iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** IX **Month of publication:** September 2021

DOI: https://doi.org/10.22214/ijraset.2021.37955

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

Anomaly based Intrusion Detection System using Machine Learning

Akshat Runwal
Department of Information Technology, Shri Govindram Seksaria Institute of Technology and Science, Indore, India

Abstract: ***Attacks on the computer infrastructures are becoming an increasingly serious issue. The problem is ubiquitous and we need a reliable system to prevent it. An anomaly detection-based network intrusion detection system is vital to any security framework within a computer network. The existing Intrusion detection system have a high detection rate but they also have mendacious alert rates. With the use of Machine Learning, we can implement an efficient and reliable model for Intrusion detection and stop some of the hazardous attacks in the network. This paper focuses on detailed study on NSL- KDD dataset after extracting some of the relevant records and then several experiments have been performed and evaluated to assess various machine learning classifiers based on dataset. The implemented experiments demonstrated that the Random forest classifier has achieved the highest average accuracy and has outperformed the other models in various evaluations.***
Keywords: Intrusion Detection System, Anomaly Detection, Machine Learning, Random Forest, Network Security

I. INTRODUCTION

Security is becoming a paramount issue as the Internet applications are burgeoning. Wireless networks are very weak and hence it faces many security problems. The system security can be enhanced by Intrusion detection. They are used to detect anomalies with the aim of catching hackers before they do real damage to a network .An intrusion detection system can be used for detecting both network and computer intrusions and misuse by monitoring system activity and classifying it as either normal or anomalous[1]. Anomaly based NIDS operate based on the idea that the ambient traffic in a network collected over a period of time reflects the nature of the traffic that may be expected in the immediate future. Anomaly intrusion detection identifies deviations from the normal usage behaviour patterns to identify the intrusion. Anomaly detection requires storage of normal usage behaviour and operates upon audit data generated by the operating system. It requires storage of normal usage behavior and operates upon audit data generated by the operating system.

A. Generally, Attacks Fall Into Four Main Groups

1) *DOS:* Denial-of-service
2) *R2L:* Unauthorized access from a remote machine
3) *U2R:* Unauthorized access to local root privileges
4) *Probing:* Surveillance and another probing

Though the existing intrusion detection systems have a high detection rate, but they also have high false alert rate i.e. providing an alert even on a normal connection. Reducing false positive alert is very important for the intrusion detection system.
Now-a-days, Machine learning techniques are heavily being adapted and developed in intrusion detection to enhance the efficacy of the systems. Machine learning methodologies are being widely used by the researchers in the field of network intrusion detection due to their generalization capabilities that helps to understand the technical knowledge about the intrusions that do not have any predefined patterns.[2]
Regarding the attack detection, it is considered a classification problem because the target is to clarify whether the packet is either normal or an attack packet. Therefore, the model of accepted intrusion detection systems can be implemented based on machine learning algorithms. The research analysis for anomaly detection is fully based on several machine learning methods on various training and testing dataset. Our study analyses NSL-KDD dataset for finding accuracy in intrusion detection. The following machine learning algorithms have been implemented: Logistic Regression, Gaussian Naive Bayes, Decision Tree, Random Forest, Support Vector Machine to evaluate and accurate the model of intrusion detection system in this paper.
The rest of the paper is structured as follows: section II present description of NSL-KDD dataset. Section III summarize in detail about analysis of various Machine Learning techniques and implementation. Section IV explain the experimental analyses on various attacks using different machine learning techniques. The conclusion and future work is summarized in section V.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

II. NSL-KDD DATASET

The data-set to be used in this research is the NSL-KDD dataset [3] which is a new dataset which is the benchmark of researches in network intrusion detection system. It is also pointed out that the NSL-KDD dataset contains selected records of the complete KDDCup 1999 dataset and does not suffer from any of the inherent problems. NSL-KDD dataset solve the issues of KDD 99 benchmark and connection record contains 42 features. Each of the 42 feature is not needed to implement an intrusion detection system as there are some redundant features which doesn't affect the targe variable much. In order to conduct a thorough analysis of the proposed research, we apply five machine learning algorithms with various evaluation methods, to build a network intrusion detection system.The main contribution of this dataset is the Content features within a connection suggested by domain knowledge which help to understand the behaviour of different types of attacks, the essential characteristics to detect DOS, PROBE, R2L and U2R given in Table 2.

Table I
Basic Features of TCP Connection

Feature name	Description	Type
duration	Length (number of seconds) of the connection	Continuous
Protocol_type	Type of the protocol, e.g. tcp, udp, etc.	Discrete
service	Network service on the destination, e.g., http, telnet, etc.	Discrete
src_bytes	Number of data bytes from source to destination	Continuous
dst_bytes	Number of data bytes from destination to source	Continuous
flag	Normal or error status of the connection	Discrete
land	1 if connection is from/to the same host/port; 0 otherwise	Discrete
wrong_fragment	number of "wrong" fragments	Continuous
urgent	Number of urgent packets	Continuous

TABLE II
Content features within a connection suggested by domain knowledge

Feature name	Description	Type
hot	Number of "hot" indicators	Continuous
num_failed_logins	Number of failed login attempts	Continuous
logged_in	1 if successfully logged in; 0 otherwise	Discrete
num_compromised	Number of "compromised" conditions	Continuous
root_shell	1 if root shell is obtained; 0 otherwise	Discrete
su_attempted	1 if "su root" command attempted; 0 otherwise	Discrete
num_root	Number of "root" accesses	Continuous
num_file_creations	Number of file creation operations	Continuous
num_shells	Number of shell prompts	Continuous
num_access_files	Number of operations on access control files	Continuous
num_outbound_cmds	Number of outbound commands in an ftp session	Continuous
is_hot_login	1 if the login belongs to the "hot" list; 0 otherwise	Discrete
is_guest_login	1 if the login is a "guest" login; 0 otherwise	Discrete

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue IX Sep 2021- Available at www.ijraset.com

III.MACHINE LEARNING MODELS AND IMPLEMENTATION

Machine Learning is the field of study that gives computers the capability to learn and improve from experience without being programmed explicitly automatically. Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs[4]. Under supervised learning approach the classification problem comes into account when the instances belong to two or more classes and our intention is to forecast the class of the unlabeled instances.

A. Logistic Regression

Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. Logistic regression uses the concept of predictive modelling as regression; therefore, it is called logistic regression, but is used to classify samples. Therefore, it falls under the classification algorithm. Logistic regression models the data using the sigmoid function.

B. Naive Bayes

Naive Bayes is based on Bayes theorem. This algorithm assumes that the occurrence of a certain feature is independent of the occurrence of other features. We create a frequency table for all the predictors against the classes and calculate the likelihood of all the predictors. Using the Naive Bayes equation, the posterior probability is calculated for all the classes. The outcome of the Naive Bayes Classifier will be the class with the highest probability amongst all the class probabilities[5]. The formula of Bayes theorem is given by

$$P(A|B)=\frac{P(B|A)P(A)}{P(B)}$$

Fig 1 Bayes Theorem

In Fig 1,
P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.
P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.
P(A) is Prior Probability: Probability of hypothesis before observing the evidence.
P(B) is Marginal Probability: Probability of Evidence.

C. Decision Tree

Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree.The goal of using a Decision Tree is to create a training model that can use to predict the class or value of the target variable by learning simple decision rules inferred from prior data[6]. One of the methods to select the attribute that will split the dataset is by calculating Entropy and Information Gain. Entropy captures the amount of impurity in the variable. The Information Gain is the entropy of the parent node minus the sum of entropies of the child nodes.

$$Entropy=\sum_{i=1}^{C}-p_i*\log_2(p_i) \qquad Gini=1-\sum_{i=1}^{C}(p_i)^2$$

Fig 2 Entropy and Gini Index Formula

In Fig 2,
c is number of classes.

D. Random Forest

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. The forest it builds, is an ensemble of decision trees, usually trained with the bagging method. The general idea of the bagging method is that a combination of learning models. To implement this algorithm the number of trees within the forest should be figured because each individual tree within a forest predicts the expected output and after that the voting technique used to select the expected output, that have the largest votes number.[7]

E. Support Vector Machine(SVM)

Basically, SVM finds a hyper-plane that creates a boundary between the types of data. It finds the optimal hyperplane by maximizing the margin distance between the observations of the classes using the Hinge loss function. Support vector machine works comparably well when there is an understandable margin of dissociation between classes. The loss function is give below

$$\ell(y) = \max(0, 1 + \max_{y \neq t} \mathbf{w}_y \mathbf{x} - \mathbf{w}_t \mathbf{x})$$

Fig 3 Loss Function

In Fig 3, t is Target variable, w is Model parameters and x is Input variable.

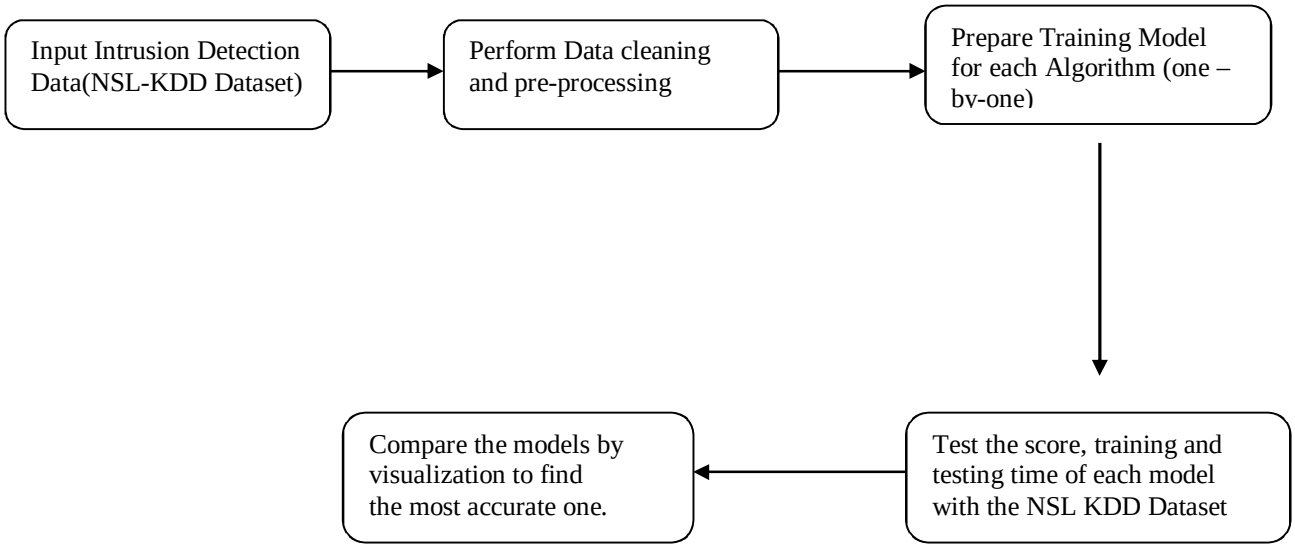


Fig 4 Implementation

In short, the steps of implementation are shown in Figure 4.

The output that is predicted by the model is the attack type whether it is in DoS, Probe, R2L or U2R category or Normal.

IV. EXPERIMENTAL RESULT AND ANALYSIS

The relevant features were extracted after analysing the correlation matrix i.e. seeing the relation of each variable with the other.

The dataset obtained after this was divided into training and testing set.67% of the data was used for training our model whereas 33% data was used for testing. The evaluation metrics used was confusion matrix, which correctly help us classify the true positives and helped in detecting the false alerts. The confusion matrix provided a thorough evaluation of the model.

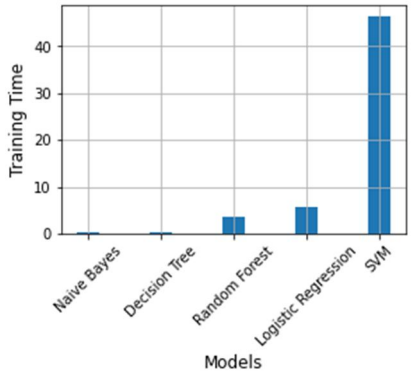


Fig 5. Training time of various models

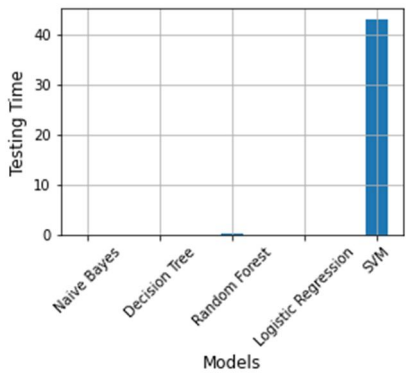


Fig 6. Testing time of various models

The highest training time as well as testing time was taken by Support Vector Machine (SVM) model. The least training time was taken by Naive Bayes. The testing time of Naive Bayes, Decision Tree and Logistic Regression were negligible. Random Forest took moderate time in training and testing as compared to the other models. The training and testing time of various models are depicted in Fig 5 and Fig 6 respectively.

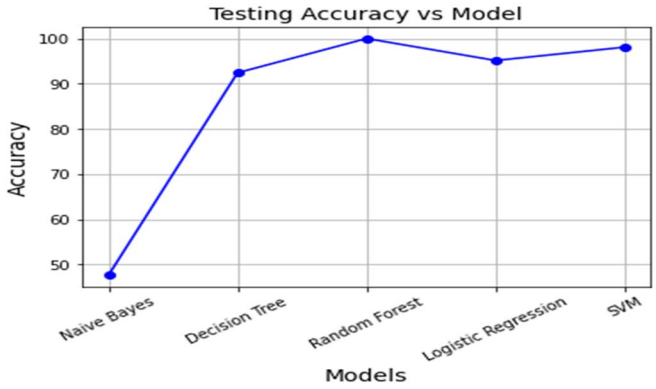


Fig 7. Accuracy on Test data

Fig 7 depicts the accuracy of various models on the testing dataset. Random Forest shows the highest test accuracy compared to all other algorithms with an accuracy of 99.97% on the testing data. Naive Bayes model should not be taken in consideration for Intrusion Detection System as the score obtained by the model on the test data set was only 47.63%. Some of the other models including Decision Tree, Logistic Regression and Support Vector Machine were able to perform well giving an accuracy greater than 90% but Random Forest outperformed them. We can though improve the accuracy of the models by hypertuning their various parameters.

TABLE III
AVERAGE ACCURACY RATE

ML models on various attacks	Correctly Classified	Incorrectly Classified	Accuracy
Naive Bayes			
Normal	1307	20792	5.91%
R2L	142	174	44.93%
Probe	3506	360	90.68%
DoS	14831	439	97.12%
U2R	16	4	80.0%
Decision Tree			
Normal	21096	1003	95.46%
R2L	0	316	0.0%
Probe	3209	3866	83.01%
DoS	14106	1164	92.37%
U2R	0	20	0.0%
Random Forest			
Normal	22094	5	99.97%
R2L	315	1	99.68%
Probe	3861	5	99.87%
DoS	15270	0	100%
U2R	19	1	95.0%

Logistic Regression			
Normal	21263	836	96.21%
R2L	175	141	55.38%
Probe	3262	604	84.37%
DoS	14835	435	97.15%
U2R	6	14	30.0%
Support Vector Machine (SVM)			
Normal	21864	235	98.93%
R2L	245	71	77.53%
Probe	3707	159	95.85%
DoS	14943	327	97.85%
U2R	6	14	30.0%

In Table 3. the accuracy rate of all models on various attacks has been shown. To gauge the accuracy of machine learning models on different attacks we used various parameters and calculated the true positives, false positives, true negatives and false negatives using the confusion matrix. Random Forest model was able to classify all the attacks with a commendable accuracy and was also able to solve problem of false alert rate which existed with the other models and also the traditional Intrusion detection system i.e. alarming even when the attack has not happened.

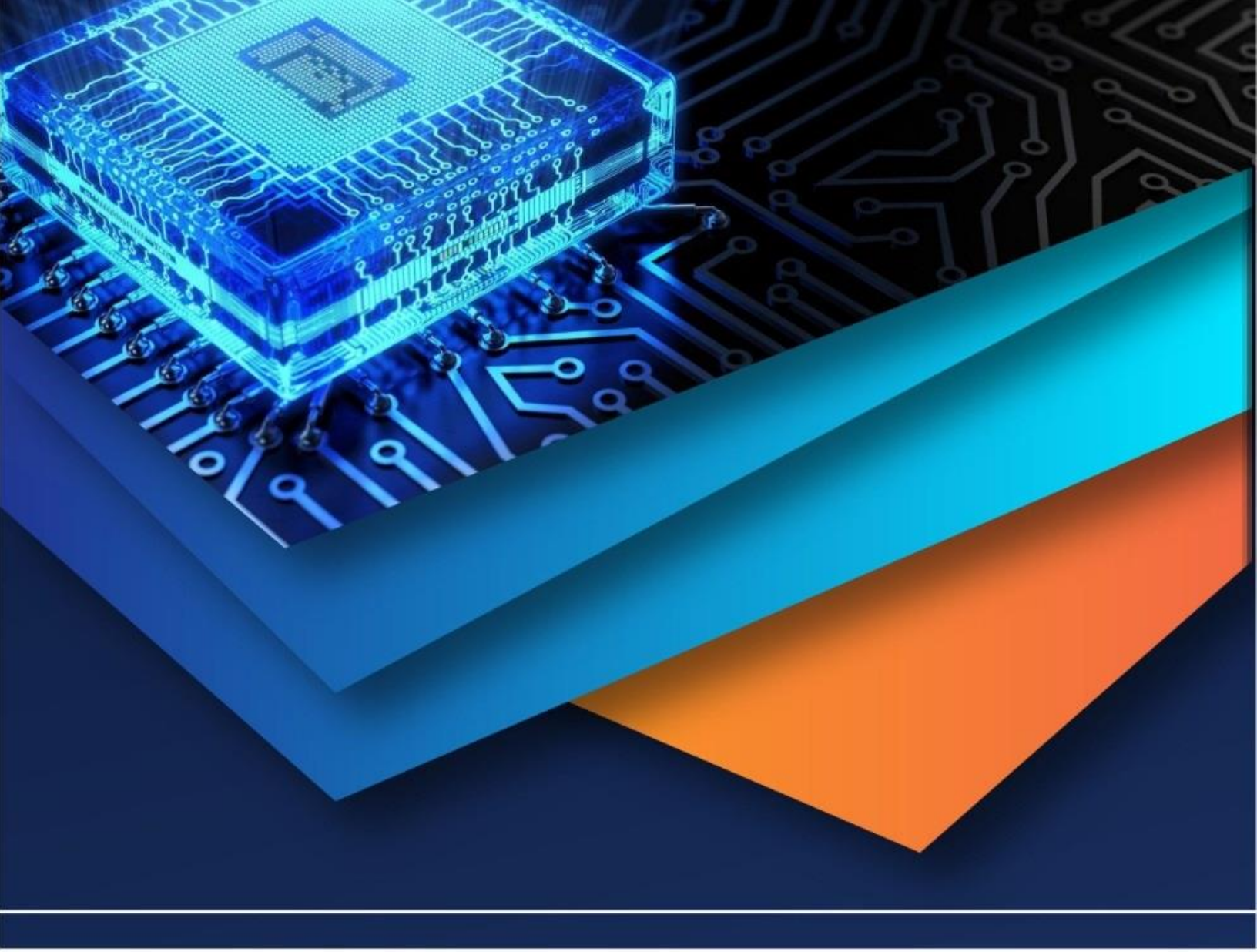
V. CONCLUSIONS

The importance of network security is growing commensurately with people's increasing dependence on technology. The goal of making everything available readily and everywhere has led to a revolution in the field of networks. In spite of the tremendous growth of technologies, we still lack in preventing our resources from theft and cyber attacks. The primary benefit of an intrusion detection system will be to notify when an attack or network intrusion might be taking place. With the help of it, various crimes and larceny can be prevented.

In this paper, several experiments were performed and tested to evaluate the efficiency and the performance of the following machine learning classifiers: Logistic Regression, Gaussian Naive Bayes, Decision Tree, Random Forest and Support Vector Machine for developing an anomaly based intrusion detection system. The tests were based on NSL-KDD Train Dataset and Random forest classifier registered the highest accuracy rate. Additionally, we can exclude some of the other models which could not perform up to the mark as seen in the results. Furthermore, we will require more data and need to explore various other subcategories of different attacks which were not included in this dataset to avoid the system getting exposed to new threats which our model has not encountered. Also, we can perform hyper-tuning of the various parameters of the model to improve the performance.

REFERENCES

[1] Ben Lutkevich “intrusion detection system (IDS)” searchsecurity.techtarget.com https://searchsecurity.techtarget.com/definition/intrusion-detection-system (February 2020)
[2] Cuelogic Technologies “Evaluation of Machine Learning Algorithms for Intrusion Detection System” medium.com https://medium.com/cuelogic-technologies/evaluation-of-machine-learning-algorithms-for-intrusion-detection-system-6854645f9211 (May 2019)
[3] NSL-KDD Data set for Network-based Intrusion Detection Systems. http://nsl.cs.unb.ca/NSL-KDD
[4] Stuart J. Russell, Peter Norvig “Artificial Intelligence: A Modern Approach, Third Edition” https://www.pearson.com/ https://www.pearson.com/us/higher-education/program/Russell-Artificial-Intelligence-A-Modern-Approach-4th-Edition/PGM1263338.html (2020)
[5] Prathamesh Thakar ” The math behind Machine Learning Algorithms” https://towardsdatascience.com https://towardsdatascience.com/the-math-behind-machine-learning-algorithms-9c5e4c87fff (Jul 2020)
[6] Nagesh Singh Chauhan “Decision Tree Algorithm, Explained” www.kdnuggets.com https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html
[7] L. Breiman, “Random forests,” Machine learning, vol. 45, no. 1, pp. 5– 32, 2001



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY