Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 01, January-2023

Enhancement of Intrusion Detection System using Machine Learning

Mukesh Kumar Yadav[1], Mahaiyo Ningshen[2]
Central Research Laboratory Bharat Electronics Limited Ghaziabad,
India

***Abstract*—The widespread use of internet in recent years has resulted in an exponential growth in the amount of information transmitted across different devices, as well as the number of novel methods of network attack. Many polls demonstrate that intrusion has been steadily increasing, which leads to personal information and privacy theft. Traditional intrusion detection system (IDS) approaches, such as firewalls, which rely on data filtering, may not be capable of detecting all sorts of attacks in real-time. Machine learning based intrusion detection system is particularly useful in effectively processing the enormous data, detecting any harmful behavior, and efficiently controlling and promptly identifying any attacks of such sorts. The detection system is used to predict the four different types of attacks namely Denial of Service (DOS), Probe, Remote to Local (R2L), and User to Root (U2R) attacks. This paper proposed an ensemble model which enhances the performance of IDS. The chi-squared feature selection method selects the attribute of the NSL-KDD dataset which are more dependent on the class label. We have used performance parameters such as Accuracy, Precision, Recall, and F1-Measure for evaluating the performance of the models. The experiment result reveals that the ensemble model which is AdaBoost with Logistic Regression performs better than all other models which are discussed in this paper. The paper also compares the proposed model with other relevant research papers. This suggested IDS has better performance than the existing state of the art. In the end, the challenges and the future scope are discussed briefly.**

***Index Terms*—Machine Learning, Datasets, Feature Selection, Machine Learning algorithms, Intrusion Detection System.**

I. Introduction

In this modern era, the internet has been playing an essential role in everyone’s daily life because it provides useful information on a wide range of topics, including business, education, and entertainment. Network attacks are also on the upsurge because of the widespread use of the internet. Intrusion detection systems and firewalls are just a few of the methods that have been proposed to counter these attacks. Firewall filters all incoming and outgoing packets based on predefined rules, while IDS just examines the network and delivers an alert message to the network administrator if any harmful activities are detected [1]. When compared to firewalls, intrusion detection system is more secure and performs better [2].

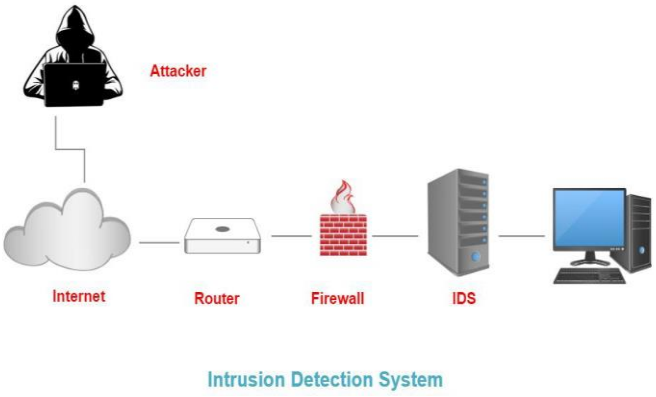

Figure 1. Intrusion Detection System

An intrusion detection system monitors the functions of firewall and routers as demonstrated in figure 1. The intrusion detection system (IDS) assists users in recognizing harmful activities. If a security breach is identified, the IDS alerts the network administrator. Based on the detecting mechanism, IDS may be divided into two categories. The first type is signature-based [3] and as the name suggests, in such types of detection systems look for signatures or distinct patterns which can be used to identify data that are harmful. This method is most suitable for attacks which are known and have no false alarms. The second is anomaly-based detection [4] which uses machine learning approaches to compare the previously reported free real-time malicious traffic against the actual real-time traffic. Such systems are capable of detecting unknown attacks. These are the most popular sorts of systems.

Machine learning algorithms are applied in intrusion detection system to reliably identify various kinds of attacks, reduce false alarm rates, and enhance detection rates [5]. Numerous intrusion detection systems have emerged using several machine learning algorithms since machine learning algorithms are regarded one of the most effective techniques for detecting malicious activities or attacks in a timely and accurate manner. Some research [6] gives detailed insights on algorithms based on single machine learning like Decision Tree, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM). Furthermore, other research uses an ensemble machine learning method that employs many categorization algorithms, such as Random Forest [7].

The main aim of the research is to propose an ensemble learning model which combines all the weak classifiers and creates a strong classifier which can detect different types of attacks more precisely. This paper uses the NSL-KDD dataset and some common supervised and unsupervised learning algorithms like- Support Vector Machine, Logistic Regression, K-Means Clustering, etc. Logistic Regression with Adaboost is the proposed ensemble model which boosts the performance of IDS.

The paper is organized as follows

1. Section II demonstrates about the various types of attacks and their broad categories.
2. Section III discusses about the background and the related work.
3. Section IV explains about the proposed approach and the experiment results.
4. Conclusion and future scope are presented in section V.

II. Network Attacks and their types

Network attacks are defined as an attempt to gain or perform an unauthorized action to an organizational network with the goal of looting data or carrying out other harmful activities. Network attacks are divided into two categories: passive attack and active attack [8]. During passive attack, the attackers intercept the network and monitor or obtain confidential details without altering it. Release of message contents and analysis of traffic are examples of passive attacks. In active attack, the attackers get illegal access and further modify, delete, encrypt and decrypt the data. Active attacks include message modification, repudiation, service denial, replay, and masquerade. It is very hard to detect passive attacks since they have no effect on the data or device. IDS perform a significant part in detecting various forms of attacks. Any attack, whether passive or active or any one of the attacks which fall in the following categories can be considered.

- Denial of Service (DOS): Here, the network is filled with unusable traffic by intruders such that the resources are kept busy and users are prevented from using the network. Land, Back, and Mail Blood Smurf attacks are examples of DOS attack.

- Probe attack: It makes use of a software/program to monitor or collect information about the network activity. Satan, Ipsweep, Mscan, Saint, and Nmap are examples of these attacks.

- Remote to Local (R2L): Here, an intruder can transmit packets via certain devices but does not have access to the device’s authorized account. In this situation, the intruder often exploits any weakness to get access to the device as a user. Named, Phf, Sendmail, and Guest are the examples of this type of attack.

- User to Root (U2R): An attacker has gained access to the user and is attempting to get superuser benefits. Perl, Ps, Eject, and Ffbconfig are examples of this class.

III. Related Work

In the domain of intrusion detection, many researchers have experimented with machine learning techniques and used the public NSL-KDD data set to enhance the detection rates [9]. The different IDS models that will be discussed in this chapter are developed using machine learning, feature selection, and ensemble-based techniques. Various machine learning algorithms are used to built the intrusion detection system. Some are supervised and some are unsupervised.

Various detection of intrusions methods based on supervised machine learning algorithms such as Gaussian Naive Bayes, Random Forest, Support Vector Machines and Logistic Regression (LR) are analyzed by Manjula C. Belavagi et al. [10]. Data are pre-processed to divide it into training and testing sets. Random Forest Classifier outperformed the other methods whereas the Support Vector Machine under-performed the other methods. They also mentioned that multi-class classification may improve the performance of the system. The intrusion detection [11] based on K-Means and Genetic K-Means algorithm are studied by Anand Sukumar J V et al. They applied addition or multiplication or reciprocal operation to reduce the data set so that the runtime can be minimized. K-Means gave better performance when smaller subset of data are used but Genetic K-Means algorithm achieved the highest accuracy when it comes to larger set of data. A model which used Information Gain ratio as a feature selection technique was proposed by S. Krishnaveni et al. [12]. Here, various machine learning techniques such as Radial Basis Function (RBF) SVM, Linear SVM, K-Nearest Neighbor and Logistic Regression are implemented. RBF SVM achieved the highest accuracy than the other methods. The main advantage of the model was that it was able to generate low false alarm rate. Mr. Subhash Waskle et al. [13] set up a model with PCA and Random Forest technique on the KDD data set. The model was compared with Support Vector Machine, Naive Bayes, and Decision Tree. This model obtained better accuracy than the other techniques. The benefit of this model is that the error rate is very low.

Decision Tree and K-Nearest Neighbor algorithm based intrusion detection method was used by Ashwini Pathak et al. [3]. The univariate feature selection with ANOVA F-test is performed to find the appropriate features. Both the algorithms are compared and it is found that the Decision Tree performed better than K-Nearest Neighbor. Further, the performances of machine learning techniques against various attacks are also demonstrated. Shadi Aljawarneh et al. [14] come with an ensemble learning technique and proposed Information Gain Ratio as a feature selection method. They implemented J48, Naive Bayes, Support Vector Machine, and the proposed model. The proposed model is an ensemble of J48, Meta Pagging, Random Tree, REPTree, AdaBoostM1, DecisionStump and Naive Bayes. The ensemble model obtained the highest accuracy than the other models. The merit of the model is that it has high accuracy and low detection time.

Sumaiya Thaseen et al. [15] proposed a model in which they used the chi-square feature selection method to select the features which are most dependent on the target class. They fixed a threshold value of 0.55 such that all features which fall above the value of 0.55 are selected as the optimal subset. They analyzed the machine learning techniques on 31 as well as on 41 features of the data set separately. The advantage of the model is that it performed better on multi-class classification as well. Amar Meryem et al. [16] applied K-Nearest Neighbor, Naive Bayes, Logistic Regression, and Support Vector Machine. K-Nearest Neighbor performed better on binary as well as on multi-class classification when compared with other techniques. The model has a high detection rate and low error rate. N. Kaja et al. [17] studied Random Forest, J48, Adaptive Boosting, and Naive Bayes. The over-fitting problem was removed from the dataset using a four-step of data pre-processing method. Out of these four techniques Random Forest achieved the highest accuracy but the computational time is higher than J48 and Naive Bayes. M. C. Belavagi et al. [18] employed a model using a multi-class classifier. They implemented Random Forest, Logistic Regression, Gaussian Naive Bayes, and Support Vector Machine. Decision Tree classifier technique is used to select the ten best features of the data set. Random Forest shows very good performance in identifying DOS, Probe, and U2R attacks, whereas the performance of all the algorithms is poor towards the identification of R2L attacks. Selvakumar et al. [19] employed a model using Decision Tree (C4.5) and Bayesian Network (BN) on the NSL-KDD dataset. They applied filter-based and wrapper-based feature selection methods. The NSL-KDD dataset consists of 41 features. The experiment was performed on 41 as well as on 10 features of the dataset separately. The Decision Tree performed better on DOS and R2L attack while Bayesian Network performed better on Probe and U2R attack. The merit of the model is that it has a high detection rate and the demerit of the model is that it has high computational time.

IV. Proposed Approach

In this work, we present an intrusion detection system using supervised and unsupervised learning. An ensemble model is proposed which is AdaBoost with Logistic Regression. The architecture of the proposed approach is explained in figure 2. This section is divided into four parts. Section I discusses about the used data set. Section II explains about the feature selection method. Various machine learning algorithms are explained in section III. The fourth one is performance metrics followed by the experiment results in section five.

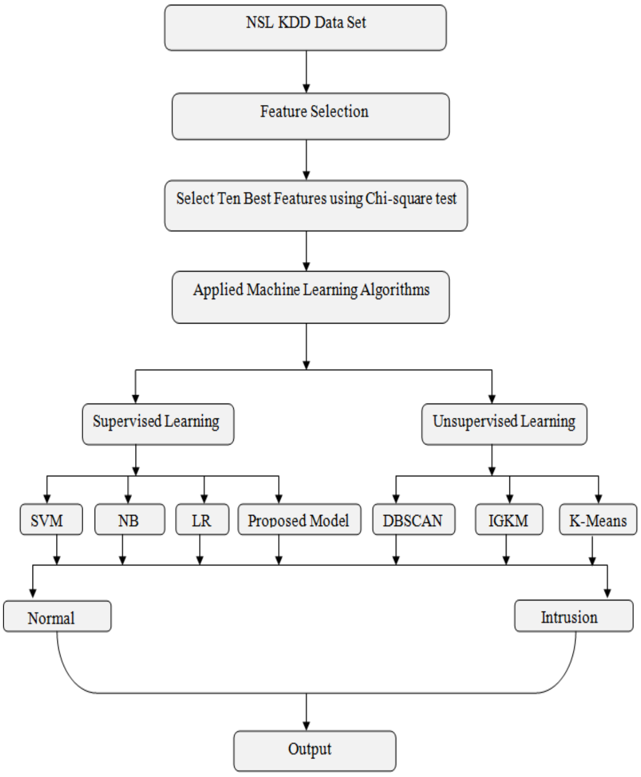

Figure 2. Structure of Intrusion Detection Model

A. Description of the used Dataset

In any machine learning algorithm and intrusion detection system, data played an important role. Data sets are necessary for evaluating and validating the performance of intrusion detection systems. [20]. The datasets are often divided into two sections: one for training and one for testing. The set of training data are the actual data that are used to train the models and the set of testing data are the input that are needed to instruct the models to carry out various operations. The KDD dataset captures the best explanation for numerous intrusions or attacks. The newer version of KDD CUP99 data set is NSL KDD [3] that is implemented in this paper [21]. It has limited number of records with fewer redundancies. The testing and training data set contains a large number of records, making evaluation easy and eliminating the need to choose specific data from it. The dataset contains 42 features, 41 of the features referring to the traffic input itself and the last feature is the class label or the target value i.e, Attack. The dataset contains four different types of attacks: Denial of Service (DOS), Probe, Remote to Local (R2L), and User to

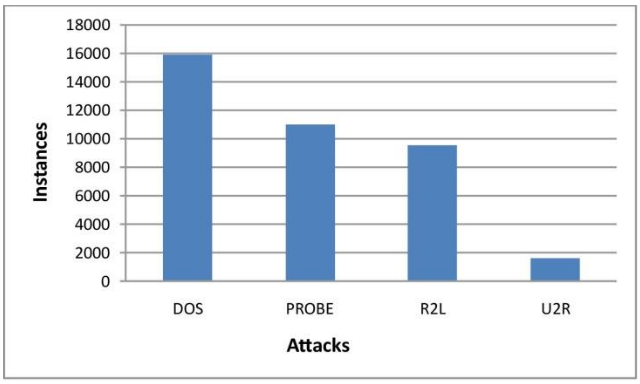

Figure 3. Instances in Training Dataset

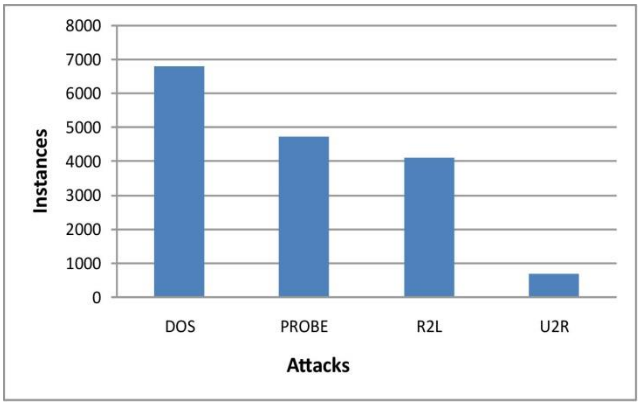

Figure 4. Instances in Testing Dataset

Root (U2R). NSL KDD dataset description is given in table 2. Figure 3. represents the instances in the training data set and figure 4. represents the instances in the testing data set. DOS has maximum instances in the training as well as in the testing data set whereas U2R has the least number of instances in both training and testing sets.

B. Feature Selection

In machine learning, feature selection is necessary in data pre-processing. It eliminates all unnecessary elements and retains only the necessary attributes. The importance of any attribute is reduced or eliminated if it does not help anticipate the target value [3]. Chi-square feature selection method is implemented in this research model because it's performs better on multi-class classification [22]. The chi-square test [15] is used to determine the best feature for a given dataset by determining the feature on which the output class is most dependent. The Chi-square test formula is explained in equation 1.

$$X^2 = \sum_{i=1}^{M} \sum_{j=1}^{K} \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \qquad (1)$$

Where X^2 is the Chi-square statistic, M is the attribute value and the output has K class labels. O_{ij} is the observed frequency. E_{ij} is the expected frequency. The X^2 value is calculated for all the features of the dataset and then arranged them in descending order.

Table I
DESCRIPTION OF THE SELECTED FEATURES

S.No	Feature	Description
1.	Src-bytes	The amount of data (bytes) transmitted from source to destination
2.	Diff-srv-rate	% of connections to the different service
3.	Same-srv-rate	% of connections to the same service
4.	Flag	Status of flag
5.	Service	Type of Network
6.	Dst-bytes	The amount of data (bytes) transmitted from destination to source
7.	Count No.	Total no. of connections made to the same host in the last two sec
8.	Dst-host-same- srv-rate	Proportion of connections with the same destination host and service
9.	Dst-host-diff-srv-rate	Proportion of connections on the current host that use a different service
10.	Logged-in	Logged-in=1 if user is logged in, otherwise logged-in=0

Higher the value of X^2, the more dependent the output label is on the feature. The top ten features of the dataset are selected using the chi-square feature selection method as shown in table 1. The top ten features of the data set are selected to avoid the over-fitting problem and to reduce the redundancy from the dataset.

C. Introduction of Implemented Machine Learning Algorithms

Machine Learning algorithms are the advancement of conventional algorithms [23]. Such algorithms allow systems to automatically learn themselves from data and make them smarter. Because of their learning and classification skills, these algorithms are currently employed in practically every industry to handle a wide range of issues. These algorithms are primarily categorized into supervised or unsupervised. In the section, we will go over several relevant machine learning approaches for detecting and classifying network attacks using IDS.

1) Supervised Learning: In supervised learning [24], the data are split into two sets, one is the training set and the other is the testing set. Training set data are used to train the model and the testing set are used for input in that model as shown in

Table II
DESCRIPTION OF THE NSL KDD DATA SET

S. NO.	*Feature*	**Definition**
1.	Duration	Connection’s length (in sec)
2.	Src-bytes	The amount of data(bytes) transmitted from source to destination
3.	Dst-bytes	The amount of data(bytes) transmitted from destination to source
4.	Land	Land=1 if the connection belongs to the same host, otherwise 0
5.	Wrong-fragment	Total no. of wrong fragments
6.	Urgent	Total no. of urgent messages
7.	Hot	Total no. of hot symbols
8.	Num-failed-logins	The total no. of unsuccessful login attempts
9.	Logged-in	Logged-in=1 if user is logged in, otherwise logged-in=0
10.	Num-compromised	The no. of conditions that have been compromised
11.	Root-shell	Root-shell=1 if root shell is generated, otherwise 0
12.	Su-attempted	Su-attempted=1 If su root attempted, otherwise 0
13.	Num-root	Total no. of connected roots
14.	Num-file-creations	The total no. of file created
15.	Num-shells	Total no. of shell prompt
16.	Num-access-files	Total no. of operations performed on access files
17.	Num-outbound-cmds	Total no. of outgoing commands
18.	Is-host-login	Is-host-login=1 If host is login, otherwise 0
19.	Is-guest-login	Is-guest-login=1 If guest is login, otherwise 0
20.	Count No.	Total no. of connections made to the same host in the last two sec
21.	Srv-count.	Total no. of connections made to the same service in the last two sec
22.	Serror-rate	Proportion of connections with a syn error
23.	Srv-serror-rate	Proportion of connections with a syn error
24.	Rerror-rate	Proportion of connections with a rej error
25.	Srv-rerror-rate	Proportion of connections with a rej error
26.	Same-srv-rate	Proportion of connections to the same service
27.	Diff-srv-rate	Proportion of connections to the different service
28.	Srv-diff-host-rate	Proportion of connections to the different hosts
29.	Dst-host-count	The total no. of connections to the same destination host
30.	Dst-host-srv-count	The total no. of connections to the same destination host and service
31.	Dst-host-same-srv-rate	Proportion of connections that have the same destination host and service
32.	Dst-host-diff-srv-rate	Proportion of connections on the current host that use a different service
33.	Dst-host-same-src-port-rate	Proportion of current host connections with the same source port
34.	Dst-host-srv-diff-host-rate	Proportion of connections of same service and different hosts
35.	Dst-host-serror-rate	Proportion of current host’s connections with serror
36.	Dst-host-srv-serror-rate	Proportion of serror connections on the current host of a service
37.	Dst-host-rerror-rate	Proportion of current host connections with an rst error
38.	Dst-host-srv-rerror-rate	Proportion of current host of service connections with rst error
39.	Protocol-type	Protocol type, tcp, udp, etc.
40.	Service	Type of network
41.	Flag	Status of flag
42.	xAttack	Attack type

figure 5. Some of the supervised learning algorithms are: Naive Bayes, Logistic Regression and Support Vector Machine. The supervised model has divided into two categories. The first one is the classification in which the output variable is categorical data. The second one is the regression in which the output class is a real value. The advantage of supervised learning is that it helps to solve the various types of real-world problems.

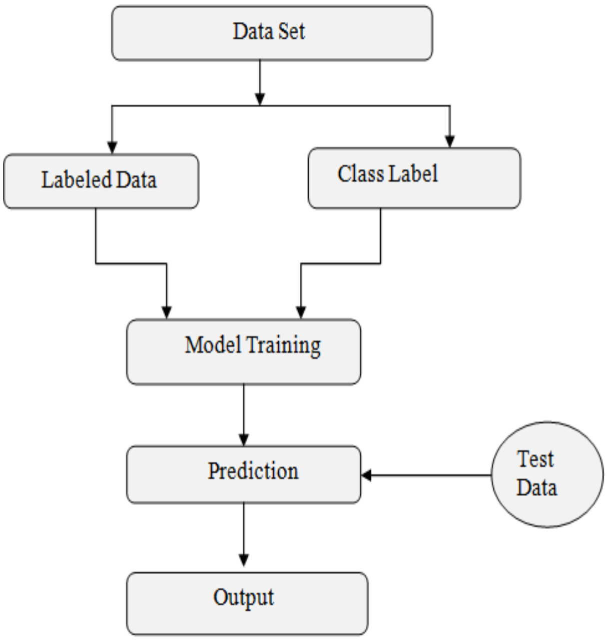

Figure 5. Architecture of Supervised Learning

- Naive Bayes
  A group of supervised learning algorithms built on the Bayes theorem is referred to as the Naive Bayes [25]. It aids in the resolution of classification-related problems. It is typically used in text classification tasks where the training data sets are of high dimension. Being a probabilistic classifier, predictions are done based on the probability of the object. The Bayes theorem is discussed in equation 2.

$$P(A|B) = P(A)\frac{P(B|A)}{P(B)} \quad (2)$$

where A and B are the events. P(A) and P(B) are the independent probability. P(A/B) is the probability of A given that B is true. P(B/A) is the probability of B given that A is true. It performs well in multi-class classification as compared to binary classification. One of the disadvantages is that all the relationships between features are not learned as the algorithm makes an assumption that all features are unrelated or independent.

- Support Vector Machine
  Support Vector Machine (SVM) is a form of supervised learning technique. It is used for both regression and classification purposes however, most of the time it is used in classification problems. It is mostly used for two group classification problems due to its excellent accuracy and capacity to analyze high dimensional data. Support Vector Machine is a fast and dependable classification algorithm that performs very well with a limited amount of data.

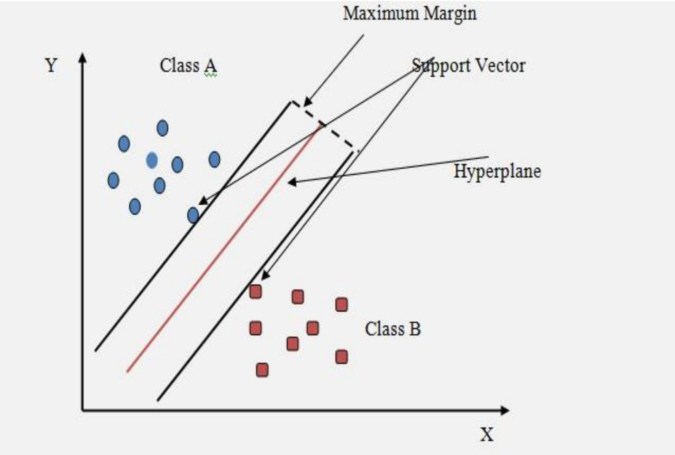

Figure 6. Architecture of Support Vector Machine

The SVM algorithm's [26] objective is to find a hyperplane which can separate the data set into a certain number of groups that contains data with similar features as shown in figure 6. The Support Vector Machine is basically of two types: Linear SVM and Nonlinear SVM. Linear SVM is mainly used for data that are linearly separable i.e a straight line can divide the data set into two classes. When there is a nonlinearly separable data in that case we use nonlinear SVM.

- Logistic Regression
  It [10] is used in solving classification problems. Logistic Regression evaluates the relationship between the dependent and independent variables.

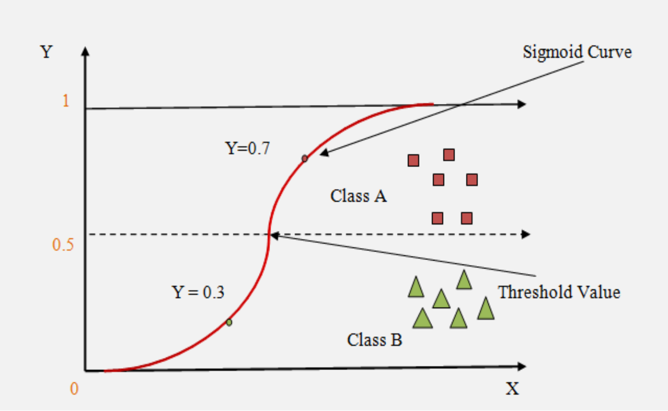

Figure 7. Logistic Regression Architecture

LR works for both binary and multi-class classification but LR performs better on binary class classification. The probability of an event occurring is anticipated by fitting the data to the logistic function. The logistic function selects values between 0 and 1. If the value is greater or equal to 0.5, it is labeled as 1, else it is labeled as 0 as shown in figure 7. The advantages of logistic regression are that it performs better when the data are linearly separable and it is less prone to the over-fitting problems. The major disadvantages of the algorithm are that nonlinear problems can't be solved using logistic regression and if there is a high dimensional dataset then there is a chance of over-fitting.

- Proposed Model

The Proposed model is an ensemble technique in which the Adaboost is combined with Logistic Regression. Adaboost is a machine learning technique developed to improve classification efficiency [27]. The basic working idea of boosting algorithm is as follows: data are initially divided into groups using draft rules. Every time the algorithm is executed, additional rules are added to this preliminary set of rules. In this manner, misclassification is reduced. In this approach, all the weak classifiers combine to create a strong classifier capable of detecting different types of attacks. The main advantage of the Adaboost approach is that net classification error is evaluated in each learning step. The architecture of the proposed model is explained in figure 8.

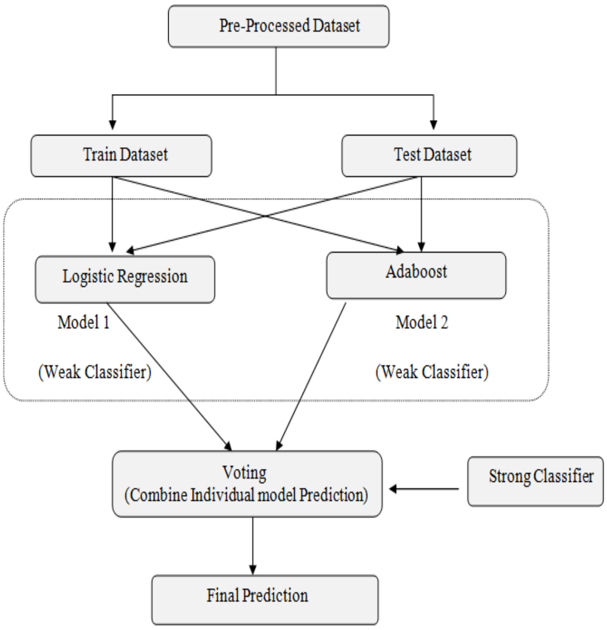

Figure 8. Architecture of Proposed Model

2) Unsupervised Learning: Unsupervised learning is a type of machine learning technique in which models are not supervised by training data sets [28], Here, without any prior training of data, the machine's objective is to categorize the unsorted data according to similarities and patterns as demonstrated in figure 9. Some of the unsupervised learning algorithms are DBSCAN, K-Means, and Genetic K-Means clustering.

- DBSCAN

DBSCAN stands for Density-Based Spatial Clustering of Applications with Noise. DBSCAN is a member of the unsupervised machine learning algorithm. DBSCAN [29] is a density-based clustering and it forms the cluster based on the density. It can find clusters of various shapes and sizes from huge quantity of data that include noise and outliers. The architecture of DBSCAN is illustrated in figure 10.

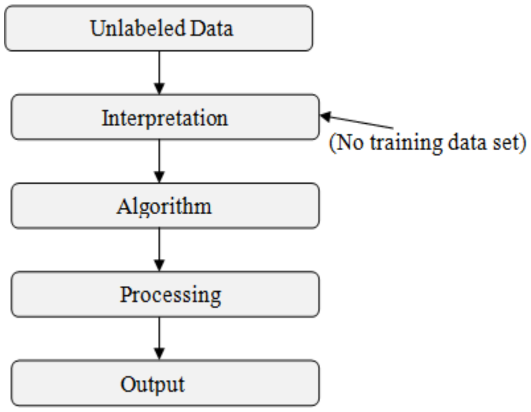

Figure 9. Architecture of Unsupervised Model

- K-MEANS CLUSTERING

K-Means clustering [16] is one of the simplest and most popular unsupervised machine learning algorithm. The K-Means algorithm identifies the K number of centroids. The centroid concept is used to cluster the data points. After every iteration, the centroid value is evaluated using the averaging concept. The objective of the algorithm is to minimize the sum of distances of data points from their respective clusters. The method takes unlabeled data as input, separate it into k number of clusters and performs the same procedure till the optimal cluster is discovered. The main advantage of K-Means is that if the data sets are distinct, then it gives the best results. The main disadvantage of the algorithm is that it needs prior specification for the number of clusters and sometimes choosing the centroid randomly cannot give fruitful results.

- IGKM

Genetic K-Means (IGKM) is a method in which the number of clusters is not known in advance. Genetic Al-

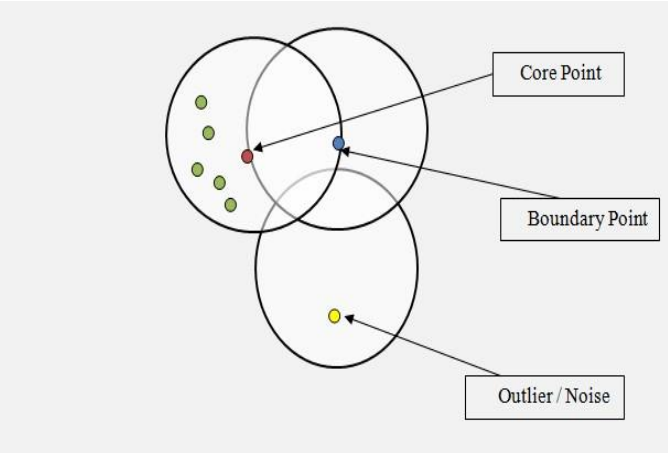

Figure 10. DBSCAN Architecture

gorithm (GA) [11] is used to determine the optimal value K. The fitness function (evaluating function) minimizes the amount of clusters while maximizes the separation and effectiveness as much as possible.

D. Performance Metrics

The following performance measures are used to measure and compare the effectiveness of various IDS based on machine learning [11].

- True Positive (TP) - Here, an attack is identified and confirmed to be an attack. This sort of circumstance is classified as a true positive.

- False Positive (FP) – Here, an attack is detected but it is not actually an attack. A false positive is therefore only a false warning.

- True Negative (TN) - Data that are appropriately classified as normal and is normal. This sort of circumstance is classified as a true negative.

- False Negative (FN) - Attack data that has been erroneously classified as normal. This is the most vulnerable stage since there is no information of the attack that has been already occurred.

- The sum of the TP and TN observations to the total number of observed values is known as accuracy. Accuracy typically determines the total number of classifications that are valid. The formula of accuracy is explained in equation 3.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

- Precision is the ratio of true positive observation to the summation of true and false positive observations as shown in equation 4.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

- Recall calculates the number of valid classifications penalized by number of missing entries. The formula of recall is discussed in equation 5.

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

- F1-Score is a combination of precision and recall as shown in equation 6. A good F1 score means that there are lesser false positives and negatives. Its value lies within 0 and 1. An F1 score of 1 is depicted as perfect while an F1 score of 0 is a failure.

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

E. Experiment Results

Weka is open-source software that offers tools for pre-processing of data, execution of various machine learning algorithms, and visualization tools, allowing us in building machine learning algorithms and help to apply in real-life scenario. It is written in Java and runs on almost any platform [30].

Table III
Experiment Results of Unsupervised Learning Models

Unsupervised Learning					
Algorithm	Attack	Accuracy	Precision	Recall	F-Score
DBSCAN	DOS	94.18%	96.7%	88.6%	92.5%
	Probe	79.51%	38.8%	79.4%	52.2%
	R2L	80.91%	70%	99.9%	13.1%
	U2R	79.4%	26.2%	87.5%	40.3%
K-Means	DOS	94.54%	97.1%	89.1%	92.9%
	Probe	79.05%	38.9%	85.5%	53.5%
	R2L	79.83%	54%	99.9%	10.3%
	U2R	78.47%	26.2%	94.6%	41.1%
IGKM	DOS	82.7%	97.2%	95%	96.1%
	Probe	54.77%	95.7%	99.9%	97.8%
	R2L	31.58%	73%	99.9%	13.7%
	U2R	29.60%	64.3%	99.9%	78.3%

The hardware specification is as follows: Intel i5 10 generation, 1.19 GHz machine with 8GB of Random Access Memory (RAM) and 512GB of Read-only Memory (ROM). In this experiment, after the pre-processing phase, The public NSL KDD data set are split into 70-30 ratio, 70% for training the model and 30% for testing the model. The target class of the dataset is attack and the IDS identifies four different types of attacks i.e, DOS, Probe, R2L and U2R. The well-known algorithms of supervised and unsupervised learning are applied on the pre-processed dataset. In supervised learning, we have used Support Vector Machine, Naive Bayes and Logistic Regression. The proposed model also falls in the category of supervised learning. In unsupervised learning, we have employed DBSCAN, K-Means and Genetic K-Means clustering. The performances of different unsupervised learning algorithms are discussed in table 3. while the performance of different supervised learning algorithms are shown in table 4. Out of all these supervised and unsupervised learning algorithms, the proposed/ensemble model performs better than any other algorithms either it is supervised or unsupervised. The ensemble model obtained an accuracy equal to 99.91%, 99.60%, 99.90% and 98.15% on DOS, Probe, R2L and U2R Respectively.

Table IV
EXPERIMENT RESULTS OF SUPERVISED LEARNING MODELS

Supervised Learning					
Algorithm	Attack	Accuracy	Precision	Recall	F-Score
SVM	DOS	91.18%	87%	91.8%	89.3%
	Probe	81.33%	42.7%	95.2%	58.9%
	R2L	85.57%	63%	64.4%	11.4%
	U2R	84.98%	34.2%	96.4%	50.5%
NB	DOS	93.9%	89.8%	95.8%	92.7%
	Probe	93.40%	69.2%	95.8%	80.4%
	R2L	97.8%	38.1%	76.3%	50.8%
	U2R	86.1%	35.8%	94.6%	52%
LR	DOS	98.4%	98.7%	97.4%	98%
	Probe	98.19%	93.8%	93.4%	93.6%
	R2L	98.95%	75%	40.7%	52.7%
	U2R	97.8%	90.2%	82.1%	86%
Proposed Model	DOS	99.91%	99.80%	99.99%	99.99%
	Probe	99.6%	99.4%	98.2%	98.8%
	R2L	99.90%	98.2%	94.9%	96.6%
	U2R	98.15%	87.7%	89.3%	88.50%

F. Performance Analysis

In this section, we have compared the results obtained by our proposed model with the results obtained by previously proposed models. Table 5 presents the results that we obtained and compares them with the results of B. Selvakumar [19]. It is found that the Proposed model performs better than the Decision tree (C4.5) and Bayesian Network (BN).

Table V
COMPARISON WITH STATE OF ART MODEL

Model	Attack	Accuracy
B.N (B. Selvakumar et al.)	DOS	99.95%
	Probe	93.42%
	R2L	97.83%
	U2R	68.97%
C4.5 (B. Selvakumar et al.)	DOS	99.98%
	Probe	63.85%
	R2L	98.73%
	U2R	17.24%
Proposed Model	DOS	99.91%
	Probe	99.6%
	R2L	99.90%
	U2R	98.15%

V. CONCLUSION AND FUTURE WORK

The paper initially provided a background on the intrusion detection system and its importance in the cyber security space. The NSL-KDD dataset were analyzed and pre-processed using the chi-square test, which reduces the number of features from the dataset and avoids the over-fitting problem. Supervised and unsupervised machine learning algorithms are applied to the pre-processed dataset. When the performance of all the algorithms are compared then it is found that the ensemble model outperforms all other models. In the future, we will conduct an extensive study of ML algorithms to provide a better solution for the IDS by taking a real-time dataset.

REFERENCES

[1] Suman Thapa and Akalanka Mailewa. The role of intrusion detection/prevention systems in modern computer networks: A review. In *Conference: Midwest Instruction and Computing Symposium (MICS)*, volume 53, pages 1–14, 2020.

[2] Tejvir Kaur, Vimmi Malhotra, and Dheerendra Singh. Comparison of network security tools-firewall, intrusion detection system and honeypot. *Int. J. Enhanced Res. Sci. Technol. Eng*, 200204, 2014.

[3] Ashwini Pathak and Sakshi Pathak. Study on decision tree and knn algorithm for intrusion detection system.

[4] Asmaa Shaker Ashoor and Sharad Gore. Importance of intrusion detection system (ids). *International Journal of Scientific and Engineering Research*, 2(1):1–4, 2011.

[5] L Haripriya and MA Jabbar. Role of machine learning in intrusion detection system. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pages 925–929. IEEE, 2018.

[6] Marcel Jung, Octavian Niculita, and Zakwan Skaf. Comparison of different classification algorithms for fault detection and fault isolation in complex systems. *Procedia Manufacturing*, 19:111–118, 2018.

[7] MA Jabbar, Rajanikanth Aluvalu, et al. Rfaode: A novel ensemble intrusion detection system. *Procedia computer science*, 115:226–234, 2017.

[8] C Kalimuthan and J Arokia Renjit. Review on intrusion detection using feature selection with machine learning techniques. *Materials Today: Proceedings*, 33:3794–3802, 2020.

[9] Xianwei Gao, Chun Shan, Changzhen Hu, Zequn Niu, and Zhen Liu. An adaptive ensemble machine learning model for intrusion detection. *IEEE Access*, 7:82512–82521, 2019.

[10] Manjula C Belavagi and Balachandra Muniyal. Performance evaluation of supervised machine learning algorithms for intrusion detection. *Procedia Computer Science*, 89(2016):117–123, 2016.

[11] JV Anand Sukumar, I Pranav, MM Neetish, and Jayasree Narayanan. Network intrusion detection using improved genetic k-means algorithm. In *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 2441–2446. IEEE, 2018.

[12] S Krishnaveni, Palani Vigneshwar, S Kishore, B Jothi, and S Sivamohan. Anomaly-based intrusion detection system using support vector machine. In *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, pages 723–731. Springer, 2020.

[13] Subhash Waskle, Lokesh Parashar, and Upendra Singh. Intrusion detection system using pca with random forest approach. In *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 803–808. IEEE, 2020.

[14] Shadi Aljawarneh, Monther Aldwairi, and Muneer Bani Yassein. Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model. *Journal of Computational Science*, 25:152–160, 2018.

[15] I Sumaiya Thaseen, Ch Aswani Kumar, and Amir Ahmad. Integrated intrusion detection model using chi-square feature selection and ensemble of classifiers. *Arabian Journal for Science and Engineering*, 44(4):3357–3368, 2019.

[16] Amar Meryem and Bouabid EL Ouahidi. Hybrid intrusion detection system using machine learning. *Network Security*, 2020(5):8–19, 2020.

[17] Nevrus Kaja, Adnan Shaout, and Di Ma. An intelligent intrusion detection system. *Applied Intelligence*, 49(9):3235–3247, 2019.

[18] Manjula C Belavagi and Balachandra Muniyal. Multi class machine learning algorithms for intrusion detection-a performance study. In *International Symposium on Security in Computing and Communication*, pages 170–178. Springer, 2017.

[19] B Selvakumar and Karuppiah Muneeswaran. Firefly algorithm based feature selection for network intrusion detection. *Computers & Security*, 81:148–155, 2019.

[20] Ansam Khraisat, Iqbal Gondal, Peter Vamplew, and Joarder Kamruzzaman. Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, 2(1):20, 2019.

[21] NSL-KDD Dataset. https://github.com/initroot/nslkdd-dataset.

[22] Ikram Sumaiya Thaseen and Cherukuri Aswani Kumar. Intrusion detection model using fusion of chi-square feature selection and multi class svm. *Journal of King Saud University-Computer and Information Sciences*, 29(4):462–472, 2017.

[23] T Saranya, S Sridevi, C Deisy, Tran Duc Chung, and MKA Ahamed Khan. Performance analysis of machine learning algorithms in intrusion detection system: A review. *Procedia Computer Science*, 171:1251–1260, 2020.

[24] Ramadass Sathya and Annamma Abraham. Comparison of supervised and unsupervised learning algorithms for pattern classification. *International Journal of Advanced Research in Artificial Intelligence*, 2(2):34–38, 2013.

[25] Chowdhury Mofizur Rahman, Dewan Md Farid, and Mohammad Zahidur Rahman. Adaptive intrusion detection based on boosting and naive bayesian classifier. 2011.

[26] Mustapha Belouch, Salah El Hadaj, and Mohamed Idhammad. Performance evaluation of intrusion detection based on machine learning using apache spark. *Procedia Computer Science*, 127:1–6, 2018.

[27] Mohammed Alrowaily, Freeh Alenezi, and Zhuo Lu. Effectiveness of machine learning based intrusion detection systems. In *International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage*, pages 277–288. Springer, 2019.

[28] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Unsupervised learning. In *The elements of statistical learning*, pages 485–585. Springer, 2009.

[29] Kamran Khan, Saif Ur Rehman, Kamran Aziz, Simon Fong, and Sababady Sarasvady. Dbscan: Past, present and future. In *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)*, pages 232–238. IEEE, 2014.

[30] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.