A Two Layer Machine Learning System for Intrusion Detection Based on Random Forest and Support Vector Machine

Abstract*— Unauthorized access or intrusion is a massive threatening issue in the modern era. This study focuses on designing a model for an ideal intrusion detection system capable of defending a network by alerting the admins upon detecting any sorts of malicious activities. The study proposes a two layered anomaly-based detection model that uses filter co-relation method for dimensionality reduction along with Random forest and Support Vector Machine as its classifiers. It achieved a very good detection rate against all sorts of attacks including a low rate of false alarms as well. The contribution of this study is that it could be of a major help to the computer scientists designing good intrusion detection systems to keep an industry or organization safe from the cyber threats as it has achieved the desired qualities of a functional IDS model.

Keywords—Intrusion Detection, Anomaly, Feature Selection, Random Forest, SVM

I. INTRODUCTION

One of the most significant developments in modern history is the Internet. While most individuals use the Internet for constructive purposes, it is used by certain individuals as an opportunity for malicious intent. As more users are connected to the Internet and machines are more prevalent in our everyday lives, the Internet and its connected computers are gradually becoming more attractive targets for attacks [1]. These attacks could lead to network flooding, information theft or even hacking.

Intrusion refers to the unwanted interruption or unauthorized access in a network or a system. It is the process of entering a computer system by manipulating the security systems. Intrusion can lead to networking flooding, information theft or even hacking. An IDS (intrusion detection system) is a program or device that focuses on observing a network to detect malicious behavior or violations of policy. The system administrator is informed of any malicious behavior or compromise using a security information and event management system [2]. The detector eradicates redundant information from the inspected data and then determines the possibility of an intrusion perceived as a sign of these activities. In addition, it is difficult to keep the signature sets for intrusion detection up to date as the number of vulnerabilities continuously found continues to increase [3]. Anomalous activities can be divided into four categories- DoS, Probe, U2R and R2L.

In this study, a two layered intrusion detection system is proposed that uses Filter Co-relation Method for selecting the features to reduce dimensionality. Support Vector Machine and Random Forest algorithms are used as the classifiers in the two layers. This study proposes a model which can detect anomalies of all categories at a low computational cost and false alarm rate.

II. RELATED WORKS

The problem of the study mentions the lack of good detection rates and higher false alarm rates against all sorts of attacks. This study proposes a two layered model, capable of detecting anomalies. In state-of-the-art works, different anomaly detection models have been proposed using supervised machine learning approaches.

In [6], authors proposed an intrusion detection system based on SVM along with the combination of a hierarchical clustering algorithm. They were able to detect common types of attacks easily but failed to detect rare attacks efficiently. In paper [4], they have proposed a two-tier anomaly detection model that uses LDA for feature selection and in the two tiers, they have used Naïve Bayes and KNN - Classifier as their classification algorithm. This study gained a very good detection rate against unseen and rare attacks like U2R and R2L. But detection rate against the seen ones (DoS, Probe) were comparatively lower than the existing models. In paper [7], an unsupervised anomaly detection model is proposed that use clustering method based on subspace algorithm to inspect and detect anomalous activities. The model failed to mention how specific attack types should be dealt with. In paper [8], they have proposed a two layered model that uses Random Forest and SVM as their classifiers. The study achieved very good detection rates. In paper [10], the writers explained the functionalities of the feature selection process and stated the significant benefits it is able to provide a dataset ready to be classified.

In comparison with most of the coexisting models, the proposed model was able to gain a higher detection rate but the detection rate against rare attacks was not close to 100%. Another limitation is that the proposed model used simple and common techniques.

III. PROPOSED MODEL

In the proposed two-layered model, the first layer performs data training and preprocessing along with dimension reduction. Here, classification has been performed using Support Vector Machine (SVM). At the second layer of the proposed model, distinguishment in between anomalous activities and normal behavior has been ensured by the implementation of a specific classification using Random Forest algorithm.

For better disjunction between unusual and normal behaviors, feature selection process would be performed prior to performing classification. In the newly formed feature space containing five dimensions, filter co-relation feature selection method would be performed which would select five most effective features out of the existing 41 and then the classification would be performed. After that SVM algorithm would be applied to the dataset as a classifier which would distinguish between the normal and the anomalous data. Next to improve the performance of the

classifiers that have been used, a technique using RF-d tree would be applied to hoard the diminished data training set. By using two machine learning classifiers to perform its classification, the model could be considered as a two layered one.

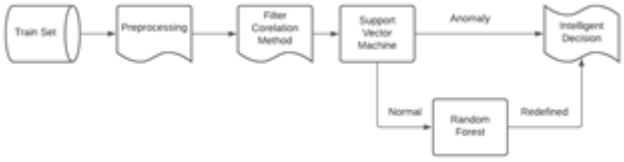

Fig-1: Research framework for the proposed method

IV. Feature Selection and Algorithms

A. *Feature Selection*

Feature selection could be defined as a pre-processing stage widely used in both data mining and machine learning. It helps to define the features that are convenient for class prediction [11]. Feature selection is a technique that is often used to minimize the number of total features in a certain application requiring high dimensional data. The technique of selecting features can improve learning performance, improving predictive precision and decreasing the complexity of the results learned [9].

For feature selection procedure, the linear correlation co-efficient method is the most common and the most used measure all over the world. For a variables pair (X, Y), the linear correlation coefficient r could be derived by the following formula,

$$\frac{\sum_i (x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2}\sqrt{\sum_i (y_i - \bar{y}_i)^2}} \dots\dots\dots (1)$$

Here, $\bar{x}_i$ and $\bar{y}_i$ are considered as the mean of X and Y, respectively. The value of r lies in between -1 and 1. If X and Y are correlated, r is defined as the value of either 1 or -1. If X and Y are not correlated or in other words, if they are independent, r would be considered as 0 [9].

B. *Support Vector Machine (SVM)*

Support Vector Machine, generally referred to as SVM could be considered as a supervised learning approach [8]. It creates an N-dimensional hyperplane and ideally separates the information into different categories. SVM is capable of performing classifications. Within the fundamental classification, SVM can classify the data into two different categories.

Given a training set of occasions with labeled sets {(x, y)}, where y is the name of occurrence x, SVM works by maximizing the edge to achieve the most effective execution of performance in terms of classification [6].

The computation issue of a margin-maximizing boundary function in SVM can be indicated by the subsequent QP problem:

Minimize:

$$W(\alpha) = \sum_{i=1}^{l} \alpha_i + \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} y_i y_j \alpha_i \alpha_j X_i X_j \dots\dots\dots (2)$$

Subject to:

$$\sum_{\substack{i=1 \\ 0 \le \alpha_i \le C}}^{l} y_i \alpha_i = 0 \dots\dots\dots (3)$$

Here, the number of training data is represented by 1, a is l variable vector where every component a_i communicates with the training data xi. Soft margin parameter C supervises the impact of the deviations in the training data. The data points situated closest to the class boundary could be expressed as the boundary functions SVM. Equation (3) is the computation of vector a, where every component indicates the weight of each of the data points. The support vectors are the data points possessing a_i value greater than zero. Data points containing a_i value equal to 0 are the non-support vectors on the other hand. SVM maps all training data points into a greater dimensional space. SVM is able to discover a differentiating hyperplane containing a maximal margin in this greater dimensional space [6] [8].

C. *Random Forest*

A group of unpruned classification or regression trees is referred to as the random forest algorithm[12]. On the other hand, an ensemble classifier used to boost accuracy is Random Forest (RF). The random forest consists of several decision trees.

Many classification trees can be created by Random Forest and every tree is developed by a bootstrap sample that is different from the original data. The task is done by using a tree classification algorithm. After the formation of the forest, a completely new object is placed for classification which needs to be categorized. It is performed on each and every tree of the forest[8].

Random Forest can be implemented with scikit-learn software. The ultimate feature importance of a level in Random Forest algorithm is its average on all over the trees. Summation of the importance of feature values on each and every tree is computed and divided by the entire number of trees using the following formula:

$$RFfi_i = \frac{\sum_{jealltrees}\, norm\, fi_{ij}}{T} \dots\dots\dots (4)$$

Here,

- RFfi sub(i)= the importance of feature i calculated from all trees in the Random Forest model
- normfi sub(ij)= the normalized feature importance for i in tree j
- T = total number of trees

V. Dataset and Implementation

A. *NSL- KDD Dataset*

NSL-KDD benchmark dataset was used to test the proposed model. The KDD99 dataset [12] is the newer and improved edition of the NSL-KDD dataset. This data set was implemented for competition in NIDS. A host-to - host

link is included in each NSL-KDD record, which includes 41 distinct features (e.g. Num failed, logins, Root shell, Count) named either as natural or one of the specific names of the attacks. Since the original KDD dataset had some defects, the NSL-KDD dataset was used to test the proposed model [12]. KDD99 is extracted from DARPA by extracting functions. Then, by eliminating the duplicates and reducing the number, NSL-KDD is derived from KDD99. Two training sets and one test set containing DoS, Probe, U2R, and R2L attack classes are included in the dataset.

B. Data Preprocessing

The original dataset will be transformed to a normal form in the data preprocessing stage so that it can achieve improved decision-making and diminish computational overhead [4], it will be as follows:

- All categorical feature values would be allotted a distinctive integer number as follows: (TCP = 1, UDP = 1, ICMP = 1).

- To avoid any sort of bias, features containing continuous values would be converted to discrete values by using base 2 logarithm. After that the result discrete values would be converted to integers using the following equation considering z as each of the Continuous values,

$$\text{if } (z \geqslant 2)\ z = \log(z + 1) \quad \ldots\ldots (5)$$

For better classification, normalization process has been implemented after that and attack labels has been grouped into a normal and four different classes [4].

VI. Experiment Result and Discussion

The proposed model has been trained by both of the following training sets (Train_20%, $Train^+$) and then it has been evaluated by the given test set. NSL-KDD contains 22544 instances and the evaluation has been done with the help of projection matrix (W_r). So, it could be stated that most of the results given here in the research are evaluated mostly by the test set. But the proposed model is capable of almost solving this issue by using Random Forest as its second classifier. Following the normalization process of the test set, the projection matrix ($Test^+$) that was for the most part obtained from the training set, has been applied to the test set. At this step around 45 iterations have been tested. Considering this experiment's detection rates, k=5 has been selected in order to obtain greater detection rate against the rare and unseen attacks in comparison with the other chosen values.

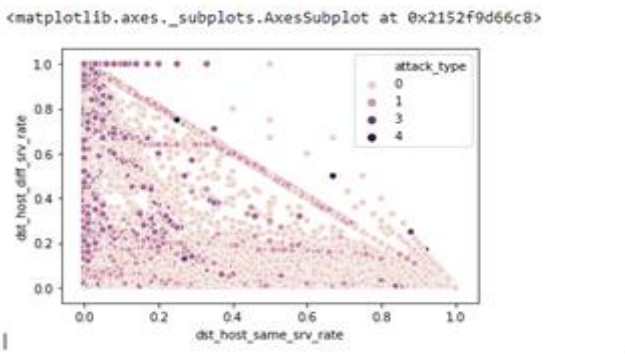

Fig-2: Generated image of the total number of attacks in the $Test^+$ dataset.

Fig-2, shows the different attack types that are present in the test+ dataset. It also reflects on the total number of the specific attack types. Here 0 stands for DoS, 1 for Probe, 3 for U2R and 4 for R2L. The generated image is the visible explanation of the attack types.

Table-1: Distribution of NSL-KDD Data Classes.

Dataset	Total Records	Normal	Probe	Dos	U2R	R2L
Train_20%	25192	13449	2289	9234	11	209
$Train^+$	125973	67343	11656	45927	52	995
$Test^+$	22544	9711	2421	7458	67	2887

Table-1 represents the number of attacks in the whole dataset. It is visible that the biggest number of attacks are present in the test data set and the train_20% possesses the lowest number of attacks. This makes the test set more threatening than the rest.

Table-2: Anomalous behavior detection rate in different levels of classifications (%).

Level	Probe	DoS	U2R	R2L
First Level of Classification	98	97	11	9
Second Level of Classification	88	95	69	54

In Table-2, it can be seen that after applying SVM in the first layer, the gained detection rate of the common attack types such as DoS and Probe has been satisfying. But the detection rate of the rare ones has been poor. In the second layer the results have changed drastically. After applying Random forest classifier, the detection rates of all the attack types have been satisfactory.

Table-3: Confusion matrix of existing and proposed models for the rare attacks.

Model	Normal	Probe	DoS	U2R	R2L
The proposed model	99	88	95	69	54
SVM with BIRCH clustering [6]	99.3	99.5	97.5	28.8	19.7
SVM with Random Forest [8]	92.99	63.81	96.32	34.48	45.1
Two Tier with Naïve Bayes and KNN [4]	94.56	79.76	84.68	67.16	34.81

In Table-3, a comparison in between the co-existing IDS models and the proposed model has been shown. It is visible that the proposed model has better detection rate against all of the existing ones. Difference of algorithms has brought about huge changes in terms of detection rates. Used techniques contribute a lot in these significant changes. Difference in techniques and classifiers contribute a lot in these significant changes. SVM with BIRCH clustering model [6] contains one layer but uses a mixture of hierarchical clustering algorithms. The rest of them consist of two layers similar to the proposed model. Detection rates are better in the multiple layers in comparison with the one layered model.

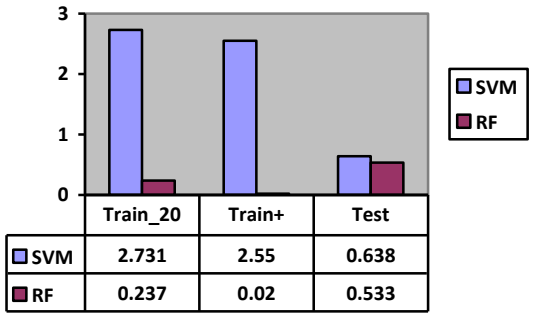

Fig-3: Bar diagram for the false alarm rate of the proposed model.

Analyzing Fig-3, it can be seen that in the first layer (after applying SVM as the classifier), the false alarm rate was less than the coexisting models. But in the second layer the rate decreased even more making the proposed model more efficient to use. In the second layer, the proposed model obtained 0.237 % false alarm using Train_20 % dataset and 0.0204 % in using $Train^+$ as training set and 0.533% in using $Test^+$ respectively for testing the proposed model.

Table-4: Comparison of detection rate and false alarm rate in between the proposed model and coexisting Single-layer model in terms of Test+ (%)

Method	Dataset	Detection Rate	False Alarm Rate
Proposed Model	Train-20 %	93.89	0.237
Naïve Bayes [12]	Train_20%	76.56	N/A
Random Forest [11]	Train-20 %	80.67	N/A
SVM [6]	Train-20 %	69.52	N/A

In Table-4, a comparison has been shown between the proposed model and the existing one layered model. It can be seen that the overall detection rate of the proposed two layered model is far better than the existing one layered model. The false alarm rate is also comparatively better in the proposed model making it even more desirable.

After analyzing the derived detection and false alarm rates it is safe to state that other works that had been done prior to this were either efficient in detecting common and seen attacks or the rare attacks. Nevertheless, the proposed model has achieved higher detection rates against both common and rare types of attacks. The proposed model has a comparatively low false alarm rate which adds as an advantage to the model. Overall, the detection rate of the proposed model is 93.89% which is better than most other coexisting models. The detection rate for the specific attack types are 99%, 95%, 88%, 69%, 54% respectively for Normal, DoS, Probe, U2R and R2L. it could be seen from the result section that the rates are higher than the others making the proposed model the better one.

VII. CONCLUSION

In this study, the writers have explored the field of intrusion detection system and proposed a two layered model that could work against common and rare attacks. The challenges of designing a working model cable of detecting attacks has been described in the study. The implementation was done by a machine learning parametric approach using various algorithms and techniques. Finally, it is safe to state that the aim of the proposed model has been fulfilled as it has higher detection rate for all types of attacks, lower false alarm rate and lower computational cost in comparison with the coexisting models. In near future, aim of the proposed model is to use nonparametric techniques to obtain more useful features and to use fuzzy clustering techniques to improve the detection rate by separating the normal instances from the anomalous ones. It could be an efficient solution for the organizations and industries to keep their server away from anomalous activities thus ensure their server safety.

VIII. REFERENCES

[1] P. K. Chan, M. V. Mahoney, and M. H. Arshad, “Learning Rules and Clusters for Anomaly Detection in Network Traffic,” *Manag. Cyber Threat.*, pp. 81–99, 2005, doi: 10.1007/0-387-24230-9_3.

[2] D. E. Denning, “An Intrusion-Detection Model,” *IEEE Trans. Softw. Eng.*, 1987, doi: 10.1109/TSE.1987.232894.

[3] Z. Tan, A. Jamdagni, X. He, and P. Nanda, “Network intrusion detection based on LDA for payload feature selection,” *2010 IEEE Globecom Work. GC'10*, pp. 1545–1549, 2010, doi: 10.1109/GLOCOMW.2010.5700198.

[4] H. H. Pajouh, G. H. Dastghaibyfard, and S. Hashemi, “Two-tier network anomaly detection model: a machine learning approach,” *J. Intell. Inf. Syst.*, vol. 48, no. 1, pp. 61–74, 2017, doi: 10.1007/s10844-015-0388-x.

[5] A. N. Toosi and M. Kahani, “A new approach to intrusion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers,” *Comput. Commun.*, vol. 30, no. 10, pp. 2201–2212, 2007, doi: 10.1016/j.comcom.2007.05.002.

[6] S. J. Horng *et al.*, “A novel intrusion detection system based on hierarchical clustering and support vector machines,” *Expert Syst. Appl.*, vol. 38, no. 1, pp. 306–313, 2011, doi: 10.1016/j.eswa.2010.06.066.

[7] K. Leung and C. Leckie, “Unsupervised anomaly detection in network intrusion detection using clusters,” *Conf. Res. Pract. Inf. Technol. Ser.*, vol. 38, no. January, pp. 333–342, 2005.

[8] M. A. M. Hasan, M. Nasser, B. Pal, and S. Ahmad, “Support Vector Machine and Random Forest Modeling for Intrusion Detection System (IDS),” *J. Intell. Learn. Syst. Appl.*, vol. 06, no. 01, pp. 45–52, 2014, doi: 10.4236/jilsa.2014.61005.

[9] S. Y. Jiang and L. X. Wang, “Efficient feature selection based on correlation measure between continuous and discrete features,” *Inf. Process. Lett.*, vol. 116, no. 2, pp. 203–215, 2016, doi: 10.1016/j.ipl.2015.07.005.

[10] M. A. M. Hasan, M. Nasser, S. Ahmad, and K. I. Molla, “Feature Selection for Intrusion Detection Using Random Forest,” *J. Inf. Secur.*, vol. 07, no. 03, pp. 129–140, 2016, doi: 10.4236/jis.2016.73009.

[11] V. Svetnik, A. Liaw, C. Tong, J. Christopher Culberson, R. P. Sheridan, and B. P. Feuston, “Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling,” *J. Chem. Inf. Comput. Sci.*, vol. 43, no. 6, pp. 1947–1958, 2003, doi: 10.1021/ci034160g.

[12] Institute of Electrical and Electronics Engineers. and IEEE ITSS., “2010 Sixth International Symposium on Information Assurance and Security (IAS) : [Atlanta, GA, USA : Georgia Tech Global Learning Center, Atlanta : 23-25 Aug. 2010],” pp. 5–10, 2010.