Sparkling Light Publisher

Sparklinglight Transactions on Artificial Intelligence and Quantum Computing(STAIQC)

Website: **https://sparklinglightpublisher.com/ ISSN (Online):2583-0732**

Performance Analysis of Network Attack DetectionFramework using Machine Learning

Mustafa Basthikodi[a,*], Ananth Prabhu G[b], Anush Bekal[c]

[a,*,b,c]*Sahyadri College of Engineering and Management, Mangaluru, Karnataka, India*

Abstract

The network attack detection frameworks are developed to find out the access to computing systems that are unauthorizedly connected across the networks. The intrusion detection is one of such frameworks, developed by that has a higher accuracy for all majority attacks in comparison to existing works. The models deploy different classifiers to demonstrate that the approach is modular in structure. Intrusion detection model developed in this analytical research utilises various machine learning classifiers like Random Forest, SVM, K-Nearest Neighbor, and Naïve Bayes. Experimentation was conducted on dataset NSLKDD, The Performance of classifiers improved as dimensionality reduction and feature selection improves accuracy and reduces false alarm rate. A better generalization is also achieved while integrating multiple classifiers. High accuracy is obtained for all majority attacks in the NSLKDD datasets which is the widelyavailable benchmark datasets for intrusion detection.

Keywords: IoT, Smart Healthcare, Sensors, Physical health, Tele-medicine, Patient monitoring system, Network of Things.

1. Introduction

Intrusion Detection Systems are one among the numerous techniques utilized within the field of cyber security. This system does not supplement any other security instruments, yet they complement those by making an effort to detect when harmful conduct happens. Reason for an IDS, by and large, is to distinguish if the conduct of the client clashes along with the expected utilization of the PC, or the computer networks, for instance hacking the systems to procure the data, committing fraud, directing an attack to prevent the framework from functioning appropriately or even breaking down. During 1990s, the system managers used to perform the intrusion detection, by system messages and physically checking the logs of client count, without any option of having the chance of recognizing interruptions in development [1]. This has step by step changed, with early works of [2] and [3], by creating software programming to naturally examine the information for the system managers. The initial IDS to accomplish this progressively were created in the mid-1990s. Nonetheless, because of the expanded utilization of PCs, the size of data in contemporary PC networks actually delivers this a critical challenge as mentioned below in Fig.1 sites is extremely difficult to stay aware of sites is extremely difficult to stay aware of.

E-mail address of authors: [a,*]*mustafa.cs@sahyadri.edu.in,* [b]*ananthprabhu@sahyadri.edu.in,* [c]*anush.ec@sahyadri.edu.in*

Please cite this article as: Mustafa Basthikodi, Ananth Prabhu G, & Anush Bekal (2021). Performance Analysis of Network Attack Detection Framework using Machine Learning. *Sparklinglight Transactions on Artificial Intelligence and Quantum Computing (STAIQC)*, *1*(1), 11–22. **ISSN (Online):2583-0732**

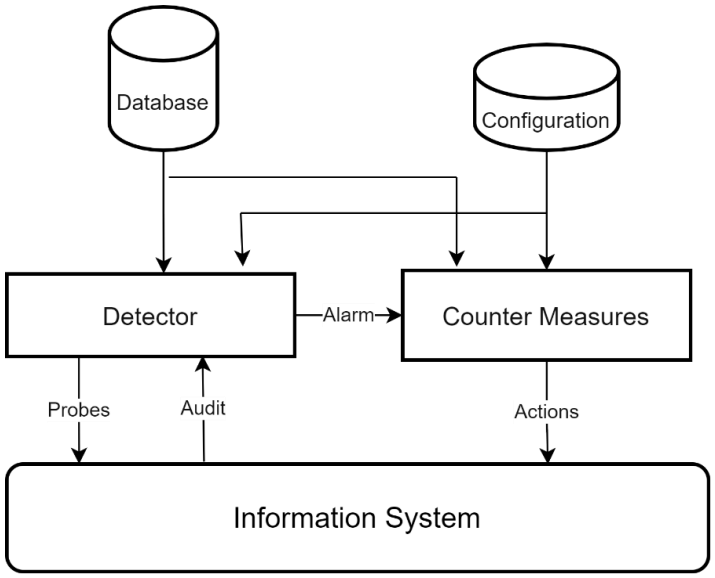

Fig.1. The General Structure of Attack (Intrusion) Detection System

This has prompted research to develop network safety applications to reduce the loss of individual information and decrease the harms caused by cybercrimes on the grounds that an appropriately organized cyber-attacks can make broad harm to an organisation. The conventional techniques in place can't stay aware of the fast advancements that happing in the cybercrime space. The use of blacklists is a standard technique for relieving phishing attacks. A blacklist is a compiled list of unsafe URLs which are refreshed and curated by the security business, for example Avast. A network removes all URLs that which occurs on the blacklist and gives permission to the networks to pass through various URL's. By the year 2016, there were about 300,000 exceptional malicious sites detailed month to month, this possesses a challenge for the security organization to make the blacklist in two phases, first a phishing site should effectively attack a network before it has been hailed to be ill-conceived & blacklisted in light of the fact that all phishing is made to copy authentic sites, and second the large number of new phishing sites is extremely difficult to stay aware of.

The approaching traffic of the Internet is separated by the customary firewall built in; however, the intruders easily discover approaches to break the firewall. As standard outline, any irrelevant individual will have the option to interface with the Intranet of private network by dialling in through a modem. The firewall will not be able to anticipate this sort of an access. Accordingly, an Intrusion Detection System is a security structure that screens PC network and system traffic and checks that development for a feasible undermining assault beginning from outer part of the association inaddition to the misuse or attacks starting from within the organisation.

Standard firewalls can't recognize inside attacks, for instance User-to-Root attacks, port scanning and flooding attacks because they simply track down framework packs at the organization limits. This astounding attack cannot be distinguished by the standard firewall, for instance, Denial of Service (DoS) and DDoS. Additionally, ordinary firewalls can't separate between normal activity and DoS attack movement. Access control fills is the cutting edge of intrusion obstruction that supports trustworthiness and privacy parameters. Intrusion identification is the means of logically watching functions happening in a network or computer, examining for detection of probable episodes and frequently eliminating the unapproved access. Given the difficult and quick moving horizon of cyber safety, it is difficult to hard code a machine or segment with specific features and anticipate that it should be working adequately at all the occasions. Or maybe the main focus should be devising a module which incorporates the past information and bases the outcomes on the experience that it is having. This makes the field profitable for the use of Machine Learning strategies, wherein machines are not expressly customized, rather they are put in some ecological conditions wherein they sense the patterns of premium and patterns of premium, for this situation, are interruptions. The Table 1 lists

outa portion of the guarded mechanisms pointed toward perceiving system attacks.

Table 1. Security Options for attacks

Security Technique	**Illustration**	**Protection Type**	**Example of Attacks**
Intrusion Detection System	Equipment aimed at monitoring network traffic for any connections which is potentially harmful	Internal + External	U2R, DDoS, IP Scanning, Flooding, Port Scanning,
Firewall	Designed to stop unauthorized access.	External	DoS, Eavesdrop,Port Scan
Access Control	Equipment aimed at Controlling illegal access to a system.	External	Sniffer attacks, Password attacks, Dictionary attacks
Cryptography	Maintain the confidentialityof data aimed at stopping encoding ordecoding of secret messages.	External	Meet in the middle attacks,Brute force attacks

An IDS is made out of different segments specialized in detecting the information on the network, from that point breaking down the network connection for potential assaults lastly disturbing the administrator. Figure 2 depicts the general design of attack system IDS. This has three significant parts i.e., Traffic Capture, Attack Detection, and the Response Agent.

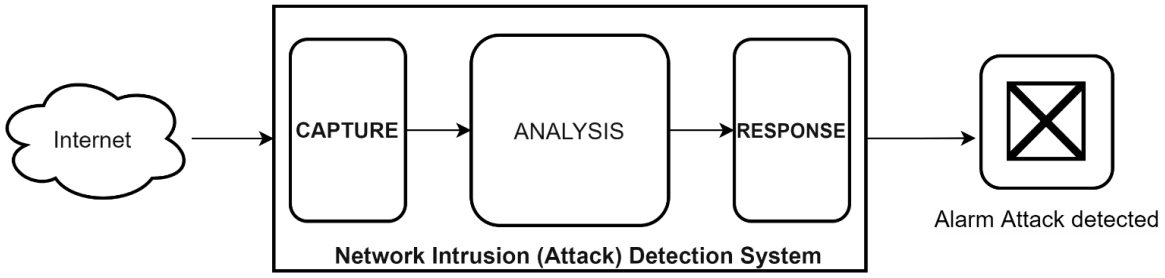

Fig. 2: General Design of Network Intrusion (Attack) System

The key component of the above mentioned System is detecting the network traffic, and the catching module is the one which acknowledges this work. The module catches the raw traffic information at packet level utilizing Gulp, Wireshark, and so on. The caught packet level traffic must be pre-handled prior to shipping off the discovery engine. Flow level information, if there arises an occurrence of networks, is made out of data summed up from at least one packets.

When the traffic has been caught and pre-handled by the capturing unit, the following stage in the process is attack identification. At the core of IDS is a discovery methodology which can be founded on anomaly identification or misuse recognition, subsequently now a NIDS can have a mark coordinating strategy or inconsistency location. When the connection has been distinguished as an assault it is the obligation of Response Module to complete the proper action. The action can be many, such as, cautioning the manager about the incidents, dropping the bundles, shutting the connections and hindering the machine from sending further parcels. Not simply this, the module needs to give the criticism to the assault recognition module withthe point of refreshing its conduct accordingly.

Generally, interruption detection frameworks are highly dependent on a human interaction to keep up with the recent updates. This dependence incorporates adding new guidelines to guideline-based frameworks, adding newly found phishing URLs to blacklists, being curation of whitelists and making exemptions. The sheer size of networks that exist and development of the web and the rapid creation pace of zero-day assaults, the deficiency of human based interruption discovery framework is simple to observe. One approach to battle this fault is Machine Learning. Most importantly, Machine Learning is something that improves with scale, an ML framework is just framework that depends on gaining knowledge from previous events of attempted interruptions and assaults, learning the examples which join them and how different it is from normal conduct [15]. When an ML algorithm has found these pattern acknowledgments and grouping occurs at a quicker rate for a bigger scope that any human- driven framework can operate. The Fig.3 illustrates the workflow of machine learningalgorithm.

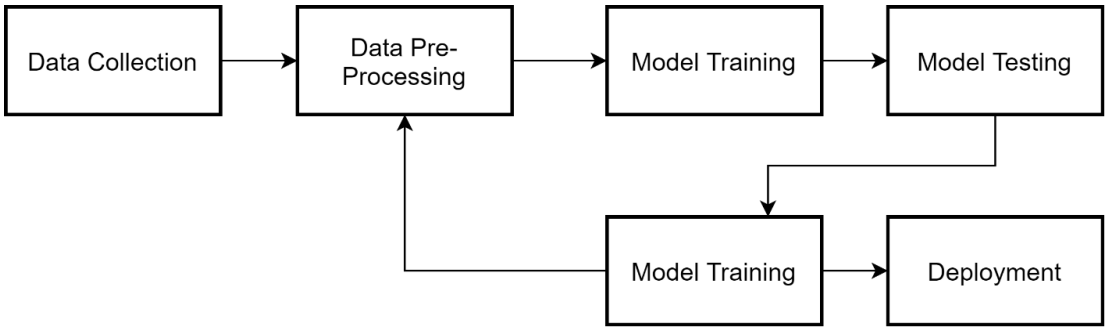

Fig.3. Workflow of ML Algorithm

Data Collection: Determining significant information on the project explicit for the research is gathered and then retained in memory. Data prepossession: Now information gathered is properly arranged, and changed into a configuration that can be taken care of into the AI algorithm. As a rule, this is put away in the form of a numpy arrays or a table. Highlight removal additionally happens now (all the data through the information assortment stage will not be pertinent to examination, so a few things are completely ignored). The previously handled information is later divided for testing datasets and preparation [4].

Feature Extraction: This is a pre-handling task which includes choosing particular applicable highlights for making the preparation and test datasets which would later be fed into the algorithm. That accomplishes several points, it reduces the probability of over fitting, this also makes interpretations simpler and it enhances the opportunity in making speculation. Inappropriate feature extraction will be able to prompt the model to run for long rather than would normally be appropriate as it needs to experience more information than it actually requires for the purpose of testing and training. Training**:** The training dataset from the pre handling stage of information is placed into the chosen machine learning algorithm at this pointand a model is assembled. This process can be carried out once or may be repeated, dependingon the algorithm.

Testing: After the construction of the model, test dataset obtained through the information pre-processing stage is fed into, to demonstrate in very similar procedure as that of a training dataset. Forecast or Characterization authenticity has been done. More like the model will be better if the accuracy is hundred percent. Obviously, in this stage it is factually difficult to construct a model with full accuracy as the information size keeps developing also changes have been made to the model, it is replicated in stages 2 through 4. Deployment**:** Now, based on the analysis, the model with the best classification or expectation is selected, on the basis of examination of the results.

Since the focal point here is to develop a ML based attack Detection System, we present an outline of Machine Learning based Network Intrusion Detection System (MLNIDS) displayed in Figure-4. The cycle begins with the catching of traffic records and the caught connections are sent to the IDS. The IDS process starts with the sending of captured connections with the information pre- handling unit. Here the records are changed into suitable structure in order to be prepared by ML strategies. As the measure of data captured can be excessively huge, a fitting decrease of

the information is vital, in order to leave away the less significant factors, without losing a greater part of the data. After the excess attributes of the data set are eliminated, the subsequent stage is to advance the information to the suitable classifier. There can be a solo classifier or a group of them laid in some order. The classifier will bring about the model for the normal data. When the model is prepared, it very well may be tried for the adequacy, utilizing the test data. At that point there is a decision-making element, whose point is to choose if the connection is normal or a disaster, for the situations where the connection is destructive, the IDS needs to create the caution and start the corrective method, by educating the system admin.

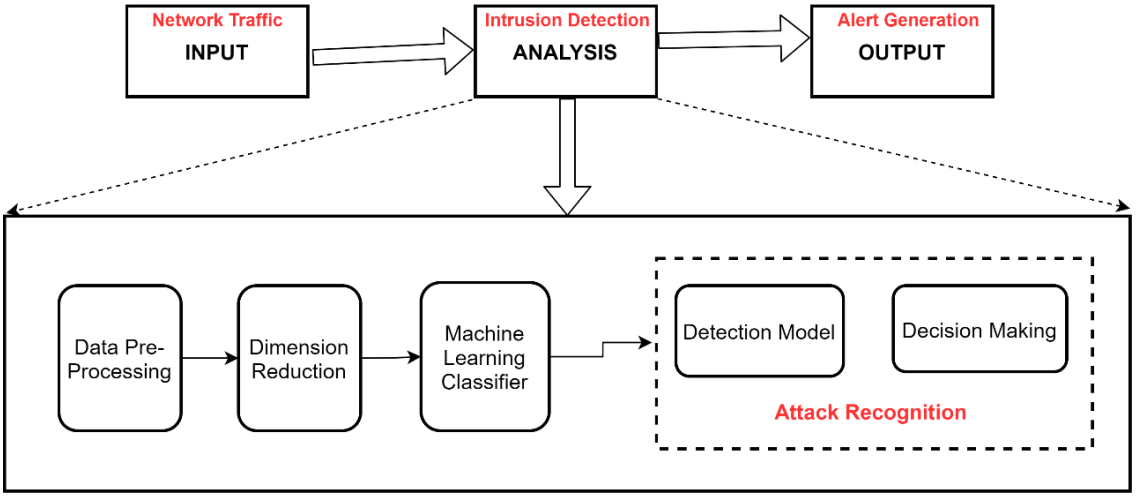

Fig.4: Architecture of Machine Learning based Network Intrusion Detection System

In this work, we have carried out detail survey on the existing Intrusion Detection Systems (IDS) and various classifiers of machine learning technologies, which are used for detecting malicious user attacks in the network. The experimental results were conducted on dataset NSLDB. An integrated network IDS model has been progressed by making use of base classifiers, also ensemble of classifiers separately to analyse the accuracy and time computation of the classifiers. The different classifiers utilized are Naïve Bayes, Random Forest, K-NN, Decision tree and SVM. Experimental results show the classifiers producing higher accuracy as well as bring down false alarm level in relation to base classifiers with a trade-off of computation time.

2. Literature Survey

The methodology tends to group current contemporary wireless IDS strategies which is dependent on the track recognition strategy, trust model, wireless network, and collection measure and analysis procedure. Summing up advantages as well as the disadvantages of various or similar kinds of considerations and concerns for the wireless interruption identification regarding explicit aspects of target wireless networks including Mesh networks, LANs, Ad-hoc networks, sensor networks, Mobile telephony, PAN, and Cyber physical systems [5] [6]. Intrusion detection systems assume a fundamental function in research undertaking with an increase in assaults on networks and PCs [7]. IDS display functions that occur in a networks and PC device to dissect patterns of intrusion. IDS aim to identify intrusions with a good rate in detection and a law false caution rate [8]. Regardless, the fact that arrangement related data extraction techniques are famous, they aren't compelling to recognize unknown attacks. Reducing false alarm rate has become a difficult task in spite of the fact that the present intrusion detection techniques give attention to the most recent kinds of assaults like R2L, U2R, Probe and DoS, framework conduct is a significant boundary on

which the anomaly-based detection framework depends upon. In event that the system conduct is within predefined conduct, at that point the system exchange will be acknowledged or else, it activates the alarm in the AIDS [9].

SIDS (Signature intrusion detection system) relies on structure coordinating procedures for locating the known attack; they are otherwise called Misuse Detection or Knowledge-based Detection [10]. In the Signature Based Intrusion Detection systems, in order to locate a past intrusion coordinating strategies are utilized. As such, when the signature of a past intrusion matches with an intrusion signature which as of now resides in the signature data base, a warning sign will be set off. Host's logs for SIDS, is being examined for discovering arrangements regarding orders as well as activities which was earlier distinguished to be malware. SIDS has likewise been marked in the literature as detecting Knowledge-Based or Misuse [30]. SIDS mostly gives superb detection accuracy for recently known intrusions [11]. In general, various techniques have been used to create a signature for SIDS, where signatures are regarded as state machines [12], string pattern, formal language or semantic conditions [13]. The expanding pace of zero-day assaults [14] has delivered SIDS strategies progressively less successful on the grounds that no earlier signature exists for any such assaults. The IDS can likewise be characterized on the basis of input data sets being used to recognize unusual exercises. Regarding the source of data, basically there are two kinds of IDS technologies; they are Network-based IDS (NIDS) and Host-based IDS (HIDS). Host based IDS examines the information that begins from the audit sources and host system, for example, several logs namely windows server, firewalls, application system audits, operating system or database logs. Host based IDS can distinguish interior assaults which don't include network commuters [15]. NIDS screens the framework traffic which is extracted from a framework by the method of Net Flow, packet capture and other network data sources. IDS on the basis of network can be utilized for screening numerous PCs that which are connected to a network.

NIDS conveyed at various situations within specific network geography, along with firewalls and HIDS, which will be able to provide solid, versatile, and security on multi levels, both against insider and outer attacks. A description of resemblance between NIDS and HIDS is shown in table 4. Creech et al. suggested a HIDS philosophy that applies dis-continuous system call assemblies, with an intention to bring up location levels where as diminishing fake caution levels [16]. AIDS methods may be processed into three primary gatherings: Data-based [17], information based [18], and AI based [19] [20].

Extraction of information from huge volume of data can be done through machine learning. ML models include a bunch of techniques, rules, or complex "transfer functions" which may be implemented to anticipate or perceive conduct, or to discover fascinating information designs [21]. ML methods has been broadly applied in region of Anomaly based Intrusion Detection System. A few procedures and algorithms, like, grouping, neural organizations, hereditary algorithms, rules affiliated, decision trees and closest neighbour methods, is being used for finding information from intrusion data base [22]. Some earlier research has inspected the utilization of various procedures to build AIDSs. The researchers examined the presentation of element choice algorithms namely Classification Regression Trees (CRC) and Bayesian networks (BN) and consolidated those strategies for high precision [23]. The research works done likewise proposed a method for feature choice utilizing a blend of highlight determination algorithms, for example, Correlation Attribute evaluation and Information Gain (IG). They tried presentation of feature chosen by using algorithms of distinctive arrangement, for example, NB-Tree, Multi-Layer Perceptron, native Bayes, and C4.5 [24] [25]. The vital focal point of intrusion detection system dependent on ML research is for identifying examples as well as for fabricating IDS dependent on the data base. For the most part, there are two sorts of machine learning strategies, administered.

As referenced in [26], information with countless aspects influence the learning model which shows over fits and diminishes exhibition, increasing size of memory use, and computational cost for insightful. Indeed, exceptionally uncommon analysts who think about computational time in their works, particularly in peculiarity identification. Then again, Information Gain has been broadly utilized by analysts to dissect critical and pertinent highlights. As per works in [27] to Information Gain is utilized to lessen dimensionality by choosing more applicable highlights through component weight figuring. Disposing of unimportant highlights may improve the exhibition of the discovery framework. The research work under [28] incorporates EFS, Synthetic Minority Oversampling Technique and Principal Component Analysis for enhancing output of AdaBoost-based intrusion detection system on CICIDS-2017 Data base. Researchers say that integrated technique surpasses support vector machine-based strategy with

respect to precision, accuracy, F1 Score, and recall. Research performed by [28] involves several anomalies identification experiments using Random Forest. Few anomalies identification research which makes use of Bayesian theorem comprise research works by [29] and [30]. The decision tree which is based on a random attribute array. A node is a test component & the outcomes are represented by branches. The end decision made after measuring every characteristic in context of class labels [31] is displayed by the decision leaves. ML algorithms implement mathematical formulas to evaluate information sets as well as forecast values dependent on data base. ML algorithms may be used in the field of cyber safety for evaluating as well as training IDS on safety-related data sets. We evaluated various ML algorithms in paper [32], for examining NSL-KDD data set by using KNIME algorithms. The study [33] intends to develop an approach to improve performance IDS to deal with disparity in training dataset.

3. Proposed Methodology and Implementation

The Machine Learning was resulted by the quick growth of data extraction methods and techniques. The fundamental thought of any ML mission is to prepare model, on basis of few algorithms, in doing a specific activity: regression, classification, and cauterization. Etc. The input dataset is basis for training and model built is utilized for making predictions. The procedure to determine the work is shown below in the Fig.5. Data Pre-processing using One-hot encoding: Essentially, in this progression the dataset needs to experience a cleaning cycle to eliminate duplicate records, as the NSL KDD dataset was utilized which has just been cleaned, this progression isn't any longer required. Next a Pre- preparing activity must be assumed in position on the grounds that the dataset contains mathematical and non-mathematical cases. For the most part, the assessor (classifier) characterizes in the scikit-learn functions admirably with mathematical data sources, so a one-of-K or one-hot encoding technique is utilized to make that change. This procedure will change each categorical element with m potential contributions to n binary features with one dynamicat the time in particular.

Features scaling: The Features scaling is a typical necessity of ML strategies, to dodge that features with enormous qualities may weight a lot on the end-results. For each feature, compute the normal, deduct the mean a value from the feature values, and then divide the outcome by their standard deviation. Subsequent to scaling, each component will have a zero normal, witha standard deviation of one.

Features Selection: The feature determination may be utilized to remove repetitive and immaterial information. One of the strategy in choosing a subset of significant features that completely speaks to this issue close by a base disintegration of presentation, two potential explanations were investigated why it would be prescribed to confine the quantity offeatures:

Although it may be conceivable initially, the immaterial highlights could recommend connections among the features and target classes that emerge just by some coincidence and don't accurately show the issue. This viewpoint is likewise identified with over-fitting, typically in a decision tree classifier. Also, many of its features could incredibly expand calculation time without a comparing classifier improvement. A univariate feature determination with ANOVA F-test for feature scoring begins first and feature examinations will be done for each element exclusively to know the power of association of these elements with their names. The Select Percentile technique in the scikit-learn feature selection module was utilized; this strategy selects features dependent on a percentile of the most noteworthy scores. After knowing the top subset of features. Once, the best subset of features was discovered, a recursive element disposal can be applied and that can be over and over form a model, setting the component aside and afterward rehashing the cycle with the remained features until all highlights in the dataset are depleted. All things considered; it will be a decent enhancement to know the best discovery executing subset of features. The thought here to utilize loads of a classifierto deliver a feature ranking.

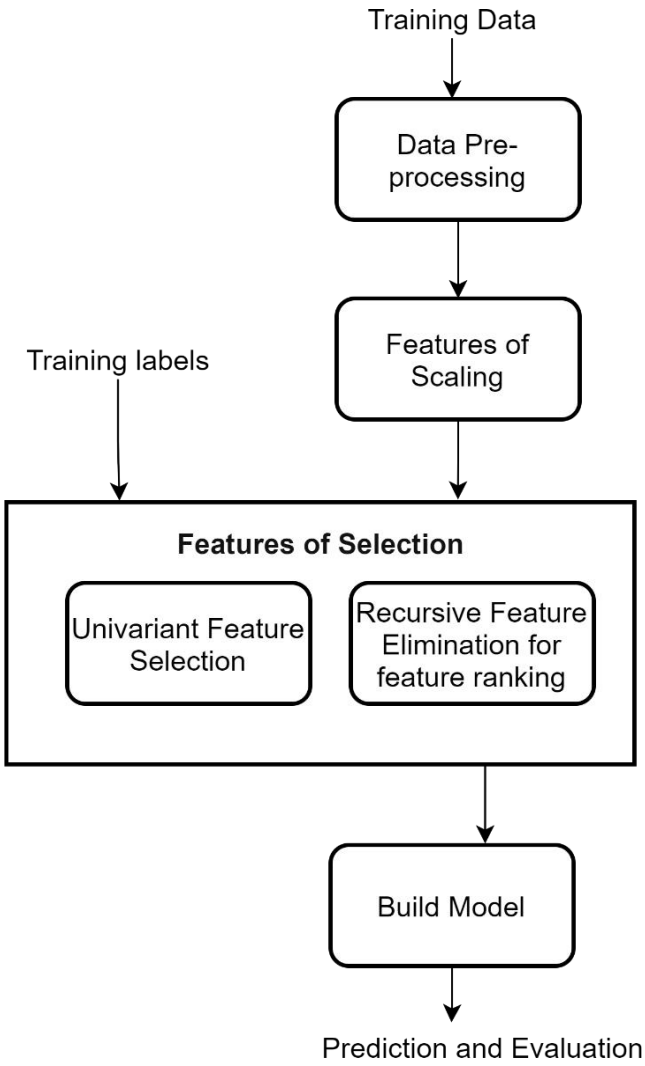

Fig. 5: Proposed methodology

Build Model: The model worked to segment the information utilizing data expansion until examples of every leaf node consume uniform class labels. Even though it is traditional one, however, yet a powerful various levelled strategy for supervised learning whereby the nearby space is perceived in a succession of dull parts in a diminished number of steps. In every test, a solitary feature is utilized to part the node as per the element's qualities. On the off chance that after the split, for each branch, all the occasions chose have a place with the comparative class, the split is viewed as pure or complete. For prediction of our model and for evaluation, test data used and its numerous settings was viewed as, for example, the exactness score, accuracy, review, f-measure and a confusion matrix. For experimentation, NSL KDD data set is used in this work. Features are given below:

- No redundant instances in the training set, therefore no biased model for IDS.
- No duplication of instances in the data-set, therefore a good accuracy rate.
- The total count of instances for each difficult attack category is inversely proportional to percentage of total instances in full KDDCUP'99.

The additional attack categories are present in data are not present in the training set. The training data-set has 21 different attack categories whereas testing data- set has 37 different attack categories. The Table 4.4 lists out the

various types of attacks and their presences in various flavours of the KDD full data-set and further lists out the new attackcategories and its attack family.

3.1 Performance Metrics for IDS

The IDS have sample number of classification matrix. Some of them are addressed by various names. In order to assess the variety of accurate estimates, the ground reality regard becomes very important. This reality can be made out of cluster of framework connection record set as an Attack or Benign because of Binary distribution, and nine types of assaults if there is an occurrence of multi-class distribution.

In order to define and decide on the nature of order models the accompanying terms are used. TP (Truly Positive): No. of association records (NAS) correctly grouped to benign class.

TN (Truly Negative): NAS accurately grouped to Attack class.

FP (Falsely Positive): Number of Normal association records incorrectly predicted as Attack connectionrecord.

FN (Falsely Negative): No. of Attack association records wrongly predicted as Normal connectionrecord.

Hence, by utilizing the aforesaid terms, the below listed assessment metrics were regularly measured:-Recall or TPR (True Positive Rate), Precision, F1-Score, accuracy, Confusion Matrix, Support, and the FPR (False Positive Rate). The evaluation of IDS is dependent on the accompanying standard measures of performance: Hit rate called TPR (Truly positive rate) is affectability measure which belongs to recognized samples which are malwares among all the samples. The formula for TPR is shown below.

$$TPR = \frac{TP}{TP+FN}$$

The miss rate known as Falsely negative rate (FNR) demonstrates the piece of unidentified samples which are malwares among the total amount of samples. The formula for FNR is shown below:

$$FNR = \frac{FN}{FN+TP}$$

The Falsely certain rate (FRP) shows the part of the samples that are benign applications recognised as malware among all the other samples. The formula for FRP is:

$$FRP = \frac{FP}{FP+TN}$$

The proportion of the particularity known as Truly negative rate (TNR) demonstrates the samples parts that is benign applications recognized as benign apps, among all the samples. The TNR is formulated as below:

$$TNR = \frac{TN}{TN+FP}$$

The part of malware samples identified among the total samples identified as malwares, indicates the precision. Normally a high value for precision is desirable. The precision formulais given below:

$$Precision = \frac{TP}{TP+FP}$$

Accuracy shows the parts of samples that are accurately recognized among the total samples. Precision performs well when the datasets are balanced. High accuracy is expected in general.The formula for accuracy is shown below

$$Accuracy = \frac{(Tp+TN)}{Tp+TN+FP+FN}$$

Harmonic mean of the precision is named as F-measure. In general, higher F-measure is usually expected. It is formulated as below:

$$F\text{-}measure = \frac{2TP}{(2TP+FP+FN)}$$

4. Experimentation and Result Analysis

The experiments are done using Jupyter in Windows environment, An open-source tool along with Scikit-learn for implementing ML classification algorithms. Accuracy is calculated as evaluation metrics to find the

performance using NSL-KDD datasets. The Table 2 Gives the accuracy of various ML algorithms used for different types of attacks.

Table 2. Accuracy values for various attacks

Attack /ML Algorithms	NB	RF	KNN	SVM
U2R	0.847	0.976	0.974	0.983
PROBE	0.756	0.849	0.827	0.867
DoS	0.885	0.716	0.819	0.896
R2L	0.880	0.740	0.760	0.717
Avg	0.842	0.820	0.845	0.866

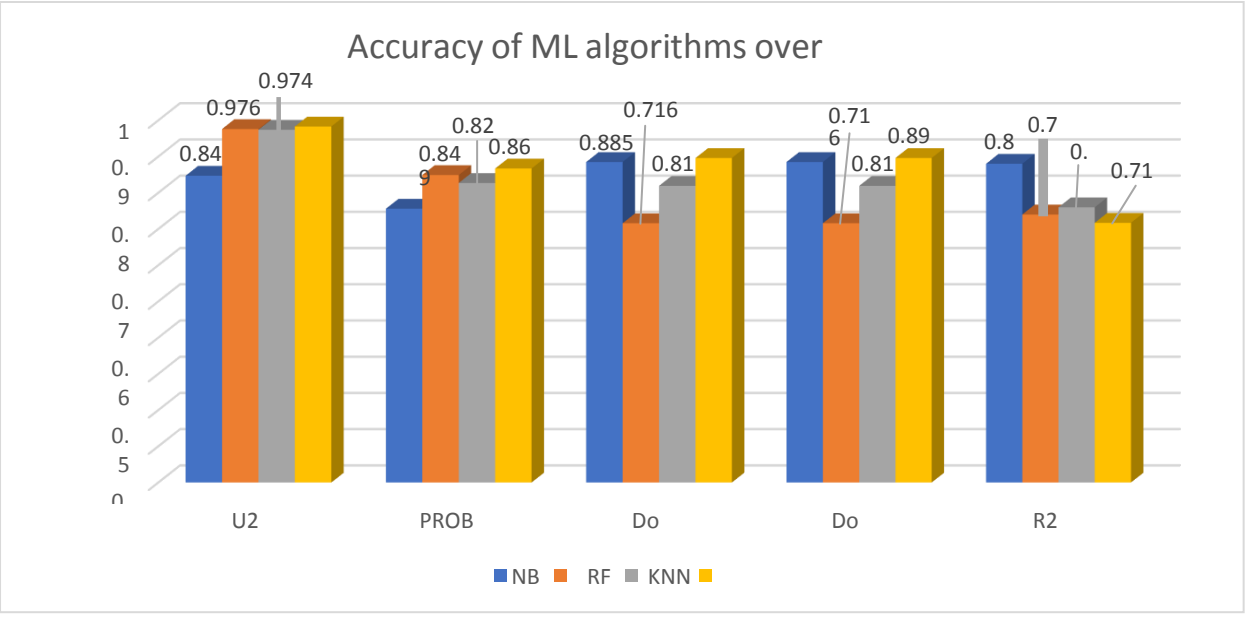

Fig.6. Graphical analysis of ML classifiers versus Attacks

It is found that the relative accuracy rate which was obtained from the above mentioned four ML classifier in detecting normal and four-attack classes. We have picked 13 features from all NSLKDD dataset and the following shows the value obtained with various algorithms. It is found that the relative accuracy rate which was obtained from the above mentioned four ML classifier in detecting normal and four-attack classes. We have picked 13 features from all NSLKDD dataset and the following shows the value obtained with various algorithms.

- Less accuracy is obtained for the DOS attack considering 13 features dataset over complete dataset.
- Accuracy is same from all the classifiers for U2R attack class, RF and SVM
- Increased accuracy for selected features of R2L attack when learned with NB and KNN
- Graphical analysis in Fig.6 demonstrates the accuracy of ML algorithms applied. The graphical analysis of average accuracy is demonstrated in Fig.7.

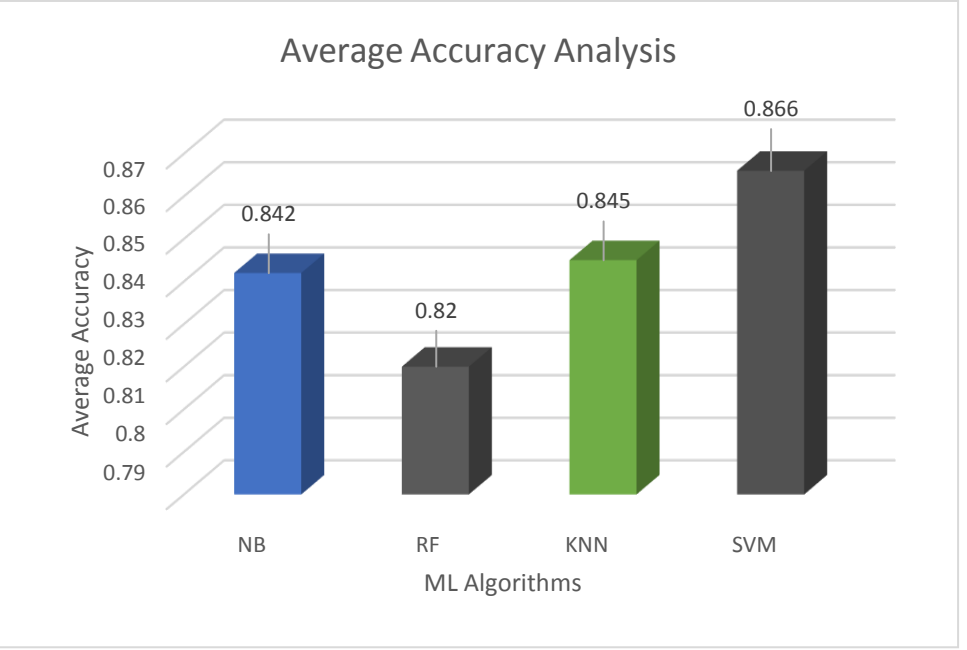

Fig. 7: Graphical Analysis Average Accuracy values of ML classifiers

5. Conclusion

The various types of attacks that can occur in a network are discuss in the initial part of the paper. The general architecture of network intrusion detection system and Machine learning approaches for modeling the IDS is proposed. The database NSLKDD is used to implement the model making use of machine learning classifiers. The result achievedusing NB, RF, KNN and SVM classifiers are tabulated and analysed.

References

[1] Stefanova, Z. S. (2018). *Machine Learning Methods for Network Intrusion Detection and Intrusion Prevention Systems*. University of South Florida.

[2] Mishra J.S., Gupta N.K. (2015). Framework for Host-based Botnets Detection System. *Journal of Computing Technologies*, 4(1), 34-38.

[3] Azeez, N. A., Bada, T. M., Misra, S., Adewumi, A., Van der Vyver, C., & Ahuja, R. (2020). Intrusion detection and prevention systems: an updated review. *Data management, analytics and innovation*, 685-696.

[4] K.Chumachenko (2017). Machine Learning Methods for Malware Detection and Classification. Bachelor's Thesis, Dept. Information Technology, Univ. of Applied Science.

[5] Khraisat, A., Gondal, I., & Vamplew, P. (2018, June). An anomaly intrusion detection system using C5 decision tree classifier. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 149-155). Springer, Cham.

[6] Symantec (2017). Internet security threat report 2017. Available at https://www.symantec.com/content/dam/symantec/docs/reports/istr-22-2017-en.pdf.

[7] Buczak, A. L., & Guven, E. (2015). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications surveys & tutorials*, *18*(2), 1153-1176.

[8] Walkinshaw, N., Taylor, R., & Derrick, J. (2016). Inferring extended finite state machine models from software executions. *Empirical Software Engineering*, *21*(3), 811-853.

[9] Studnia, I., Alata, E., Nicomette, V., Kaâniche, M., & Laarouchi, Y. (2018). A language-based intrusion detection approach for automotive embedded networks. *International Journal of Embedded Systems*, *10*(1), 1-12.

[10] Kshetri, N., & Voas, J. (2017). Hacking power grids: A current problem. *Computer*, *50*(12), 91-95.

[11] Xiao, L., Wan, X., Lu, X., Zhang, Y., & Wu, D. (2018). IoT security techniques based on machine learning: How do IoT devices

use AI to enhance security?. *IEEE Signal Processing Magazine*, *35*(5), 41-49.
[12] Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, *2*(1), 1-22.
[13] Alcaraz, C. (2018). Cloud-assisted dynamic resilience for cyber-physical control systems. *IEEE Wireless Communications*, *25*(1), 76-82.
[14] Lyngdoh, J., Hussain, M. I., Majaw, S., & Kalita, H. K. (2018, July). An intrusion detection method using artificial immune system approach. In *International conference on advanced informatics for computing research* (pp. 379-387). Springer, Singapore.
[15] Rath, P. S., Barpanda, N. K., Singh, R. P., & Panda, S. (2017). A prototype Multiview approach for reduction of false alarm rate in network intrusion detection system. *International Journal of Computer Networks and Communications Security*, *5*(3), 49.
[16] Sadreazami, H., Mohammadi, A., Asif, A., & Plataniotis, K. N. (2017). Distributed-graph-based statistical approach for intrusion detection in cyber-physical systems. *IEEE Transactions on Signal and Information Processing over Networks*, *4*(1), 137-147.
[17] Yuan, Y., Kaklamanos, G., & Hogrefe, D. (2016, November). A novel semi-supervised adaboost technique for network anomaly detection. In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (pp. 111-114).
[18] Aburomman, A. A., & Reaz, M. B. I. (2016). A novel SVM-kNN-PSO ensemble method for intrusion detection system. *Applied Soft Computing*, *38*, 360-372.
[19] Sun, Y., & Wu, D. (2008, April). A relief based feature extraction algorithm. In *Proceedings of the 2008 SIAM International Conference on Data Mining* (pp. 188-195).
[20] Stiawan, D., Idris, M. Y. B., Bamhdi, A. M., & Budiarto, R. (2020). CICIDS-2017 dataset feature analysis with information gain for anomaly detection. *IEEE Access*, *8*, 132911-132921.
[21] Popoola, E., & Adewumi, A. O. (2017). Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision. *Int. J. Netw. Secur.*, *19*(5), 660-669.
[22] Farnaaz, N., & Jabbar, M. A. (2016). Random forest modeling for network intrusion detection system. *Procedia Computer Science*, *89*, 213-217.
[23] Sheikhpour, R., Sarram, M. A., Gharaghani, S., & Chahooki, M. A. Z. (2017). A survey on semi-supervised feature selection methods. *Pattern Recognition*, *64*, 141-158.
[24] Aljawarneh, S., Aldwairi, M., & Yassein, M. B. (2018). Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, *25*, 152-160.
[25] Manzoor, I., & Kumar, N. (2017). A feature reduced intrusion detection system using ANN classifier. *Expert Systems with Applications*, *88*, 249-257.
[26] Hadi, A. A. A., & Al-Furat, A. A. (2018). Performance analysis of big data intrusion detection system over random Forest algorithm. *International Journal of Applied Engineering Research*, *13*(2), 1520-1527.
[27] Chen, F., Ye, Z., Wang, C., Yan, L., & Wang, R. (2018, September). A feature selection approach for network intrusion detection based on tree-seed algorithm and K-nearest neighbor. In *2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)* (pp. 68-72). IEEE.
[28] El Boujnouni, M., & Jedra, M. (2018). New Intrusion Detection System Based on Support Vector Domain Description with Information Gain Metric. *Int. J. Netw. Secur.*, *20*(1), 25-34.
[29] Peng, H., Ying, C., Tan, S., Hu, B., & Sun, Z. (2018). An improved feature selection algorithm based on ant colony optimization. *IEEE Access*, *6*, 69203-69209.
[30] Tao, P., Sun, Z., & Sun, Z. (2018). An improved intrusion detection algorithm based on GA and SVM. *Ieee Access*, *6*, 13624-13631.
[31] Yulianto, A., Sukarno, P., & Suwastika, N. A. (2019, March). Improving adaboost-based intrusion detection system (IDS) performance on CIC IDS 2017 dataset. In *Journal of Physics: Conference Series* (Vol. 1192, No. 1, p. 012018). IOP Publishing.
[32] Jiang, J., Wang, Q., Shi, Z., Lv, B., & Qi, B. (2018, March). RST-RF: a hybrid model based on rough set theory and random forest for network intrusion detection. In *Proceedings of the 2nd International Conference on Cryptography, Security and Privacy* (pp. 77-81).
[33] Singh, R. K., Dalal, S., Chauhan, V. K., & Kumar, D. (2019, March). Optimization of FAR in intrusion detection system by using random forest algorithm. In *Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE)*.

****

ISSN (Online):2583-0732