IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2320-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 10, Issue 6, June 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.512

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

Performance Analysis of Decision Tree Classifier in Network Intrusion Detection System

Harshitha S[1], Ramesh B[2]

Assistant Professor, Department of Computer Engineering, Malnad College of Engineering, Hassan, Karnataka, India[1]

Professor, Department of Computer Engineering, Malnad College of Engineering, Hassan, Karnataka, India[2]

ABSTRACT:The speedy upward push in the use of the internet, communication, and social networking increased the network size and corresponding data. So, many strange attacks are being generated. This posed challenges for network security in order to detect intrusions accurately. An Intrusion Detection System (IDS) tool can be used to ensure confidentiality, integrity, and availability by continuously inspecting network traffic to prevent the network from intrusions. The Intrusion detection system is facing various challenges to reduce false alarm rates and achieve high accuracy in detection. The Machine Learning (ML) and Deep Learning (DL)-based IDS systems can be used to enhance the accuracy rate of detection of various attacks in the network traffic. By learning the patterns of network traffic, the ML and DL technique tools predict the normal and abnormal activities. Machine Learning is the method of self-learning from various experiences without human intervention. Machine Learning techniques can be used in IDS to reduce false alarm rates (FAR) and enhance accuracy in detection. In this paper, we investigated and experimented the use of a decision tree machine learning classifier in a network intrusion detection system. And also, we analyzed the performance of the decision tree classifier under various factors.

KEYWORDS:Network Intrusion detection, Machine learning, Decision tree classifier, KDD database.

I. INTRODUCTION

The intrusion Detection system comprises of two words, 'Intrusion' and 'Detection'. Intrusion refers to unauthorized/illegitimate users trying to access the data within a computer or network by trading-off the confidentiality, integrity and availability which is known as CIA triad. So Detection system refers to a security technique for the identification of such illegal activities. The IDS will generate alerts to the host or network administrators when malicious behavior is detected. IDS are classified into three types, as shown in Fig 1.1.

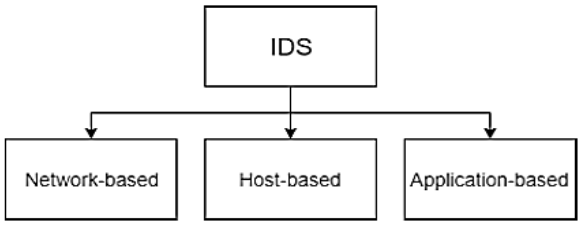

Fig: 1.1: Types of IDS

(i)Network-based IDS: A Network-based IDS (NIDS) typically uses sensors at various points on the network. The analysis of the traffic is either accomplished by the sensor itself or remotely by a central controller. Network intrusion detection system gathers information directly from a network. As packets travel in the network, it performs auditing on the attacks. This type of IDS grants users the privilege to specify its signature.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

(ii) Host-based IDS: A Host-based IDS (HIDS) is an agent installed on individual hosts, which analyzes their activity: files, processes, system logs, etc. So, Host-based IDS views the sign of intrusion in the local system. For analysis, they use the host system's logging and other information. Host-based handler is referred to as sensor.

(iii) Application-based IDS: An Application-based IDS is a special type of HIDS designed to monitor a specific application. Application-based IDSs analyze the interactions between users and applications: file executions or modifications, logs, authorizations, and any other potential abnormal activities. This will check the effective behavior and event of the protocol. The system or agent is placed between a process and a group of servers that monitors and analyzes the application protocol between devices.

Further IDS is subdivided into Signature-based intrusion detection (SIDS) and Anomaly detection-based intrusion detection (AIDS). SIDS, also known as misuse intrusion detection or knowledge-based intrusion detection, is based on the idea of detecting attack patterns by defining a signature. For attack detection, the data patterns are matched with the signatures stored in the signature database. This method achieves a high detection rate because of the availability of signatures for the known attacks. But on the other hand, this method fails to detect a very different and new type of attack without the signature patterns. This approach consumes a lot of resources as a database with signatures is compared with data packets for possible intrusions.

AIDS additionally known as the behavior-based IDS, is primarily based on the idea of clearly defining a profile for regular interest. Any deviation from this normal profile will be taken into consideration as an anomaly or bizarre behavior. The most important benefits of AIDS are its capacity to detect unknown and new assaults and the custom designed nature of the normal pastime profile for special networks and packages. However, the primary drawback is the high FAR as it's miles tough to discover the boundary among the ordinary and unusual profiles for intrusion detection. The classification of IDS is shown in Fig 1.2.

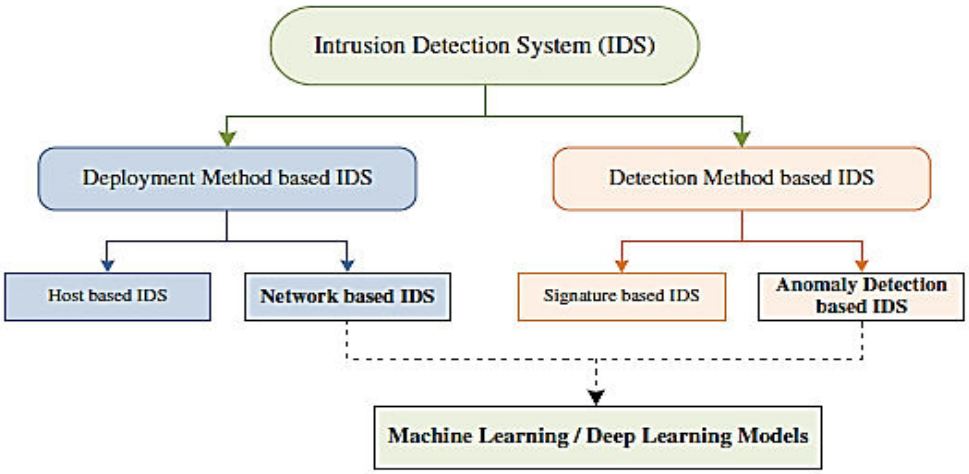

Fig: 1.2: Classification of Intrusion Detection System [1].

A NIDS evolved using ML and DL methods usually involves the following three major steps, that is, Data preprocessing phase, Training phase, and Testing phase. The dataset for the proposed solution is transformed into a suitable format to get applied with algorithms. This phase typically involves encoding and normalization. Removing entries with missing and redundant data is also performed in this step. The preprocessed data is then partitioned randomly into two parts, the training dataset and the testing dataset. Typically, the training dataset consists of 80% of the original dataset, and the remaining 20% forms the testing dataset.The ML and DL algorithm is then trained using the training dataset in the training phase. The size of the data set and the complexity of the proposed model impact the time taken by the algorithm in learning. A model is trained, it is tested using the testing dataset and evaluated based on the predictions made by it.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

II. LITERATURE SURVEY

Abhishek et al [2] discussed the statistical analysis and evaluation of labeled CIDDS data sets to detect anomaly based network intrusion detection system. Here k-nearest neighbour classification and k-means clustering are used to measure the difficulty . Based on the results it is observed that both k-nearest neighbor classifier and k-means clustering perform well over CIDDS-001 dataset. Vasilomanolakis et al [3] explained about the collaborative intrusion detection system (CIDS) which has several monitors acting as sensor to collect data. DARPA intrusion detection system datasets were used. They have applied neural network classifiers. This paper gives the gives an overview on the current state of the art in the area of distributed and collaborative intrusion detection. Buczak. [4] used clustering and Bayesian networks machine learning classifier to find out Cyber security intrusion detection. They have experimented cyber security detection on Defense Advanced Research Projects Agency (DARPA) 1998 ,1999,2000,KDD cup 99. Thomas [5] introduced neural networks concepts on Anomaly-based Network for Intrusion Detection Systems. They used NSL-KDD data set to process the network simulation. The new approach probability based Naïve Bayes classifier is applied on KDDcup 99, NSL-KDD data sets to detect Intrusion in a network [6].Yan Zhai et al[7] designed Bayesian network for find out the Potentially reduce False alarm rate in a network, for experimentation they used 1999 DARPA datasets. X.D. Hoang [8] proposed A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference using a hybrid fuzzy-based anomaly IDS using hidden Markov model (HMM) detection engine and a normal database detection engine is proposed to Reduce False Alarm Rate on TCP protocol datasets. The Application of Hybrid Neural Network Algorithms in Intrusion Detection System is introduced by Li Xiangmei [9], they implemented using Neural network model where Genetic algorithm and Learning model algorithm is combined to form hybrid neural network algorithm to Increased detection rate of the multiple classifiers intrusion detection system for KDD CUP 99 data base. Another approach on Naïve Bayes classifier was introduced by Mrutyunjaya [10] which performed better in terms of false positive rate, cost, and computational time on same KDD CUP 99 data sets . A unsupervised K-means classifier is implemented to Anomaly novel attacks detection Using an Evolutionary Extension method by T.Lunt [11]. Moon Sun [12] proposed False Alarm Classification Model for Network-Based Intrusion Detection System using Asoociation rule classification model, they used Darpa 1998 dataset. Mukkamala et al [13] used Neural networks, SVM classifiers to find out Intrusion Detection in KDD 99 and DARPA. Zhang Y et al [14] introduced Intrusion detection techniques for mobile wireless networks, they used Decision Tree (RIPPER) and SVM machine learning classifier, they run the experiments on NS-2 test scripts. An intrusion detection system based on combining probability predictions of a tree of classifiers is proposed by Ahmim et al [15], they used combination of the probability predictions of a tree classifiers, experiments identified High detection rate and low false alarm rate on KDD'99 and NSL-KDD data sets. Mohamed Amine et al[16] proposed Rules and Decision Tree-Based Intrusion Detection System for Internet-of-Things Networks, they got High accuracy, detection rate, Reduce false alarm rate and time overhead for CICIDS2017 dataset and BoT-IoT dataset.

From the above Literature survey, it is observed that not much work is investigated on Intrusion detection systems using a Decision tree classifier with a larger Dataset. In this paper, we used KDD Cup 1999 data sets with more than 20000 training samples, and 4693 test samples are considered for experimentation. And also, we explored the work by varying the data size for training the model. From the literature, it has been observed that the performance of the training models is not compared. In this paper, we compared the training model by considering various factors like accuracy, prediction speed, and time taken.

III. MACHINE LEARNING TECHNIQUES USED FOR NIDS

Machine learning (ML) techniques make our computing processes more efficient, reliable, and cost-effective. By analysing even more complex data automatically, quickly, and more accurately, ML can produce models. It is mainly classified into supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. The strength of ML lies in its ability to provide generalized solutions through an architecture that can learn to improve its performance. Because of its interdisciplinary nature, it plays a pivotal role in various fields, including engineering, medical, and computing. ML is a subset of AI that includes all of the strategies and algorithms which allow the systems to study mechanically the use of mathematical functions so that we can extract beneficial information from massive datasets. The common ML algorithms used for IDS are Decision Tree, K-Nearest Neighbour (KNN), Artificial Neural Network (ANN), Support Vector Machine (SVM), K-Mean Clustering, Fast Learning Network, and Ensemble Methods [1].

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

(i) Decision tree

DT is one of the basic supervised ML algorithms that's used for both classification and regression of the given dataset via applying the series of selections (rules). The model has a traditional tree structure with nodes, branches, and leaf. Each node represents an attribute/ feature. The branch represents a decision/rule, while each leaf represents a possible outcome or class label. The DT algorithm mechanically selects the first-class features for constructing a tree after which plays a pruning operation to eliminate beside the point branches from the tree to keep away from over-becoming. The most not unusual DT are CART, C4.Five, and ID3.

(ii) K-Nearest Neighbor

KNN is one of the best supervised ML algorithms that make use of the concept of "characteristic similarity" to predict a certain records sample. It identifies a sample based totally on its neighbors through calculating its distance from the neighbors. In the KNN set of rules, the parameter k affects the overall performance of the model. If the value of k may be very smaller, the model can be susceptible to over-fitting. While a totally massive choice of k cost may bring about misclassification of the sample instance.

(iii) Support vector machine

SVM is a supervised ML set of rules based totally on the idea of max-margin separation hyperplane in n-dimensional feature area. It is used for the solution of each linear and nonlinear issues. For nonlinear issues, kernel capabilities are used. The concept is to first map a low-dimensional input vector into a high-dimensional feature space the usage of the kernel feature. Next, an most desirable most marginal hyper-plane is acquired, which matches as a choice boundary the usage of the help vectors. For NIDS, the SVM algorithm may be used to beautify its performance and accuracy through successfully predicting the ordinary and malicious classes

(iv) K-mean clustering

Clustering is an idea of dividing information into meaningful clusters (or groups) by placing the pretty comparable information into the same cluster. K-Mean clustering is one of the popular centroid-based totally iterative ML algorithms that learn in an unsupervised way. K represents the wide variety of centroids (middle of the cluster) within a dataset. For assigning certain data factors to a cluster, normally, distance is calculated. The major concept is to reduce the sum of the distances between the information points and their respective centroids inside a cluster.

(v) Artificial neural network

ANN is likewise a supervised ML set of rules and is inspired by the running of the nervous system of the human brain. It is made of the processing elements called the neurons and the connection between them. These nodes/neurons are organized in an input layer, many hidden layers, and an output layer. The back propagation algorithm is used as a gaining knowledge of method for the ANN. The fundamental benefit of using an ANN method is its ability to perform nonlinear modelling via studying from larger datasets.

(vi) Naïve Bayes

The Naïve Bayes algorithm is based on conditional probability and the hypothesis of attribute independence. For every sample, the Naïve Bayes classifier calculates the conditional probabilities for different classes. The sample is classified into the maximum probability class. The conditional probability formula is calculated.

IV. EVALUATION METRICS

The most typically used assessment metrics for measuring the performance of ML and DL techniques for IDS. All the evaluation metrics are primarily based at the one of a kind attributes used within the Confusion Matrix[1], that's a -dimensional matrix presenting records approximately the Actual and Predicted elegance and consists of: True Positive (TP): The facts instances effectively expected as an Attack by way of the classifier. False Negative (FN): The data times wrongly expected as Normal instances. False Positive (FP): The facts times wrongly classified as an Attack. True Negative (TN): The instances efficiently classified as Normal instances.

The diagonal of the confusion matrix denotes the correct predictions, even as nondiagonal elements are the wrong predictions of a certain classifier. The table depicts these attributes of the confusion matrix. Further, the exclusive assessment metrics used in the latest studies are,

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

(i)Precision: It is the ratio of correctly predicted Attacks to all the samples predicted as Attacks.

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

(ii)Recall: It is a ratio of all samples correctly classified as Attacks to all the samples that are actually Attacks. It is also called a Detection Rate.

$$\text{Recall} = \text{Detection Rate} = \frac{TP}{TP+FN}$$

(iii)F-Measure: It is defined as the harmonic mean of the Precision and Recall. In other words, it is a statistical technique for examining the accuracy of a system by considering both the precision and recall of the system.
F-Measure = 2(Precision × Recall/Precision + Recall)
These evaluation metrics are calculated by testing the methodologies using benchmark datasets.

$$\text{F-Measure} = \frac{2*P*R}{P+R}$$

(iv)False-negative rate: is defined as the ratio of false-negative samples to total positive samples. In attack detection, the FNR is also called the missed alarm rate.

$$\text{False-negative rate} = \frac{FN}{TP+FN}$$

(v)False alarm rate: It is also called the false positive rate and is defined as the ratio of wrongly predicted Attack samples to all the samples that are Normal.

$$\text{False Alarm Rate} = \frac{FP}{FP+TN}$$

(vi)True negative rate: It is defined as the ratio of the number of correctly classified Normal samples to all the samples that are Normal.

$$\text{True Negative Rate} = \frac{TN}{TN+FP}$$

(vii)Accuracy: It is the ratio of correctly classified instances to the total number of instances. It is also called Detection Accuracy and is a useful performance measure only when a dataset is balanced.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

V. BENCHMARK DATASETS OF NIDS

This section provides detail about the popular datasets used by the researcher for testing the performance of their methodology[1]. Table 5.1 shows the list of benchmark datasets of NIDS available.

Table 5.1: Publicly available Benchmark Data sets for Network intrusion detection system [1].

Dataset	Year	Attack types	Attacks
KDD Cup'99[129]	1998	4	DoS, Probe, R2L, U2R
Kyoto 2006+[130]	2006	2	Known Attacks, Unknown Attacks
NSL-KDD[131]	2009	4	DoS, Probe, R2L, U2R
UNSW-NB15[132]	2015	9	Backdoors, DoS, Exploits, Fuzzers, Generic, Port scans, Reconnaissance, Shellcode, worms
CIC-IDS2017[133]	2017	7	Brute Force, HeartBleed, Botnet, DoS, DDoS, Web , Infiltration
CSE-CIC-IDS2018[133]	2018	7	HeartBleed, DoS, Botnet, DDoS, Brute Force, Infiltration, Web.

(i) KDD Cup'99: It is one of the maximum famous and extensively used datasets for IDS. It carries about 5 and two million facts for training and trying out, respectively. Each report contains 41 specific features or attributes and is

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

categorized as both normal and attack. The attacks are categorised into 4 different sorts as Denial of Service (DoS), Probe, Remote to Local (R2L), and User to Root (U2R).

(ii) Kyoto 2006+: This dataset is constructed from the network site visitors information, acquired by means of deploying honeypots, dark net sensors, e-mail servers, net crawler, and different network security features with the aid of Kyoto University. The present day dataset includes the traffic document from 2006 to 2015. Each record has 24 statistical features, 14 of which are derived from the KDD Cup'ninety nine dataset, even as the last 10 are extra capabilities.

(iii) NSL-KDD: This is the revised and subtle model of the KDD Cup' ninety nine dataset by means of doing away with several of its integral troubles. This dataset is likewise a 41 characteristic dataset with the assaults divided into four lessons as discussed in KDD Cup'99.

(iv) UNSW-NB15: This dataset is created by using the Australian Center for Cyber Security. It includes about million statistics with a total of forty nine capabilities that are extracted the use of Bro-IDS, Argus tools, and some newly developed algorithms. This dataset carries the forms of attacks named as Worms, Shell code, Reconnaissance, Port Scans, Generic, Backdoor, DoS, Exploits, and Fuzzers.

(v) CIC-IDS2017: This dataset is created through the Canadian Institute of Cyber Security (CIC) in 2017. It carries the regular flows and up to date real-world attacks. The network visitors is analyzed by means of CIC Flow Meter the usage of the statistics primarily based on timestamps, supply, and destination IP addresses, protocols, and attacks. Moreover, CICIDS 2017 consists of common assault eventualities like Brute Force Attack, Heart Bleed Attack, Botnet, Denial of Service (DoS) Attack, Distributed DoS (DDoS) Attack, Web Attack, and Infiltration Attack.

(vi) CSE-CIC-IDS2018: This dataset is jointly created by Communications Security Establishment (CSE) and CIC in 2018.The user profiles containing the abstract representation of the different events is created.

VI. METHODOLOGY

In this paper, we used the Decision tree algorithm, which is one among the Machine Learning classifiers. The simulated datasets KDD Cup 99 [16] were taken from the Military environment, which consists of raw TCP/IP dump data, which is considered as original Dataset. In the Data Pre-processing phase, the raw data was arranged in a proper suitable format to train the data samples using a decision tree classifier. In Training Phase, we considered training data samples with different counts in each model. We considered Model-1 with 20499, Model-2 with 15499, and Model-3 with 10499 data samples. All the models are trained and tested using a Decision Tree classifier. In the testing phase, we used 4693 samples without any class labels. The class labels were predicted by the decision tree classifier as normal or anomalous. The prediction results of each model are compared in terms of accuracy, speed, and the time taken for prediction. The model of NIDS using decision tree classifier is shown in Fig 6.1.

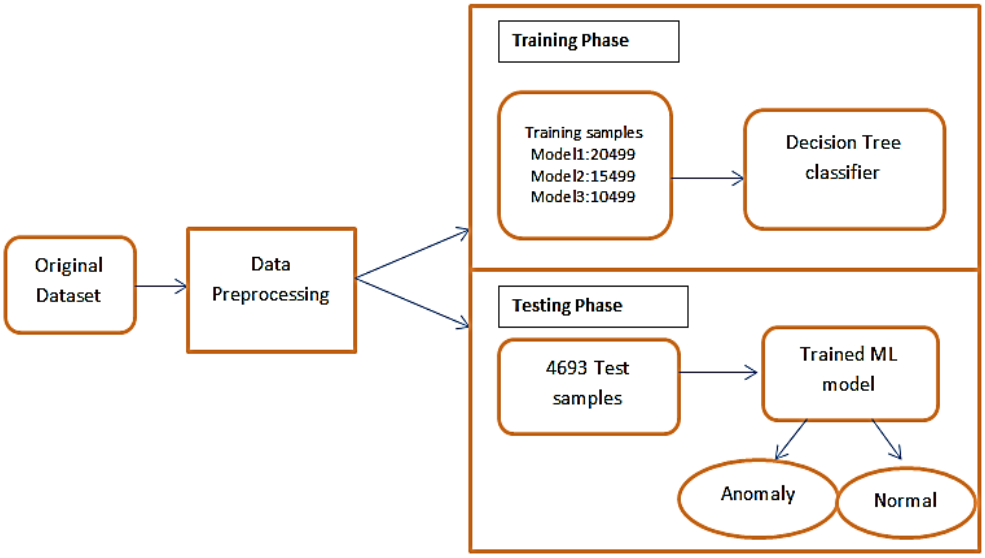

Fig 6.1: Model for NIDS using Decision Tree classifier

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

VII. EXPERIMENTS AND RESULTS

The KDD Cup 1999[16] dataset examined consists of a wide variety of intrusions simulated in a military network environment. A typical US Air Force LAN was used to create an environment to acquire raw TCP/IP dump data for a network. Each connection is labelled as either normal or as an attack with exactly one specific attack type.. The 41 quantitative and qualitative features are obtained from normal and attack data (3 qualitative and 38 quantitative features) for each TCP/IP connection. The class variable has two categories Normal and Anomalous.

The Model-1 was trained using 20499 data samples using a decision tree classifier and tested using 4693 samples where the attacks were correctly predicted and classified as normal and anomaly. Model-1 was achieved with an accuracy of 99.7%. The Model-2 was trained using 15499 data samples using a decision tree classifier and tested using 4693 samples where the attacks were correctly predicted and classified as normal and anomaly. Model-2 was achieved with an accuracy of 99.6%. The Model-3 was trained using 10499 data samples using a decision tree classifier and tested using 4693 samples where the attacks were correctly predicted and classified as normal and anomaly. Model-3 was achieved with an accuracy of 99.5%. The Confusion matrix for each model is shown in Fig 7.1.

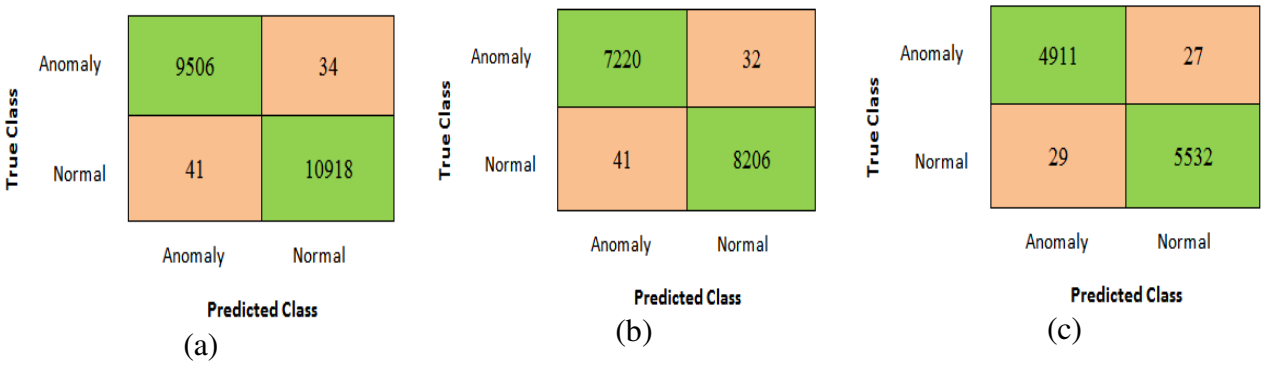

Fig 7.1: (a) Model1 with 20499 training data sets (b) Model2 with 15499 training data sets (c) Model3 with 10499 training data sets

VIII. COMPARISON STUDY OF EXPERIMENTED RESULTS

In this section three models which are investigated by varying training and testing samples are compared and analyzed based on the classification accuracy, time taken to train the model and prediction speed.

Case1: The Model-1 with 20499 data samples was trained using a Decision tree classifier using 4693 test samples and accurately predicted and classified intrusion at the accuracy rate of 99.7%. The time taken to train this model is 5.172 seconds, and the prediction speed is ~930000 obs/sec.

Case2: The Model-2 with 15499 data samples was trained using a Decision tree classifier using 4693 test samples and accurately predicted and classified intrusion at the accuracy rate of 99.6%. The time taken to train this model is 4.07 seconds, and the prediction speed is ~790000 obs/sec.

Case3: The Model-3 with 10499 data samples was trained using a Decision tree classifier using 4693 test samples and accurately predicted and classified intrusion at the accuracy rate of 99.5%. The time taken to train this model is 3.23 seconds, and the prediction speed is ~520000 obs/sec.

The comparison graph of all the models is shown in Fig 8.1, Fig 8.2, and Fig 8.3.

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

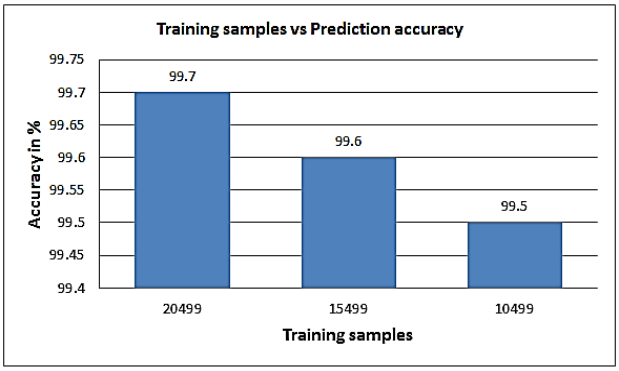

Fig 8.1 : Comparison of Training samples vs. Prediction accuracy

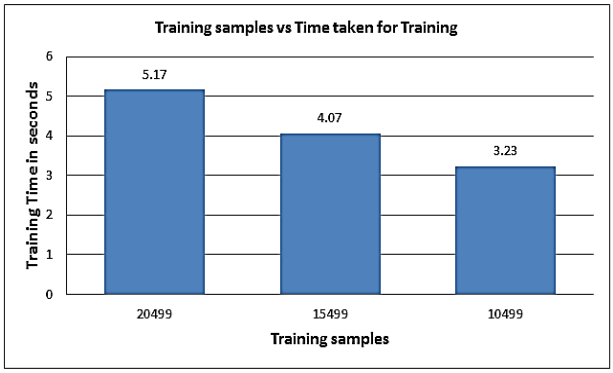

Fig 8.2: Comparison of Training samples vs. Time taken for Training

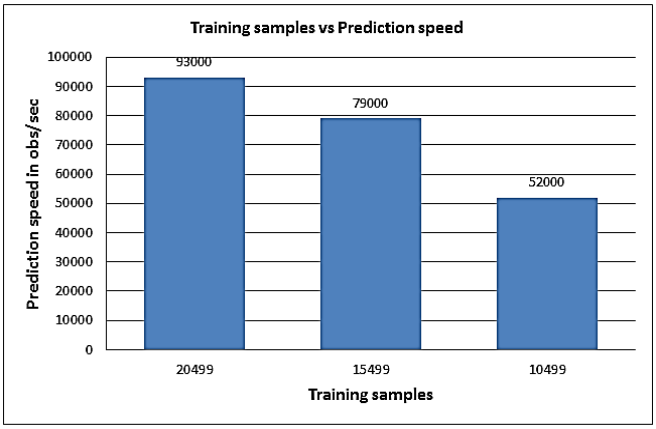

Fig 8.3: Comparison of Training samples vs. Prediction Speed

IX. CONCLUSION

The performance of Network Intrusion Detection systems can be enhanced using Machine Learning Techniques. In this paper, we have investigated a model for a network intrusion detection system using a Decision Tree classifier. The decision tree classifier showed good results when explored on KDD Cup 99 Datasets. In this paper, three models were designed and implemented by varying training data samples. From the investigation, it has been observed that the accuracy of the system can be improved by using more data samples while training the system. It has been identified that the prediction speed and training time can be reduced with the use of limited datasets for training. From the experiments, we can conclude that using the average number of training samples as implemented in the Model-2 experiment, the factors like accuracy, prediction speed, and training time can be improved. So while using a decision tree classifier in a Network intrusion Detection system, it is always better to consider an average number of data samples that is not too much or not too little.

REFERENCES

[1] Zeeshan Ahmad et al., "Network intrusion detection system: A systematic study of machine learning and deep learning approaches" Trans Emerging Tel Tech. 2021;32:e4150, https://doi.org/10.1002/ett.4150

[2] AbhishekVermaa, VirenderRangaa, "Statistical analysis of CIDDS-001 dataset for Network Intrusion Detection Systems using Distance-based Machine Learning", ProcediaComputer Science 125, (2018),

[3] EmmanouilVasilomanolakis et al.," Taxonomy and Survey of Collaborative Intrusion Detection" ACM Computing Surveys, Vol. 47, No. 4, Article 55, Publication date: May 2015

[4] Buczak AL et al.," A survey of datamining and machine learningmethods for cyber security intrusion detection".IEEECommunSurvTutor. 2015;18(2):1153-1176. https://doi.org/10.1109/COMST.2015.2494502

| e-ISSN: 2319-8753, p-ISSN: 2320-6710| www.ijirset.com | Impact Factor: 7.512|

|| Volume 10, Issue 6, June 2021 ||

| DOI:10.15680/IJIRSET.2021.1006384 |

[5] Thomas R, Pavithran D. "A survey of intrusion detection models based on NSL-KDD data set". Paper presented at: Proceedings of the 5th HCT Information Technology Trends (ITT). Dubai, United Arab Emirates: IEEE; 2018:286-291

[6] Liu H, Lang B." Machine learning and deep learning methods for intrusion detection systems: a survey". Appl Sci. 2019;9(20):4396. https://doi.org/10.3390/app9204396.

[7] Y. Zhai, P. Ning, P. Iyer, D.S. Reeves, "Reasoning about complementary intrusion evidence," in: Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC 04), December 2004

[8] X.D. Hoang, J. Hu, P. Bertok, "A program-based anomaly intrusion detection scheme using multiple detection engines and fuzzy inference," Journal of Net- work and Computer applications 32 (2009) 1219–1228.

[9] Li Xiangmei Qin Zhi "The Application of Hybrid Neural Network Algorithms in Intrusion Detection System "978-1-4244-8694-6/11 ©2011 IEEE

[10] Mrutyunjaya Panda, and ManasRanjanPatra " NETWORK INTRUSION DETECTION USING NAÏVE BAYES ", IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.12, December 2007

[11] T.Lunt and I.Traore, "Unsupervised Anomaly Detection Using an Evolutionary Extension of K-means Algorithm",International Journal on Information and computer Science, InderscienePulisher 2 (May, 2008), 107-139.

[12] Moon Sun Shin ,Eun Hee Kim ,Keun Ho Ryu," False Alarm Classification Model for Network-Based Intrusion Detection System" International Conference on Intelligent Data Engineering and Automated LearningIDEAL 2004: Intelligent Data Engineering and Automated Learning – IDEAL 2004

[13] Mukkamala S, Janoski G, Sung A." Intrusion detection using neural networks and support vector machines". Paper presented at: Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290). Honolulu, HI, USA: IEEE; vol. 2,2002:1702-1707.

[14] Zhang Y, Lee W, Huang YA."Intrusion detection techniques formobile wireless networks". WirelNetw. 2003;9(5):545-556. https://doi.org/10.1023/A:1024600519144.

[15] Ahmim A, Derdour M, Ferrag MA." An intrusion detection system based on combining probability predictions of a tree of classifiers". Int J Commun Syst. 2018;31(9):e3547. https://doi.org/10.1002/dac.3547.

[16] The UCI KDD Archive ,Information and Computer Science,University of California, Irvine.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor:
7.512

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर
NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details