, the aggregate strategy or voting strategy among all SVM module will decide the test data class label.

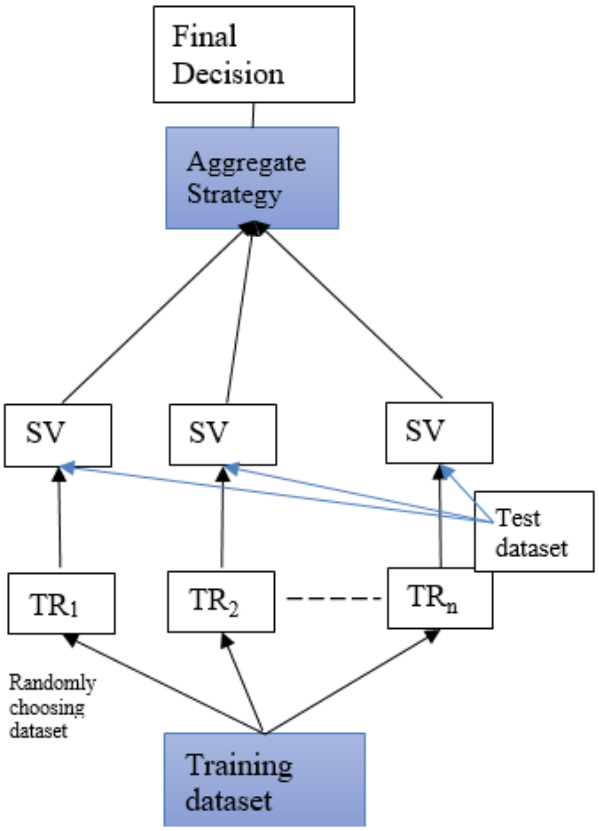

Figure 3 : Architecture of the Ensemble SVM

In ensemble SVM architecture, "n" training samples sets are constructed with "n" individual SVMs modules. To achieve higher efficiency, different training sample sets are taken in order to improve the aggregation result with higher efficiency.

IV. RESULT ANALYSIS

The performance of proposed methodology is evaluated on the basis of following parameters:

Accuracy = (TP+TN)/(TP+TN+FP+FN)*100 (iv)

Precision =TP/(TP+FP)*100 (v)

Recall= TP/(TP+FN)*100 (vi)

F_Measure = 2*Precision*Recall/(Precision + Recall) (vii)

False Negative Rate (FNR) = FN/(FN+TP) *100 (viii)

False Positive Rate (FPR) =FP/(FP+TN) *100 (ix)

False Alarm Rate (FAR)= (FPR+FNR)/2 (x)

Where,

True Positive (TP) = If predicted and actual data packet, both are anomaly in nature.

True Positive (TP) = If predicted and actual data packet, both are anomaly in nature.

True Positive (TP) = If predicted and actual nature of data packet, both, are anomaly.

True Negative (TN) = If predicted and actual nature of data packet, both, are not anomaly.

False Positive (FP) = If predicted nature of data packet is anomaly but actual nature of data packet is normal.

False Negative (FN) = If predicted nature of data packet is normal but actual nature of data packet is anomaly.

Table II-III shows the performance evaluation of multilevel classification algorithm over NSL-KDD dataset. From the result analysis it has been analyzed that performance rate of multilevel ensemble SVM classification achieved best result.Table IV shows the performance evaluation of multilevel classification algorithm over UNSW-NB 15 Dataset. Different test samples are taken to analyze performance on this dataset and proposed algorithm also outperforms better.

Table II : Performance Evaluation on NSL-KDD Dataset

Performance	Accuracy	Precision	Recall	F_Measure	False alarm rate
Sample 1	99.91	99.44	99.64	99.54	0.20
Sample 2	99.79	98.62	99.80	99.20	12.6
Sample 3	99.82	99.39	98.88	99.13	0.58
Sample 4	99.78	99.52	98.60	99.056	9.05
Sample 5	99.79	98.78	99.28	99.03	0.43
MESVM Average	99.82	99.15	99.24	99.19	4.58
Shone et al [12]	85.42	100	85.42	87.37	14.58

Table III : Category wise Evaluation on NSL-KDD Dataset

Attack Class	Accuracy		Precision		Recall		F_measure		False Alarm Rate	
	MESVM	Shone et al [12]	MESVM	Shone et al [12]	MESVM	Shone et al [17]	MESVM	Shone et al [17]	MESVM	Shone et al [17]
DoS	99.37	94.5	95.59	100	98.73	94.5	97.14	97.2	0.903	1.07
Probe	100	94.6	100	100	100	94.6	100	97.2	0	16.8
U2R	100	2.7	100	100	100	2.7	100	5.26	0	50
R2L	100	3.82	100	100	100	3.82	100	7.36	0	3.45
Total	99.84	48.9	98.89	100	99.68	48.9	99.28	51.7	0.225	17.8

Table IV : Performance Evaluation on UNSW-NB 15 Dataset

Performance	Accuracy	Precision	Recall	F_Measure	False alarm rate
TestSet-1	96.84	100	92.37	96.0	3.81
TestSet-2	95.66	100	88.79	94.06	5.60
TestSet-3	97.58	100	92.70	96.21	3.64
TestSet-4	97.90	100	94.11	96.96	2.94
TestSet-5	98.61	100	96	97.95	2
Average	97.318	100	92.794	96.236	3.598

V. CONCLUSION

This paper proposes a new hybrid multilevel Ensemble-SVM based intrusion detection or anomaly detection technique. The proposed model is designed to determine co-relation among features or attributes of dataset. By applying co-relation among them related features which are dependent on each other are found. This is performed by f_score correlation algorithm. Then further clustering of different attack categories are performed using affinity clustering which reduces the overall processing efficiency as well time. The simulation results are performed on NSL-KDD dataset as well UNSW-NB 15 dataset and result shows accuracy improvement of 12% and false alarm rate shows improvement of approx. 12%.

VI. REFERENCES

[1]. De Boer, P., Pels, M, "Host-Based Intrusion Detection Systems", Amsterdam University, Amsterdam, 2005.

[2]. Garcia-Teodoro, P., "Anomaly-based network intrusion detection: techniques", systems and challenges. Comput. Security vol. 28.issue, pp. 18–28, 2009.

[3]. J. Ryan, M. Lin, and R. Miikkulainen, "Intrusion Detection with Neural Networks," Conference in Neural Information Processing Systems, 943–949.

[4]. A. K. Ghosh and A. Schwartzbard, "A Study in Using Neural Networks for Anomaly and Misuse

Detection," Conference on USENIX Security Symposium, Volume 8, pp. 12–12, 1999.

[5]. P. L. Nur, A. N. Zincir-heywood, and M. I. Heywood, "Host-Based Intrusion Detection Using Self-Organizing Maps," in Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 1714–1719, 2002.

[6]. K. Labib and R. Vemuri, "NSOM: A Real-Time Network-Based Intrusion Detection System Using Self-Organizing Maps," 2000.

[7]. Sharma, R.K., Kalita, H.K., Issac, B., "Different firewall techniques: a survey", International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2014.

[8]. Meng, Y.-X., "The practice on using machine learning for network anomaly intrusion detection", International Conference on Machine Learning and Cybernetics (ICMLC), vol. 2, IEEE, 2011.

[9]. Feng, W., "Mining network data for intrusion detection through combining SVMs with ant colony networks", Future Generation Computer, System, vol. 37, pp. 127–140, 2014.

[10]. Manjula C. Belavagi and BalachandraMuniyal, "Performance Evaluation of Supervised Machine Learning Algorithms for Intrusion Detection, Procedia Computer Science", Elsevier, 2016.

[11]. Saad Mohamed Ali Mohamed Gadal and Rania A. Mokhtar, "Anomaly Detection Approach using Hybrid Algorithm of Data Mining Technique", International Conference on Communication, Control, Computing and Electronics Engineering, IEEE, 2017.

[12]. N. Shone, T. N. Ngoc, V. D. Phai and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," in IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41-50, Feb. 2018.

[13]. He, L., "An improved intrusion detection based on neural network and fuzzy algorithm. Journal of Networks, vol. 9, issue 5, pp. 1274–1280, 2014.

[14]. Hoque, M. S., Mukit, M. A. ,&Bikas, M. A. N., "An implementation of intrusion detection system using genetic algorithm", International Journal of Network Security & Its Applications, vol 4, issue 2, pp. 109–120, 2012.

[15]. Prasanta Gogoi, D.K. Bhattacharyya, B. Borah1 and Juga, K. Kalita, "MLH-IDS: A Multi-Level Hybrid Intrusion Detection Method", The Computer Journal, Vol. 57 issue 4, pp. 602-623, 2014.

Cite this article as :

Sadhana Patidar, Priyanka Parihar, Chetan Agrawal, "Multilevel Intrusion Detection System with Affinity Clustering and Ensemble SVM", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 4, pp. 522-529, July-August 2020. Available at doi : https://doi.org/10.32628/CSEIT206431
Journal URL : http://ijsrcseit.com/CSEIT206431