DDoS Attacks Detection using Machine Learning

by

Mohammed Younus Sabir

Bachelor of Technology, Electrical and Electronics Engineering,
Gandhi Institute of Technology and Management, Visakhapatnam, India, 2017

A Report Submitted in Partial Fulfillment of the Requirements for the Degree of
MASTER OF ENGINEERING
in the Department of Electrical and Computer Engineering

University of Victoria

DDoS Attacks Detection using Machine Learning

by

Mohammed Younus Sabir
Bachelor of Technology, Electrical and Electronics Engineering,
Gandhi Institute of Technology and Management, Visakhapatnam, India, 2017

Supervisory Committee

Dr. Fayez Gebali, Supervisor

(Department of Electrical and Computer Engineering)

Dr. M. Watheq El-Kharashi, Supervisor

(Department of Electrical and Computer Engineering)

ABSTRACT

The advancement in information technology has created a new era named as Internet of Things (IoT). This new technology has allowed things to be connected to the Internet, for example smart TVs, printers, cameras, smartphones, smartwatches, etc. This trend has enhanced the lifestyle of the users of these devices, and it provides new services and applications to them. The fast growth of IOT has resulted in inclusion and connection of these devices a predominant procedure. Though there are many advantages due to usage of IoT devices, there are different challenges as well due to its usage. Among the many existing challenges, Distributed Denial of Service(DDoS) attack is a relatively simple but very powerful technique to attack intranet and Internet resources. Usually, in this attack, the legitimate users are deprived of using web-based services by many compromised machines. DDoS attacks can be implemented in network, transport and application layers using different protocols, such as TCP, UDP, ICMP and HTTP.

The CIC-DDoS2019 dataset consists of 11 different DDoS attacks and benign traffic with 88 features. In this report, data for six DDoS attacks and benign data has been used. Info Gain Attribute Evaluator was used to extract the twenty-four most important features. The Machine Learning (ML) algorithms studied are Bayesian Network (BayesNet) , K-Nearest Neighbors (KNN) , J48. The experiments have been performed using the Waikato Environment for Knowledge Analysis (WEKA) tool with five-fold validation. Accuracy, Precision, Recall, F-measure, and execution time have been used as the performance metrics. From the results obtained, J48 performed better among all the algorithms in terms of accuracy, precision, recall and F-measure.

Contents

List of Tables

List of Figures

Abbreviation

AI..... Artificial Intelligence
API..... Application Programming Interface
CIC..... Canadian Institute for Cybersecurity
CNN..... Cable News Network
DDos..... Distributed Denial of Service
DNS..... Domain Name System
FN..... False Negative
FP..... False Positive
GUI..... Graphical User Interface
HTTP..... Hypertext Transfer Protocol
ICMP..... Internet Control Message Protocol
IDS..... Intrusion Detection System
IOT ..... Internet of Things
KNN..... K-Nearest Neighbor
LDAP..... Lightweight Directory Access Protocol
ML..... Machine Learning
MSSQL..... Microsoft SQL
MTC..... Machine Type Communication
NTP..... Network Time Protocol
SNMP..... Simple Network Management Protocol
SSDP..... Simple Service Discovery Protocol
TCP..... Transmission Control Protocol
TFTP..... Trivial File Transmission Protocol
TN..... True Negative
TP.....True Positive
UDP..... User datagram Protocol
WEKA..... Waikato Environment for Knowledge Analysis

Acknowledgement

First, I would like to thank my Supervisor, Dr. Fayez Gebali, whose valuable suggestions, guidance, and insights have helped me throughout my Master of Engineering project research and coursework.

Further, I would like to thank Legislative Assembly of British Columbia, Victoria, for giving me the opportunity to work as an intern. The experience has helped me gain invaluable skills and practical knowledge.

Finally, I would like to thank my Parents for all their support and motivation throughout my Master's.

CHAPTER 1

INTRODUCTION

The critical threats to many areas of our life such as IoT, healthcare, smart cities, information technology and commercial parts are Distributed Denial of service (DDoS) attacks [1]. Despite their size, because of their increases in complexity, volume, and frequency these attacks continue to threaten the network security of all the business sectors. The DDoS attacks are one of the top of threats due to the accessibility of business applications, services and networks. DDoS attacks and non malicious availability issues have a similarity between them, such as system administrators performing maintenance or technical problems with the network [2,3]. These problems make it extremely difficult to recognise and effectively defend against these kinds of attacks. When trying to identify a DDoS assault, the network speed for accessing files or the inaccessibility of a certain website may be poor [4]. Continuously, the attackers are increasing their computing capacities to carry out DDoS attacks. To protect the compromised IoT devices against attacks generated from them, intelligent security solutions need to be designed and developed. Hence, different machine learning (ML) algorithms in WEKA tool have been used in this project to analyze their detection performance for DDoS attacks using the CIC-DDoS2019 dataset. In this work, three classifiers Bayesian Network (BayesNet), K-Nearest Neighbors (KNN) , J48 have been assessed on the dataset to predict six attacks and valid (benign) to find the best effective machine learning technique.

1.1 Motivation

Cybercriminals have employed DDoS attacks to breach venture networks that can bring down the servers that are being targeted. Modern attacks are difficult for many firms to manage because of the magnitude and complexity of DDoS attacks, which are becoming more prevalent. Cybercriminals are aware of the newest technologies and their flaws since smart devices and IoT are particularly vulnerable to a variety of DDoS attacks due to resource limitations such as limited memory and processing capability [5]. In 2016, a number of organizations, including Netflix, CNN, and Twitter, had a nine-hour outage as a result of an attacks on their internet service providers. Numerous problems were brought on by this technological issue, including financial losses, productivity losses, brand damage, insurance rating drops, tense relationships between clients and providers, and going over IT budget [6]. DDoS attacks might be used by cybercriminals to prevent users from accessing a server or website [1]. We need to develop an IDS system to identify and stop DDoS attacks in order to safeguard data processing, information technology, and commercial components. Cybersecurity expenses will be greatly lowered if security teams use cutting-edge technology like ML, automation, and AI [7]. This project aims to evaluate performance of different machine learning (ML) algorithms in terms of accuracy, precision, recall, F-measure, and execution time in detection of DDoS attacks.

1.2 Related Work

The numerous studies done on the application of deep learning in intrusion detection (ID) of DDoS attacks has been presented in this section. The detection techniques used till now in Intrusion detection of DDoS attacks can be divided into three types: signature-based detection techniques, anomaly-based detection techniques and hybrid-based detection techniques.

Signature-Based Detection Techniques:

This type of detection technique uses a database of attack signatures to compare network traffic to the signatures in the database. A detection warning is raised when the match is discovered. Even if it is ineffective against current attack mutations, this method can detect known attacks for which signatures are recorded in the database, but it cannot discover zero day or new attacks [8].

A signature-based IDS was presented in this study [9] to find DDoS attacks in IoT networks. A hybrid deployment consists of two units: (i) IDS detectors and (ii) IDS routers. The border gateway houses the IDS router which is a firewall and a detection device. The IDS detectors uses sensors to keep an eye on internal traffic. The scheme's identification of version number change and hello flooding assaults is demonstrated by the results.

Anomaly-Based Detection Techniques:

This particular type of detection method is predicated on the baseline usual behaviour profile of the observed environment [10]. The system's network traffic activities at any particular time are then compared to this typical baseline. When compared to signature-based detection techniques, anomaly-based detection techniques are more effective in finding new threats. In anomaly-based detection techniques, ML algorithms are utilised to establish a baseline normal profile of the monitored systems. The deployment of ML algorithms in resource and energy limited IoT systems continues to be difficult due to the large computer resources needed to train and test them.

This study [11] proposed a real-time method for detecting wormhole attacks in RPL-based IoT. By analysing routing information and the Received Signal Strength Indicator, it may identify rogue users and nodes (RSSI). The real-time IDS systems are looked at in both centralized and scattered setups. A 90% detection rate is achieved.

Hybrid-Based Detection Techniques:

In order to overcome the drawbacks and to maximise the benefits of both current and novel attack detection, this type of detection technique combines the two earlier methods.

The authors of this study [12] developed a new technique based on unique device traffic characteristics for identifying DDoS traffic on various device classes. The classifications of machine type communication (MTC) traffic produced by IoT devices were evaluated by the study's authors. Their approach compared traffic variations produced by the IoT device to the legitimate traffic class that the device initially belonged to in order to determine if the observed IoT device generated legitimate or DDoS activity.

1.3 Report Organization

The report has been structured as follows.

Chapter 1 presented the issue and gave a general description of the project. The report's format was described, along with the project's objective and related work.

Chapter 2 provides the background, classification of DDoS attacks, introduces Machine Learning and gives details of the WEKA tool which has been used for detecting DDoS attacks.the proposed framework, and details about the dataset used for this project.

Chapter 3 presents the proposed framework, gives details about the dataset, data preprocessing filters, feature reduction technique and classifiers used in this project, explains data splitting as well.

Chapter 4 explains the method used to detect DDoS attacks as well as the test environment. Hardware and software configurations and the performance metrics have been presented as well as the results have been discussed.

Chapter 5 presents the conclusion and scope for future work.

CHAPTER 2

BACKGROUND

A distributed denial of service (DDoS) attack is a malicious effort to disable access to an online service for users, often by briefly suspending or halting the hosting server's operations. A DDoS attack is conducted from a large number of infected devices that are frequently dispersed globally in a botnet. It differs from other denial of service (DoS) attacks in which it only makes use of one network connection and one Internet-connected device to bombard a target with malicious traffic.

2.1 DDoS Attacks Classification:

This sub-section explains the detailed classification of DDoS attacks and illustrates the same in Figure 2.1, in terms of reflection-based DDoS attacks and exploitation-based DDoS attacks.

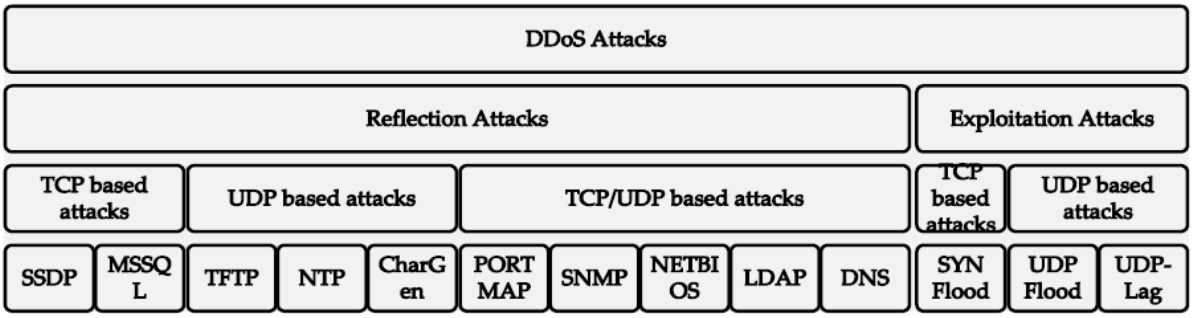

Figure 2.1. Classification of DDos Attacks [2]

2.1.1 Reflection-based DDoS Attacks: In these type of attacks cyberspace gadgets are used to transmit attack traffic such as HTTP calls to the target, and the identity of the attacker is hidden. The requests are sent from the source IP address targeting the IP addresses in the reflector servers(bots). Application layer protocols, such as Transmission control protocol (TCP), User datagram protocol (UDP), or a mix of both, can be used to carry out these attacks. As seen in Figure 2.1, attacks in this category that use TCP include MSSQL and SSDP whereas those that use UDP include CharGen, NTP, and TFTP. DNS, LDAP, NETBIOS, and SNMP are some attacks that can be carried out using either TCP or UDP [13], [14].

2.1.1.1 Transmission Control Protocol (TCP): Transmission Control Protocol, or TCP, is a communications standard that enables computer hardware and software to exchange messages over a network [15]. It is made to transfer packets across the internet and make sure that data and messages are successfully sent through networks. It provides end-to-end data transmission and is one of the most widely utilised protocols in digital network communications. The TCP based attacks are:

MSSQL: This attack is based on a technique that tampers with the Microsoft SQL Server Resolution Protocol in order to execute a reflection-based DDoS attack. The attempt to exploit the Microsoft SQL Server Resolution Protocol (MC-SQLR), which is listening on UDP port, happens when a Microsoft SQL Server answers to a query or request.

SSDP: A Simple Service Discovery Protocol (SSDP) attack is a reflection-based distributed denial-of-service (DDoS) attack that takes advantage of Universal Plug and Play (UPnP) networking protocols to send a greater volume of traffic to a targeted victim, overwhelming their infrastructure, and taking their web resource offline [16].

2.1.1.2 User Datagram Protocol (UDP): User Datagram Protocol (UDP) is a protocol which is used for communication throughout the internet. For time-sensitive activities like gaming, watching films, or Domain Name System (DNS) lookups, it was particularly chosen [17]. Because UDP does not need time to establish a secure connection with the destination before delivering the data, communication is faster. Since connecting to a network requires some time, skipping this step speeds up data flow. UDP based attacks are:

CharGen: A very outdated protocol called CharGEN Character Generator Protocol can be used to carry out magnified attacks. Sending tiny packets with a fake IP of the target to internet-enabled devices running CharGEN is how a CharGEN amplification attack is carried out. These fake requests are then used to cause these devices to respond to the target with UDP floods.

NTP: An internet protocol called Network Time Protocol (NTP) is used to synchronise with networked computer clock time sources. It is a component of the TCP/IP suite and one of the original components. Both the protocol and the client-server computer applications are referred to as NTP.

TFTP: Trivial File Transmission Protocol (TFTP) is a simple high-level data transfer protocol that servers utilise to start diskless workstations, X-terminals, and routers utilising User Data Protocol (UDP). The main purpose of TFTP is to read or write files to or from a remote server. TFTP, however, is a flexible protocol that may be used for a variety of different applications.

2.1.1.3 TCP/ UDP based attacks:

NetBIOS: A network service called NetBIOS (Network Basic Input/Output System) makes it possible for applications running to communicate with each other via a local area network (LAN). It was created in the 1980s to be used with early IBM PC networks. Microsoft adopted NetBIOS a few years later, and it eventually became the de facto industry standard. At the moment, NetBIOS is mostly restricted to a small number of legacy application use cases that continue to rely on the communication service package.

DNS: Domain names are converted to IP addresses via the Domain Name System (DNS), which browsers utilise to load internet pages. Every device which is connected to the internet has a unique IP address that other device may use to find the connected device. People may enter common phrases into their browsers, like Fortinet.com, thanks to DNS servers, saving them from having to remember the IP addresses of every website.

LDAP: An open, cross-platform standard called LDAP (Lightweight Directory Access Protocol) is used for directory services authentication. Applications can connect with other directory services servers using LDAP, which offers a communication language. Users, passwords, and computer accounts are stored by directory services, which also exchange this data with other network nodes.

SNMP: An Internet Standard protocol called Simple Network Management Protocol (SNMP) is used to control and monitor network devices linked through IP. Using SNMP, a variety of devices, including wireless devices, servers, CCTV cameras, load balancers, routers, switches, and firewalls, may interact. These devices' data are gathered, organised, and sent through SNMP for network monitoring, administration, and fault isolation. Both the monitored endpoints and the monitoring system rely on SNMP in one way or another.

2.1.2 Exploitation-based attacks: These kinds of attacks are ones in which the attacker's identity is hidden by using a reliable third-party component. Attackers send the packets to reflector servers with the target victim's IP address specified as the source IP address in an effort to overwhelm the target with response packets. These attacks may also be carried out utilising transport layer protocols, such as TCP and UDP, through application layer protocols. SYN Flood comes under TCP based exploitation attacks and UDP flood, UDP Lag come under UDP based attacks. By delivering several UDP packets to the remote host, a UDP flood attack is launched. These UDP packets are transmitted at a rapid rate to arbitrary ports on the target system. As a result, the network's available bandwidth is used up, the system crashes, and performance suffers. However, SYN flood also makes use of the TCP three-way handshake to drain server resources. This attack is launched by repeatedly delivering SYN packets to the target computer until the server malfunctions or crashes. An attack of this type that breaks up the connection between the client and the server is known as an UDP-Lag attack. This attack is frequently utilised in online gaming, when players try to hinder or obstruct rival players' movements in order to outwit them. Lag switches, a type of hardware switch, or a software programme that runs on the network and consumes other users' bandwidth are two ways to carry out this attack.

SYN Flood: A network-tier attack known as a SYN flood, often referred to as a half-open attack, involves flooding a server with connection requests while ignoring the related acknowledgements. Since there are so many open TCP connections as a result, the server's resources are quickly depleted, effectively crowding out legitimate traffic, making it impossible to establish new, authorised connections and making it difficult or impossible for those who are already connected to the server to use it properly.

UDP Flood: A User Datagram Protocol (UDP) flood is a type of volumetric Denial-of-Service (DoS) attack in which the attacker targets and floods arbitrary ports on the host with IP packets that include UDP packets. The host searches for apps connected to these datagrams in this kind of attack. The host sends a "Destination Unreachable" packet back to the sender if none are discovered. As a result of the cumulative effects of such a flood, the system overburdens and becomes unresponsive to legal traffic.

2.2 Machine Learning:

Machine Learning (ML) is an area of Artificial Intelligence (AI). ML is a method of data analysis that creates analytical models automatically. These models require little human interaction to learn from data, identify patterns, and make judgments [18][19]. The selection of features is the most crucial task in Machine Learning. Since ML algorithms are non-interactive and created based on the findings of training data, predictions are made using historical data. Making accurate predictions may be difficult [20]. In this project, three ML classifiers were used for detection of DDos attacks. ML classifiers can be divided into two categories, namely supervised learning, and unsupervised learning [21], as described below.

2.2.1 Supervised Learning:

The datasets which are labelled are utilised with supervised ML. To train the model and make predictions, this data is used. Typically, the result is a class or value. Numerous difficult issues may be resolved with supervised. Examples of supervised ML classifiers include Random Forest (RF), Logistic Regression, Neural Networks, Linear Regression, Naive Bayes, and Support Vector Machine (SVM). The below figure 2.2, will help understand what Supervised Learning is in a very simple way.

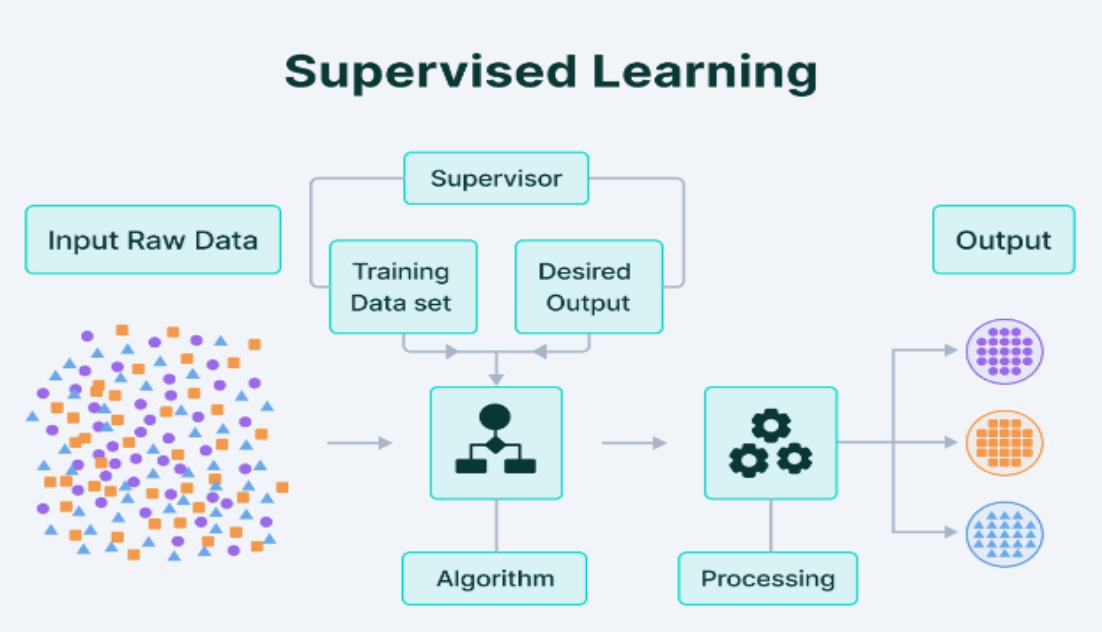

Figure 2.2 Model of Supervised Machine Learning [18].

2.2.2 Unsupervised Learning:

The datasets which are Unlabelled are utilised with unsupervised ML methods. These algorithms can find hidden patterns without the help of a person. They are frequently employed in data analysis, product marketing techniques, pattern identification, and client segmentation due to their capacity to spot variances and similarities in data. Feature extraction using dimensionality reduction is another use of unsupervised learning. K-means clustering, and probabilistic clustering are examples of unsupervised ML techniques [22].

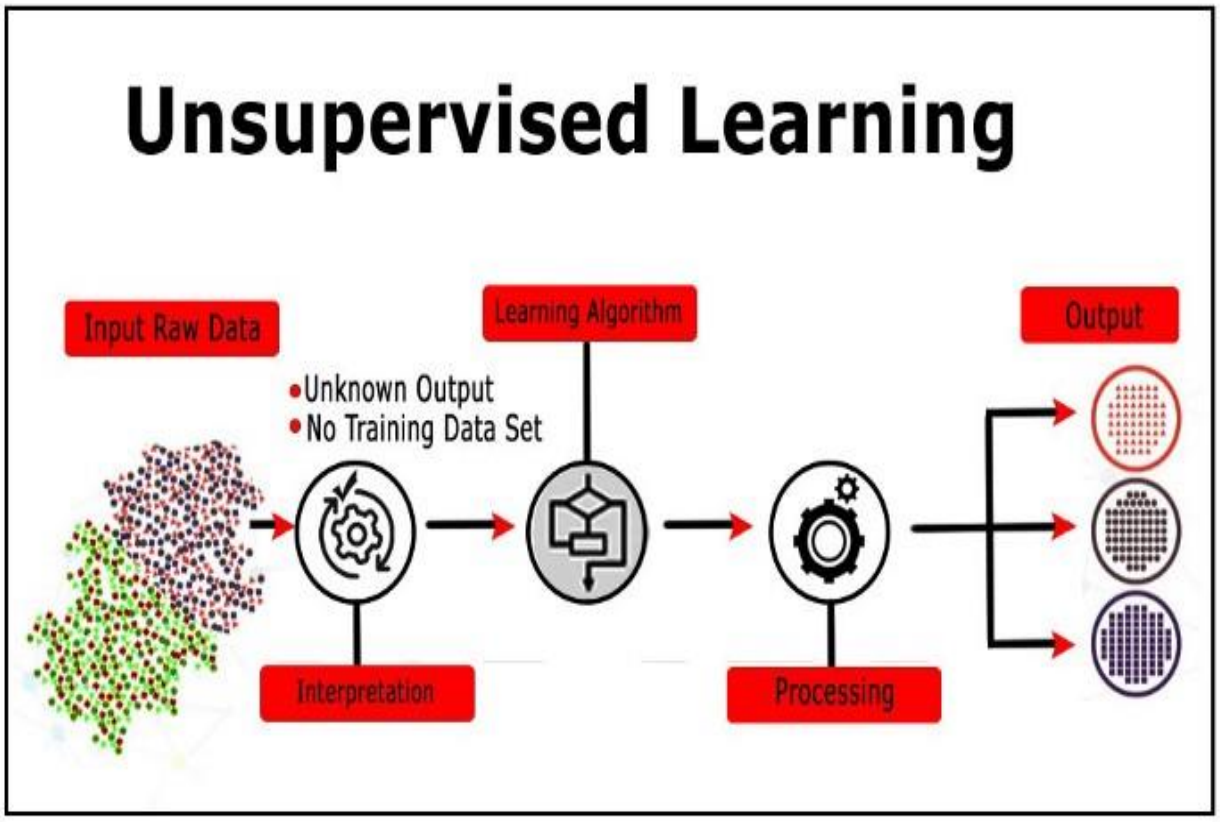

Figure 2.3. Model of Unsupervised Machine Learning [20].

2.3 The WEKA Machine Learning Tool:

Waikato Environment for knowledge Analysis (or) WEKA is a free and open-source machine learning tool which has been built on the Java platform. It can be accessed by a Java Application Programming Interface (API), a Graphical User Interface (GUI), or other standard terminal programmes [27]. It was created at the University of Waikato in New Zealand and has since seen widespread use in applications in academia, science, and industry. It consists of supervised, semi-supervised, and unsupervised machine learning classifiers such as Decision Tree, Linear Regression, Random Forest, KNN, and BayesNet. This technology is quite potent since these classifiers may be adjusted by modifying their parameters, or hyperparameters. Although tuning can increase classifier accuracy, this is frequently done experimentally because it highly depends on the Machine Learning problem [28]. The Graphical User Interface of the WEKA tool has been shown in Figure 2.4.

Figure 2.4: Graphical User Interface of WEKA

One of the applications in WEKA GUI is the "Explorer" which is used to analyze the data from the dataset we have by uploading the dataset in WEKA. The dataset can be chosen from the pre-process panel. The pre-processing tab in WEKA also allows us to use filters like RemoveDuplicates , Randomize, Normalize, etc. in order to process the data on the instances and the attributes. RemoveDuplicates and Randomize filters have been used in this project. Various formats of the dataset like .csv and .arff are supported by WEKA.

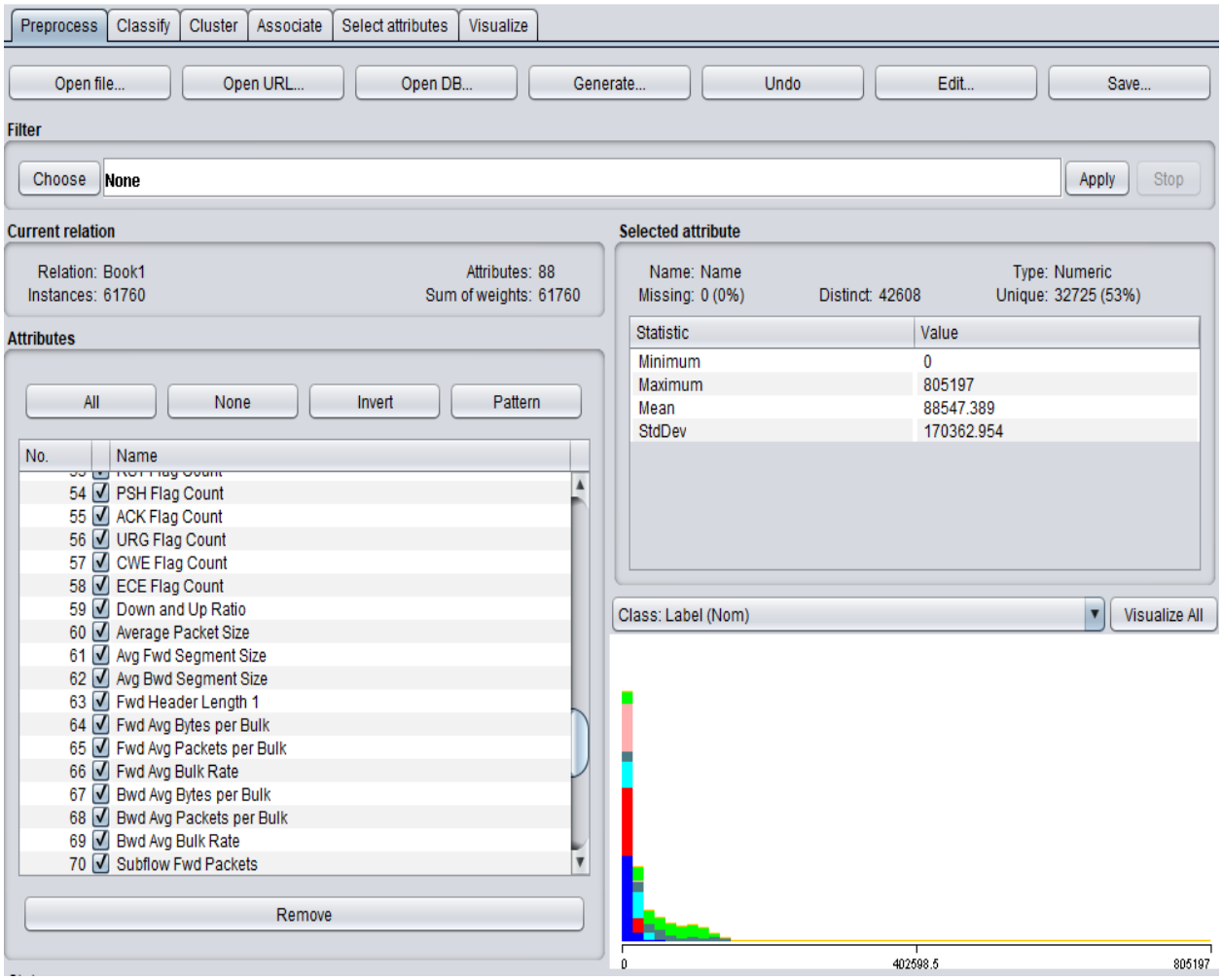

Figure 2.5: WEKA Explorer Pre-process panel

Figure 2.5 highlights the WEKA explorer pre-process panel. This panel provides dataset information such as the number of features and classes. Figure 2.6 visualizes the data within our dataset in WEKA.

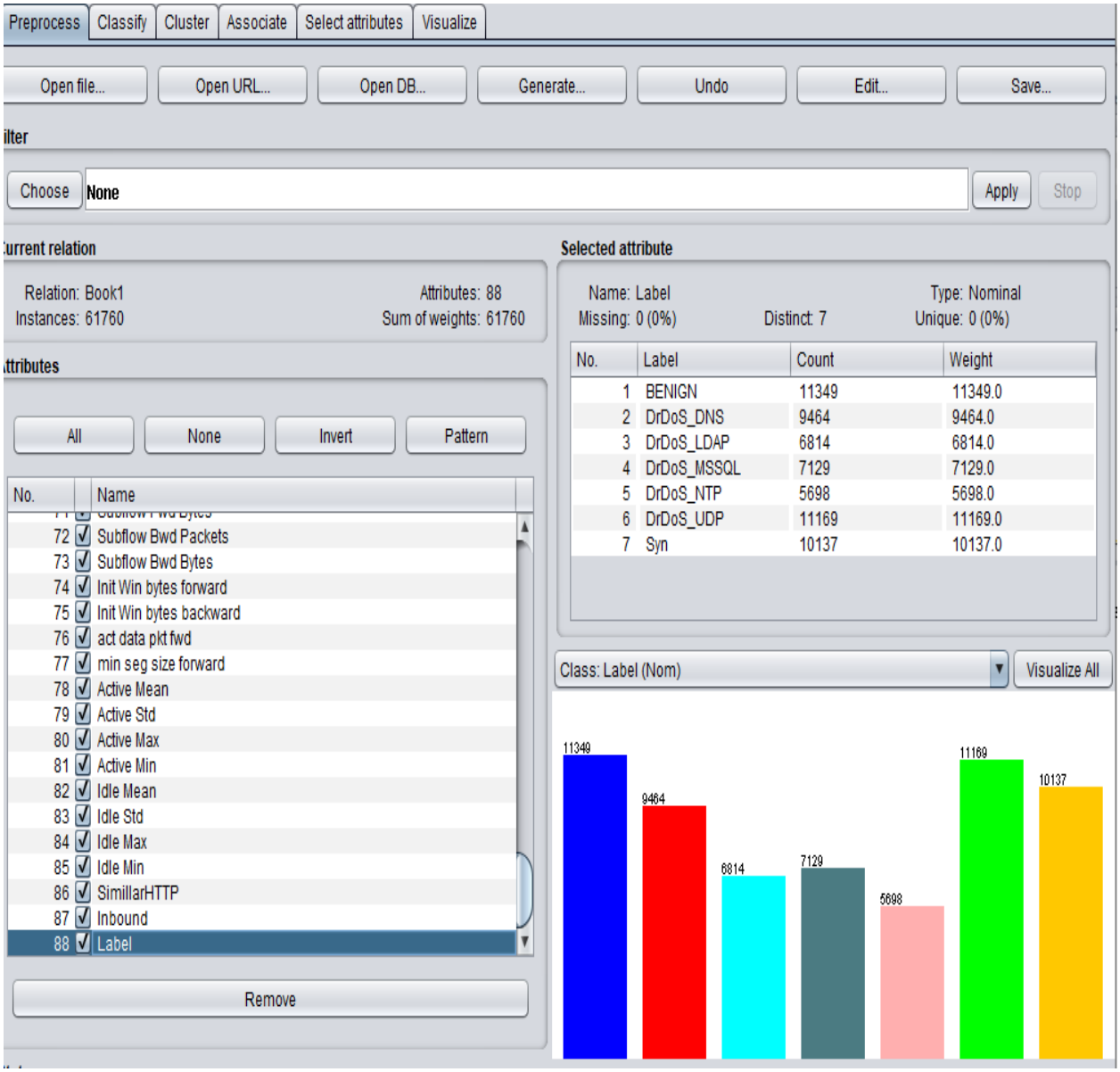

Figure 2.6: Data Visualization in WEKA

CHAPTER 3

PROPOSED FRAMEWORK

Figure 3.1 gives the proposed framework for training and testing the ML models. The first step is to preprocess the CIC-DDoS 2019 dataset. The WEKA tool is used for preprocessing and building the ML models. All the steps used in the proposed framework have been discussed in detail in the next sub-sections.

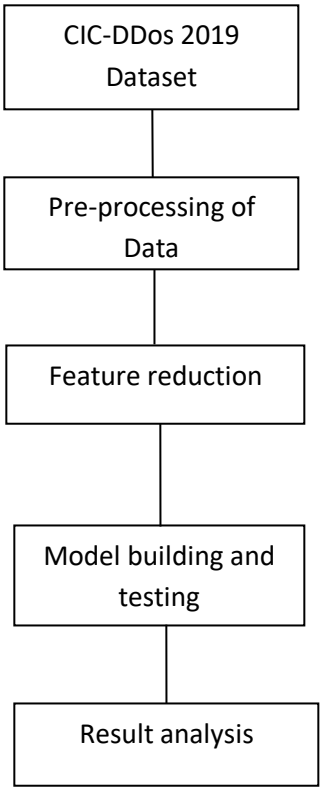

Figure 3.1 Proposed Framework

3.1 CIC-DDoS2019 Dataset:

The CIC-DDoS2019 dataset was collected from the University of New Brunswick Canadian Institute for Cybersecurity. To forecast DDoS attacks, this complete dataset contains 50,063,112 instances with 88 features and 11 class labels.

The dataset consists of 11 different DDoS attacks (DNS, LDAP, MSSQL, NetBIOS, NTP, SNMP, SSDP, UDP, Syn, TFTP, UDPLag) and benign traffic with 88 features. For this project, six attacks and benign data has been considered. A detailed overview of the attacks has been discussed in the previous sections. Table 3.1 shows the attack names and benign traffic and the number of instances which have been used in this project.

Name of Attacks	Number of Instances
BENIGN	11,349
DDos_DNS	9464
DDos_LDAP	6814
DDos_MSSQL	7129
DDos_NTP	5698
DDos_UDP	11169
Syn	10137

Table 3.1: Attack Names and Benign Data and number of instances used

3.2 Data Preprocessing:

One of the most critical steps before data analysis is data pre-processing. Raw data contains a significant amount of noise, duplicate values, missing values, inaccuracy, etc. in addition to important and usable information [29]. Therefore, to increase the effectiveness and simplicity of data analysis, improving the quality of the raw data is important [30]. It can be accomplished significantly and effectively with data pre-processing [31].

Different approaches for different purposes are used in the data pre-processing process. In this project for data pre-processing data cleansing has been used. Data cleansing, also called data scrubbing or data clean, which is used to process raw data by detecting errors, eliminating duplicated data, filling the missing data, or removing invalid data [32].

RemoveDuplicates: In order to check whether we have duplicate values or not in the dataset the filter "RemoveDuplicates" in the WEKA tool was used, but as there were no duplicate values in the dataset, the number of instances remained the same even after applying the filter. Figure 3.2 shows the use of 'RemoveDuplicates' filter.

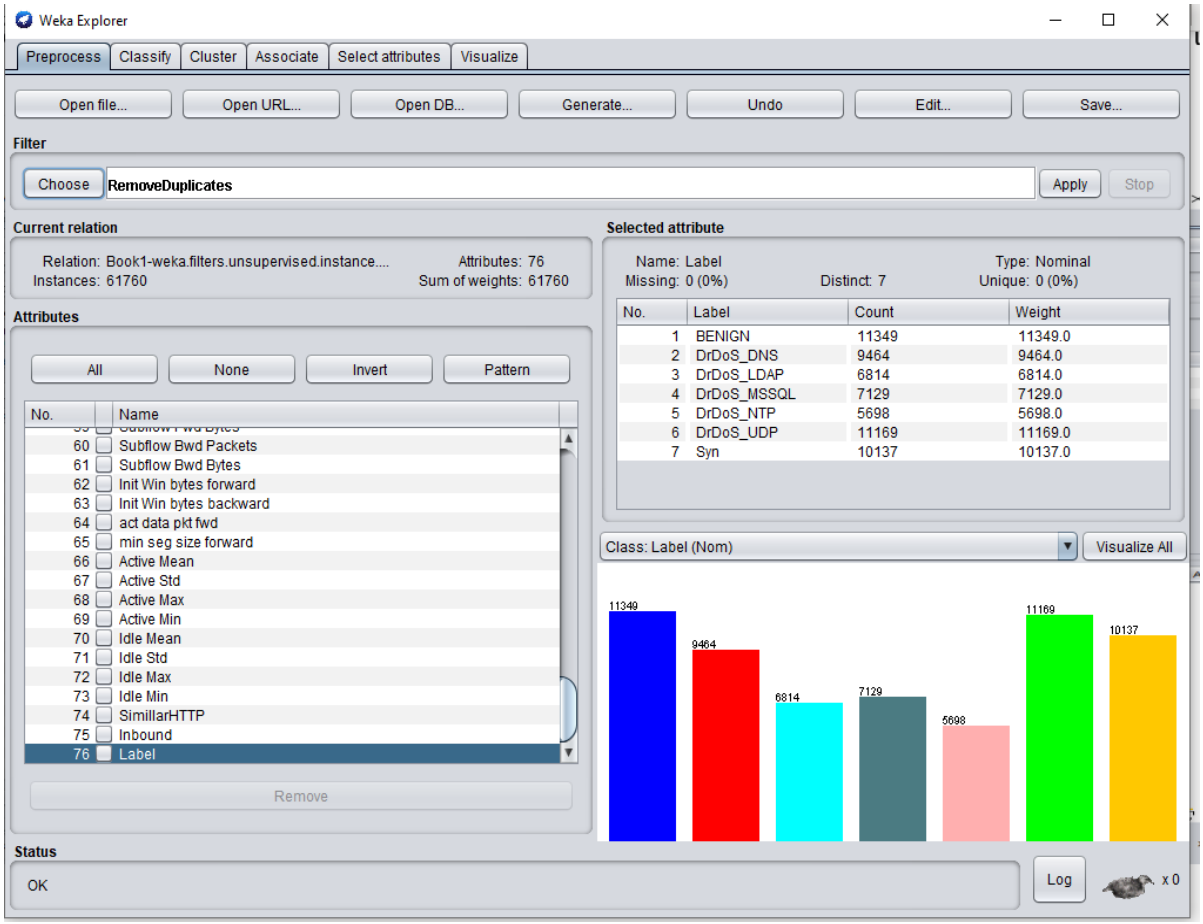

Figure 3.2: RemoveDuplicates Filter

Removing Attributes with Zero Instances: The next step taken in data cleaning process was to remove the attributes from the dataset which had zero number of instances. In the dataset used there were 12 such attributes which had zero instances. All these attributes were removed, and the total number of instances reduced to “76” from “88”. Figure 3.3 shows the removal of attributes with zero instances.

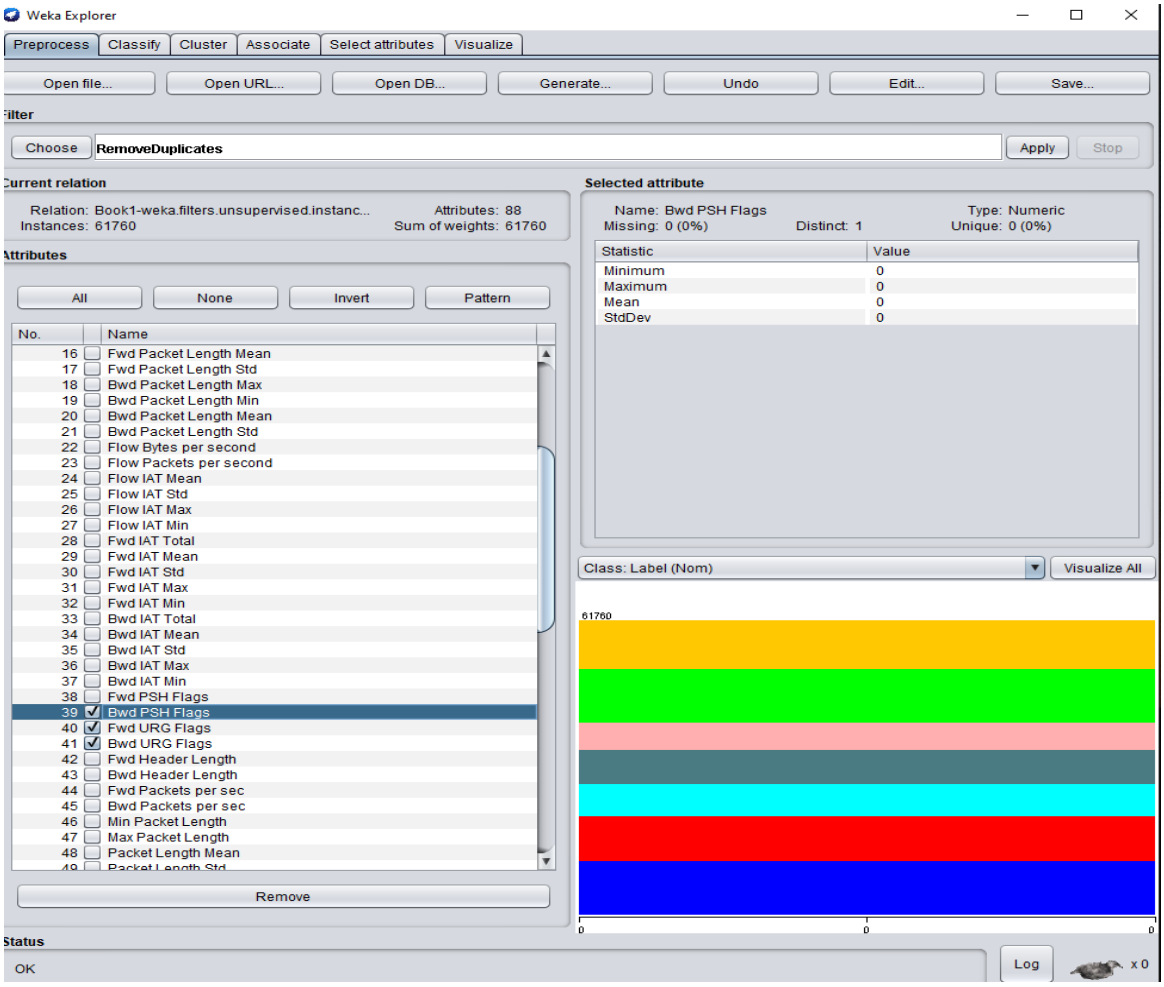

Figure 3.3 Removing Zero Instances Attributes

Randomize: After checking for duplicate values and removing the attribute with zero instances, the instances order was randomized to make sure that the output obtained is accurate. As we can see in the Figure 3.4 all the data points are related to "BENIGN". To avoid this the data has been randomized to get an accurate output. Once, the randomize filter has been used from the Figure 3.5, we can see that the data points are now in a random order.

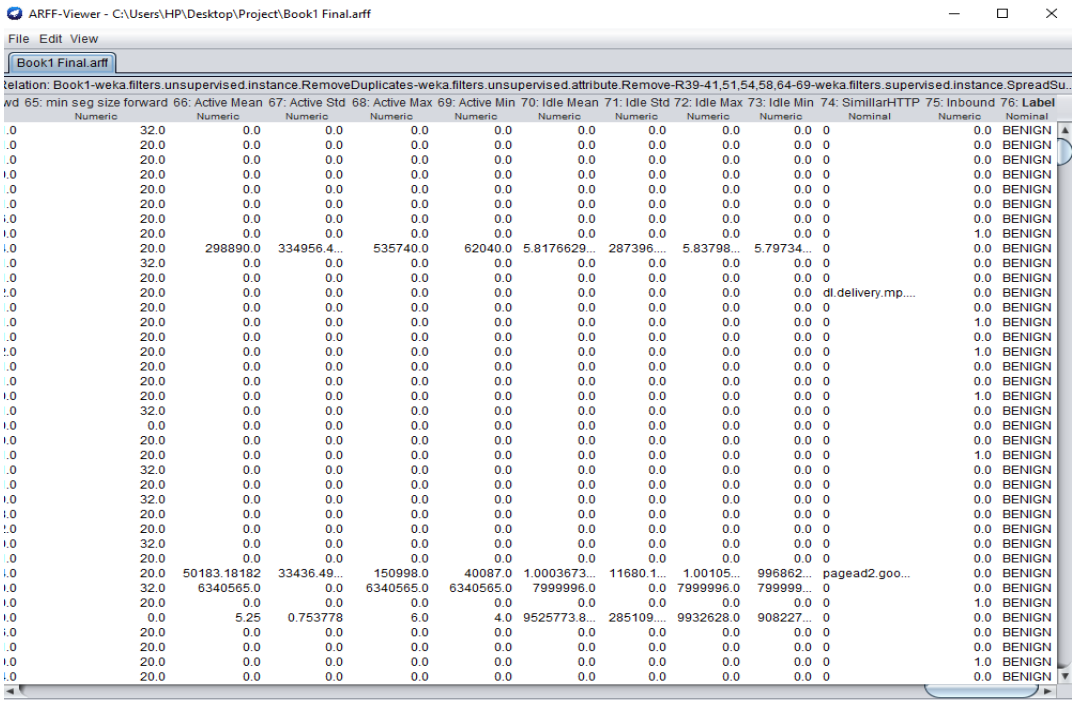

Figure 3.4 Before using 'Randomize' filter

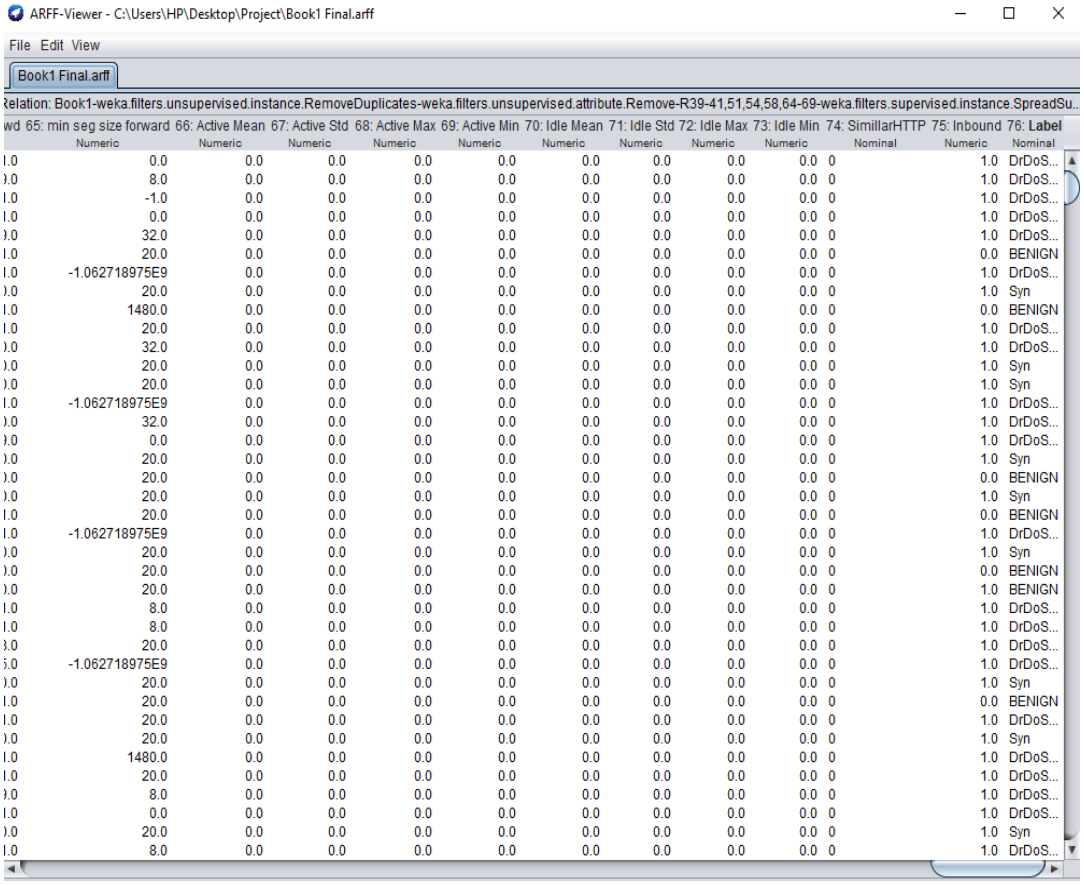

Figure 3.5 After using 'Randomize' filter

3.3 Feature Reduction:

3.3.1 InfoGainAttributeEval:

"InfoGainAttributeEval" measures the information gain in relation to the class to assess the value of an attribute. Machine Learning models are more efficient at exploring and visualizing smaller datasets with extraneous features removed [33]. Hence, for feature reduction, as shown in Figure 3.6 "InfoGainAttributeEval" was used. Firstly, in order to use the Info Gain Attribute Evaluator, we must also use the Ranker search method. Here the attributes which will contribute the most information for the model will have a higher information gain value and ranking in the ranking filter and the attributes which contribute less information will have a lower information gain value. Hence, in order to make use of the attributes which can contribute the most information and increase the chances of building a good testing model, I set the threshold as "1" which means the attributes with Information Gain Ranking "1" or above will only be used further for building/testing our model. Hence, by using threshold value as 1 the number of features got reduced to 24.

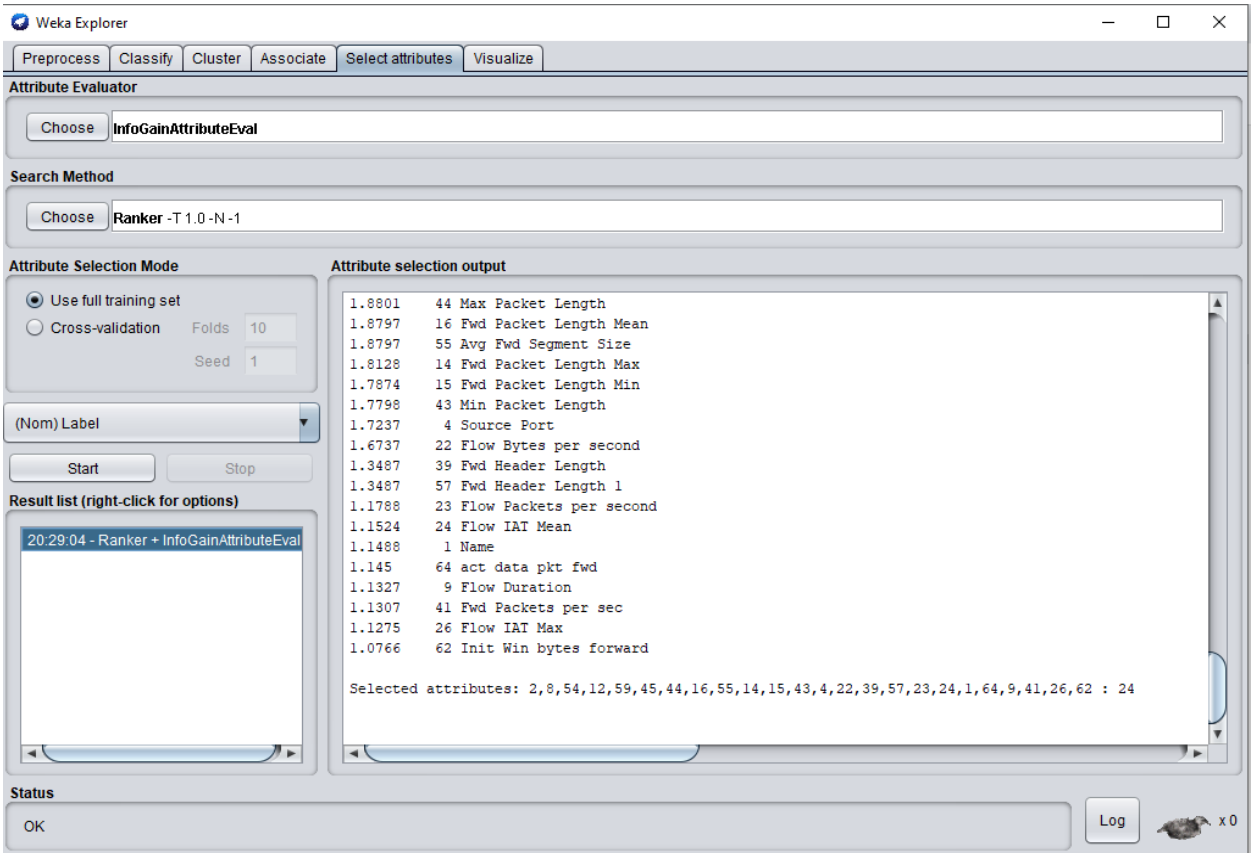

Figure 3.6: Attribute selection using InfoGainAttributeEval

Table 3.2 lists the features selected based on InfoGainAttributeEval feature reduction technique. The top 24 features were further used to train the classifiers.

Number	Attribute Name	Ranking
1	Flow ID	2.7588
2	Timestamp	2.7227
3	Average Packet Size	1.9312
4	Total Length of Fwd Packets	1.8884
5	Subflow Fwd Bytes	1.8884
6	Packet Length Mean	1.8838
7	Max Packet Length	1.8801
8	Fwd Packet Length Mean	1.8797
9	Avg Fwd Segment Size	1.8797
10	Fwd Packet Length Max	1.8128
11	Fwd Packet Length Min	1.7874
12	Min Packet Length	1.7798
13	Source Port	1.7237
14	Flow Bytes per second	1.6737
15	Fwd Header Length	1.3487
16	Fwd Header Length 1	1.3487
17	Flow Packets per second	1.1788
18	Flow IAT Mean	1.1524
19	Name	1.1488
20	act data pkt fwd	1.145
21	Flow Duration	1.1327
22	Fwd Packets per sec	1.1307

23	Flow IAT Max	1.1275
24	Init Win bytes forward	1.0766

Table 3.2: Features selected based on InfoGainAttributeEval

3.4 Data Splitting:

It is crucial to decide how to divide the data for training and testing. The most popular techniques are percentage split and k-fold cross-validation. Data may be divided into training and test sets simply by using percentages, with the ratios 80:20 and 70:30 being popular choices. In this work, k-fold cross-validation is used with k = 5. This has been used to produce less biased results than percentage split [25]. With one partition serving as the test set, a cross-validation model is trained using k partitions. The results for the k various test partitions are averaged to get the cross-validation accuracy [26]. Figure 3.7 shows the folds for k-fold cross-validation with k = 5.

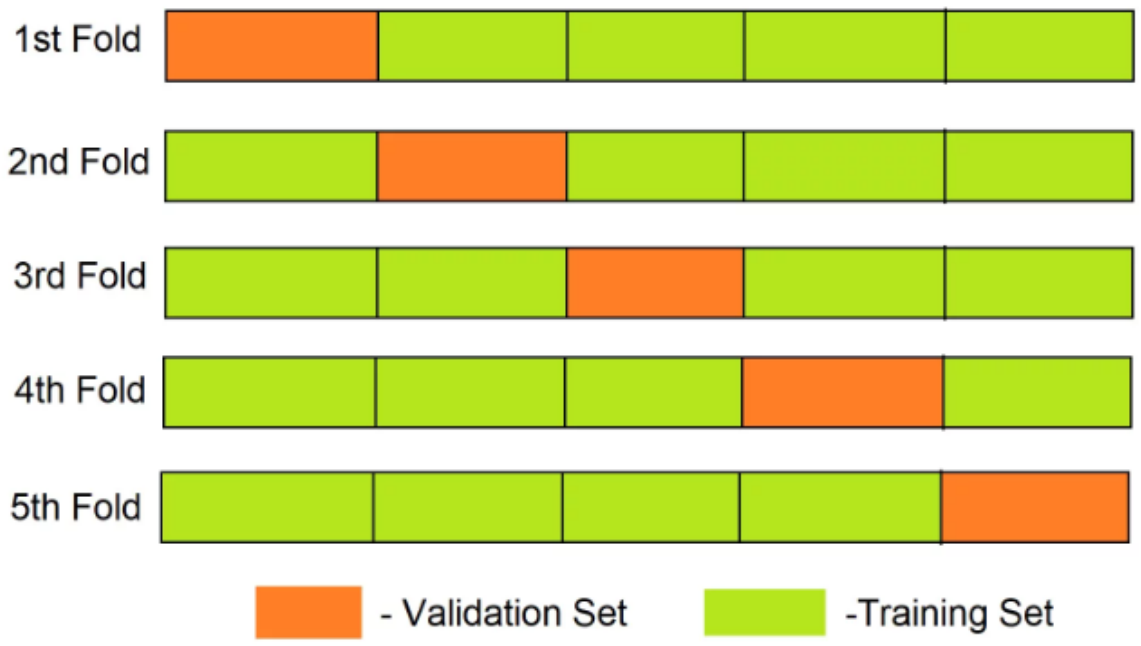

Figure 3.7: K-fold cross-validation with k = 5.

3.5 Machine Learning Classifiers:

In this project, three ML classifiers have been used for data classification, namely Bayesian Network (BayesNet), K-Nearest Neighbors (KNN) and J48. These classifiers are widely used in practice for supervised learning purposes. In addition, these classifiers have been shown to perform well for such classification problems.

3.5.1 Bayesian Network (BayesNet):

In order to resolve complicated issues, a Bayesian network classifier takes into account a collective probability model. The network is made up of nodes and the causal connections between them, which stand in for random variables and edges, respectively. Information from random variables associated to the nodes and statistical probability connected to the edges are intended to be provided. When it comes to modelling probabilistic relationships and estimating the likelihood that potential known causes will be contributing factors, Bayesian networks excel.

3.5.2 K-Nearest Neighbors (KNN):

Problems involving classification and regression can be solved with KNN. KNN determines all of the points distances from the unknown data and selects the ones with the least distances. It is sometimes referred to as a distance-based algorithm as a result. With missing values, KNN classifier training can be time-consuming. If the dataset is particularly large, KNN can also be computationally costly in terms of both time and storage. When it comes to other supervised learning methods, this is typically not the case [23].

3.5.3 J48:

The J48 method is used to accurately produce classification results for a variety of applications. One of the best machine learning algorithms for categorical and ongoing data analysis is the J48 algorithm. It consumes more memory when used for this purpose, which lowers its performance and accuracy when categorising data [24].

3.6 Model Building and Testing:

In this step, the models were built and tested on the CIC-DDoS2019 dataset before and after the data was pre-processed and feature reduction technique was used. To evaluate the classifiers metrics, five-fold cross-validation has been used for training and testing.

CHAPTER 4

PERFORMANCE EVALUATION

In this chapter, the performance results have been presented and discussed. Three ML classifiers with default parameters are used namely BayesNet, KNN (K = 1) and J48. The results were obtained using a personal computer with the hardware and software specifications given in Table 4.1. The CIC-DDoS2019 dataset has been used to evaluate the ML models. There are seven classes of DDoS attributes including malicious and benign. Table 2.1 shows the number of instances in the classes. After processing, the number of benign instances are 11349, DNS instances are 9464, LDAP instances are 6814, MSSQL instances are 7129, NTP instances are 5698, UDP instances are 11169 and Syn instances are 10137.

Item	Specification
Manufacturer	HP
Model	Pavilion Notebook - 15-p211nx
Operating System	Windows 10 Professional
System Type	64-bit Operating System, x64-based Processor
Processor Type	Intel(R) Core(TM) i7-5500U
Installed Memory (RAM)	8 GB
Processor Speed	2.40GHz
Number of Cores	2
Number of Threads	4
Machine Learning Tool	WEKA version 3.9.4

Table 4.1: Hardware and Software Specifications

4.1 Evaluation Metrics:

The performance metrics used have all been expressed as percentages and are as follows.

Precision is the ratio of true positives to the sum of true positives and false positives

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

where true positive (TP) is the number of DDoS instances correctly classified and false positive (FP) is the number of incorrect classifications of benign instances as an attack.

Recall is the ratio of true positives to the sum of true positives and false negatives

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

where false negative (FN) is the incorrect classification of an attack as a benign instance.

Accuracy is the number of correct classifications of either as a DDoS attack instance or benign instance out of all instances in the dataset

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

where true negative (TN) is correct classification of benign instances as benign.

Execution Time is the required time to train and test the classification model.

F-Measure is the harmonic mean of recall and precision

$$F\text{-}Measure = \frac{2TP}{2TP+FP+FN}$$

4.2 Performance of the Classifiers with 5-Fold Cross-Validation without Data Preprocessing and Feature Selection:

In this section, the performance of the classifiers for the data used from CIC-DDoS2019 dataset without applying any Data Preprocessing and any Feature Selection has been presented. The results obtained from the ML classifiers have been show in Table 4.2. From the results obtained, we can see that J48 classifier has the highest Accuracy, Precision, Recall and F-Measure at 83.28, 83.3, 83.3, 83.2 which is followed by Bayes Net which has Accuracy 79.62, Precision, 82.5, Recall 79.6 and F-Measure 79.7 and KNN at 78.56, 78.8, 78.6,77.2. Whereas, when we see the execution times of these classifiers, KNN had the lowest at 0.03 seconds followed by BayesNet at 6.02 seconds and J48 at 41.39 seconds.

Classifier	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F-Measure (%)**	**Execution Time (seconds)**
BayesNet	79.62	82.5	79.6	79.7	6.02
KNN	78.56	78.8	78.6	77.2	0.03
J48	83.28	83.3	83.3	83.2	41.39

Table 4.2: Performance of the ML Classifiers with 5-Fold Cross-Validation without Data Preprocessing and Feature Selection

4.3 Performance of the Classifiers with 5-Fold Cross-Validation after Data Preprocessing and Feature Selection:

In this section, the performance of the classifiers for the data used from CIC-DDoS2019 dataset after Data Preprocessing and Feature Selection has been presented. For preprocessing of the data present in the dataset, filters like “RemoveDuplicates” , “Randomize” and removing attributes which have zero values were used. Initially, the total number of attributes in the dataset were 88, after removing the attributes which have zero values, the number of attributes decreased to 76 from 88 and then the “Randomize” filter was used to randomly shuffle the order of instances within the dataset. It was used to make sure to get an accurate output while testing the model. Later, for feature reduction “InfoGainAttributeEval” was used through which the number of features got reduced to 24 from 76. The results obtained from the ML classifiers have been show in Table 4.3. From the results obtained, we can see that J48 classifier has the highest Accuracy, Precision, Recall and F-Measure at 98.31, 98.3, 98.3, 98.3 which is followed by KNN which has Accuracy 96.45, Precision, 96.6, Recall 96.5 and F-Measure 96.4 and BayesNet at 91.7, 93.9, 91.7,91.8 Whereas, when we see the execution times of these classifiers, BayesNet had the lowest at 2.47 seconds followed by KNN at 3.01 seconds and J48 at 5.32 seconds.

Classifier	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F-Measure (%)**	**Execution Time (seconds)**
BayesNet	91.7	93.9	91.7	91.8	2.47 seconds
KNN	96.45	96.6	96.5	96.4	3.01 seconds
J48	98.31	98.3	98.3	98.3	5.32 seconds

Table 4.3: Performance of the ML Classifiers with 5-Fold Cross-Validation after Data Preprocessing and Feature Selection

4.4 Discussion:

Without any data preprocessing and feature reductions, J48 classifier performed the better among all the classifiers in terms of Accuracy, Precision, Recall and F-measure followed by BayesNet and KNN. But, interms of execution time KNN was the fastest with 0.03 seconds, followed by BayesNet and J48 with 6.02 and 41.39 seconds respectively. The same has been presented in Table 4.2.

After data preprocessing and reduction of the features, J48 again performed the better among all the classifiers in terms of Accuracy, Precison, Recall and F-Measure followed by KNN and BayesNet this time. This time KNN fetched better results than BayesNet in terms of Accuracy, Precision, Recall and F-measure. In terms of execution time BayesNet was the fastest with 2.47 seconds after data preprocessing and reduction of features, followed by KNN with 3.01 seconds and J48 with 5.32 seconds.

We can observe that we got a substantial increase in Accuracy for all the classifiers once the data was preprocessed and the feature set was reduced using evaluation techniques. Also, the execution time was comparatively faster for both BayesNet and J48 after the data was preprocessed and reduced features. Overall, J48 performed better in both the results fetched in terms of accuracy and precision though it did have slower execution times.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This project has considered Machine Learning using the Waikato Environment for Knowledge Analysis (WEKA) tool for DDos attacks detection. The CIC-DDoS2019 dataset was used in this project for evaluation. The whole dataset consists of 11 different DDoS attacks (DNS, LDAP, MSSQL, NetBIOS, NTP, SNMP, SSDP, UDP, Syn, TFTP, UDPLag) and benign traffic with 88 features. For this project six attacks and benign data were considered. The dimensionality of the dataset was reduced using InfoGainAttributeEval. Three supervised Machine learning classifiers were used, namely BayesNet, KNN and J48. Accuracy, Precision, F-Measure, Recall and Execution time were considered as the performance metrics. Data pre-processing was done by applying some filters and Feature reduction method was used to reduce the features and 5-fold cross validation was used for evaluation.

From the results obtained, J48 using both 88 features and 24 features and 5-fold cross-validation gave the highest accuracy at 83.28 % and 98.31% respectively with an execution time of 41.39 seconds and 5.32 seconds. BayesNet using 88 features and 5-fold cross-validation gave the second-best accuracy of 79.62% with an execution time of 6.02 seconds, whereas with 24 features and 5-fold cross-validation it gave the lowest accuracy among the three classifiers at 91.7% with an execution time of 2.47 seconds.KNN using 88 features and 5-fold cross-validation gave the lowest accuracy among the three classifiers of 78.56% with the fastest execution time of 0.03 seconds. Among all the classifiers, J48 performed the best of the three classifiers in terms of accuracy with a moderate execution time.

For future work, gathering and examining real-time packets against the classified training datasets can be done. Deep Learning Techniques can be utilized for analysis of the data. Splitting techniques can be used to obtain training and testing sets instead of using the *k*-fold cross-validation. Detecting attacks using Unsupervised Machine Learning can also be considered.

BIBLIOGRAPHY

[1]. Covington, M.J.; Carskadden, R. Threat implications of the Internet of Things. In Proceedings of the 2013 5th International Conference on Cyber Conflict, Tallinn, Estonia, 4–7 June 2013; pp. 1–12.

[2]. Anstee, D.; Escobar, J.; Sockrider, C. 10th Annual Worldwide Infrastructure Security Report. 2015. Available online: https://www.netscout.com/blog/cloud-crosshairs.

[3]. Mouli, V.R.; Jevitha, K. Web Services Attacks and Security- A Systematic Literature Review. Procedia Comput. Sci. 2016, 93, 870–877.

[4]. Oliveira, R.A.; Laranjeiro, N.; Vieira, M. Assessing the security of web service frameworks against Denial of Service attacks. J.Syst. Softw. 2015, 109, 18–31. Available online: https://www.sciencedirect.com/science/article/pii/S0164121215001454 (accessed on 26 October 2021).

[5]. Subbulakshmi, T.; Balakrishnan, K.; Shalinie, S.M.; Anandkumar, D.; Ganapathisubramanian, V.; Kannathal, K. Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset. In Proceedings of the 3rd International Conference on Advanced Computing, ICoAC 2011, Chennai, India, 14–16 December 2011; pp. 17–22.

[6]. Gupta, B.; Joshi, R.C.; Misra, M. Defending against Distributed Denial of Service Attacks: Issues and Challenges. Inf. Secur. J. A Glob. Perspect. 2009, 18, 224–247.

[7]. Samtani, S.; Kantarcioglu, M.; Chen, H. Trailblazing the Artificial Intelligence for Cybersecurity Discipline. ACM Trans. Manag. Inf. Syst. 2020, 11, 1–19.

[8]. Kumar, R.; Kumar, P.; Tripathi, R.; Gupta, G.P.; Kumar, N.; Hassan, M.M. A Privacy-Preserving-Based Secure Framework Using Blockchain-Enabled Deep-Learning in Cooperative Intelligent Transport System. IEEE Trans. Intell. Transp. Syst. 2021.

[9]. Ioulianou, P.; Vasilakis, V.; Moscholios, I.; Logothetis, M. A Signature-based Intrusion Detection System for the Internet of Things. Jun 2018. Available online: https://eprints.whiterose.ac.uk/133312

[10]. Keshk, M.; Turnbull, B.; Moustafa, N.; Vatsalan, D.; Choo, K.-K.R. A Privacy-Preserving-Framework-Based Blockchain and Deep Learning for Protecting Smart Power Networks. IEEE Trans. Ind. Inform. 2019, 16, 5110–5118

[11]. Deshmukh-Bhosale, S.; Sonavane, S.S. A Real-Time Intrusion Detection System for Wormhole Attack in the RPL based Internet of Things. Procedia Manuf. 2019, 32, 840–847

[12]. Cviti´c, I.; Perakovi´c, D.; Periša, M.; Botica, M. Novel approach for detection of IoT generated DDoS traffic. Wirel. Netw. 2019, 27, 1573–1586

[13] “Request for comments: 1001 (rfc1001),” in PROTOCOL STANDARDFOR A NetBIOS SERVICE ON A TCP/UDP TRANSPORT: CONCEPTS AND METHODS, 1987.

[14] “Request for comments: 7766 (rfc7766),” in DNS Transport over TCP - Implementation Requirements, 2016.

[15] I. Martinez and V. Ramos, "Choosing a TCP Version over Static Ad Hoc Wireless Networks: Wired TCP or Wireless TCP?," 2013 Seventh International Conference on Next Generation Mobile Apps, Services and Technologies, 2013, pp. 170-174, doi: 10.1109/NGMAST.2013.38.

[16] S. Saruwatari, J. Hjelm, T. Oda and H. Morikawa, "A system for logging operation histories of DLNA devices by combining ARP spoofing and SSDP," 2011 IEEE International Conference on Consumer Electronics (ICCE), 2011, pp. 233-234, doi: 10.1109/ICCE.2011.5722557.

[17] N. M. Garcia, F. Gil, B. Matos, C. Yahaya, N. Pombo and R. I. Goleva, "Keyed User Datagram Protocol: Concepts and Operation of an Almost Reliable Connectionless Transport Protocol," in IEEE Access, vol. 7, pp. 18951-18963, 2019, doi: 10.1109/ACCESS.2018.2886707.

[18] S. Angra, and S. Ahuja, Machine Learning and its Applications, International Conference on Big Data Analytics and Computational Intelligence, pp. 57-60, Chirala, Andhra Pradesh, India, 2017.

[19] E. Alpaydin, Introduction to Machine Learning, MIT Press, Cambridge, MA, USA, 2020.

[20] R. E. Schapire, The Boosting Approach to Machine Learning: An Overview in Non-linear Estimation and Classification, pp. 149-171, Springer, New York, NY, USA, 2003.

[21] C. M. Bishop, Pattern Recognition and Machine Learning, 2006, https://link.springer.com/book/9780387310732.

[22] S. Ahmed, Y.Lee, S. Hyun, and I. Koo, Unsupervised Machine Learning-based Detection of Covert Data Integrity Assault in Smart Grid Networks Utilizing Isolation Forest, IEEE Transactions on Information Forensic and Security, vol. 14 no. 10, pp. 2765-2777, 2019.

[23] K. Taunk, S. De, S. Verma, and A. Swetapadma, A Brief Review of Nearest Neighbor Algorithm for Learning and Classification, International Conference on Intelligent Computing and Control Systems, pp. 1255-1260, Madurai, India, 2019.

[24]. N. Saravanan and V. Gayathri, Performance and Classification Evaluation of J48 Algorithm and Kendall's Based J48 Algorithm (KNJ48), International Journal of Computational Intelligence and Informatics, Vol. 7: No. 4, March 2018.

[25]. T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, New York, NY, USA, 2009.

[26] S. Gupta, Decision Trees Towards Data Science, 2017, https://medium.com/towards-data-science/decision-trees-in-machine-learning-641b9c4e8052.

[27] I. Charalampopoulos and I. Anagnostopoulos, "A Comparable Study Employing WEKA Clustering/Classification Algorithms for Web Page Classification," 2011 15th Panhellenic Conference on Informatics, 2011, pp. 235-239, doi: 10.1109/PCI.2011.52.

[28] A. K. Pandey, D. S. Rajpoot and D. S. Rajpoot, "A comparative study of classification techniques by utilizing WEKA," 2016 International Conference on Signal Processing and Communication (ICSC), 2016, pp. 219-224, doi: 10.1109/ICSPCom.2016.7980579.

[29] B. Saha and D. Srivastava, "Data quality: The other face of big data", IEEE 30th International Conference on Data Engineering Chicago ICDE 2014, pp. 1294-1297, March 31 - April 4, 2014, 2014.

[30] H. Zou, Y. Yu, W. Tang and H.-W. M. Chen, "Flexanalytics: A flexible data analytics framework for big data applications with i/o performance improvement", Big Data Research, vol. 1, pp. 4-13, 2014.

[31] T. Iliou, C.-N. Anagnostopoulos, I. M. Stephanakis and G. Anastas-sopoulos, "A novel data preprocessing method for boosting neural network performance: A case study in osteoporosis prediction", Information Sciences, vol. 380, pp. 92-100, 2017.

[32] E. Rahm and H. H. Do, "Data cleaning: Problems and current approaches", IEEE Data Eng. Bull., vol. 23, no. 4, pp. 3-13, 2000.

[33] I. Badache, "Exploring Differences in the Impact of Users' Traces on Arabic and English Facebook Search," 2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI), 2019, pp. 225-232.