Performance Analysis of IDS Using Feature Selection and ML Methods

By

Muhammad Muheet Khan

00000320763 NS

Supervisor

Brig (Retd) Associate Professor Dr. Fahim Arif

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Software Engineering MSSE

In

Department of Computer Software Engineering Military College of Signals,(MCS),
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(December 2022)

Thesis Acceptance Certificate

Certified that final copy of MS/MPhil thesis entitled “**Performance Analysis of IDS Using Feature Selection and ML Methods**" written by **Muhammad Muheet Khan**, (Registration No **00000320763 NS**), of Department of Computer Software Engineering Military College of Signals,(MCS)has been vetted by the undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS/M Phil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: ____________________

Name of Advisor: **Brig (Retd) Associate Professor Dr. Fahim Arif**

Date: ____________________

Signature (HoD): ____________________

Date: ____________________

Signature (Dean/Principal): ____________________

Date: ____________________

Dedication

This thesis is dedicated to my beloved parents and all the siblings

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at at Department of Computer Software Engineering Military College of Signals,(MCS)or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at Department of Computer Software Engineering Military College of Signals,(MCS)or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Muhammad Muheet Khan**

Signature: ____________________

Abstract

Intrusion detection system (IDS) in past many years has played an important part in improving performance of systems by avoiding and preventing false attacks on the systems to make the networks more safe and secure. Now it has become very difficult in this world to work on the internet due to the cyber-attacks and security risks on the internet like intrusion detections. Intrusion detection system is very much powerful so researchers have produced the different types of intrusion detection system for different types of environments, because IDS can detect the abnormal behaviors of the system very accurately. On the other hand many other problems are raising for the researchers and the intrusion detection systems due to the change of the behaviors of the attacks on the system. Which is very much frustrating because the attacks are highly impactful as well as the life and the accuracy of the system is on the stake. So relying on these intrusion detection systems IDS and prevention systems PS is very risky due to their inability to detect the threats against the new level and nature of attacks on the systems.Recently machine leaning has reached its heights in terms of detection of threats and anomalies as compared to the anomaly based detection systems with good potential where these kinds of systems generally fail. For that purpose state of the art classical machine learning algorithms are used on the UNSW-NB15 dataset as a benchmark for experimentation while other datasets are also available. In this paper, many types of techniques related to machine learning are used to detect the attacks accurately on the system. On the other hand the performance of the system is degraded when the data is multi-dimensional; therefore numbers of dimensions from 49 to 12 were decreased for achieving more accuracy. In addition to that, features or dimensions which have less importance or not having bigger impact or sparse on the results were filtered out. Hence ensemble methods were applied like Random Forest (RF), support vector machine (SVM), XGBoost, Logistic Regression (LR) and Decision Tree (DT) to check the accuracy of these models. Also the wrapper based feature selection technique RFE (Recursive Feature Elimination) is used to gain the desired features. All the machine learning algorithms are applied for binary classification as well as for the multi classification. In our results it was found out that the Random Forest with the feature selection method (RFE) has the accuracy of 99.70% for the binary classification which is most while using the Anaconda 3.0 with python 3.0 and google colab in this research work. We also applied the machine learning algorithms for multi-classification the accuracy is 70.72% for training and 65.86% for the training dataset, and other wrapper based techniques like sequential feature selection which gives the accuracy of 99.70% with random forest using feature elimination.

Acknowledgments

All praises to Allah for the strengths and His blessing in completing this thesis. I would like to convey my gratitude to my supervisor Brig. Fahim Arif, PhD for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the experimental and thesis works are major contributions to the success of this research. Also, I would thank my committee members; Assoc Prof Dr Tauseef Rana and Assoc Prof Dr Ihtasham ul Islam for their support and knowledge regarding this topic. Also last, but not the least, I am highly thankful to my parents and my Siblings. They have always stood by my dreams and aspirations and have been a great source of inspiration for me. I would like to thank them for all their care, love and support through my times of stress and excitement.

Muhammad Muheet Khan

List of Abbreviations

Abbreviations

RF; Random forest

LR ; Logistic regression

SVM ; Support vector machine

ML ; Machine learning DT ; Decision tree

RFE ; Recursive feature elimination

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

In recent times it is observed that intrusion detection system have become very important for organizations. The main reason to develop these systems is the constant attacks by the hackers to intrude the system and get personal information. For that purpose these IDS system have become more relevant in recent technologies [37]. There is much type of attacks that are breaking the privacy of the different systems. Also with the technology is improving the chance of new type of attacks have also been improved. The severity level of security in networks is very high in this era of big data. If we look at the old and traditional techniques of network security today, that is not viable in recent time. They are not suitable at all for latest threats that are accruing now. Dynamic behaviors of the threats as well as the multidimensional data are a big concern when it comes to the old intrusion detection methods. Hackers intrude the network with the help of different types of applications to access the targets. Hence it is very crucial now a day as the technology is advancing with time for the security of the network on the internet. These cyber-attacks must be detected through the advanced intrusion detection systems. There are many research areas that has loop holes and require the advancement in it cyber security or intrusion detection systems is one of them. Internet of things(IOT) , cloud computing and other areas have big impact and challenges faced by the researchers how to fight with cyber-attacks in the modern world. Function of the IDS is given in the figure 1.1.

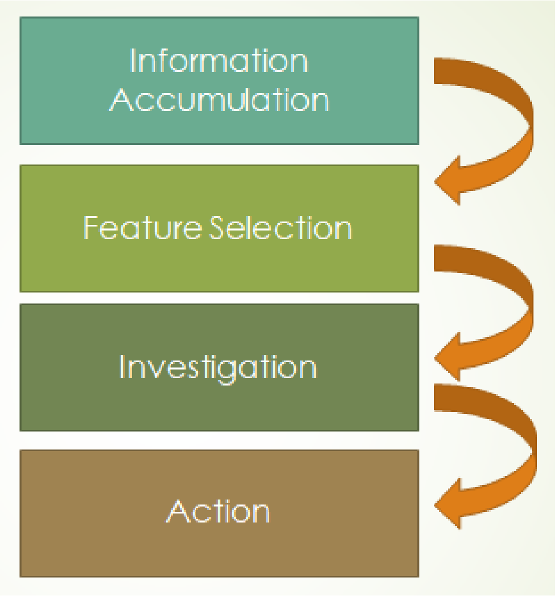


Figure 1.1: Function of the IDS.

1.2 Motivation

The real motivation of the IDS is to detect the attack accurately in quick time. So that many machine learning algorithms are being used by the developers and researchers to gain maximum accurate results so that attacks are identified before they enter into the system and harm them. [58]. That's why many companies require the intrusion detection systems. According to the latest surveys many companies are fear of the threats from the hackers and intruders. That's why they need security systems to be applied in their companies to prevent their data from them. Cyber security revenue now has increased to the over hundred billion dollars per year. Companies now do spend a lot of money to protect their data because the attackers are now more smart and aggressive with the new technology. Machine learning algorithms are the smarter and more secure for these systems. There are many machine algorithms used for that purpose so that these malicious attacks are identified before they enter the system. IDS detect the abnormal behaviors of the system. It does recognize the threat first after those threats are handled with the accuracy as it is very powerful. More often the intrusion detection system detects the powerful interruption on the system. Theses system recognizes the personal computer attacks as it is obvious from its name.

1.3 Scope

The future in this area is very broad. As the time goes on the data is increasing as well as the chances of the security issues has also increased. There is a lot of room to increase its accuracy using different machine learning models. These algorithms can be implemented for different data sets as well as more best and well known machine learning algorithms and deep learning methods. Every machine learning model has different way of evaluating the data. Different data set and machine learning algorithm will give different results according to the requirement. Intrusion detection system can be used for the evaluation of the risk assessment. Also the consumption of time can be handled well so that the higher accuracy rate can be gained through that.

1.4 Problem Statement

In terms of security the intrusion detection systems IDS works three very important functions. One is the screening second is identifying and third is the response to the unwanted and unapproved interventions to the system. The intrusion detection system distinguishes the different types of attack on the system and does the course of action. So there is a dire need to carry out a complete comparison of different algorithms to find the best model for intrusion detection in network traffic. Also, Performance Analysis of IDS using Feature Selection techniques and Machine learning Algorithms will be carried out.

1.5 Purpose and Research Question

In this thesis supervised machine learning algorithms are used on the dataset UNSW NB-15 for the intrusion detection systems on the basis of binary classification accuracy in terms of attack or no-attack on the system. Further this work evaluates the accuracy of different different machine learning algorithms and compare the accuracy between the different machine learning algorithms. We address the following issues regarding our work.

- How does the machine learning algorithms evaluates the accuracy on the given dataset?

- Second is the accuracy of the algorithms increase with the feature selection methods which are available on given dataset or not?
- Does sparsity of data affects the accuracy of machine learning algorithms?

1.6 Approach and Methodology

Different types of machine learning algorithms will be applied on the data set in this section; for the evaluation of the results using the binary classification. Then the hybrid approach that consists of feature selection with the machine learning classifier will be discussed to know the difference between the results by applying the different algorithms on the data set. Other features selection methods will also be applied in order to check the best accuracy results. which will be easy to understand the difference on how good our model works. At the end all the results will be discussed and their accuracies will be recorded in the form of tables.

1.7 Target Group

The works done in this thesis is based on the machine learning algorithms for the intrusion detection. So it is very much in the interests of the researchers who wants to do work on intrusion detection in the field of machine learning. Moreover it has very huge importance for the different organizations who wants to protect their data from the external threats.

1.8 Thesis Breakdown

The research has been done in a lot of different phases and steps. It is divided into the following chapters.

Chapter 1 Introduction: An Overview of Intrusion detection system (IDS).

Chapter 2 Related Work: Discussion and highlighting of work already carried out on this topic by other people.

Chapter 3 Proposed Methodology: The explanation of the proposed methodology to overcome the problems which are observed.

Chapter 4 Experimentation and Results: Testing the validity of the methodology by using dataset in python and calculating results of accuracy.

Chapter 5 Discussion: Critical analysis of the results will be done.

Chapter 6 Conclusion and Future Work: This section provides a recap of all the work done and also shows a direction for the future of this research.

1.9 Summary

In machine learning methods and techniques the model on the data is trained then the model's accuracy is calculated. There are two different types of machine learning Algorithms which are used known as supervised learning and un-supervised learning. Firstly the model is trained on the data to learn and give prediction with accuracy in case of machine learning type of supervised. There are many supervised learning methods and techniques are used for that purpose. Supervised learning includes the classification methods which have the support vector machine and decision trees, and regression have linear regression and logistic regression and many more. In this research paper the supervised machine learning methods and techniques are used to analyze the performance [19]. Which includes the Random forest classifier, logistic regression, XGBoost, Support vector machine [18] and Decision tree for that matter. The dataset is spliced into train and test and then it is fitted into the model. Moreover the wrapper based techniques will be used for dimensionality reduction of the data to extract the reverent features and eliminate the less important features to enhance the accuracy of the dataset. In this way the experiment will perform better and quickly. Next section the related work which is performed on the relevant field will be discussed on the UNSW-NB15 dataset. The techniques used by the other authors and their results will be discussed as well. Further discussion is about our proposed methodology and results on UNSW-NB15 dataset. In last, the conclusion of the entire research work and future work which can be done with intuition to these machine learning methods will be mentioned.

Chapter 2

Related Work

2.1 Introduction

In this chapter the comprehensive discussion on previous work is done related to the specific topic. A literature review examine the publication done by the scholars on that particular topic as well as the books written on that topic also the other ways of getting knowledge. In the review, this previous work should be listed, detailed, summarized, objectively appraised, and clarified. It should serve as a theoretical framework for the study and aid the author in selecting the scope of the investigation. The literature review acknowledges the contributions of previous researchers, ensuring the reader that your work is well-considered. It is assumed that the author has read, assessed, and assimilated a previous work in the subject of study by mentioning it in the current work. A literature review gives the reader a "map" of the field's progress, allowing them to fully appreciate it. The author has included all (or the vast majority) of earlier, noteworthy publications in the area into her or his research, as seen in this diagram. Keeping this in mind, this chapter presents the work of other authors on this specific topic, as well as a conclusive and accurate conclusion based on their work. A literature review provides the reader with a "landscape," allowing them to fully comprehend the field's advances. This diagram shows that the author has incorporated all (or the vast majority) of earlier, significant publications in the topic into her or his research. Keeping that in mind this chapter provides the work carried out by other authors on this specific topic and by using their work a conclusive and accurate.

2.2 Literature Review

In their paper [37] applied different machine learning algorithms which are support vector machine, Xgboost, Decision tree and KNN. In Decision tree the accuracy increases from 88.13 to 90.85 using feature selection method. Moreover Xgboost increased the performance of DT. The SVM method accuracy is increased from 70.63 to 75.51. When compared with the other algorithms the Xgboost accuracy is increased from 88.13 to 90.85 using feature selection.

In other paper [18] has used SVM and other machine learning algorithms. As a result of that support vector machine accuracy is 85.99 as compared to Expectation minimization which has accuracy of 78.47 for binary classification.

In this paper, they used algorithms like support vector machine, gradient boost and logistic regression. They found in their results that support vector machine has the highest accuracy of 82.11% as compared to the other algorithms [23].

In another paper [58] uses logistic regression, KNN and random forest classifiers. Filter based technique chi-square is used for the feature selection. As a result of that random forest indicates that is more accurate in terms of its performance with 99.59 when it is in comparison to algorithms like LR 98.48, NB 76.59 and KNN has 98.28.

Auhtor [19] has used random forest classifier in their paper. The results shows that the precision is 98.3 on the trained data. Also the recall on the proposed dataset is 98 percent. On the other hand the accuracy which is F1 score is 97.5. Using the recursive feature elimination it has the accuracy of 98 percent for the four features only which are (sload, sttl ,ctdstsrcltm, sbytes).

[50] another author used algorithms like random forest, support vector machine and ANN. Among all the algorithms tried and tested the most accurate for binary classification is random forest which is 98.67 and 97.37 for the multi-classification. For feature selection irrelevant features were removed for better results for clustering. they applied different supervised machine learning classifiers like random forest, artificial neural networks and support vector machine. In their results they found out that the random forest classifier has achieved 98.67 accuracy in binary classification as well as 97.37 in multi-classification. In cluster base techniques they achieved 96.96, 91.4 and 97.54 and they have used the random forest classifier on flow and MQTT for accuracy predictions.

[17] have used Xgboost classifier to distinguish the type of the network attack. They have selected 23 features from dataset and used information gain. They trained their model using 23 features on Xgboost which resulted into higher accuracy result.

[51] have used deep learning models like ANN, DNN and RNN to check the performance and accuracy of the IDS. They have used the UNSW-NB15 enhance dataset to compare their respective deep learning models using binary classification and multi-classification. The results shows that deep learning models they have applied have very good performance. The accuracy is 99.59 percent for multi-classification as well as 99.26 percent in binary classification.

[26] in their research have applied different machine leaning algorithms. They proposed the feed forward neural network to gain higher accuracy in comparison with the others algorithms of machine learning. The proposed feed forward neural network has shown the accuracy score of 99.50

[36] in his work used the machine learning algorithms like random forest, j48 and Kmeans. Also expectation maximization is used. The results shows the accuracy using random forest is high with 97.59 on the other hand the j48 performed 93.78 . He also used the feature selection method which correlation based feature selection CFS which enhanced the accuracy of the results.

[31] have used deep neural networks for the experiments. They found out that the results shown using deep neural network have been good with the accuracy of over 90 for the attacks for the different datasets like KDD-Cup'99, NSL-KDD and UNSW-NB15.

[74] in this paper the author represented the implementation feature reduction technique. The purpose of feature reduction is to reduce the features from the dataset in order to achieve high accuracy. In his work the author proposed the five number of features which are selected from feature space. By reducing the features the machine learning algorithms can work more efficiently. In this way results shows more accurate results with accuracy of 99 as well as the testing time is also reduced to 84.

[81] have used UNSW NB-15 dataset for testing and implementing the machine learning models. In proposed work different machine learning algorithms are used to find out the accuracy. Support vector machine algorithm which is supervised machine learning model is used which gives the highest accuracy. For feature selection recursive feature elimination method is used. The exactness is 99.99 from 98.89 in case of decision tree

implementation for the two fold plot. In this way this proposed framework has increased the accuracy of the IDS on the systems of the organizations.

[9] in their research work they have ued the six machine learning algorithms. Decision tree J48, random forest RF, k nearest neighbor KNN, naïve bays NB, support vector machine SVM and artificial neural network ANN are the classifiers used. The dataset for experiments is used is UNSW NB-15 dataset and NSL-KDD. The results represents that the accuracy of the J48 and KNN is very much higher as compared to the other classifiers like random forest and decision trees and others in terms of detecting the accuracy and false rates.

[54] has used the dataset UNSW NB-15 while other datasets are also available. Authors ahave used different machine learning algorithms as comparison between each other. The algorithms like random forest RF, decision tree DT, gradient boosting tree GBT as well as the multi-layer perceptron used an experimentation. Features were eliminated to increase the accuracy and chi-square is used for tests. In the results after experiments the decision tree DT prove to be best classifier among all other classifier with high accuracy and low false positive rate. The performance of the model is increase due to the elimination of non-important features.

[57] have used classical machine learning algorithms like decision tree DT, support vector machine SVM and k nearest neighbor KNN. The dataset used for the experimentation is UNSW NB-15, ISCX-2012, NSL-KDD and CIDDS-001. The results shows that the accuracy using proposed classifier is 99.18 , 99.81 and 99.92 for the CSE-CIC-IDS 2018 data set which better as compared to the other datasets.

[2] have developed the framework which has four well known machine learning classifiers which were used to find the accuracy with dataset of UNSW NB-15. The classifiers are support vector machine, naïve bays, decision trees and random forest using apache spark. The results shows that the random forest classifier has the advantage over the other classifiers in terms of their accuracy detection. The total number of features which are used 42 in the dataset.

[14] in this paper has used the machine learning algorithms which include the deep neural network and auto encoder for intrusion detection. The dataset for experiments used is UNSW NB-15. There are two phases which are used in this method. In first the auto encoder is used for engineering of the different features in the dataset. In second

phase the deep neural network is used for the classification. The results are predicted in terms of their accuracy, F1 score, false positive and ROC as comparison with other classifiers.

[24] in this research work state of the machine learning algorithms were applied on the dataset UNSW NB-15 to detect the attacks on the system. Ensemble method random forest is used with the wrapper based feature selection technique tabu search TS. The results show that the accuracy of the intrusion detection on the system is decreased comprehensively by reducing the 60 of the features from the given dataset.

2.3 Machine Learning Algorithms

Machine learning is defined as the process which is based on the learning from the experience rather than actual programming of the problems or tasks. Machine learning process actually starts with first by training the data from the given dataset and then different machine learning algorithms are applied on it; moreover after applying the algorithms the required results are obtained. Different machine learning which are used, their selection is based on the data and the results we want to achieve in our problem statement. Machine learning is different from the traditional programming in which the data is given as input in the program and output is then get after running the program. The difference between the traditional programming and the machine learning is given in the figure 2.1

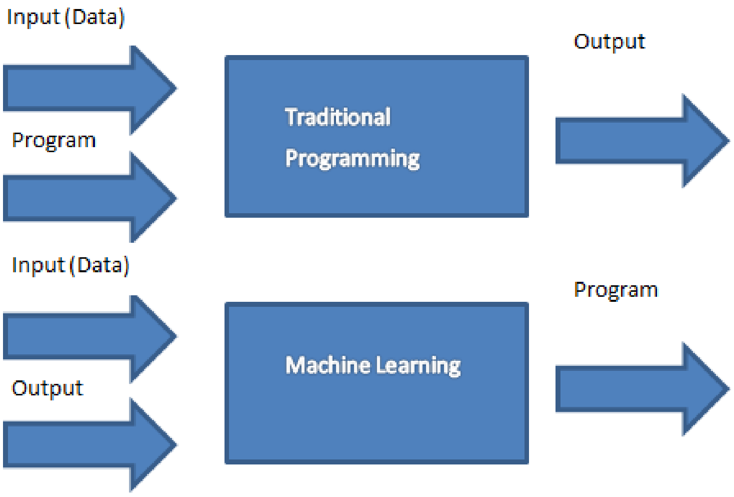


Figure 2.1: Traditional programming vs machine learning.

2.3.1 Decision Tree

Decision tress is a very strong and powerful algorithm which is used for the classification and the predictions. This algorithm is known as supervised machine learning models.. It is very impactful and used for both continuous and the categorical values. As shown in the figure 2.2 decision tree has levels and nodes every node is carrying some value and the weight.

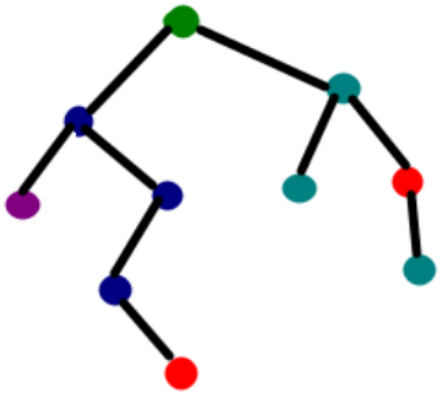


Figure 2.2: Decision tress .

2.3.2 Random Forest

Random forest (RF) is also an algorithms used in supervised machine learning for the classification and regression problems. Output in random forest is calculated using the majority voting method. In case of regression output is calculated using mean of all the outputs which known as aggregation as described in the figure 2.3 There are few methods related to ensemble from which random forest is one of and mighty helpful for the classification problem as well as for the regression problems. It used many decision tress which is called bagging. The features are selected randomly from the dataset for sample dataset in the model known as bootstrap.

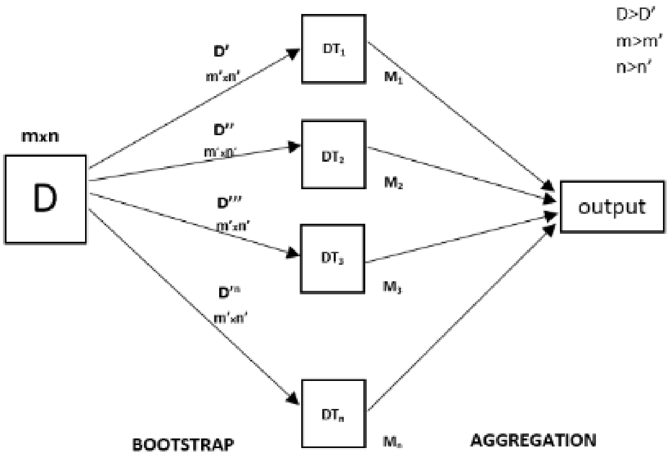


Figure 2.3: Random Forest.

2.3.3 Logistic Regression

Logistic regression (LR) is basically an algorithm used related to the supervised machine learning tasks. It is used in the classification problem where Y is the target value which is considered as a discrete value on the other hand the input is X.

Logistic regression(LR) is homogeneous to the linear regression as linear regression uses the sigmoid function which is given in the figure 2.4 below.

$g(z) = \frac{1}{1+e^{-z}}$

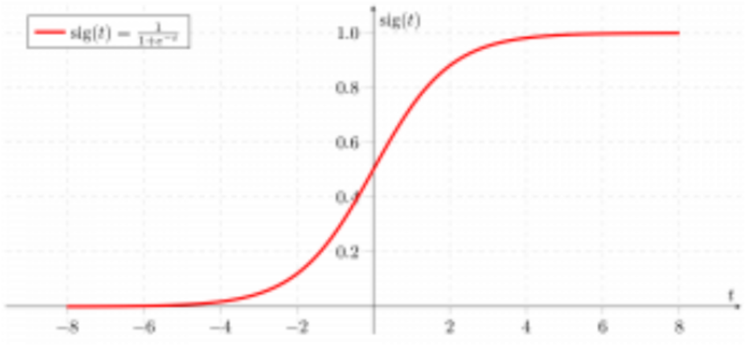


Figure 2.4: Sigmoid function.

Logistic regression can be used as classification to solve the problem. It can only happen when there is a threshold which is defined. Hence the decision which has to be made for the threshold is dependent on the precision and recall. In ideal case the precision and recall is 1.

2.3.4 XGboost

Xgboost algorithm is a supervised machine learning algorithm which is used as an implementation of gradient boost decision trees. Decision trees are sequential in this algorithm. Weights play an important role in this algorithm. All the independent variables are assigned different weights. Which then are fed to the decision trees and then results are predicted. If the tree predicts the wrong weight then its weight is increased and then fed into the next decision tree. It is used for all the problem solving condition either it is classification or the regression. Mathematical form of the model is given below

2.3.5 SVM (Support Vector Machine)

It is a supervised machine learning algorithm with is used for classification. In some cases it can be used as a regression depending on the problem we have to solve. If the problem is to find the find the hyper plane in the data set depends on the number of features. If there are two dimensions or the features then it haves the line as a result of that. If there are more than two or three features lets say it will give the two dimensional hyper plane. Supposing two variables like x1 and x2 as independent variables and also a dependent variable considering as blue filled circle as well as red colored circle in the figure 2.5

As it is clearly shown in the figure that there are many line of hyper plane which are dividing the two circles blue and reds. It is because we have only two input of dataset or number of features are only two. The best hyper plane is chosen on the basis of biggest margin between the two classes shown in figure 2.6.

Now as seen in the figure the hyper plane with the biggest margin is l2 hence it is selected. Consider the below scenario as given in figure 2.7 .

Svm is pretty much pro active for the outliers. It ignore the outliers quickly and adjust them accordingly to find the best suited hyper plane which maximizes its margin figure 2.8

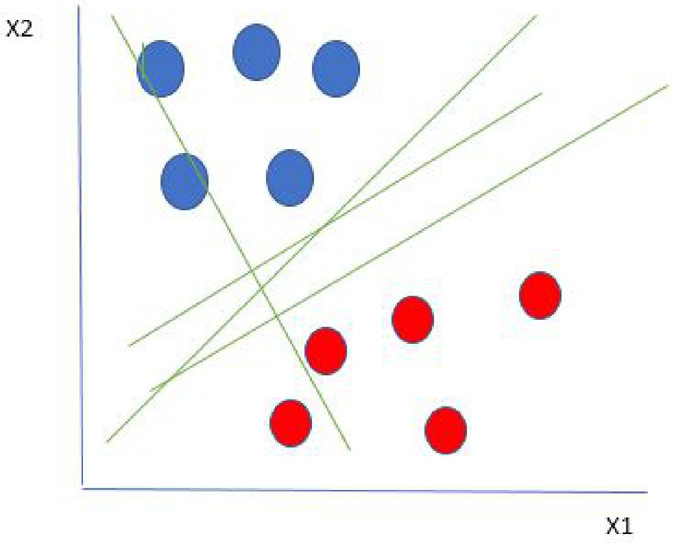


Figure 2.5: XGboost 1.

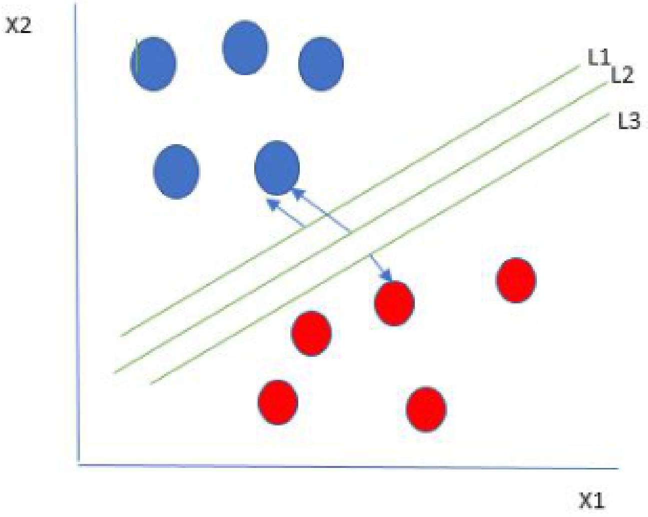


Figure 2.6: XGboost 2.

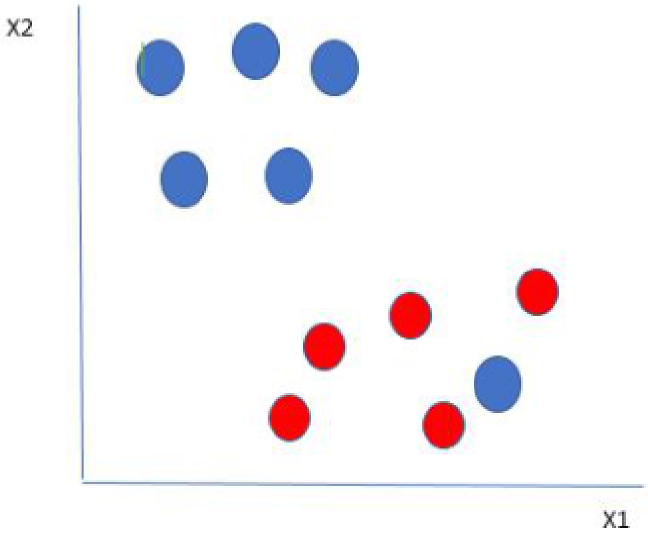


Figure 2.7: XGboost 3.

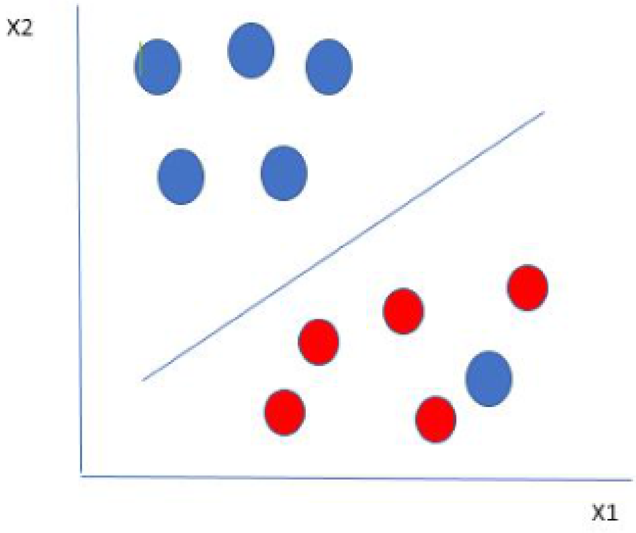


Figure 2.8: XGboost 4.

2.4 Supervised Machine learning

As it is indicated from the name supervised machine learning is all about the learning of the models on the data which is given. First data is given as an input then these models are trained on different models for the required results. For example there is given different types of fruits.

- First the model will learn about the shapes of the fruit.
- Then the shape of the fruit is defined.
- After that the model will be labeled with different name regarding to the shape and color of the fruit.

Now the machine learning model will learn from the data given and give the results according to it. If there is new entry or query it will definitely not identify.

2.4.1 Classification

Classification is used for the prediction of class in the given dataset.It is also know as labels or the categories.The classification is used when the problem is about “yes or no”, “disease or no disease”. The classifiers uses the training data to known the relation with class. There are many example for the classifiers likes of spam emails etc. There are two types of classifiers.

- Lazy Learners:It stores the training data and then waits the testing data to come and then evaluates the results on the basis of most related data in the training set. It takes a lot time for the prediction as compared to the eager ones does. Knn is the most famous lazy learner classifier used in machine learning.
- Eager Learners:Supervised learning algorithm which is more fast than the lazy learners it takes the training data and evaluates the results. As compared to lazy leaner it takes a lot of time for the training of data but less time for the prediction of the results. Decision trees and artificial neural networks are one of the famous eager learner classifiers in machine learning.

2.4.2 Regression

The evaluation of relationship among two different variables on is dependent and others is independent; this refers to the regression. Regression is used in the machine learning for the prediction purposes. In regression the model is trained first on the given data and then the results are predicted for the future. There are many types of regression but linear regression and the logistic regression are used frequently in the machine learning for the predictions. Linear regression is used for fitting the hyper plane(the 2D data points in x axis and the y axis). Sum of mean squared error is calculated and then the minimization is done. On the other hand the logistic regression is used for the classification problems for the results. It uses the sigmoid function with the threshold to identify the classification between the two classes. It has two benefits.The process of regression is given in the figure 2.9

- First you can check the missing values in the dataset.
- Secondly it can evaluate the future data which is not given.

There are few advantages and also the disadvantages of regression is well which are given below.

Pros

- It works on the past experience and produces the output on that basis.
- Supervised learning can be used for solution of many problems.
- It can optimize the solution on the past experience.

Cons

- When dataset is big it is difficult to classify.
- It is very time taking process. It takes a lot of time for computations.

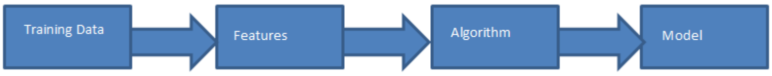


Figure 2.9: Regression.

2.5 Unsupervised Machine learning

Unsupervised learning in machine learning does not work on the previous experience and guidance. Data is not trained as it is trained in the supervised learning. It works on the given data and its similarities in it prior to the training of the data. The unsupervised learning has two types.

2.5.1 Clustering

Clustering is unsupervised machine learning technique in which data is divided in the groups. Grouping is based on the similar number of features or their non similar behaviours. The purpose of the clustering is to do the statistical analysis of the data. In clustering data is dived on the basis of the previous liking behavior. Like the customer purchasing behavior. There are many types of clustering is used in the machine learning. K-means is a frequent clustering which is used. In k-means clustering data is divided into different groups.Then the center points on the groups are initialized. the distance between the center point and the data point is calculated. After that the mean of all the center point is calculated. This process is repeated until for many iterations until the best results come out. K-means clustering is very fast algorithm it has the complexity of O(n). Expectation minimization is the other clustering technique which is also used in machine learning. This technique performs well where k-means is not workable like when the mean of the data centers is very close. The EM works on the guassian distribution which is far more flexible than the k-means. So the data groups can be of any shape. On contrary with k-means uses the means of the data centers based on the circles EM uses the probability or the standard deviation. So it does not matter that the data is in cilcular form or the eclipse form or something else.

2.5.2 Association

Association is very important term used in the machine learning. To understand this terms lets talk about the super mart where all the items are grouped in different section like vegetables crockery and many others. So when a customer buys a product he can also see the relevant products that he might have to purchase. Same way in association works on the concept of buying behavior of the customer where From the big data set relevant products are suggested to the customer to increase the sale. Association is based on the behavior of the previous liking of the customers like if customer like X he might like Y too. For example the bread and the egg have big association with the milk. So if the customer has purchased bread it is possible that he will purchase the other items too. The metrics used for the association are confidence , support and lift. Support is basically the how frequent an item is purchased. For example if there are two item like bread and other is brush. Now we can see that the bread has the high percentage of purchases rather than the brush so its support is high. Lets look other example of item bread;milk, second item is bread;brush. It is obvious that the bread and the egg are more frequently purchased unlike the bread and the brush. Confidence is the possibility of chance that the item will be purchased. Like we can say that if bread is purchased then the likeliness of purchasing the milk is high. On the other hand likeness of the bread with brush is uncertain. Lift is the calculation of the support with probability of Y when X is given.

2.6 Binary Classification

Binary classification is the type of classification which has only two classes. There are many examples of the binary classification. Like it identifies the types of the behaviors either there is spam or not-spam or disease or no-disease or churn or no-churn. Hence it works fro those problems which has to be answers in yes or no. There are many classification algorithms are used for binary classification like logistic regression, decision trees and support vector machines which are most commonly used in machine learning. There are some algorithms that can only used for the binary classification or have only two classes but cant use for more than two classes like support vector machine and logistic regression.

2.7 Multiclass Classification

As compared to the binary classification multi classification does not have the two classes or out puts in the form of yes or no. But it contains more than two classes. For example if there is a problem of classification with the data of many different images is given. So there are many classes in the picture data set. If one picture of one class is identified from hundreds of pictures from different classes in face recognition system. There are many machine learning algorithms are used for the multi class classification problems. Random forest , decision tree and the naive bays are most commonly used multi classification algorithms. These algorithms can also be used for the binary classification problems as well. The following figure 2.10 shows the difference between the binary and multi classification.

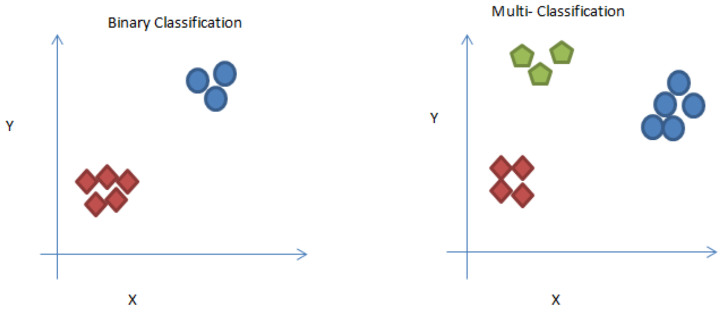


Figure 2.10: Difference between Binary and Multi-classification.

2.8 Feature Selection

Feature selection has a large importance in machine learning. Because all the features in the dataset are not relevant or important while using them for better results. Hence all those features are filtered out and those feature are selected manually or automatically through python libraries. It increases the results. It has following benefits.

- It Reduces over fitting of the selected data.
- It improves model's accuracy and gives better results.
- It reduces the time for training the data.

- Three types of feature selection techniques which can be used to choose the important features in the dataset.

2.8.1 Filter Based

In Filter based method features are selected statistically. Then they are trained on the machine learning algorithms. It filters out the irrelevant feature from the data on the basis of ranking of the each feature. It uses very little time for the computations and doesn't over fit is well that's why are good. Different types of filter based techniques are given below. Information gain: Information basically reduces the entropy in the data set. Information gain is calculated for all the features that are selected on the basis of our target variable which is defined. Like label is our target variable. Chi-square: Chi-square is calculated on the features when they are in categorical form. All those features and he target features chi-squares score are calculated. Those features which have high chi-square values are considered for the application of machine learning model. Fisher score: Fisher score is used a lot in machine learning. It calculates the ranks of the features. All the features with good score are considered. Missing value ratio results is also used in machine learning algorithms.

In filter based technique set of features are given. From these set of features important features are filtered out. Then after that machine learning algorithms are applied on these features so that the performance of the model is estimated. The figure 2.11 shows how the filter based feature selection technique work in hierarchy.

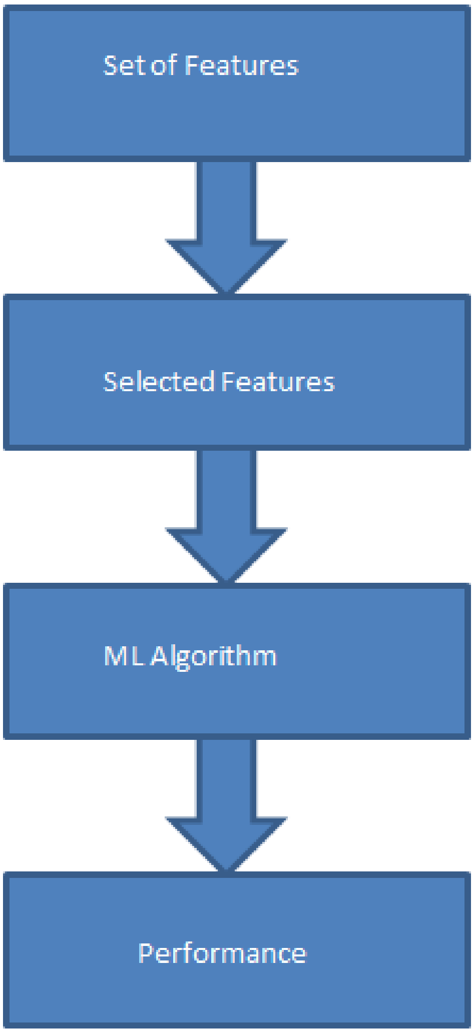


Figure 2.11: Filter based.

2.8.2 Embedded

Embedded has both features of filters and the wrapper based techniques. This method is very fast in computations but its accuracy is better than the filter. Embedded based techniques works on the iterations. In every iteration during the training of the data it carefully fetches those features from the training set which have more importance. Lasso regression and random forest are mostly used in embedded based techniques. In lasso1 it reduces the freedom o the machine learning models. For that purpose it adds the penalty on the different parameters of the machine learning algorithms. In embedded based technique set of features are selected first. Then the subset is generated with the features . machine learning algorithms are applied and their performance is calculated. This process is done in iterations until the best performance results are calculated.Figure 2.12 describe the layout of the embedded based technique.

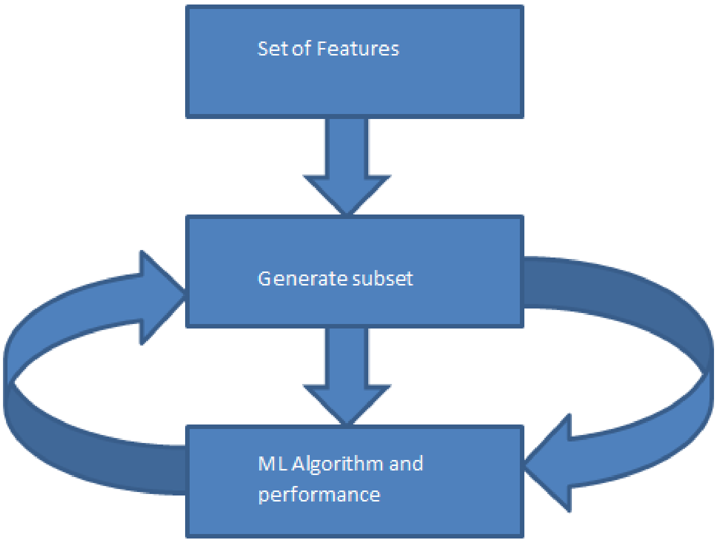


Figure 2.12: Embedded based.

2.8.3 Wrapper Based

In wrapper based method features are selected on the basis of different combinations which are made and then these combinations are compared with each other to select the best one and applied in the model. Then the training of these features is done one by one. There are three types of the features which are used in wrapper based methods.

Forward Selection: It starts with the empty string, then one features is added it checks the performance of the model and then new feature is added on basis of that features is improving the performance of the model or not and so on.

sklearn.feature_ selection.SequentialFeatureSelector(estimator,*, n_ features_ to_ select='warn', tol=None, direction='forward', scoring=None, cv=5, n jobs=None).

Backward Selection: This approach is opposite to the forward selection method. It starts with all the features and performance of the model is checked. Then those features which are less important are removed to increase the performance of the model.

from mlxtend.feature selection import SequentialFeatureSelector as sfs(k features=n, forward=False, verbose=1, scoring=none)

Exhaustive feature selection: This feature selection method uses brute force mechanism to select the features. It makes the best set of features using this method. In python exhaustive features are selected using mlxtend library

from mlxtend.feature selection import ExhaustiveFeatureSelector as EFS

Recursive Features Elimination Method: Recursive feature selection method uses the greedy search approach to select the features from the dataset, then recursively the important features are selected on the basis of its importance and feature subset is narrowed down. The figure 2.13 is shown which describes that the set of features are selected from the data set and the machine learning algorithms are applied to get the performance of the model.

In python RFE is selected from sklearn library

from sklearn.feature selection import RFE

sklearn.feature selection.RFE(estimator, *, n features to select=None, step=1, verbose=0, importance getter='auto')

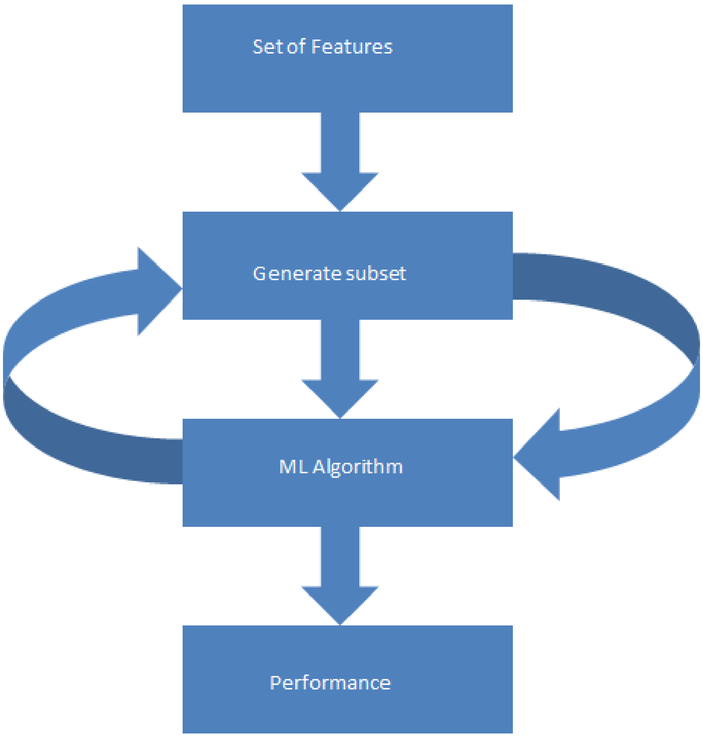


Figure 2.13: Recursive feature elimination.

2.9 Data Normalization

The machine learning algorithms like RF, DT, SVM, LR have bigger impact on the numerical values. There are many techniques which are used for the data normalization like normalize from sklearn, minmax scaling and decimal scaling. The equation for min-max scaling is given.

$$X_{sc} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2.9.1}$$

The dataset given has features which are represented by

D $(\mathrm{F}_1, F_2, F_3, \ldots\ldots\ldots, F_n)$

where 1<n<N. N is the all count of features present in the dataset.

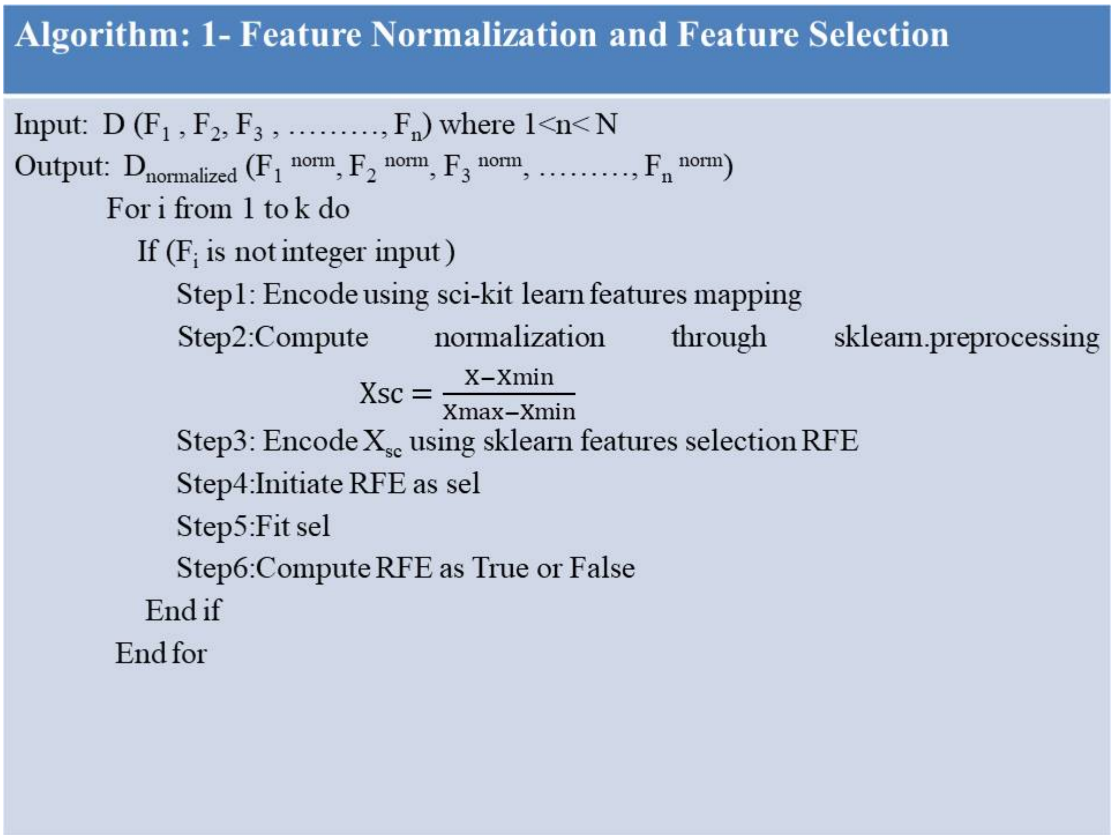
$$d_j = (w_{1j}, w_{2j}, \ldots\ldots.., w_{tj}) \tag{2.9.2}$$

$$q = (w_{1q}, w_{2q}, \ldots\ldots.., w_{nq}) \tag{2.9.3}$$

Where dj is number of features in the document and the q is the query or the predicted accuracy of the dataset. The equation for the normalization is given below.

$$
\begin{aligned}
Normalize &= \check{\text{N}}(d\_j, q) \\
&= \frac{\sum_{i=1}^{N} w_{i,j} w_{i,q}}{\sum_{i=1}^{N} w_{i}^{2},_{j} \sum_{i=1}^{N} w_{i}^{2},_{q}}
\end{aligned}
\qquad (2.9.4)
$$

After the normalization all the features are selected using recursive features elimination from sklearn feature selection which gives the best features in the features space showing true or false for the selected features after fitting the model. The algorithm is given in the figure 2.14



Algorithm: 1- Feature Normalization and Feature Selection

```
Input: D (F_1 , F_2, F_3 , ………, F_n) where 1<n< N
Output: D_normalized (F_1 ^norm, F_2 ^norm, F_3 ^norm, ………, F_n ^norm)
      For i from 1 to k do
        If (F_i is not integer input )
           Step1: Encode using sci-kit learn features mapping
           Step2:Compute   normalization   through   sklearn.preprocessing
                        $Xsc = \frac{X-Xmin}{Xmax-Xmin}$
           Step3: Encode X_sc using sklearn features selection RFE
           Step4:Initiate RFE as sel
           Step5:Fit sel
           Step6:Compute RFE as True or False
         End if
       End for
```

Figure 2.14: Feature Normalization and Feature Selection.

2.10 Random Forest

This machine learning classifier is an ensemble method. It has many trees that are built to get best possible results. It uses begging and boosting technique.

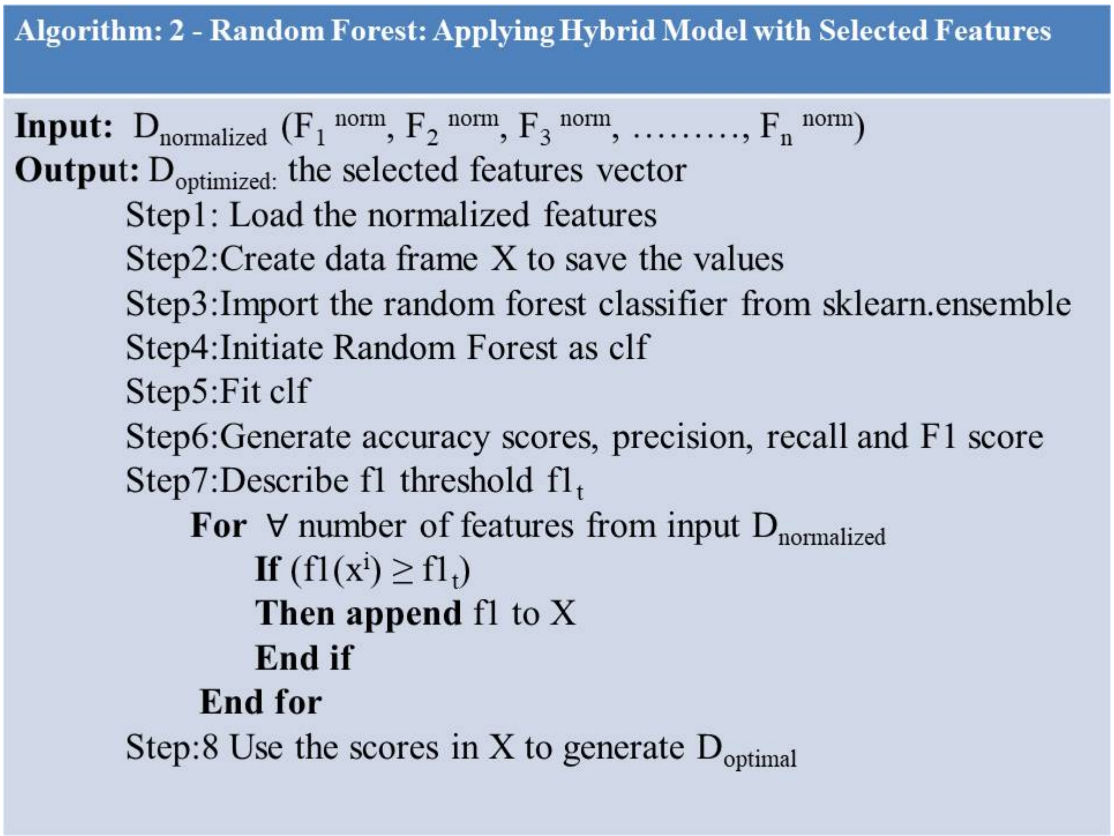
For data X = $\mathrm{x}_1, x_2, x_3, \ldots\ldots x_n$

For response Y = $\mathrm{x}_1, x_2, x_3, \ldots\ldots x_n$

It repeats the begging to B from b=1 and x' is calculated by the average predictions.

$$j = \frac{1}{B} + \sum_{b=1}^{B}(f(x')) \tag{2.10.1}$$

The algorithm of the random forest is given in the figure 2.15. Which shows the method using the sklearn library.



```
Algorithm: 2 - Random Forest: Applying Hybrid Model with Selected Features

Input: $D_{normalized}$ ($F_1^{norm}$, $F_2^{norm}$, $F_3^{norm}$, ........., $F_n^{norm}$)
Output: $D_{optimized:}$ the selected features vector
        Step1: Load the normalized features
        Step2:Create data frame X to save the values
        Step3:Import the random forest classifier from sklearn.ensemble
        Step4:Initiate Random Forest as clf
        Step5:Fit clf
        Step6:Generate accuracy scores, precision, recall and F1 score
        Step7:Describe f1 threshold $f1_t$
            For $\forall$ number of features from input $D_{normalized}$
                If ($f1(x^i) \geq f1_t$)
                Then append f1 to X
                End if
            End for
        Step:8 Use the scores in X to generate $D_{optimal}$
```

Figure 2.15: Random Forest: Applying Hybrid Model with Selected Features.

2.11 RFE (Recursive Feature Elimination)

Recursive Feature Elimination is a not a filter based, it is feature selection techniques known as wrapper based which can be used using sci-kit learn library in python. This technique is pretty much different from the filter based techniques. Different type of machine learning algorithms can be used using RFE. When the desired number of features is given, it selects the features from all the features on training dataset. The model is fitted and it continues unless the features are selected. This approach is very efficient for eliminating the features or irrelevant features from a large feature pace and when data is very sparse to gain the maximum accuracy for better prediction. RFE works on the predictors which are fitted to the model first. The base of these predictors in the raking is its importance which it has to the model being used. Let P is the number of values used in a predictors where (P1>P2) and so on. Where Pi is the most ranked predictor which selected and fitted to the model after those iterations goes on and the model is refitted and so on. The following equation is about Friedmans1 used to test the algorithm.

$$y = 10sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + N(0, \delta)^2 \tag{2.11.1}$$

Where n is the number of patterns, sd is the standard deviation, x is the input values in data frame which are independent variables and y is the output values which denotes dependent values.

2.12 Summary

The literature review is concerned with the overview of all the work related to that field previously. It contains different machine learning algorithms used by different authors for many years so that improvement in the accuracy of the intrusion detection systems is done. Machine learning algorithms which include the supervised as well as the unsupervised machine learning algorithms are adopted by the researchers in their work. Supervised machine learning algorithms which are applied when the data is given are trained on the dataset in order to achieve the results. Is consists of two types which are classification and the regression. It depends on the problem given. On the other hand unsupervised machine learning algorithms includes the clustering, association. Binary classification is used when the given problem results must be in yes or no or have only two classes. In our case the researchers have used binary classifications as intrusion detections systems are based on this classification method to get the accuracy of threats detection. On the other hand multi-classification is also used when there are more than two classes in the output Feature selection is very important in the intrusion detection systems. Hence many researchers have applied different kind of techniques to improve the accuracy of their model. Feature selection methods reduces the irrelevant features from the datasets so that the machine learning models work more efficiently for large and sparse datasets. Recursive feature elimination method is one of the most effective used by many authors as well as many others. This portion includes the different machine learning algorithms as well as the feature selection techniques used by the different authors.

Chapter 3

Proposed Methodology

3.1 Introduction

In this thesis we address the steps taken in order to build the intrusion detection system using the supervised machine learning algorithms using the UNSW NB-15 dataset. There are few questions to be answered in this section is given below.

- How the machine learning classifiers used to increase the accuracy of IDS system?
- Importance of feature selection method RFE in defining the features?
- Comparing the machine learning algorithm on the defined data set?
- Which features can give the maximum results?
- How can these results be further applied?

Now the answers of these questions are given in details. In section 1 all the possible goals are determined regarding to the data set and the machine learning algorithms. The details about the knowledge of machine learning models are discussed is well. In section 2 data is gathered from the source on which the machine learning techniques are applied. All the features in the data set are discussed. In section 3 data is reprocessed. All the important features are selected. In section 4 the machine learning classifiers are chosen. In section 5 the results are compared between the different classifiers that are applied in previous section to analyze the results. Now the task is divided into parts.

- In first part all the pre processing of the data set is done. In which all irrelevant features are removed. After that the process of normalization is conducted where all the null values from the data is removed and cleaned form of the data is created.

- secondly features are selected through the features selection methods.
- In third phase all the features which are selected are converted into data frame. After that machine learning classifiers like decision trees, random forest and others are implemented. In the end the results from the models are calculated and recorded in the form of tables to compare the accuracy of the models.

3.2 Goals Identification For Data and ML Classifiers

The basic goal in this section is to determine the accuracy of machine learning algorithms in intrusion detection system. The data is collected which is in raw form. This section contains these research question.

- How does the machine learning algorithms evaluates the accuracy on the given dataset?
- Second is the accuracy of the algorithms increase with the feature selection methods which are available on given dataset or not?
- Does sparsity of data affects the accuracy of machine learning algorithms?

The answers of these question can be split in few steps. In first step the machine learning are defined and their accuracy of intrusion detection system is related to first research question. With addition to that the data collection is also related to that question is well. In second step we make the complete framework with collection of data from the data source. As well as the features selection method is applied to increase the accuracy of the machine learning classifiers. To address the third question the data cleaning is done in the next section and all the irrelevant features and null values are removed. Next section involves the results created after the accuracy of these models in this thesis work.

3.3 Data Collection

In order to calculate the accuracy of intrusion detection using machine learning classifiers it is important that the data is gathered in the raw form. This section contains the few question about the data collection.

- Which type of data is required?
- Does this data fulfil the requirement or not?
- How many data files are available?
- What is the real source of data?

In machine learning classifiers the data is used which is of two types. Training data and the testing data. The data is converted in these two types. Also it is clear from the previous sections that our main focus in this thesis is about the binary classification. Hence the data is gathered which have important features and converted into the new data file. The data set has 49 features and nine classes. Raw data is collected from the Australian lab. There four data files which are combined to form the one data file. The data which is used in the thesis for the intrusion detection is integer type of data.

3.4 Choice of Features

This section of thesis is consisted of the data pre-processing. The features from the data set are taken which will take part in the next section for the machine learning methods. Following question will be addressed in this section.

- Which features are selected for proposed model?
- Which type of data is required?
- What to do in case of sparse data?
- How to handle multi type of data ?

In this thesis we are focused on the maximum accuracy for the intrusion detection system hence those features are selected which will results in to the best accurate results. Those features are selected which have the integer data type. Moreover the null values in the data is removed so that the accuracy can be improved. Those features which have different type of data is converted into through the type conversion.

3.5 Specification of a Machine Learning Approach

This section includes all the machine learning approaches which re used for the propose work are discussed. The algorithms which have chose are picked up. It has following question to be answered in this section.

- How many machine learning classifiers are used?
- How the problems are countered during execution?

Comparison between the different algorithms is done on the basis of which algorithms gives the best accuracy. So the algorithms which gives the best accuracy will be selected for our propose work in this thesis. Total five classifiers are used for the experiments. The algorithms which are used are Decision trees, Random forest, Support vector machine, Logistic regression and the xgboost classifier.

- Decision Tree: As discussed in the section 2.3.1 the decision tree is a supervised machine learning algorithm. This classifier works very well. It is very robust algorithm for the outliers and the missing values in the data set. Moreover it has a disadvantage due to its complex structure. Also it seems to be very unreliable because of the change in its structure whenever there is an addition of a data into it. It is becomes more complex whenever the amount of different parameters are added during the algorithms are processed.
- Random Forest: Random forest(RF) method is related to supervised machine learning which is described in the section 2.3.2. Random forest consists of many different trees. Hence every tree in the random forest predicts a class. So the class which has most votes become the prediction for the model. Random forest is very simple but a very powerful algorithm too. To work well the random forest consider the two things. first of all there must a low correlation between the features which are selected to get the maximum results in your favor. Secondly Features which

are selected should have signals which indicate their relation to some extent so that the random forest classifier selects those features rather than selecting the random ones from the features space.

- Support Vector Machine: This classifier is also an supervised machine learning classifier which is discussed in the section 2.3.5. This algorithms is as effective in regression as well as with classification. As compared to the other classification techniques support vector machine required more deep knowledge about the kernal and its testing processing. It contain different types of the kernal which is selected while training the model. It works according to the required kernal which is selected.

- XGboost: Xgboost is an other supervised machine learning technique which is used in this thesis work. It is explained in the previous section 2.3.4. It uses the decision tree while every decision tree carries the different type of weights on them. These decision trees work together to give the predictions. On giving the wrong prediction value the weights are reassigned to them. It can also be used to solve other problems like regression.

- Logistic Regression: Logistic regression(LR) is a supervise machine learning algorithm. As discussed in the section 2.3.3 it is a classifier it can be used for the regression problems as well. Logistic regression uses the sigmoid function. threshold is given in case of the logistic regression is used. Threshold for precision and recall is 1 in ideal scenario.

As given above the approaches which are used; applied one by one. The classifiers which gives the more accuracy is selected to give the outcome for the given data set. Moreover for building the model the classifer is tuned with the different types of parameter and then the model is fitted. Random forest is selected to implement on the data set. There are four number of data files. These files are combined. Then the features are selected. AS random forest works very well and data set is huge. So there is need to have greater RAM in order to run this algorithm. Other models like support vector machine uses more RAM in PC and takes more time for the evaluation. Likewise decision tree also take time due to the large number of tree nd their calculations time.

3.6 Interpretation

All the work in this section consists on the machine learning algorithms applications on the UNSW NB-15 dataset. Machine learning algorithms which are supervised machine learning algorithms like Random forest, decision tree, support vector machine with others xgboost are used. All the experiments are coded in the python using google collab. First of all the dataset is cleaned and null values in the dataset is removed so that the refined dataset is used for experimentation. The purpose of the data cleaning is done to achieve the higher accuracy. Dataset contains nine classes which are given in the section below. All the tables of results are also given. The figure which includes the graphical layout is also described in the results and experimentation portion. Accuracy of all the machine learning classifier is calculate like training accuracy, testing accuracy, precision, recall and f1 score.

3.7 Hybrid Model

In proposed research work, first of all the dataset is cleaned and combined, then normalized form of the dataset is fitted into data frame. The relevant features selected to be used. As all the features are not important, hence irrelevant features are filtered out. For training the data the label column is used. Total features which selected are twelve, which then divided into 25% for testing and 75% for training the model. The proposed methodology is described in figure 3.1.

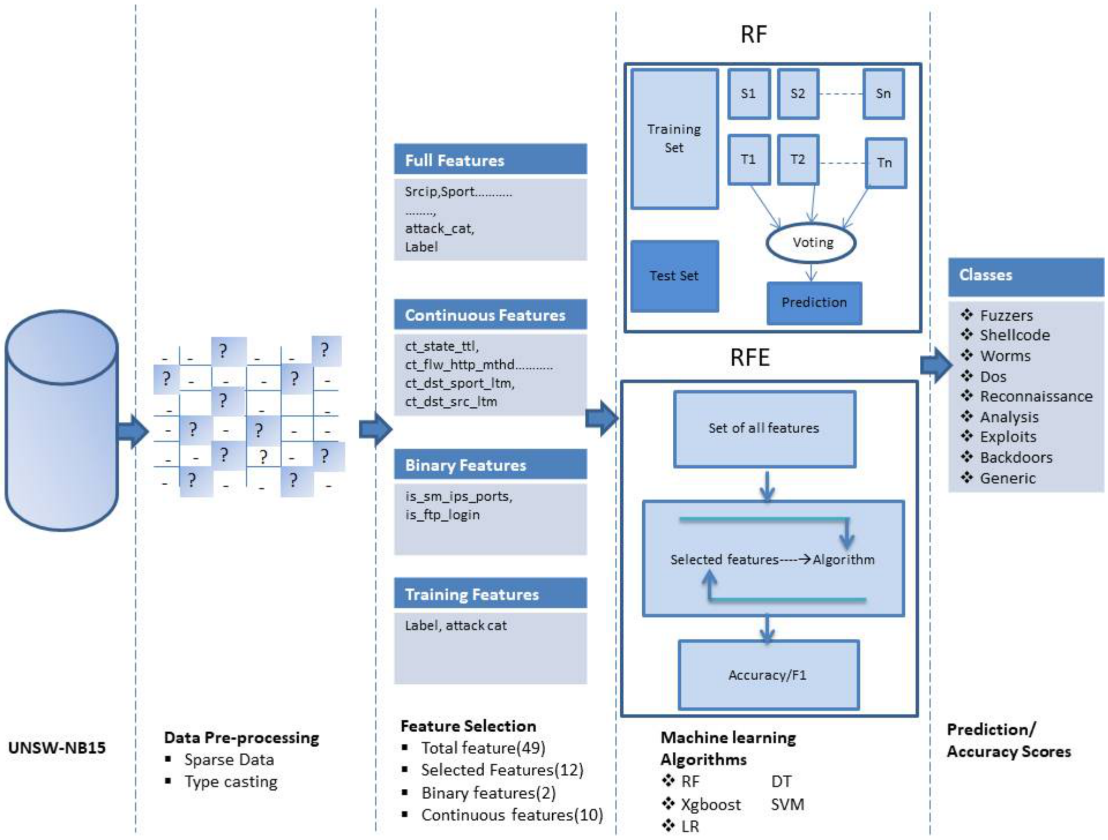


Figure 3.1: FSRF: Proposed Hybrid model using feature selection RFE and Random Forest.

3.8 Summary

In the proposed framework a hybrid technique is used in order to get the novel results. Supervised machine learning classifier random forest is used with the recursive features elimination which is wrapper based technique which reduces the number of features from the UNSW NB-15 dataset which is very sparse. So irrelevant features are eliminated and all the null value from the dataset is removed and 12 best features are selected which are then normalized. After the normalization the data is split into training and testing datasets which 75% and 25% respectively.

Chapter 4

Experiments and Results

4.1 Introduction

This chapter gives the complete overview of all the results which are achieved and the data which is used to achieve those results is given to answer the research questions. On the other hand section 4.1 will address all the experiments which are used and the next section 4.2 describes the results achieved from those experiments is discussed.

- How many features are used?
- Tools which are used for the metrics?
- Which algorithms is most influential?
- Which algorithms achieved better results?
- Which feature selection technique is adopted?

4.2 Empirical Data

There are 49 numbers of features exists in the dataset as given in table 4.1. The Cyber Range Lab of the Australian Centre for Cyber Security (ACCS) is the place from where the dataset is gatherd. Nine kinds of attacks are included in the dataset which has shellcode, analysis, DoS, backdoors, reconnaissance, generic, exploits, worms and fuzzers. There are four files of the dataset which are combined to form one file.

Serial No	Feature name	Serial No	Feature name
1	Srcip	26	trans_depth
2	Sport	27	res_bdy_len
3	Dstip	28	Sjit
4	Dsport	29	Djit
5	Proto	30	Stime
6	State	31	Ltime
7	Dur	32	Sintpkt
8	Sbytes	33	Dintpkt
9	Dbytes	34	tcprtt
10	Sttl	35	ackdat
11	Dttl	36	is_sm_ips_ports
12	Sloss	37	ct_state_ttl
13	Dloss	38	ct_flw_http_mthd
14	Service	39	ct_ftp_cmd
15	Sload	40	is_ftp_login
16	Dload	41	ct_srv_src
17	Spkts	42	ct_srv_dst
18	Dpkts	43	ct_dst_ltm
19	Swin	44	ct_src_ltm
20	Dwin	45	ct_src_dport_ltm
21	Stcpb	46	ct_dst_sport_ltm
22	Dtcpb	47	ct_dst_src_ltm
23	Smeansz	48	attack_cat
24	Dmeansz	49	Label
25	synack	0	0

Table 4.1: Features of UNSW-NB15 dataset.

4.2.1 Classes

The dataset includes nine types of classes in it. Table 4.2 shows these classes. Every class has different number of values.

Type	Value Count
Fuzzers	19195
Shellcode	1501
Worms	174
Dos	16353
Analysis	2677
Exploits	44525
Backdoors	1795
Generic	215481

Table 4.2: Values Count of UNSW-NB15 dataset Classes.

The values count for all the classes is shown in the form of bar chart in figure 4.1.

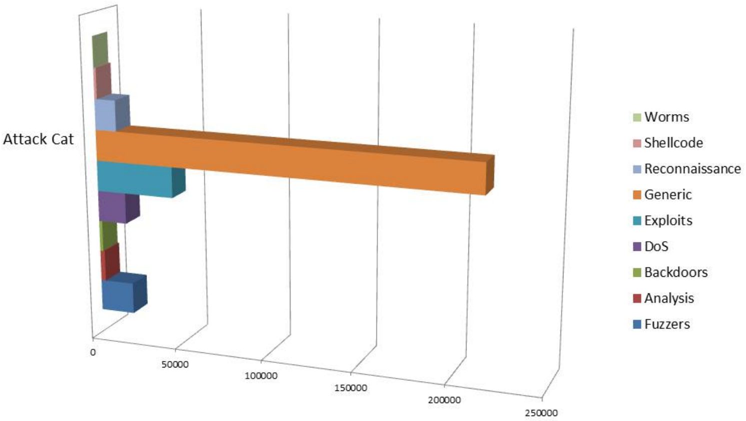


Figure 4.1: Bar Chart of the classes used in UNSW NB-15 data set.

4.2.2 Environment (Hardware and Software)

All the experiments are done in python 3 with Anaconda 3 in googleColab. 8 GB RAM with windows 10 on Core i7 PC is used in this research work. The libraries used in this model are sklearn, pandas and numpy. For plotting the graphs matplotlib library is used.

4.2.3 Metrics for Performance

Intrusion detection systems (IDS) is built in such a way that it has to predict the attacks coming from outside world as quickly as possible. To maximize their prediction machine learning algorithms like random forest, decision tree and others uses accuracy metrics to maximize. There are few equations given for accuracy, precision, recall and F 1 scores calculation.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.2.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.2.3}$$

F1 score is normally used for calculating the average of recall and precision or the harmonic mean of recall and precision.

$$F1 = 2.\frac{Precision.Recall}{Precision + Recall} \tag{4.2.4}$$

4.3 Results

For the proposed model the selected binary columns are Is_Sm_Ips_Ports and Is_Ftp_Login. In continuous columns Ct_State_Ttl, Ct_Flw_Http_Mthd, Ct_Ftp_Cmd, Ct_Srv_Src, Ct_Dst_Ltm,Ct_Src_Ltm, Ct_Src_Dport_Ltm, Ct_Dst_Sport_Ltm and Ct_Dst_Src_Ltm are selected.

The Ct_Flw_Http_Mthd is feature which has float values so this float value of this feature is converted into integer type through type conversion. After normalization of data, the data is split into set of train and test split using sklearn library importing train test split method from sklearn. The distribution of data is given in the figure 4.2 for training and testing.

The classifier from scikit-learn library is then imported to train our model and fit it. After fitting the model evaluation of the model performance through accuracy score is done. Also calculated the F1 score, recall and precision. The machine learning model used for the research are Decision tree with model accuracy of 99.68%, Logistic regression has accuracy 98.93%, XGboost is 99.63% accurate , Support vector machine has 99.65%

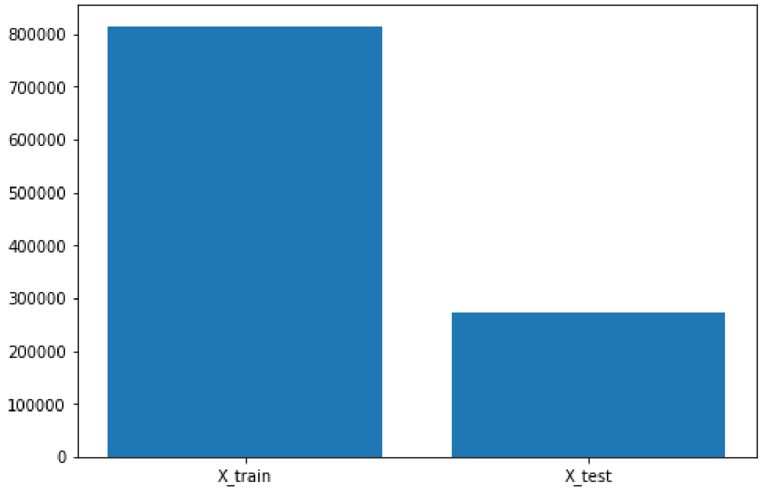


Figure 4.2: Data Distribution for Training data and Testing data.

accuracy and Random forest has 99.69% model accuracy. The performance of these models one by one is measured using all the features that are reduced from 49 features to 12 best features.

The comparison of different classifiers for the binary classification is shown in the table 4.3. Binary classification means that the accuracy of the model is either true or false. Hence these algorithms of machine learning are adopted to check the best performance between these models; it is found that Random Forest is proved to be outperformed in terms of accuracy which is 99.69% for attacks.

ML Method	Training Accuracy	Testing Accuracy	Precision	Recall	F1 Score
DT	99.73	99.68	88	97	92
LR	98.96	98.93	80	64	71
XGboost	99.66	99.63	87	97	92
SVM	99.67	99.65	87	98	92
RF	99.74	99.69	88	98	93

Table 4.3: Binary Classification comparison between Machine Learning Models .

The table proves that the random forest is showing more accuracy for training and testing as well as recall, F1 score and precision. The confusion matrix of the random forest classifier is given in the figure 4.3

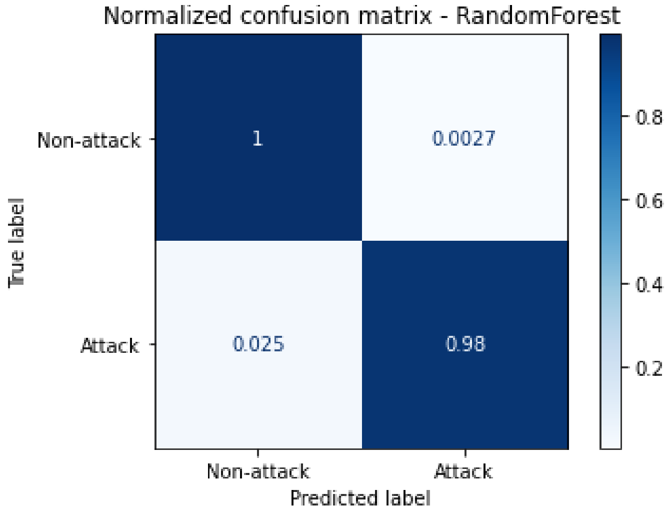


Figure 4.3: Confusion Matrix for Random Forest for Binary Classification.

4.3.1 Feature Selection Method

Wrapper based technique is used for the feature selection in our proposed model which is RFE (recursive feature elimination) from sklearn library. The fusion technique is applied to combine the RFE using different classifiers to select best machine learning classifier to detect the intrusion of the attacks. For this purpose the algorithms of machine learning learning used are Decision tree, logistic regression, XGboost, support vector machine and Random forest. 25% from the dataset is test dataset and 75 % for training purpose. First of all data is cleaned and normalized. After that all the null values in the dataset are removed. Then train test split using sklearn liibray is applied on the dataset. The ensemble methods from sklearn is imported and get fitted to the model. After fitting the model the recursive feature elimination is implemented on the data frame and selected different number of features to check the accuracy of the different models. These experiment results shows that DT has accuracy score of 99.69%, LR has model accuracy 98.94%, XGboost results 99.66% for the model, SVM has 99.65% and Random forest 99.70% which is best accuracy so far in feature selection method. Different machine learning classifiers used for analysis which has all 12 features were selected using feature selection method RFE is shown in the table 4.4

ML Method	Training Accuracy	Testing Accuracy	Precision	Recall	F1 Score
DT	99.74	99.69	90	95	93
LR	98.94	98.93	85	58	69
XGboost	99.66	99.64	86	99	92
SVM	99.67	99.65	87	98	92
RF	99.74	99.70	90	96	93

Table 4.4: Comparison of ML classifiers using RFE (Recursive Feature Elimination).

All the machine learning model are compared but the random forest is proven to be the best while using feature selection method. The confusion matrix is given in the figure 4.4

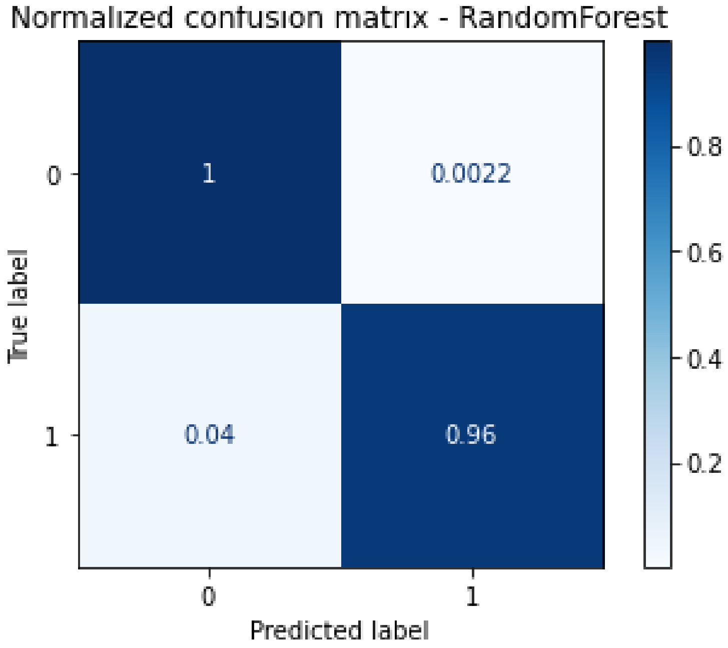


Figure 4.4: Confusion Matrix for Random Forest Using RFE (Recursive Feature Elimination).

Now Random forest is performing well in all types of fusion methods which were used in these experiments with comparison to other classifiers as well as feature selection using RFE so far. That's why implementation of the random forest for forward selection SFS feature selection is used as an experiment option which is also a wrapper based feature selection method. The parameters like k_features = 7 and k_features=11 with n_estimators = 100 , random_state=0,

Library mlextend is used in this method which has feature selection method sequential feature selector as SFS. 75% data is used for training in this process and 25% for test data. After training the model on random forest and applying SFS on that model were recorded following results in the table 4.5 .

Random Forest with Forward Selection — SFS()	7 features	11 features
Training Accuracy	99.70	99.69
Testing Accuracy	99.68	99.65
Precision	90	89
Recall	96	97
F1 Score	93	93

Table 4.5: Binary Classification Results for Forward Selection — SFS () using Random Forest.

The confusion matrix of the forward selection is shown in figure 4.5. Which shows the good F1 score.

run_randomForest(X_train_sfs, X_test_sfs, Y_train, Y_test)

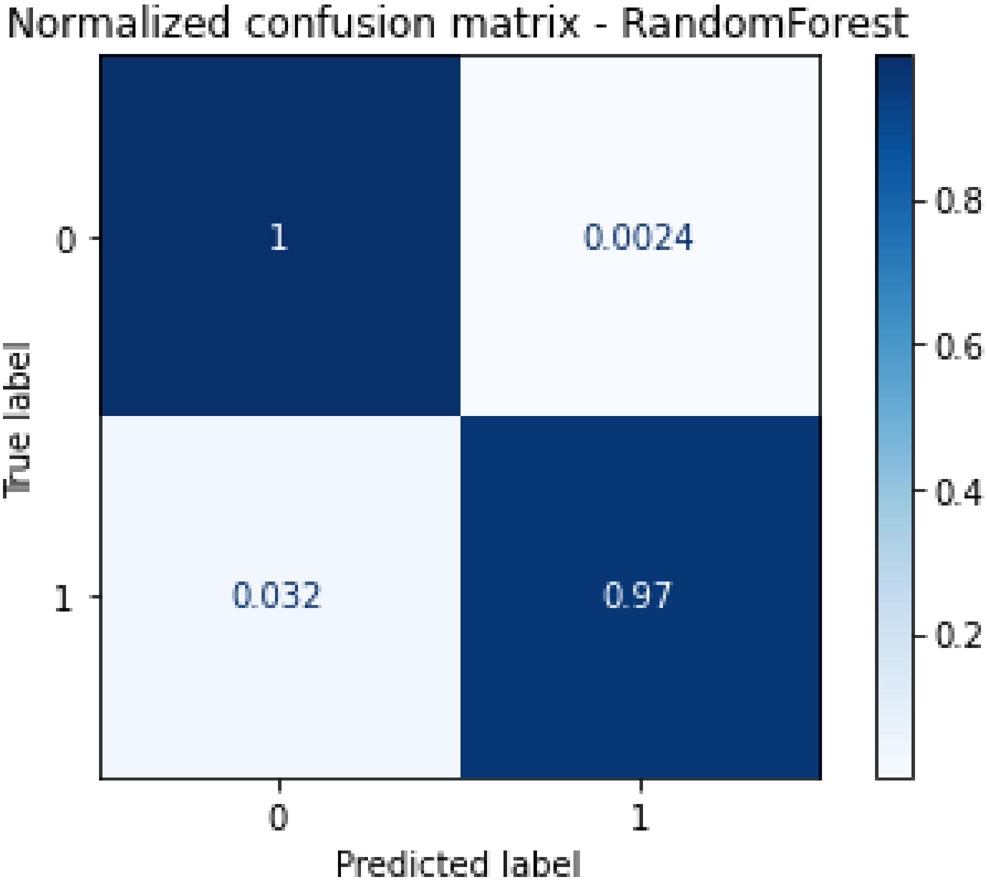


Figure 4.5: Confusion Matrix Using Random Forest with Forward Selection — SFS ().

The performance of the model using the five number of features is shown in the figure 4.6 below

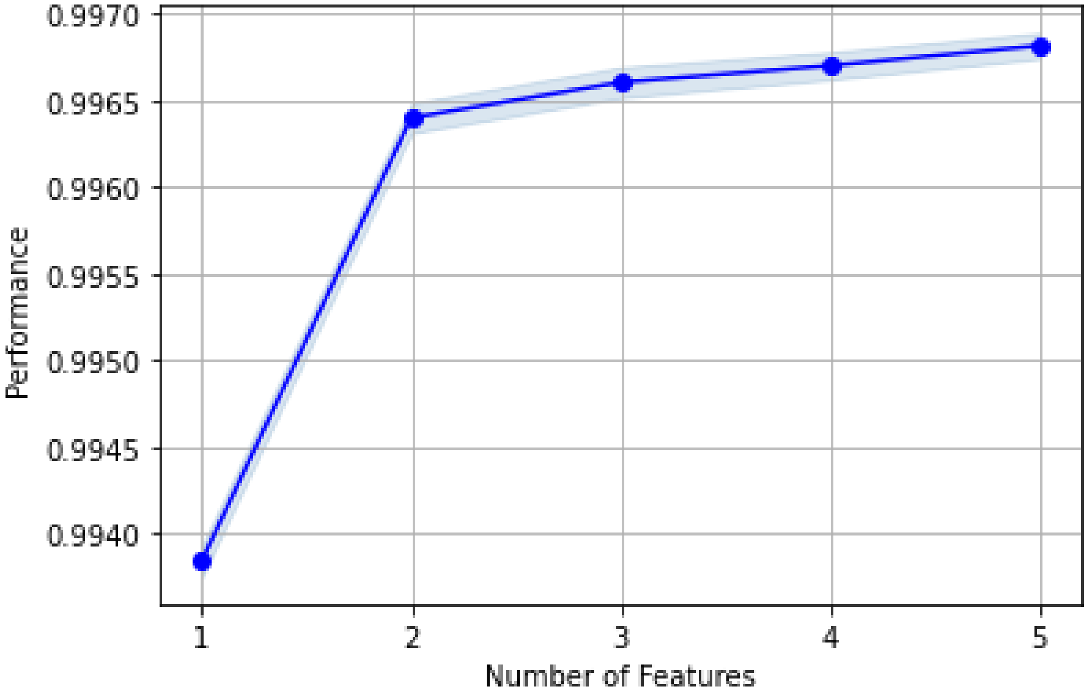


Figure 4.6: The performance of the model with feature selection FSF.

4.3.2 Using Multi-classification

Multi-classification includes 9 classes and 12 features; recall, precision and f1 score is on the basis of support (The total number of samples of the true response that lie in that class in which Analysis having support number 1850). Data is combined and cleaned first. After removing the null values in the data frame the features were selected from the data and combining the binary and the continuous values of the dataset. Also converting the string data used is converted to the integer type data. In next step the data is spliced into training and testing split using sklearn model selection to 25% test and 75% train using train, test split method. The target feature is now the Attack category which is encoded in the label using label encoder. Random forest, decision tree , Xgboost, SVM, and LR are applied one after the other to check the accuracy comparisons between all the ML classifiers. All the classes in terms of their support are shown using the bar chart in figure 4.7.

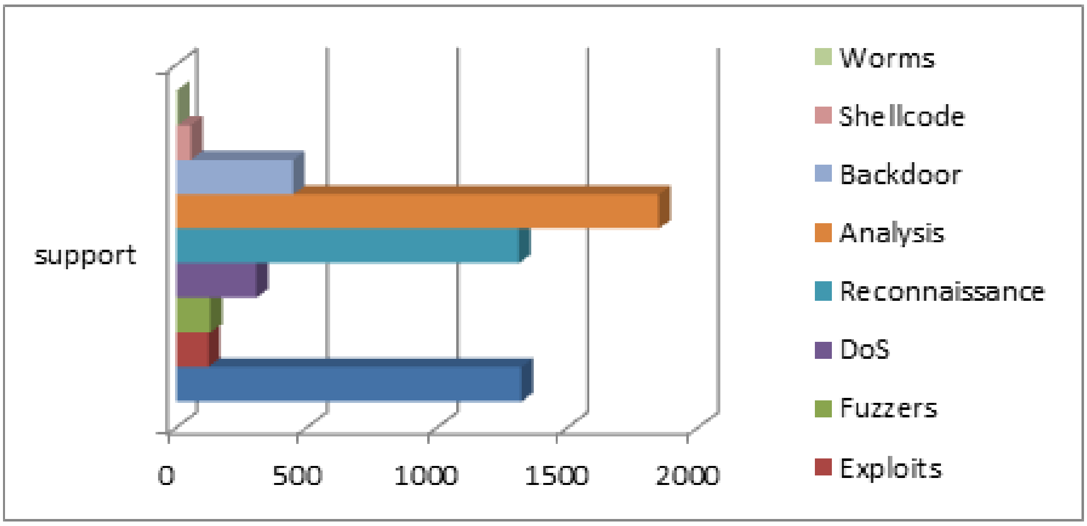


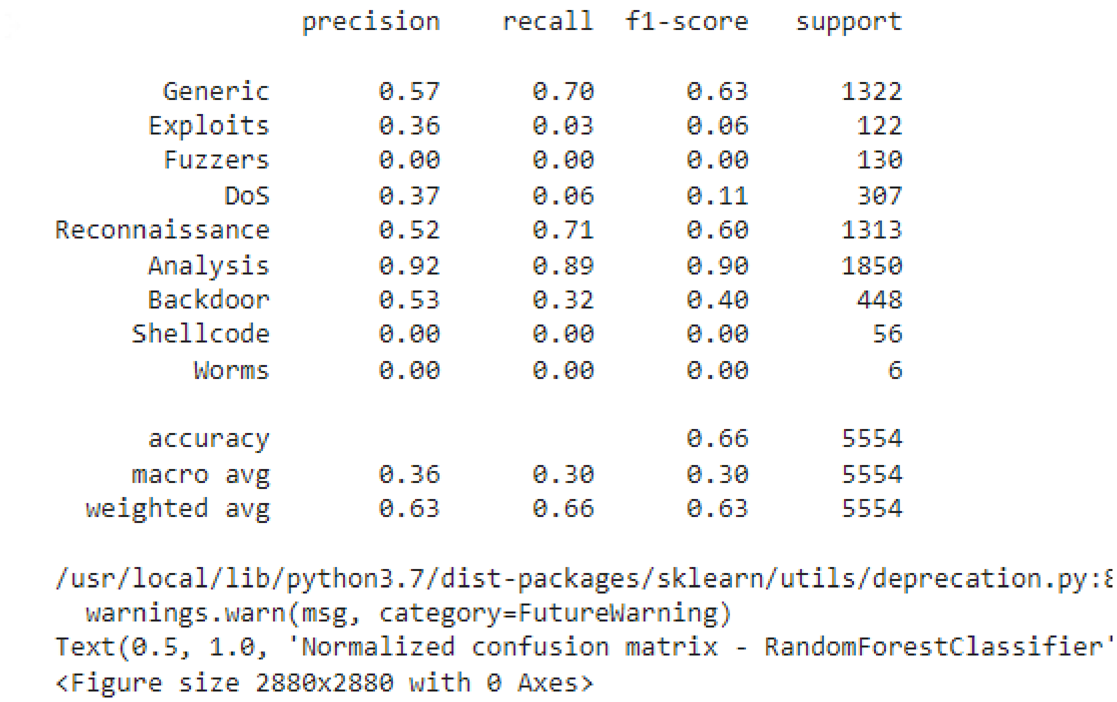
Figure 4.7: Distribution of classes in terms of their support.

After applying the experiments these results were recorded in the form of table for different algorithms for multi-classification. Results are shown in the table 4.6

ML Method	Training Accuracy	Testing Accuracy	Precision	Recall	F1 Score
DT	70.72	65.50	91	88	90
LR	56.22	54.48	58	83	68
XGboost	65.02	63.11	79	86	82
SVM	63.24	61.65	80	81	80
RF	70.72	65.86	92	89	90

Table 4.6: Multi-classification for Different Machine Learning Models.

Random forest resulted the accuracy of 70.72% for training and 65.86% for testing dataset which is best among all the other machine learning models. The confusion matrix of random forest for multi-classification is given the figure 4.8.



```
                  precision    recall  f1-score   support

       Generic       0.57      0.70      0.63      1322
      Exploits       0.36      0.03      0.06       122
       Fuzzers       0.00      0.00      0.00       130
           DoS       0.37      0.06      0.11       307
Reconnaissance       0.52      0.71      0.60      1313
      Analysis       0.92      0.89      0.90      1850
      Backdoor       0.53      0.32      0.40       448
     Shellcode       0.00      0.00      0.00        56
         Worms       0.00      0.00      0.00         6

      accuracy                           0.66      5554
     macro avg       0.36      0.30      0.30      5554
  weighted avg       0.63      0.66      0.63      5554

/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:8
  warnings.warn(msg, category=FutureWarning)
Text(0.5, 1.0, 'Normalized confusion matrix - RandomForestClassifier'
<Figure size 2880x2880 with 0 Axes>
```

Figure 4.8: Confusion Matrix for multi-classification using Random Forest.

4.3.3 Deliverables

Following are the deliverables of this thesis work.

- This work created two classification systems. First is the binary classification system with the feature elimination method called RFE recursive feature elimination with the accuracy of 99.70% with the Random forest as compared to the decision tree, logistic regression, xgboost and support vector machine.

- Second is the multi-class classification with RFE; comparison between DT, LR, SVM, XGBoost and RF. In which random forest stands out with the accuracy of 65.86% among all the classifiers.

- The complete framework with diagram and data set with addition to the algorithms for feature selection and normalization are also in that.

- For future work other algorithms are also applied on the data given.

4.4 Summary

In this section all the experiments with results are recorded in the form of tables and figures. First of all the data is retrieved using python coed in google colab. UNSW NB-15 data set has four files which has 49number of features in it. Hence the data is very sparse so the number of features is reduced to 12. After that the data is normalized all the null values are removed as well as the features which have different data types are converted into integer type of data from float. The feature on which data is trained is label. Then the features selection methods like recursive feature selection methods is used. Different machine learning classifiers like random forest, decision tree, xgboost, support vector machine and logistic regression is applied using feature selection. The results shows that the random forest classifier has the highest accuracy with the 99.74% as comparison to decision tree, support vector machine and other classifiers. The results are then recorded in the table. Also with that the sequential feature selections SFS is also applied which shows that the random forest out class the other classifiers is well. Using multi-classification also applied using different machine learning classifiers. Using multi-classification also proves that the random forest performs better in multi-classification.

Chapter 5

Discussion

5.1 Introduction

In this chapter we will discuss about the proposed methodology, the results that are achieved and the reliability and the validity of the results. the results which are achieved will be in discussion in the first section 5.1 and also the problems that are faced during the experimentation. Furthermore the evaluation of the results will be performed and their interpretation is well. In second section which is 5.2 we will discuss the proposed approaches and its benefits for choosing it and the flaws which comes out to check whether this method is right or wrong. Finally in third section 5.3 evaluation of the algorithms regarding its validation and data set features will be discussed with addition to the results that are achieved. Hence the following question are answered in this chapter during discussion.

- What conclusion can be pulled out from the methods we applied?
- Was the given task of IDS best suitable for the applied method?
- What are the gains and flaws in the given work?
- How much the data and its its results are reliable?

5.2 Results Interpretation

5.2.1 Binary Classification With Machine Learning Classifiers

The machine learning algorithms are decision tree, random forest , svm and xgboost which are implemented to compare the results. The best classifier which results the best accuracy is random forest with the accuracy of 99.70%. On the other hand decision tree has the accuracy of the 99.69%. Xgboost has the accuracy of 99.64% and logistic regression with accuracy of 98.93% and support vector machine with the accuracy of 99.65% while using the recursive feature elimination method. While we have the 49 total number of features in our data set. Hence the feature elimination method reduced the number of features to the 12 number of features. Feature 'Label' is the one on which we trained the features. The data is divided into two parts. One is training data other is testing data which are 75% and 25% respectively. While talking about the precision of the classifiers every classifier has different precision values as well as he recall and the accuracy score. One main issue is the data is very sparse so data is cleaned after the data is converted into the data frame in order to implement the algorithms. Our data label contains the two types which is attack or no-attack in our thesis. The following figure 5.1 shows the number of distribution of the attacks and non-attacks in the data set.

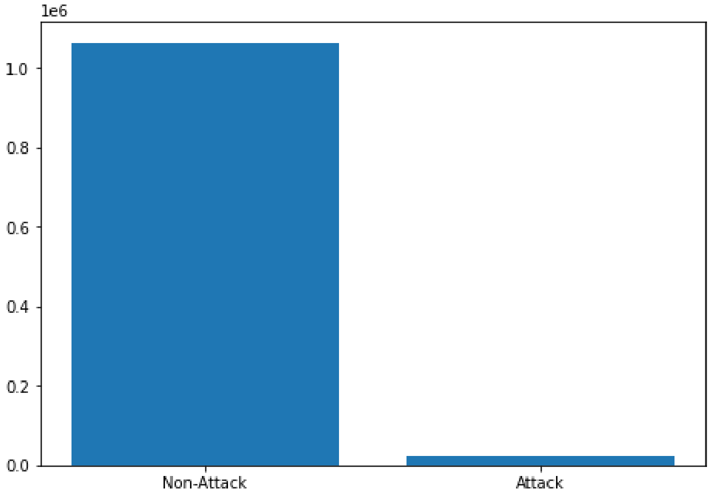


Figure 5.1: Distribution of the attacks and non-attacks in the data set.

Now we will take a look at the precision for the algorithms we have used. Random forest which we have used has the best precision which is 90%. On the other hand decision tree has also 90% precision value. SVM has relatively less precision value with 87% due to its complex nature for the given data set. Xgboost is more less in terms of its precision with the value of 86%. At the last the least precision value is for the logistic regression with 85% precision value.

Recall for the random forest is quite in terms of percentage with the 96%. But for decision trees classifier is reduces to one point by 95% which is also very good. But SVM has more recall value with 98%. XGboost has 99% and the logistic regression with the 58% according to the experimental results as described in the results section as well.

F1 score for the classifier random forest is 93% which is quite good. decision tree is close to the random forest in terms of f1 score which is also 93 %. SVM resulted the f1 score 92% which is close to the DT and RF. XGboost and logistic regression also has the 92% and 69% respectively.

5.2.2 Multi-class Classification With Machine Learning

Machine learning algorithms which we have implemented for the multi classification are same which we applied for the binary classification as discussed in the chapter 2 which are random forest, decision tree, support vector machine and the logistic regression and xgboost. In binary classification we have the two classes but in this case the there are nine classes in it which are described in previous section. The results suggest that the random forest classifiers gives the higher accuracy for the intrusion detection system with 65.86% which is most among all the other classifiers. On the other hand decision tree 65.50% which is also good accuracy and close to our recommended classifier. SVM gives us the accuracy of 61.58% and the xgboost 63.11% and the LR with the accuracy of 54.48% which least among all the classifiers.

Lets discuss the precision of the machine learning algorithms for the multi classification process. The precision of the random forest is 92% which is highest than other machine learning methods used in this process. Decision tree with the 91% is on the second spot in terms of the precision; logistic regression resulted in to 58% which is very low due to the regression which given multiple hyperplane; xgboost uses gradient boost algorithm so its precision is 79%, support vector machine gives the precision of 80%.

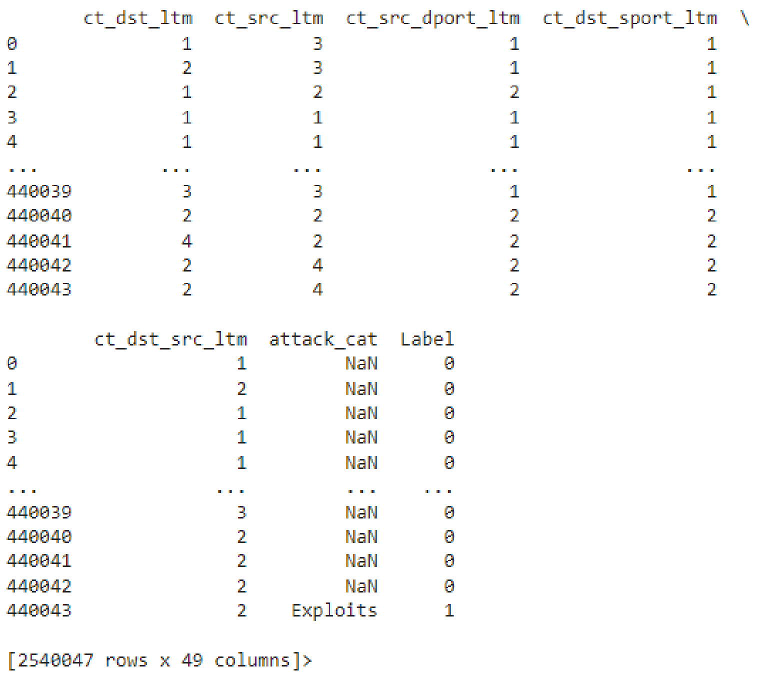
Recall of these algorithms is very important to discuss now we will look into that part is well. Recall for random forest is recorded by the algorithm is 89% which good; decision tree is very complex algorithms which has a lot of tress so this machine learning algorithms generates the recall 88%; xgboost gives the 86% recall value; logistic regression is also the supervised machine learning algorithms which can work for the binary classification is well so it gives the recall of 83%, support vector machine produces the recall value of 81%.

F1 score for the machine learning algorithms are recorded with RF 90% of score; DT has the score of 90%; SVM has the less than random forest ehich is 80% with xgboost has 82% and LR has 68% f1 score. It is clear from the results that this proposed framework has been remarkable in terms of its results produced. The results have been validated too. But this framework and the results have not been use and implemented by the professionals and the organizations.

5.3 Method Reflection

The proposed framework has been very suitable in order to find the results using the UNSW NB-15 data set. We have used different techniques to achieve our goals and remained focus on that. At the start we begin the process by extracting the data from the files. After that data was pre-processing. The quality of the method gives the good results by choosing the good features. Moreover the validity of the machine learning algorithms which are used for the training the model never remain the question. It worked perfectly for on the given data set. Total number of features which are presented in our dataset are 49 with over 2540047 number of data entries with it. The total number of features are given in the following figure 5.2.

we selected the features from this large data set to achieve our goal. Features selection methods does the great work to pick the best features from the whole data set which was reduced to the 12 features. Then the data set was normalized and the null values were removed. After that those features which had the different data types was converted into the same types to get the most accurate results. For data to be used for the creation f the models and application data is divided into two portion. training data is 75% and the test data is 25%. Then the different models are fitted like random forest; decision tree, logistic regression; xgboost and svm. All these machine learning models uses the

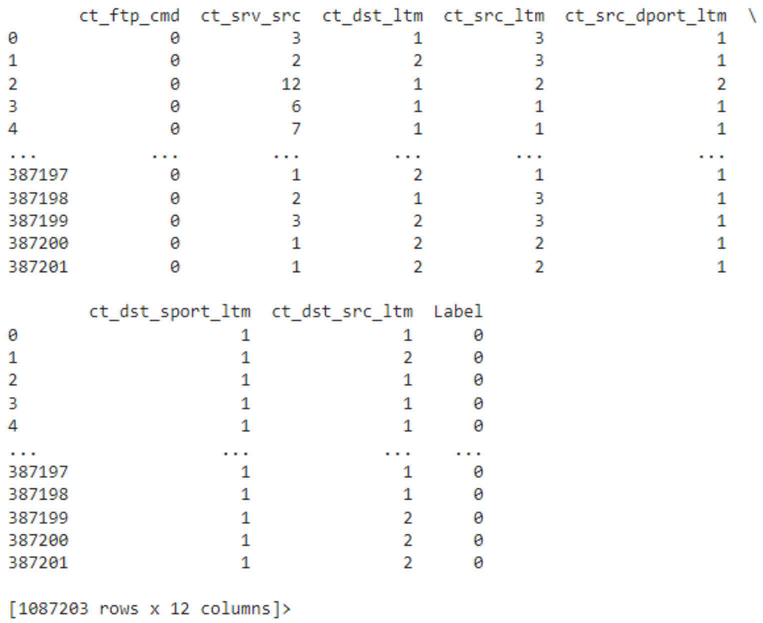


```
        ct_dst_ltm  ct_src_ltm  ct_src_dport_ltm  ct_dst_sport_ltm  \
0                1           3                 1                 1
1                2           3                 1                 1
2                1           2                 2                 1
3                1           1                 1                 1
4                1           1                 1                 1
...            ...         ...               ...               ...
440039           3           3                 1                 1
440040           2           2                 2                 2
440041           4           2                 2                 2
440042           2           4                 2                 2
440043           2           4                 2                 2

         ct_dst_src_ltm  attack_cat  Label
0                     1         NaN      0
1                     2         NaN      0
2                     1         NaN      0
3                     1         NaN      0
4                     1         NaN      0
...                 ...         ...    ...
440039                3         NaN      0
440040                2         NaN      0
440041                2         NaN      0
440042                2         NaN      0
440043                2    Exploits      1

[2540047 rows x 49 columns]>
```

Figure 5.2: Description of data in terms of columns and rows.

sklearn libraries and mlxtend in order to implement on them. In the following figure 5.3 describes the number of features that we select for the our proposed framework.

The correlation between the features is very important to find because the correlation gives the idea of relation between the features how close they are and which feature can be selected to achieve the more accuracy. The figure 5.4 shows the correlation between all the features of the data set.



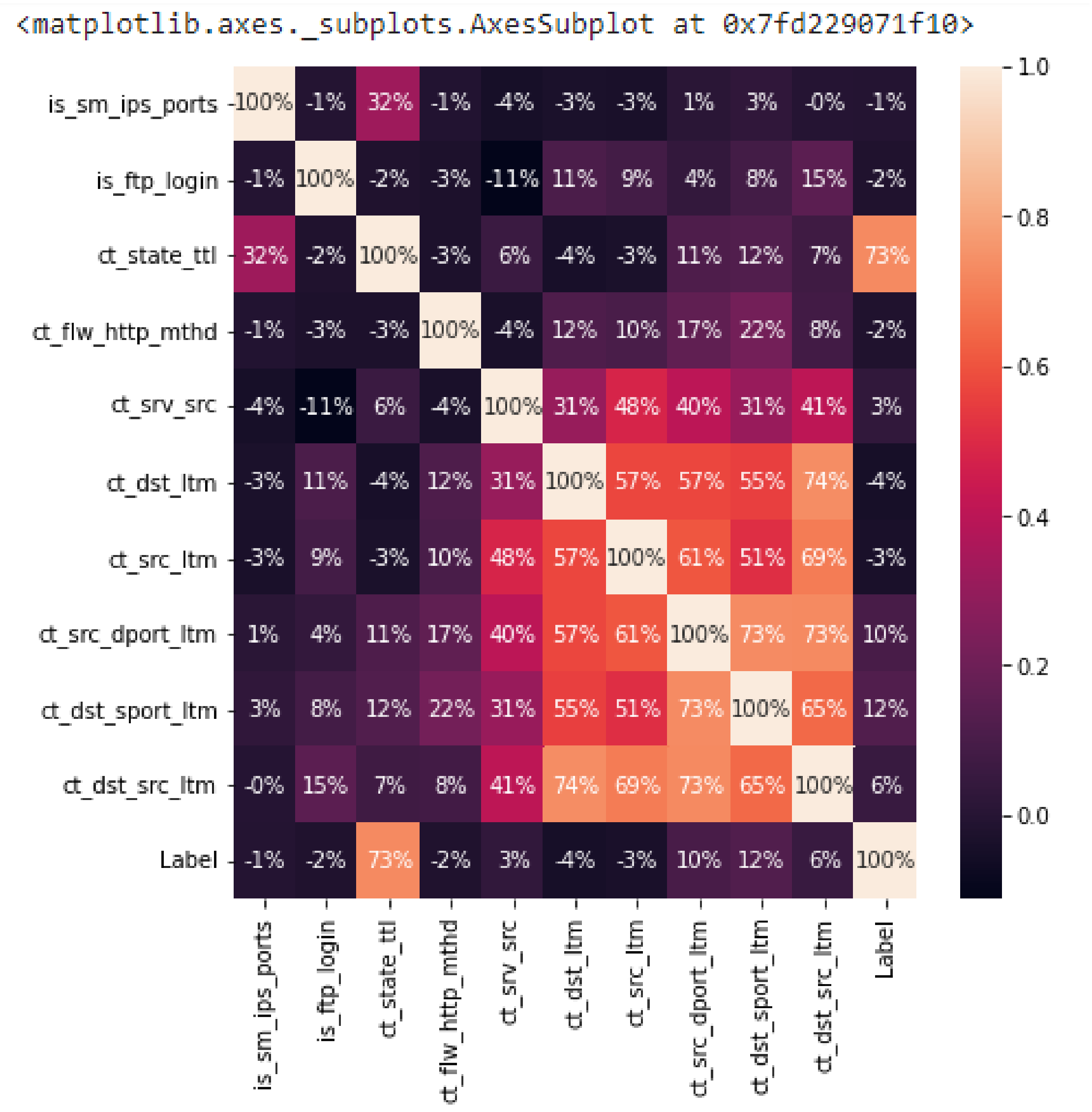
```
        ct_ftp_cmd  ct_srv_src  ct_dst_ltm  ct_src_ltm  ct_src_dport_ltm  \
0                0           3           1           3                 1
1                0           2           2           3                 1
2                0          12           1           2                 2
3                0           6           1           1                 1
4                0           7           1           1                 1
...            ...         ...         ...         ...               ...
387197           0           1           2           1                 1
387198           0           2           1           3                 1
387199           0           3           2           3                 1
387200           0           1           2           2                 1
387201           0           1           2           2                 1

         ct_dst_sport_ltm  ct_dst_src_ltm  Label
0                       1               1      0
1                       1               2      0
2                       1               1      0
3                       1               1      0
4                       1               1      0
...                   ...             ...    ...
387197                  1               1      0
387198                  1               1      0
387199                  1               2      0
387200                  1               2      0
387201                  1               2      0

[1087203 rows x 12 columns]>
```

Figure 5.3: Selected features for experimentation.

5.4 Reliability

The final section reflects the validation and the reliability of the results shown in section 4.2. The results that we have achieved gives the answer to our research question. Data set which is set as benchmark has not achieved so much accurate good results like the other researchers. The strategy has been proven valid for these classifiers because of the big and sparse nature of the data. By applying the general structure for the knowledge discovery first of all the data was pre processed and validated and the redundant data was removed effectively. In this work we have to mention the flaws which can be found. First of all the large nature to data set can never be easy to implement and achieve highly accurate results. Secondly the deep learning algorithms does not give that good results as we have achieved.

However this framework is validated through the simple validation technique which is adopted. On the other hand the professional and institution have not yet proved and validated on their end. In conclusion the research question related to the final results and proposed approach about the intrusion detection system is given.



```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd229071f10>
```

	is_sm_ips_ports	is_ftp_login	ct_state_ttl	ct_flw_http_mthd	ct_srv_src	ct_dst_ltm	ct_src_ltm	ct_src_dport_ltm	ct_dst_sport_ltm	ct_dst_src_ltm	Label
is_sm_ips_ports	100%	-1%	32%	-1%	-4%	-3%	-3%	1%	3%	-0%	-1%
is_ftp_login	-1%	100%	-2%	-3%	-11%	11%	9%	4%	8%	15%	-2%
ct_state_ttl	32%	-2%	100%	-3%	6%	-4%	-3%	11%	12%	7%	73%
ct_flw_http_mthd	-1%	-3%	-3%	100%	-4%	12%	10%	17%	22%	8%	-2%
ct_srv_src	-4%	-11%	6%	-4%	100%	31%	48%	40%	31%	41%	3%
ct_dst_ltm	-3%	11%	-4%	12%	31%	100%	57%	57%	55%	74%	-4%
ct_src_ltm	-3%	9%	-3%	10%	48%	57%	100%	61%	51%	69%	-3%
ct_src_dport_ltm	1%	4%	11%	17%	40%	57%	61%	100%	73%	73%	10%
ct_dst_sport_ltm	3%	8%	12%	22%	31%	55%	51%	73%	100%	65%	12%
ct_dst_src_ltm	-0%	15%	7%	8%	41%	74%	69%	73%	65%	100%	6%
Label	-1%	-2%	73%	-2%	3%	-4%	-3%	10%	12%	6%	100%

Figure 5.4: Correlation between the selected features.

5.5 Summary

In this section all the results are discussed we achieved in the results chapter. Binary classification of all the machine learning algorithms are discussed to answer the question in the problem statement. Moreover other techniques regarding to the experiments which are carried out related to our problem question are also briefed. Multi classification as well as the features elimination methods are described in their importance in large data set is stated.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Above discussion provides the comprehensive explanation for the intrusion detection in terms of its accuracy for detection. Also this study helps and enables any intrusion detection system to make quick response and real time decisions. In this research work, proposed hybrid system is using UNSW-NB15 dataset which has 49 features in it and the best 12 features from the 49 features were selected because all the features are not that important. Different machine learning classifiers used to select the best one among them. Like random forest RF, logistic regression LR, support vector machine SVM, xgboost classifier and decision trees DT. All these algorithms are supervised machine learning algorithms which has been discussed very comprehensively in above sections. Features selections methods like recursive feature elimination sequential feature selectors and forward selection and backward selection are the few wrapper based techniques which have been discussed and few are implemented as well. For that purpose all the code is implemented in Anaconda 3 and python 3. All the work is done in python using sklearn library. The proposed model using feature selection with random forest has resulted us the best results for binary classification and multi-class classification in IDS (intrusion detection system) with highest accuracy.

6.2 Further Research

The future in this area is very broad. These algorithms can be implemented for different datasets as well as more best and well known machine learning algorithms and deep learning methods. Also the assessment of the risk can also be evaluated with the less time consumption. Cyber security and intrusion detection system has vast range and dynamics for old researchers as well as new researchers. Moreover the need for such systems has a lot of importance for the organization. That's why the need for the new and accurate systems will always be there. New feature selection methods can play crucial rule for next few years. On the other hand sparse data is a big issue in terms of accuracy. Hence filtration of data and ensemble methods can be used as good combination for the accuracy of these advanced system as compared to the old protection system which are now have zero worth in front of these advanced protection systems.

6.3 Limitations

There are many advantages of having a very efficient intrusion detection system with high accuracy of detection. On the other hand it also has few limitations as well. There should be a very strong authentication system otherwise weak IDS system cannot prevent the malware from the attacker. Also some attacks are software specific if there is no proactive solutions for those soft wares it can easily be attacked by intruders so database signatures must be updated with the time. Many attacks are with different behaviors keep attacking the system but the real attacks have low percentage than the true attacks, that's why sometimes the real attacks are ignored by the systems.

Sometimes the data packets contains the false internet protocol addresses by hackers because these intrusion detection system work on the internet protocols to identify the threats.Noise in the dataset affects the performance of this system on the larger scale. The biggest worry is the encrypted packets which are very hard to identify as a threat. These packets can easily fool the intrusion detection systems until they are identified.

Bibliography

[1] Hebatallah Mostafa Anwer, Mohamed Farouk, and Ayman Abdel-Hamid. “A framework for efficient network anomaly intrusion detection with features selection”. In: *2018 9th International Conference on Information and Communication Systems (ICICS)*. IEEE. 2018, pp. 157–162.

[2] Mustapha Belouch, Salah El Hadaj, and Mohamed Idhammad. “Performance evaluation of intrusion detection based on machine learning using Apache Spark”. In: *Procedia Computer Science* 127 (2018), pp. 1–6.

[3] Ilyas Benmessahel, Kun Xie, and Mouna Chellal. “A new evolutionary neural networks based on intrusion detection systems using multiverse optimization”. In: *Applied Intelligence* 48.8 (2018), pp. 2315–2327.

[4] Priyanka Dahiya and Devesh Kumar Srivastava. “A comparative evolution of unsupervised techniques for effective network intrusion detection in hadoop”. In: *International Conference on Advances in Computing and Data Sciences*. Springer. 2018, pp. 279–287.

[5] Abhishek Divekar et al. “Benchmarking datasets for anomaly-based network intrusion detection: KDD CUP 99 alternatives”. In: *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*. IEEE. 2018, pp. 1–8.

[6] Sathish P Kumar and Arun Raaza. “Study and analysis of intrusion detection system using random forest and linear regression”. In: *Periodicals of Engineering and Natural Sciences (PEN)* 6.1 (2018), pp. 197–200.

[7] Sasanka Potluri, Shamim Ahmed, and Christian Diedrich. “Convolutional neural networks for multi-class intrusion detection system”. In: *International Conference on Mining Intelligence and Knowledge Exploration*. Springer. 2018, pp. 225–238.

[8] Mohammed F Suleiman and Biju Issac. "Performance comparison of intrusion detection machine learning classifiers on benchmark and new datasets". In: *2018 28th International Conference on Computer Theory and Applications (ICCTA)*. IEEE. 2018, pp. 19–23.

[9] Mohammed F. Suleiman and Biju Issac. "Performance Comparison of Intrusion Detection Machine Learning Classifiers on Benchmark and New Datasets". In: *2018 28th International Conference on Computer Theory and Applications (ICCTA)*. 2018, pp. 19–23. DOI: 10.1109/ICCTA45985.2018.9499140.

[10] Charles Wheelus, Elias Bou-Harb, and Xingquan Zhu. "Tackling class imbalance in cyber security datasets". In: *2018 IEEE International Conference on Information Reuse and Integration (IRI)*. IEEE. 2018, pp. 229–232.

[11] Wei Zong, Yang-Wai Chow, and Willy Susilo. "A two-stage classifier approach for network intrusion detection". In: *International Conference on Information Security Practice and Experience*. Springer. 2018, pp. 329–340.

[12] Marwan Ali Albahar. "Recurrent neural network model based on a new regularization technique for real-time intrusion detection in SDN environments". In: *Security and Communication Networks* 2019 (2019).

[13] Anurag Das, Samuel A Ajila, and Chung-Horng Lung. "A comprehensive analysis of accuracies of machine learning algorithms for network intrusion detection". In: *International Conference on Machine Learning for Networking*. Springer. 2019, pp. 40–57.

[14] Vibekananda Dutta et al. "Hybrid model for improving the classification effectiveness of network intrusion detection". In: *Computational Intelligence in Security for Information Systems Conference*. Springer. 2019, pp. 405–414.

[15] Hyeokmin Gwon et al. "Network intrusion detection based on LSTM and feature embedding". In: *arXiv preprint arXiv:1911.11552* (2019).

[16] Ying-Feng Hsu et al. "Toward an online network intrusion detection system based on ensemble learning". In: *2019 IEEE 12th international conference on cloud computing (CLOUD)*. IEEE. 2019, pp. 174–178.

[17] Anwar Husain et al. “Development of an efficient network intrusion detection model using extreme gradient boosting (xgboost) on the unsw-nb15 dataset”. In: *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*. IEEE. 2019, pp. 1–7.

[18] Dishan Jing and Hai-Bao Chen. “SVM based network intrusion detection for the UNSW-NB15 dataset”. In: *2019 IEEE 13th international conference on ASIC (ASICON)*. IEEE. 2019, pp. 1–4.

[19] V Kanimozhi and Prem Jacob. “UNSW-NB15 dataset feature selection and network intrusion detection using deep learning”. In: *Int. J. Recent Technol. Eng* 7 (2019), pp. 443–446.

[20] Farrukh Aslam Khan and Abdu Gumaei. “A comparative study of machine learning classifiers for network intrusion detection”. In: *International conference on artificial intelligence and security*. Springer. 2019, pp. 75–86.

[21] Farrukh Aslam Khan et al. “A novel two-stage deep learning model for efficient network intrusion detection”. In: *IEEE Access* 7 (2019), pp. 30373–30385.

[22] Yun Lin et al. “Time-Related Network Intrusion Detection Model: A Deep Learning Method”. In: *2019 IEEE Global Communications Conference (GLOBECOM)*. 2019, pp. 1–6. DOI: 10.1109/GLOBECOM38437.2019.9013302.

[23] Souhail Meftah, Tajjeeddine Rachidi, and Nasser Assem. “Network based intrusion detection using the UNSW-NB15 dataset”. In: *International Journal of Computing and Digital Systems* 8.5 (2019), pp. 478–487.

[24] Anjum Nazir and Rizwan Ahmed Khan. “Combinatorial optimization based feature selection method: A study on network intrusion detection”. In: *arXiv preprint arXiv:1906.04494* (2019).

[25] Bayu Adhi Tama, Marco Comuzzi, and Kyung-Hyune Rhee. “TSE-IDS: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system”. In: *IEEE access* 7 (2019), pp. 94497–94507.

[26] Liu Zhiqiang et al. “Modeling network intrusion detection system using feed-forward neural network using unsw-nb15 dataset”. In: *2019 IEEE 7th International Conference on Smart Energy Grid Engineering (SEGE)*. IEEE. 2019, pp. 299–303.

[27] Shamis N Abd, Mohammad Alsajri, and Hind Raad Ibraheem. “Rao-SVM machine learning algorithm for intrusion detection system”. In: *Iraqi Journal For Computer Science and Mathematics* 1.1 (2020), pp. 23–27.

[28] Benjamin Ampel et al. “Labeling hacker exploits for proactive cyber threat intelligence: a deep transfer learning approach”. In: *2020 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE. 2020, pp. 1–6.

[29] Anouar BACHAR, Noureddine EL MAKHFI, and Omar EL Bannay. “Towards a behavioral network intrusion detection system based on the SVM model”. In: *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*. 2020, pp. 1–7. doi: 10.1109/IRASET48871.2020.9092094.

[30] Zina Chkirbene et al. “Hybrid machine learning for network anomaly intrusion detection”. In: *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*. IEEE. 2020, pp. 163–170.

[31] Sarika Choudhary and Nishtha Kesswani. “Analysis of KDD-Cup’99, NSL-KDD and UNSW-NB15 datasets using deep learning in IoT”. In: *Procedia Computer Science* 167 (2020), pp. 1561–1573.

[32] Muataz Salam Al-Daweri et al. “An analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system”. In: *Symmetry* 12.10 (2020), p. 1666.

[33] Antoine Delplace, Sheryl Hermoso, and Kristofer Anandita. “Cyber Attack Detection thanks to Machine Learning Algorithms”. In: *arXiv preprint arXiv:2001.06309* (2020).

[34] Smirti Dwibedi, Medha Pujari, and Weiqing Sun. “A comparative study on contemporary intrusion detection datasets for machine learning research”. In: *2020 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE. 2020, pp. 1–6.

[35] Ramy Elhefnawy, Hassan Abounaser, and Amr Badr. “A Hybrid Nested Genetic-Fuzzy Algorithm Framework for Intrusion Detection and Attacks”. In: *IEEE Access* 8 (2020), pp. 98218–98233. doi: 10.1109/ACCESS.2020.2996226.

[36] Mohamed Hammad, Wael El-medany, and Yasser Ismail. “Intrusion detection system using feature selection with clustering and classification machine learning algorithms on the unsw-nb15 dataset”. In: *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT).* IEEE. 2020, pp. 1–6.

[37] Sydney M Kasongo and Yanxia Sun. “Performance analysis of intrusion detection systems using a feature selection method on the UNSW-NB15 dataset”. In: *Journal of Big Data* 7.1 (2020), pp. 1–20.

[38] Sydney Mambwe Kasongo and Yanxia Sun. “A deep learning method with wrapper based feature extraction for wireless intrusion detection system”. In: *Computers & Security* 92 (2020), p. 101752.

[39] Shahid Latif et al. “DRaNN: A deep random neural network model for intrusion detection in industrial IoT”. In: *2020 International Conference on UK-China Emerging Technologies (UCET).* IEEE. 2020, pp. 1–4.

[40] J Olamantanmi Mebawondu et al. “Network intrusion detection system using supervised learning paradigm”. In: *Scientific African* 9 (2020), e00497.

[41] Smitha Rajagopal, Katiganere Siddaramappa Hareesha, and Poornima Panduranga Kundapur. “Feature relevance analysis and feature reduction of UNSW NB-15 using neural networks on MAMLS”. In: *Advanced Computing and Intelligent Engineering.* Springer, 2020, pp. 321–332.

[42] Smitha Rajagopal, Poornima Panduranga Kundapur, and Katiganere Siddaramappa Hareesha. “A stacking ensemble for network intrusion detection using heterogeneous datasets”. In: *Security and Communication Networks* 2020 (2020).

[43] Kamalakanta Sethi et al. “Deep reinforcement learning based intrusion detection system for cloud infrastructure”. In: *2020 International Conference on COMmunication Systems & NETworkS (COMSNETS).* IEEE. 2020, pp. 1–6.

[44] Xiujin Shi, Yifan Cai, and Yang Yang. “Extreme trees network intrusion detection framework based on ensemble learning”. In: *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA).* IEEE. 2020, pp. 91–95.

[45] Hoang Ngoc Thanh and Tran Van Lang. “Evaluating Effectiveness of Ensemble Classifiers When Detecting Fuzzers Attacks on the Unsw-Nb15 Dataset”. In: *Journal of Computer Science and Cybernetics* 36.2 (2020), pp. 173–185.

[46] Mubarak Albarka Umar, Chen Zhanfang, and Yan Liu. “Network intrusion detection using wrapper-based decision tree for feature selection”. In: *Proceedings of the 2020 International Conference on Internet Computing for Science and Engineering*. 2020, pp. 5–13.

[47] Hongpo Zhang et al. “An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset”. In: *Computer Networks* 177 (2020), p. 107315.

[48] Jianwu Zhang et al. “Model of the intrusion detection system based on the integration of spatial-temporal features”. In: *Computers & Security* 89 (2020), p. 101681.

[49] Jielun Zhang, Fuhao Li, and Feng Ye. “An Ensemble-based Network Intrusion Detection Scheme with Bayesian Deep Learning”. In: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. 2020, pp. 1–6. doi: 10.1109/ICC40277.2020.9149402.

[50] Muhammad Ahmad et al. “Intrusion detection in internet of things using supervised machine learning based on application and transport layer features using UNSW-NB15 data-set”. In: *EURASIP Journal on Wireless Communications and Networking* 2021.1 (2021), pp. 1–23.

[51] AM Aleesa et al. “Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques”. In: *Journal of Engineering Science and Technology* 16.1 (2021), pp. 711–727.

[52] Sikha Bagui et al. “Classifying UNSW-NB15 Network Traffic in the Big Data Framework using Random Forest in Spark”. In: *International Journal of Big Data Intelligence and Applications (IJBDIA)* 2.1 (2021), pp. 1–23.

[53] Raisa Abedin Disha and Sajjad Waheed. “A Comparative study of machine learning models for Network Intrusion Detection System using UNSW-NB 15 dataset”. In: *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*. IEEE. 2021, pp. 1–5.

[54] Raisa Abedin Disha and Sajjad Waheed. “A Comparative study of machine learning models for Network Intrusion Detection System using UNSW-NB 15 dataset”. In: *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*. 2021, pp. 1–5. DOI: 10.1109/ICECIT54077.2021.9641471.

[55] Mossa Ghurab et al. “A detailed analysis of benchmark datasets for network intrusion detection system”. In: *Asian Journal of Research in Computer Science* 7.4 (2021), pp. 14–33.

[56] Imran Edzereiq Kamarudin et al. “Performance Analysis on Denial of Service attack using UNSW-NB15 Dataset”. In: *2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM)*. IEEE. 2021, pp. 423–426.

[57] Ilhan Firat Kilincer, Fatih Ertam, and Abdulkadir Sengur. “Machine learning methods for cyber security intrusion detection: Datasets and comparative study”. In: *Computer Networks* 188 (2021), p. 107840.

[58] Geeta Kocher and Gulshan Kumar. “Analysis of machine learning algorithms with feature selection for intrusion detection using UNSW-NB15 dataset”. In: *Available at SSRN 3784406* (2021).

[59] Xavier Larriva-Novo et al. “An IoT-focused intrusion detection system approach based on preprocessing characterization for cybersecurity datasets”. In: *Sensors* 21.2 (2021), p. 656.

[60] Jingyu Liu et al. “Research on intrusion detection based on particle swarm optimization in IoT”. In: *IEEE Access* 9 (2021), pp. 38254–38268.

[61] Achmad Akbar Megantara and Tohari Ahmad. “A hybrid machine learning method for increasing the performance of network intrusion detection systems”. In: *Journal of Big Data* 8.1 (2021), pp. 1–19.

[62] TS Pooja and Purohit Shrinivasacharya. “Evaluating neural networks using Bi-Directional LSTM for network IDS (intrusion detection systems) in cyber security”. In: *Global Transitions Proceedings* 2.2 (2021), pp. 448–454.

[63] K Narayana Rao, K Venkata Rao, and Prasad Reddy PVGD. “A hybrid intrusion detection system based on sparse autoencoder and deep neural network”. In: *Computer Communications* 180 (2021), pp. 77–88.

[64] Nishit A Rathod et al. “Model Comparison and Multiclass Implementation Analysis on the UNSW NB15 Dataset”. In: *2021 International Conference on Computational Performance Evaluation (ComPE)*. IEEE. 2021, pp. 549–555.

[65] Aditi Roy and Khundrakpam Johnson Singh. “Multi-classification of UNSW-NB15 dataset for network anomaly detection system”. In: *Proceedings of International Conference on Communication and Computational Technologies*. Springer. 2021, pp. 429–451.

[66] Nausheen Sahar, Ratnesh Mishra, and Sidra Kalam. “Deep learning approach-based network intrusion detection system for fog-assisted IoT”. In: *Proceedings of international conference on big data, machine learning and their applications*. Springer. 2021, pp. 39–50.

[67] Neha Sharma and Narendra Singh Yadav. “Ensemble Learning based Classification of UNSW-NB15 dataset using Exploratory Data Analysis”. In: *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*. IEEE. 2021, pp. 1–7.

[68] Neha Sharma, Narendra Singh Yadav, and Saurabh Sharma. “Classification of UNSW-NB15 dataset using Exploratory Data Analysis using Ensemble Learning”. In: *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems* 8.29 (2021), e4–e4.

[69] Gautam Srivastava et al. “An ensemble model for intrusion detection in the internet of softwarized things”. In: *Adjunct proceedings of the 2021 international conference on distributed computing and networking*. 2021, pp. 25–30.

[70] PGV Suresh Kumar and Shaheda Akthar. “Building an efficient feature selection for intrusion detection system on UNSW-NB15”. In: *Proceedings of the 2nd International Conference on Computational and Bio Engineering*. Springer. 2021, pp. 641–649.

[71] Muhammad Zeeshan et al. “Protocol-Based Deep Intrusion Detection for DoS and DDoS Attacks Using UNSW-NB15 and Bot-IoT Data-Sets”. In: *IEEE Access* 10 (2021), pp. 2269–2283.

[72] Zeinab Zoghi and Gursel Serpen. “Unsw-nb15 computer security dataset: Analysis through visualization”. In: *arXiv preprint arXiv:2101.05067* (2021).

[73] Iftikhar Ahmad et al. “An Efficient Network Intrusion Detection and Classification System”. In: *Mathematics* 10.3 (2022), p. 530.

[74] Mohammed M Alani. “Implementation-Oriented Feature Selection in UNSW-NB15 Intrusion Detection Dataset”. In: *International Conference on Intelligent Systems Design and Applications*. Springer. 2022, pp. 548–558.

[75] Hakim Azeroual, Imane Daha Belghiti, and Naoual Berbiche. “Analysis of UNSW-NB15 Datasets Using Machine Learning Algorithms”. In: *International Conference on Digital Technologies and Applications*. Springer. 2022, pp. 199–209.

[76] Yee Jian Chew et al. “Benchmarking full version of GureKDDCup, UNSW-NB15, and CIDDS-001 NIDS datasets using rolling-origin resampling”. In: *Information Security Journal: A Global Perspective* 31.5 (2022), pp. 544–565.

[77] I Fosić, D Žagar, and K Grgić. “Network traffic verification based on a public dataset for IDS systems and machine learning classification algorithms”. In: *2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO)*. IEEE. 2022, pp. 1037–1041.

[78] Mohammad Humayun Kabir et al. “Network Intrusion Detection Using UNSW-NB15 Dataset: Stacking Machine Learning Based Approach”. In: *2022 International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE)*. IEEE. 2022, pp. 1–6.

[79] Ilhan Firat Kilincer, Fatih Ertam, and Abdulkadir Sengur. “A comprehensive intrusion detection framework using boosting algorithms”. In: *Computers and Electrical Engineering* 100 (2022), p. 107869.

[80] Mohanad Sarhan, Siamak Layeghy, and Marius Portmann. “Towards a standard feature set for network intrusion detection system datasets”. In: *Mobile Networks and Applications* 27.1 (2022), pp. 357–370.

[81] PGV Suresh Kumar and Shaheda Akthar. “Execution improvement of intrusion detection system through dimensionality reduction for UNSW-NB15 information”. In: *Mobile Computing and Sustainable Informatics*. Springer, 2022, pp. 385–396.

[82] Rachid Tahri et al. “A comparative study of Machine learning Algorithms on the UNSW-NB 15 Dataset”. In: *ITM Web of Conferences*. Vol. 48. EDP Sciences. 2022, p. 03002.

Appendix A

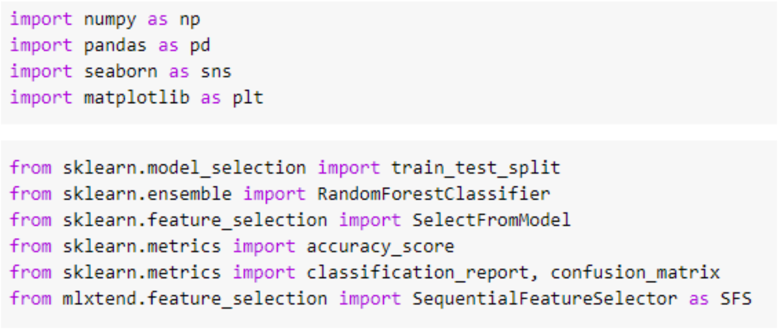
Feature Selection Discussion

A.1 Introduction

In this section we will discuss the other feature selection techniques which we have applied in this thesis. Recursive feature selection that we have used proved to be the good selection of feature selector but sequential features selector also applied on the same data set. Following are that gives the feature selections that we applied using python and gives the result.

A.2 Python Code Using FSRF(Feature Selection with Random Forest)

First of all the libraries which are used for the coding are shown in the figure A.1. The libraries numpy , pandas, seaborn and matplotlib for the data visualization is used. Then from sklearn train test split is imported for data division. Then random forest classifier is selected. Accuracy score is imported from sklearn metrics to get the accuracy of training data and the testing data. Also to get the confusion matrix the classification report and confusion matrics is imported from the sklearn.

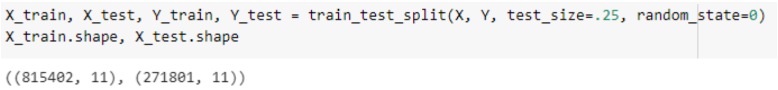


```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib as plt
```

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_selection import SelectFromModel
from sklearn.metrics import accuracy_score
from sklearn.metrics import classification_report, confusion_matrix
from mlxtend.feature_selection import SequentialFeatureSelector as SFS
```

Figure A.1: Machine learning Libraries for RFE and applying random forest classifier.

The data is divided into training and testing sub sets which is given in the figure A.2. Testing data is .25 which means from the data selected data 25% is testing data while other 75% is training data. If we look at the shape of the data which is given in the second line the features we selected.

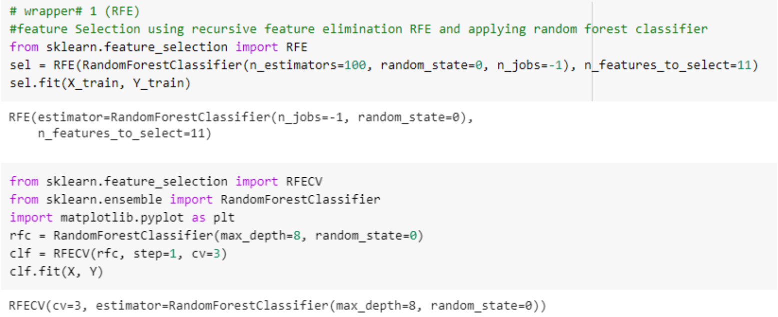


```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=.25, random_state=0)
X_train.shape, X_test.shape
```

```
((815402, 11), (271801, 11))
```

Figure A.2: Train and test split for the features.

The code for the Recursive feature selection with Random forest is given in the figure A.3. From sklearn the features selection RFE is imported. Then the features numbers which are selected is given. Note 12 feature is the [’label’] on which we are training.



```
# wrapper# 1 (RFE)
#feature Selection using recursive feature elimination RFE and applying random forest classifier
from sklearn.feature_selection import RFE
sel = RFE(RandomForestClassifier(n_estimators=100, random_state=0, n_jobs=-1), n_features_to_select=11)
sel.fit(X_train, Y_train)
```

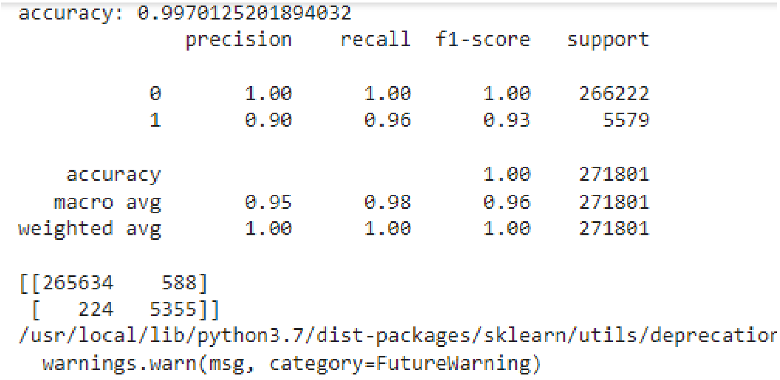
```
RFE(estimator=RandomForestClassifier(n_jobs=-1, random_state=0),
    n_features_to_select=11)
```

```
from sklearn.feature_selection import RFECV
from sklearn.ensemble import RandomForestClassifier
import matplotlib.pyplot as plt
rfc = RandomForestClassifier(max_depth=8, random_state=0)
clf = RFECV(rfc, step=1, cv=3)
clf.fit(X, Y)
```

```
RFECV(cv=3, estimator=RandomForestClassifier(max_depth=8, random_state=0))
```

Figure A.3: feature Selection using recursive feature elimination RFE and applying random forest classifier.

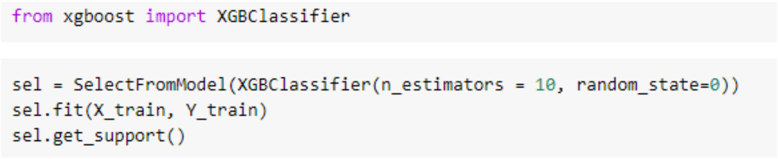
Accuracy of the Random forest classifier after applying the algorithms is shown in the figure A.4. Accuracy of the random forest is given in the following figure as well as the classification report which shows the precision, recall and the support.



```
accuracy: 0.9970125201894032
              precision    recall  f1-score   support

           0       1.00      1.00      1.00    266222
           1       0.90      0.96      0.93      5579

    accuracy                           1.00    271801
   macro avg       0.95      0.98      0.96    271801
weighted avg       1.00      1.00      1.00    271801

[[265634    588]
 [   224   5355]]
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecatior
  warnings.warn(msg, category=FutureWarning)
```

Figure A.4: Accuracy of random forest classifier with RFE.

XGboost classifier is imported from the xgboost. Then select from model is used. After that model is fitted using x_train and y_train. The code is given in figure A.5.

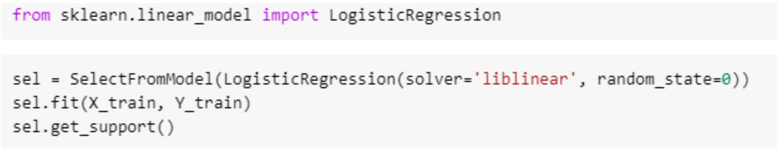


```
from xgboost import XGBClassifier
```

```
sel = SelectFromModel(XGBClassifier(n_estimators = 10, random_state=0))
sel.fit(X_train, Y_train)
sel.get_support()
```

Figure A.5: RFE with XGboost Code.

Logistic Regression classifier is imported using the library sklearn from linear model. Then we used select from model to initialize it. After the initialization model is fitted using x_train and y_train. Get method is used to check the model training. In figure A.6 the code of logistic regression is given.

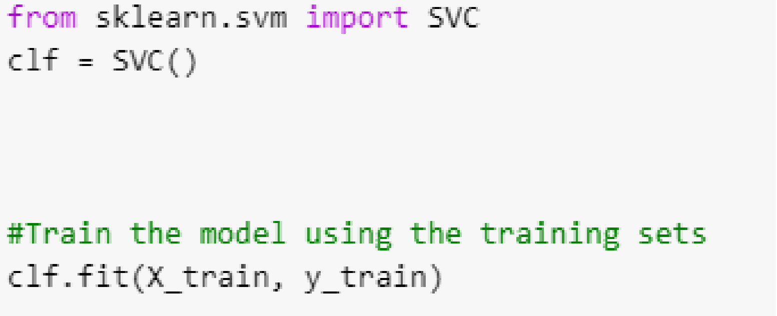


```
from sklearn.linear_model import LogisticRegression
```

```
sel = SelectFromModel(LogisticRegression(solver='liblinear', random_state=0))
sel.fit(X_train, Y_train)
sel.get_support()
```

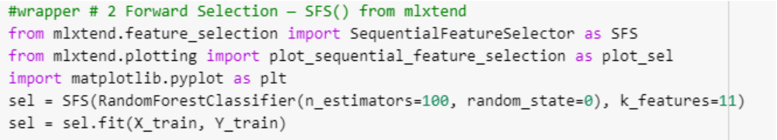
Figure A.6: RFE with Logistic Regression Code.

Support Vector Machine is a very good machine learning classifier. SVM is imported from sklearn as SVC. Then the model is trained using the x_train and y_train. The code for support vector machine is shown in figure A.7. Sequential feature selection is an other wrapper based feature selector. This feature selector is imported from the mlxtend library with feature selector as SFS(). For visualization of results from this model mlxtend plotting is imported from plot sequential feature selector as plot_sel. Then from selecting the random forest model is initialized. After that model is fitted to x_train and y_train for training. The python code for the sequential features selection is given in the figure A.8.



```
from sklearn.svm import SVC
clf = SVC()



#Train the model using the training sets
clf.fit(X_train, y_train)
```

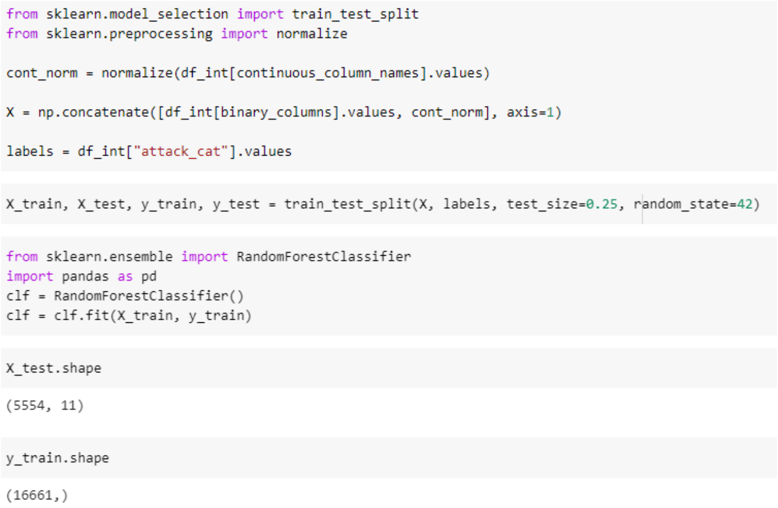
Figure A.7: RFE with Support Vector Machine Code.



```
#wrapper # 2 Forward Selection – SFS() from mlxtend
from mlxtend.feature_selection import SequentialFeatureSelector as SFS
from mlxtend.plotting import plot_sequential_feature_selection as plot_sel
import matplotlib.pyplot as plt
sel = SFS(RandomForestClassifier(n_estimators=100, random_state=0), k_features=11)
sel = sel.fit(X_train, Y_train)
```

Figure A.8: SFS feature selection .

For multi classification the classifier used is random forest. Now the number of classes are nine. Binary classifiers can also be used for this. To normalize the data normalize is imported from the sklearn pre-processing. All the binary and continuous columns are normalized first. Data is divided into train test split after.Random forest classifier is imported from the sklearn.ensemble as well as the pandas from pd. The model is fitted to after the initialization. Shape of test data is get using x_test.shape and y_train. shape for the y training data. Python code for the multi classifiers is shown in the figure A.9.



```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import normalize

cont_norm = normalize(df_int[continuous_column_names].values)

X = np.concatenate([df_int[binary_columns].values, cont_norm], axis=1)

labels = df_int["attack_cat"].values
```

```
X_train, X_test, y_train, y_test = train_test_split(X, labels, test_size=0.25, random_state=42)
```

```
from sklearn.ensemble import RandomForestClassifier
import pandas as pd
clf = RandomForestClassifier()
clf = clf.fit(X_train, y_train)
```

```
X_test.shape
```

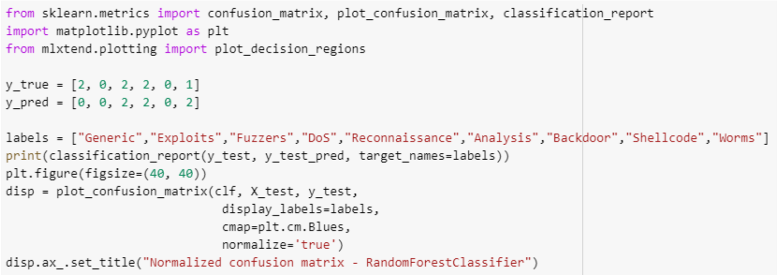
```
(5554, 11)
```

```
y_train.shape
```

```
(16661,)
```

Figure A.9: Multi Classification sung Random forest classifier .

To code the multi classification confusion matrix and classification report; confusion matrix and classification is imported from sklearn.metrics. For data visualization matplotlib is imported as plt. In label all the nine classes are used in order to get the confusion matrix for multi class classification. Multi classification confusion matrix code is given in figure A.10.



```
from sklearn.metrics import confusion_matrix, plot_confusion_matrix, classification_report
import matplotlib.pyplot as plt
from mlxtend.plotting import plot_decision_regions

y_true = [2, 0, 2, 2, 0, 1]
y_pred = [0, 0, 2, 2, 0, 2]

labels = ["Generic","Exploits","Fuzzers","DoS","Reconnaissance","Analysis","Backdoor","Shellcode","Worms"]
print(classification_report(y_test, y_test_pred, target_names=labels))
plt.figure(figsize=(40, 40))
disp = plot_confusion_matrix(clf, X_test, y_test,
                              display_labels=labels,
                              cmap=plt.cm.Blues,
                              normalize='true')
disp.ax_.set_title("Normalized confusion matrix - RandomForestClassifier")
```

Figure A.10: Multi Classification Confusion Matrix.

Appendix B

Artificial Neural Network

B.1 Introduction

Artificial neural networks consists of many nodes or units which are in form of layers. There are multiple number of layers in the artificial neural network. These layers are input layer, hidden layer and the output layer. These layers contains multiple nodes in each layer. Every layer has in the hidden layer has some weight from the input layer that gives the results fro the output layer. The input layer gets the relevant that from the source and analyzes the data. Then this data is given to the hidden layer in order to transform that data into the output layer. At the end the output layer gives the output of the artificial neural network for the data which is provided in the input layer. These layers in the neural network are connected to each other. So the data goes from one layer to the other layer so that the neural network learn the data more effectively and gives the output. The whole layout is given in the figure B.1

B.2 Python Code Using Artificial Neural Network

Following code in the figure B.2 is for the normalization. For normalization of the data train test split is imported from the sklearn.model selection. normalize is imported from the sklearn.preprocessing. After that all our selected feature columns are normalized as x. The values of label column are store in labels. For artificial neural network tensorflow library is imported as tf. Sequential is imported from the keras.models which a high level of API used in tensorflow. Pandas are imported as pg too. Then keras.layers dense

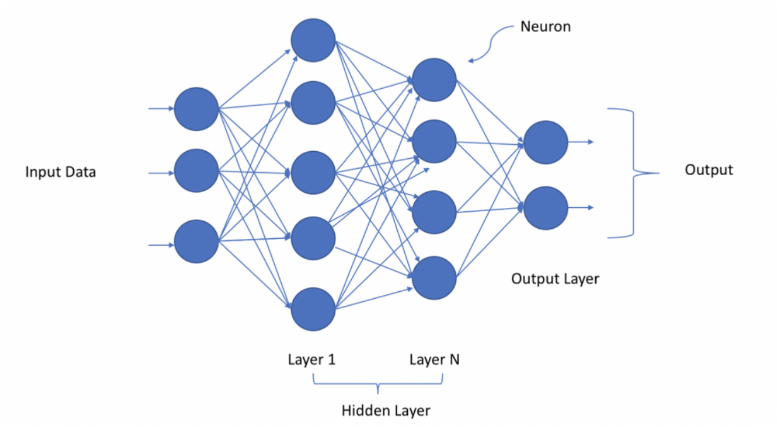
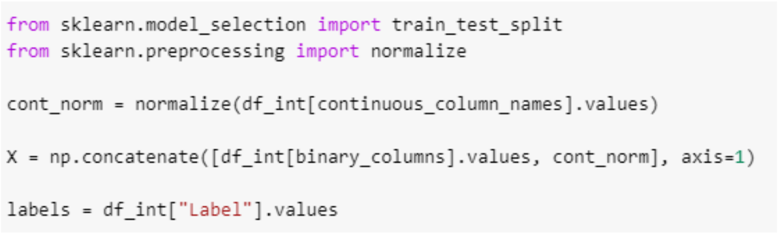


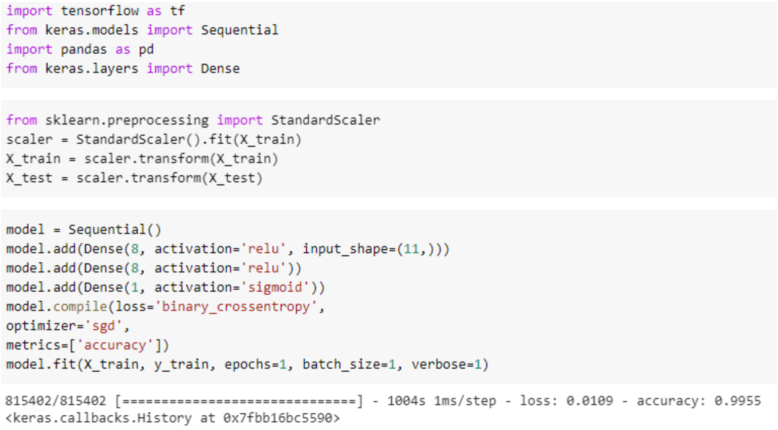
Figure B.1: Artificial Neural Network Diagram.



```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import normalize

cont_norm = normalize(df_int[continuous_column_names].values)

X = np.concatenate([df_int[binary_columns].values, cont_norm], axis=1)

labels = df_int["Label"].values
```

Figure B.2: Normalization in ANN.

is imported dense layers which refers to all the dense layers that has the input from the other layers. dense layers contains the neurons which are highly inter connected with each other. Python code for artificial neural network is in the figure B.3



```
import tensorflow as tf
from keras.models import Sequential
import pandas as pd
from keras.layers import Dense
```

```
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler().fit(X_train)
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
```

```
model = Sequential()
model.add(Dense(8, activation='relu', input_shape=(11,)))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='binary_crossentropy',
optimizer='sgd',
metrics=['accuracy'])
model.fit(X_train, y_train, epochs=1, batch_size=1, verbose=1)
```

```
815402/815402 [==============================] - 1004s 1ms/step - loss: 0.0109 - accuracy: 0.9955
<keras.callbacks.History at 0x7fbb16bc5590>
```

Figure B.3: Code for ANN.

[9] [54] [50] [32] [74] [51] [28] [75] [52] [2] [76] [31] [4] [33] [53] [5] [14] [77] [36] [17] [18] [78] [56] [19] [37] [57] [58] [59] [23] [24] [64] [65] [80] [68] [67] [8] [70] [81] [45] [10] [71] [26] [72] [21] [69] [45] [27] [41] [30] [6] [42] [79] [47] [34] [13] [7] [60] [55] [11] [25] [1] [20] [38] [48] [82] [73] [63] [43] [12] [3] [24] [40] [39] [46] [66] [62] [44] [16] [15] [61] [29] [22] [35] [49]