Detailed Analysis of Intrusion Detection using Machine Learning Algorithms

Samriddhi Verma, Nithyanandam P.

Abstract*: The number of internet users has increased exponentially over the years and so have increased intrusive activities significantly. To detect an intrusion attack in a system connected over a network is one of the most challenging tasks in today's world. A significant number of techniques have been developed which are based on machine learning approaches to detect these intrusion attacks. Even though these techniques are good, they are not good enough to detect all kinds of attacks. In this paper, the analysis of different machine learning algorithm will be performed on the NSL-KDD dataset with pre-processing steps like One-hot encoding, feature selection and random sampling to use in different machine learning models to find the best performing model to detect these attacks. The attacks are from the datasets are classified into four types of attacks: Probe, DoS, U2R, R2L while the non- attack is the Normal. The dataset is in two parts: KDD-Train and KDD-Test. The dataset is trained and tested to find accuracy and understand the performance of different machine learning algorithms and compare them. The Machine Learning algorithms used are Naive Bayes Classifier, Decision Tree Classifier, Random Forest Classifier, K-Neighbours Classifier, Logistic Regression, SVM Classifier, Voting Classifier. These techniques are compared according to their capability to detect the attacks. This comparison will help to find the algorithm which would work the best to detect different kinds of intrusion attacks.

***Keywords*: Intrusion Detection, NSL-KDD, Supervised Learning, One-Hot Encoding, Feature Selection.**

I. INTRODUCTION

To define what an attack is, any action which threatens the confidentiality, integrity and availability is called an attack. The attacks generally focus on the vulnerabilities of a user on the network by unauthorized access to a system. The Internet has become a major part of our life like banking, shopping, social media, education, entertainment, business etc. For example, online shopping stores or e-commerce run on the internet where a person can buy and sell commodities by the means of Internet Banking. Confidential information such as payments option information is shared in the process. Apart from this, even personal information like national identity information is shared within people and group over the internet and threats to steal these is high as a matter of identity theft which can be devastating for all. Hence, the issue of network security is of the utmost importance for all internet users around the world.

To prevent such threats and attacks and detect any intrusive activities, security software companies developed an Intrusion Detection System (IDS). Intrusion Detection System works on the principle that if the behaviour of a normal user is different, it might be an intrusion attempt. An intrusion detection system uses a set of techniques to detect any suspicious activities on the network level and host level. Techniques like Support Vector Machines (SVM) or Artificial Neural Network (ANN) can be used to find and fetch information by the means of models which cannot be detected easily by human observation. On the other hand, learning mechanisms are combined with other learning mechanisms called hybrid techniques. These mechanisms are classifiers which classify the network data incoming into the system to decide whether the activity is an attack or some normal activity. Support Vector Machines is an effective proven technique concerning accuracy and overcome high dimensional data. To improve the intrusion detection system and reduce the false negative and false positive, which can be tested by the application of different algorithms. In this paper, Naive Bayes Classifier, Decision Tree Classifier, Random Forest Classifier, K-Neighbours Classifier, Logistic Regression, SVM Classifier and Voting Classifiers are used for training data and testing it. Feature Selection is performed using Recursive Feature Elimination using Random Forest to find the most important features and the extracted features are used to train the data as it is prominent that all features are not so important. This method of feature selection increases the efficiency and accuracy of the intrusion detection system. The redundant features make it difficult and tedious to detect intrusion. After performing feature extraction, the dataset is trained for different models and results are evaluated for different evaluation metrics.

II. DATASET

In this research, the NSL-KDD data set is used. This dataset is derived from the KDD 99' dataset after cleaning it. The data consists of traffic records to build an Intrusion Detection System and a model which could predict if there is an attack or an intrusion attempt or is it a normal connection. The dataset consists of internet traffic records captured by an intrusion detection system. These are considered the unseen records captured by a real intrusion detection system. There is a total of 43 attributes out of which 41 attributes are the traffic data and the other two are attack class (normal or an attack) and the other attribute is the score which denoted the severity of traffic data details. There are total 39 different kinds of attacks mentioned in the dataset which belong to four major categories of attacks called Denial of Service (DoS), User to Root attack (U2R), Probing attack, Remote to Local attack (R2L). The

Revised Manuscript Received on April 25, 2020.

***** Samriddhi Verma

Samriddhi Verma*, School of Computer Science and Engineering, Vellore Institute of Technology (VIT University), Chennai, India. E-mail: samriddhi160997@gmail.com

Dr. Nithyanandam P., School of Computer Science and Engineering, Vellore Institute of Technology (VIT University), Chennai, India. E-mail: nithyanandam.p@vit.ac.in

traffic data recorded by the
trusion detection system can be arranged into four categories namely Intrinsic, Content, Host-based, and Time-based. These can be described as:
Intrinsic features are derived from the header of the packet and have basic packet information.
Content features are the information about the original packets sent in more than one way. The payload can be accessed in with this information.
Time-based features consist of the analysis of the traffic input over a three-second window. Information like the number of connections was tied to the same host and counts the traffic input.
Host-based features are quite same as Time-based features where it analyses over a series of connections made. These features access attacks, which are longer than a two-second window. [7]

Table I Distribution of Internet Traffic Attacks in The Training Dataset

Attack Type	Records
No Attacks (Normal)	67343
Denial of Service (DoS)	45927
User to Root (U2R)	52
Remote to Local (R2L)	995
Probe	11656

Table II Distribution of Internet Traffic Attacks in The Testing Dataset

Attack Type	Records
No Attacks (Normal)	9711
Denial of Service (DoS)	7458
User to Root (U2R)	200
Remote to Local (R2L)	2654
Probe	2421

III. LITERATURE SURVEY

A. Classification

In a paper on Network Intrusion Detection using Machine Learning by Md. Nasimuzzaman Chowdhury and Ken Ferens and Mike Ferens, a combination of two machine learning algorithms are proposed to classify any anomalous behaviour in the network traffic. [1] This method shows how good the algorithm works to detect intrusion with high accuracy of 98.76% and a low false-positive rate of 0.09% and false-negative rate of 1.15%. SVM based scheme achieved a detection accuracy of 88.03% and a false-positive rate of 4.2% and false-negative rate of 7.77%.[2] The major part of network security is anomaly-based IDS. Sometimes the behaviour of the anomaly seems to be similar to normal data usage. One problem in anomaly detection refers to the issue of classification problem that how to make a distinction between normal and abnormal activities effectively and efficiently. In particular, support vector machines, neural networks, decision trees seem to have efficient significant schemes in anomaly detection systems to improve classification performance and speed. A new algorithm is proposed with the combination of Simulated Annealing & SVM. It can detect anomalous behaviour of the network and can classify them as normal or an attack. It doesn't require any hardware specifications and can be used for pattern recognition of malicious behaviour.

B. Neural Networks

A paper on Network IDS using machine learning by Saroj Kr. Biswas has taken a group of important features from the initial set of features using feature selection techniques, then the set of important features is selected to train different kinds of classifiers to build the Intrusion Detection System. Five folds cross-validation is implemented on the NSL-KDD dataset to seek out the results. It is finally observed that K-NN classifier produces better performance than others and, among the feature selection methods, information gain ratio-based feature selection method has proven to give good and improved results. Sannasi Ganapathy et al. presented a survey on intelligent techniques for Intrusion Detection by feature selection and classification techniques, which consists of many statistical and machine learning algorithms that are used as classifiers or feature selection techniques.[2] Vinchurkar et al. analyzed the NN and other machine learning approaches in designing an Intrusion Detection System (IDS). Jalil et al. compared the performance of machine learning algorithms in network intrusion detection and observed that Decision Tree gives better accuracy compared to SVM and Naïve Bayes. Amor et al compared two classifiers i.e. Naïve Bayes and Decision Tree. It was concluded that the Decision Tree performed better. It was found that some comparative studies were conducted in this field but the exhaustive study was not performed.[2]

C. Clustering

In the paper, Implementation of Machine Learning Techniques Applied to the Network Intrusion Detection System by B Ida Seraphim, Shreya Palit, Kaustubh Srivastava, E Poovammal is a comprehensive survey of some major techniques of machine learning implemented on intrusion Detection is presented. Techniques based on K-means, K-means with principal component analysis, Random Forest algorithm Extreme learning the machine, techniques, classification algorithms such as Naive Bayes algorithm, Hoeffding Tree algorithm. Algorithms like SVM, Deep Leaning for Auto-encoders, Accuracy-Updated Ensemble and Accuracy-Weighted Ensemble is implemented.[6] In this paper, a detailed survey of major techniques implemented on intrusion Detection is presented. Also, Accuracy Weighted Ensemble algorithm, Support Vector Machine, Genetic algorithm and Deep learning to use Autoencoders etc. have been implemented. A comparison of experimental values a system where we try to increase the efficiency of the parameters in the intrusion detection system using the two-level approach.[6] In Level 1, a comparison of basic supervised/unsupervised learning algorithm is performed and then in Level 2, the results are trained from level 1 in deep learning to use Artificial Neural Networks (ANN) and compare the parameters such Accuracy, Precision, Recall, False Alarm, F-score. Though we compare all the attack classes, in the

Artificial Neural Network model we were not able to take in consideration the U2R and R2L attack classes.

IV.ATTACK CLASS

There are in total 39 attack types which were categorized into 4 major attack classes- Probe Attack, DoS, U2R, R2L given below:

Probing Attack (Probe): When an attacker scans a network to gather information and vulnerabilities about the host. With a map of machines and services on the network, security controls are searched and exploited. Probe attacks threaten the computer's working features. It is one in each of the foremost attacks. It is the foremost common class of attacks,

e.g. ipsweep, Nmap, portsweep and satan.

Denial of Service (DoS): It is an attack during which an adversary directs a deluge of traffic requests to a system to form the computing or memory resource too busy or too full to handle legitimate requests and within the process, denies legitimate users' access to a machine. Types of DoS attacks are Smurf, Neptune, back, teardrop, pod and land.

User to Root Attack (U2R): It is a category of exploit during which the adversary starts with access to a traditional user account on the system (gained either by sniffing passwords, a dictionary attack, or social engineering) and is ready to take advantage of some vulnerability to get root access to the system. Examples of U2R attacks are buffer_overflow, rootkit, loadmodule and perl.

Remote to Local Attack (R2L): It occurs when an attacker who has the facility to send packets to a machine over a network but who doesn't have an account on it machine exploits some vulnerability to urge local access as a user of that machine. samples of R2L attacks are ftp_write, guess_passwd, warezmaster, multihop, phf, spy.

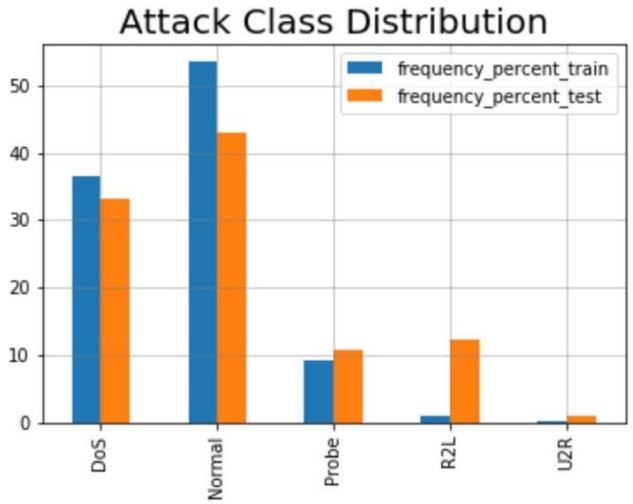

Figure 1. Training and Testing data distribution

Figure 1 shows the distribution of attack class in the training and testing dataset according to the percentage of frequency of data occurrence. After the 4 major attack classes, there is a Normal class which contains the data in which no attacks were detected by the intrusion detection system.

V. IMPLEMENTATION

The whole implementation is in five stages: Loading the **dataset**, pre-processing the dataset, feature selection and extraction, training the dataset and testing it. Now the result is analyzed.

A. Pre-Processing The Dataset

To pre-process the dataset, the NSL-KDD dataset is used which is derived from the KDD'99 dataset after cleaning it and removing the redundant records. The dataset is in two parts, the training set and testing set. The first step is to assign the column names to the dataset and then view them. This will help us to decide the pre-processing steps needed to use on data to make it fit for training. There are in total 39 different types of attacks which belong to DoS, Probe, R2L and U2R attack classes. So, in the attack class, all the attacks are mapped to these four categories to work on further. The attack column is mapped and changed to attack class. Next, exploratory data analysis is implemented and performed to get a brief overview of the statistical information of the dataset. Any column with all zero values is dropped. We need to understand the attack class distribution for the training and test set. For the training set, all the attack instances are counted and the percentage consistency of each attack class type is recorded. The same is done for the testing dataset. Next step is scaling the attributes and encoding them.

Scaling Numerical Attributes: The process of feature scaling is used to normalize the features of the dataset or the independent variable's range. We need to standardize the dataset so that the features are centered around 0 and have a standard deviation of 1. It is a general requirement for the machine learning algorithms as they might behave abnormally. A feature which has a high variance value as compared to other features, it would block the estimator to learn from other features. So, the centering and scaling happen exclusively for each feature by relevant statistics on the training dataset. The standard deviation and mean is stored and used for transform function. Since our dataset has categorical data too, the numerical attributes are extracted and scaled for training and testing dataset and converted to a data frame.

Encoding Of Categorical Attributes:

Encoding the categorical attributes means to convert whatever categorical text data exists in the dataset, it is converted to numerical data which can be understood by different models. Thus, we use the Label Encoder. The column values which are categorical text values are replaced by encoded data. A separate class column is created to which the encoded data is added for the target class which will be used to train data for the different classifiers.

Data Sampling: Data sampling is used when in a training set there is a sample in under-represented classes. The available samples are replaced by available samples using the process of random sampling. The new set is used to train the classifier instead of the original dataset. Considering the dataset used in this experiment, the attacks belonging to the attack classes R2L and U2R are extremely less in number in the training set, so to balance the classifiers, random oversampling is used to give good results. Since, in this experiment, the highest number of samples belong to the 'normal' or the non-attack class which is 67343, this is used in counter for random sampling and is applied to all the other attack classes' samples and

hence the final number of samples for each of the classes becomes 67343.

B. Feature Selection

Feature Selection is a technique which is used in machine learning to reduce the number of features by using predictive analytics and thus reduce teh dimension of the data. There are different methods to perform feature selection like the Embedded Method, the Wrapper Method and the Filter Method. In this experiment, the wrapper method is used to perform feature selection.

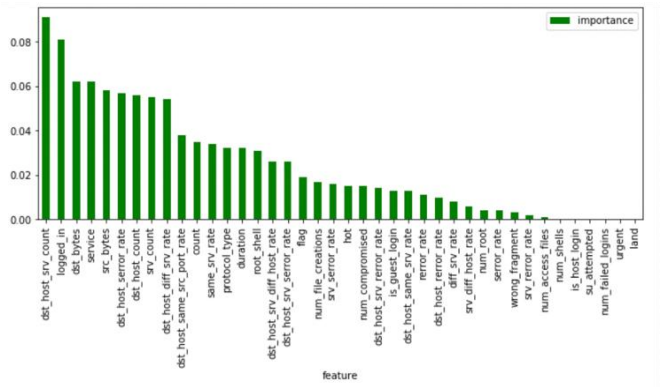

Figure 2. Features' Ranking

The Recursive Feature Elimination technique is a type of wrapper method which is used with the Random Forest Algorithm to perform feature selection. The Random Forest Algorithm is used to remove the least significant feature in iterations by fitting model until the desired strong features are achieved. The features are then ranked according to their importance by the feature importance attribute. Figure 2 shows the ranking of features based upon their importance in the NSL-KDD dataset. After performing this step, 10 attributes are selected based on their importance to train data.

Final Encoding Using One-Hot Encoder:

After performing feature selection, the training data is encoded using one-hot encoder even after using label encoder to avoid any hierarchal problems which means that if the classifier assumes that there exists a natural order among the categories which may cause abnormal results. This performs binarization of the data where the integer value provided by label encoder is replaced by a new binary variable is used to replace each unique integer value and includes it as a feature which is used to train the model [4].

C. Building Models

Support Vector Machine: It is used to solve challenges involving classification and regression. The data items are plotted as data points in an n-dimensional space (where n denotes the features' count in the dataset) with each coordinate's value being associated with the value of a particular feature.[5]Then, classification performed by finding the hyper-plane that differentiate the classes finely.[8] Finally, the number of support vectors are separated by a hyperplane.

K-Nearest Neighbour: K-nearest neighbours (KNN) is an example of a supervised machine learning algorithm used for regression and classification problems. When KNN is implemented for classification, the output is generally calculated as the class which has the highest frequency from the K-most similar instances. Each instance in essence votes for the class they belong to and the class with the highest votes is considered as the prediction. The normalized frequency of samples belonging to each class is a set of K most similar instances for a new data instance can also be used to calculate the class probabilities.

Logistic Regression: Logistic regression belongs to the category of the classification algorithm. It is used to consign observations to a discrete set of classes. The logistic sigmoid function is used by the logistic regression which transforms the output to return a value of probability. Within the classification problem, there exists a target variable 'y' which is the output, which can take discrete values only for a given feature set, also called inputs 'x'. In this implementation case, we are using the Binomial Logistic Regression where the output value is 0 (for attack) or 1(for normal). This model is easy and efficient with low variance. Although this model finds it difficult to handle categorical features well all the categorical data is encoded. Accuracy score of this model is calculated to evaluate the performance.

Naïve Bayes: Naïve Bayes classifiers are a family of regular "probabilistic classifiers" supported upon by applying this theorem with strong (Naïve) independence assumptions between the features. The basic assumption is that every feature is equal and independent. For this research work, the Bernoulli Naïve Bayes classifier is implemented because this classifier assumes that the features are binary so there exist only two values which is why the dataset is pre-processed and encoded to predict either 0 or 1 as classification result.

Decision Tree: Decision Trees (DTs) are supervised learning algorithm used for classification and regression. The data is used to learn and to approximate a sinusoid curve based on a collection of logical if-then-else decision rules. The deeper the tree, the more complex the decision rules become, the model becomes fitter too. In this algorithm, the instances are classified by sorting in the tree in a top to bottom approach, from the parent node to a child(leaf) node. An instance is classed by starting at the tree's parent node, testing the attribute specified by this node, then moving down the branch in correspondence to the value of that particular attribute. This whole process is repeated for all the branches or subtrees which are rooted at the new node [3]. The most important process is to spot the feature for the root node which is done by finding Information Gain or by finding the Gini Index. Entropy: It gives the measure of a random variable's uncertainty. The higher
value of entropy means high information content. It is a measure of purity.

$$\textbf{Entropy} = \sum i=1 - p_i * \log_2 * p_i$$

Equation 1. Entropy

Equation 1 is used to calculate the value of entropy. Here, pi denotes the probability of i's class.

Information Gain: The training data is divided into smaller sets by nodes, the entropy changes.

The change in entropy denotes the measure of Information Gain.

Information Gain = Entropy (Parent node) – average Entropy(children)

Random Forest: The random forest is a classification algorithm consisting of multiple decisions trees. It uses randomness of feature and bagging to build each tree to build an uncorrelated collection of trees (called forest) The prediction of the forest is considered to be more accurate than an individual tree. The working of the Random Forest algorithm can be explained as: whenever we train data, every tree learns data points from random samples. These samples use bootstrapping meaning that samples can be used multiple times by a single tree. The test features are taken and rules of randomly created decision tree to predict the result and store the output. The votes are calculated for each predicted target and whichever target is the highest voted, it becomes the final prediction.

Voting Classifier: The Ensemble Vote Classifier is a classifier for combining similar or conceptually different machine learning classifiers for classification with the help of majority or plurality voting. (In simpler terms, both majority and plurality voting is referred as majority voting). This Classifier implements voting based on two techniques namely soft and hard voting. In hard voting, the final class prediction is the class which was predicted the greatest number of times by different classifiers taken together as an ensemble. In soft voting, the final class label is predicted after considering the average of the different classifiers' class probabilities and whichever average is the highest in the final prediction.

Hard-Voting Classifier: It takes the aggregate of each classifier's prediction and predicts the class with the highest votes. This is also called as majority-voting.

Soft-Voting Classifier: In this model, all the classifiers can estimate the class probabilities, then Scikit-Learn is used to predict the class with the highest probability, averaged over all the individual classifiers.

VI.RESULTS ANALYSIS

A detailed study has been conducted based on the performance of different classifier models such as Naïve Bayes Classifier, Decision Tree Classifier, Random Forest Classifier, KNN Classifier, Logistic Regression, Support Vector Machine and Voting Classifier for Soft and Hard voting techniques. Various evaluation parameters have been considered for the same. Table III shows the accuracy and precision of the models on the test dataset. Table IV shows the models' performance based on sensitivity and specificity and Table V contain the record of False-Positives and False-Negatives of the test dataset.

Table III Accuracy And Precision Performance

Model Name	Accuracy	Precision
Naïve Bayes	0.84300	0.82159
Decision Tree	0.81647	0.80955
Random Forest	0.79066	0.78894
KNN	0.83976	0.81244
Logistic Regression	0.84105	0.83762
SVM	0.83324	0.81076
Voting-Hard	0.84233	0.83647
Voting- Soft	0.83534	0.81387

Table IVSensitivity and Specificity Performance

Model Name	Sensitivity	Specificity
Naïve Bayes	0.92287	0.73907
Decision Tree	0.88332	0.72941
Random Forest	0.85995	0.70045
KNN	0.93182	0.71989
Logistic Regression	0.89187	0.77487
SVM	0.91988	0.72043
Voting-Hard	0.89651	0.77178
Voting- Soft	0.91906	0.72633

Table V False-positives and False-negatives

Model Name	False Positive	False Negative
Naïve Bayes	1946	749
Decision Tree	2018	1133
Random Forest	2234	1360
KNN	2089	662
Logistic Regression	1679	1050
SVM	2085	778
Voting-Hard	1702	1005
Voting- Soft	2041	786

The results from Table III, IV and V are summarized as:

Accuracy: Among all the models, K-Neighbours Classifier has the highest testing accuracy that is 86.66. Even though they have the same model accuracies, other parameters of evaluation differ. Random Forest Classifier Model has the least model accuracy score of 80.67 among all the models.

False Positive: Logistic Regression Model has the lowest False Positive value of 1671 while Naive Bayes Classifier has the highest FP value of 1971.

False Negative: K-Neighbours Classifier Model has the lowest FN value of 619 while Random Forest Classifier Model has the highest value of FN i.e. 1602.

Sensitivity: K-Neighbours Classifier Model has the best sensitivity score of 93.62 while Random Forest Classifier has the least score of 83.50.

Specificity: Logistic Regression Model has the highest specificity score of 79.95 which means it classified all the non-attacks correctly while Voting Classifier: Hard voting Model has the second-best score of 79.57. Naive Bayes Classifier has the least score of 73.57.

Precision: Logistic Regression Model has the highest precision score of 83.76 while Voting Classifier: Hard voting Model has the second-best score of 83.64. Random Forest Classifier has the least score of 78.89.

The results are visualized in figure 3 which shows a brief comparison of the models based on various evaluation parameters.

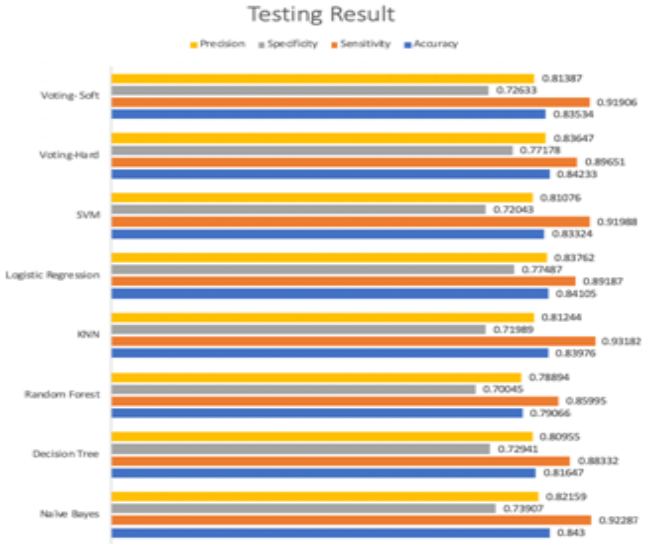

Figure 3. Test Data Results

VII. CONCLUSION

In this paper, eight different types of classification methods were applied using seven different machine learning models namely- Naïve-Bayes, Decision Tree, Random Forest, K-Nearest Neighbours, Logistic Regression, Support Vector Machine and Voting Classifier using Soft Voting method and Hard Voting method for the intrusion detection system. The performance of all these machine learning models were observed and compared based on different standard evaluation parameters such as Accuracy, Precision, False-Positive and False-Negative of the test data. The classifiers were able to handle such a volume of high dimensional data which satisfactory accuracy results. It was observed that the Voting Classifier algorithm using the Hard-voting method performed better than other models producing an accuracy of 84.233% and a precision of 0.83647. The number of False-Positives and False-Negatives summed up to 2707 which is better than most of the classifiers used. This classifier takes considerable time to train as well as test the dataset which is more than all the classifier except Support Vector Machine, which takes the most time. Apart from this, Logistic Regression and Naïve Bayes perform well out of all producing good accuracy result and lower False-Positives and False-Negatives.

The future works can include the results produced by all the classifier can be considered to make rules for the Intrusion Detection System. An example is to use Snort which is a Network Intrusion Detection Software. The work conducted in this research proves that the result produced by the machine learning techniques can be used to develop good intrusion detection system. Snort can be used to verify and validate the results. This work can be used further to develop efficient Intrusion Detection System using various techniques like that of Soft Computing.

REFERENCES

1. Md Nasimuzzaman Chowdhury and Ken Ferens, Mike Ferens, “Network Intrusion Detection Using Machine Learning” in 2016 Int’l Conf. Security and Management, SAM’16.
2. Saroj Kr. Biswas, “Intrusion Detection Using Machine Learning: A Comparison Study” in International Journal of Pure and Applied Mathematics, Volume 118 No. 19 2018, 101-114
3. Ben Amor, N., Benferhat, S., and Elouedi, "Z.: NB vs DTs in Intrusion Detection Systems", ACM Symposium on Applied Computing, pp. 420-424, (2004)
4. Jason Brownlee, “Why One-Hot Encode Data in Machine Learning” [Available]: https://machinelearningmastery.com/why-one-hot-encode
5. -data-in-machine-learning/
6. Priyankur Sarkar, “Support Vector Machines in Machine Learning” [Available]:https://www.knowledgehut.com/blog/data-science/support- vector-machines-in-machine-learning
7. B Ida Seraphim, Shreya Palit, Kaustubh Srivastava, E Poovammal, “Implementation of Machine Learning Techniques Applied to the Network Intrusion Detection System” in International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958, Volume-8 Issue-5, June 2019 Mill Valley, CA: University Science, 1989.
8. Gerry Saporito, “A Deeper Dive into the NSL-KDD dataset”, [Available]: https://towardsdatascience.com/a-deeper-dive-into-the-nsl- kdd-data-set-15c753364657
9. James Le “Support Vector Machines in R”, [Available]: https://www.datacamp.com/community/tutorials/support-vector-machi nes-r

AUTHORS PROFILE

Samriddhi Verma is a final year student pursuing undergraduate BTech. degree in Computer Science and Engineering student at the Vellore Institute of Technology (VIT University), Chennai campus who will graduate in 2020. She is going to pursue a Masters’ degree in the field of Data Science from the Warwick Business School in the class of 2020/21’. Her work focuses on handling huge volumes of data and performing analytics. She is experienced in coding using languages like C, C++, Java, Python, R and SQL. Her interests lie in the field of Applied Machine Learning, Data Analytics, Network Security, Predictive Analytics and Business Studies.

Dr. Nithyanandam Pandian is working as a Professor in the School of Computer Science and Engineering at the VIT University, Chennai Campus. He is an academician who has an experience of 18 years. He received a B.E. degree in Computer Science and Engineering from Madurai Kamaraj University in the year 2000, MTech in Computer Science and Engineering from BITS Pilani in 2003 and PhD inComputer Science and Engineering from Anna University, Chennai in 2013. He has published many national and international publications to his credit. He is also a reviewer for various leading journals such as IEEE, Elsevier, Inderscience, etc.