Applying Machine Learning to Anomaly-Based Intrusion Detection Systems

Fekadu Yihunie [1], Eman Abdelfattah [2], Amish Regmi [3]
School of Computer Science and Engineering[1], School of Theoretical & Applied Science [2, 3]
Sacred Heart University[1], Ramapo College of New Jersey[2, 3]
Fairfield, CT[1], Mahwah, NJ[2, 3]
yihunief@sacredheart.edu [1], eabdelfa@ramapo.edu [2], aregmi@ramapo.edu [3]

***Abstract*—The enormous growth of Internet-based traffic exposes corporate networks with a wide variety of vulnerabilities. Intrusive traffics are affecting the normal functionality of network's operation by consuming corporate resources and time. Efficient ways of identifying, protecting, and mitigating from intrusive incidents enhance productivity. As Intrusion Detection System (IDS) is hosted in the network and at the user machine level to oversee the malicious traffic in the network and at the individual computer, it is one of the critical components of a network and host security. Unsupervised anomaly traffic detection techniques are improving over time. This research aims to find an efficient classifier that detects anomaly traffic from NSL-KDD dataset with high accuracy level and minimal error rate by experimenting with five machine learning techniques. Five binary classifiers: Stochastic Gradient Decent, Random Forests, Logistic Regression, Support Vector Machine, and Sequential Model are tested and validated to produce the result. The outcome demonstrates that Random Forest Classifier outperforms the other four classifiers with and without applying the normalization process to the dataset.**

Index Terms – Intrusion Detection systems (IDSs), NSL-KDD, Machine Learning;

I. INTRODUCTION

Nowadays Internet-based applications and dependency of cloud services are increasing exponentially. Corporates are moving their core businesses and services to the cloud. The advancements of cloud computing, IoT, social web, mobile technology, modern peer-to-peer network are changing applications, services, and the way people communicate with each other. Furthermore, organizations are exposing their network infrastructure and systems to the public due to business need and external factors and demands.

On the other hand, the complexity and flow of malicious traffic are rapidly increasing. Targeted organizations are attacked in various techniques by organized cyber-terrorist, nation sponsored hackers, hacktivists and script kiddies. Detecting, protecting and managing intrusive traffic are challenging and costly, as organizations strive to comply with national standards, organization-specific standards and compliance requirements.

Firewalls, intrusion detection and prevention systems, web content and URL filtering are few of the recommended forefront systems in a secured network infrastructure to protect internal services from attacks launched by intruders. The advancement of attacking techniques and the intelligence of organized criminals make it difficult to completely protect sensitive information from theft, disclosure, and Denial of Service (DoS) attacks. Therefore, researchers are studying various anomaly traffic detection systems using machine learning techniques to improve traditional signature-based intrusion detection systems.

An intrusion detection system can detect suspicious activity and attacks. But, if the network is too busy, then the intruder detection can miss frames. Intrusion detection systems analyze real-time traffic and audit files. Traffic is labeled based on the Intrusion detection systems' interpretation as either true positive, false positive, true negative or false negative. The case when harmful traffic could be allowed to pass without any alerts or any proper actions to prevent it is the worst-case scenario which is the false negatives. Thus, we will examine and investigate which Machine Learning models will produce the minimum number of false negatives.

This paper focuses on applying machine learning techniques to anomaly-based intrusion detection systems. The study utilizes five machine learning classifiers applied to NSL-KDD dataset. Even though NSL-KDD dataset is not a perfect representation of present-day network traffic, it is used in this research due to the lack of publicly available datasets [3].

This paper is organized as follows: section II presents related work of analyzing intrusion detection datasets, section III describes the contents of NSL-KDD dataset, section IV presents experimental results and analysis of the five selected classification techniques. Finally, section V offers the conclusion and future work.

II. RELATED WORK

Laheeb *et al.* studied a comparison for intrusion detection dataset KDD99 and NSL-KDD based on Self Organization Map (SOM) artificial neural network [1]. They used an unsupervised artificial neural network in hierarchical anomaly intrusion detection system. SOM neural nets employed for detection and separation of normal traffic from the attacking traffic. The paper had also evaluated the efficiency of SOM in anomaly intrusion detection.

Shilpa *et al.* researched on feature reduction using principal component analysis of applicable anomaly–based Intrusion Detection on NSL-KDD dataset [2]. They reduced

the number of features in NSL-KDD dataset that were irrelevant and redundant for the anomaly detection process. They applied hybrid principal component analysis neural network algorithm to effectively detect attacks by reducing computer resource utilization.

S. Revathi *et al.* analyzed NSL-KDD dataset using various machine learning techniques for the intrusion detection system. The analysis focused on selected NSL-KDD datasets to analyze different machine learning techniques [3]. The Random Forest classification algorithm had the highest accuracy rate as demonstrated in their experimental result compared to other classification algorithms.

L.Dhanabal *et al.* studied NSL-KDD dataset for intrusion detection system by applying different classification algorithms [4]. The paper analyzed and used NSL-KDD dataset to study the effectiveness of various classification algorithms in detecting anomaly network traffic patterns. They examined the relationship of the protocols in the network protocol stack with the attacking traffic to generate anomalous network traffic.

Hee-su *et al.* examined feature selection for intrusion detection using NSL-KDD dataset [5]. They identified important selected input features in building IDS with computationally effective and efficient manner. The performance of standard feature selection methods were evaluated and the authors proposed a new feature selection method.

Rowayda *et al.* investigated effective anomaly Intrusion Detection System based on a new hybrid algorithm named neural network with Indicator Variable and Rough Set for attribute Reduction (NNIV-RS) [6]. The experimental results showed that the proposed NNIV-RS algorithm had a better and robust representation of data and was able to reduce unnecessary features to improve the reliability and efficiency of IDS.

Bhupendra *et al.* analyzed the performance of NSL-KDD dataset using an artificial neural network (ANN). The result obtained was based on various performance measures for both binary class as well as five class classification on attacking types. The accuracy of ANN was presented [7].

Ray investigated the effects of neural network architecture on the performance of intrusion detection systems (IDSs) [8]. An equation was formed to find the optimal number of hidden neurons in a multi-layer feed forward neural network (MLFFNN) IDS. This equation can be used to determine the number of hidden neurons to eliminate the lengthy trial and error calculations in case of MLFFNN.

III. DATASET DESCRIPTION

The selection and application of the dataset profoundly affect the performance of the algorithm. The used NSL-KDD dataset is solving the inherent problems of the KDDCUP'99 dataset [5]. NSL-KDD dataset has removed redundant records in the training and test dataset to enable classifiers to produce an unbiased result [3].

This paper uses the training and test dataset that is made up of two target values, normal and anomaly. The known attack types are grouped as anomaly traffic while the remaining network traffic were categorized as normal traffic. The original NSL-KDD dataset had 41 features and a label. NSL preprocessing step is conducted as KDD dataset has three features of object values that should be changed to numeric format before applying classifiers. The three features are as follows: 'protocol_type' has three unique categories, 'service' has 70 unique categories and 'flag' has 11 unique categories.

After applying one-hot encoding technique to the dataset, the number of features reaches 122 and a label, which is applied to each instance. The total instances in the dataset are 125,973 that are split into training and test dataset. The training dataset has 100,778, and the test dataset has 25,195 instances. Figure 1 and Figure 2 depict the number of normal and anomaly instance counts in the training and test datasets.

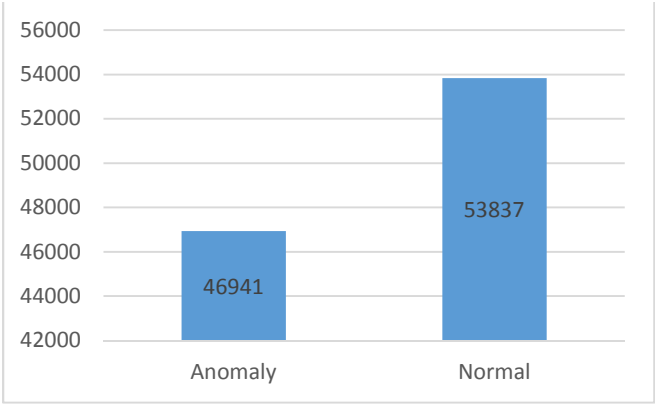

Figure 1. Training dataset target counts

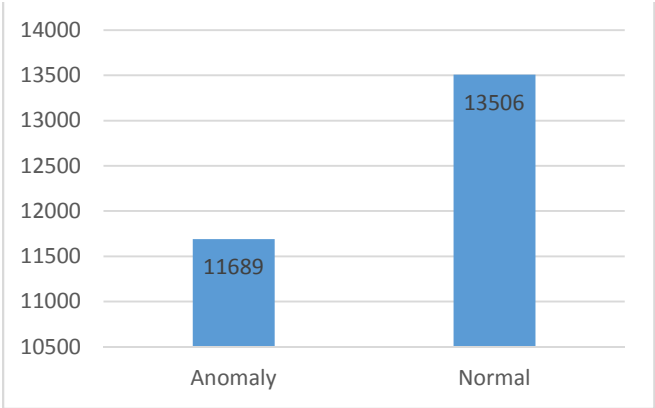

Figure 2. Test dataset target counts

On both datasets, the number of anomaly records is lower than the normal. Figure 1 and Figure 2 demonstrate the consistency and fair distribution of instances in the training and test datasets.

IV. EXPERIMENTAL RESULT AND ANALYSIS

This paper applies five techniques of classification and analyzes the NSL-KDD dataset in different ways. Distinct performance measures are calculated and compared: Precision, Recall, F_1 score, Receiver Operating Characteristic (ROC) curve, and Accuracy. The Precision is calculated by dividing the number of true positive (TP) instances over the sum of the number of true positive and false positive (FP) instances. The Recall is calculated by dividing the number of true positive instances over the sum of the number of true positive and false negative (FN) instances. The equation for calculating F_1 score is as follows [10]:

F_1 score= 2*TP/(2*TP+FN+FP)

The classifier can get a high F_1 score only if both recall and precision are high. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR). The Accuracy is calculated by dividing the sum of the number of true positive and true negative instances over the total number of instances.

Stochastic Gradient Descent (SGD) also known as incremental gradient descent classifier has advantages of handling extensive dataset and dealing with training instances independently. The classifier demonstrated poor performance initially because the features have large gaps between the minimum and maximum values. However, by applying standard feature scaling, the problem was solved. Figure 3 shows the ROC curve for SGD technique.

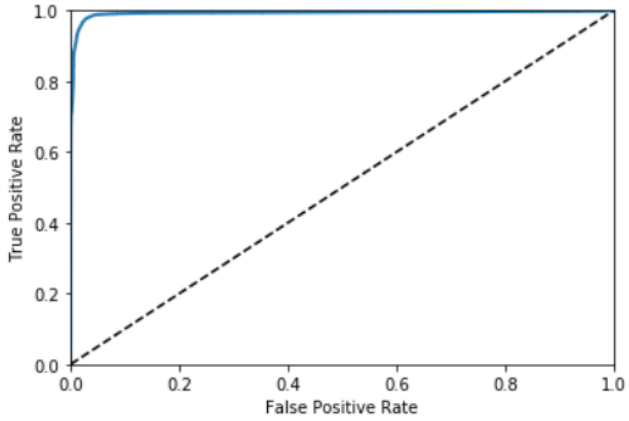

Figure 3. SGD ROC curve

Random Forests classifier works by training many Decision Trees on random subsets of the features, then averaging out their predictions [10]. Random Forests classifier demonstrate outstanding performance. The accuracy level achieved in Random Forests is near to perfection. Figure 4 depicts the ROC curve of Random Forests classifier.

Logistic Regression is one of the regression algorithms that can also be used for classification. Logistic Regression also called Logit Regression is used to estimate the probability that the instance belongs to a specific class [10]. This classifier has a lower performance compared to the other classifiers that are applied to NSL-KDD dataset. Figure 5 shows the ROC curve of Logistic Regression.

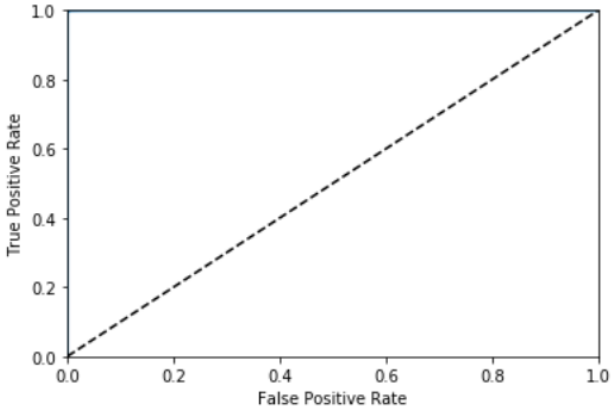

Figure 4. Random Forest ROC curve.

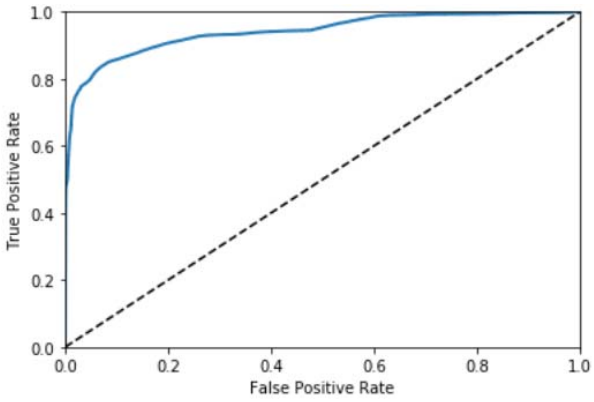

Figure 5. Logistic Regression ROC curve.

A Support Vector Machine (SVM) is a powerful and versatile Machine Learning model. It is capable of performing linear or nonlinear classification and regression. SVM is well suited for classification of complex but small or medium-sized datasets [10]. The result of applying SVM classifier in NSL-KDD dataset demonstrated reasonable performance and is comparable to the result obtained in case of Random Forests classifier. Figure 6 shows the ROC curve of Support Vector Machine.

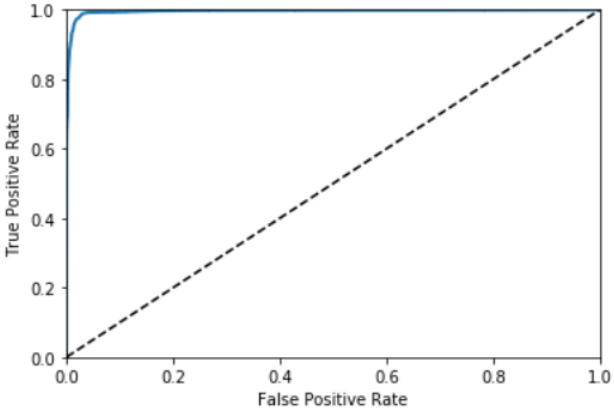

Figure 6. SVM ROC curve.

Keras is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It is recommended to use Keras because of its easy and fast prototyping of deep learning libraries, user-friendliness, modularity, and extensibility. It supports both convolutional neural networks and recurrent networks as well as combinations of the two. It also runs seamlessly on CPU and GPU. The core data structure of Keras is a model, and the simplest type of model is the Sequential model; a linear stack of layers. The input for the model is specified. Before training, the learning method needs to be configured, which is done via the compile method. However, in the conducted experiments Random Forests and SVM outperformed the Sequential model in Keras as shown in Figure 7.

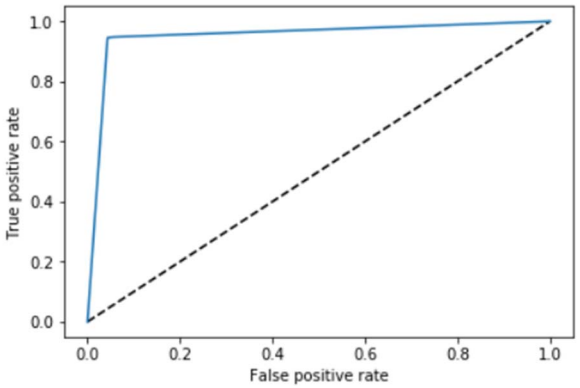

Figure 7. The ROC curve of Sequential Model in Keras

Table 1 includes a summary of precision, recall, and F_1 score. Figure 7 shows the results of accuracy for the five classifiers.

Score Type	SGD	Logistic Regression	Random Forests	SVM	Sequential Model
Precision	0.9696	0.8967	0.9992	0.9779	0.9881
Recall	0.9742	0.8507	0.9969	0.9730	0.924
F_1 score	0.9719	0.8731	0.9980	0.9755	0.95497

Table 1. Score Summary

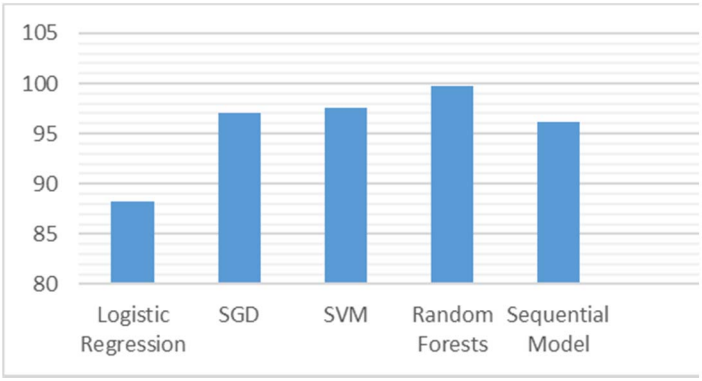

Figure 8. Accuracy results of the five classifiers

A comparison of the training time in seconds among these five different classifiers was done. Figures 9 and 10 present this comparison. As demonstrated, both Stochastic Gradient Descent (SGD) classifier and Random Forests classifier require less time during training compared to the other three classifiers.

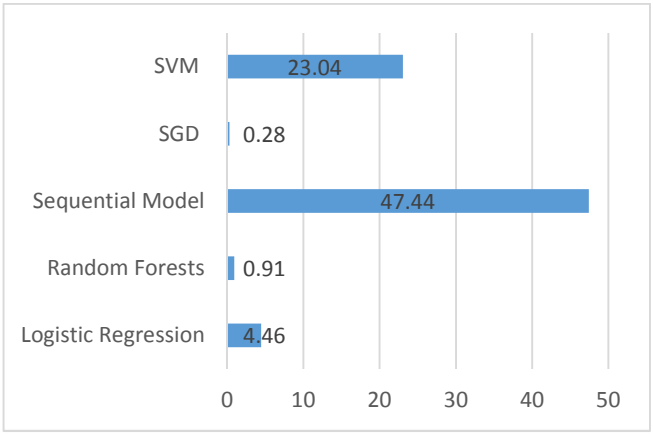

Figure 9. A comparison of the training time in seconds

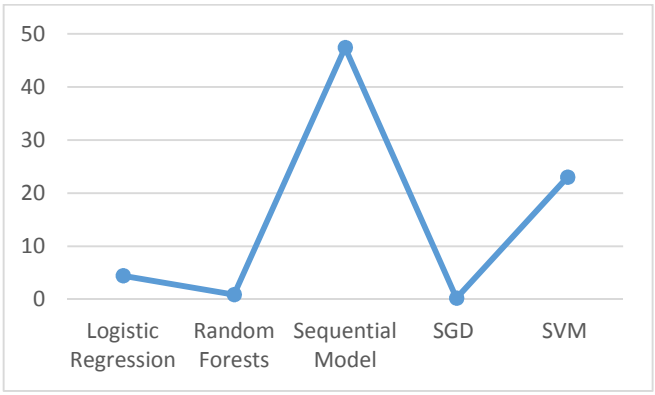

Figure 10. A comparison of the training time in seconds

V. CONCLUSION AND FUTURE WORK

The comparison of five machine learning models was conducted by applying them to NSL-KDD dataset. The research had been carried out with five different classification algorithms with and without one-hot encoding. It is evident that the Random Forests algorithm outperformed the other four classifiers. The overall results of Random Forests classifier are near to perfection. In particular, the highest value of recall is obtained in Random Forests model which means that the minimum number of false negatives are achieved. In our future work, we plan to integrate and analyze several artificial neural networks to classify different class types or attacking techniques in the available intrusion detection system datasets.

VI. REFERENCES

[1] Laheeb M. Ibrahim, Dujan T. Basheer and Mahmod S. Mahmod, "A Comparison Study for Intrusion Database (KDD99, NSL-KDD) Based on Self Organization Map (SOM) Artificial Neural Network," Journal of Engineering Science and Technology, Vol. 8, No. 1, pp. 107 – 119, 2013

https://core.ac.uk/download/pdf/25739889.pdf

[2] Shilpa Lakhina, Sini Joseph and Bhupendra Verma, "Feature Reduction using Principal Component Analysis for Effective Anomaly-Based Intrusion Detection on NSL-KDD," *International Journal of Engineering Science and Technology*, Vol. 2(6), pp. 1790-1799, 2010 http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=09445114AAB380FC3577CEFE20A6BD94?doi=10.1.1.168.1957&rep=rep1&type=pdf

[3] S. Revathi and Dr. A. Malathi, "A Detailed Analysis on NSL-KDD Dataset Using Various Machine Learning Techniques for Intrusion Detection," International Journal of Engineering Research & Technology (IJERT), Vol. 2 Issue 12, ISSN: 2278-0181, December – 2013

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.680.6760&rep=rep1&type=pdf

[4] L.Dhanabal and Dr. S.P. Shantharajah, "A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms," International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 6, June 2015

https://pdfs.semanticscholar.org/1b34/80021c4ab0f632efa99e01a9b073903c5554.pdf

[5] Hee-su Chae, Byung-oh Jo, Sang-Hyun Choi, and Twae-kyung Park, "Feature Selection for Intrusion Detection using NSL-KDD," Recent Advances in Computer Science, ISBN: 978-960-474-354-4

http://www.wseas.us/e-library/conferences/2013/Nanjing/ACCIS/ACCIS-30.pdf

[6] Rowayda A. Sadek, M. Sami Soliman and Hagar S. Elsayed, "Effective Anomaly Intrusion Detection System based on Neural Network with Indicator Variable and Rough set Reduction," IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 6, No 2, November 2013, ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784

https://pdfs.semanticscholar.org/5293/08f1120942793939dfe2b146fdc151cabb66.pdf

[7] Bhupendra Ingre and Anamika Yadav, "Performance Analysis of NSL-KDD dataset using ANN," Signal Processing and Communication Engineering Systems (SPACES), 2015 International Conference, 2015, Page(s):92- 96

[8] L. Ray, "Determining the Number of Hidden Neurons in a Multi-Layer Feed Forward Neural Network," *Journal of Information Security Research,* vol. 4, no. 2, pp. 63-70, 2013.

[9] Dataset source: https://github.com/jmnwong/NSL-KDD-Dataset

[10] Book: "Hands-On Machine Learning with Scikit-Learn and TensorFlow"