Empirical Analysis of Machine Learning Techniques for Intrusion Detection in Network System

1st Himanshu Singh
Institute of Technology
Nirma University
Ahmedabad, India
21mcei02@nirmauni.ac.in

2nd Mahima Bansal
Institute of Technology
Nirma University
Ahmedabad, India
21mcei15@nirmauni.ac.in

Abstract—Increasing network resource usage creates security risks with it. Malwares and other sources may disrupt the system operations and inadequate security holes in systems. Intrusion Detection System(IDS) is invented to alert admins in case of such security breaches. In order to enhance IDS systems, artificial intelligence as well as . In this research, literature studies employing CSE-CIC IDS-2018, UNSW-NB15, ISCX-2012, NSLKDD and CIDDS-001 data sets, frequently used to design IDS systems, updated in detail. In addition, max-min normalisation was done on these data sets and classed created utilising ,K NN algo, vector support machine (SVM), Decision Tree (DT) algorithms, which among the most ancient ML approaches. As a result, some genuinely good results have been analyzed. *Index Terms*—Intrusion, Machine Learning, Security

I. Introduction

The study of intrusion detection is generally centred on misuse or anomaly detection, with abuse typically being chosen in commercial applications. The identification of anomalies is fully reliant on theoretical approaches for coping with novel hazards.

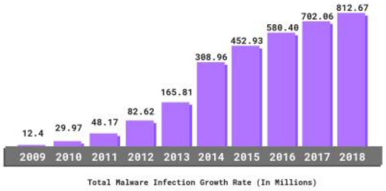

Fig. 1. Malware infection growth rate (in millions/year) Image-Source: https://purplesec.us/

The research quest for anomaly detection is totally based on many machine learning algorithms applied to a range of training and testing datasets [2].

The research quest for anomaly detection is totally based on many machine learning algorithms applied to a range of training and testing datasets [2]. This research is focused on the inherent issues in the KDDcup 99 dataset as well as the cure as a study of the NSL-KDD dataset for measuring accuracy in intrusion detection. The first evident problem in the KDD [3] data set is the significant quantity of duplicated records, with about 78 percent and 75 percent repeated in the train and test sets, respectively.

As a result, the learning algorithm becomes biassed, making User to Root (U2R) more detrimental to the network. The rest of the paper is organised as follows: Section II presents the crux of the state of the art of intrusion detection and attack prediction and motivation.Section III talks about the Literature Survey as it plays a major role in understanding the research done in past decade. Section IV talks about the dataset used for Intrustion Detection. It provides a full explanation of the assaults found in the NSL-KDD dataset. Section V represents a detailed summary of the examination of the NSL KDD dataset using various data mining techniques. Section VI describes the experimental analysis of various assaults conducted using various machine learning approaches. Section VII summarises the findings and future work.

II. MOTIVATION

With an ever-increasing number of vulnerabilities and attack vectors, no firewall and no network are immune. As a result, many attackers use extra malware or social engineering to get user credentials that provide them access to your community and data. A network intrusion detection system (NIDS) is critical for community security since it detects and responds to malicious activities. An intrusion detection system's key benefit is that it warns IT professionals when an assault or community infiltration is identified. A network intrusion detection system (NIDS) monitors both incoming and outgoing network traffic, as well as data that travels between community machines. When suspicious behaviour or possible threats are spotted, the network intrusion detection system (IDS) analyses network visits and generates notifications, enabling IT workers to conduct similar investigations and take the required actions to avert or neutralise an attack. We study the behaviour of multiple device mastery approaches on NIDS in this research, which may assist in the construction of more robust NIDS.

III. LITERATURE SURVEY

There are distinctive studies papers which had been developed for implementing techniques for Intrusion Detection in Network System. Literature survey is provided starting with the paper [1] which uses Imagenet Dataset, Deep Neural Network accelerator structure Algorithm and parameters used are controlled glitch injection into the clock sign of the DNN accelerator. Then coming to the paper of [2] this paper provided convolutional neural network Algorithm on Imagenet Dataset using a thousand pix randomly selected. Paper[3] defines Multiple Kernel Learning (MKL) and Unsupervised getting to know - Naive-Bayes, Decision Tree, KN, LSVM, Random Forest, LSTM Algorithms on CICIDS2017, AMI information, Attack vector - AMI visitors, MATLAB simumation primarily based facts, Private network visitors records,
KDD CUP, DARPA 1999, DARPA 2000, NS2 Simulation facts, Online OmNET++ simulation records,1999 DARPA IDS dataset via MIT Lincoln lab Datasets using Features from multilevel automobile-encoders are combined the usage of Multiple Kernel Learning (MKL) parametrs.The writer of Paper[4] defines Deep Learning and CNN Algorithm the usage of Data generated from electricity meter. Next is paper [5] This study paper employs DARPA Intrusion Detection, ADFA records set, Data Sets,NSL-KDD dataset, KDD Cup 99 dataset, and algorithms such as Decision Trees , Support Vector Machine, and Neuural Networks. The dataset ”CTI records and Microsoft Malware Prediction dataset” is defined in paper [6] Based on Support Vector Machine (SVM),Logistic Regression (LG), Random Forest (RF), and Decision Tree (DT), with attack and TTP as input parameters and vulnerabilities and compromise indicators as output parameters. Then coming onto studies work stated in paper [7], precise type of cyber hazard data is the time collection of cyber attacks discovered via a cyber defense device referred to as honeypots, which passively display the incoming Internet connections, alongwith algorithms Bayesian method, hidden Markov model, seasonal ARIMA Model, FARIMA version , FARIMA+GARCH model, marked factor method, vine copula model with parameters as degree of prediction accuracy, to forecast or are expecting Cyber assaults. As per the examine cited in paper [8] datasets used are ”Alert Dataset, ‘LLDDoS1.Zero DARPA’dataset”, and algorithms is Hidden Markov Model with parameters inclusive of IDS database, National Vulnerabilities Database (NVD), assault garph datasources, HMM parameters. Profiles, metrics,statistical models, and procedures for evaluating logs were all part of the concept [9]. In paper[10] The dataset includes 2,253 breach incidents from 2005 to 2015, as well as an Algorithm for Predicting the VaRa’s of Hacking Incidents Inter-Arrival Times and Breach Sizes. Separate and completed procedures involve doing qualitative and quantitative data collecting assessments. Paper [10] consists of IoT-23, the datasets utilised are LITNET-2020 and NetML-2020, and the methods used are ”Deep Neural Network (DNN) and Long ShortTerm Memory (LSTM) and a meta-classifier (i.e., logistic regression) based on the notion of stacking generalisation.” in conjunction with parameters Statistical significance was examined, and the results were compared to state-of-the-art procedures in community anomaly identification. In paper [11] turbofan engine degradation simulation statistics set, experimental transportation statistics set is used alongwith algorithms as ”Splice Heuristic, k-nearest neighbor-based algorithms, known as Bonsai and ProtoNN” and parameters exceeded are Matrix of Size M x T. In paper [13],Markov time-varying fashions are used, as well as deep variants of Bolzmann machines and Hidden Markov with parameters as follows: This model has great accuracy, practicability, intellectuality, objectivity, and realtime in the field of network chance prediction. Then paper [14] consists of dataset from the original text at the OpenNRE, Wiki80 is based on Tsinghua’s data collection FewRel and includes datasets from the original text at the OpenNRE. This data set has 80 different kinds of connections, with 700 different variations for each link, for a total of 50000 thousand and above samples. There are 56000 crucial and train collectively, and the set of rules performed is a distant-supervised set of rules, and this study also adopts remote-supervised relation extraction as a starting point, solving the issue of artificial statistics annotation. ” We chose the NSL-KDD dataset because it does not contain duplicate statistics in the teach set, thus the classifiers are no longer biassed toward more prevalent statistics. There is no duplicate data in the suggested test sets; as a result, the overall performance of the newbies isn’t influenced by approaches that have superior detection rates at common records; due of these features of the NSL-KDD Dataset, we’ve prioritised it in our research work.

IV. DATASET DESCRIPTION

The statistical analysis found that the data set has severe errors that have a rightful impact on system performance and model performance and result in a very unsatisfactory rating of anomaly detection approaches. NSL-KDD was introduced to overcome the given challanges in KDD99 [6] is proposed, which consists of selected records from the whole KDD data set.

The advantage of NSL KDD dataset are:

- No biased result as there is no redundant record.
- Better reduction rates as there is not dupicate record.

Comparing the training data and testing data, training data has 21 attack and testing data has 37 attacks. Common attacks are listed in training set while Novel attacks are present in testing set.

V. Data Mining on Dataset

Data is vital to machine learning. It's the most important factor that enables algorithm training and explains why machine learning has gained such traction in recent years. However, regardless of how many terabytes of data you have or how knowledgeable you are in data science, if you can't make sense of data records, a machine will be almost worthless, if not dangerous.

A. Data-Preprocessing

1) One-Hot-Encoding: To convert all category attributes to binary properties, One-Hot-Encoding is employed. In order to meet the criteria of one-hot-endcoding, the input to this transformer must be an integer matrix describing the values received with categorical(discrete) attributes. The result will be a sparse matrix with each column representing a potential value. It is expected that input properties have values in the range [0, n values]. As a result, before converting each category to a number, the properties must be translated using the LabelEncoder.

2) LabelEncoder: It inserts categorical features into a 2D numpy array and transforms them into numbers. Then,

- Missing columns in the test set are added.
- New numeric columnns are added to the main dataframe.
- Zero=Normal, One=DoS, Two=Probe, Three=R2L, Four=U2R. In new dataframes, the label column was replaced with new values.

3) Feature Scaling: Feature scaling is a method for standardising the independent properties of data within a certain range. It is used during data pre-processing to cope with widely varying values, magnitudes, or units. Without feature scaling, a machine learning model would take a large overhead and may cause overfitting or underfitting.

4) Feature Selection: Feature selection is the process of reducing the amount of input variables while developing a predictive model. Limiting the number of input variables is recommended in order to decrease modelling computational costs and, in certain cases, improve model performance. Recursive Feature Elemination: RFE is a feature selection approach that elminiates features by training on the model and deleting the low outcome features. 13 features are selected as a group after elimination.

5) Build the model: Classifier is trained for reduced features, for later comparison.

6) Prediction & Evaluation (validation): : Using reduced Features for each category

VI. Experimental Analysis and Results

All the results after performing Data Mining as well as running the classification algorithms are shown below.

protocol type	service	flag
tcp	ftp data	SF
udp	other	SF
tcp	private	S0
tcp	http	SF
tcp	http	SF

TABLE I
Flag Values

Transform categorical features into numbers using LabelEncoder()

Train:
Shape (rxc) of R2L: (68338, 123)
Shape (rxc) of U2R: (67395, 123)
Shape (rxc) of DoS: (113270, 123)
Shape (rxc) of Probe: (78999, 123)

Test:
Shape (rxc) of R2L: (12596, 123)

	protocol type	service	flag
1	tcp	ftp data	SF
2	udp	other	SF
3	tcp	private	S0
4	tcp	http	SF
5	tcp	http	SF

	protocol type	service	flag
0	1	20	9
1	2	44	9
2	1	49	5
3	1	24	9
4	1	24	9

TABLE II
Categorical Features Changed into Numeric Features

Shape (rxc) of U2R: (9778, 123)
Shape (rxc) of DoS: (17171, 123)
Shape (rxc) of Probe: (12132, 123)

Summary of features selected by RFE:
DOS Shape: (113270, 13)
Probe shape: (78999, 13)
R2L Shape: (68338, 13)
U2R Shape: (67395, 13)

Detailed Results:
Using 13 Features for each category: Confusion Matrices

1) Random Forest :

: 1. DoS

Predicted Attack Actual Attack	0	1
0	9591	120
1	6407	1053

Accuracy: 0.99691 (+/- 0.00195)
Precision: 0.99705 (+/- 0.00234)
Recall: 0.99692 (+/- 0.00399)
F-measure: 0.99651 (+/- 0.00274)

2. Probe

Predicted Attack Actual Attack	0	2
0	9270	441
2	998	1423

Accuracy: 0.99481 (+/- 0.00437) Precision: 0.99241 (+/- 0.00982)
Recall: 0.98823 (+/- 0.01055)
F-measure: 0.99028 (+/- 0.00687)

3. R2L

Predicted Attack Actual Attack	0	3
0	9711	000
3	2885	000

Accuracy: 0.97904 (+/- 0.00682)
Precision: 0.97201 (+/- 0.01363)
Recall: 0.96739 (+/- 0.01148)
F-measure: 0.97177 (+/- 0.00914)

4. U2R

Predicted Attack Actual Attack	0	4
0	9710	1
4	66	1

Accuracy: 0.99693 (+/- 0.00330)
Precision: 0.96793 (+/- 0.09547)
Recall: 0.83908 (+/- 0.18726)
F-measure: 0.87069 (+/- 0.08335)

2) KNeighbors

: 1. DoS

Predicted Attack Actual Attack	0	1
0	9422	287
1	1573	5887

Accuracy: 0.99715 (+/- 0.00278)
Precision: 0.99678 (+/- 0.00383)
Recall: 0.99665 (+/- 0.00344)
F-measure: 0.99672 (+/- 0.00320)

2. Probe

Predicted Attack Actual Attack	0	2
0	9437	274
2	1272	1149

Accuracy: 0.99077 (+/- 0.00403)
Precision: 0.98606 (+/- 0.00675)
Recall: 0.98508 (+/- 0.01137)
F-measure: 0.98553 (+/- 0.00645)

3. R2L

Predicted Attack Actual Attack	0	3
0	9706	5
3	2883	2

Accuracy: 0.96705 (+/- 0.00752)
Precision: 0.95265 (+/- 0.01248)
Recall: 0.95439 (+/- 0.01401)
F-measure: 0.95344 (+/- 0.01070)

4. U2R

Predicted Attack Actual Attack	0	4
0	9711	0
4	65	2

Accuracy: 0.99703 (+/- 0.00281)
Precision: 0.93143 (+/- 0.14679)
Recall: 0.85073 (+/- 0.17639)
F-measure: 0.87831 (+/- 0.11390)

3) SVM

: 1. DoS

Predicted Attack Actual Attack	0	3
0	9639	72
3	2737	148

Predicted Attack Actual Attack	0	1
0	9455	256
1	1359	6101

Accuracy: 0.96793 (+/- 0.00738)
Precision: 0.94854 (+/- 0.00994)
Recall: 0.96264 (+/- 0.01388)
F-measure: 0.95529 (+/- 0.01048)

4. U2R

Predicted Attack Actual Attack	0	4
0	9710	1
4	67	0

Accuracy: 0.99371 (+/- 0.00375)
Precision: 0.99107 (+/- 0.00785)
Recall: 0.99450 (+/- 0.00388)
F-measure: 0.99278 (+/- 0.00428)

2. Probe

Predicted Attack Actual Attack	0	2
0	9576	135
2	1285	1136

Accuracy: 0.98450 (+/- 0.00526) Precision: 0.96907 (+/- 0.01031)
Recall: 0.98365 (+/- 0.00686)
F-measure: 0.97613 (+/- 0.00800)

3. R2L

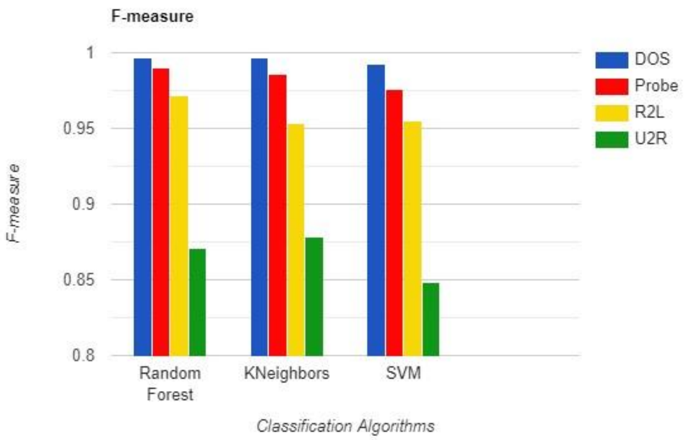

Fig. 3. Comparative analysis of algorithms on basis of F-measure

Accuracy: 0.99632 (+/- 0.00390)
Precision: 0.91056 (+/- 0.17934)
Recall: 0.82909 (+/- 0.21833)
F-measure: 0.84869 (+/- 0.16029)

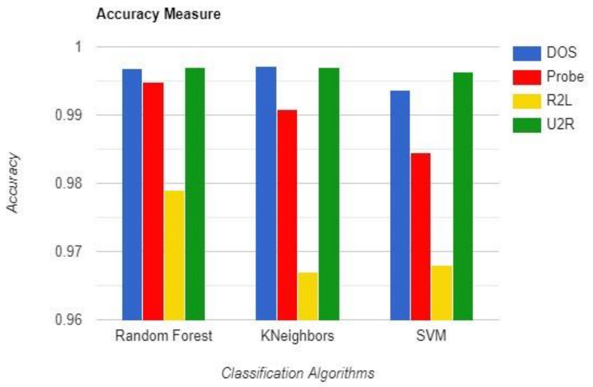

Fig. 2. Comparative analysis of algorithms on basis of accuracy

VII. Conclusion and Areas of Improvement

We investigated the NSL-KDD dataset, which overcomes some of the inadequacies of the parent dataset (KDD99), in this paper. The given dataset, according to the results, is of adequate quality to assess different intrusion detection systems. Investigating invasive patterns using all 41 characteristics in the dataset may take a long time and significantly impair device performance.

Some of the qualities in the dataset are redundant and unnecessary to the procedure. RFE alorithms reduce the dimensionality of a dataset. The experiment was carried out using multiple class strategies for the dataset without function discount, and it is clear that Random Forest beats all other algorithms. So, for a smaller dataframe, our study indicates that Random Forest is quite faster than others for intrusion and anomaly detection, which is crucial for latency based applications, while also offering the maximum testing accuracy. In the destiny, we are able to attempt to beautify the Random Forest algorithm as a way to create a greater effective intrusion detection gadget.

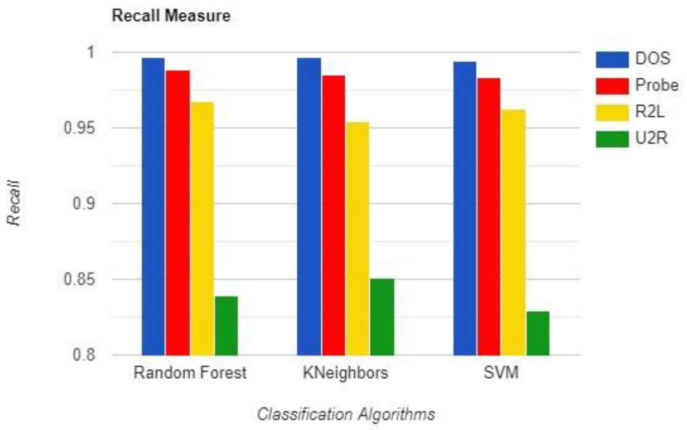

Fig. 4. Comparative analysis of algorithms on basis of Recall Measure

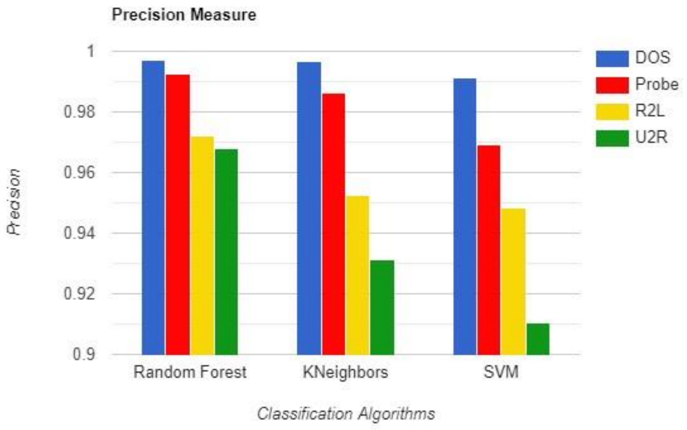

Fig. 5. Comparative analysis of algorithms on basis of Precision Measure

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, “Title of paper if known,” unpublished.

[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[8] Kalnoor, G., Gowrishankar, S. A model for intrusion detection system using hidden Markov and variational Bayesian model for IoT based wireless sensor network. Int. j. inf. tecnol. (2021).
https://doi.org/10.1007/s41870-021-00748-1

[9] James P. Anderson, “Computer security threat monitoring and surveillance,” James P. Anderson Co., Fort Washington, Pennsylvania, technical Report, April 1980.

[10] Xu, Maochao Schweitzer, Kristin Bateman, Raymond Xu, Shouhuai. (2018). Modeling and Predicting Cyber Hacking Breaches. IEEE Transactions on Information Forensics and Security. PP. 1-1. 10.1109/TIFS.2018.2834227.

[11] V. Dutta, M. Choras, M. Pawlicki, and R. Kozik, “A Deep Learning´ Ensemble for Network Anomaly and Cyber-Attack Detection,” Sensors, vol. 20, no. 16, p. 4583, Aug. 2020, doi: 10.3390/s20164583

[12] Qolomany, Basheer Mohammed, Ihab Al-Fuqaha, Ala Guizani, Mohsen Qadir, Junaid. (2020). Trust-Based Cloud Machine Learning Model Selection For Industrial IoT and Smart City Services. IEEE Internet of Things Journal. PP. 1-1. 10.1109/JIOT.2020.3022323.

[13] Kuremoto, Takashi Kimura, Shinsuke Kobayashi, Kunikazu Obayashi, Masanao. (2014). Time series forecasting using a deep belief network with restricted Boltzmann machines. Neurocomputing. 137. 47–56. 10.1016/j.neucom.2013.03.047.

[14] Han, Xu Gao, Tianyu Yao, Yuan Ye, Deming Liu, Zhiyuan Sun, Maosong. (2019). OpenNRE: An Open and Extensible Toolkit for Neural Relation Extraction. 169-174. 10.18653/v1/D19-3029.