Intrusion Detection Approach Using AI & ML Classifiers

Devendra R Chauahn[1], Manish Patel[2], Bhavik H. Prajapati[3], Gaurang J. Patel[4]

M.Tech Student, ICT Department, Sakalchand Patel College of Engineering, Visnagar, India[1]
HOD ICT, Department, Sakalchand Patel College of Engineering Visnagar, India[2]
Assistant Professor, ICT Department, Sakalchand Patel College of Engineering Visnagar, India[3]
Lecturer CE Department, Swami Sachchidanand Polytechnic College, Visnagar, India[4]

drc0305@gmail.com[1], mmpatel.it@spcevng.ac.in[2], bhavik.ec07@gmail.com[3], gaurangpatel9@gmail.com[4]

Abstract: In the Current world situation with an increase in internet speed and bandwidth, data requirement increases with a transfer of a tremendous amount of data over a network, especially internet, wired or wireless network. This poses a significant challenge to network security or cyber security i.e. unauthorized access to secure data. To counter these challenges on wireless networks is hard with its extra ordinary properties. To counter this challenge IDS (Intrusion Detection System) is used to detect various types of attacks on a network by analyzing abnormal behavior on a network. One common method to detectthis type of attack was signature-based, the other was an anomaly that provided security to the network. With the introduction oremergence of AI, ML techniques can be used in IDS to detect this type of attack with more accuracy. There are some proposed structures architectures or models to secure networks that provide some significant results. Here we are going to use different Ml algorithms (RF, SVC, GNB) and then XGBClaasifier to get better accuracy in IDS to detect various attacks.

Keywords: Machine Learning, Intrusion Detection system, Google Colab

I. INTRODUCTION

In today's world with an increase in the amount of data due to changes in the nature of the internet with increasing speed, and changes in network communication methods, large amounts of data are transferred, stored or processed across networks or the internet, with cloud computing providing various services such as SaaS(software as a service) data are stored or process with ease or internet orprivate network. As data in a network increases, to get unauthorized access to vulnerable data or to attack a network or internet system different network threats or attacks are also increasing.

To overcome this attack or threat, IDS(Intrusion Detection System) is used over a network. IDS are of two types Signature-based and anomaly-based. Signature Based IDS detects predefined types of attacks or rules. Anomaly Based IDS detects attacks based on different patterns of data.

IDS is basically based on intrusion detection principals or frameworks over a network. It is a combination of hardware and software components that runs on a server computer or machines. It inspects the activity of the user or program using a server to find potential internal threats on a server machine. It also monitors network traffic on a network connected to the server that searches for outside attacks. IDS alerts or informs the network administrator about these suspicious activities

With the introduction of AI(Artificial Intelligence) in this new era, Various ML(Machine Learning) algorithms are used in IDS to detect various attacks with accuracy. Machine Learning algorithms are divided into different types basic two types are there:-Supervised Machine Learning Algorithm-train machine on a given label dataset with a given relative output. Unsupervised Machine Learning Algorithm-Train machine on a set of unlabeled data that is output data is not paired with a given input. Instead, it finds patterns and relationships among given data.(Details of the machine learning algorithm are given in below Fig-1).

IDS can have several problems like a high false positive rate and low detection rate to overcome this problem we will use different ML approaches to get higher Accuracy and a low false positive rate with a better detection rate.

Objective:

1. To apply the preprocessing method to a dataset to remove unwanted, white space or special characters.
2. To Check outliers, imbalance data in the dataset, and to balance those data by using various sampling techniques.

http://doi.org/10.63766/spujstmr.24.000015

3. To apply different classic Machine Learning algorithms Such as RF(Random Forest), SVM(Support Vector Machine), and GB(Gaussian Bayes).
4. To apply XGBClassifier algorithm to our dataset for intrusion detection

TABLE I:

DIFFERENT MACHINE LEARNING ALGORITHM [1]

Machine Learning Algorithmn		
Supervised Machine Learning (labelled data,target/Output specified)		Unsupervised Machine Learning(unlabelled data,target/output not specified)
Regression (continous value)	Classification (categorical value)	Culstering
Linear Regression	SVM(Support Vector Machine)	K-Means Culstering
Logistic Regression	Naïve Bayes	Gaussian Mixture
Ensemble Method	Neural Network	PCA(Principal Component Analysis)
Decision Tree	KNN(K-Nearest Neighbour)	Apriori
Support Vector Regression	Random Forest	Markov Models

II. LITERATURE REVIEW

In this section, various literature works done on Intrusion Detection Systems using Machine Learning or Artificial intelligence approach are discussed.

The Author has used K-means Clustering with the KDD dataset based on the outlier Detection framework. The main aim was to remove outliers. By using the K-means Clustering algorithm it gains an accuracy of approximately 92.25% to detect attacks on the network. [1]The author published an article that used the CNN(Convolution Neural organization) algorithm to discover interruption in networks mainly wireless-based. Basic works show feature extraction or selection techniques to detect attacks with an accuracy of approximately 98 %.[2]. The author in this paper uses a Technique called PSO(Particle Swarn Optimization) in conjunction with Feature Selection for an intrusion detection system. To reduce unwanted attributes he uses Feature selection using a random forest algorithm then he applies various classifier algorithms such as K-NN(K nearest neighbor), SVM(support vector machine), DT(Decision Tree), and LR(Logistic algorithm) Then he also applied PSO(Particle Swarn Optimization) with minimum attributes of data set to acquire better accuracy and data rate. [3]

The author presented an article proposing the idea of stacking for detecting suspicious activity across a network to detect threats or attacks. They have used heterogeneous dataset UNSW NB-15 and UGR' 16. Different Classification and Regression algorithms such as K-NN(K-means nearest neighbor), and LR(Logitech regression) is applied to the dataset and then the ensemble technique is used for stacking. The final SVM(Support Vector Machine) algorithm is applied to the dataset. They get an Accuracy of 97% accuracy for UNSW NB-15 dataset. [4]

The author presented an AI(Artificial Intelligence) strategy with two algorithms first SVM(Super Vector Machine) andthen second Naïve Bayes algorithm. The author used the NSL-KDD dataset and then applied both algorithms. It shows SVM(Support Vector Machine) performed better than Navies Bayes in terms of accuracy and detection rate. [5]

The Author have proposed the XGBoost-DNN model for the characterization of organization interruption. This proposed model has three step-Normalization, Feature Selection, and Classification. NSL-KDD dataset is used. XGBoost-DNN Model is applied on said dataset with another algorithm such as SVM, Naïve Bayes, and Logistic Regression and then the comparison is done in which DNN(Deep Neural Network) revels consistent accuracy among another existing model. [6]

The Author proposed a real-time intrusion detection system for the IOT network, here data is created by a user on the basis ofnetwork traffic on the IOT(Internet of Things) Network, and data is collected on an observation made

on the network traffic. A random forest algorithm is applied to said above data. This method gets an accuracy of 91.18% in real-time testing. [7]The Author here presented a Machine Learning approach using the Feature extraction technique, to reduce dimensions PSO(Particle Swarm Optimization) algorithm is used with DT(Decision Tree) and KNN(K-Nearest Neighbor) these algorithms are applied on KDD-CUP 99 datasets, and then two algorithms are compared for accuracy. Results show PSO+KNN with an accuracy rate of 96.2% performs well compared to PSO+DT with an accuracy rate 89.6% in identifying network attacks. [8] All the above research analyses apply various Machine Learning algorithms to various datasets to get better accuracy on an average 90%. Still, challenges are there such as Data imbalance and outlier unwanted features we have to overcome said challenges withdifferent available techniques.

III. PROPOSED SYSTEM AND IMPLEMENTATION

In the proposed method, we will split our dataset in train and test data. Before that, we will apply data preprocessing and then divide data into train data with 80% of the dataset and test data with 20% of the original dataset. We use Random over Sampling and SMOTE techniques on the data set to balance data in the dataset.To normalize data in the dataset we will apply a MIN-MAX scaler. Then we will apply different algorithm such as GNB(Gaussian naïve Bayes),Random Forest(RF),SVM(Support Vector Machine).We will apply the proposed XGBClassifier algorithm form the XGBoost library. For intrusion detection and compare with other applied algorithms. when we apply XGBClassifier with oversampling and normalization we can get betteror more accuracy for intrusion detection. Then we will compare all algorithm result scores for accuracy.

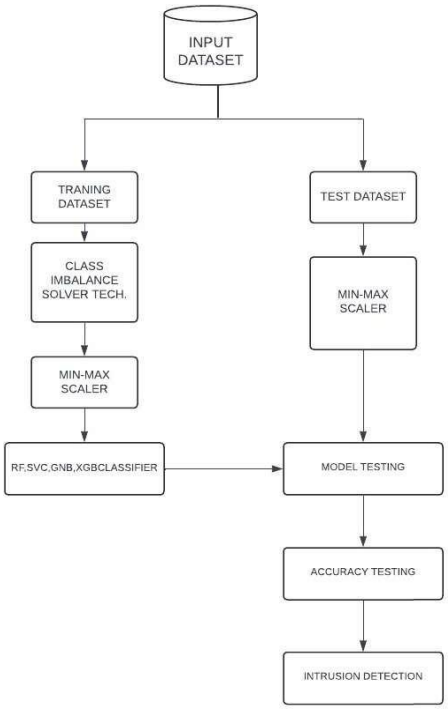


Fig. 1 Proposed Intrusion Detection System

Imbalance Data Class:

There are various techniques to solve data imbalance in datasets.

Oversampling

Oversampling is used when there are imbalanced data in the dataset that is when one class has more instances called the majority class and another class has minimum instances or datapoint. When we apply classification or regression algorithm to this type of imbalanced data, we do not get the required result that is training with majority and minority but prediction on new data will not desire result. To overcome this problem oversampling technique is used in oversampling technique. In the minority class, data points are increased by duplicating existing one or generating new ones by creating synthetic data.

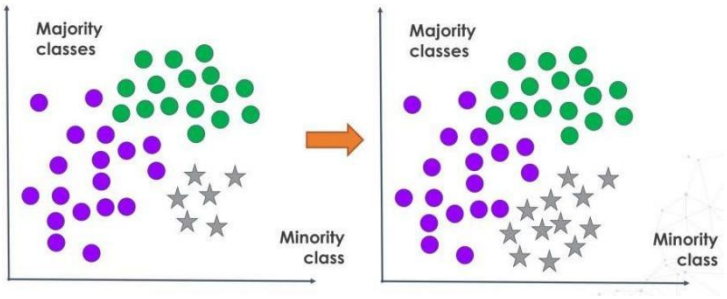


Fig. 2 Oversampling

Over-sampling technique:

Random oversampling This technique creates new instances in a minority class by duplicating or replicating existing instances at random or randomly. This technique is useful with small datasets i.e. no need to gather much data to increase no instance of the minority class. It's easy to use, but there is a disadvantage of overfitting.

SMOTE Synthetic minority oversampling technique.

It is a popular technique of oversampling used for the solution of imbalanced classes in data sets in machine learning. This algorithm creates new instances and new data points in existing minority classes with existing.

SMOTE creates a new instance from an existing instance plotting two existing instances so that new data points or instances are created. It's advantage is it creates new samples or instances based on existing ones reducing duplication for improved machine-learning model performance.

MIN – MAX Scaler

MIN – MAX Scaler is over used for normalization in datasets, used in data analysis and machine learning. MIN-MAX Scalerscale numerical feature in desire range between 0 & 1 or 1 & -1 MIN-MAX scaler scale a value using the below formula.

Fig 3: Synthetic Minority Oversampling Technique- SMOTE

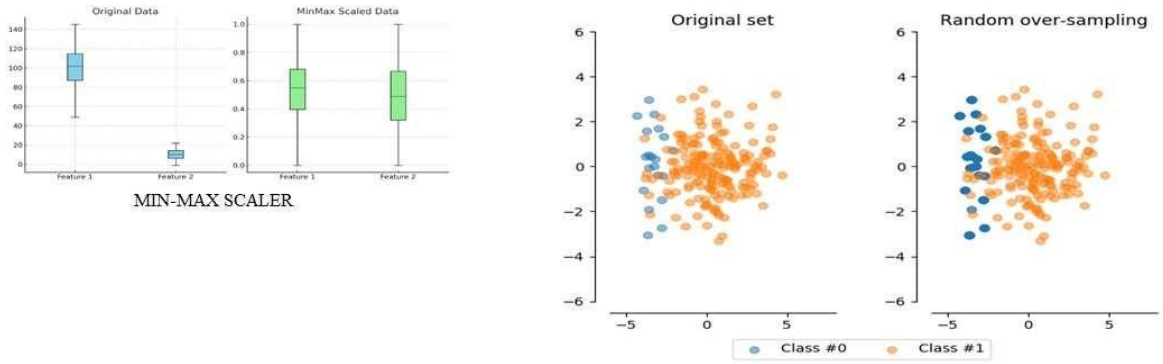


Fig. 4 Min-Max scalar & random oversampling

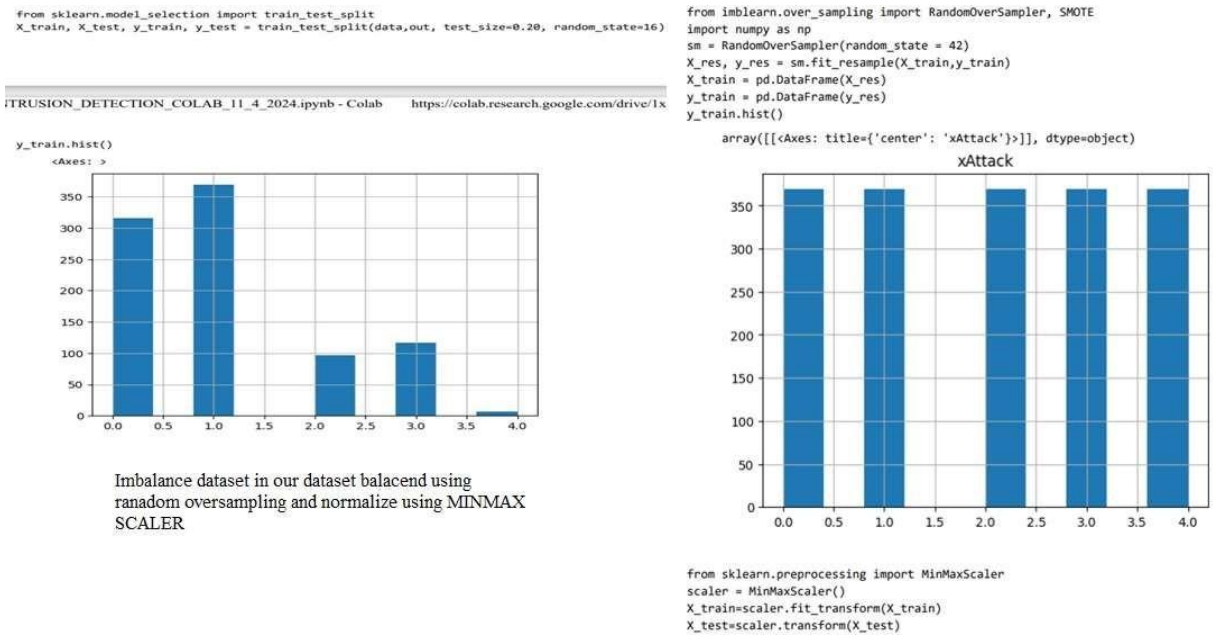


Fig.5 Imbalanced data in our dataset balanced data using random oversampling, smote, and normalizing using mix-maxscaler in our dataset

Random Forest Algorithm

It is a classification and regression-supervised machine learning algorithm based on ensemble learning. It generates several decision trees from a subset of a given dataset. Each decision tree generates on prediction or output then it combines all predictions of all decision trees, based on the majority or average final output or prediction generated. In a random forest classifier from training data random data samples are selected then the random forest classifier will generate a decision tree for every training data by averaging decision tree voting will take place and lastly most voted prediction will be Selected as the final prediction. Random forest is based on ensemble learning, in ensemble learning we combine multiple models for prediction. Random forest classifiers use the bagging ensemble learning technique which combines multiple models, each model is processed parallel, and the final prediction is made by combining the output of all models and voting done for the same.

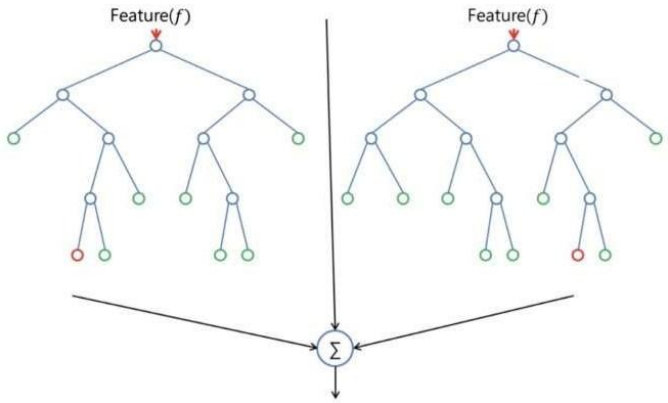


Fig. 6 Random Forest classifier

FINAL_INTRUSION_DETECTION_COLAB_11_4_2024.ipynb - Colab https://colat

```
from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier(max_depth=1, random_state=34,n_jobs=-1)
RF.fit(X_train,y_train)
RF.score(X_test,y_test)
```

```
0.7004405286343612
```

Fig. 7 Implementation of Random forest classifier in our model

Gaussian Navies Bayes Classifier:

It is a classification-supervised machine learning algorithm. Is based on Baye's theorem.

GNB (Gaussian Navies Bayes) is an extension of Navies Bayes. It calculates the mean and standard deviations for the trainingdata. It is a probability density function with a formula based on a problem involving continuous numeric data.

Bayes theorem says for the probability of Y based on evidence X conditional probability formula

$P(X|Y)=P(X|Y).P(Y)/P(X)$ here P(X),P(Y) is previous probability event Y and evidence X

In Gaussian distribution which is also known as the normal distribution probability of X is calculated using:

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Here standard deviation=σ,,mean =μ

```
from sklearn.naive_bayes import GaussianNB
GNBmodel = GaussianNB()
GNBmodel.fit(X_train, y_train)
GNBmodel.score(X_test,y_test)
      0.5594713656387665
```

Fig.8 Implementation of Gaussian NB classifier

Support Vector Machine/Classifier

It is a supervised machine-learning algorithm used for classification and regression. It can be used for categorical as well as multiple continuous values. It used as a method to find a hyperplane in an N-dimension or N- no of a feature that uniquely classifies the data point in a particular class to separate two classes there may be many hyperplane lines. We have to choose those lines that have a maximum margin between or maximum distance between data of both classes.

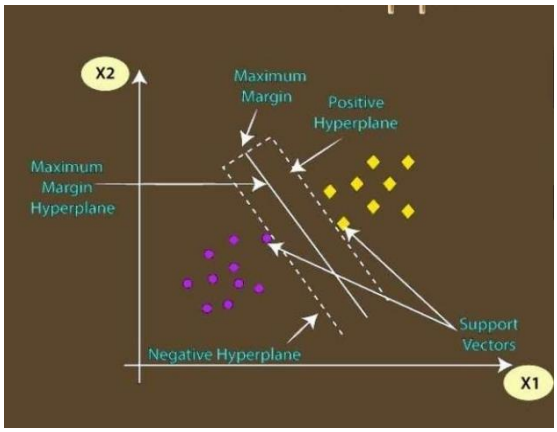


Fig. 9 Support vector machine

```
from sklearn.svm import SVC
SVCmodel=SVC(kernel='rbf',random_state=15)
SVCmodel.fit(X_train, y_train)
SVCmodel.score(X_test,y_test)
      0.9162995594713657
```

Fig. 10 Support vector Classifier Implementation

Xgbboost & Xgbclassifier

XGBBOOST known as extreme gradient boosting is a machine learning algorithm or library used for supervised learning. i.e. for classification, and regression. It is based on gradient boosting architecture used for their better accuracy result. It works easily on large data sets. It is specially built for high performance and speed which is why it is also used in real-time applications to solve fast & accurate machine-learning problems. It provides parallel tree boosting also known as GBDT, or GBN.

XGBClassifier uses a gradient tree boosting algorithm it trains an ensemble of decision trees by training each tree to predict theprediction error of all previous trees in the ensemble.

XGBClassifier combines multiple weaker models to predict stronger ones. There is an objective function with which eachDT(Decision tree) is trained.

$$obj = \min(\sum^{n}{}_{i=1}\, l\ (yiyi^{\wedge\prime}) + \sum^{k}{}_{k=1}\ r\ (fk))$$

Where l=loss function which must be minimum yi=actual value and yi' is predicted value, f_k i-kth decision tree and r= regularization function.in short-Obj=l + r where l=loss function and r=regularization function

```
from xgboost import XGBClassifier as ARTC
ARTCmodel = ARTC(max_depths=300,random_sta
ARTCmodel.fit(X_train, y_train)
ARTCmodel.score(X_test,y_test)
```

```
0.9823788546255506
```

Fig. 11 XGBClassifier implementation

IV. RESULT AND PERFORMANCE EVALUATION

Our proposed model uses Google collab and Python language to write, compile, and run our Machine learning code. We have used different available Python libraries such as Numpay, Panda,sci-kit learn, Imblearn, and XGBoost. Our dataset has the following type of attack as an output feature, We have given the following attack a numerical value from 0 to 4(shown in the table)

TABLE: II
ATTACK TYPES

Attacks Types	Value Assigned	ML algorithm (classifier)	Score(227 test data)	Score (>5000 data set)
Dos(Denial of services)	0	RF	0.70	0.55
Normal(not a attack)	1	GNB	0.55	0.34
Probe	2	SVC	0.91	0.84
R2L(Remote to user attack)	3	XGBClassifier	0.98	0.99
U2R(user to root attack)	4			

.
Among all applied machine learning algorithms we have got 99% accuracy in the XGBClassifier

Evaluation

We will use a confusion matrix to evaluate different algorithm predictions. A confusion matrix is used to measure the performance of a machine-learning classification algorithm. It can have two or more classes.In the confusion matrix, we have 4 different combinations. fig-14-Confusion Matrix

TP=True Positive-How many actual true values predicated, TN=True Negative-how many actual true false values are predicated, FP=False Positive-How many true but that value is predicted false, FN=False Negative-How many false but that value is predicted true. We have different measurements for valuation.

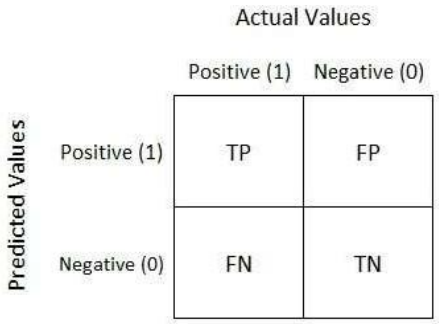


Fig. 12 Confusion Matrix

When we applied RF,GaussianNB, SVC classifier algorithms we had an accuracy of 70%,55%, 91% respectively.. When we applied XGBClassifier we got an accuracy of 98.24%, When we increased testing data and applied XGBClassifier we got an accuracy of 99% in detecting attacks as shown in the image.

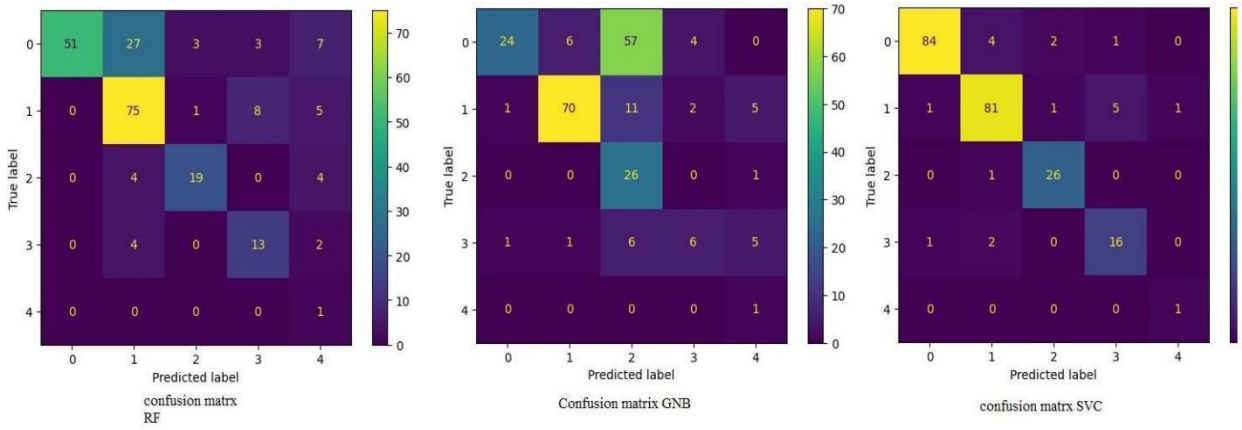
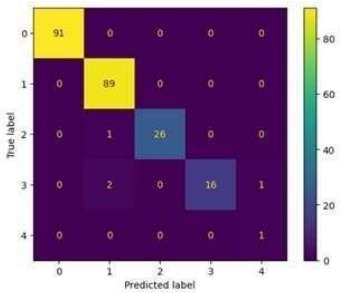


Fig.13 RF, GuassianNB, SVC classifier valuation and confusion matrix

	precision	recall	f1-score	support
0	1.00	1.00	1.00	91
1	0.97	1.00	0.98	89
2	1.00	0.96	0.98	27
3	1.00	0.84	0.91	19
4	0.50	1.00	0.67	1
accuracy			0.98	227
macro avg	0.89	0.96	0.91	227
weighted avg	0.99	0.98	0.98	227



	precision	recall	f1-score	support
0.0	1.00	1.00	1.00	2679
1.0	1.00	1.00	1.00	1933
2.0	0.88	0.95	0.91	38
3.0	0.99	0.99	0.99	478
4.0	1.00	0.50	0.67	4
accuracy			1.00	5132
macro avg	0.97	0.89	0.91	5132
weighted avg	1.00	1.00	1.00	5132

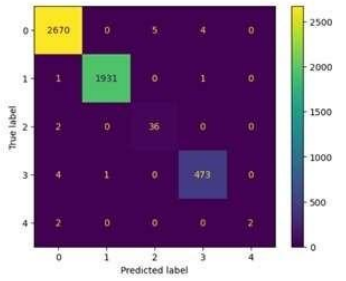


Fig.14 XGBClassifier Performance evalution and Confusion Matrix.(With 227 and 5132 test data)

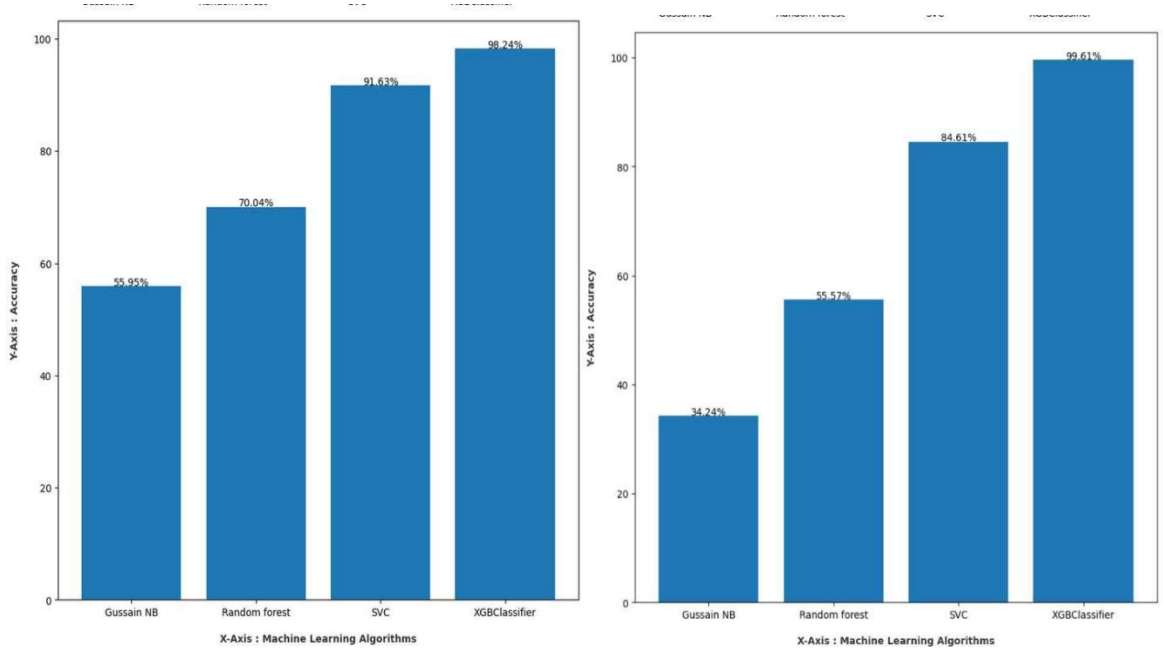


Fig.15 Accuracy bar chart of all machine learning algorithms applied

V. CONCLUSION

In this research paper, I have proposed a system to pre-process data & then normalize with scaler, balance data with random oversampling techniques and then we have applied machine learning classification algorithm RF giving an accuracy of 55.95%, gaussion Bayes gives as accuracy of 74.40%, SVC Classification give as 91.63% in detecting inclusion attack. But when we apply XGBClassifier we have an accuracy of 98.24%. When we increase the amount of data in data then we set an accuracy of 99.61% in XGBClassifier in our proposed work we set an accuracy of 99% in XGBClassifier out performance after applying machine learning. The above two figures (Bar Chart) Fig. 15 shows a comparison analysis one with a small number of datasets and the other with an increasing number of datasets. Machine Learning Classification algorithm XBGclassifier performs well with an Accuracy 98.24% with 227 datasets (test) and 99.61% with above 5000 datasets (test)

ACKNOWLEDGEMENT

I would like to acknowledge My guide and HOD Dr.Manish Patel sir for his kindness and support to me in doing myresearch work.

REFERENCES

[1] S.Sandosh,V.Govindaswamy, G.Alikaj, *"Enhanced intrusion detection system via agent clustering and classification based on outlier detection",* springer 2020.

[2] B. Riyaz, Sannasi Ganapathy, *"Deep learning approach for effective intrusion detection in wireless network using* CNN" Springer 2020.

[3] Nilesh Kunhare, Ritu Tiwari, Jaydip Dhar, "*Particle swarm optimization and feature selection for intrusion detection system"* Springer 2020.

[4] Smita Rajagopal, Poornima Kandapur andKatigareil, Siddarampaa Haresha, *"A stacking ensemble for network intrusion detection using heterogeneous dataset"* Springer 2020.

[5] Anisha Halima, Dr. K. Sundra Kantham., "*Machine learning based intrusion detection system*", IEE Springer 2019.

[6] Preeti Devan NeeluKhare, "*An efficient XGBoost – DNN based classification model for network intrusion detection system"* Springer 2020.

[7] Rishabh Hattanki, Shruti Houji, Manisha Dhag, "*Real-time intrusion detection system for IOT network"*

I2CT.2021.

[8] Roseline Oluwaeseum ogundokun, JosephBamidele Awotunde Peter Sadiku, "*Enhanced IDS using particle swarm optimization feature extraction techniques*", Elsevier 2021.

[9] Hariharan Rajadurai, Usha Devi Gandhi, "*A stacked ensemble learning model for intrusion detection in wireless network"*SPRINGER 2020

[10] Raisa Abedin, Sajjad Waheed, "*Performance analysis of machine learning models for intrusion detection system using Gini Impurity-based weightedRandom Forest(GIWRF) feature selection technique"* Springer 2022

[11] Ahmed Abdelkhalek, Maggie Mashaly, "*Addressing the class imbalance problem in network intrusion detection system using data resampling and deeplearning*", The Journal of Supercomputing 2023