DATA MINING TECHNIQUES FOR EFFECTIVE DETECTION OF DISTRIBUTED DENIAL-OF-SERVICE ATTACKS

LEE YUEN HUI

A project report submitted in partial fulfilment of the requirements for the award of Master of Information Systems

Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman

December 2023

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature : LeeYH

Name : Lee Yuen Hui

ID No. : 21UEM00893

Date : 4/12/2023

APPROVAL FOR SUBMISSION

I certify that this project report entitled **"DATA MINING TECHNIQUES FOR EFFECTIVE DETECTION OF DISTRIBUTED DENIAL-OF-SERVICE ATTACKS"** was prepared by **LEE YUEN HUI** has met the required standard for submission in partial fulfilment of the requirements for the award of Master of Information Systems at Universiti Tunku Abdul Rahman.

Approved by,

Signature : kckhor

Supervisor : Khor Kok Chin

Date : 4/12/2023

Signature : ____________

Co-Supervisor : ____________

Date : ____________

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my research supervisor, Dr. Khor Kok Chin for his invaluable advice, guidance and his enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents and friends who had helped and given me encouragement to make it through completing the project on a timely manner.

ABSTRACT

A study on using data mining techniques on classification of Distributed Denial-of-Service (DDoS) attacks is carried out by first performing preliminary classification of DDoS attacks using five (5) selected classifiers available on the Waikato Environment for Knowledge Analysis (WEKA), namely Naive Bayes, J48, Random Forest, JRip and K-Nearest Neighbour (KNN/IBk), among which, the J48 Classifier was selected to further test different values of confidence factor (C) and minimum number of objects per leaf (M) parameters of the J48 Classifier to observe the results obtained from classification on a sampled data set created from the Consolidated DDoS Data Set (created from both the CICIDS2017 and the CIC-DDoS2019 data sets). Two types of classification (and optimisation via testing different values of C and M in both the Experimenter and the Explorer module in WEKA) were performed, preliminary ungrouped classification and simplification of classification via hierarchical grouped classification (with the hierarchy being defined by Sharafalddin et. al., originally made for the CIC-DDoS2019 data set and grouping from the top three (3) levels of the hierarchy). The first grouping (Level 0 Grouped Classification) involves reducing the classification from multi-class classification to bi-class classification between Normal/BENIGN and DDoS attack instances. In Level 1 Grouped Classification, DDoS attacks are grouped based on whether they are Exploitation, Reflection or HTTP/WebDDoS attacks, while in Level 2 Grouped Classification, DDoS attack labels are grouped into TCP (Reflection), TCP (Exploitation), UDP (Reflection), UDP (Exploitation), TCP/UDP (Reflection) and WebDDoS (all while BENIGN instances are relabelled Normal). Evidently, Level 1 Grouped Classification emerged as the winner in terms of overall TPR and GMEAN, while being only second in terms of overall F-Measure to Level 2 Grouped Classification, and performed worse in terms of PREC and had the highest overall False Positive Rates (FPR) among all classifications done. While preliminary ungrouped classification does highlight the problems of unbalanced data sets with only marginal changes in True Positive Rates (TPR) for individual DDoS attack labels for different values of C and M tested (with the highest increase being TPR for SSDP attacks increasing from 2.0% at C = 0.25 to 4.2% at C = 0.5), hierarchical grouped classification, while shows marginal increase in overall TPR for DDoS attacks, still show errors in classifying certain DDoS attacks like Portmap, SSDP, UDPLag, DNS and LDAP, as other DDoS attack types (especially true in Level 1 and 2 Grouped Classification, where the errors are

predominantly between separate DDoS attack groups), while potentially resulting in oversimplification of classifying DDoS attacks (especially true for Level 0 and 1 Grouped Classification), as grouping DDoS attacks this way increases overall TPR of classification by including DDoS attacks classified as other DDoS attacks into the calculation of TPR.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS / ABBREVIATIONS

N	number of instances
NetChange(x, C)	net change for evaluation metric x as compared to C = 0.25
NetChange(x, M)	net change for evaluation metric x as compared to M = 2 (using whichever value of C from testing values of C yielded most positive and/or least negative Net Change for classification done)
log(N)	base-10 logarithm of number of instances
P(A)	probability of A
P(B)	probability of B
P(A | B)	probability of A, given B
P(B | A)	probability of B, given A

ACC	Accuracy
ADASYN	Adaptive Synthetic Sampling Method
AI	Artificial Intelligence
AIMM	Artificial Intelligence Merged Methods
ANN	Artificial Neural Network
AUC	Area Under ROC Curve
C	Confidence Factor (WEKA J48 Classifier Parameter)
CART	Classification and Regression Trees
CoNN	Condensed Nearest Neighbour
DDoS	Distributed Denial-of-Service
DL	Description Length
DoS	Denial-of-Service
FL	Federated Learning
FLAD	Adaptive Federated Learning Approach to DDoS attack detection
FN	False Negative
FNR	False Negative Rate
FP	False Positive
FPR	False Positive Rate
F1	F1-Measure
GBM	Gradient Boosting
GEV	Generalised Extreme Value

GNB	Gaussian Naïve Bayes
GRU	Gated Recurrent Unit
GUI	Graphical User Interphase
HAIDS	Hybrid Anomaly Intrusion Detection System
HTTP	Hypertext Transfer Protocol
IDS	Intrusion Detection System
IoT	Internet of Things
IR	Imbalance Ratio
KNN	K-Nearest Neighbours
LBR	Lazy Bayesian Rules
LDDM	Low-rate DDoS Detection Method
LR	Logistic Regression
LSTM	Long Short-Term Memory
M	Minimum Number of Objects (WEKA J48 Classifier Parameter)
MLP	Multilayer Perception
NB	Naïve Bayes
NN	Neural Network
NN-MLP	Neural Network-Multilayer Perception
OS	Operating System
PREC	Precision
QoS	Quality of Service
REP	Reduced Error Pruning
RF	Random Forest
RIPPER	Repeated Incremental Pruning to Produce Error Reduction
RNN	Recurring Neural Network
ROC	Receiver Operating Characteristics
ROS	Random Over Sampling
RUS	Random Under Sampling
SMOTE	Synthetic Minority Over-sampling Technique
SVM	Support Vector Machine
TCP	Transmission Control Protocol
TL	Tomek Links
TN	True Negative
TNR	True Negative Rate

TP	True Positive
TPR	True Positive Rate
UDP	User Datagram Protocol
WEKA	Waikato Environment for Knowledge Analysis
1R	OneR/One Rule

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 General Introduction

Distributed Denial-of-Service (DDoS) attacks are harmonious attacks towards a server via a huge number of affected servers. This starts with hackers finding an opening in the firewall to install malicious code into the servers to access the servers remotely, in which these servers are used to attack linked machines. By getting access to a large enough number of servers, hackers can compromise an entire Web server. DDoS attacks usually target routers, links, firewalls and defence systems, network infrastructure, operating systems (OS), communications and applications. (Bhattacharyya & Kalita, 2016) As compared to their older counterpart, Denial-of-Service (DoS) attacks, DDoS attacks are much more sophisticated, and are more catastrophic in terms of the effects caused, not to mention that DDoS attacks are evolving constantly. DDoS involves the use of multiple infected computers/servers (known as botnets) to overload traffic to a server, causing the impacted server to unable to deal with legitimate (BENIGN) requests. (Kassim, 2015)

A DDoS attack happens in four stages. DDoS attacks start with the hacker choosing the machines/servers (known as agents) to launch the DDoS attack, followed by finding loopholes in the security system of the machine to compromise the security systems of agents, all while preventing the malware code from being detected by the internal security system of the agent, followed by communicating with “handlers” to check the status of all affected agents, via protocols like ICMP, TCP and UDP, to determine which agents are ready to launch the attack. The final attack is launched to initiate the DDoS attack using existing bandwidth, overwhelming the target host or network with a huge traffic of requests. (Bhattacharyya & Kalita, 2016)

The existence and evolution of DDoS attacks mandates the need for better systems to detect DDoS attacks. While detection of DDoS attacks has been long studied in the field of cybersecurity, some rare DDoS attacks are not detectable using current classification algorithms. Although the existence of a generic DDoS defence system that can detect all DDoS attacks without affecting the Quality of Service (QoS) is logically impossible, the effectiveness of utilising statistical methods for anomaly detection still proves itself to be useful in terms of differentiating DDOS attacks from legitimate (or benign) user behaviour. (Bhattacharyya & Kalita, 2016)

1.2 Importance of the Study

The class unbalance problem is a frequently encountered problem in not just detecting DDoS attacks, but in multiple fields that involve machine learning and data mining (fraud detection, medical diagnosis, and oil spill detection, just to name a few), due to various reasons, the collection of data for the desired class is often restricted. (Sun et al., 2009)

While countless machine learning algorithms are very good at detecting DDoS attacks, boasting over 90% accuracy, the problem with current methods lies in the training data used, as DDoS attacks come in many different types and do not all exist equally as often. In fact, in currently used data sets, rare types of DDoS attacks comprise a very small portion (less than 0.1%) of all the DDoS attacks in the data set. This greatly affects the data quality, as this is a ‘balancing problem’ that needs to be rectified to promote the principle of data diversity. (Bolodurina et al., 2020)

This is especially true when frequently used classification algorithms are not capable enough to handle data sets of unbalanced classes, as the utilization of unbalanced data sets as training data for DDoS attack systems leads to inaccuracies and biases in the algorithms used to detect DDoS attacks. (Sun et al., 2009; Merino et al., 2019)

1.3 Problem Statement

DDoS Attacks are common threats on the Internet nowadays, even in situations least expected. This is evident with DDoS attacks happening between Russia and Ukraine in the ongoing Russian invasion of Ukraine, when in February 2022, Ukrainian government and banking websites were ambushed with the biggest DDoS attack in Ukrainian history. (Microsoft Security, 2023) Hackers, both pro-Russian and pro-Ukrainian, are stealing and leaking information to create havoc, pushing the war from the physical battlefield to the digital world. As compared to the same period in 2021, the number of threatening DDoS attacks globally is 203% higher in the 1st half of 2022. (Lohrmann, 2022) Even in Malaysia, DDoS attacks are not just getting more frequent, but also longer, with the average duration of DDoS attacks approaching 3000 minutes, a 10000% increase from the previous year. (Bernama News, 2022)

Data sets formed using network traffic are usually unbalanced, particularly the attacks. One of the common attacks is DDoS. Rare DDoS Attacks are common in these data sets, and they cause low detection rate using classifications algorithms. Attacks are small in data sets, detections are low. In the CICIDS2017 database from the Canadian Cybersecurity Institute, for example, out of a total of 2 830 743 instances, BENIGN instances make up a bit over 80% of all instances (2 273 097 instances) in the data set. Within the remaining 557 646 attack instances, the top 3 most common instances of DDoS attacks (Dos Hulk, PortScan and DDoS) combined, makes up 92.90% of all attack instances (combined total of 518 030 instances), as compared to the 11 least common instances (namely Dos Golden Eye, FTP-Patator, SSH-Patator, Dos slowloris, Dos Slowhttptest, Bot, Web Attack-Brute Force, Web Attack-XSS, Infiltration, Web Attack-SQL Injection and Heartbleed) combined (combined total of 39 616 instances, or 7.10% of all attack instances), making this data set have a ratio of majority instances like Dos Hulk, PortScan and DDoS, to minority instances like Infiltration, Web Attack-SQL Injection and Heartbleed, of around 13 to 1. (Ho et al., 2021) We can clearly see that the CICIDS2017 data set is not only staggeringly imbalanced between BENIGN instances and actual DDoS attacks, but even within the actual attack instances, a select few DDoS Attack Labels dominate all other attack labels in the data set.

Although DDoS attacks such as Infiltration, Web Attack-SQL Injection and Heartbleed are rare, they cannot be ignored as their effects are just as significant (if not more so) as more common DDoS attacks. Nevertheless, using such data sets is problematic, as more often, Intrusion Detection Systems (IDS) will often maximize accuracy at the cost of detecting rare attacks. (Ho et al., 2021) Especially in the world of Big Data and Internet of Things (IoT), where more focus is put on the disproportionality of such attacks for better detection of DDoS attacks. (Krawczyk, 2016)

Usually, using only 1 classification model to detect all DDoS attacks is insufficient. High classification rates can be obtained if the model is trained with large amount of DDoS attack classes. As we have seen, certain DDoS attack classes are rare. Thus, the model is not well-learned of these rare classes, thus perform weakly in detecting them.

1.4 Aims and Objectives

This project aims to perform reduction of a selected data set to increase detection rate of popular classification algorithms on DDoS attacks. The project has four objectives:

(i) To identify a suitable unbalanced DDoS data set for this project.

(ii) To generate a reduced data set by random under-sampling the identified data set, considering the limited computing resources.

(iii) To compare the preliminary performance of the classification algorithms with default parameter values on the reduced data set and pick the best performing classifier.

(iv) To optimise the performance of the selected classifier on the reduced data set with hierarchical grouping of the attack types for balancing the class distribution.

1.5 Scope of the Study

The study starts with identifying a suitable data set to identify DDoS attacks (particularly rare ones), from sources such as the Canadian Institute of Cybersecurity (where the CICIDS databases originate), then using the selected data set, preliminary classification is first carried out, to assess the preliminary performance of classification algorithms on a sampled data set from the selected data set, especially on rare DDoS attack types, using Naïve Bayes, Random Forest, J48, JRip and K-Nearest Neighbours (KNN). Based on the preliminary classification results, the best performing classifier, J48/C4.5 was then selected to continue with, in which changing values of the confidence factor (C) and minimum number of objects in a leaf (M), followed by hierarchical grouping of DDoS attacks are used to increase the effectiveness of the selected classifier in detecting rare DDoS attacks using selected evaluation metrics that will be explored in Chapter 2.

1.6 Contribution of the Study

Currently, as only a small number of methods can effectively handle an unequally balanced traffic of DDoS attacks, the study will contribute slightly to the effective detection of DDoS attacks, especially rare kinds of DDoS attacks. (Li et al., 2022) Moreover, methods such as grouping similar attacks and splitting the data set based on formed groups to classify separately, can also be proven useful in terms of detecting DDoS attacks that are similar in nature, especially if a pre-existing hierarchy of DDoS attacks can be made use to help classify rare DDoS attacks better and improve detection rate for these attack types. The utilisation of data-level methods to deal with unbalanced data sets can give insights on how to increase detection rates of DDoS attacks and how to better use machine learning algorithms to better classify minority classes.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The problem of unbalanced data sets is itself not a recent problem. Currently, there is a lack of a comprehensive way to detect all forms of Distributed Denial-of-Service (DDoS) attacks, as current methods mostly require forming an attack profile, making it challenging for them to detect new types of attacks, not to mention that the algorithms used to detect DDoS attacks are usually tailored to suit the network it works on, making them incapable of detecting DDoS attacks on other networks, or even new attacks on their networks that they previously never encountered. (Bolodurina et al., 2020) Nevertheless, this has not stopped the research into finding ways to detect DDoS attacks more effectively.

2.2 DDoS/DoS Overview

DDoS attacks are a type of Denial-of-Service (DoS) attack, which involve a hacker denying a web service to users. (Cybersecurity & Infrastructure Security Agency (CISA), 2021) DDoS and DoS attacks operate by introducing spam traffic to disrupt e-mail and web services, typically by overwhelming the servers with more requests than the servers can handle. (Kassim, 2015; Microsoft, 2023; Bhattacharyya & Kalita, 2016) As compared to traditional DoS attacks, which only involves a single machine/server attacking a target machine/server, DDoS attacks involve a network of machines coordinated to attack a single target machine/server. (Cybersecurity & Infrastructure Security Agency (CISA), 2021) DDoS attacks can either attack the network, by taking up the network bandwidth, slowing down the network greatly, or the application layer, which overwhelms the server with requests, causing the server to fail. (Kassim, 2015)

2.3 Overview of Popular Intrusion Detection Data Sets

While numerous Intrusion Detection Data Sets are available to perform classification, only a handful three are of consideration to use for the project. The three data sets that are considered for the study are the Consolidated DDoS, UNSW Bot-Iot and the Boğaziçi University Distributed Denial of Service (BOUN DDoS) Data sets.

2.3.1 Consolidated DDoS Data Set from CICIDDoS2017 and CICIDDoS2019

The Consolidated DDoS Data Set is a combination of the CICIDDoS2017 and the CICIDDoS2019 Data Sets from the Canadian Cybersecurity Institute and the University of Brunswick, created by Enoch and Khor (2021), as a more ready-made data set consisting of DDoS attacks and BENIGN labels from both the CICIDDoS2017 and the CICIDDoS2019 Data Sets. The data set was pre-processed as specified:

(i) All labels of DDoS attacks standardised.
(ii) Removal of infinity values.
(iii) Introduced BENIGN labels from CICIDDoS2017 Data Set to balance out the large number of DDoS attacks in CICIDDoS2019 Data Set.

(Enoch & Khor, 2021)

The data set consists of a total of 9 209 309 instances (excluding 1974 incomplete data rows), consisting of 2 384 051 BENIGN instances and 6 825 258 DDoS attack instances, and 78 columns, which are listed as followed:

1. Flow Duration
2. Total Fwd Packets
3. Total Backward Packets
4. Total Length of Fwd Packets
5. Total Length of Bwd Packets
6. Fwd Packet Length Max
7. Fwd Packet Length Min
8. Fwd Packet Length Mean
9. Fwd Packet Length Std
10. Bwd Packet Length Max
11. Bwd Packet Length Min
12. Bwd Packet Length Mean
13. Bwd Packet Length Std
14. Flow Bytes/s
15. Flow Packets/s
16. Flow IAT Mean
17. Flow IAT Std
18. Flow IAT Max
19. Flow IAT Min
20. Fwd IAT Total
21. Fwd IAT Mean
22. Fwd IAT Std
23. Fwd IAT Max
24. Fwd IAT Min
25. Bwd IAT Total
26. Bwd IAT Mean

27. Bwd IAT Std
28. Bwd IAT Max
29. Bwd IAT Min
30. Fwd PSH Flags
31. Bwd PSH Flags
32. Fwd URG Flags
33. Bwd URG Flags
34. Fwd Header Length
35. Bwd Header Length
36. Fwd Packets/s
37. Bwd Packets/s
38. Min Packet Length
39. Max Packet Length
40. Packet Length Mean
41. Packet Length Std
42. Packet Length Variance
43. FIN Flag Count
44. SYN Flag Count
45. RST Flag Count
46. PSH Flag Count
47. ACK Flag Count
48. URG Flag Count
49. CWE Flag Count
50. ECE Flag Count
51. Down/Up Ratio
52. Average Packet Size
53. Avg Fwd Segment Size
54. Avg Bwd Segment Size
55. Fwd Header Length.1
56. Fwd Avg Bytes/Bulk
57. Fwd Avg Packets/Bulk
58. Fwd Avg Bulk Rate
59. Bwd Avg Bytes/Bulk
60. Bwd Avg Packets/Bulk
61. Bwd Avg Bulk Rate
62. Subflow Fwd Packets
63. Subflow Fwd Bytes
64. Subflow Bwd Packets
65. Subflow Bwd Bytes
66. Init_Win_bytes_forward
67. Init_Win_bytes_backward
68. act_data_pkt_fwd
69. min_seg_size_forward
70. Active Mean
71. Active Std
72. Active Max
73. Active Min
74. Idle Mean
75. Idle Std
76. Idle Max
77. Idle Min
78. Label

The class distribution of labels (column 78 of data set) in the resulting Consolidated DDoS Data Set, by decreasing number of instances, N, is as shown in Table 2.1.

Table 2.1: Class Distribution of Consolidated DDoS Data Set

Normal/Attack Label	**Number of Instances, N**	**% of Total Instances**
BENIGN	2 384 051	25.887
TFTP	1 951 336	21.189
MSSQL	998 191	10.839
NetBIOS	747 772	8.120
UDP	688 393	7.475

Table 2.1 (Continued)

SYN	594 129	6.451
SNMP	514 957	5.592
DNS	490 813	5.330
LDAP	410 301	4.455
SSDP	256 832	2.789
NTP	119 528	1.298
UDPLag	34 891	0.379
Portmap	17 676	0.192
WebDDoS	439	0.005
Total	**9 209 309**	**100.000**

(Enoch & Khor, 2021)

The class distribution of the labels in the Consolidated DDoS Data Set is plotted as shown in Figure 2.1.

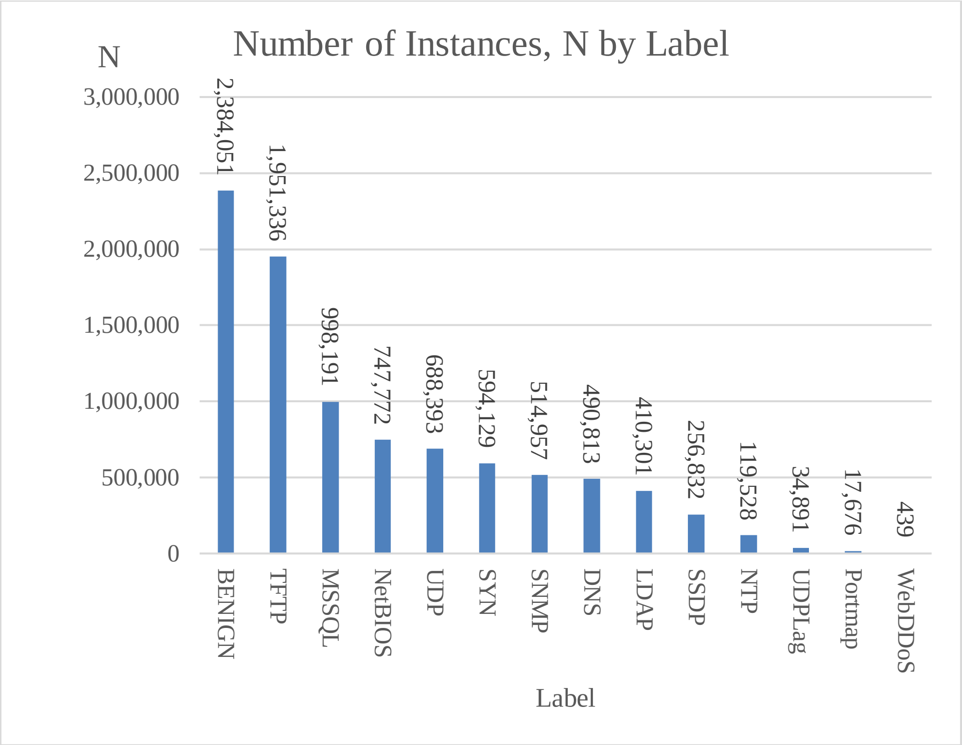


Figure 2.1: Class distribution of Consolidated DDoS Data Set

From Table 2.1 and Figure 2.1, while TFTP attack labels are relatively well balanced to the BENIGN instances, all the other attack labels are disproportionately represented, even more severely can be seen with attack labels like UDPLag, Portmap and WebDDoS, each and combined, with less than 100 000 instances (with WebDDoS having only 439 instances, barely even showing up in the chart). Although combined, the total number of attack instances is 6 825 258, around 2.86 times the number of BENIGN instances, TFTP make up 28.59% of all attack instances, followed by still massively many instances like MSSQL, NetBIOS and UDP, while the three least common attack labels, UDPLag, Portmap and WebDDoS, combined make up 34 891 total instances, which is 0.777% of all attack labels (0.576% of the entire data set). As a comparison, the number of TFTP attack labels is 2 384 051, which is over 36 times this number. To even show the least common attack labels properly, the graph is plotted in logscale as shown in Figure 2.2.

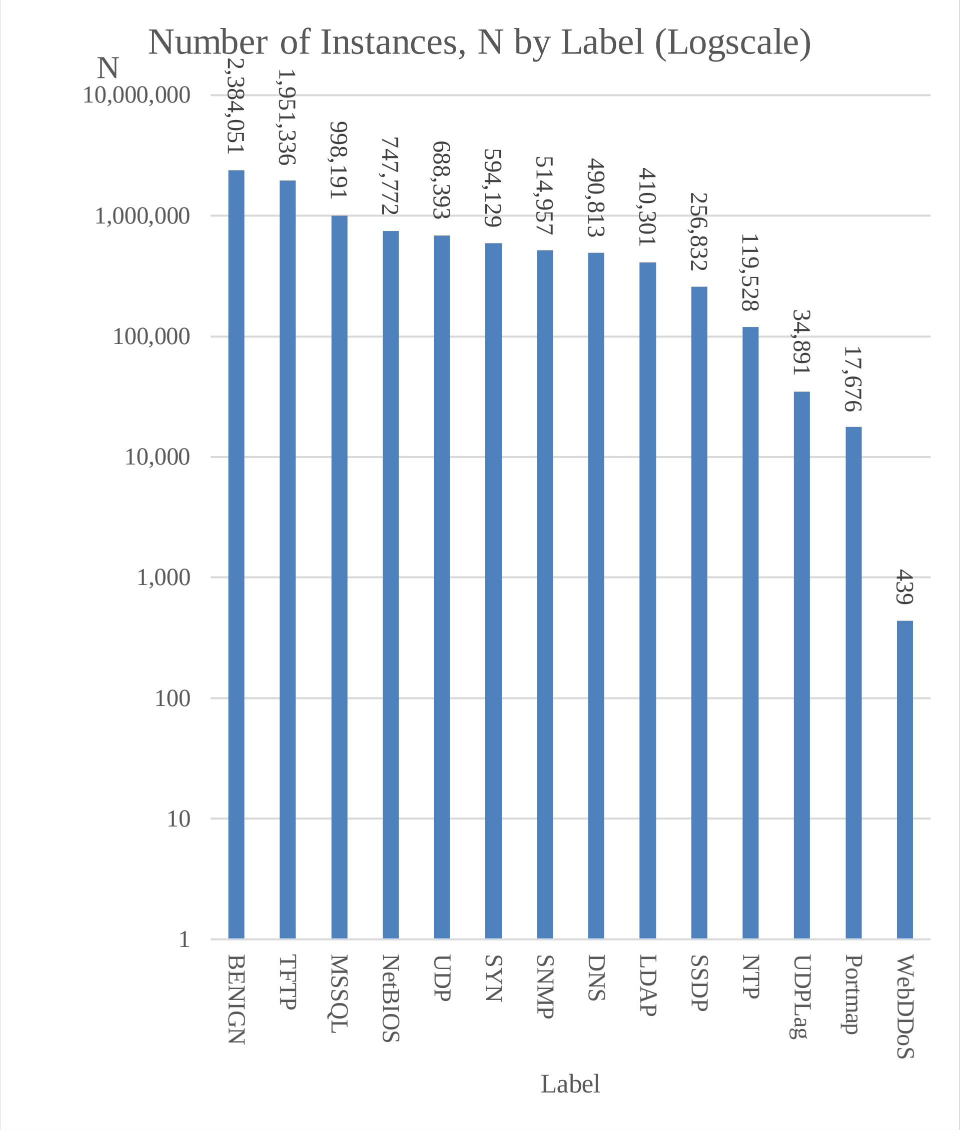


Figure 2.2: Class distribution of Consolidated DDoS Data Set (Logscale)

From Figure 2.2, the number of BENIGN instances and attack labels like TFTP and MSSQL, each are over three orders of magnitude larger than that of WebDDoS attack labels, highlighting the severity of imbalance of attack instances in the Consolidated DDoS Data Set.

2.3.2 UNSW Bot-Iot Data Set

The UNSW Bot-Iot Data Set is a data set developed by Koroniotis et al. (2018), at the University of New South Wales (UNSW), Australia. (Peterson et al., 2021; Koroniotis et al., 2019) The data set is built on simulating Botnet attacks on multiple IoT infrastructure, which comprises of DoS, DDoS, Reconnaissance and Information Theft attack instances, plus a handful bunch of Normal/BENIGN instances. The data set is built to address the lack of a realistic data set for IoT cybersecurity research, in which makes it a problem as data sets then do not accurately depict real world cyber-attacks, nor do they cover a wide variety of botnet scenarios, not to mention problems such as traffic redundancy and missing ground truth. By creating the UNSW Bot-Iot Data Set, Koroniotis et al. (2018) contributed to IoT cybersecurity research as stated:

> (i) Developed a realistic data set, complete with comprehensive instructions to design the testbed configuration and simulated IoT infrastructure.
>
> (ii) Performed statistical analysis of suggested attributes in the developed data set using Correlation Coefficient and Joint Entropy methods
>
> (iii) Assessed the effectiveness of machine learning and deep learning techniques on the developed data set and compared the performance with other data sets.
>
> (Koroniotis et al., 2019)

As the study is specific to DDoS attacks, only the DDoS attacks section, together with the Normal/BENIGN instances of the UNSW Bot-Iot Data Set is considered for the study. The class distribution of the UNSW Bot-Iot Data Set (for the DDoS attacks, together with the Normal/BENIGN instances) is displayed in Table 2.2.

Table 2.2: Class Distribution of the UNSW Bot-Iot Data Set (DDoS Attacks and Normal/BENIGN traffic)

DDoS Attack Subcategory	**Number of Instances**
Normal/BENIGN	9543
TCP	19 547 603
UDP	18 965 106
HTTP	19 771
Total	**38 542 023**

(Peterson, et al., 2021; Koroniotis, et al., 2019)

From Table 2.3, the number of instances of TCP and UDP subcategory of DDoS attacks is disproportionally larger than that of HTTP subcategory, with each of TCP and UDP having about 1000 times the number of instances of HTTP subcategory DDoS attacks. Another issue to take note of is that Normal/BENIGN traffic is severely underrepresented in the UNSW Bot-Iot Data Set.

2.3.3 BOUN DDoS Data Set

The BOUN DDoS Data Set is a data set created by Erhan and Anarim (2020) at Boğaziçi University, Istanbul, Turkey. The data set comprises of resource depletion-type DDoS attacks that are separated into the TCP SYN and UDP Flood data sets, each with 4 attack periods that contain both legitimate/BENIGN and attack packets. The data sets contain attacks of different densities to aid in training and evaluation of Intrusion Detection Systems (IDS) to allow a broader understanding of resource depletion-type DDoS attacks to aid in the development and evaluation of network-based attack techniques. (Erhan & Anarim, 2020) The information of the attack instances in the BOUN DDoS data sets are shown in Table 2.3.

Table 2.3: Information of attack instances in BOUN DDoS Data Set

Data Set	**TCN SYN**		**UDP**	
Attack Period	**Attack Packets**	**Legitimate Packets**	**Attack Packets**	**Legitimate Packets**
1	19 035	370 746	37 216	268 882
2	27 121	428 168	55 029	337 036
3	35 936	352 296	75 023	393 450
4	43 463	401 553	93 378	404 330
Total	**125 557**	**1 552 763**	**260 646**	**1 403 698**

(Erhan & Anarim, 2020)

From Table 2.4, in both the TCN SYN and UDP Flood attack data sets of the BOUN DDoS Data Set, Legitimate/BENIGN packets outnumber attack packets (12.367 to 1 in TCN SYN data set and 5.385 to 1 in UDP data set) in the data sets. In total, there are 386 203 attack packets and 2 956 461 legitimate/BENIGN packets in the BOUN DDoS Data Set, making the ratio of legitimate to attack packets, 7.655 to 1 for the whole data set.

2.3.4 Selection of Data Set

From investigating the data sets chosen for the study, the pros and cons for each data set investigated are displayed in Table 2.4.

Table 2.4: Table of Advantages and Disadvantages for using each investigated Data Set

Data Set	Advantages	Disadvantages
Consolidated DDoS Data Set	• A ready-made consolidation of CICIDDoS2017 and CICIDDoS2019 data sets. • Have a wide variety of DDoS attack types, excellent for majority vs	• File size of 3.2 GB is quite large, and difficult to be processed by WEKA.

Table 2.4 (Continued)

	minority attack classification. - Large number of Normal/BENIGN instances allow for performing under sampling of Normal/BENIGN instances.	
UNSW Bot-Iot	- Readily able to perform minority (DDoS HTTP) vs Normal and majority (DDoS TCP and DDoS UDP) vs Normal classifications separately.	- Files are disorganised, do not have all DDoS vs Normal file on its own, have to concatenate files manually. - Normal/BENIGN instances are severely underrepresented, not suitable to perform under-sampling of Normal/BENIGN instances on data set. - Each of DDoS_TCP and DDoS_UDP files are also huge (4.39 and 4.33 GB respectively), so concatenating files result in even larger file size (nearly 9 GB), which is also hard to be processed by WEKA.

Table 2.4 (Continued)

BOUN DDoS	- Combined file size of 1.71 GB is more manageable for WEKA to handle. - Large number of legitimate packets make it good for performing under sampling of legitimate packages.	- Does not differentiate types of DDoS attacks, can’t do majority vs minority attack classification.

Considering the pros and cons of using each data set investigated, it seems the Consolidated Data Set is most suitable for the study, albeit the big file size, although compared to the UNSW Bot-Iot Data Set, is still smaller in terms of total file size. This is especially because the other two data sets investigated are simply not suitable to use for the study. However, sampling must be carried out so that WEKA to be able to handle the Consolidated DDoS Data Set.

2.4 Unbalanced Characteristics of Data Sets

Unbalanced classes are a type of distribution-based data irregularities that occur in data sets. (Das et al., 2018) They exist when one class of data (known as the majority class) exists far more frequently than other classes (also known as minority classes). Unbalanced data sets cause a handful of problems in the field of data mining and classification, in which will be investigated in the following sections.

2.4.1 Problems Caused by Unbalanced Data Sets

Often is the case where the classifier algorithm will output a solution that differs from the ideal solution, or results in lower accuracy. To demonstrate, a simple classification of 30 circles and crosses are done, with 4 different data sets of 30 circles and crosses are used to demonstrate the effects of imbalance ratio and overlapping on the effectiveness of classification via accuracy. Starting off with a data set of equal number of circles and crosses (15 circles and 15 crosses), varying by arbitrary X and Y coordinates, distributed as shown in Figure 2.3, without any overlapping.

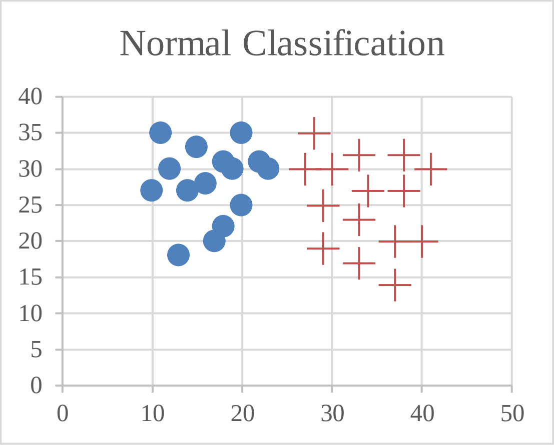
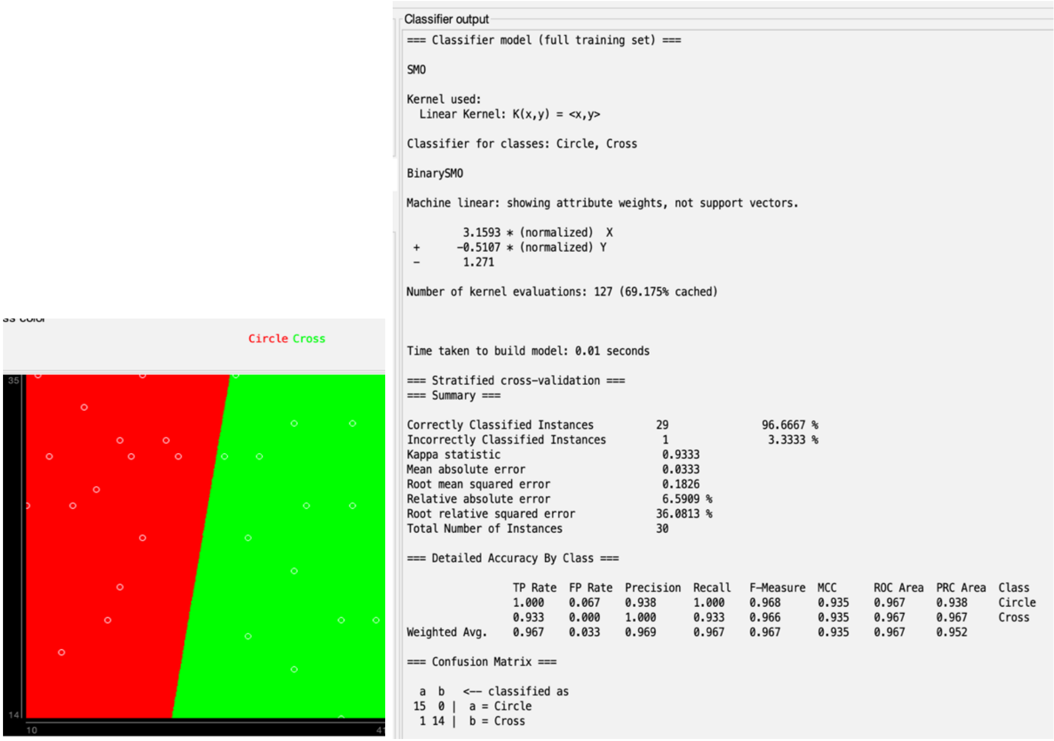


Figure 2.3: Distribution of 15 circles and 15 crosses by X and Y position.

Using the SMO classifier in WEKA, the boundary detected, with the classification results are as shown in Figure 2.8.



```
=== Classifier model (full training set) ===

SMO

Kernel used:
  Linear Kernel: K(x,y) = <x,y>

Classifier for classes: Circle, Cross

BinarySMO

Machine linear: showing attribute weights, not support vectors.

         3.1593 * (normalized)  X
 +      -0.5107 * (normalized) Y
 -       1.271

Number of kernel evaluations: 127 (69.175% cached)



Time taken to build model: 0.01 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          29               96.6667 %
Incorrectly Classified Instances         1                3.3333 %
Kappa statistic                          0.9333
Mean absolute error                      0.0333
Root mean squared error                  0.1826
Relative absolute error                  6.5909 %
Root relative squared error             36.0813 %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.067    0.938      1.000    0.968      0.935    0.967     0.938     Circle
                 0.933    0.000    1.000      0.933    0.966      0.935    0.967     0.967     Cross
Weighted Avg.    0.967    0.033    0.969      0.967    0.967      0.935    0.967     0.952

=== Confusion Matrix ===

  a  b   <-- classified as
 15  0 |  a = Circle
  1 14 |  b = Cross
```

Figure 2.4: Boundary detected by SMO classifier in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for normally balanced data set.

From Figure 2.4, using a well-balanced and non-overlapping data set, the algorithm can easily classify both classes, with a very good accuracy of 96.67%. However, data sets in practice are hardly equally distributed and non-overlapping, as will be the case due to two factors, namely Imbalance Ratio (IR), overlapping classes and small disjuncts, as will be seen in the following sections.

2.4.1.1 Imbalance Ratio (IR)

The ratio of majority to minority class instances is known as the Imbalance Ratio (IR). Generally, the overwhelming presence of majority classes (especially in cases of high IR), will affect the ability of classifier algorithms to correctly classify minority classes, due to high error rate from error minimisation done by classification algorithms. (Das et al., 2018) Classifier algorithms run on the assumption that data sets are relatively balanced, even though this is hardly the case in practice, as certain classes are rare in data sets as they are also rare in nature, classification rules that predict these rare classes are often hardly considered by the classification algorithm due to its rarity or is undetected by the algorithm. (Sun et al., 2009) This is especially true with classifiers like Logistic Regression (LR), Support Vector Machines (SVM) and Decision Trees (DT), which runs well with well-balanced data sets, but when we use them on unbalanced data sets, they will run into problems trying to classify rare cases while more common classes are better classified. (Guo et al., 2017; López et al., 2013) This is evident with the fact that misclassification is often caused by the insufficient weight and information of the minority classes. (Song, et al., 2022) Often is the case that classification algorithms will favour majority classes for higher accuracy while rare classes are treated as random noise. (Guo et al., 2017; Loyola-González et al., 2016; Beyan & Fisher, 2015) This results in misclassification of rare classes, which is a problem as sometimes, correctly classifying these rare cases is beneficial. Using classification algorithms on unbalanced data sets results in poor generalisation of minority classes, which often leads to poor predictive accuracy. (Sun et al., 2009; Pozzolo et al., 2015) This problem is also known as the small/rare class problem. (Sun et al., 2009) This can be illustrated further using a fairly imbalanced distribution of 25 circles and 5 crosses (IR of 5:1), like as shown in Figure 2.5.

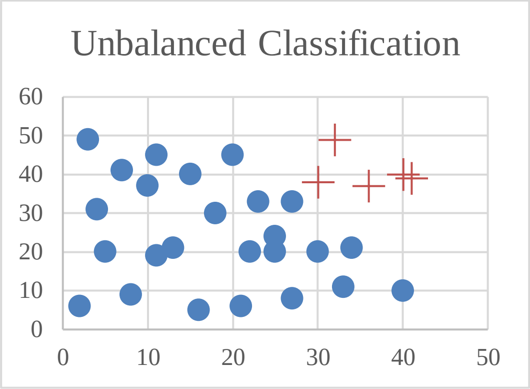
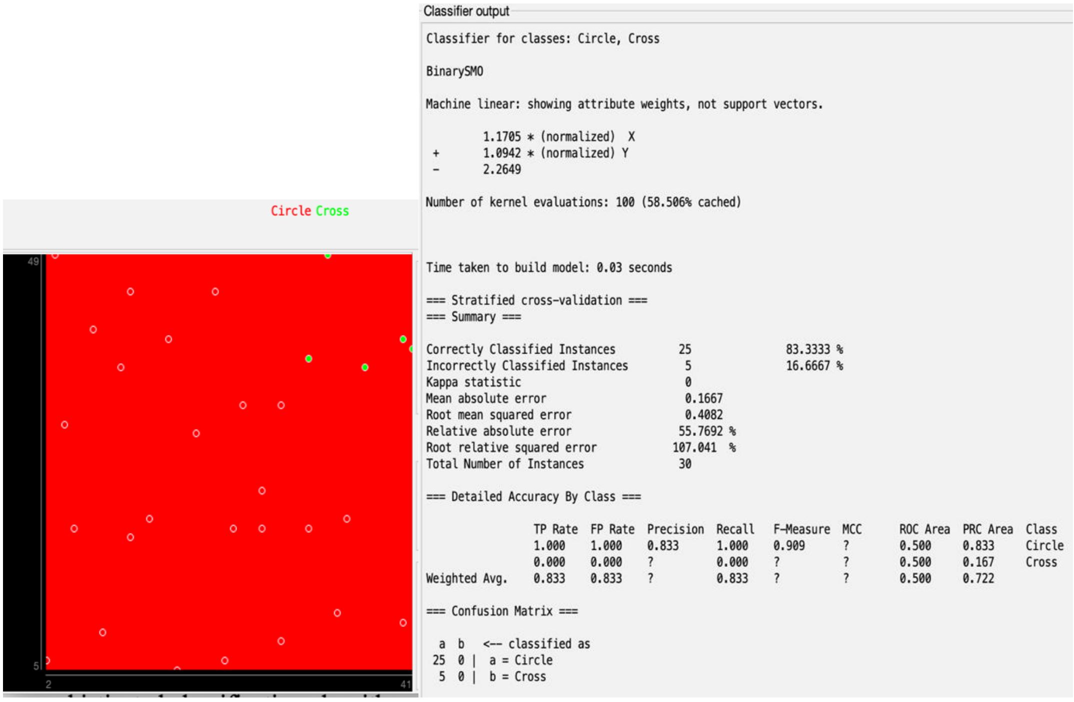


Figure 2.5: Distribution of unbalanced data set of 25 circles to 5 crosses, without overlapping.

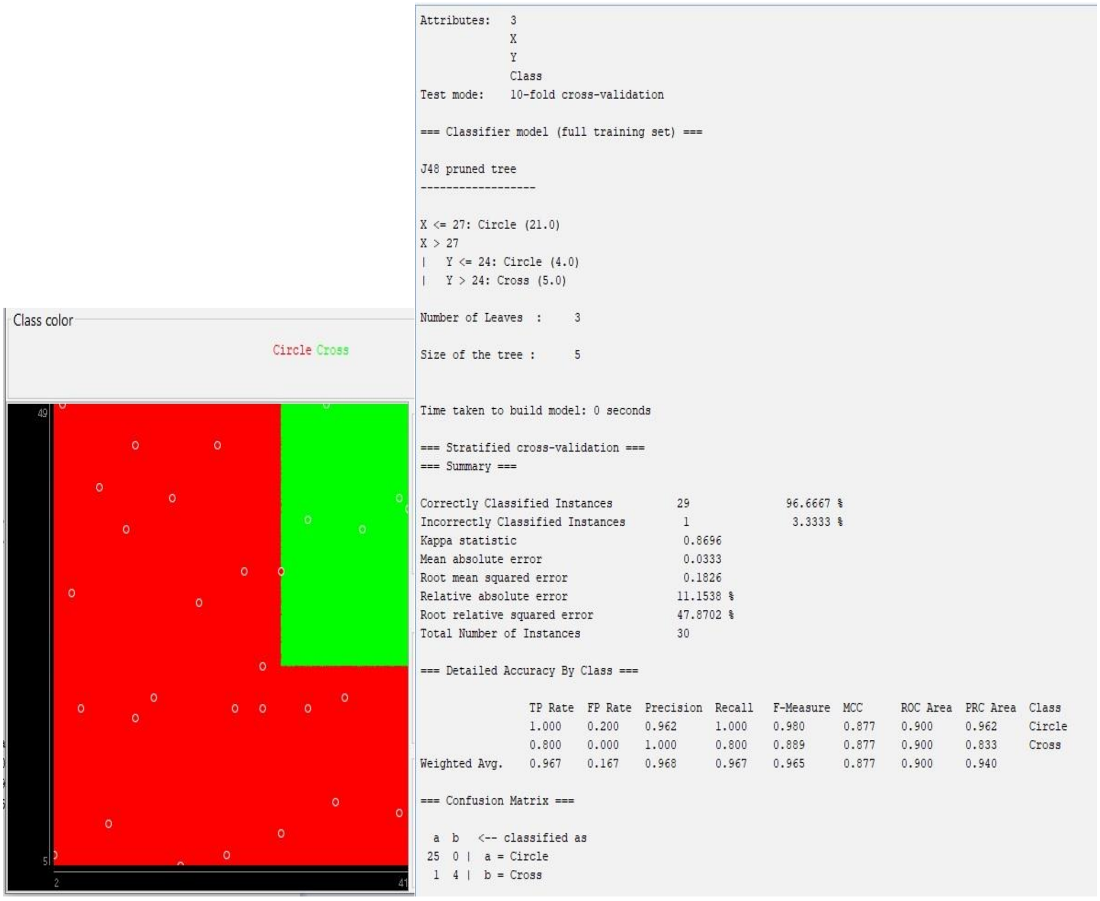
Using the SMO classifier again in WEKA, the detected boundary is showcased in Figure 2.6, together with the classification results.



```
Classifier for classes: Circle, Cross

BinarySMO

Machine linear: showing attribute weights, not support vectors.

         1.1705 * (normalized)  X
 +       1.0942 * (normalized) Y
 -       2.2649

Number of kernel evaluations: 100 (58.506% cached)



Time taken to build model: 0.03 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          25               83.3333 %
Incorrectly Classified Instances         5               16.6667 %
Kappa statistic                          0
Mean absolute error                      0.1667
Root mean squared error                  0.4082
Relative absolute error                 55.7692 %
Root relative squared error            107.041  %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    1.000    0.833      1.000    0.909      ?        0.500     0.833     Circle
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.167     Cross
Weighted Avg.    0.833    0.833    ?          0.833    ?          ?        0.500     0.722

=== Confusion Matrix ===

  a  b   <-- classified as
 25  0 |  a = Circle
  5  0 |  b = Cross
```

Figure 2.6: Boundary detected by SMO algorithm in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for unbalanced data set of 25 circles and 5 crosses (with IR of 5:1).

From Figure 2.6, it is seen that an unbalanced data set of 25 circles and 5 crosses results in lower accuracy as compared to the boundary detected by the same algorithm in Figure 2.4, keeping the total size of the data set (30 total instances) constant. Moreover, the SMO algorithm was unable to detect the instances of crosses, which is why there is a lack of green area (for crosses) in Figure 2.6. Normally, if the classification algorithm is sensitive enough, highly unbalanced data by itself is not really an issue, such as the case as when the J48 classifier is used instead of SMO, as shown in Figure 2.7.



```
Attributes:   3
              X
              Y
              Class
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

X <= 27: Circle (21.0)
X > 27
|   Y <= 24: Circle (4.0)
|   Y > 24: Cross (5.0)

Number of Leaves  :     3

Size of the tree :      5


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          29               96.6667 %
Incorrectly Classified Instances         1                3.3333 %
Kappa statistic                          0.8696
Mean absolute error                      0.0333
Root mean squared error                  0.1826
Relative absolute error                 11.1538 %
Root relative squared error             47.8702 %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.200    0.962      1.000    0.980      0.877    0.900     0.962     Circle
                 0.800    0.000    1.000      0.800    0.889      0.877    0.900     0.833     Cross
Weighted Avg.    0.967    0.167    0.968      0.967    0.965      0.877    0.900     0.940

=== Confusion Matrix ===

  a  b   <-- classified as
 25  0 |  a = Circle
  1  4 |  b = Cross
```

Figure 2.7: Boundary detected by J48 classifier in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for unbalanced data set of 25 circles and 5 crosses (with IR of 5:1), without overlapping data points.

As shown in Figure 2.7, despite the data imbalance, the J48 classifier can detect the instances of crosses and classify them as such, with very good accuracy of 96.67%, better than SMO did with the same data set, as the classes do not overlap with each other making classification effective. However, when class instances overlap with each other (or worse, break off into small disjuncts), this complicates the classification process further, which will be investigated in Section 2.4.1.2.

2.4.1.2 Overlapping Classes and Small Disjuncts

Another issue of unbalanced data sets is that minority classes tend to overlap with each other, especially if two or more minority classes have an equal likelihood of appearing in the data set. (Guo et al., 2017) Overlapping classes can cause over-regularisation of classes. Often is the case that overlapping classes, paired with high IR will cause several problems in classification, namely wrongly classifying minority class training instances, poor generalisation of minority classes and unfixed boundaries at varying levels of overlapping and IR ratio, considering the size of the data set and the classifier used to classify the instances. Even at low IR, high amount of overlapping can cause unstable learned boundaries. (Das et al., 2018) Pairing with overlapping classes is the small disjunct problem, which is the problem with a handful of data points being completely disjointed from the rest of the class. Small disjuncts in data are known to be diminished subcategories of classes. Although they make up an insignificant portion of data points, they can represent a large percentage of the data set, especially for rule-based classification algorithms. As rules for classifying these small disjuncts are often unconcise, this results in higher error rates for classifying them. (Das et al., 2018) To demonstrate, a data set of 15 circles and 15 crosses (again), is created, with some overlapping, is distributed as shown in Figure 2.8.

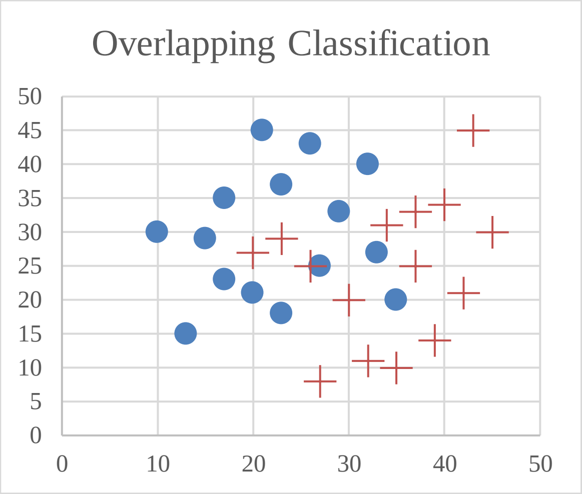
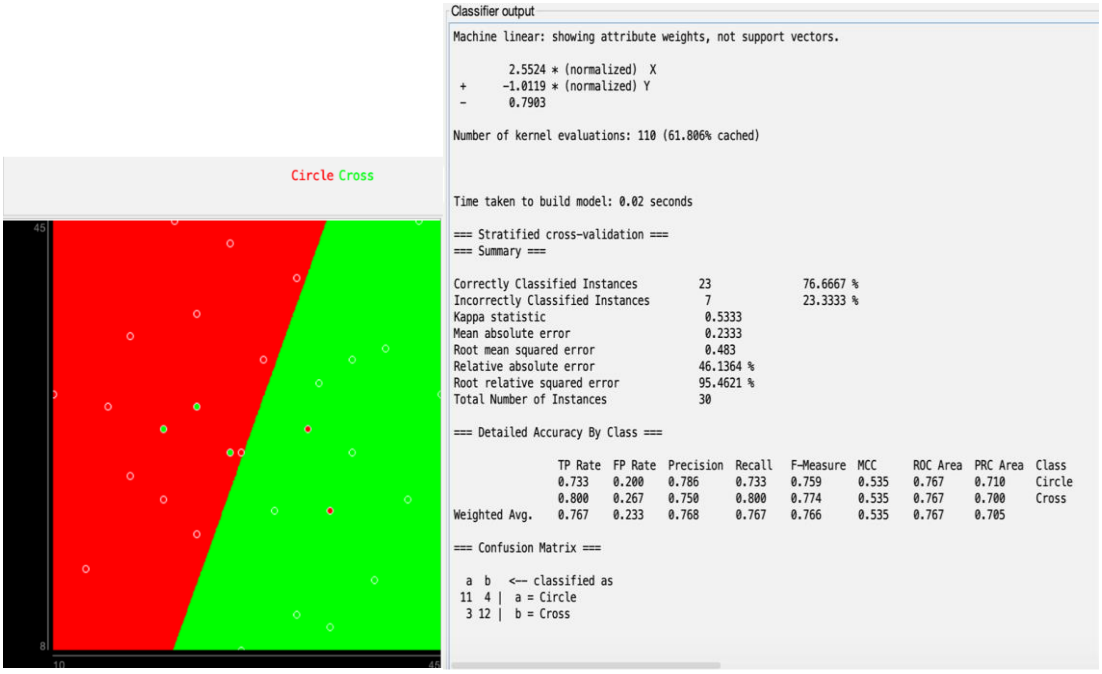


Figure 2.8: Distribution of 15 circles and 15 crosses, with some overlapping.

Using the SMO algorithm again in WEKA, the boundary detected, with the results are as shown in Figure 2.9.



```
Machine linear: showing attribute weights, not support vectors.

        2.5524 * (normalized)  X
 +     -1.0119 * (normalized) Y
 -      0.7903

Number of kernel evaluations: 110 (61.806% cached)



Time taken to build model: 0.02 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          23               76.6667 %
Incorrectly Classified Instances         7               23.3333 %
Kappa statistic                          0.5333
Mean absolute error                      0.2333
Root mean squared error                  0.483
Relative absolute error                 46.1364 %
Root relative squared error             95.4621 %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.733    0.200    0.786      0.733    0.759      0.535    0.767     0.710     Circle
                 0.800    0.267    0.750      0.800    0.774      0.535    0.767     0.700     Cross
Weighted Avg.    0.767    0.233    0.768      0.767    0.766      0.535    0.767     0.705

=== Confusion Matrix ===

  a  b   <-- classified as
 11  4 |  a = Circle
  3 12 |  b = Cross
```

Figure 2.9: Boundary detected by SMO algorithm in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for well-balanced but overlapping data set.

From Figure 2.9, the effects of overlapping can be clearly seen with a lower accuracy calculated as compared to the classifying done in Figure 2.4. The overlapping data points from both circles and crosses make it difficult for the algorithm to effectively classify circles and crosses. This can be repeated, yet again with an unbalanced data set (with the same IR as the data set in Figure 2.7), that is also overlapping as shown in Figure 2.10.

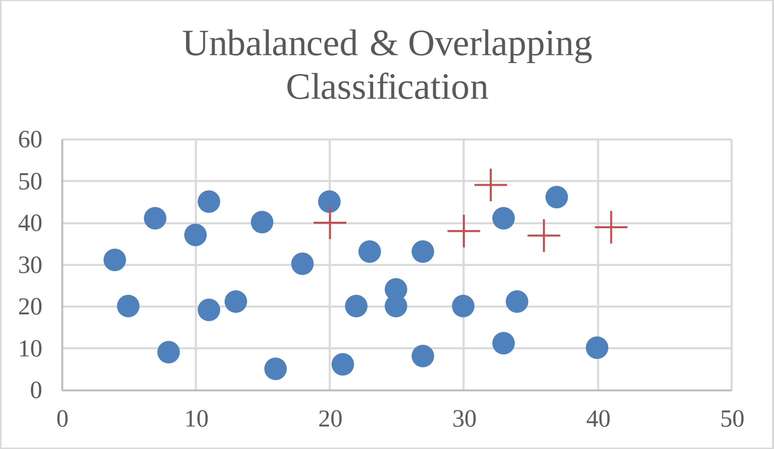
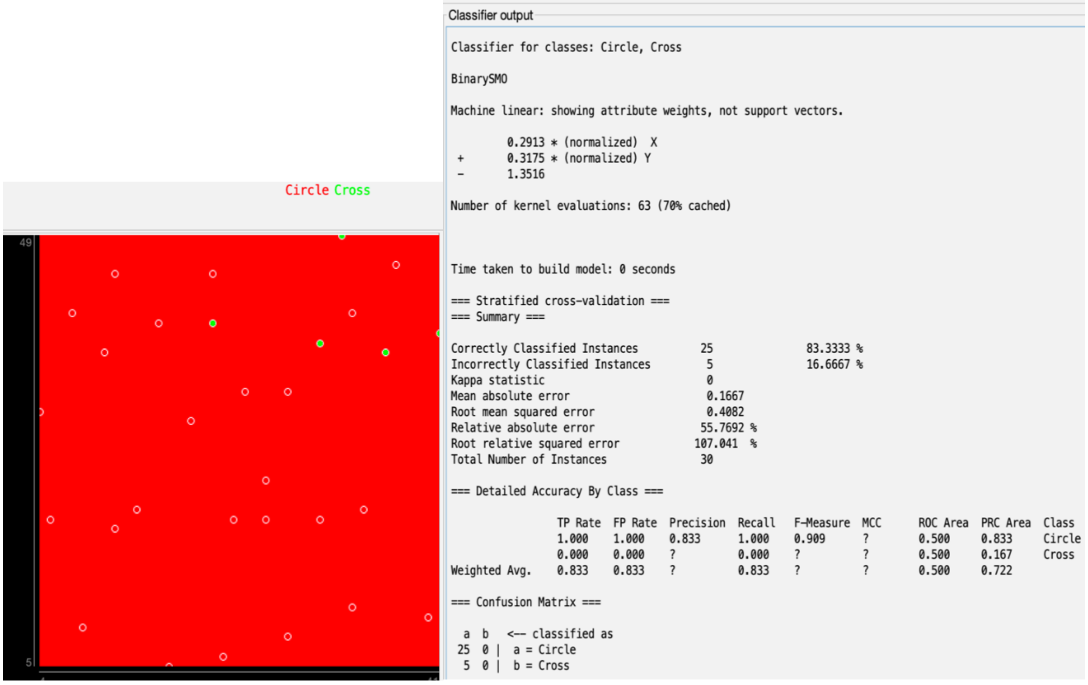


Figure 2.10: Distribution of unbalanced data set of 25 circles to 5 crosses, with overlapping data points.

Performing classification, again using the SMO algorithm in WEKA, yields the boundary and results as shown in Figure 2.11.

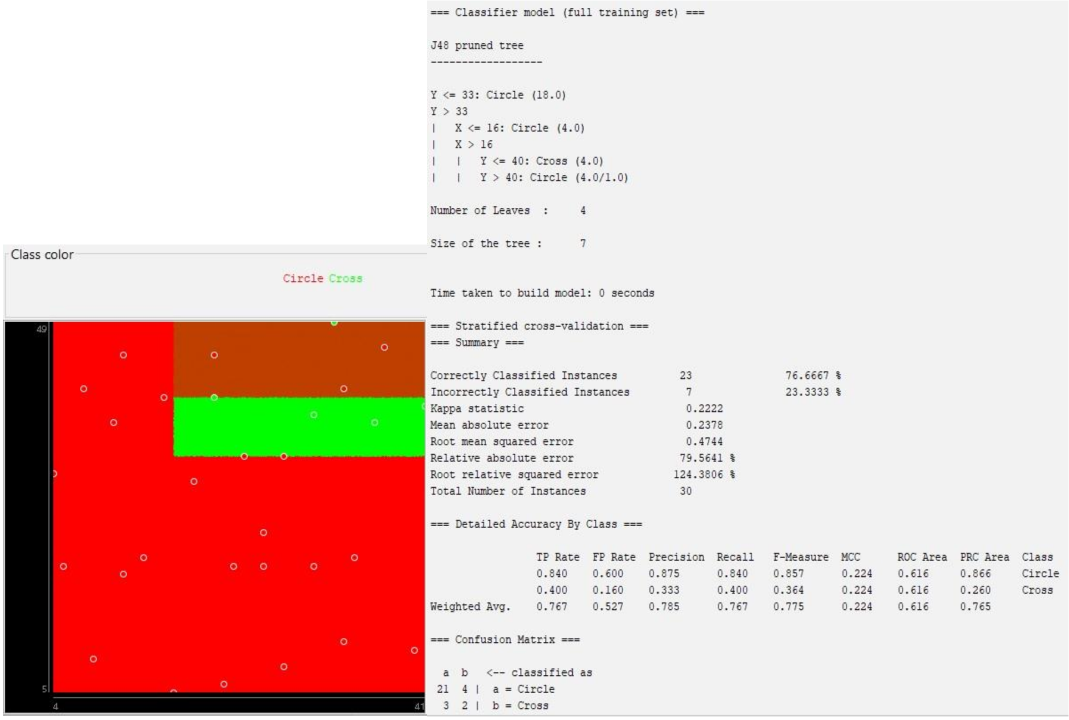


Classifier output

```
Classifier for classes: Circle, Cross

BinarySMO

Machine linear: showing attribute weights, not support vectors.

         0.2913 * (normalized)  X
 +       0.3175 * (normalized) Y
 -       1.3516

Number of kernel evaluations: 63 (70% cached)



Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          25               83.3333 %
Incorrectly Classified Instances         5               16.6667 %
Kappa statistic                          0
Mean absolute error                      0.1667
Root mean squared error                  0.4082
Relative absolute error                 55.7692 %
Root relative squared error            107.041  %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    1.000    0.833      1.000    0.909      ?        0.500     0.833     Circle
                 0.000    0.000    ?          0.000    ?          ?        0.500     0.167     Cross
Weighted Avg.    0.833    0.833    ?          0.833    ?          ?        0.500     0.722

=== Confusion Matrix ===

  a  b   <-- classified as
 25  0 |  a = Circle
  5  0 |  b = Cross
```

Figure 2.11: Boundary detected by SMO algorithm in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for unbalanced and overlapping data set.

From Figure 2.11, it is clear that the effects of overlapping and data imbalance makes it impossible for the algorithm to distinguish between different classes and random noise. Even when using the J48 classifier (which effectively classified the instances in Figure 2.7), the detected boundary and accuracy calculated is not better (even worse than so), as compared to the SMO algorithm, as shown in Figure 2.12.



```
=== Classifier model (full training set) ===

J48 pruned tree
------------------

Y <= 33: Circle (18.0)
Y > 33
|   X <= 16: Circle (4.0)
|   X > 16
|   |   Y <= 40: Cross (4.0)
|   |   Y > 40: Circle (4.0/1.0)

Number of Leaves  :     4

Size of the tree :      7


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          23               76.6667 %
Incorrectly Classified Instances         7               23.3333 %
Kappa statistic                          0.2222
Mean absolute error                      0.2378
Root mean squared error                  0.4744
Relative absolute error                 79.5641 %
Root relative squared error            124.3806 %
Total Number of Instances               30

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.840    0.600    0.875      0.840    0.857      0.224    0.616     0.866     Circle
                 0.400    0.160    0.333      0.400    0.364      0.224    0.616     0.260     Cross
Weighted Avg.    0.767    0.527    0.785      0.767    0.775      0.224    0.616     0.765

=== Confusion Matrix ===

  a  b   <-- classified as
 21  4 |  a = Circle
  3  2 |  b = Cross
```

Figure 2.12: Boundary detected by J48 classifier in WEKA (Left) and the classification results and confusion matrix generated in WEKA (Right) for unbalanced and overlapping data set.

From Figure 2.12, while the J48 classifier was able to detect the instances of crosses, unlike the SMO classifier, **it did so at the cost of lowering the overall accuracy** (accuracy of 76.67% as compared to 83.33% in Figure 2.11). Comparing the results in Figures 2.15 and 2.16, it is clear that in this case, selecting rules that favour classification of minority classes (i.e. crosses), results in lower accuracy than otherwise, as the presence of overlapping data points and small disjuncts make it near impossible for the algorithm to effectively set a boundary to classify data points.

Situations like this are even more severe when dealing with data sets of huge volume (millions of instances), very high overlapping and IR of 1000:1 and higher. Luckily, a number of methods exist to deal with highly unbalanced data sets, which will be investigated in Section 2.4.2.

2.4.2 Methods to Handle Unbalanced Data Sets

Unbalanced data sets are a widely studied topic in the field of data science, with a multitude of methods and algorithms, often done with popular classification algorithms, to deal with them. Krawczyk, 2016 proposed **three main ways of handling unbalanced data sets** as listed:

(i) Data-level: Manipulate data points to even out classes and/or deleting problematic data points.

(ii) Algorithm-level: Adjust the bias of the classification algorithm used towards majority/minority classes.

(iii) Hybrid: Utilising both data-level and algorithm-level techniques to get the best of both methods to handle unbalanced data sets.

(Das et al., 2018; Krawczyk, 2016)

The three main ways to handle unbalanced data sets are investigated in the following sections.

2.4.2.1 Data-level Approaches

At the data level, **resampling data from unbalanced data sets is conducted to rebalance the data set between majority and minority classes**, which involves either oversampling minority classes or under-sampling majority classes (sometimes both are used simultaneously). (Das et al., 2018) A wide variety of techniques for under sampling and oversampling are used when dealing with unbalanced data sets.

Under sampling majority classes is conducted to reduce the number of instances of majority classes in the data set. Examples of under sampling methods include Random Under Sampling (RUS), Condensed Nearest Neighbour (CoNN) and Tomek Links (TL). RUS, for example removes instances of majority classes at random, while CoNN removes majority class instances that are far from the decision boundary and deems them irrelevant, and TL removes instances that are considered noise and are close to the boundary, which is at the opposite end of the approach as compared to CoNN. (Das et al., 2018)

Oversampling minority classes is done to increase the number of minority class instances to balance out the majority class instances. This can be done in numerous ways, with the most popular method being Synthetic Minority Over-sampling Technique (SMOTE), along with extensions of SMOTE including Borderline-SMOTE, Adaptive Synthetic (ADASYN), LN-SMOTE and safe-level SMOTE. SMOTE works by introducing new minority class instances within clusters of minority classes at random. (Das et al., 2018)

While resampling by undersampling majority classes or oversampling minority classes is a widely used approach to unbalanced data sets, resampling may not be an ideal way to handle unbalanced data sets, as **resampling can remove data points that can significantly help with classifying (in undersampling) and introduce more unnecessary data points that cause overlapping (in oversampling).** (Sun et al., 2009; Krawczyk, 2016) In addition to that, resampling does not do well if minority classes have subcategories. (Sun et al., 2009) This has lead to new resampling techniques, like utilising Principal Component Analysis (PCA), fuzzy logic and clustering to cater for subcategories in majority and minority classes. (Das et al., 2018)

For this study, **hierarchical classification will be utilised**. Hierarchical classification is a **type of supervised learning classification that makes use of already known structure, taxonomy or hierarchy of classes to group simlar classes** for classification. (Silla & Freitas, 2011) This is especially true when classification problems are hierarchical in nature, hierarchical classification is a novel method to make use of existing hierarchies to aid in the classification of unbalanced data sets, especially if paired up with resampling techniques (under or over-sampling), can potentially outperform resampling alone (known as flat sampling in literature). (Pereira, Costa, & Silla, 2021) For this study, **hierarchical classification will be paired up with random under sampling to test detection rates of DDoS attacks with or without grouping under hierarchy.**

2.4.2.2 Algorithm-level Approaches

In terms of the algorithms used to classify majority and minority classes, to cater for bias in the classification algorithms used, several methods can be implemented to adjust the bias towards majority class instances.

As classification algorithms typically run-on cost minimisation, cost-sensitive learning can be applied to the algorithms. This is typically done by significantly increasing the cost for misclassifying minority class instances. (Das et al., 2018; Krawczyk, 2016) The implementation of this method differs based on the classification algorithm used, from setting weights, to considering the number of degrees of freedom that exist in the classification algorithm for the data set. (Das et al., 2018) This is a recommendation for future studies on DDoS attack detection due to the nature of the minority class instances being DDoS attacks, as the cost of not detecting them in real life is also high, reflecting real-life costs of poor detection.

Besides cost based learning, other methods like boundary shifting methods, single class learning, active learning, kernel perturbation techniques and discriminate regression based supervising learning methods can also be used at the algorithm level to handle unbalanced data sets. (Das et al., 2018)

2.4.2.3 Hybrid Approaches

Hybrid methods incorporate both data and algorithm-level techniques to not only get the best of both methods, but also minimise the disadvantages from both. Such techniques include performing resampling (both under sampling and over-sampling) with classifier ensembles (especially bagging and boosting) or with cost-sensitive classifying using popular classifiers like Random Forest (RF), Support Vector Machines (SVM) and C4.5 (which is known as J48 in WEKA). In some cases, hybrid methods outcompete data-level and algorithm-level methods, though more research is still needed to increase the cooperation between data-level and algorithm-level techniques. (Das, et al., 2018) Some of these Hybrid methods include EasyEnsemble and BalanceCascade. (López et al., 2013)

In this demonstration of classification of circles and crosses in Section 2.4.1, the **problems that arise from unbalanced data sets and their characteristics are highlighted.** Section 2.4.2 also investigated the various methods that are used to deal with unbalanced data sets, which can be implemented on the data itself, the classification algorithm used, or both. In the next section, **the classification algorithms to use in the study are thoroughly investigated.**

2.5 Popular Classification Algorithms

Among all the classification algorithms available on WEKA, the algorithms that are of interest are **Naïve Bayes (NB), Random Forest (RF), J48, JRip and K-Nearest Neighbours (KNN).** The selected algorithms for the study are explained within this section as followed.

2.5.1 Naïve Bayes (NB)

NB is a supervised, probabilistic classifier, which like in its name, is completely based on Bayesian Statistics. (IBM, n.d.) Bayesian Statistics is fundamentally based on Bayes Theorem in conditional probability, which is defined by formula (2.1).

$$P(H|D) = \frac{P(D|H)P(H)}{P(D)} \tag{2.1}$$

where

H = Hypothesis

D = Data

P(H) = Probability of H

P(D) = Probability of D

P(H | D) = Probability of H given D

P(D | H) = Probability of D given H

In Bayesian Statistical terms, P(H | D) is known as the **posterior probability**, P(H) is the **prior probability**, P(D | H) is the **likelihood** and P(D) is the **marginal likelihood**. They are named this way as the goal of Bayesian Statistics is to find the posterior probability based on prior beliefs and likelihoods. (Brewer, n.d.) Basically, we want to know the probability of the hypothesis H, given new evidence D is true.

This is, however, the case for a single hypothesis H. In general, where there are N hypotheses, we can rewrite equation 2.1 as followed:

$$P(H_i|D) = \frac{P(D|H_i)P(H_i)}{P(D)} \quad (2.2)$$

where i ranges from 1, 2…, N.

In a general Bayes box, the properties of the prior and posterior probabilities are as followed:

$$\sum_{i=1}^{N} P(H_i) = 1, \sum_{i=1}^{N} P(H_i)P(D|H_i) = P(D), \sum_{i=1}^{N} P(H_i|D) = 1 \quad (2.3)$$

(Brewer, n.d.)

By default, **NB uses probability distributions to decide and classify data,** though within NB, kernel density estimators and supervised discretization may also be used.

WEKA 3.8.6 is pre-installed with a wide variety of NB classifiers, each tailored to different types of data sets used. Among the NB classifiers available in WEKA are NaiveBayes, NaiveBayesSimple, ComplementNaiveBayes, HNB, DMNBText, NaiveBayesMultinomial-Updateable and NaiveBayesUpdateable. (Witten et al., 2011)

NB is popular due to its simplicity in terms of the algorithm itself, as NB assumes that the features are independent of one another, not to mention that it performs relatively well on classification tasks. (Novakovic, 2010)

2.5.2 Random Forest (RF)

The **RF Classifier is an ensemble of decision trees, drawn from a sample from the training data set (known as the bootstrap sample), based on a set node size, number of trees and features sampled, which classifies the data set based on the decision trees built.** (IBM, n.d.) RF accepts randomly selected features to generate decision trees with controlled deviation, where the classification is done based on majority or weighted scores. (Sharafaldin, et al., 2019)

RF randomly generates decision trees of set sizes (hence the name Random Forest), based on the rules inferred from data sets. A notable feature of RF is that the variance of the model goes down as more trees are used in the forest, while keeping bias constant. (Sharafaldin et al., 2019)

2.5.3 J48

Much like RF, **J48 is a decision tree classifier that is based on the C4.5 learner, that utilises binary trees to classify.** (Witten et al., 2011; C, 2014) Developed by Ross Quinlan, J48 is the Java adaptation of the C4.5 algorithm, in which its primary goal is to aid in supervised learning and classification to generate decision trees. (Meena & Choudhary, 2017) As compared to RF, **J48 has reportedly higher accuracy, but takes longer to create the decision models for classification.** (Hermawan et al., 2021) Besides J48, there is J48graft, which is an enhanced version of J48 that grafts more branches during postprocessing to utilise the pros of clustering methods like bagged and boosted trees, keeping the model simple to interpret. J48graft also finds parts of instance space that are devoid of data samples, or contain wrongly classified data points, while looking at different classifications and tests that could have been chosen at the nodes of the leaves bounding the region. (Witten et al., 2011)

2.5.4 JRip

The **JRip Classifier is a decision tree classifier that uses the Repeated Incremental Pruning to Produce Error Reduction (RIPPER) algorithm.** (Rajeswari & Arunesh, 2016) In the RIPPER algorithm, classes are investigated in ascending order of size while a preliminary set of rules is created via incremental reduced-error pruning, which takes into account description lengths (DL), which by itself is calculated using a complicated formula, involving the number of bits required to send a cluster of examples based on a collection of rules, where each rule is based on an integer *k* number of conditions, and to send *k* multiplied by half to adjust for feature repetition. (Witten et al., 2011)

2.5.5 K-Nearest Neighbours (KNN)

The **K-Nearest Neighbours (KNN) classifier (known as IBk in WEKA), is a supervised, non-parametric learning classifier that utilises distances functions (Euclidean, Manhattan, Minkowski and Hamming) to cluster and classify data points using clustering algorithms like K-Means and K-Median clustering.** (IBM, n.d.)

While KNN is a simple and adaptive classifier, KNN however is poor in terms of scalability, dealing with high number of dimensions of data and is susceptible to over-fitting. (IBM, n.d.)

The next section tabulates all the research papers that utilise each classification algorithm that are of significance to the study.

2.5.6 Tabulation of Research Papers Researched

Table 2.5: Table of Research Papers, displaying Author & Year, Problem Investigated, Classifier(s) and Sampling Technique(s) used to study, results and any remarks on the study.

Author & Year	**Problem Investigated**	**Classifier(s) Used**	**Sampling Technique(s) Used**	**Results**	**Remarks**
Bolodurina, Shukman, Parfenov, Zhigalov & Zabrodina, 2020.	Impact of data balancing algorithms in network traffic classification problem on various types of DDoS attacks in the CICIDDoS2019 data set.	Support Vector Machines (SVM), RF, Gradient Boosting (GBM).	Synthetic Samples of Minority (SMOTE), ADASYN.	SVM paired with SMOTE has highest balanced accuracy on Portmap DDoS attack (94.81%).	
Gohil & Kumar, 2020.	Used heavily supervised classification algorithms on the CICIDDoS2019 data set.	Decision Trees (DT), NB, Logistic Regression (LR), SVM, k-Nearest Neighbours (KNN), RF.	RUS for both DDoS attack and BENIGN instances (Choosing 30k out of 200k rows).	DT, RF and KNN performed best with 100% accuracy, NB still performed well at 96.25%, while LR and SVM did poorly at 79.34% and 50.10% respectively.	

Table 2.5 (Continued)

Chkirbene, Eltanbouly, Baschendy, AlNaimi & Erbad, 2020.	Utilising a hybrid of Random Forest (RF) and Classification and Regression Trees (CART), called Hybrid Anomaly Intrusion Detection System (HAIDS) on the UNSW-NB15 and K99 data sets.	RF and CART.	None used, only acknowledged to perform oversampling in future studies.	HAIDS performed better in terms of accuracy and lower False Positive Rates (known as False Alarm Rate in study).	
Ho, Yap & Khor, 2021.	Improving detection rate of intrusions using sampling techniques, namely Random Under-Sampling (RUS), Synthetic Minority Over-sampling Technique (SMOTE), and both, using the CICIDDoS2017 data set.	Gaussian Naïve Bayes (GNB), C4.5/J48, Neural Network-Multilayer Perception (NN-MLP), KNN, and Logistic Regression (LR).	SMOTE, RUS, Random Over Sampling (ROS), Combining Sampling.	C4.5/J48 has highest average TPR of 99.27% without using sampling techniques, increased overall TPR (99.85%) and TPR of 12 types of DDoS attacks when using RUS to reduce BENIGN instances to between 30-90% of original size, when using SMOTE, highest average TPR when size of minority classes increased to 250% of data set, best result achieved when under sampled 30% on	

Table 2.5 (Continued)

				BENIGN, and 300% over sampled on seven minority classes.	
Almaraz-Rivera, Perez-Diaz & Cantoral-Ceballos, 2022.	Build a novel Intrusion Detection System (IDS) based on Machine Learning and Deep Learning to address class imbalance problem using the UNSW Bot-IoT data set.	SVM, DT and RF from Machine Learning, Recurring Neural Network (RNN), Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) and MLP from Deep Learning.	Feature-based sampling.	Machine Learning models like RF and DT outperform Deep Learning models in terms of accuracy, precision, recall/TPR and F1-score for both binary and multiclass classification, DT is better at generalising among all Machine Learning models. Achieved >99% accuracy in all three feature sets, and 100% across accuracy, precision, recall/TPR and F1-score for multiple combinations of Normal/BENIGN vs DDoS/DoS protocols using RF and DT.	
Koroniotis, Moustafa, Sitnikova	Developed the UNSW Iot-Bot Data Set to address the problem of a lack of a	SVM from Machine Learning, RNN and	None used.	For SVM, performed better in terms of accuracy, TPR/recall for full feature model, but better precision for 10-best	As addressed by Almaraz-Rivera,

Table 2.5 (Continued)

& Turnbull, 2018.	realistic data set that can cover various botnet attacks, and perform statistical analysis using machine and deep learning algorithms.	LSTM from Deep Learning.		feature model. For RNN and LSTM, performed slightly worse on full feature model in terms of accuracy, TPR/recall and precision as compared to 10-best feature model.	Perez-Diaz & Cantoral-Ceballos, 2022, they did not perform class balancing time performance evaluation.
Erhan & Anarim, 2019	Examines structured probability distribution functions (namely Gaussian/Normal, Generalised Extreme Value/GEV and Logistic), by using probability distribution fitting together with binary hypothesis testing, among	Gaussian/Normal, Generalised Extreme Value/GEV & Logistic Distributions, probability distribution fitting using Likelihood	None used.	Overall, for BOUN DDoS data set, Gaussian has highest accuracy from likelihood ratio test among the features tested, while for CAIDA data sets, GEV is the winner in terms of accuracy.	

Table 2.5 (Continued)

	frequently used traffic features in DDoS detection to aid DDoS researchers to select better statistical techniques, using the BOUN DDoS, CAIDA 2007 & CAIDA 2008 data sets.	Ratio Test, Akailike & Bayesian Information Criterion.			
Jaszcz & Połap, 2022.	Propose a framework called AIMM (Artificial Intelligence Merged Methods), which is based on three modules, pre-processing data, classification and decision-making. The decision-making module, obtains probabilities from all AI methods used (i.e. KNN and ANN) to analyse to make the final decision for the attack, together with soft sets inference and weighted	KNN, ANN.	RUS on BENIGN/non-attack classes.	Obtained accuracy of 99.5% on unbalanced data set, 100% on balanced data set. As compared to other state-of-art studies, namely Isolation Forest (97.1%), Statistical models (97.5%), K-means clustering (98.2%), Low-rate DDoS Detection Method (LDDM) (95.0%), Clustering (99.4%) and Fuzzy Logic with entropy analysis (99.4%), AIMM did considerably better.	

Table 2.5 (Continued)

	averaging technique on the BOUN DDoS data set. Then, results are compared to other state-of-art studies.				
Meena & Choudhary, 2021	Using various algorithms in WEKA (namely J48 and Naïve Bayes) to perform intrusion detection via 10-fold classification on KDD99, NSL KDD data sets.	J48graft, NB.	None used.	J48graft Accuracy 99.435%, TPR 99.426%, precision 99.4%, F1-score 0.993. NB Accuracy 92.715%, TPR 85.635%, precision 85.2%, F1-score 0.916.	
Ahmim, Maglaras, Ferrag, Derdour, & Janicke, 2019.	Proposes a Hierarchical Intrusion Detection System (IDS) that combines three types of classifiers, namely Reduced Error Pruning (REP) Tree, JRip and Forest PA, using the CICIDDoS2017 data set.	REP Tree, JRip & Forest PA.	None used.	Model reports lower False Positive Rate (False Alarm Rate in study), higher overall detection rate and accuracy than multiple other studies, including studies that utilise REP Tree, JRip and Forest PA individually.	

From investigating all the different classification algorithms to be used in the study, it is evident that **every classifier investigated is different in terms of the implementation of the classifier and the custom inputs needed to run the classifier algorithm,** not to mention that all the classifiers have their own pros and cons when dealing with different types and volume of data. In this study, the effectiveness of each classifier will be compared based on the evaluation metrics that will be investigated in the next section.

2.6 Evaluation Metrics

The project involves generating confusion matrices using WEKA, like the one shown in Table 2.6.

Table 2.6: Confusion Matrix

Predicted/Actual	DDoS Attack	BENIGN
DDoS Attack	True Positive (TP)	False Positive (FP)
BENIGN	False Negative (FN)	True Negative (TN)

The following terms for the study are defined as followed:

(i) True Positive (TP): Number of instances of a class of interest that are correctly classified as such.

(ii) True Negative (TN): Number of instances that are not of the class of interest correctly classified as such.

(iii) False Positive (FP): Number of instances that are not of the class of interest incorrectly classified as the class of interest.

(iv) False Negative (FN): Number of instances that are the class of interest incorrectly classified as members of class not of interest.

(Dutt, et al., 2019)

From the four terms defined above, the evaluation metrics that are used for the project, namely **Accuracy (ACC), True Positive Rate (TPR), Precision (PREC) and F1-measure (F1, sometimes known simply as F-measure),** are defined in Equations 2.4 to 2.7:

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \tag{2.4}$$

$$TPR = \frac{TP}{TP+FN} \tag{2.5}$$

$$PREC = \frac{TP}{TP+FP} \tag{2.6}$$

$$F1 = \frac{2}{\frac{1}{PREC}+\frac{1}{TPR}} = \frac{2\times PREC\times TPR}{PREC+TPR} = \frac{2\times TP}{2\times TP+FP+FN} \tag{2.7}$$

ACC measures how accurate the classifier is at classifying instances correctly, TPR (sometimes known as recall, sensitivity or hit/detection rate) is the proportion of correctly classified DDoS attack instances to the total number of DDoS attack instances, PREC (also known as positive predictive value) measures the rate of correctly classified DDoS attack instances among all instances classified as DDoS attacks, and F1 is a statistical measure defined as the harmonic mean of PREC and TPR, taking into consideration FP and FN. Another measure that is used in the study is the AUC, which is short for Area Under the Receiver operating characteristics (ROC) Curve, which is a curve that plots TPR against False Positive Rate (FPR). AUC takes a value between 0.5 and 1, where 0.5 is the AUC for a completely random classifier, and 1 is the AUC for a perfect classifier. (Abro, et al., 2021; Abro, et al., 2020; Fawcett, 2006; Sun, et al., 2009) FPR (sometimes also known as False Alarm Rate) is defined in Equation 2.8.

$$FPR = \frac{FP}{TN+FP} \tag{2.8}$$

(Hajian-Tilaki, 2013)

Another measure that can be used is the **G-Mean**, which is defined as the **square root of the product of TPR and True Negative Rate (TNR)**, where TNR (sometimes known as specificity) is defined as the rate of instances that are not of the class of interest correctly classified as such. The G-Mean and TNR are defined in equations 2.9 and 2.10 respectively.

$$GMean = \sqrt{TPR \times TNR} \tag{2.9}$$

$$TNR = \frac{TN}{TN+FP} = 1 - FPR \qquad (2.10)$$

(Sun et al., 2009)

For an unbalanced data set, **accuracy is the least reliable metric** to evaluate classifiers, as even with high accuracy, the correctly classified instances could still easily be mostly (if not comprised entirely of) instances of the majority classes. Especially in cases where positive cases (and hence True Positives) far outnumber negative cases in the data set $(TP + FN \gg TN + FP)$, accuracy could just be approximated as TPR (TNR if negative cases appear in far greater numbers than positive cases, vice versa). All the other metrics described (TPR, PREC, F1, AUC and GMean) however, are suitable for unbalanced data sets, as they consider FPR and FNR to normalise classification of both majority and minority classes. However, ACC will be used in case of a complete tie between all the other metrics. A complete literature review is therefore, conducted to lay out the background understanding for the study, and henceforth, the methodology used for the study is investigated in the next chapter.

The chosen metrics are listed below:

1. TPR/Recall
2. F1
3. AUC
4. G-Mean
5. ACC (To complement first four metrics)

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

The study starts with **data pre-processing and some preliminary classification using default parameters on all classifiers (K = 10 for KNN/IBk) on a sampled data set (at 3% under sampling)** created from the Consolidated DDoS Data Set. Based on the results of preliminary classification, **the J48 Classifier was selected to continue with testing different values of the selected parameters, confidenceFactor (C) and minNumObj (M)** to obtain the values of ACC, TPR, PREC, F1, AUC and GMEAN (both overall and for individual classes) for preliminary ungrouped classification.

Using the same sampled data set, **grouping of attack classes on three levels of a pre-defined hierarchy of DDoS attacks (see Figure 3.3) are performed**, followed by first performing classification using the default parameters of the J48 Classifier (C = 0.25, M = 2) and then, different values of C and M from the J48 Classifier were tested in WEKA to obtain values of ACC, TPR, PREC, F1, AUC and GMEAN (both overall and for individual classes) for each value of C and M tested. The detailed methodology for the study, first illustrated with the flowchart for the whole study in Section 3.2, then thoroughly explained in Section 3.3, with the technical details for the software used, the Waikato Environment for Knowledge Analysis (WEKA), outlined in Section 3.4.

3.2 Flowchart of Study

The flowchart for the study is as shown in Figure 3.1, in which the complete explanation for the flowchart fully described in Section 3.3.

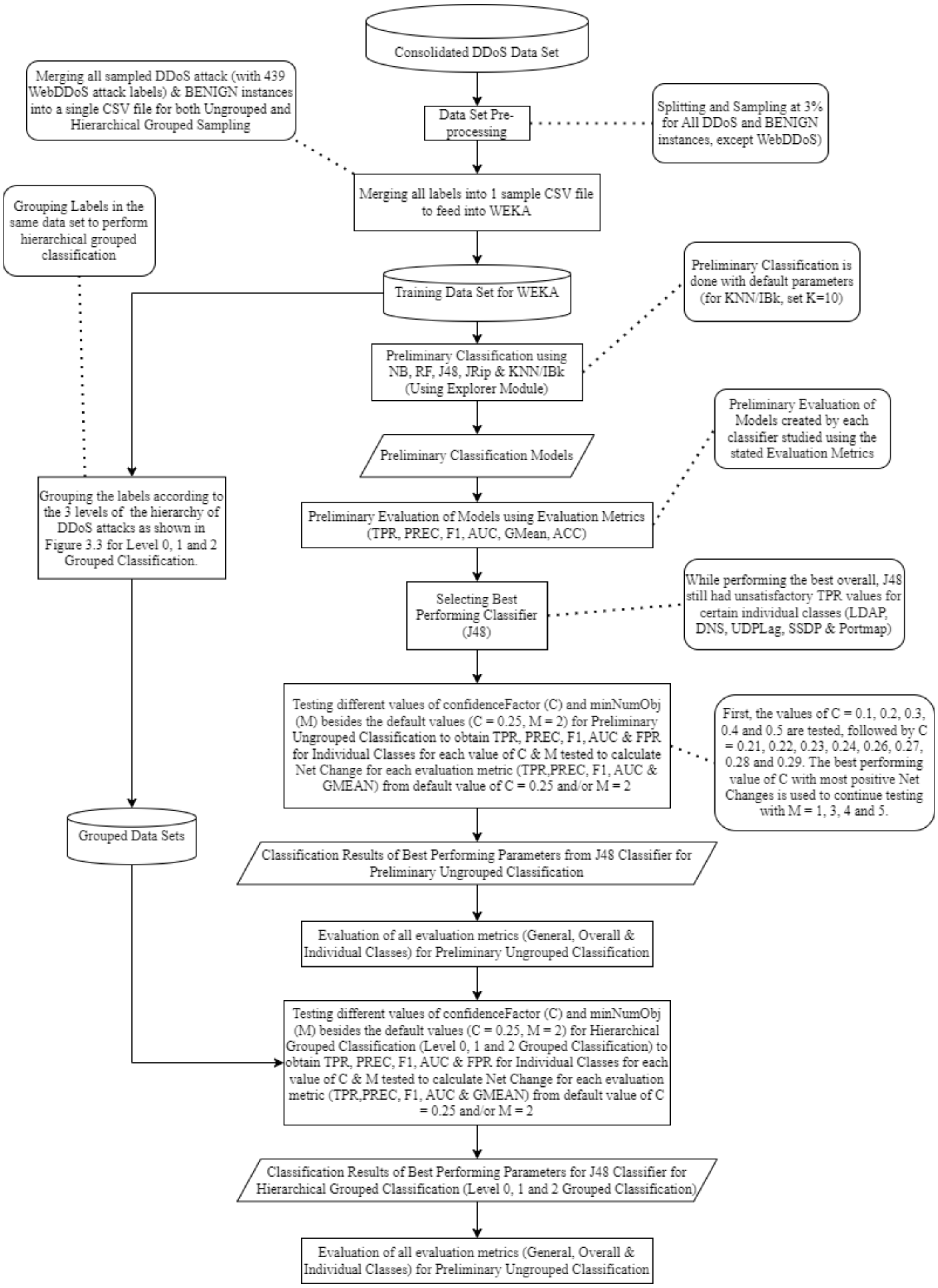

Consolidated DDoS Data Set
Merging all sampled DDoS attack (with 439 WebDDoS attack labels) & BENIGN instances into a single CSV file for both Ungrouped and Hierarchical Grouped Sampling
Data Set Pre-processing
Splitting and Sampling at 3% for All DDoS and BENIGN instances, except WebDDoS)
Merging all labels into 1 sample CSV file to feed into WEKA
Grouping Labels in the same data set to perform hierarchical grouped classification
Training Data Set for WEKA
Preliminary Classification is done with default parameters (for KNN/IBk, set K=10)
Preliminary Classification using NB, RF, J48, JRip & KNN/IBk (Using Explorer Module)
Preliminary Evaluation of Models created by each classifier studied using the stated Evaluation Metrics
Preliminary Classification Models
Grouping the labels according to the 3 levels of the hierarchy of DDoS attacks as shown in Figure 3.3 for Level 0, 1 and 2 Grouped Classification.
Preliminary Evaluation of Models using Evaluation Metrics (TPR, PREC, F1, AUC, GMean, ACC)
Selecting Best Performing Classifier (J48)
While performing the best overall, J48 still had unsatisfactory TPR values for certain individual classes (LDAP, DNS, UDPLag, SSDP & Portmap)
Testing different values of confidenceFactor (C) and minNumObj (M) besides the default values (C = 0.25, M = 2) for Preliminary Ungrouped Classification to obtain TPR, PREC, F1, AUC & FPR for Individual Classes for each value of C & M tested to calculate Net Change for each evaluation metric (TPR,PREC, F1, AUC & GMEAN) from default value of C = 0.25 and/or M = 2
First, the values of C = 0.1, 0.2, 0.3, 0.4 and 0.5 are tested, followed by C = 0.21, 0.22, 0.23, 0.24, 0.26, 0.27, 0.28 and 0.29. The best performing value of C with most positive Net Changes is used to continue testing with M = 1, 3, 4 and 5.
Grouped Data Sets
Classification Results of Best Performing Parameters from J48 Classifier for Preliminary Ungrouped Classification
Evaluation of all evaluation metrics (General, Overall & Individual Classes) for Preliminary Ungrouped Classification
Testing different values of confidenceFactor (C) and minNumObj (M) besides the default values (C = 0.25, M = 2) for Hierarchical Grouped Classification (Level 0, 1 and 2 Grouped Classification) to obtain TPR, PREC, F1, AUC & FPR for Individual Classes for each value of C & M tested to calculate Net Change for each evaluation metric (TPR,PREC, F1, AUC & GMEAN) from default value of C = 0.25 and/or M = 2
Classification Results of Best Performing Parameters for J48 Classifier for Hierarchical Grouped Classification (Level 0, 1 and 2 Grouped Classification)
Evaluation of all evaluation metrics (General, Overall & Individual Classes) for Preliminary Ungrouped Classification

Figure 3.1: Flowchart of workflow of study.

3.3 Detailed Explanation of Flowchart

The study was conducted in **three parts.** The first part of the study involved **sampling of the Consolidated DDoS Data Set to create a sampled Data Set** to use in WEKA as thoroughly explained in Section 3.3.1. The second part of the study is **Preliminary Classification,** where the performance of all selected classifiers (with default parameters used) were compared as highlighted in Section 3.3.2, followed by **testing different values of C and M of the selected classifier, J48** to obtain evaluation metrics (general, overall and detailed). The third and penultimate part of the study is **Hierarchical Classification,** where after the best performing classifier is selected, the **labels in the data set are grouped by 3 levels of the DDoS hierarchy as shown in Figure 3.2,** as clearly described in Section 3.3.3, followed by yet again, testing different values of C and M on classification results.

3.3.1 Sampling of Consolidated DDoS Data Set

Due to limited heap memory of WEKA, random under sampling is performed on the Consolidated DDoS Data Set. **All the different labels are extracted from the Consolidated Data Set, which are then randomly sampled without repetition using Random Under-Sampling (RUS) at 3% (except for WebDDoS attack labels)** into separate CSV files. The resulting number of instances sampled for each label in the Consolidated DDoS Data Set are as shown in Table 3.1.

Table 3.1: Number of instances sampled for each label present in Consolidated DDoS Data Set (under 3% sampling)

Label	# instances	# rows to sample
WebDDoS	439	439*
Portmap	17 676	530
UDPLag	34 891	1046
NTP	119 528	3585
SSDP	256 832	7704
LDAP	410 301	12 309
DNS	490 813	14 724
SNMP	514 957	15 448
Syn	594 129	17 823

Table 3.1 (Continued)

UDP	688 393	20 651
NetBIOS	747 772	22 433
MSSQL	998 191	29 945
TFTP	1 951 336	58 540
BENIGN	2 384 051	71 521
Total # instances sampled		**276 698**

*As WebDDoS has only 439 instances in Consolidated DDoS Data Set, WebDDoS attack labels are not under sampled, hence all WebDDoS instances are included into the sampled data set.

The **sampled files (together with the WebDDoS file) are combined to form a combined sampled file** of file size 65.8MB (a small portion of the original 2.96 GB data set), to be inputted into WEKA for performing both preliminary and hierarchical grouped classifications, outlined in Sections 3.3.2 and 3.3.3.

3.3.2 Preliminary Classification

Preliminary classification is done in WEKA using all the selected machine learning classifiers for the study back in Section 2.5, namely **Naïve Bayes (NB), Random Forest (RF), J48, JRip and K-Nearest Neighbours (KNN), under 10-fold cross validation**, the classification is done using **default parameters for all classifiers** used (for KNN, set K=10) to obtain preliminary detection rates (measured by TPR) of each DDoS attack type. The default parameters for each classifier are set as tabulated in Table 3.2.

Table 3.2: Command Line Interface (CLI) command settings for each classifier used to classify DDoS attack labels for each classifier used in WEKA.

Classifier	CLI Command
NB	weka.classifiers.bayes.NaiveBayes
J48	weka.classifiers.trees.J48 -C 0.25 -M 2
RF	weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
JRip	weka.classifiers.rules.JRip -F 3 -N 2.0 -O 2 -S 1
KNN	weka.classifiers.lazy.IBk -K 10 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A \"weka.core.EuclideanDistance -R first-last\""

Based on the preliminary classification results as shown in Chapter 4 and Appendix A, J48 is the clear winner among all classifiers used. Therefore, **J48 is selected to be the classifier studied** to continue testing different values of confidenceFactor (-C) and minNumObj (-M) to obtain different values of the evaluation metrics selected (both general and for individual classes) in Section 2.6 (and explained in Section 3.4.2) for Preliminary Ungrouped Classification, and also used for the final part of the study, Hierarchical Grouped Classification.

3.3.3 Hierarchical Grouped Classification

Hierarchical grouped classification is done to improve the detection of the attack labels, involving grouping labels in the sampled data set (particularly those that came from CICIDDoS2019 Data Set), based on the existing hierarchy of DDoS attack labels as shown in Figure 3.2.

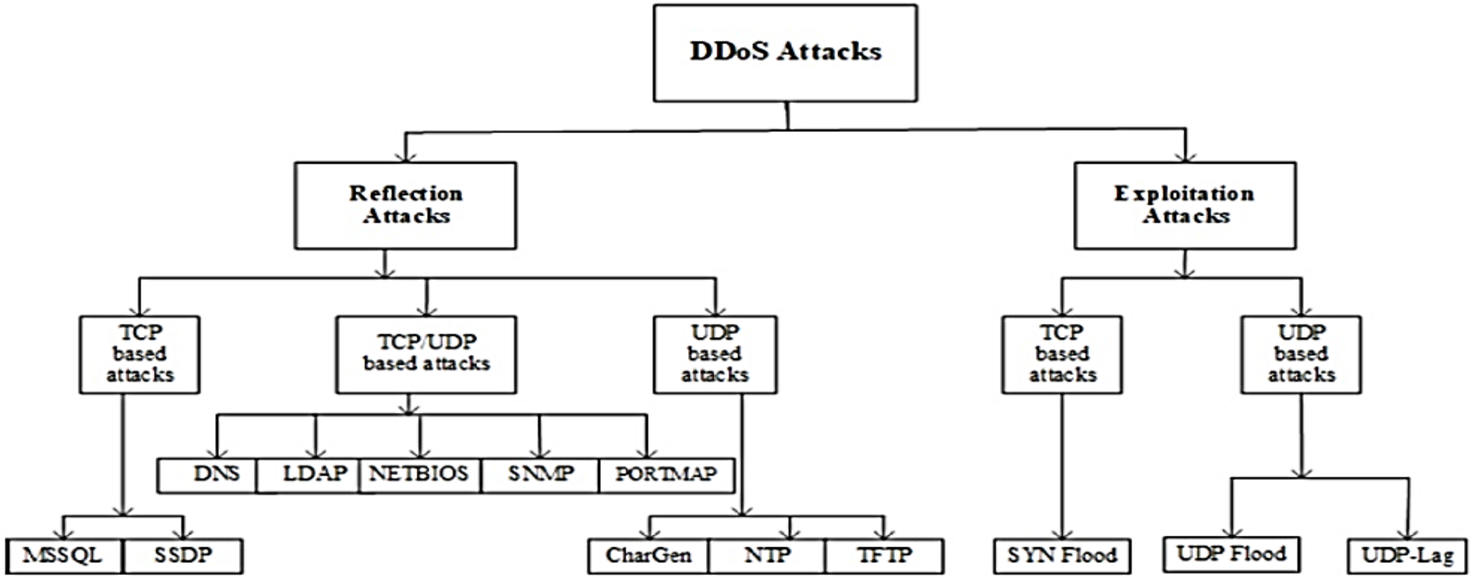


Figure 3.2: Hierarchy of attack labels that came from CICIDDoS2019 Data Set.

(University of New Brunswick, n.d.)

The top three levels of the hierarchy in Figure 3.2 are labelled Level 0, Level 1 and Level 2 as shown in Figure 3.3.

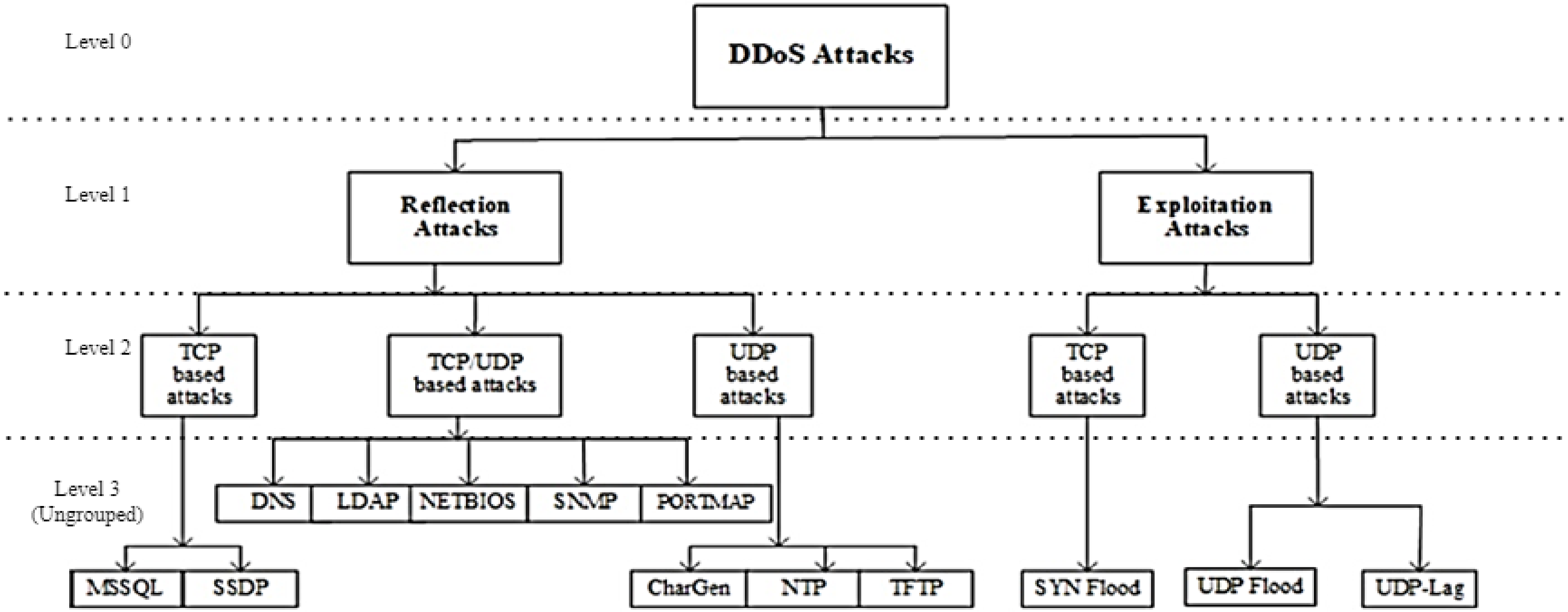


Figure 3.3: Figure 3.2 but with top three levels labelled Level 0, 1 and 2 for the purpose of clarity for the study.

As **most of the attack instances from the Consolidated DDoS Data Set originated from CICIDDoS2019 Data Set**, three types of grouping can be performed based on the three levels highlighted in Figure 3.3, which are described in Sections 3.2.2.3.1 to 3.2.2.3.3. The Experimenter module is used to obtain general values of Accuracy (ACC), True Positive Rate (TPR), Precision (PREC), F1-measure (F1), Area Under ROC Curve (AUC) and True Negative Rate (TNR), using different values of confidenceFactor (-C) and minNumObj (-M) parameters of the J48 classifier (number 3 and 15 respectively in the list in Section 3.3.1). The Explorer module is used again to obtain the values of TPR, PREC, F1, AUC and False Positive Rates (FPR) for the individual classes. To ensure that WEKA does not infer the new labels based on the original label column, the original label column was deleted from the sampled data set for all of Level 0, 1 and 2 Grouped Classification.

3.3.3.1 Level 0 Grouped Classification

In Level 0 Grouped Classification, **all DDoS attack labels (irrespective of the label) are grouped into one single DDoS attack class**, effectively reducing the classification from a multi-class classification to a bi-class classification (DDoS vs BENIGN/Normal). The resulting class distribution is listed as shown in Table 3.3.

Table 3.3: The new labels for DDoS attack classes after Level 0 Grouping.

New Label	Old Label (Number of Instances)	Total Number of Instances
Normal	BENIGN (71 521)	71 521
DDoS	TFTP (58 540) MSSQL (29 945) NetBIOS (22 433) UDP (20 651) Syn (17 823) SNMP (15 448) DNS (14 724) LDAP (12 309) SSDP (7704) NTP (3585) UDPLag (1046) Portmap (530) WebDDoS (439)	205 177
Total		**276 698**

3.3.3.2 Level 1 Grouped Classification

In Level 1 Grouped Classification, the **DDoS attacks are grouped based on whether they are Reflection attacks or Exploitation attacks**. WebDDoS attacks however are not part of the hierarchy as shown in Figures 3.2 and 3.3. Despite this, judging from the preliminary classification results as shown in Appendix A, all the classifiers had no problem effectively classifying WebDDoS attack labels (with TPR over 90% just for WebDDoS), despite having the least number of occurrences in the Consolidated DDoS Data Set (unlike the second least common DDoS attack label, Portmap). Therefore, **WebDDoS attack labels can exist as a class of its own for both Level 1 and Level 2 Grouped Classification.** For Level 1 Grouped Classification**, WebDDoS labels are renamed as Hypertext Transfer Protocol (HTTP),** as from literature, HTTP attacks are an alternative name for WebDDoS attacks. The resulting class distribution of attack labels are as shown in Table 3.4.

Table 3.4: The new labels for DDoS attack classes after Level 1 Grouping.

New Label	Old Label (Number of Instances)	Total Number of Instances
Normal	BENIGN (71 521)	71 521
HTTP	WebDDoS (439)	439
Reflection	MSSQL (29 945) SSDP (7704) NTP (3585) TFTP (58 540) DNS (14 724) LDAP (12 309) NetBIOS (22 433) SNMP (15 448) Portmap (530)	165 218
Exploitation	Syn (17 823) UDP (20 651) UDPLag (1046)	39 520
Total		**276 698**

3.3.3.3 Level 2 Grouped Classification

Going one more level down the hierarchy, it is evident that the **DDoS attacks can be further grouped into the protocols** where the DDoS attack happened, namely **Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and TCP/UDP**. Just like in Section 3.3.3.2, **WebDDoS is put in its own group** as conducted in the Level 1 Grouped Classification but retaining the label of WebDDoS. For TCP and UDP attacks, they can be distinguished between Reflection and Exploitation type attacks, hence can be separated into TCP (Reflection), TCP (Exploitation), UDP (Reflection) and UDP (Exploitation). TCP/UDP attacks, however, are only Reflection attacks (hence labelled as such). The resulting class distribution for Level 2 Grouped Classification is as tabulated in Table 3.5.

Table 3.5: The new labels for DDoS attack classes after Level 2 Grouping.

New Label	Old Label (Number of Instances)	Total Number of Instances
Normal	BENIGN (71 521)	71 521
WebDDoS	WebDDoS (439)	439
TCP (Reflection)	MSSQL (29 945) SSDP (7704)	37 649
TCP (Exploitation)	Syn (17 823)	17 823
UDP (Reflection)	NTP (3585) TFTP (58 540)	62 125
UDP (Exploitation)	UDP (20 651) UDPLag (1046)	21 697
TCP/UDP (Reflection)	DNS (14 724) LDAP (12 309) NetBIOS (22 433) SNMP (15 448) Portmap (530)	65 444
Total		**276 698**

3.4 Software Used to Carry Out Study

The project will largely be **implemented with the help of WEKA**, which is short for Waikato Environment for Knowledge Analysis. WEKA is a freely available data mining software developed by the University of Waikato, New Zealand. WEKA is pre-programmed with numerous machine learning algorithms pre-programmed as followed:

(i) 100+ algorithms for classification.
(ii) 75 for data processing.
(iii) 25 to assist with feature solution.
(iv) 20 for clustering, finding association rules, etc.

(Witten, 2014)

For data pre-processing, the usage of Microsoft Excel Power Queries proved useful as a user-friendly tool to split all the attack labels in the Consolidated DDoS Data Set and perform under sampling on all attack labels (excluding WebDDoS attack labels).

As the J48 classifier is selected from Preliminary Classification, the parameters of the J48 classifier are listed in Section 3.4.1.

3.4.1 Parameters for J48 Classifier

Besides the default hyperparameters that are set when selecting the classifier, WEKA allows for some tweaking of parameters to try for better detection, the changeable parameters for the J48 classifier are as listed below:

1. seed (-S, default 1): The seed used for randomizing the data when reduced-error pruning is used.
2. unpruned (-U, default False): Whether pruning is performed.
3. confidenceFactor (-C, default 0.25): The confidence factor used for pruning (smaller values incur more pruning).

4. numFolds (-N, default 3): Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the tree. (Can only be changed if -R is True).
5. numDecimalPlaces (-num-decimal-places, default 2): The number of decimal places to be used for the output of numbers in the model.
6. batchSize (-batch-size, default 100): The preferred number of instances to process if batch prediction is being performed. More or fewer instances may be provided, but this gives implementations a chance to specify a preferred batch size.
7. reducedErrorPruning (-R, default False): Whether reduced-error pruning is used instead of C.4.5 pruning.
8. useLaplace (-A, default False): Whether counts at leaves are smoothed based on Laplace.
9. doNotMakeSplitPointActualValue (-doNotMakeSplitPointActualValue, default False): If true, the split point is not relocated to an actual data value. This can yield substantial speed-ups for large datasets with numeric attributes.
10. debug (-D, default False): If set to true, classifier may output additional info to the console.
11. subtreeRaising (-S, default True): Whether to consider the subtree raising operation when pruning.
12. saveInstanceData (-L, default False): Whether to save the training data for visualization.
13. binarySplits (-B, default False): Whether to use binary splits on nominal attributes when building the trees.
14. doNotCheckCapabilities (-do-not-check-capabilities, default False): If set, classifier capabilities are not checked before classifier is built (Use with caution to reduce runtime).
15. minNumObj (-M, default 2): The minimum number of instances per leaf.
16. useMDLcorrection (-J, default True): Whether MDL correction is used when finding splits on numeric attributes.
17. collapseTree (-O, default True): Whether parts are removed that do not reduce training error.

(University of Waikato, 2022)

From the list of parameters, the parameters that are of interest to be tested in the Experimenter Module in the WEKA GUI are **confidenceFactor (-C), and minNumObj (-M).** The classification is first optimised with C, followed by M.

3.4.2 Testing Different Values of confidenceFactor (-C) and minNumObj (-M) in WEKA

For both Preliminary and Hierarchical Grouped Optimisation, the Experimenter Module in WEKA is used to perform testing on different values of parameters C and M from the J48 classifier. **For every different parameter value(s) tested, five repetitions of 10-fold cross validations and 0.05 confidence T-testing** is used to obtain the average weighted average values for Accuracy (ACC), True Positive Rate (TPR), Precision (PREC), F-Measure (F1), Area Under ROC Curve (AUC) and True Negative Rate (TNR). The Explorer Module is then used to obtain TPR, PREC, F1, AUC and False Positive Rate (FPR) for Individual Classes.

The value of C is first tested at C = 0.1, 0.2, 0.3, 0.4 and 0.5 to test values of ACC, TPR, PREC, F1, AUC and TNR at each value of C. Should any of the values give higher values of ACC, TPR, PREC, F1, AUC and/or TNR than the default value of 0.25, the **adjacent values of C (in 2 decimal places) are tested to see which values of C give equal or higher values of the evaluation metric in question.** To consider a value of C to use to optimise on M. To do so, **a new measure called Net Change is defined** as the sum of differences of the selected metric x (which is any one of TPR, PREC, F1, AUC, GMEAN) between the tested value of C, with the default value of C = 0.25, across all classes. The formula for Net Change for evaluation metric x, is defined in Equation 3.1.

$$Net\ Change(x, C) = \sum_{All\ classes} x(C) - x(C = 0.25) = \sum_{All\ classes} x(C) - \sum_{All\ classes} x(C = 0.25) \tag{3.1}$$

where x = TPR, PREC, F1, AUC, GMEAN, and $Net\ Change(x, C = 0.25) \equiv 0$.

The Net Change values for each evaluation metric used in the study are calculated from every value of C tested. The selected value of C to optimise with M is the **one with most positive and/or least negative net changes of evaluation metrics.**

After optimising with C, the values of M are tested at M = 1, 3, 4 and 5 and optimised using the same method using the default value of M = 2 for the net change formula for evaluation metric x as shown in equation 3.2.

$$Net\ Change(x, M) = \sum_{All\ classes} x(M) - x(M = 2) = \sum_{All\ classes} x(M) - \sum_{All\ classes} x(M = 2) \tag{3.2}$$

Similarly with C, and $Net\ Change(x, M = 2) \equiv 0$, the **value of M with the most positive and/or least negative net changes of evaluation metrics is chosen** to obtain the best performance of the J48 classifier.

3.5 Summary

A complete, detailed methodology for the study is outlined and thoroughly explained, which encompasses two semesters, utilising data-level methods (namely under sampling and hierarchical grouping) to combat unbalanced data sets and their problems that were highlighted by preliminary classification. Utilising the hierarchical structure of the DDoS attacks, three levels of grouping were implemented to compare the results from each level of grouping with the ungrouped results. For both Preliminary and Hierarchical Grouped Classification, the Net Change value for every evaluation metric used is utilised to pick the best performing values of parameters C and M of the J48 Classifier. The complete results from both Preliminary and Hierarchical Grouped Classification and Optimisation are investigated and discussed in Chapter 4 and supported by Appendix A and B.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

The results from classification and optimisation for the preliminary and hierarchical grouped classification are generated and saved into text files, which contains the models and results buffers built for each classifier under 10-fold cross validation, the confusion matrices and metrics generated for each classifier used, as shown in Appendix A. The **results for preliminary classification and optimisation of the combined sample file are then summarised in Section 4.2. Section 4.3 summarises the results for hierarchical grouped classification and optimisation using the J48 classifier,** based on the top three levels of the hierarchy in Figure 3.2. **Section 4.4 and 4.5 thoroughly discusses the results of classification and optimisation for preliminary and hierarchical classification respectively**.

4.2 Preliminary Classification Results (3% Under Sampling on All Classes except WebDDoS using Default Parameters on 10-fold Cross Validation)

4.2.1 Preliminary Classification using all Classifiers on Default Parameters

Table 4.1: General Evaluation Metrics obtained from WEKA for Preliminary Classification on Sampled Data Set using default parameters (K = 10 for KNN) under 10-fold Cross Validation.

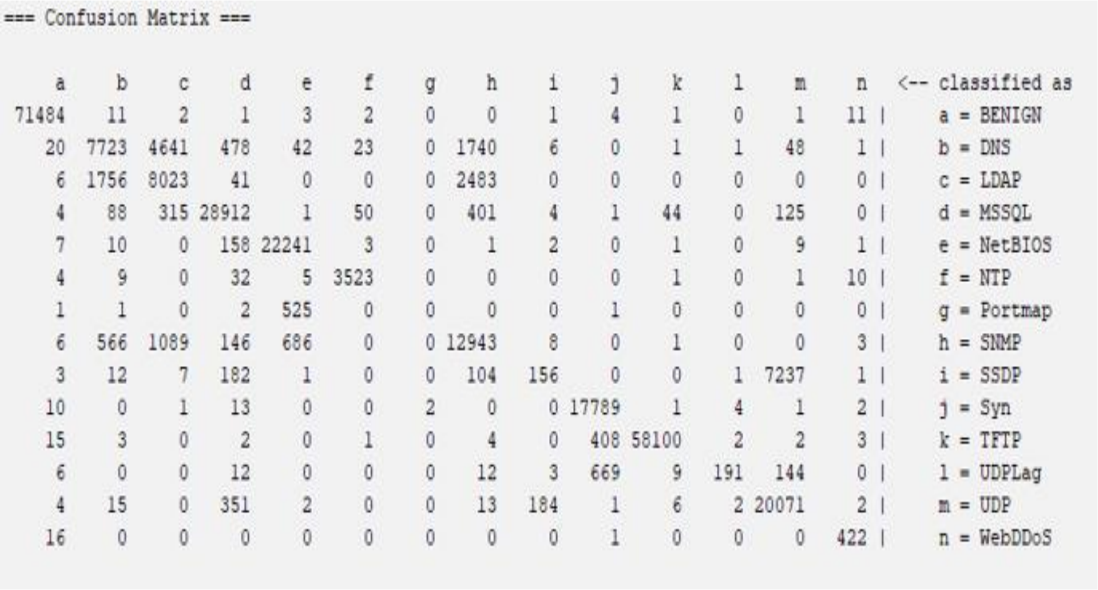
Classifier	Evaluation Metric (Weighted Average)							
	ACC (%)	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
NB	49.48	49.5	75.3	0.530	0.959	2.6	97.4	0.694 356
RF	90.59	90.6	89.9	0.898	0.992	0.6	99.4	0.948 981
J48	90.92	90.9	90.0	0.896	0.993	0.6	99.4	0.950 550
JRip	90.14	90.1			0.990	0.0	100.0	0.949 210
KNN/IBk	90.35	90.3			0.992	0.7	99.3	0.946 931

Note: Black cells denote undefined due to Portmap attack labels not able to be classified by JRip and KNN/IBk (see FigureA-4 and FigureA-5 in Appendix A).

From Table 4.1, it is evident that **J48 is the clear winner in terms of ACC, TPR, PREC, F1, AUC and GMEAN**. For JRip and KNN/IBk, the overall PREC and F1 were undefined due to the inability of the classifiers to classify Portmap instances as seen in FigureA-4 and FigureA-5 in Appendix A, despite JRip having FPR of 0.0%. Going into the detailed Evaluation Metrics for each class, for J48, the detailed evaluation metrics (TPR, PREC, F1, AUC and GMEAN), broken down by label, together with the Confusion Matrix generated in WEKA are tabulated as shown in Table 4.2 and Figure 4.1 respectively, then summarised for comparison with later classifications as in Table 4.3.

Table 4.2: Detailed Evaluation Metrics for each class in Preliminary Classification using J48 Classifier with default parameters under 10-fold Cross Validation

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
BENIGN	100.0	99.8	0.999	1.000	0.1	99.9	0.999 500
DNS	52.4	75.8	0.620	0.972	0.9	99.1	0.720 614
LDAP	65.2	57.0	0.608	0.972	60.8	39.2	0.505 553
MSSQL	96.6	95.3	0.959	0.995	0.6	99.4	0.979 900
NetBIOS	99.1	94.6	0.968	0.996	0.5	99.5	0.992 998
NTP	98.2	97.6	0.979	0.998	0.0	100.0	0.990 959
Portmap	0.0	0.0	0.000	0.959	0.0	100.0	0.000 000
SNMP	83.9	73.1	0.781	0.980	1.8	98.2	0.907 688
SSDP	2.0	43.1	0.038	0.959	0.1	99.9	0.141 351
SYN	99.8	94.3	0.970	0.999	0.4	99.6	0.996 999
TFTP	99.2	99.9	0.996	1.000	0.0	100.0	0.995 992
UDPLag	18.3	94.6	0.306	0.973	0.0	100.0	0.427 785
UDP	97.2	72.6	0.831	0.985	3.0	97.0	0.970 999
WebDDoS	95.4	93.9	0.947	0.988	0.0	100.0	0.976 729
Weighted Average	90.9	90.0	0.896	0.993	0.6	99.4	0.950 550
Accuracy							**90.9215%**



```
=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71484    11     2     1     3     2     0     0     1     4     1     0     1    11 |     a = BENIGN
    20  7723  4641   478    42    23     0  1740     6     0     1     1    48     1 |     b = DNS
     6  1756  8023    41     0     0     0  2483     0     0     0     0     0     0 |     c = LDAP
     4    88   315 28912     1    50     0   401     4     1    44     0   125     0 |     d = MSSQL
     7    10     0   158 22241     3     0     1     2     0     1     0     9     1 |     e = NetBIOS
     4     9     0    32     5  3523     0     0     0     0     1     0     1    10 |     f = NTP
     1     1     0     2   525     0     0     0     0     1     0     0     0     0 |     g = Portmap
     6   566  1089   146   686     0     0 12943     8     0     1     0     0     3 |     h = SNMP
     3    12     7   182     1     0     0   104   156     0     0     1  7237     1 |     i = SSDP
    10     0     1    13     0     0     2     0     0 17789     1     4     1     2 |     j = Syn
    15     3     0     2     0     1     0     4     0   408 58100     2     2     3 |     k = TFTP
     6     0     0    12     0     0     0    12     3   669     9   191   144     0 |     l = UDPLag
     4    15     0   351     2     0     0    13   184     1     6     2 20071     2 |     m = UDP
    16     0     0     0     0     0     0     0     0     1     0     0     0   422 |     n = WebDDoS
```

Figure 4.1: Confusion Matrix generated in WEKA for Preliminary Classification using J48 Classifier with default parameters at 10-fold Cross Validation

Table 4.3: Summarised Confusion Matrix generated in WEKA for Preliminary Ungrouped Classification (using default parameters) from Figure 4.1 after accounting for 24 981 DDoS attack labels incorrectly classified as other DDoS attack labels and the 37 BENIGN labels classified as DDoS attack classes for comparison in performance with Level 0 and 2 Grouped Classification.

Actual/Predicted	BENIGN	DDoS (Any class)
BENIGN	71 484	37
DDoS (Any class)	102	205 075

The observations on preliminary classification are described in Section 4.2.2.

4.2.2 Observation on Preliminary Classification using Default Parameters

From Table 4.1, **the unbalanced sample data set used results in two of the evaluation metrics used (namely precision/PREC and F1-measure/F1) to be undefined**, as looking at FigureA-4 and FigureA-5 in Appendix A, none of the 530 instances of Portmap attack labels have been correctly classified. In contrast, **the classifiers performed worse on Portmap attack labels than they did with WebDDoS attack labels** (with over 90% TPR for just WebDDoS), which is a surprising find, given that there were only 439 WebDDoS attack labels in the sampled data set, which is roughly the same number as Portmap instances in the sampled data set. Even taking J48 as an example, as shown in Figure 4.1, that out of the 530 Portmap labels, none of them were correctly classified (with 525 of them being classified as NetBIOS attacks, two of them classified as MSSQL attacks and one more as SYN attack), as compared to 422 out of 439 WebDDoS attack labels correctly classified, resulting in a 95.4% TPR for WebDDoS as shown in Table 4.2. This **highlights a problem of unbalanced classes resulting in minority classes being mistaken as more popular classes.** This is supported further with the observation that out of 14 724 DNS attack labels in the Consolidated DDoS Data Set, 4641 were wrongly classified as LDAP, 1740 as SNMP and 478 as MSSQL, not to mention that out of 7704 SSDP attack labels in the data set, a whopping 7237 were incorrectly classified as UDP attacks. Similarly, out of 1046 UDPLag attack labels, 669 of them were classified as SYN attacks.

From Table 4.2, excluding BENIGN, **J48 had very good TPR values for SYN (99.8%), TFTP (99.2%), NetBIOS (99.1%), NTP (98.2%), UDP (97.2%), MSSQL (96.6%) and WebDDoS (95.4%),** while only moderately good with SNMP (83.9%). J48 had poor TPR for LDAP (65.2%) and DNS (52.4%), and very poor TPR for UDPLag (18.3%), SSDP (2.0%) and finally Portmap (0.0%), with none of the Portmap labels correctly classified (with a majority of Portmap labels classified as NetBIOS).

As shown in Figure A-2, **Random Forest (RF) was the only classifier to be able to correctly classify a single Portmap instance** out of 530 instances of Portmap attack labels in 10-fold cross validation.

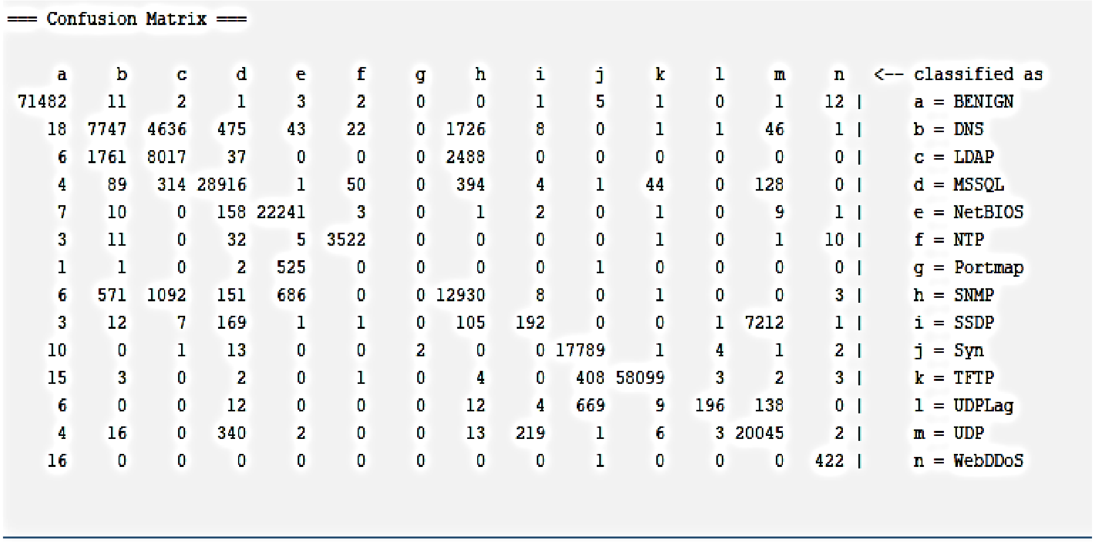
From Table 4.1, **NB performed worse than J48 and RF in terms of Accuracy (ACC), True Positive Rate (TPR), precision (PREC), F1-measure (F1), Area Under ROC curve (AUC), and G-Mean (GMEAN), with J48 ranking highest in all evaluation metrics used**, with RF. The position of JRip and KNN/IBk, are inconclusive as these two classifiers were unable to have a defined PREC and F1, despite having a comparable ACC, TPR, AUC and GMEAN.

Based on the results in Table 4.1, **the J48 classifier is selected to optimise** on using the parameters of Confidence Factor (C) and Minimum Number of Objects (M). Section 4.2.3 and 4.2.4 summarises the results from preliminary optimisation using the J48 classifier, along with summarised confusion matrices for classifying any DDoS attack for comparison with hierarchical classification later on.

4.2.3 Preliminary Optimisation on Ungrouped Data Set using J48 Classifier in WEKA

Table 4.4: Detailed Evaluation Metric table for Individual Class using Best Performing Parameter for J48 Classifier in Preliminary Ungrouped Classification under 10-fold Cross Validation using WEKA.

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
BENIGN	99.9	99.9	0.999	1.000	0.0	100.0	0.999 500
DNS	52.6	75.7	0.621	0.972	0.9	99.1	0.866 133
LDAP	65.1	57.0	0.608	0.972	2.3	97.7	0.746 251
MSSQL	96.6	95.4	0.960	0.995	0.6	99.4	0.979 900
NetBIOS	99.1	94.6	0.968	0.996	0.5	99.5	0.992 998
NTP	98.2	97.8	0.980	0.998	0.0	100.0	0.990 959
Portmap	0.0	0.0	0.000	0.959	0.0	100.0	0.000 000
SNMP	83.7	73.2	0.781	0.980	1.8	98.2	0.906 606
SSDP	2.5	43.8	0.047	0.961	0.1	99.9	0.158 035
SYN	99.8	94.2	0.969	0.999	0.4	99.6	0.996 999
TFTP	99.2	99.9	0.996	1.000	0.0	100.0	0.995 992
UDPLag	18.7	94.2	0.313	0.974	0.0	100.0	0.432 435
UDP	97.1	72.7	0.831	0.985	2.9	97.1	0.971 000
WebDDoS	96.1	92.3	0.942	0.991	0.0	100.0	0.980 306
Weighted Average	90.9	90.1	0.896	0.993	0.6	99.4	0.950 550
Accuracy							**90.9215%**



```
=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71482    11     2     1     3     2     0     0     1     5     1     0     1    12 |     a = BENIGN
    18  7747  4636   475    43    22     0  1726     8     0     1     1    46     1 |     b = DNS
     6  1761  8017    37     0     0     0  2488     0     0     0     0     0     0 |     c = LDAP
     4    89   314 28916     1    50     0   394     4     1    44     0   128     0 |     d = MSSQL
     7    10     0   158 22241     3     0     1     2     0     1     0     9     1 |     e = NetBIOS
     3    11     0    32     5  3522     0     0     0     0     1     0     1    10 |     f = NTP
     1     1     0     2   525     0     0     0     0     1     0     0     0     0 |     g = Portmap
     6   571  1092   151   686     0     0 12930     8     0     1     0     0     3 |     h = SNMP
     3    12     7   169     1     1     0   105   192     0     0     1  7212     1 |     i = SSDP
    10     0     1    13     0     0     2     0     0 17789     1     4     1     2 |     j = Syn
    15     3     0     2     0     1     0     4     0   408 58099     3     2     3 |     k = TFTP
     6     0     0    12     0     0     0    12     4   669     9   196   138     0 |     l = UDPLag
     4    16     0   340     2     0     0    13   219     1     6     3 20045     2 |     m = UDP
    16     0     0     0     0     0     0     0     0     1     0     0     0   422 |     n = WebDDoS
```

Figure 4.2: Confusion Matrix generated in WEKA for Preliminary Classification using best performing parameters (C = 0.29, M = 2) of J48 Classifier under 10-fold Cross Validation.

Table 4.5: Summarised Confusion Matrix generated in WEKA for Preliminary Ungrouped Classification (with C = 0.29, M = 2) from Figure 4.2 after accounting for 24 962 DDoS attack labels incorrectly classified as other DDoS attack labels and the 39 BENIGN labels classified as DDoS attack classes.

Actual/Predicted	BENIGN	DDoS (Any class)
BENIGN	71 482	39
DDoS (Any class)	99	205 078

The observations on preliminary optimisation are described in Section 4.2.4.

4.2.4 Observation on Preliminary Optimisation

Based on the results in Tables 4.4 and 4.5, when comparing to using the default parameters in Tables 4.2 and 4.3, while there are indeed some improvements in terms of ACC, TPR, PREC, F1, AUC and GMEAN (for both weighted average and for individual classes), the **improvements are marginal, still highlighting the problems of unbalanced data sets in classification.**

Supported further by the results collected in Tables A-1 to A-8 (and illustrated in Figures A-6 to A-12) in Appendix A, **mixed results on performance can be seen when trying different values of C** to perform classification. Taking TPR as an example, from Table A-2, classification of SNMP and UDP attack labels perform better in terms of TPR at values of C lower than default value of 0.25), while classification of DNS, LDAP, UDPLag and WebDDos overall perform better at higher values of C than C = 0.25 (except for LDAP performing worse at C = 0.28, 0.29 and 0.3). However, the TPR values is still only marginally better than the TPR values obtained from using C = 0.25. The biggest improvement is however, the TPR value for SSDP attack class, from having TPR of 2.0% at C = 0.25, to TPR of 4.2% at C = 0.5. At values of C lower than 0.25, the increase in TPR values for SNMP and UDP attack labels are outweighed by the decrease in TPR values for DNS, LDAP, MSSQL, NTP, SSDP and UDPLag classes (with TPR for SSDP labels going as low as 0.5% at C = 0.1), which explains mostly negative net change for TPR values for values of C less than 0.25 tested. For values of C more than 0.25, the increase in TPR values for classification of DNS, UDPLag and WebDDoS attack labels outweighed the decrease in TPR values for MSSQL, SNMP and UDP attack classes (resulting in positive net change value for C more than 0.25). The same phenomenon can also be seen for F1 (Table A-4 ad Figure A-9) and GMEAN (Table A-7 and Figure A-12). For PREC and AUC, only a handful of values of C actually resulted in positive net change in values (C = 0.22 and 0.23 for PREC, C = 0.2, 0.3 and 0.5 for AUC), and even with these values of C used, they are outweighed by negative net changes in TPR, F1 and/or GMEAN values for the same value of C used. The value of C = 0.29 was selected as the value for ACC was also still the same as C = 0.25 at 90.93%, with most positive net changes (and no negative net changes), as evident in Table A-8.

While optimising for M at C = 0.29, the same cannot be said in terms of TPR at M =1, as compared to M = 3, 4 and 5. As shown in Table A-10, while TPR values have indeed increased for NetBIOS, NTP, UDP and WebDDoS at M = 1, the increase in TPR values were marginal, and went to classes that already had very high TPR (over 95% TPR), while TPR values for MSSQL and SSDP labels dropped slightly. From Table A-11 and A-12, for M = 4 and 5, the J48 classifier was unable to produce values for PREC and F1 (due to no PREC and F1 values obtained for Portmap labels), hence cannot be selected despite notable increases in AUC across multiple classes. Even the next best value used, M = 3, while also having notable increase in AUC values, fell short of the default value of M = 2 by negative net changes in TPR, PREC, F1 and GMEAN as shown in Table A-16. Therefore, **the default value of M = 2 is used as the best performing value of M to be used (along with C = 0.29) as the best performing parameter of the J48 Classifier in Preliminary Classification and Optimisation.** In Section 4.3, the results from hierarchical grouping will be presented and interpreted.

4.3 Hierarchical Grouped Classification Results (Classification and Optimisation)

4.3.1 Level 0 Grouped Classification

In Level 0 Grouped Classification, **all DDoS attack classes are grouped into one single DDoS group**, while BENIGN instances are relabelled as Normal. The results of Level 0 Grouped Classification, along with the Confusion Matrix created are tabulated in Tables 4.6 and 4.7.

Table 4.6: Detailed Evaluation Metric table for Individual Class using Best Performing Parameter for J48 Classifier in Level 0 Grouped Classification under 10-fold Cross Validation using WEKA.

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
Normal/ BENIGN	99.9	99.9	0.999	1.000	0.0	100.0	0.999 500
DDoS	100.0	100.0	1.000	1.000	0.1	99.9	0.999 500
Weighted Average	100.0	100.0	1.000	1.000	0.1	99.9	0.999 500
Accuracy							**99.9606%**

Table 4.7: (Summarised) Confusion Matrix generated in WEKA for Level 0 Grouped Classification using best performing parameters from J48 Classifier at 10-fold Cross Validation.

Actual/Predicted	Normal/BENIGN	DDoS
Normal/BENIGN	71 477	44
DDoS	65	205 112

4.3.2 Level 1 Grouped Classification

In Level 1 Grouped Classification**, DDoS attack classes are grouped whether if they are Reflection or Exploitation attacks** based on the hierarchy in Figure 3.3, while **WebDDoS attacks are relabelled HTTP/WebDDoS**, and BENIGN instances are relabelled as Normal. The results of Level 1 Grouped Classification, along with the Confusion Matrix created are tabulated in Tables 4.8 and 4.9, then summarised for comparison with later classifications as in Table 4.10.

Table 4.8: Detailed Evaluation Metric table for Individual Class with C = 0.23, M = 3 for J48 Classifier in Level 1 Grouped Classification under 10-fold Cross Validation using WEKA Explorer Module.

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
Normal/ BENIGN	99.9	99.9	0.999	1.000	0.0	100.0	0.999 500
Reflection	95.1	98.3	0.973	0.994	0.6	99.4	0.972 262
Exploitation	98.3	83.0	0.900	0.989	3.4	96.6	0.974 463
HTTP/ WebDDoS	96.1	92.5	0.943	0.994	0.0	100.0	0.980 306
Weighted Average	96.8	97.3	0.969	0.995	0.8	99.2	0.979 927
Accuracy							**96.846%**

Table 4.9: Confusion Matrix generated in WEKA for Level 1 Grouped Classification using best performing parameters (C = 0.23, M = 3) from J48 Classifier at 10-fold Cross Validation.

Actual/Predicted	Normal/ BENIGN	Reflection	Exploitation	HTTP/ WebDDoS
Normal/ BENIGN	71 479	27	5	10
Reflection	51	157 155	7993	19
Exploitation	19	582	38 915	4
HTTP/ WebDDoS	17	0	0	422

Table 4.10: Summarised Confusion Matrix generated in WEKA for Level 1 Grouped Classification using best performing parameters from J48 Classifier at 10-fold Cross Validation, accounting for 8598 DDoS attack instances classified as other DDoS attack types.

Actual/Predicted	Normal/BENIGN	DDoS (Any Class)
Normal/BENIGN	71 479	42
DDoS	87	205 090

4.3.3 Level 2 Grouped Classification

In Level 2 Grouped Classification, **DDoS attack classes are grouped whether if they are TCP (Reflection), TCP (Exploitation), UDP (Reflection), UDP (Exploitation) or TCP/UDP (Reflection) attacks** based on the hierarchy in Figure 3.3, while WebDDoS attacks are put into a group on its own, and BENIGN instances are relabelled as Normal. The results of Level 2 Grouped Classification, along with the Confusion Matrix created are tabulated in Tables 4.11 and 4.13, then summarised for comparison with later classifications as in Table 4.12. The observations for all of Level 0, 1 and 2 Grouped Classifications are reported in Section 4.3.4.

Table 4.11: Detailed Evaluation Metric table for Individual Class using Best Performing Parameter for J48 Classifier in Level 2 Grouped Classification under 10-fold Cross Validation using WEKA Explorer Module.

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
Normal/ BENIGN	100.0	99.9	0.999	1.000	0.0	100.0	0.999 500
TCP/UDP (Reflection)	98.5	98.5	0.985	0.981	0.5	99.5	0.989 987
TCP (Reflection)	77.4	95.8	0.856	0.984	0.1	99.9	0.978 285
UDP (Reflection)	99.2	99.8	0.995	0.999	0.1	99.9	0.995 494
TCP (Exploitation)	99.8	94.3	0.970	0.999	0.4	99.6	0.996 999
UDP (Exploitation)	94.6	73.1	0.825	0.983	3.0	97.0	0.957 925
WebDDoS	96.6	92.8	0.946	0.993	0.0	100.0	0.963 328
Weighted Average	96.0	96.5	0.960	0.996	0.5	99.5	0.979 885
Accuracy							**95.9544%**

Table 4.12: Summarised Confusion Matrix generated in WEKA for Level 2 Grouped Classification using best performing parameters from J48 Classifier at 10-fold Cross Validation, accounting for 10 644 DDoS attack instances classified as other DDoS attack types.

Actual/Predicted	Normal/BENIGN	DDoS (Any Class)
Normal/BENIGN	71 490	31
DDoS (Any Class)	95	204 658

Table 4.13: Confusion Matrix generated in WEKA for Level 2 Grouped Classification using best performing parameters from J48 Classifier at 10-fold Cross Validation.

Actual/Predicted	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	WebDDoS
Normal/ BENIGN	71 490	15	2	1	1	2	10
TCP/UDP (Reflection)	34	64 493	816	27	1	68	5
TCP (Reflection)	6	900	29 151	108	0	7483	1
UDP (Reflection)	19	19	34	61 626	410	4	13
TCP (Exploitation)	11	2	9	2	17 787	10	2
UDP (Exploitation)	10	42	426	16	668	20 533	2
WebDDoS	15	0	0	0	0	0	424

4.3.4 Observation on Hierarchical Grouped Classification

4.3.4.1 Level 0 Grouped Classification

In Level 0 Grouped Classification, the default parameters are selected, as **changing the values of C and M do not affect the performance of the J48 Classifier** (at least at the resolution provided by WEKA).

As shown in Tables 4.7 to 4.9, although DDoS attacks are shown to have TPR of 100.0%, upon further inspection of the confusion matrix as shown in Table 4.9, if even one more decimal place is used to represent the answer, the **actual TPR value for DDoS attack class is calculated to be around 99.97%** but rounded up to 100.0% by WEKA. Even so, as compared to preliminary ungrouped classification using default parameters, the **Level 0 Grouped Classification performed only marginally better** with only 205 112 DDoS attack labels correctly classified as DDoS attacks, as compared with 205 075 DDoS attacks successfully classified as DDoS attacks (with TPR of 99.95%) in Preliminary Ungrouped Classification (with default parameters), that is if **including the 24 978 DDoS attack labels incorrectly classified as other DDoS attack labels**, from Table 4.3.

Referring to the results from the as shown in Tables B-1 to B-16, and graphed in Figures B-1 to B-14 in Appendix B, **no changes in ACC, TPR, PREC, F1, AUC and/or GMEAN (for both individual classes and weighted average values) were observed** (at least at the resolution offered by WEKA, which is 2 decimal places in Experimenter module and 3 decimal places in Explorer module), regardless of the values of C and M tested, which justifies **using the default parameters as best performing settings of the J48 Classifier**.

4.3.4.2 Level 1 Grouped Classification

Going down one level of the hierarchy defined in Figure 3.3, it is evident as shown in Tables 4.10 to 4.12, that although not as good as in Level 0 Grouped Classification in terms of TPR values for DDoS attacks, **Level 1 Grouped Classification still did pretty well, with over 95% TPR for all DDoS attack class,** with the best performing attack type being Exploitation type attacks reporting 98.3% TPR.

Comparing the overall summarised confusion matrix in Table 4.13, with the summarised confusion matrix for Ungrouped Classification using the same parameters of J48 Classifier in Table C-2, **in Level 1 Grouped Classification, the J48 Classifier did marginally better** with classifying 205 090 DDoS attacks (TPR of 99.9576%) as DDoS attacks (any class), as compared to 205 074 DDoS attacks (TPR of 99.9498%) classified as DDoS attacks (any class) for Ungrouped Classification using the same parameters.

Despite a very good TPR of 98.3%, **Exploitation type attacks also have the highest False Positive Rate (FPR) among all classes**, with 3.4% FPR, as compared to Reflection type attacks with only 0.6% FPR, and basically 0.0% FPR for Normal/BENIGN and HTTP/WebDDoS attack labels, not to mention that Exploitation type attacks have also the lowest precision among all classes at 83.0%, and hence the lowest F1 value of 0.900.

Going through the results from optimisation using C, as shown in Tables B-17 to B-24 and also graphed in Figures B-15 to B-21 in Appendix B, from Table B-17 and Figure B-15, it is evident that ACC does go up slightly from 96.84% at C = 0.25, to 96.85% at C = 0.2 and 0.21, albeit a slight drop in AUC value from 1.00 to 0.99. At the same time, from Table B-18 and Figure B-16, the **TPR values for Exploitation type attacks actually go up for values of C smaller than 0.25 (peaking at 98.6% TPR at C = 0.1)** and go down for values of C larger than 0.5 (going down to 97.7% TPR at C = 0.5). For HTTP/WebDDoS attack types, TPR values do not go over 96.1%, and for some values of C tested, actually dipped, especially at C = 0.1, 0.21 and 0.22, when the TPR for HTTP/WebDDoS attacks was 95.4%. From Table B-19 and Figure B-17, PREC values for Exploitation attack increase slightly from C = 0.28 onwards, reaching PREC of 83.3% at C = 0.4 and 0.5. As shown in Table B-20 and Figure B-18, for Exploitation and HTTP/WebDDoS attacks, F1 values generally decrease with increasing value of C, with the sharpest drop being that of HTTP/WebDDoS attacks from 0.954 at C = 0.1 to 0.942 at C = 0.2. Table B-21 and Figure B-19 shows generally higher values for AUC for both Exploitation and Reflection attacks at higher values of C, while WebDDoS attacks experiences the lowest AUC value of 0.988 at C = 0.21 and 0.5. From Table B-22 and Figure B-20, Exploitation attacks still have the highest

FPR among all classes with even the lowest FPR value for Exploitation attacks being 3.3% at all higher values of C than 0.25. From Table B-23 and Figure B-21, GMEAN values for Exploitation and Reflection type attacks peak at lower values of C than 0.25, with the highest observed values being 0.9728 for Reflection attacks and 0.9759 for Exploitation attacks at C = 0.5. From Table B-24, the value **of C = 0.23 was selected to continue optimising with M**, as it had the most positive and least negative net changes.

Now dealing with M, looking at the results tabulated in Tables B-25 to B-32, and also graphed in Figures B-22 to B-28. Starting with Table B-25 and Figure B-22, the only the weighted average AUC value rises up to 1.00 at M = 3, 4 and 5. In Table B-26 and Figure B-23, **TPR for Exploitation attacks peaks slightly at 98.5% at M = 1, 3 and 4, whereas TPR for HTTP/WebDDoS attacks are the highest at 96.4% at M = 1,** and lowest at 95.7% at M = 5. From Table B-27 and Figure B-24, HTTP/WebDDoS attack labels have a slight increase in PREC values from 92.5% at M = 2 to 92.8% at M = 1, not to mention PREC value of 92.7% at M = 3, 4 and 5. As shown in Table B-28 and Figure B-25, F1 value for Exploitation attacks go up slightly at M = 4, and also for HTTP/WebDDoS attacks at M = 1, 3 and 4. From Table B-29 and Figure B-26, AUC values for Reflection and HTTP/WebDDoS attacks increase for M larger than 2. From Table B-30 and Figure B-27, Exploitation attacks consistently have the highest FPR values at 3.4% for every value of M tested while for Reflection attacks, FPR drops slightly at M = 1 and 3. Table B-31 and Figure B-28 shows that GMEAN values for Reflection, Exploitation and HTTP/WebDDoS attacks show the most increase (and thus most positive net change) at M = 1. Therefore, judging from all the net changes in Table B-32, it is therefore clear, that the **value of M to use with the most positive net changes is M = 3**.

4.3.4.3 Level 2 Grouped Classification

Going one more level down the hierarchy, it can be seen that the **J48 classifier was still performing well**, with very good (> 90% TPR) TPR values for all DDoS attack classes **except TCP (Reflection) attacks, which has a TPR of 77.4%**, as shown in Table 4.11. However, **UDP (Exploitation),** despite having a very good TPR of 94.6%, lagged behind in terms of PREC (73.1%), not to mention a **very high FPR of 3.0%** as compared to all other classes.

From Tables B-33 to B-48 in Appendix B, and also graphed in Figures B-29 to B-42, it is clear that the **default parameters (C = 0.25 and M = 2) are chosen as the best performing parameters,** as all other values of C and M used resulted in negative net change in TPR, PREC, F1, AUC and GMEAN (with the exception of AUC having positive net change at C = 0.1, and M = 3, 4 and 5) as shown in Tables B-40 and B-48. **For all of TPR, PREC, F1 and GMEAN, any increase in these evaluation metrics for one class is always overshadowed by decrease in the same evaluation metric for other classes.** This is evident, as taking TPR as an example, using Table B-34 and Figure B-30, for every value of C tested, while TPR for TCP/UDP (Reflection) does increase for all values of C other than the default value of C = 0.25 (albeit only an increase from 98.5% to 98.6%), and also TCP (Reflection) showing bigger increase in TPR for C = 0.29, 0.3, 0.4 and 0.5, this increase in TPR is overshadowed by decrease in TPR for UDP (Exploitation) (except for C = 0.1) and WebDDoS. For C = 0.2, 0.21 and 0.22, the decrease in TPR for TCP (Reflection) attacks also overwhelmed the increase in TPR for TCP/UDP (Reflection), UDP (Exploitation) and WebDDoS attacks. This is also reflected in the GMEAN values in Table B-39 and Figure B-35. The only evaluation metrics that actually shows positive net changes is AUC, with Table B-37 and Figure B-33, showing increase in AUC values for mainly TCP (Reflection) for C = 0.21, 0.22, 0.23, 0.24, 0.26, 0.27, 0.29, 0.3, 0.4 and 0.5. This is however, outweighed by the bigger decrease in AUC value for WebDDoS attack class for all the aforementioned values of C. Just about the same scenario can be seen when testing different values of M, with the increase in TPR and GMEAN for TCP/UDP (Reflection) overshadowed by the decrease in the same evaluation metrics for UDP (Exploitation) and WebDDoS, as shown in Tables B-42 and B-47 respectively.

4.3.5 Overall Evaluation Metrics

To summarise, considering all DDoS attack classes misclassified, the overall evaluation metrics for each classification performed in the study are summarised in Table 4.14 and Figure 4.3.

Table 4.14: Overall Evaluation Metrics (TPR, PREC, F1 and GMEAN) for all Classifications in Study. Parameters used are default parameters unless otherwise stated.

	Evaluation Metric (Overall)				
Classification	TPR	PREC	F1	FPR	GMEAN
Preliminary Ungrouped	99.950%	99.982%	0.999661	0.052%	0.999493
Level 0 Grouped	99.952%	99.981%	0.999664	0.055%	0.999486
Level 1 Grouped (C 0.23 M 3)	**99.958%**	**99.980%**	**0.999686**	**0.059%**	**0.999494**
Level 2 Grouped	99.954%	99.985%	0.999693	0.043%	0.999552

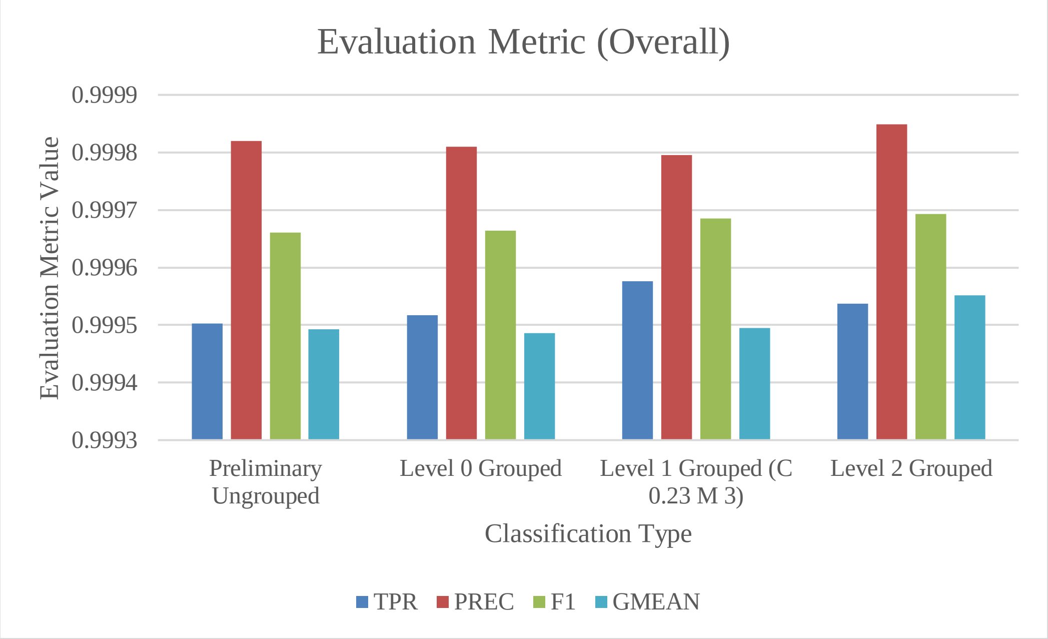


Figure 4.3: Chart of overall evaluation metric values for each classification. Parameters used are default parameters unless otherwise stated.

From Table 4.14, Level 1 Grouped Classification with C = 0.23 and M = 3 performed the best in terms of overall TPR and GMEAN, second to Level 2 Grouped Classification in terms of overall F1 value, lagged behind Preliminary Ungrouped

Classification in terms of overall PREC and have the highest FPR as compared to other classifications in the study.

4.4 Discussion on Preliminary Classification and Optimisation

4.4.1 Preliminary Classification in WEKA using all Classifiers on Default Parameters

From what can be seen in Section 4.2, **the effects of unbalanced data sets can be clearly seen with the inconclusiveness of ranking due to the inability for PREC and F1 to be generated due to the inability of classifiers to classify Portmap instances**. From looking at the TPR values for each class as shown in Table 4.2, with WebDDoS being a complete outlier (due to surprisingly high TPR of 95.4% despite being the least common label in the data set), the low TPR values for LDAP, DNS, UDPLag, SSDP and Portmap by the J48 classifier are caused by the fact that these attack labels are severely underrepresented in the sampled data set.

A plausible explanation for the surprisingly high TPR for WebDDoS labels, is that **WebDDoS attack labels have features that effectively isolate themselves from other classes** so that they do not overlap with other attack classes, making it easy for classifiers to detect and classify them as so, something that Portmap attack instances lack in. This is **similar to the unbalanced classification case as shown in Figure 2.7,** where even though there are disproportionately more circles to crosses, the J48 classifier was still able to effectively classify and create a boundary for the minority class (crosses), as the classes do not overlap, and the classifier was sensitive enough to detect the minority class, only for this case, it is expanded to higher dimensional space (77D hyperspace classification as there are 77 numeric columns and 1 Label column in the sampled data set). For example, by taking a closer look at the decision tree generated by J48 in 10-fold cross validation, the rules for WebDDoS attack labels are shown in lines 48-58 of the decision tree as followed:

Line 48-58 of Decision Tree generated by J48 Classifier

```
|  |  |  |  |  CWE Flag Count > 0
|  |  |  |  |  |  min_seg_size_forward <= 26: BENIGN (423.0)
|  |  |  |  |  |  min_seg_size_forward > 26
|  |  |  |  |  |  |  Init_Win_bytes_forward <= 246
|  |  |  |  |  |  |  |  Init_Win_bytes_backward <= 249
|  |  |  |  |  |  |  |  |  Total Fwd Packets <= 1
|  |  |  |  |  |  |  |  |  |  Init_Win_bytes_forward <= 61: BENIGN (3.0)
|  |  |  |  |  |  |  |  |  |  Init_Win_bytes_forward > 61
|  |  |  |  |  |  |  |  |  |  |  Init_Win_bytes_forward <= 91: WebDDoS (147.0/8.0)
|  |  |  |  |  |  |  |  |  |  |  Init_Win_bytes_forward > 91
|  |  |  |  |  |  |  |  |  |  |  |  Init_Win_bytes_backward <= 228: WebDDoS (152.0/10.0)
```

From what can be seen in the decision tree, the J48 classifier was able to deduce that (with Min Packet Length <= 265, Min Packet Length <= 118, Packet Length Std <= 0, Init_Win_bytes_forward <= 5839, CWE Flag Count > 0 and min_seg_size_forward > 26) WebDDoS attack labels have Init_Win_bytes_forward value of more than 61, but less than or equal to 91 (where 147 instances were correctly classified by this rule), and also Init_Win_bytes_backward less than or equal to 228 (where 152 instances were correctly classified by this rule). Another instance in the decision tree is line 101 to 107 of the decision tree as shown below:

Line 101-107 of Decision Tree generated by J48 Classifier

```
|  |  |  |  Init_Win_bytes_forward > 5840
|  |  |  |  |  Fwd Header Length <= 142: BENIGN (3035.0/12.0)
|  |  |  |  |  Fwd Header Length > 142
|  |  |  |  |  |  Init_Win_bytes_forward <= 64 999: BENIGN (47.0)
|  |  |  |  |  |  Init_Win_bytes_forward > 64 999
|  |  |  |  |  |  |  Init_Win_bytes_backward <= 18 420: BENIGN (5.0)
|  |  |  |  |  |  |  Init_Win_bytes_backward > 18 420: WebDDoS (30.0/1.0)
```

The decision tree generated by J48 classifier also deduced that (with Min Packet Length <= 265, Min Packet Length <= 118, Packet Length Std <= 0 and Fwd

Header Length > 142), WebDDos attack instances have Init_Win_bytes_forward of more than 64 999 and Init_Win_bytes_backward of more than 18 420 (where 30 instances were correctly classified by this rule). In total, **there are four rules that classify WebDDoS attack instances in the decision tree generated by J48**. In contrast, the same decision tree **only generated one rule that classifies Portmap instances,** which is shown in lines 64 to 67 in the decision tree as shown below:

Line 64-67 of Decision Tree generated by J48 Classifier

```
| | | | | | | Init_Win_bytes_forward > 246
| | | | | | | |  Bwd IAT Std <= 361.331 565
| | | | | | | | | Init_Win_bytes_forward <= 247
| | | | | | | | | | Flow Duration <= 398 534: Portmap (2.0/1.0)
```

The decision tree generated by J48 can only deduce that (with Min Packet Length <= 265, Min Packet Length <= 118, Packet Length Std <= 0, Init_Win_bytes_forward <= 5839 and CWE Flag Count > 0) Portmap instances have Init_Win_bytes_forward of more than or equal to 247, Bwd IAT Std of less than 361.331 565 and Flow Duration of less than or equal to 398 534, which only correctly classified two instances.

This is evidence that **WebDDoS attack labels were more able to be correctly classified than Portmap instances**, due to specific features that favoured WebDDoS attack labels to be effectively classified by classifiers, whereas Portmap instances were hard to distinguish from other classes, as they probably overlapped with other attack classes, which is likely the case that Portmap attack labels were basically unable to be detected as such is in the case of classifiers like JRip and KNN/IBk, in which were unable to classify Portmap instances correctly, as is similarly the case as in Figure 2.10, where the classifier was unable to classify minority class instances as they are treated as random noise, only this time in 77D hyperspace instead of 2D space.

4.4.2 Preliminary Optimisation of Ungrouped Classification using J48 Classifier in WEKA

As supported by the results tabulated in Tables A-1 to A-16 (and also graphed in Figures A-1 to A-19) in Appendix A, it is evident that while setting different values of C and M **can improve the performance of evaluation metrics for certain classes** (TPR for DNS, LDAP, NetBIOS, NTP, SNMP, SSDP UDPLag, UDP and WebDDoS, PREC for DNS, LDAP, MSSQL, NTP, SNMP, SSDP, Syn, UDPLag and WebDDoS, F1 for DNS, MSSQL, NTP, SSDP, UDPLag, UDP and WebDDoS, AUC for DNS, LDAP, NetBIOS, NTP, Portmap, SNMP, SSDP, UDPLag, UDP and WebDDoS, and GMEAN for DNS, LDAP, NetBIOS, NTP, SNMP, SSDP, UDPLag, UDP and WebDDoS), the **changes are ever so marginal**, with even the most drastic change of values for evaluation metrics being that of TPR for SSDP attack labels more than doubled from 2.0% at C = 0.25 to 4.0% at C = 0.5 as shown in Table A-2, not to mention a subsequent increase in F1 and GMEAN values for SSDP attack class from 0.039 and 0.1414 respectively at C = 0.25, to 0.077 and 0.2047 respectively at C = 0.5, shown in Tables A-4 and A-7 respectively, as F1 and GMEAN are by definition, directly related to TPR by mathematical formula definition.

Regardless of the parameter settings used, **classes like Portmap, SSDP, UDPLag, DNS and LDAP still show unsatisfactory performance** (especially in terms of TPR), with the fact that **all of the Portmap instances are still unable to be correctly classified** (with TPR of 0.0% all the way), even more so when PREC and F1 were unable to be produced when using M = 4 and 5 (with C = 0.29).

When optimising with M, **increasing the value of M decreases the performance of the J48 Classifier on certain classes**, which is evident with a slight drop of TPR values for SSDP and UDPLag attack classes for M = 3, NTP, SSDP, UDP and WebDDoS attack classes for M = 4, DNS, LDAP, MSSQL, NTP, UDP and WebDDoS attack classes for M = 5 as shown in Table A-10, not to mention a slight drop in accuracy from 90.93% using M = 2 (with C = 0.29), to 90.92% for M = 3 and 4, and 90.90% for M = 5. This is also evident that by increasing the value of M (minimum number of instances per leaf), the **J48 Classifier completely ignores rules that effectively classify certain classes** (most notably those that only correctly classifies number of instances less than the value of M set). Using the decision tree generated by the J48 Classifier using C = 0.29 and M = 2, lines 21 to 44 of the decision tree are shown below, under Min Packet Length <= 118, Packet Length Std <= 0, Init_Win_bytes_forward <= 5839 and min_seg_size_forward > 17,

Line 21-44 of Decision Tree generated by J48 Classifier with C = 0.29, M = 2

```
| | | | | | | | Total Length of Fwd Packets <= 112
| | | | | | | | | Total Length of Fwd Packets <= 94
| | | | | | | | | |  Total Fwd Packets <= 1: BENIGN (6.0)
| | | | | | | | | |  Total Fwd Packets > 1
| | | | | | | | | | | Total Length of Fwd Packets <= 55
| | | | | | | | | | | | Flow Bytes/s <= 17000000: NTP (2.0/1.0)
| | | | | | | | | | | | Flow Bytes/s > 17000000
| | | | | | | | | | | | | Total Length of Fwd Packets <= 52
| | | | | | | | | | | | | |  Fwd Header Length <= 50: DNS (4.0/1.0)
| | | | | | | | | | | | | |  Fwd Header Length > 50: BENIGN (2.0)
| | | | | | | | | | | | | Total Length of Fwd Packets > 52: BENIGN (7.0)
| | | | | | | | | | | Total Length of Fwd Packets > 55
| | | | | | | | | | | |  Fwd Header Length <= 52: DNS (15.0/2.0)
| | | | | | | | | | | |  Fwd Header Length > 52: TFTP (3.0/1.0)
| | | | | | | | | Total Length of Fwd Packets > 94
| | | | | | | | | | Flow Bytes/s <= 33307692.31: BENIGN (2.0)
| | | | | | | | | | Flow Bytes/s > 33307692.31: UDP (4.0/2.0)
| | | | | | | | Total Length of Fwd Packets > 112
| | | | | | | | |  Average Packet Size <= 118.461539: BENIGN (178.0)
| | | | | | | | |  Average Packet Size > 118.461539
| | | | | | | | | |  Total Fwd Packets <= 7
| | | | | | | | | | | Total Length of Fwd Packets <= 171: NTP (6.0/2.0)
| | | | | | | | | | | Total Length of Fwd Packets > 171: DNS (2.0)
| | | | | | | | | |  Total Fwd Packets > 7: BENIGN (5.0)
```

Now to show the same part of the decision tree generated by C = 0.29, M = 5, with lines 6 to 15 of decision tree shown below.

Line 6-15 of Decision Tree generated by J48 Classifier with C = 0.29, M = 5

```
|   |   |   |   |   Flow Bytes/s <= 116541.3534
|   |   |   |   |   |   Bwd Packets/s <= 0.236007: BENIGN (128.0/2.0)
|   |   |   |   |   |   Bwd Packets/s > 0.236007: LDAP (5.0/3.0)
|   |   |   |   |   Total Length of Fwd Packets <= 39: NTP (7.0/1.0)
|   |   |   |   |   |   Total Length of Fwd Packets > 39
|   |   |   |   |   |   |   Flow Bytes/s <= 133666666.7: DNS (171.0/6.0)
|   |   |   |   |   |   |   Flow Bytes/s > 133666666.7
|   |   |   |   |   |   |   |   Total Length of Fwd Packets <= 163: NTP (8.0)
|   |   |   |   |   |   |   |   Total Length of Fwd Packets > 163: DNS (8.0/1.0)
```

Comparing the two separate decision trees generated by different values of M (2 and 5 respectively for C = 0.29), it is clear that **when M = 5, many of the leaves of the decision tree that would otherwise be able to correctly classify certain classes were cut off as the leaves had less than five instances** (hence resulting in a smaller branch of the decision tree created, hence reducing the size of the decision tree when M = 5), such as the case of the leaf Total Length of Fwd Packets <= 94 -> Total Fwd Packets > 1 -> Total Length of Fwd Packets <= 55 -> Flow Bytes/s <= 17000000: NTP, which correctly classified two NTP labels, but was cut off when M = 5, as the leaf only had three instances (two correctly classified instances and one incorrectly classified instance), which is less than five.

4.5 Discussion on Hierarchical Grouped Classification and Optimisation using J48 Classifier in WEKA

4.5.1 Level 0 Grouped Classification

The very high TPR value for DDoS attacks in Level 0 Grouped Classification is most likely due to the fact that in Level 0 Grouped Classification, that unlike in Preliminary Ungrouped Classification, classifying certain DDoS attacks as other DDoS attack classes, in which normally would be treated as an error and not contribute to the Accuracy and TPR value for DDoS attack class, whereas here **in Level 0 Grouped Classification, classifying DDoS attack label as DDoS attack (regardless of which type of DDoS attack class), is treated as effective detection as all DDoS attacks** are grouped into one single DDoS attack group, hence contributes to the Accuracy and TPR of DDoS attack detection.

The only downfall for this type of grouping is **oversimplification of all DDoS attack types**. Overfitting and overgeneralising in classification problems very often cause varying values false positive and false negative rates during classification. (Pham Nguyen & Triantaphyllou, 2007) This is especially true, even with very high ACC and TPR, the J48 Classifier may still perform very badly with previously rare and basically undetectable DDoS attack labels (i.e., Portmap, SSDP, UDPLag), due to their small numbers in the group of DDoS attacks, as even if some more do get detected, the contribution to the overall TPR from these attack types would be negligible. However, this would be impossible to know as in the grouping process, as the original labels had to be removed for effective classification.

4.5.2 Level 1 Grouped Classification

Evidently, the lower PREC and F1 values for Exploitation attack labels as compared to other classes (despite having the highest TPR value) can be attributed to the fact that based on the confusion matrix in Table 4.10, a whopping **7993 Reflection attack labels were incorrectly classified as Exploitation attacks** (as compared to only 582 Exploitation attack labels classified as Reflection attacks), which contributes to the high False Positive Rate (FPR) (and hence lower PREC, F1 and GMEAN) for Exploitation attack class. Upon closer inspection on the decision tree generated by the J48 Classifier using C = 0.23, M = 3, the two rules that contribute most to the astoundingly large number of Reflection attack labels misclassified as Exploitation attacks are shown with lines 124 to 129 (Under Min Packet Length > 118, Fwd Packet Length Std <= 22.366642, Fwd Packet Length Max <= 401 and Fwd Packet Length Max > 320), and in lines 144 to 146 (Under Min Packet Length > 118) of the decision tree generated as shown below.

Line 124 to 129 of Decision Tree generated by J48 Classifier with C = 0.23, M = 3

```
|  |  |  |  Total Length of Fwd Packets > 737
|  |  |  |  |  min_seg_size_forward <= 426
|  |  |  |  |  |  Fwd Header Length <= 670: Exploitation (13277.0/3641.0)
|  |  |  |  |  |  Fwd Header Length > 670
|  |  |  |  |  |  |  min_seg_size_forward <= 350: Reflection (12.0/1.0)
|  |  |  |  |  |  |  min_seg_size_forward > 350: Exploitation (281.0/131.0)
```

Line 144 to 146 of Decision Tree generated by J48 Classifier with C = 0.23, M = 3

```
|  Fwd Packet Length Std > 22.366642
|  |  Packet Length Std <= 34.521008
|  |  |  Fwd Packet Length Mean <= 416: Exploitation (14005.0/3683.0)
```

From the decision tree generated by the J48 Classifier, it is evident that the instances misclassified by rules Min Packet Length > 118 -> Fwd Packet Length Std <= 22.366642 -> Fwd Packet Length Max <= 401 -> Fwd Packet Length -> Total Length of Fwd Packets > 737 -> min_seg_size_forward <= 426 -> Fwd Header Length <= 670: Exploitation, and Min Packet Length > 118 -> Fwd Packet Length Std > 22.366642 -> Packet Length Std <= 34.521008 -> Fwd Packet Length Mean <= 416: Exploitation were the **main contributors of misclassifying Reflection attacks as Exploitation attacks** (each having misclassified over 3000 instances each), as only five Normal/BENIGN instances were misclassified as Exploitation attacks, while none of the HTTP/WebDDoS attack labels were misclassified as such.

This is further supported by the fact that from the confusion matrix generated in Ungrouped Classification using the same parameters in Figure C-1, it is evident that **SSDP labels (a type of Reflection attack label) had a surprising 7285 instances misclassified as UDP labels (an Exploitation attack label).** Therefore, it is plausible that at least a majority of the Reflection attack labels misclassified as Exploitation labels in Level 1 Grouped Classification could very well be the same SSDP attack labels also misclassified as UDP attack labels in Preliminary Ungrouped Classification, making this the most plausible explanation for this misclassification of Reflection attacks as Exploitation attacks.

Meanwhile, the rule that contributes the most to the number of Exploitation attacks wrongly classified as Reflection attacks is likely shown in lines 131 to 134 (under Min Packet Length > 118 and Fwd Packet Length Std <= 22.366642) of the decision tree shown below:

Line 131 to 134 of Decision Tree generated by J48 Classifier

```
|  |   Fwd Packet Length Max > 401
|  |  |   Packet Length Std <= 21.884311
|  |  |  |   Fwd Packet Length Min <= 507
|  |  |  |  |   Flow IAT Mean <= 175.333333: Reflection (18952.0/384.0)
```

From the decision tree, the rule Min Packet Length > 118 -> Fwd Packet Length Std <= 22.366642 -> Fwd Packet Length Max > 401 -> Packet Length Std <= 21.884311 -> Fwd Packet Length Min <= 507 -> Flow IAT Mean <= 175.333333: Reflection is likely responsible for misclassifying over 300 Exploitation attacks as Reflection attacks, due to the fact that overall, only 27 Normal/BENIGN and none of the HTTP/WebDDoS attack labels were misclassified as such.

Looking again at the confusion matrix generated by WEKA using the same parameters of the J48 Classifier on the original ungrouped data set in Figure C-1**, the most significant case of Exploitation attack labels being misclassified as Reflection attack labels would be the 355 UDP labels being misclassified as MSSQL labels.** Coming in second, would be the 111 UDP labels being misclassified as SSDP labels, thus making a total of 466 UDP labels being misclassified as either MSSQL or SSDP labels, both of which are Reflection attacks. This makes the most plausible explanation for at least a majority of the misclassified Exploitation attack instances as Reflection attacks.

Although not as oversimplified as Level 0 Grouped Classification, **Level 1 Grouped Classification still exhibit some of the problems of oversimplification** (though not as much as Level 0 Grouped Classification). This is evident due to the fact that, with the exception of HTTP/WebDDoS attack labels, which has surprisingly high TPR, PREC, F1, AUC and GMEAN, despite being the smallest occurrence of DDoS attack labels with only 439 instances in the data set, as shown in Table 4.9, **Exploitation attack instances are very underrepresented in the data set,** with only 39 520 total instances in the data set. In contrast, there are 165 218 instances of Reflection attack labels in the data set, which is more than four times as many as Exploitation attack instances. This is due to the fact that based on the hierarchy in Figure 3.2, there are in fact, **more types of Reflection attacks as compared to Exploitation attacks,** with nine of the DDoS attack classes from the data set belong to Reflection attack class, whereas only three belong to Exploitation attack class, not to mention that the **more popular attack labels also fall under Reflection class,** while Exploitation attacks consists of not so popular attack labels (UDP and Syn), and one of the rare classes of DDoS attack labels (UDPLag). Even within the groups themselves, the problems of unbalanced problematic classes still show up, with SSDP and Portmap being grouped into Reflection attacks, and UDPLag grouped into Exploitation attacks, showcasing how even these attacks can still be misclassified even when grouped with other attacks.

4.5.3 Level 2 Grouped Classification

The slightly lower TPR for TCP (Reflection) attacks (77.4%) can be explained due to the fact that **almost one-fifth of the class consists of SSDP attacks**, which in the first place, had very low TPR (2.0%) during the Preliminary Ungrouped Classification, which were very likely misclassified during Level 2 Grouped Classification as well, despite being grouped with MSSQL attack labels into one class. From the confusion matrix in Table 4.17, it is clear that while 29 151 out of the 37 649 TCP (Reflection) were successfully classified as such, a staggering 7483 were misclassified as UDP (Exploitation) attacks, more than the next three cases of misclassification of attack classes (900 TCP (Reflection) labels as TCP/UDP (Reflection), 816 TCP/UDP (Reflection) labels as TCP (Reflection) and 668 UDP (Exploitation) labels as TCP (Exploitation) for a total of 2384 instances misclassified) combined. By investigating the decision tree generated by the J48 Classifier, the evidence is shown in lines 197 to 201 of the decision tree (with Min Packet Length > 118, Min Packet Length <= 1280, Total Length of Fwd Packets > 640, Fwd Packet Length Max <= 439, Average Packet Size > 290.363636, Average Packet Size <= 602.179104, Total Length of Fwd Packets > 737 and Flow IAT Mean <= 69893.42857) as shown below.

Line 197-201 of Decision Tree generated by J48 Classifier

```
| | | | | | | | min_seg_size_forward <= 426
| | | | | | | | | Total Length of Fwd Packets <= 801
| | | | | | | | | | Total Length of Fwd Packets <= 799: UDP (Exploitation) (8976.0/2441.0)
| | | | | | | | | | Total Length of Fwd Packets > 799: TCP (Reflection) (98.0/3.0)
| | | | | | | | | Total Length of Fwd Packets > 801: UDP (Exploitation) (18546.0/4953.0)
```

From the snippet of the decision tree above, it is clear that the rules that imply Total Length of Fwd Packets <= 799 and Total Length of Fwd Packets > 801 are UDP (Exploitation) attacks, while were able to correctly classify 8976 and 18 546 UDP (Exploitation) attack instances respectively, at the same time, **thousands of TCP (Reflection) attacks were likely mixed into each group,** which is very likely the cause of the lower TPR for TCP (Reflection) attacks.

Just like in Level 1 Grouped Classification, this can be further supported due to the fact that back in Preliminary Ungrouped Classification, the confusion matrix in Figure 4.1 also shows 7237 SSDP labels (a type of TCP (Reflection) attack) being misclassified as UDP labels (a type of UDP (Exploitation) attack), making at least the **majority of the 7483 TCP (Reflection) attack labels misclassified as UDP (Exploitation) attacks, plausibly the very same SSDP labels misclassified as UDP labels.**

For the next two cases of misclassification, the **900 TCP (Reflection) attacks misclassified as TCP/UDP (Reflection) attacks** are most likely the combined total of **716 MSSQL labels (a TCP (Reflection) attack) misclassified as either SNMP or LDAP attacks** (both TCP/UDP (Reflection) attack types) from Preliminary Ungrouped Classification, while the **816 TCP/UDP (Reflection) misclassified as TCP (Reflection) attacks** are most likely the **478 DNS or 146 SNMP attack labels** (both from TCP/UDP (Reflection) group) **misclassified as MSSQL labels** in Preliminary Classification as shown in the confusion matrix in Figure 4.1.

4.6 Implications

From what can be seen, while even with the best performing parameters (C = 0.29, M = 2), **ungrouped classification and optimisation still cannot solve the problems that appear with unbalanced data sets**, especially low TPR from rare classes, as is evident with TPR values for classes like Portmap, SSDP and UDPLag being very unsatisfactory no matter the values of C and M tested, having only marginal changes at different values of C (with even the biggest change in TPR being the TPR for SSDP classes increase from 2.0% at C = 0.25 to 4.2% at C = 0.5).

On the other hand, hierarchical grouped classification, while being able to increase overall performance by TPR (albeit only marginally), did so by **including misclassification of DDoS attacks as other DDoS attacks in the calculation of TPR**, which would normally not be done in ungrouped classification by WEKA. Depending on the level of the hierarchy to perform grouping of DDoS attacks (especially true for Level 0 and 1 Grouped Classification), **the problem of oversimplification can arise, while problems that arise from classifying unbalanced data sets do not completely disappear** (even so making it harder to find out). For Level 1 and 2 Grouped Classification, misclassification still happens, especially if misclassification of DDoS attacks happened within between the same groups in ungrouped classification using the same parameters (default or otherwise), like between Exploitation and Reflection attacks for Level 1 Grouped Classification, and between TCP (Reflection), UDP (Exploitation) and TCP/UDP (Reflection) attacks in Level 2 Grouped Classification, all contributing to lower (and in some cases, missing) PREC values and higher FPR values.

Ultimately, from seeing all the results in preliminary and hierarchical grouped classification, **it is still up to the administrator in setting the sensitivity of the model to detect DDoS attacks** by values of C and M, or **perform the level of grouping attacks needed, based on the needs of detection** (and adjusting the parameters of the J48 Classifier as such), whether if just detecting a DDoS attack is sufficient (like the bi-class classification done Level 0 Grouped Classification), if the type of attack is essential (Reflection or Exploitation attacks for Level 1 Grouped Classification), or if the protocol of the attack is needed (TCP, UDP, TCP/UDP attack types for Level 2 Grouped Classification).

4.7 Summary

Full results from both Preliminary Ungrouped Classification (and Optimisation by testing different values of C and M) and Hierarchical Grouped Classification are tabulated, graphed, illustrated, interpreted and discussed thoroughly to answer the problem statement for the study, which is to find out if hierarchical grouping helps mitigate the problems of unbalanced classification. While testing different values of C and M only provided marginal changes in evaluation metrics (especially TPR), hierarchical grouped classification failed to mitigate these problems, only make them harder to find out as the original labels had to be removed to effectively classify instances, with misclassification still happening between groups (for Level 1 and 2 Grouped Classification) and risk oversimplification of DDoS attack classification (especially for Level 0 Grouped Classification). In terms of overall TPR and GMEAN, Level 1 Grouped Classification performed the best, albeit being only second in terms of overall F1, and last in terms of overall PREC and highest overall FPR.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

A full study on data mining techniques has been done to investigate the effects of unbalanced data sets and ways to effectively detect and classify DDoS attacks (especially rare DDoS attacks). From performing a complete literature review of DDoS attacks (and the available data sets to use for the study) and unbalanced data sets, it is clear how **being able to deal with unbalanced data sets is vital, not only in the field of cybersecurity, but in many other fields as well,** as rare DDoS attacks are not any less significant (if not more so) in terms of the level of threat posed to everyone, as compared to their more common counterparts, and what evaluation metrics are available and suitable to deal with unbalanced classification.

From a thoroughly investigated and explained methodology, **a structured and well-planned study highlighting all the phases of the study**, starting with pre-processing the Consolidated DDoS Data Set via under-sampling all attack labels at 3% (except WebDDoS attack labels) to create a sampled data set, performing preliminary classification using default parameters for five classification algorithms selected for the study, ending up with **selecting the J48 classifier to continue with classification,** followed by **testing different values for the confidence factor (C) and minimum number of objects in a leaf (M) in the J48 Classifier**, getting mixed results in terms of changing True Positive Rates (TPR) for different attack labels (although only marginal changes in TPR values were observed), then **performing hierarchical grouping of DDoS attack labels in the same sampled data set** based on an existing hierarchy to observe the performance of the J48 classifier on group(s) of DDoS attacks by Level 0, 1 and 2 of the hierarchy in Figure 3.3.

While **hierarchical grouping DDoS attacks does increase overall TPR of DDoS attack classification** (by including DDoS attacks grouped as other DDoS attacks in the calculation of TPR), **the increase in TPR is almost negligible, and does nothing to address class imbalance issues and misclassification** (especially true for

Level 1 and 2 Grouped Classification), **while potentially leading to oversimplification of classification** (especially true for Level 0 Grouped Classification). Even with the highest overall TPR, **Level 1 Grouped Classification (with C = 0.23 and M = 3) still fell short in terms of other evaluation metrics** like PREC, F1 and have the **highest FPR** among the other classifications studied. Even with testing different values of C and M, **Level 0 Grouped Classification shows basically uniform results regardless of the value of C and M used**, whereas for Level 1 Grouped Classification, the parameters **C = 0.23 and M = 3 were the best performing setting for the J48 Classifier,** with most positive and least negative Net Changes. For Level 2 Grouped Classification, **the default parameters of the J48 Classifier (C = 0.25 and M = 2) were the best performing settings.** For both Level 1 and 2 Grouped Classification, comparisons with the same settings performed on the ungrouped data set were done to check for similarities of misclassified DDoS attack instances, to show that **even when grouped, misclassification of DDoS attacks (especially those of problematic attack labels like Portmap, SSDP and UDPLag) still happen and contribute to relatively high False Positive Rate (FPR)** for in Level 1 and 2 Grouped Classification (3.4% for Exploitation attacks in Level 1 Grouped Classification and 3.0% for UDP (Exploitation) attacks in Level 2 Grouped Classification).

5.2 Recommendations

Further study on other ways to improve detection rate for rare classes of DDoS attacks (data-level, algorithm-level, and hybrid approaches) will still be needed, and Intrusion Detection Systems (IDS) currently in use must also continue to evolve to handle more sophisticated and undetectable DDoS attacks sent by hackers that are always one step ahead when it comes to conducting cybercrime. Being able to detect rare DDoS attacks are especially important, especially when these attacks are **newer attack types which do not previously have much data to collect to begin with due to their recency,** hence their rarity in available DDoS attack databases. Nevertheless, **newer attacks are likely more sophisticated than their more common counterparts,** as they are created to mitigate the weaknesses of older attacks and hence, can better evade detection from currently used IDS, which is **evident with the low TPR values of problematic DDoS attacks like Portmap, SSDP and UDPLag** from the J48 Classifier in the study. While the study mainly focused on hierarchical grouping based on existing hierarchy, **more study on creating different hierarchies and grouping methods for DDoS attacks will still be needed** due to the lack of studies on DDoS attack grouping in literature.

REFERENCES

Abro, A. A., Khan, A. A., Talpur, M. S., Kayijuka, I., & Yasar, E. (2021). Machine Learning Classifiers: A Brief Primer. *University of Sindh Journal of Information and Communication Technology (USJICT), 5*(2), 63-68.

Abro, A. A., Taşcı, E., & Ugur, A. (April, 2020). A Stacking-based Ensemble Learning Method for Outlier Detection. *Balkan Journal of Electrical and Computer Engineering, 8*(2), 181-185. doi:10.17694/bajece.679662

Ahmim, A., Maglaras, L., Ferrag, M. A., Derdour, M., & Janicke, H. (2019). A Novel Hierarchical Intrusion Detection System Based on Decision Tree and Rules-Based Models. *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)* (pp. 228-233). Santorini, Greece: IEEE. doi:10.1109/DCOSS.2019.00059

Almaraz-Rivera, J. G., Perez-Diaz, J. A., & Cantoral-Ceballos, J. A. (28 April, 2022). Transport and Application Layer DDoS Attacks Detection to IoT Devices by Using Machine Learning and Deep Learning Models. (G. Ghinea, Ed.) *Sensors, 22*(9), 3367. doi:10.3390/s22093367

Bernama News. (5 August, 2022). *DDoS Attacks rose 100 times to reach 3,000 minutes in Q2 2022 - Kapersky*. Retrieved 24 March, 2023, from Bernama News: https://bernama.com/en/business/news.php?id=2107914

Beyan, C., & Fisher, R. (May, 2015). Classifying imbalanced data sets using similarity based hierarchical decomposition. *Pattern Recognition, 48*(5), 1653-1672. doi:10.1016/j.patcog.2014.10.032

Bhattacharyya, D. K., & Kalita, J. K. (2016). *DDoS Attacks: Evolution, Detection, Prevention, Reaction, and Tolerance.* Tezpur, Assam, India: CRC Press.

Bolodurina, I., Shukman, A., Parfenov, D., Zhigalov, A., & Zabrodina, L. (1 November, 2020). Investigation of the problem of classifying unbalanced datasets in identifying distributed denial of service attacks. *Journal of Physics: Conference Series, 1679*(4). doi:10.1088/1742-6596/1679/4/042020

Brewer, B. J. (n.d.). *Stat331: Introduction to Bayesian Statistics.* Retrieved 7 May, 2023, from https://www.stat.auckland.ac.nz/~brewer/stats331.pdf

C, L. D. (2014). Comparative Analysis of Random Forest, REP Tree and J48 Classifiers for Credit Risk Prediction. *International Journal of Computer Applications, 975*(8887), 30-36.

Chkirbene, Z., Eltanbouly, S., Baschendy, M., AlNaimi, N., & Erbad, A. (2020). Hybrid Machine Learning for Network Anomaly Intrusion Detection. *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)* (pp. 163-170). Doha, Qatar: IEEE. doi:10.1109/ICIoT48696.2020.9089575

Cybersecurity & Infrastructure Security Agency (CISA). (1 February, 2021). *Understanding Denial-of-Service Attacks | CISA*. Retrieved 5 May, 2023, from Cybersecurity & Infrastructure Security Agency (CISA): https://www.cisa.gov/news-events/news/understanding-denial-service-attacks

Das, S., Chaudhuri, B. B., & Datta, S. (September, 2018). Handling data irregularities in classification: Foundations, trends, and future challenges. *Pattern Recognition, 81*, 674-693. doi:10.1016/j.patcog.2018.03.008

Doriguzzi-Corin, R., & Siracusa, D. (23 August, 2022). *FLAD: Adaptive Federated Learning for DDoS Attack Detection.* doi:10.48550/arXiv.2205.06661

Dutt, S., Chandramouli, S., & Kumar Das, A. (2019). *Machine Learning.* Chennai, Tamil Nadu, India: Pearson India Education Services.

Enoch, M., & Khor, K.-C. (9 December, 2021). DDoS Attacks Data Set - Consolidated from CICDDOS2019 and CICIDS2017. Sungai Long, Cheras, Selangor, Malaysia: Zenodo. doi:10.5281/zenodo.5770290

Erhan, D., & Anarim, E. (2019). Statistical Properties of DDoS Attacks. *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT'19)* (pp. 1238-1242). Paris, France: IEEE. Retrieved 22 May, 2023, from https://www.researchgate.net/profile/Emin-Anarim/publication/334084373_Statistical_Properties_of_DDoS_Attacks/links/5d15e8fda6fdcc2462ab989c/Statistical-Properties-of-DDoS-Attacks.pdf

Erhan, D., & Anarim, E. (2020). Boğaziçi University distributed denial of service dataset. *Data in Brief, 32*(106187). doi:10.1016/j.dib.2020.106187

Fawcett, T. (June, 2006). An introduction to ROC analysis. *Pattern Recognition Letters, 27*(8), 861-874. doi:10.1016/j.patrec.2005.10.010

Ferrag, M. A., Shu, L., Hamouda, D., & Choo, K.-K. R. (May, 2021). Deep Learning-Based Intrusion Detection for Distributed Denial of Service Attack in Agriculture 4.0. *Electronics, 10*(1257). doi:10.3390/electronics10111257

Gohil, M., & Kumar, S. (December, 2020). Evaluation of Classification algorithms for Distributed Denial of Service Attack Detection. *2020 IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, 138-141. doi:10.1109/AIKE48582.2020.00028

Guo, H., Li, Y., Shang, J., Gu, M., Huang, Y., & Gong, B. (1 May, 2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems With Applications*, *73*, 220-239. doi:10.1016/j.eswa.2016.12.035

Hajian-Tilaki, K. (2013). Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation. *Caspian J Intern Med*, *4*(2), 627-635. Retrieved 21 May, 2023, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3755824/

Hermawan, D. R., Fatihah, M. F., Kurniawati, L., & Helen, A. (2021). Comparative Study of J48 Decision Tree Classification Algorithm, Random Tree, and Random Forest on In-Vehicle Coupon Recommendation Data. *International Conference on Artificial Intelligence and Big Data Analytics*, 1-6.

Ho, Y.-B., Yap, W.-S., & Khor, K.-C. (2 October, 2021). The Effect of Sampling Methods on the CICIDS2017 Network Intrusion Data Set. (H. Kim, & K. Kim, Eds.) *IT Convergence and Security. Lecture Notes in Electrical Engineering*, 33-41. doi:10.1007/978-981-16-4118-3_4

IBM. (n.d.). *What is Naive Bayes | IBM*. Retrieved 6 May, 2023, from IBM: https://www.ibm.com/topics/naive-bayes#:~:text=The%20Na%C3%AFve%20Bayes%20classifier%20is,a%20given%20class%20or%20category.

IBM. (n.d.). *What is Random Forest? | IBM*. Retrieved 8 May, 2023, from https://www.ibm.com/topics/random-forest

IBM. (n.d.). *What is the k-nearest neighbours algorithm? | IBM*. Retrieved 2 June, 2023, from https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20algorithm%2C%20also%20known%20as%20KNN%20or,of%20an%20individual%20data%20point.

Jaszcz, A., & Połap, D. (23 July, 2022). AIMM: Artificial Intelligence Merged Methods for flood DDoS attacks detection. *Journal of King Saud University -*

Computer and Information Sciences, 34(10), 8090-8101. doi:10.1016/j.jksuci.2022.07.021

Kassim, S. R. (2015). *What Organizations Need to Know About Distributed Denial-of-Service (DDOS) Attacks.* Retrieved 30 April, 2023, from Malaysia Computer Emergency Response Team (MyCERT): https://www.mycert.org.my/portal/publicationdoc?id=7f3a6d8b-c723-4cf0-b6f3-9f9fe0af1a9a

Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (November, 2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Generation Computer Systems, 100*, 779-796. doi:10.1016/j.future.2019.05.041

Krawczyk, B. (2016). Learning from imbalanced data: open challenges. *Prog Artif Intell, 5*, 221-232. doi:10.1007/s13748-016-0094-0

Krawczyk, B. (22 April, 2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence, 5*, 221-232. doi:10.1007/s13748-016-0094-0

Li, K., Ma, W., Duan, H., Xie, H., Zhu, J., & Liu, R. (24 October, 2022). Unbalanced network attack traffic detection based on feature extraction and GFDA-WGAN. *Computer Networks, 216*, 109283. doi:10.1016/j.comnet.2022.109283

Lohrmann, D. (25 August, 2022). *Opinion: Hacktivism and DDOS attacks rise dramatically in 2022*. Retrieved 24 March, 2023, from The Star: https://www.thestar.com.my/tech/tech-news/2022/08/25/opinion-hacktivism-and-ddos-attacks-rise-dramatically-in-2022

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences, 250*, 113-141. doi:10.1016/j.ins.2013.07.007

Loyola-González, O., Martínez-Trinidad, J. F., Carrasco-Ochoa, J. A., & García-Borroto, M. (2016). Study of the impact of resampling methods for contrast pattern based classifiers in imbalanced databases. *Neurocomputing, 175*(Part B), 935-947. doi:10.1016/j.neucom.2015.04.120

Meena, G., & Choudhary, R. R. (2017). A Review Paper on IDS Classification using KDD 99 and NSL KDD Dataset in WEKA. *2017 International Conference on*

Computer, Communications and Electronics (Comptelix), (pp. 553-558). Jaipur, India. doi:10.1109/COMPTELIX.2017.8004032

Merino, T., Stillwell, M., Steele, M., Coplan, M., Patton, J., Stoyanov, A., & Deng, L. (25 July, 2019). Expansion of Cyber Attack Data from Unbalanced Datasets Using Generative Adversarial Networks. (R. Lee, Ed.) *Software Engineering Research, Management and Applications (SERA), 845*, 131-145. doi:10.1007/978-3-030-24344-9_8

Microsoft. (17 February, 2023). *DoS vs. DDoS Attacks: What's The Difference? | Microsoft 365*. Retrieved 5 May, 2023, from Microsoft 365: https://www.microsoft.com/en-us/microsoft-365-life-hacks/privacy-and-safety/dos-vs-ddos-attacks-whats-the-difference

Microsoft Security. (21 February, 2023). *2022 in review: DDoS attack trends and insights*. Retrieved 24 March, 2023, from Microsoft Security Blog: https://www.microsoft.com/en-us/security/blog/2023/02/21/2022-in-review-ddos-attack-trends-and-insights/

N.G., B. A., & Selvakumar, S. (December, 2020). Anomaly detection framework for Internet of things traffic using vector convolutional deep learning approach in fog environment. *Future Generation Computer Systems, 113*, 255-265. doi:10.1016/j.future.2020.07.020

Nevill-Manning, C., Holmes, G., & Witten, I. H. (1995). The Development of Holte's 1R Classifier. *Proceedings 1995 Second New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems* (pp. 239-242). Dunedin, New Zealand: IEEE. doi:10.1109/ANNES.1995.499480

Novakovic, J. (2010). The Impact of Feature Selection on the Accuracy on the Naive Bayes Classifier. *Telecommunications Forum TELFOR*.

Panigrahi, R., & Borah, S. (2018). A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems. *International Journal of Engineering & Technology, 7*(3.24), 479-482. Retrieved 2 August, 2023, from https://www.researchgate.net/publication/329045441_A_detailed_analysis_of_CICIDS2017_dataset_for_designing_Intrusion_Detection_Systems

Parsania, V. S., Jani, N. N., & Bhalodiya, N. H. (2014). Applying Naïve bayes, BayesNet, PART, JRip and OneR Algorithms on Hypothyroid Database for Comparative Analysis. *International Journal of Darshan Institute on*

Engineering Research & Emerging Technologies, 3(1), 60-64. Retrieved 12 May, 2023

Pereira, R. M., Costa, Y. M., & Silla, C. N. (7 July, 2021). Toward hierarchical classification of imbalanced data using random resampling algorithms. *Information Sciences, 578,* 344-363. doi:10.1016/j.ins.2021.07.033

Peterson, J. M., Leevy, J. L., & Khoshgoftaar, T. M. (2021). A Review and Analysis of the Bot-IoT Dataset. *2021 IEEE International Conference on Service-Oriented System Engineering (SOSE)* (pp. 20-27). Oxford, United Kingdom: IEEE. doi:10.1109/SOSE52839.2021.00007

Pham Nguyen, A. H., & Triantaphyllou, E. (2007). The Impact of Overfitting and Overgeneralization on the Classification Accuracy in Data Mining. In O. Maiman, & L. Rokach (Eds.), *Soft Computing for Knowledge Discovery and Data Mining* (pp. 391-431). Baton Rouge, Louisiana, USA: Springer. Retrieved 21 November, 2023

Pozzolo, A. D., Caelen, O., Johnson, R. A., & Bontempi, G. (2015). Calibrating Probability with Undersampling for Unbalanced Classification. *2015 IEEE Symposium Series on Computational Intelligence.* Cape Town, South Africa: IEEE. doi:10.1109/SSCI.2015.33

Rajeswari, V., & Arunesh, K. (May, 2016). Analysing Soil Data using Data Mining Classification Techniques. *Indian Journal of Science and Technology, 9*(19), 1-4. doi:10.17485/ijst/2016/v9i19/93873

Sharafaldin, I., Lashkari, A. H., Hakak, S., & Ghorbani, A. A. (2019). Developing Realistic Distributed Denial of Service (DDoS) Attack Dataset and Taxonomy. *2019 International Carnahan Conference on Security Technology (ICCST)* (pp. 1-8). Chennai, India: IEEE. doi:10.1109/CCST.2019.8888419

Silla, C. N., & Freitas, A. A. (January, 2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery, 22,* 31-72. doi:10.1007/s10618-010-0175-9

Song, H., Leng, H., Hou, Z., Gao, R., Chen, C., Meng, C., . . . Ma, B. (December, 2022). Grouped-sampling technique to deal with unbalance in Raman spectral data modeling. *Photodiagnosis and Photodynamic Therapy, 40*(103059).

Sun, Y., Wong, A. C., & Kamel, M. S. (2009). Classification of imbalanced data: a review. *International Journal of Pattern Recognition and Artificial Intelligence, 23*(4), 687-719. doi:10.1142/S0218001409007326

University of New Brunswick. (n.d.). *DDoS Evaluation Dataset (CIC-DDoS2019)*. Retrieved 1 August, 2023, from https://www.unb.ca/cic/datasets/ddos-2019.html

University of Waikato. (28 January, 2022). *IBk*. Retrieved 8 August, 2023, from WEKA Source Forge: https://weka.sourceforge.io/doc.dev/weka/classifiers/lazy/IBk.html

University of Waikato. (28 January, 2022). *J48*. Retrieved 8 August, 2023, from WEKA Source Forge: https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/J48.html

University of Waikato. (28 January, 2022). *JRip*. Retrieved 1 August, 2023, from WEKA Source Forge: https://weka.sourceforge.io/doc.dev/weka/classifiers/rules/JRip.html

University of Waikato. (28 January, 2022). *Naive Bayes*. Retrieved 8 August, 2023, from WEKA Source Forge: https://weka.sourceforge.io/doc.dev/weka/classifiers/bayes/NaiveBayes.html

University of Waikato. (28 January, 2022). *Random Forest*. Retrieved 8 August, 2023, from WEKA Source Forge: https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/RandomForest.html#RandomForest--

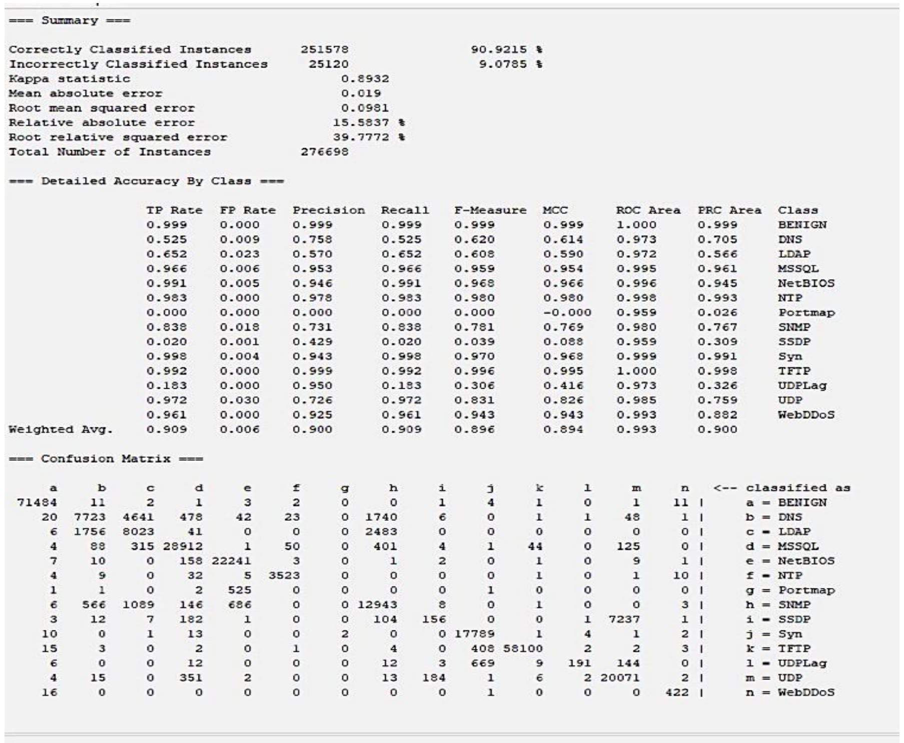
Witten, I. H. (28 April, 2014). More Data Mining with Weka (1.1: Introduction). Hillcrest, Hamilton, New Zealand. Retrieved 16 March, 2023

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques* (3rd Edition ed.). Burlington, Massachusetts, USA: Morgan Kaufmann Publishers.

APPENDICES

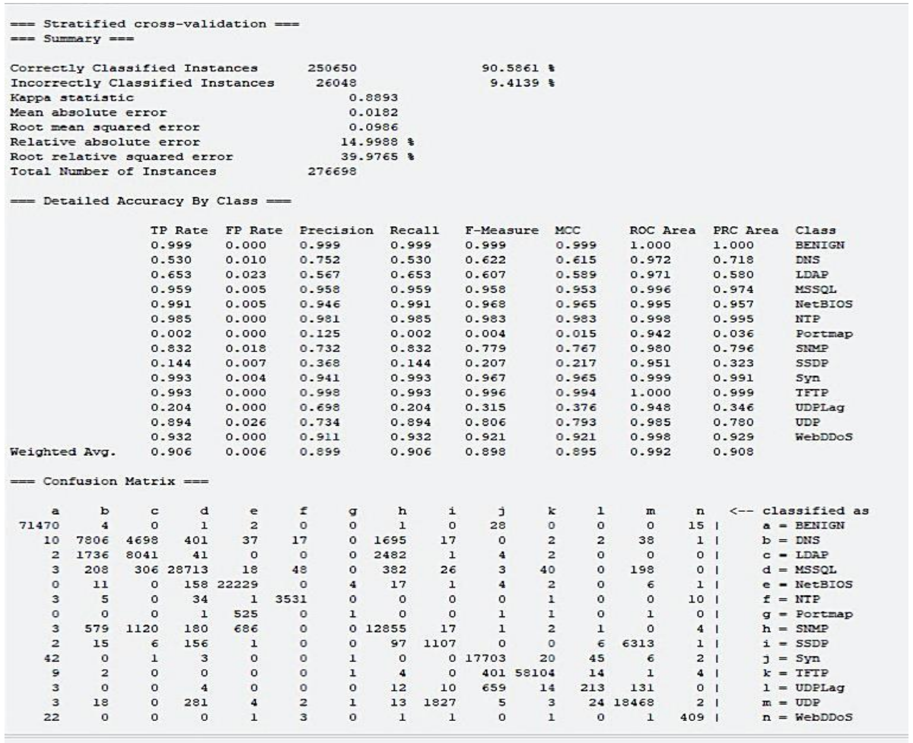
APPENDIX A: Preliminary Classification and Optimisation Results

Preliminary Classification Results (All Classifiers)



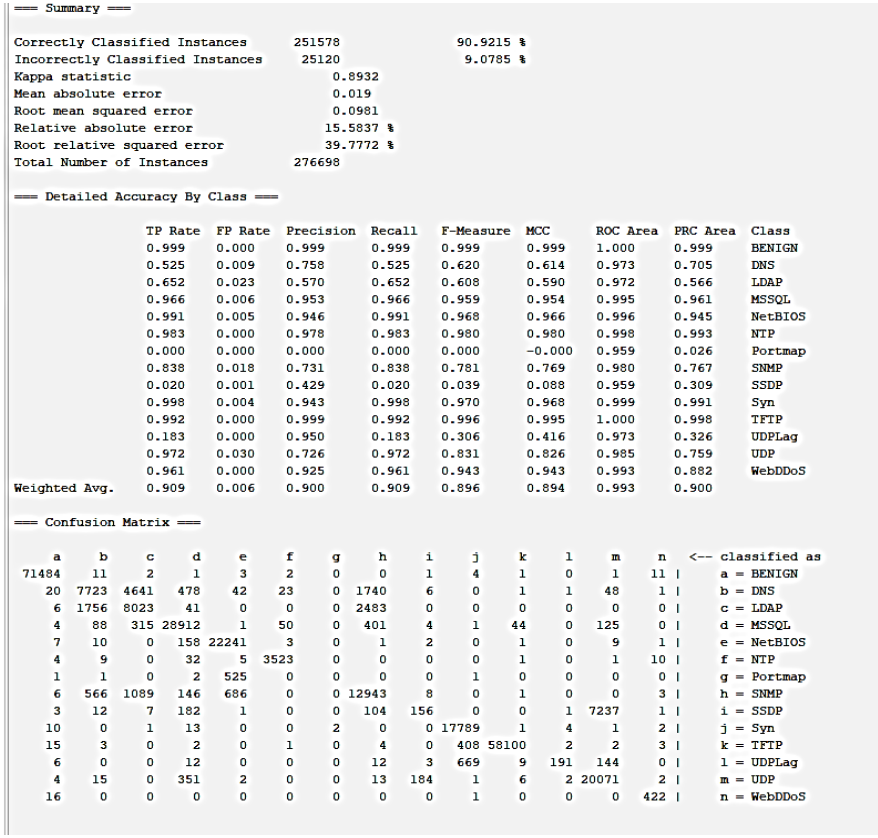
```
=== Summary ===

Correctly Classified Instances      251578               90.9215 %
Incorrectly Classified Instances     25120                9.0785 %
Kappa statistic                          0.8932
Mean absolute error                      0.019
Root mean squared error                  0.0981
Relative absolute error                 15.5837 %
Root relative squared error             39.7772 %
Total Number of Instances           276698

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.999    0.000    0.999      0.999    0.999      0.999    1.000     0.999     BENIGN
                 0.525    0.009    0.758      0.525    0.620      0.614    0.973     0.705     DNS
                 0.652    0.023    0.570      0.652    0.608      0.590    0.972     0.566     LDAP
                 0.966    0.006    0.953      0.966    0.959      0.954    0.995     0.961     MSSQL
                 0.991    0.005    0.946      0.991    0.968      0.966    0.996     0.945     NetBIOS
                 0.983    0.000    0.978      0.983    0.980      0.980    0.998     0.993     NTP
                 0.000    0.000    0.000      0.000    0.000      -0.000   0.959     0.026     Portmap
                 0.838    0.018    0.731      0.838    0.781      0.769    0.980     0.767     SNMP
                 0.020    0.001    0.429      0.020    0.039      0.088    0.959     0.309     SSDP
                 0.998    0.004    0.943      0.998    0.970      0.968    0.999     0.991     Syn
                 0.992    0.000    0.999      0.992    0.996      0.995    1.000     0.998     TFTP
                 0.183    0.000    0.950      0.183    0.306      0.416    0.973     0.326     UDPLag
                 0.972    0.030    0.726      0.972    0.831      0.826    0.985     0.759     UDP
                 0.961    0.000    0.925      0.961    0.943      0.943    0.993     0.882     WebDDoS
Weighted Avg.    0.909    0.006    0.900      0.909    0.896      0.894    0.993     0.900

=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71484    11     2     1     3     2     0     0     1     4     1     0     1    11 |     a = BENIGN
    20  7723  4641   478    42    23     0  1740     6     0     1     1    48     1 |     b = DNS
     6  1756  8023    41     0     0     0  2483     0     0     0     0     0     0 |     c = LDAP
     4    88   315 28912     1    50     0   401     4     1    44     0   125     0 |     d = MSSQL
     7    10     0   158 22241     3     0     1     2     0     1     0     9     1 |     e = NetBIOS
     4     9     0    32     5  3523     0     0     0     0     1     0     1    10 |     f = NTP
     1     1     0     2   525     0     0     0     0     1     0     0     0     0 |     g = Portmap
     6   566  1089   146   686     0     0 12943     8     0     1     0     0     3 |     h = SNMP
     3    12     7   182     1     0     0   104   156     0     0     1  7237     1 |     i = SSDP
    10     0     1    13     0     0     2     0     0 17789     1     4     1     2 |     j = Syn
    15     3     0     2     0     1     0     4     0   408 58100     2     2     3 |     k = TFTP
     6     0     0    12     0     0     0    12     3   669     9   191   144     0 |     l = UDPLag
     4    15     0   351     2     0     0    13   184     1     6     2 20071     2 |     m = UDP
    16     0     0     0     0     0     0     0     0     1     0     0     0   422 |     n = WebDDoS
```

Figure A-1: Confusion Matrix and Evaluation Metrics Generated from 10-fold cross validation for Preliminary Classification in WEKA using Naïve Bayes (NB) classifier with default parameters.



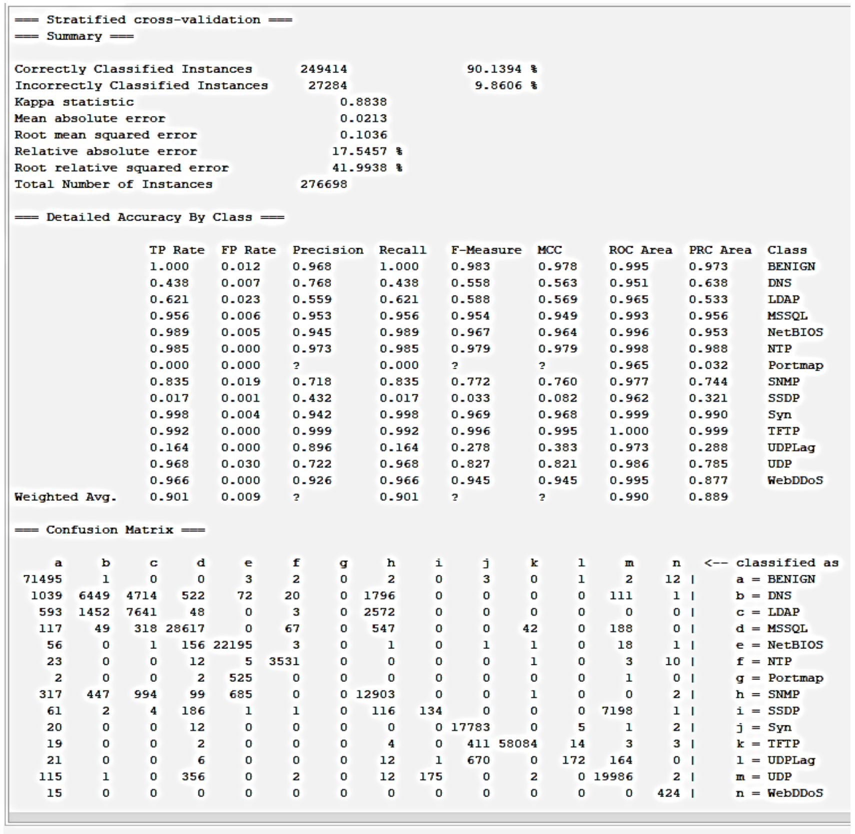
```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      250650               90.5861 %
Incorrectly Classified Instances     26048                9.4139 %
Kappa statistic                          0.8893
Mean absolute error                      0.0182
Root mean squared error                  0.0986
Relative absolute error                 14.9988 %
Root relative squared error             39.9765 %
Total Number of Instances           276698

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.999    0.000    0.999      0.999    0.999      0.999    1.000     1.000     BENIGN
                 0.530    0.010    0.752      0.530    0.622      0.615    0.972     0.718     DNS
                 0.653    0.023    0.567      0.653    0.607      0.589    0.971     0.580     LDAP
                 0.959    0.005    0.958      0.959    0.958      0.953    0.996     0.974     MSSQL
                 0.991    0.005    0.946      0.991    0.968      0.965    0.995     0.957     NetBIOS
                 0.985    0.000    0.981      0.985    0.983      0.983    0.998     0.995     NTP
                 0.002    0.000    0.125      0.002    0.004      0.015    0.942     0.036     Portmap
                 0.832    0.018    0.732      0.832    0.779      0.767    0.980     0.796     SNMP
                 0.144    0.007    0.368      0.144    0.207      0.217    0.951     0.323     SSDP
                 0.993    0.004    0.941      0.993    0.967      0.965    0.999     0.991     Syn
                 0.993    0.000    0.998      0.993    0.996      0.994    1.000     0.999     TFTP
                 0.204    0.000    0.698      0.204    0.315      0.376    0.948     0.346     UDPLag
                 0.894    0.026    0.734      0.894    0.806      0.793    0.985     0.780     UDP
                 0.932    0.000    0.911      0.932    0.921      0.921    0.998     0.929     WebDDoS
Weighted Avg.    0.906    0.006    0.899      0.906    0.898      0.895    0.992     0.908

=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71470     4     0     1     2     0     0     1     0    28     0     0     0    15 |     a = BENIGN
    10  7806  4698   401    37    17     0  1695    17     0     2     2    38     1 |     b = DNS
     2  1736  8041    41     0     0     0  2482     1     4     2     0     0     0 |     c = LDAP
     3   208   306 28713    18    48     0   382    26     3    40     0   198     0 |     d = MSSQL
     0    11     0   158 22229     0     4    17     1     4     2     0     6     1 |     e = NetBIOS
     3     5     0    34     1  3531     0     0     0     0     1     0     0    10 |     f = NTP
     0     0     0     1   525     0     1     0     0     1     1     0     1     0 |     g = Portmap
     3   579  1120   180   686     0     0 12855    17     1     2     1     0     4 |     h = SNMP
     2    15     6   156     1     0     0    97  1107     0     0     6  6313     1 |     i = SSDP
    42     0     1     3     0     0     1     0     0 17703    20    45     6     2 |     j = Syn
     9     2     0     0     0     0     1     4     0   401 58104    14     1     4 |     k = TFTP
     3     0     0     4     0     0     0    12    10   659    14   213   131     0 |     l = UDPLag
     3    18     0   281     4     2     1    13  1827     5     3    24 18468     2 |     m = UDP
    22     0     0     0     1     3     0     1     1     0     1     0     1   409 |     n = WebDDoS
```

Figure A-2: Confusion Matrix and Evaluation Metrics Generated from 10-fold cross validation for Preliminary Classification in WEKA using Random Forest (RF) classifier with default parameters.



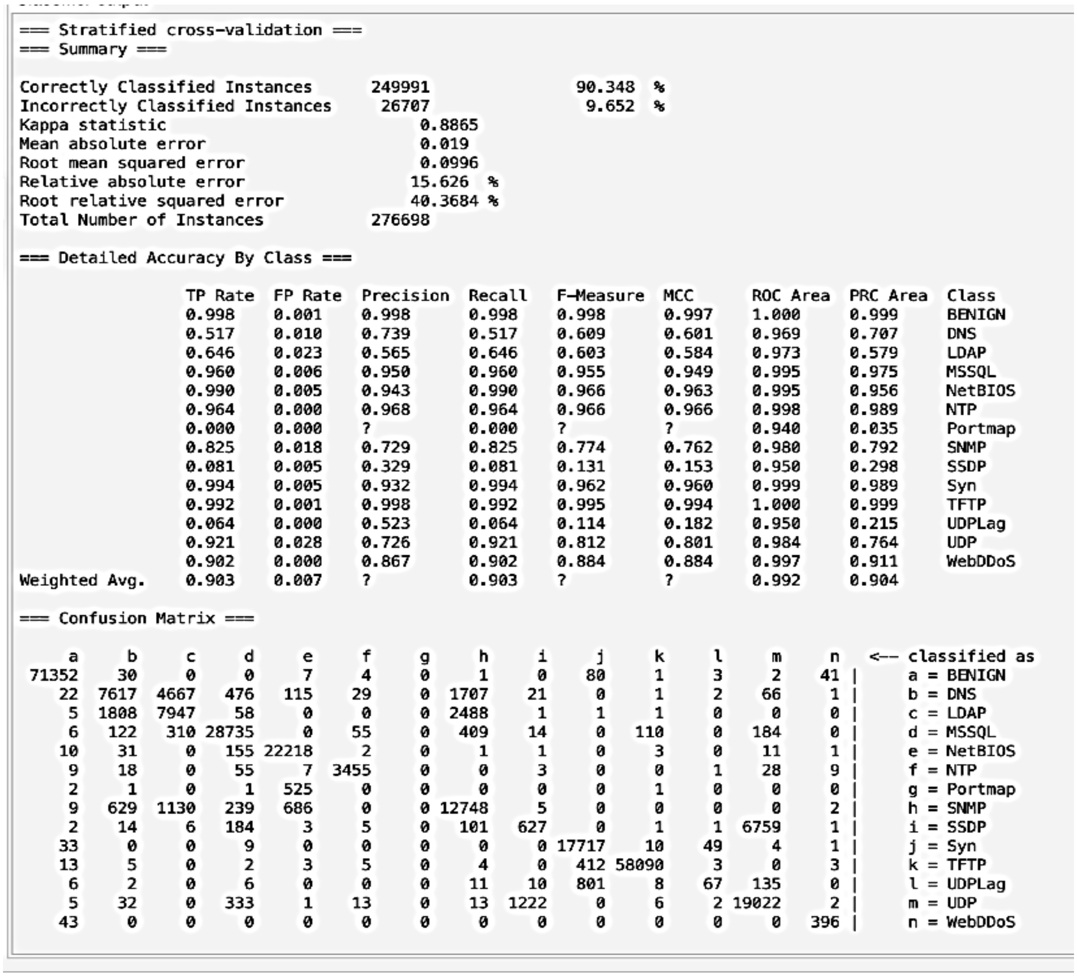
```
=== Summary ===

Correctly Classified Instances      251578               90.9215 %
Incorrectly Classified Instances     25120                9.0785 %
Kappa statistic                          0.8932
Mean absolute error                      0.019
Root mean squared error                  0.0981
Relative absolute error                 15.5837 %
Root relative squared error             39.7772 %
Total Number of Instances           276698

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.999    0.000    0.999      0.999    0.999      0.999    1.000     0.999     BENIGN
                 0.525    0.009    0.758      0.525    0.620      0.614    0.973     0.705     DNS
                 0.652    0.023    0.570      0.652    0.608      0.590    0.972     0.566     LDAP
                 0.966    0.006    0.953      0.966    0.959      0.954    0.995     0.961     MSSQL
                 0.991    0.005    0.946      0.991    0.968      0.966    0.996     0.945     NetBIOS
                 0.983    0.000    0.978      0.983    0.980      0.980    0.998     0.993     NTP
                 0.000    0.000    0.000      0.000    0.000      -0.000   0.959     0.026     Portmap
                 0.838    0.018    0.731      0.838    0.781      0.769    0.980     0.767     SNMP
                 0.020    0.001    0.429      0.020    0.039      0.088    0.959     0.309     SSDP
                 0.998    0.004    0.943      0.998    0.970      0.968    0.999     0.991     Syn
                 0.992    0.000    0.999      0.992    0.996      0.995    1.000     0.998     TFTP
                 0.183    0.000    0.950      0.183    0.306      0.416    0.973     0.326     UDPLag
                 0.972    0.030    0.726      0.972    0.831      0.826    0.985     0.759     UDP
                 0.961    0.000    0.925      0.961    0.943      0.943    0.993     0.882     WebDDoS
Weighted Avg.    0.909    0.006    0.900      0.909    0.896      0.894    0.993     0.900

=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71484    11     2     1     3     2     0     0     1     4     1     0     1    11 |     a = BENIGN
    20  7723  4641   478    42    23     0  1740     6     0     1     1    48     1 |     b = DNS
     6  1756  8023    41     0     0     0  2483     0     0     0     0     0     0 |     c = LDAP
     4    88   315 28912     1    50     0   401     4     1    44     0   125     0 |     d = MSSQL
     7    10     0   158 22241     3     0     1     2     0     1     0     9     1 |     e = NetBIOS
     4     9     0    32     5  3523     0     0     0     0     1     0     1    10 |     f = NTP
     1     1     0     2   525     0     0     0     0     1     0     0     0     0 |     g = Portmap
     6   566  1089   146   686     0     0 12943     8     0     1     0     0     3 |     h = SNMP
     3    12     7   182     1     0     0   104   156     0     0     1  7237     1 |     i = SSDP
    10     0     1    13     0     0     2     0     0 17789     1     4     1     2 |     j = Syn
    15     3     0     2     0     1     0     4     0   408 58100     2     2     3 |     k = TFTP
     6     0     0    12     0     0     0    12     3   669     9   191   144     0 |     l = UDPLag
     4    15     0   351     2     0     0    13   184     1     6     2 20071     2 |     m = UDP
    16     0     0     0     0     0     0     0     0     1     0     0     0   422 |     n = WebDDoS
```

Figure A-3: Confusion Matrix and Evaluation Metrics Generated from 10-fold cross validation for Preliminary Classification in WEKA using J48 classifier with default parameters.



```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      249414               90.1394 %
Incorrectly Classified Instances     27284                9.8606 %
Kappa statistic                          0.8838
Mean absolute error                      0.0213
Root mean squared error                  0.1036
Relative absolute error                 17.5457 %
Root relative squared error             41.9938 %
Total Number of Instances           276698

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.012    0.968      1.000    0.983      0.978    0.995     0.973     BENIGN
                 0.438    0.007    0.768      0.438    0.558      0.563    0.951     0.638     DNS
                 0.621    0.023    0.559      0.621    0.588      0.569    0.965     0.533     LDAP
                 0.956    0.006    0.953      0.956    0.954      0.949    0.993     0.956     MSSQL
                 0.989    0.005    0.945      0.989    0.967      0.964    0.996     0.953     NetBIOS
                 0.985    0.000    0.973      0.985    0.979      0.979    0.998     0.988     NTP
                 0.000    0.000    ?          0.000    ?          ?        0.965     0.032     Portmap
                 0.835    0.019    0.718      0.835    0.772      0.760    0.977     0.744     SNMP
                 0.017    0.001    0.432      0.017    0.033      0.082    0.962     0.321     SSDP
                 0.998    0.004    0.942      0.998    0.969      0.968    0.999     0.990     Syn
                 0.992    0.000    0.999      0.992    0.996      0.995    1.000     0.999     TFTP
                 0.164    0.000    0.896      0.164    0.278      0.383    0.973     0.288     UDPLag
                 0.968    0.030    0.722      0.968    0.827      0.821    0.986     0.785     UDP
                 0.966    0.000    0.926      0.966    0.945      0.945    0.995     0.877     WebDDoS
Weighted Avg.    0.901    0.009    ?          0.901    ?          ?        0.990     0.889

=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71495     1     0     0     3     2     0     2     0     3     0     1     2    12 |     a = BENIGN
  1039  6449  4714   522    72    20     0  1796     0     0     0     0   111     1 |     b = DNS
   593  1452  7641    48     0     3     0  2572     0     0     0     0     0     0 |     c = LDAP
   117    49   318 28617     0    67     0   547     0     0    42     0   188     0 |     d = MSSQL
    56     0     1   156 22195     3     0     1     0     1     1     0    18     1 |     e = NetBIOS
    23     0     0    12     5  3531     0     0     0     0     1     0     3    10 |     f = NTP
     2     0     0     2   525     0     0     0     0     0     0     0     1     0 |     g = Portmap
   317   447   994    99   685     0     0 12903     0     0     1     0     0     2 |     h = SNMP
    61     2     4   186     1     1     0   116   134     0     0     0  7198     1 |     i = SSDP
    20     0     0    12     0     0     0     0     0 17783     0     5     1     2 |     j = Syn
    19     0     0     2     0     0     0     4     0   411 58084    14     3     3 |     k = TFTP
    21     0     0     6     0     0     0    12     1   670     0   172   164     0 |     l = UDPLag
   115     1     0   356     0     2     0    12   175     0     2     0 19986     2 |     m = UDP
    15     0     0     0     0     0     0     0     0     0     0     0     0   424 |     n = WebDDoS
```

Figure A-4: Confusion Matrix and Evaluation Metrics Generated from 10-fold cross validation for Preliminary Classification in WEKA using JRip with default parameters.



```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances      249991               90.348  %
Incorrectly Classified Instances     26707                9.652  %
Kappa statistic                          0.8865
Mean absolute error                      0.019
Root mean squared error                  0.0996
Relative absolute error                 15.626  %
Root relative squared error             40.3684 %
Total Number of Instances           276698

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.998    0.001    0.998      0.998    0.998      0.997    1.000     0.999     BENIGN
                 0.517    0.010    0.739      0.517    0.609      0.601    0.969     0.707     DNS
                 0.646    0.023    0.565      0.646    0.603      0.584    0.973     0.579     LDAP
                 0.960    0.006    0.950      0.960    0.955      0.949    0.995     0.975     MSSQL
                 0.990    0.005    0.943      0.990    0.966      0.963    0.995     0.956     NetBIOS
                 0.964    0.000    0.968      0.964    0.966      0.966    0.998     0.989     NTP
                 0.000    0.000    ?          0.000    ?          ?        0.940     0.035     Portmap
                 0.825    0.018    0.729      0.825    0.774      0.762    0.980     0.792     SNMP
                 0.081    0.005    0.329      0.081    0.131      0.153    0.950     0.298     SSDP
                 0.994    0.005    0.932      0.994    0.962      0.960    0.999     0.989     Syn
                 0.992    0.001    0.998      0.992    0.995      0.994    1.000     0.999     TFTP
                 0.064    0.000    0.523      0.064    0.114      0.182    0.950     0.215     UDPLag
                 0.921    0.028    0.726      0.921    0.812      0.801    0.984     0.764     UDP
                 0.902    0.000    0.867      0.902    0.884      0.884    0.997     0.911     WebDDoS
Weighted Avg.    0.903    0.007    ?          0.903    ?          ?        0.992     0.904

=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71352    30     0     0     7     4     0     1     0    80     1     3     2    41 |     a = BENIGN
    22  7617  4667   476   115    29     0  1707    21     0     1     2    66     1 |     b = DNS
     5  1808  7947    58     0     0     0  2488     1     1     1     0     0     0 |     c = LDAP
     6   122   310 28735     0    55     0   409    14     0   110     0   184     0 |     d = MSSQL
    10    31     0   155 22218     2     0     1     1     0     3     0    11     1 |     e = NetBIOS
     9    18     0    55     7  3455     0     0     3     0     0     1    28     9 |     f = NTP
     2     1     0     1   525     0     0     0     0     0     1     0     0     0 |     g = Portmap
     9   629  1130   239   686     0     0 12748     5     0     0     0     0     2 |     h = SNMP
     2    14     6   184     3     5     0   101   627     0     1     1  6759     1 |     i = SSDP
    33     0     0     9     0     0     0     0     0 17717    10    49     4     1 |     j = Syn
    13     5     0     2     3     5     0     4     0   412 58090     3     0     3 |     k = TFTP
     6     2     0     6     0     0     0    11    10   801     8    67   135     0 |     l = UDPLag
     5    32     0   333     1    13     0    13  1222     0     6     2 19022     2 |     m = UDP
    43     0     0     0     0     0     0     0     0     0     0     0     0   396 |     n = WebDDoS
```

Figure A-5: Confusion Matrix and Evaluation Metrics Generated from 10-fold cross validation for Preliminary Classification in WEKA using KNN/IBk classifier with K=10.

Optimisation by Confidence Factor (-C, default value of 0.25)

Table A-1: Table of General Evaluation Metrics (Weighted Average) Obtained for each value of C tested in WEKA for Preliminary Ungrouped Classification using 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value of 0.25, while light red denotes lower than default value)

Parameter Tested	**Evaluation Metric (Weighted Average)**							**Standard Deviation**					
confidenceFactor (-C)	**ACC (%)**	**TPR (%)**	**PREC (%)**	**F1**	**AUC**	**TNR (%)**	**GMEAN**	**ACC (%)**	**TPR (%)**	**PREC**	**F1**	**AUC**	**TNR (%)**
0.1	90.88	91	90	0.89	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.2	90.92	91	90	0.90	0.99	99	0.9492	0.12	0.00	0.00	0.00	0.00	0.00
0.21	90.92	91	90	0.90	0.99	99	0.9492	0.12	0.00	0.00	0.00	0.00	0.00
0.22	90.92	91	90	0.90	0.99	99	0.9492	0.12	0.00	0.00	0.00	0.00	0.00
0.23	90.92	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.24	90.92	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.25 (Default Value)	**90.93**	**91**	**90**	**0.90**	**0.99**	**99**	**0.9492**	**0.13**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
0.26	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.27	90.92	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.28	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.29	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.3	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00

Table 4.3 (Continued)

0.4	90.92	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
0.5	90.90	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00

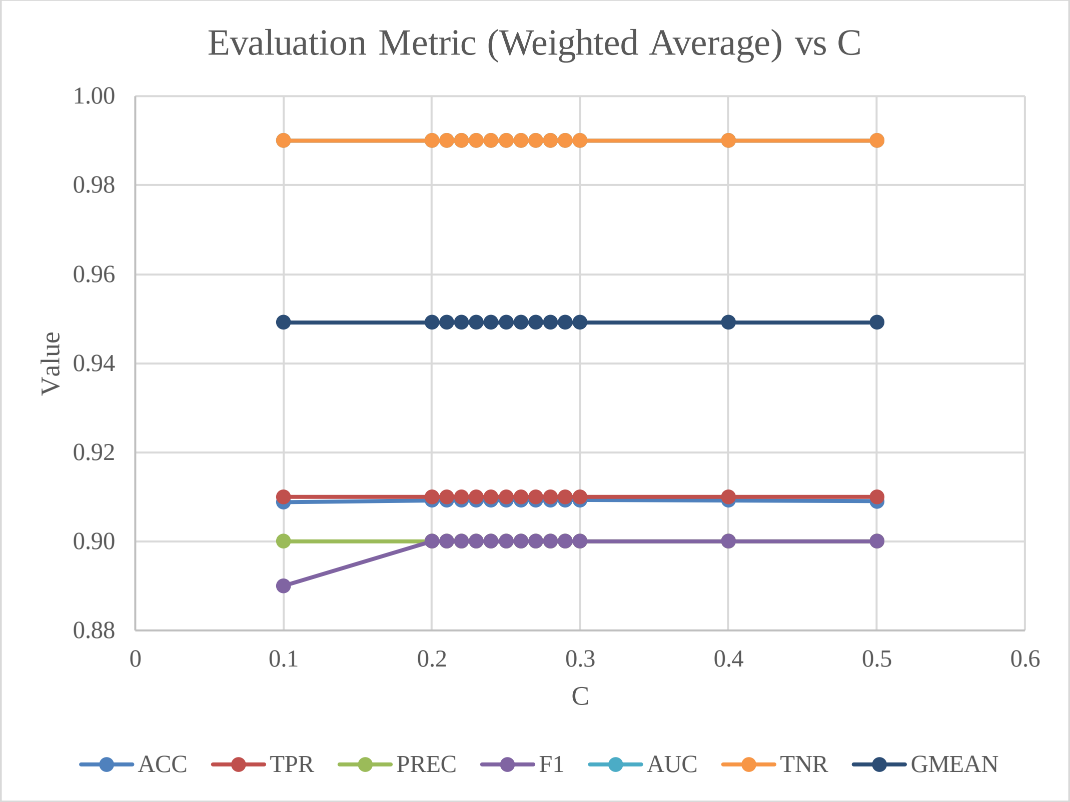


Figure A-6: Graph of General Weighted Average Evaluation Metrics by value of C tested in Experimenter for Preliminary Ungrouped Classification.

Table A-2: Table of True Positive Rates (TPR) Obtained for each class for each value of C tested in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	True Positive Rate (TPR) for each class (%)														Net Change for TPR (%)
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
0.1	99.9	51.7	65.0	96.5	99.1	98.0	0.0	84.3	0.5	99.8	99.2	16.5	97.7	95.4	-4.4
0.2	99.9	52.2	65.1	96.5	99.1	98.2	0.0	84.0	1.0	99.8	99.2	17.5	97.6	96.4	-1.5
0.21	99.9	52.3	65.1	96.5	99.1	98.2	0.0	83.9	1.1	99.8	99.2	17.5	97.6	96.1	-1.7
0.22	99.9	52.3	65.1	96.5	99.1	98.2	0.0	83.9	1.3	99.8	99.2	17.6	97.6	96.1	-1.4
0.23	99.9	52.4	65.1	96.5	99.1	98.2	0.0	83.8	1.4	99.8	99.2	17.7	97.5	96.1	-1.3
0.24	99.9	52.4	65.1	96.5	99.1	98.2	0.0	83.8	1.7	99.8	99.2	17.8	97.3	96.1	-1.1
0.25 (Default Value)	**99.9**	**52.5**	**65.2**	**96.6**	**99.1**	**98.3**	**0.0**	**83.8**	**2.0**	**99.8**	**99.2**	**18.3**	**97.2**	**96.1**	**0.0**
0.26	99.9	52.4	65.2	96.5	99.1	98.3	0.0	83.8	2.2	99.8	99.2	18.7	97.1	96.1	+0.3
0.27	99.9	52.5	65.2	96.5	99.1	98.2	0.0	83.8	2.1	99.8	99.2	18.5	97.2	96.1	+0.1
0.28	99.9	52.5	65.1	96.5	99.1	98.2	0.0	83.8	2.4	99.8	99.2	18.7	97.1	96.1	+0.4
0.29	99.9	52.6	65.1	96.6	99.1	98.2	0.0	83.7	2.5	99.8	99.2	18.7	97.1	96.1	+0.6
0.3	99.9	52.6	65.1	96.5	99.1	98.2	0.0	83.7	2.8	99.8	99.2	18.7	97.0	96.1	+0.7
0.4	99.9	52.7	65.2	96.4	99.1	98.3	0.0	83.6	3.6	99.8	99.2	19.0	96.7	96.4	+1.9
0.5	99.9	52.8	65.3	96.3	99.1	98.3	0.0	83.5	4.2	99.8	99.2	19.1	96.4	96.4	+2.3

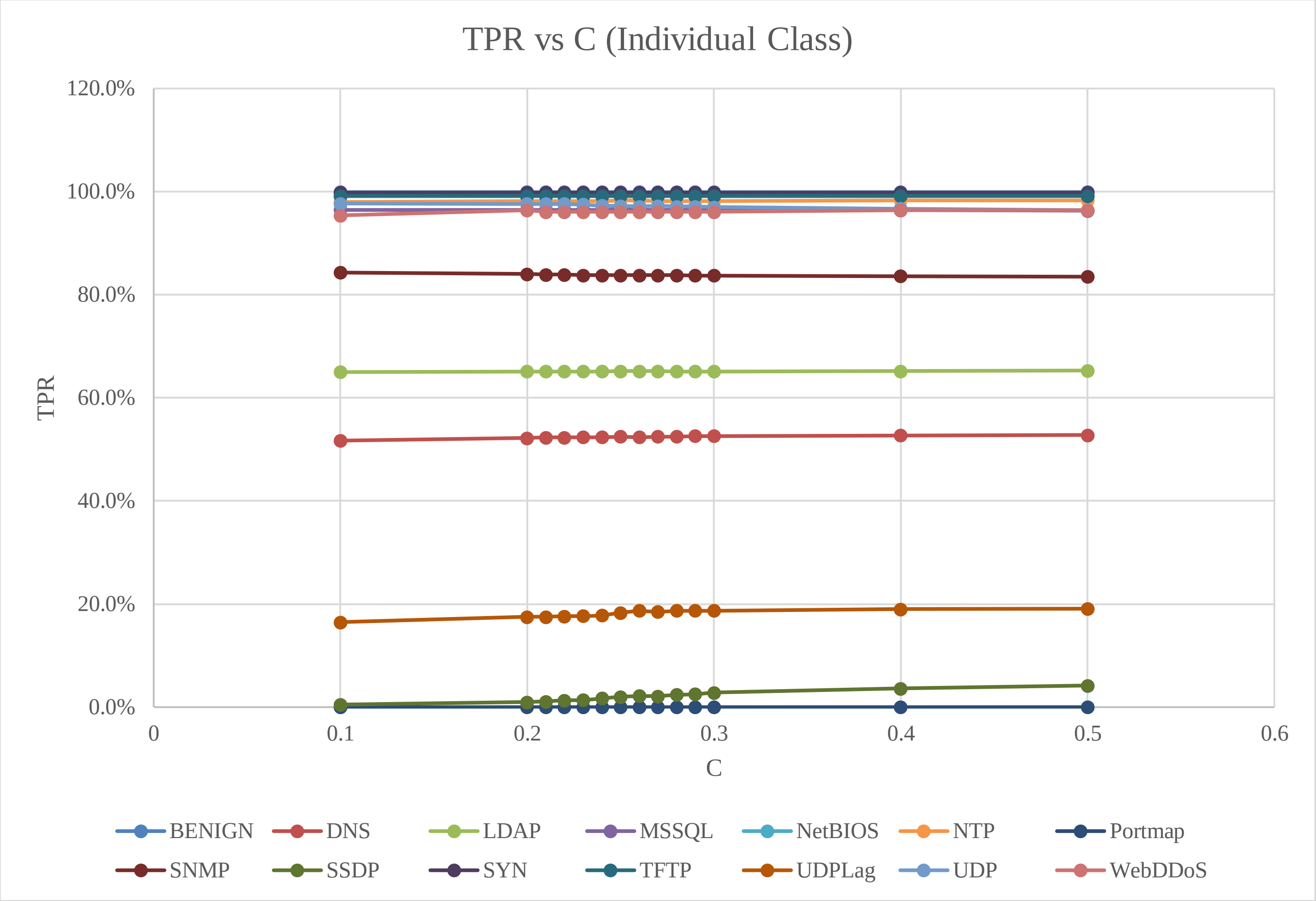


Figure A-7: Plot of TPR values for each individual class for values of C tested using WEKA for Preliminary Ungrouped Classification.

Table A-3: Table of Precision (PREC) values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	Precision (PREC) for each class (%)														Net Change For PREC (%)
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
0.1	99.8	76.1	56.9	95.0	94.5	97.9	0.0	72.8	36.8	94.2	99.9	94.5	72.4	92.3	-7.6
0.2	99.9	76.0	57.0	95.2	94.6	97.8	0.0	73.0	42.2	94.3	99.9	95.3	72.5	92.2	-0.8
0.21	99.9	76.0	57.0	95.3	94.6	97.8	0.0	73.0	41.5	94.3	99.9	95.3	72.5	92.5	-1.1
0.22	99.9	75.9	57.0	95.3	94.6	97.8	0.0	73.0	43.5	94.3	99.9	95.3	72.5	92.5	+0.8
0.23	99.9	75.8	57.0	95.3	94.6	97.8	0.0	73.1	43.1	94.3	99.9	95.4	72.5	92.5	+0.5
0.24	99.9	75.8	57.0	95.3	94.6	97.8	0.0	73.1	41.6	94.3	99.9	94.9	72.6	92.5	-1.4
0.25 (Default Value)	**99.9**	**75.8**	**57.0**	**95.3**	**94.6**	**97.8**	**0.0**	**73.1**	**42.9**	**94.3**	**99.9**	**95.0**	**72.6**	**92.5**	**0.0**
0.26	99.9	75.8	57.0	95.4	94.6	97.8	0.0	73.1	42.5	94.2	99.9	94.7	72.6	92.5	-0.7
0.27	99.9	75.7	57.0	95.4	94.6	97.8	0.0	73.1	42.7	94.2	99.9	94.6	72.6	92.3	-0.9
0.28	99.9	75.7	57.0	95.4	94.6	97.8	0.0	73.1	43.3	94.2	99.9	94.7	72.7	92.3	-0.1
0.29	99.9	75.7	57.0	95.4	94.6	97.8	0.0	73.2	43.8	94.2	99.9	94.2	72.7	92.3	0.0
0.3	99.9	75.7	57.0	95.4	94.6	97.8	0.0	73.1	43.9	94.2	99.9	92.5	72.7	92.3	-1.7
0.4	99.9	75.6	56.9	95.5	94.6	97.7	0.0	73.3	44.2	94.3	99.9	91.3	72.7	92.6	-2.2
0.5	99.9	75.4	56.9	95.6	94.6	97.8	0.0	73.4	43.1	94.2	99.9	88.9	72.7	92.6	-5.7

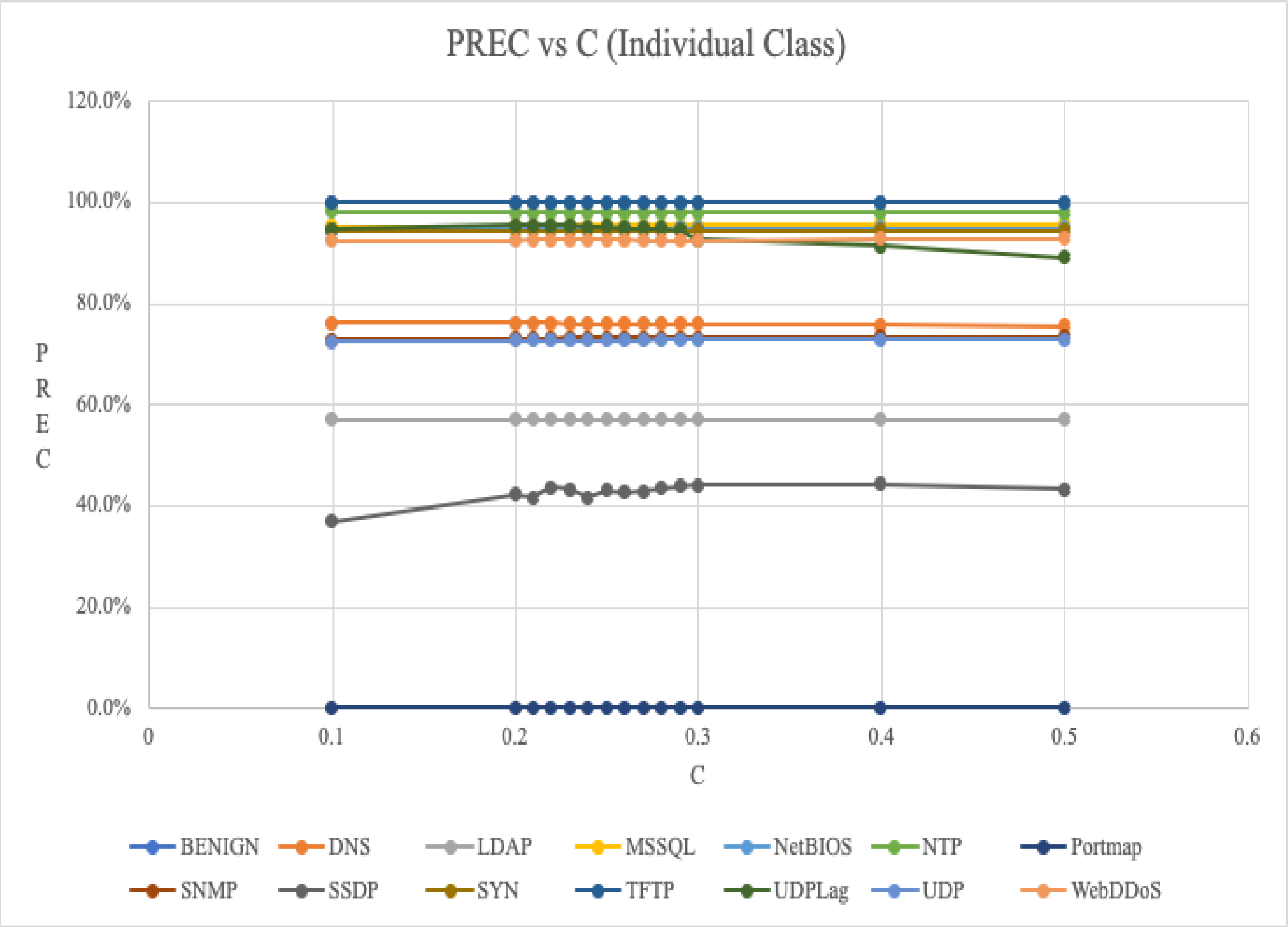


Figure A-8: Plot of PREC values for each class by value of C using WEKA for Preliminary Ungrouped Classification.

Table A-4: Table of F1-measure (F1) values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	F1 for each class														Net Change for F1
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Portmap	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
0.1	0.999	0.616	0.607	0.958	0.968	0.979	0.000	0.781	0.010	0.969	0.996	0.282	0.832	0.938	-0.065
0.2	0.999	0.619	0.608	0.959	0.968	0.980	0.000	0.781	0.020	0.970	0.996	0.296	0.832	0.942	-0.030
0.21	0.999	0.619	0.608	0.959	0.968	0.980	0.000	0.781	0.021	0.970	0.996	0.296	0.832	0.943	-0.028
0.22	0.999	0.619	0.608	0.959	0.968	0.980	0.000	0.781	0.024	0.970	0.996	0.297	0.832	0.943	-0.024
0.23	0.999	0.620	0.608	0.959	0.968	0.980	0.000	0.781	0.027	0.970	0.996	0.298	0.832	0.943	-0.019
0.24	0.999	0.620	0.608	0.959	0.968	0.980	0.000	0.781	0.033	0.970	0.996	0.300	0.831	0.943	-0.012
0.25 (Default Value)	**0.999**	**0.620**	**0.608**	**0.959**	**0.968**	**0.980**	**0.000**	**0.781**	**0.039**	**0.970**	**0.996**	**0.306**	**0.831**	**0.943**	**0.000**
0.26	0.999	0.620	0.608	0.959	0.968	0.980	0.000	0.781	0.041	0.969	0.996	0.313	0.831	0.943	+0.008
0.27	0.999	0.620	0.608	0.959	0.968	0.980	0.000	0.781	0.041	0.969	0.996	0.310	0.831	0.942	+0.004
0.28	0.999	0.620	0.608	0.960	0.968	0.980	0.000	0.781	0.046	0.969	0.996	0.313	0.831	0.942	+0.013
0.29	0.999	0.621	0.608	0.960	0.968	0.980	0.000	0.781	0.047	0.969	0.996	0.313	0.831	0.942	+0.015
0.3	0.999	0.615	0.608	0.960	0.968	0.980	0.000	0.781	0.052	0.969	0.996	0.312	0.831	0.942	+0.013
0.4	0.999	0.621	0.608	0.960	0.968	0.980	0.000	0.781	0.066	0.969	0.996	0.315	0.830	0.944	+0.037
0.5	0.999	0.621	0.608	0.960	0.968	0.980	0.000	0.781	0.077	0.969	0.996	0.315	0.829	0.944	+0.047

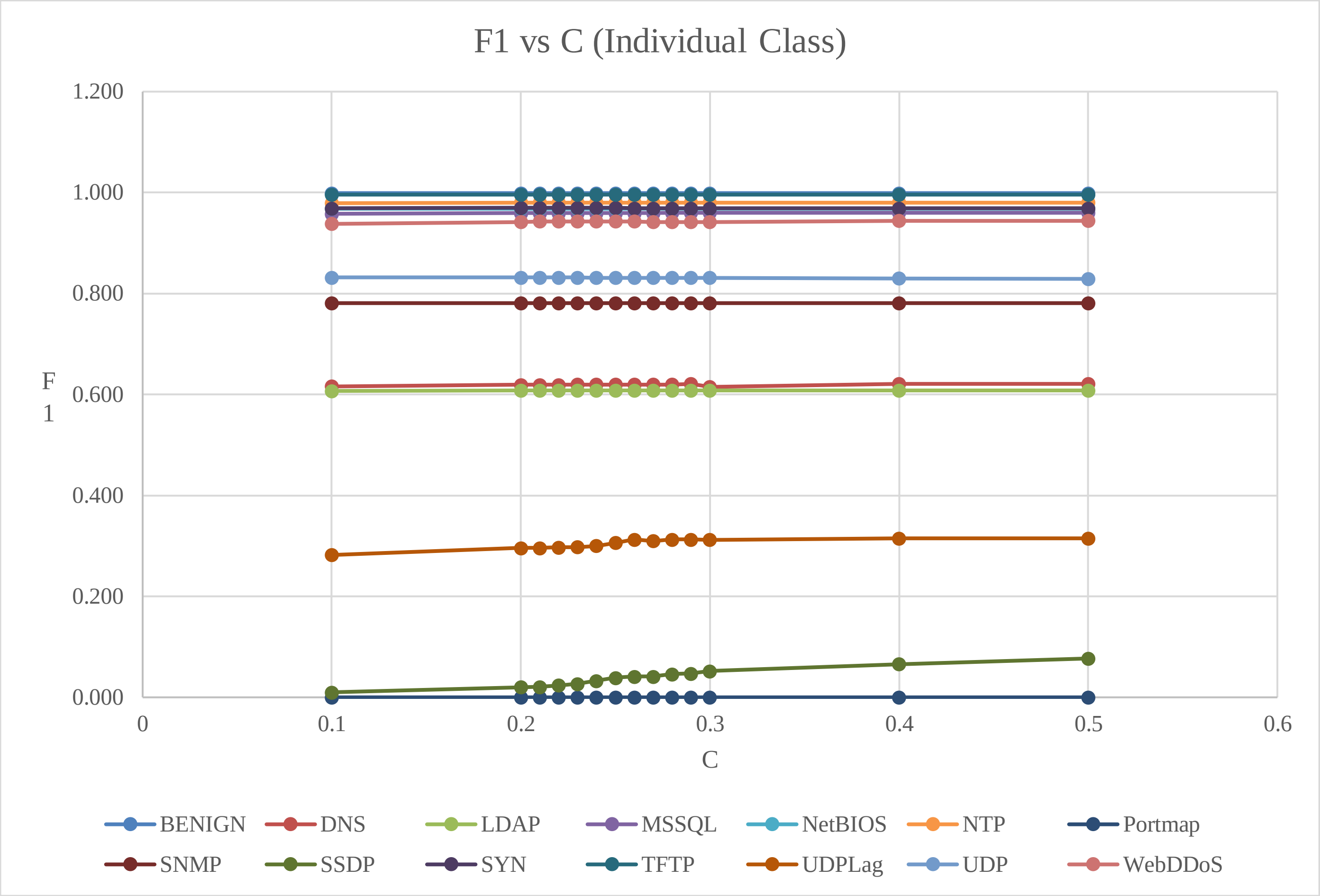


Figure A-9: Plot of F1 values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification.

Table A-5: Table of Area under ROC (AUC) values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	AUC for each Class														Net Change (AUC)
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
0.1	1.000	0.973	0.973	0.995	0.996	0.999	0.955	0.981	0.957	0.999	1.000	0.975	0.984	0.994	-0.001
0.2	1.000	0.973	0.973	0.995	0.996	0.999	0.959	0.981	0.957	0.999	1.000	0.974	0.984	0.993	+0.001
0.21	1.000	0.973	0.972	0.995	0.996	0.999	0.959	0.981	0.957	0.999	1.000	0.974	0.984	0.993	0.000
0.22	1.000	0.973	0.972	0.995	0.996	0.999	0.959	0.981	0.957	0.999	1.000	0.974	0.984	0.993	0.000
0.23	1.000	0.973	0.972	0.995	0.996	0.998	0.959	0.981	0.957	0.999	1.000	0.974	0.984	0.993	-0.001
0.24	1.000	0.973	0.972	0.995	0.996	0.998	0.959	0.980	0.958	0.999	1.000	0.974	0.984	0.993	-0.001
0.25 (Default value)	**1.000**	**0.973**	**0.972**	**0.995**	**0.996**	**0.998**	**0.959**	**0.980**	**0.959**	**0.999**	**1.000**	**0.973**	**0.985**	**0.993**	**0.000**
0.26	1.000	0.972	0.972	0.995	0.996	0.998	0.959	0.980	0.960	0.999	1.000	0.974	0.985	0.991	-0.001
0.27	1.000	0.972	0.972	0.995	0.996	0.998	0.959	0.980	0.960	0.999	1.000	0.974	0.985	0.991	-0.001
0.28	1.000	0.972	0.972	0.995	0.996	0.998	0.959	0.980	0.960	0.999	1.000	0.975	0.985	0.991	0.000
0.29	1.000	0.972	0.972	0.995	0.996	0.998	0.959	0.980	0.961	0.999	1.000	0.974	0.985	0.991	0.000
0.3	1.000	0.972	0.972	0.995	0.996	0.998	0.959	0.980	0.962	0.999	1.000	0.973	0.986	0.991	+0.001
0.4	1.000	0.972	0.972	0.995	0.996	0.998	0.958	0.980	0.962	0.999	1.000	0.972	0.986	0.989	-0.003
0.5	1.000	0.972	0.971	0.995	0.997	0.998	0.962	0.980	0.963	0.999	1.000	0.973	0.985	0.990	+0.003

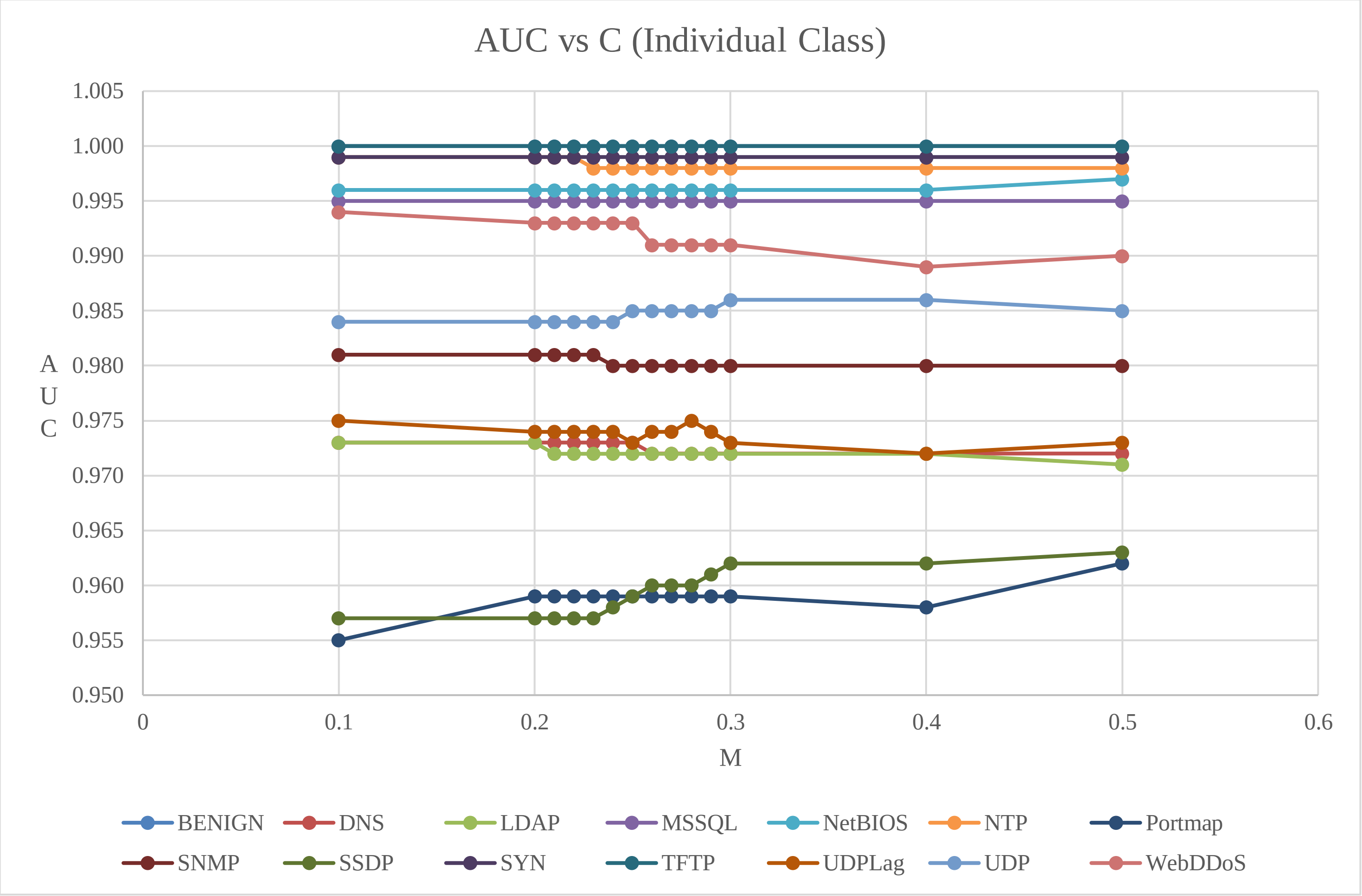


Figure A-10: Plot of AUC values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification.

Table A-6: Table of False Positive Rate (FPR) values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes lower than value obtained with default value of C = 0.25, while light red denotes higher than default value obtained)

Value Tested for C	FPR for each Class													
	BENIGN	DNS	LDAP	MSSQL	NetBIOS	NTP	Portmap	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS
0.1	0.1%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.9%	0.0%	0.4%	0.0%	0.0%	3.0%	0.0%
0.2	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.0%	0.4%	0.0%	0.0%	3.0%	0.0%
0.21	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.0%	0.4%	0.0%	0.0%	3.0%	0.0%
0.22	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.0%	0.4%	0.0%	0.0%	3.0%	0.0%
0.23	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	3.0%	0.0%
0.24	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	3.0%	0.0%
0.25 (Default Value)	**0.0%**	**0.9%**	**2.3%**	**0.6%**	**0.5%**	**0.0%**	**0.0%**	**1.8%**	**0.1%**	**0.4%**	**0.0%**	**0.0%**	**3.0%**	**0.0%**
0.26	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	3.0%	0.0%
0.27	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	3.0%	0.0%
0.28	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
0.29	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
0.3	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
0.4	0.0%	1.0%	2.3%	0.5%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
0.5	0.0%	1.0%	2.3%	0.5%	0.5%	0.0%	0.0%	1.8%	0.2%	0.4%	0.0%	0.0%	2.9%	0.0%

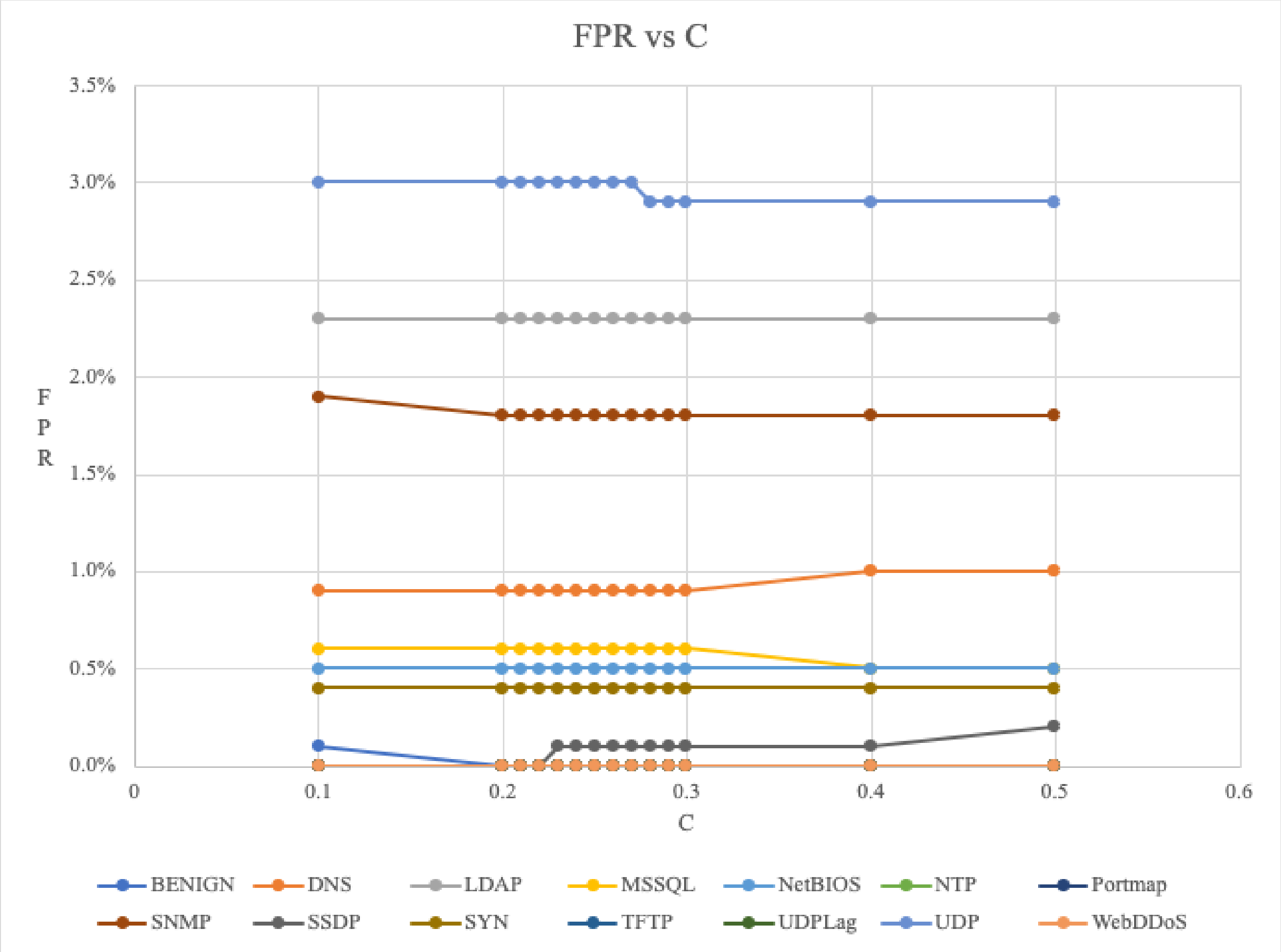


Figure A-11: Plot of FPR values for each class for each value of C tested using WEKA for Preliminary Ungrouped Classification.

Table A-7: Table of G-Mean (GMEAN) values for each class for each value of C tested for Preliminary Ungrouped Classification. (Light green highlight denotes higher than default value or 0.25, while light red denotes lower than default value)

Value Tested for C	G-Mean value for each Class														Net Change For GMEAN
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
0.1	0.9990	0.7158	0.7969	0.9794	0.9930	0.9899	0.0000	0.9094	0.0707	0.9970	0.9960	0.4062	0.9735	0.9767	-0.1003
0.2	0.9995	0.7192	0.7975	0.9794	0.9930	0.9910	0.0000	0.9082	0.1000	0.9970	0.9960	0.4183	0.9730	0.9818	-0.0499
0.21	0.9995	0.7199	0.7975	0.9794	0.9930	0.9910	0.0000	0.9077	0.1049	0.9970	0.9960	0.4183	0.9730	0.9803	-0.0464
0.22	0.9995	0.7199	0.7975	0.9794	0.9930	0.9910	0.0000	0.9077	0.1140	0.9970	0.9960	0.4195	0.9730	0.9803	-0.0361
0.23	0.9995	0.7206	0.7975	0.9794	0.9930	0.9910	0.0000	0.9071	0.1183	0.9970	0.9960	0.4207	0.9725	0.9803	-0.0310
0.24	0.9995	0.7206	0.7975	0.9794	0.9930	0.9910	0.0000	0.9071	0.1303	0.9970	0.9960	0.4219	0.9715	0.9803	-0.0187
0.25 (Default Value)	**0.9995**	**0.7213**	**0.7981**	**0.9799**	**0.9930**	**0.9915**	**0.0000**	**0.9071**	**0.1414**	**0.9970**	**0.9960**	**0.4278**	**0.9710**	**0.9803**	**0.0000**
0.26	0.9995	0.7206	0.7981	0.9794	0.9930	0.9915	0.0000	0.9071	0.1482	0.9970	0.9960	0.4324	0.9705	0.9803	+0.0099
0.27	0.9995	0.7213	0.7981	0.9794	0.9930	0.9910	0.0000	0.9071	0.1448	0.9970	0.9960	0.4301	0.9710	0.9803	+0.0048
0.28	0.9995	0.7213	0.7975	0.9794	0.9930	0.9910	0.0000	0.9071	0.1548	0.9970	0.9960	0.4324	0.9710	0.9803	+0.0165
0.29	0.9995	0.7220	0.7975	0.9799	0.9930	0.9910	0.0000	0.9066	0.1580	0.9970	0.9960	0.4324	0.9710	0.9803	+0.0204
0.3	0.9995	0.7220	0.7975	0.9794	0.9930	0.9910	0.0000	0.9066	0.1672	0.9970	0.9960	0.4324	0.9705	0.9803	+0.0286
0.4	0.9995	0.7223	0.7981	0.9794	0.9930	0.9915	0.0000	0.9061	0.1896	0.9970	0.9960	0.4359	0.9690	0.9818	+0.0553
0.5	0.9995	0.7230	0.7987	0.9789	0.9930	0.9915	0.0000	0.9055	0.2047	0.9970	0.9960	0.4370	0.9675	0.9818	+0.0703

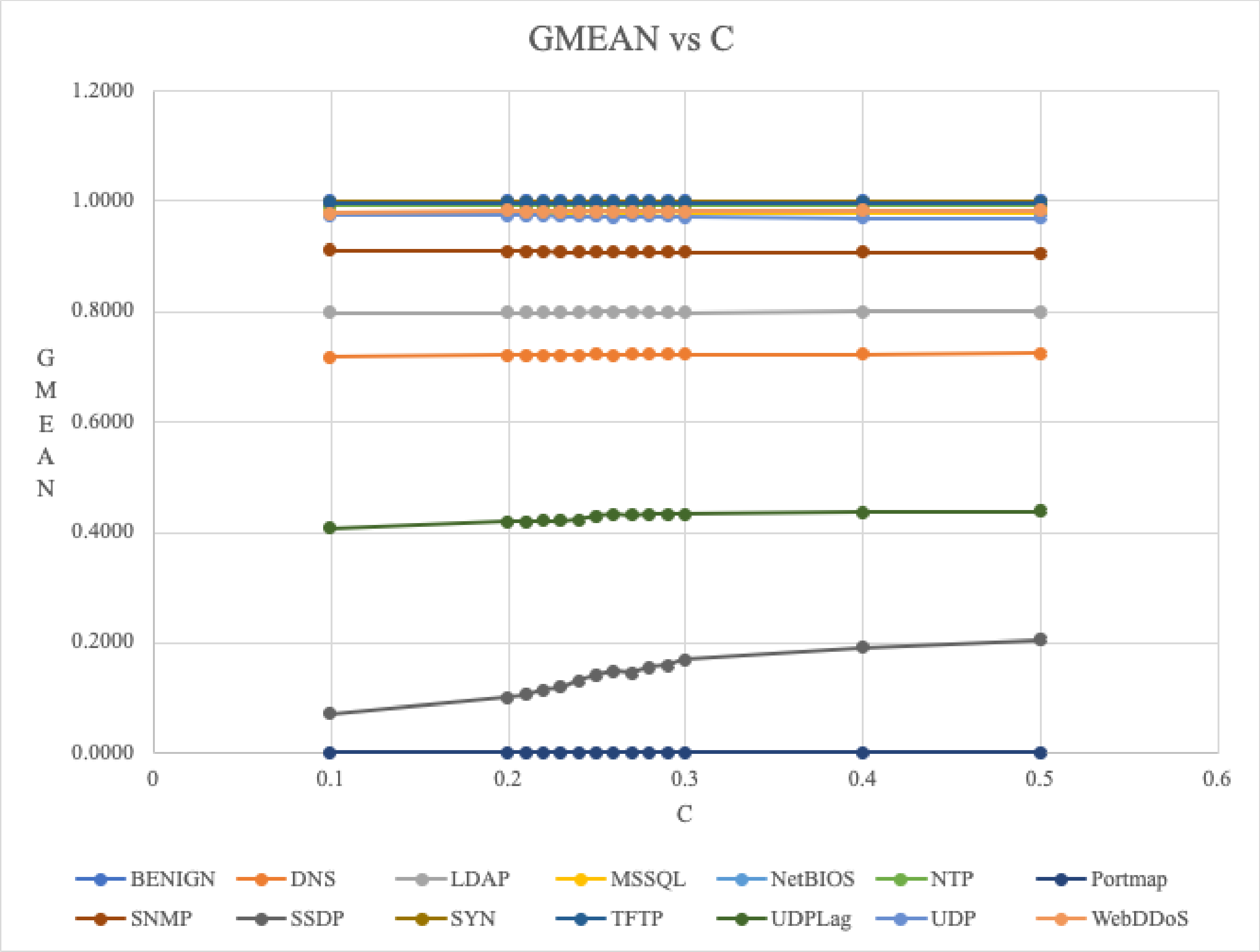


Figure A-12: Plot of GMEAN values for each class for each value of C tested for Preliminary Ungrouped Classification.

Table A-8: Table of Net Change values for every evaluation metric used by value of C tested for Preliminary Ungrouped Classification.

Value Of C Tested	Net Change				
	TPR	PREC	F1	AUC	GMEAN
0.1	-4.4%	-7.6%	-0.065	-0.001	-0.1003
0.2	-1.5%	-0.8%	-0.030	+0.001	-0.0499
0.21	-1.7%	-1.1%	-0.028	0.000	-0.0464
0.22	-1.4%	+0.8%	-0.024	0.000	-0.0361
0.23	-1.3%	+0.5%	-0.019	-0.001	-0.0310
0.24	-1.1%	-1.4%	-0.012	-0.001	-0.0187
0.25 (Default Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
0.26	+0.3%	-0.7%	+0.008	-0.001	+0.0099
0.27	+0.1%	-0.9%	+0.004	-0.001	+0.0048
0.28	+0.4%	-0.1%	+0.013	0.000	+0.0165
0.29 (Selected value)	**+0.6%**	**0.0%**	**+0.015**	**0.000**	**+0.0204**
0.3	+0.7%	-1.7%	+0.013	+0.001	+0.0286
0.4	+1.9%	-2.2%	+0.037	-0.003	+0.0553
0.5	+2.3%	-5.7%	+0.047	+0.003	+0.0703

Optimisation by Minimum Number of Objects (-M, default value of 2, with C = 0.29)

Table A-9: Table of General Evaluation Metrics (Weighted Average) Obtained for each value of M tested in WEKA (with C = 0.29) for Preliminary Ungrouped Classification using 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value of 0.25, while light red denotes lower than default value)

Parameter Tested	Evaluation Metric (Weighted Average)							Standard Deviation					
minNumObj (-M)	ACC (%)	TPR (%)	PREC (%)	F1	AUC	TNR (%)	GMEAN	ACC (%)	TPR (%)	PREC	F1	AUC	TNR (%)
1	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
2 (Default Value)	90.93	91	90	0.90	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00
3	90.92	91	90	0.90	0.99	99	0.9492	0.12	0.00	0.00	0.00	0.00	0.00
4	90.92	91	■	■	0.99	99	0.9492	0.12	0.00	0.00	0.00	0.00	0.00
5	90.90	91	■	■	0.99	99	0.9492	0.13	0.00	0.00	0.00	0.00	0.00

*Black cells denote no values of PREC and F1 obtained with M = 4 and 5.

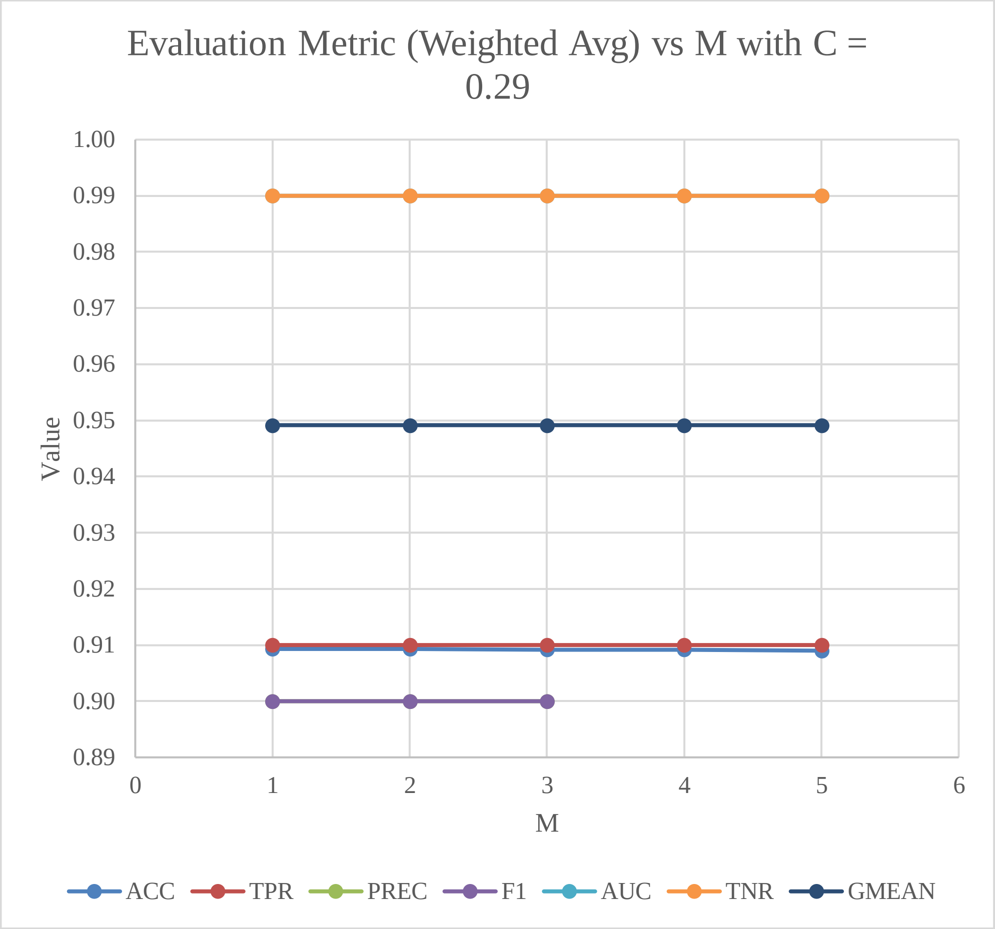


Figure A-13: Graph of General Weighted Average Evaluation Metrics value by value of M tested in WEKA (with C = 0.29) for Preliminary Ungrouped Classification.

Table A-10: Table of True Positive Rates (TPR) obtained for each class for each value of M tested (with C = 0.29) in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value of M tested	TPR for Each Class														Net Change for TPR (%)
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
1	99.9	52.6	65.1	96.5	99.2	98.4	0.0	83.7	2.3	99.8	99.2	18.7	97.2	96.4	0.4
2 (Default Value)	**99.9**	**52.6**	**65.1**	**96.6**	**99.1**	**98.2**	**0.0**	**83.7**	**2.5**	**99.8**	**99.2**	**18.7**	**97.1**	**96.1**	**0.0**
3	99.9	52.6	65.2	96.6	99.1	98.2	0.0	83.7	2.4	99.8	99.2	18.6	97.1	95.4	-0.8
4	99.9	52.6	65.1	96.6	99.1	98.1	0.0	83.8	2.4	99.8	99.2	18.7	97.0	95.7	-0.6
5	99.9	52.5	65.0	96.5	99.1	98.0	0.0	83.9	2.5	99.8	99.2	18.7	96.9	95.0	-1.6

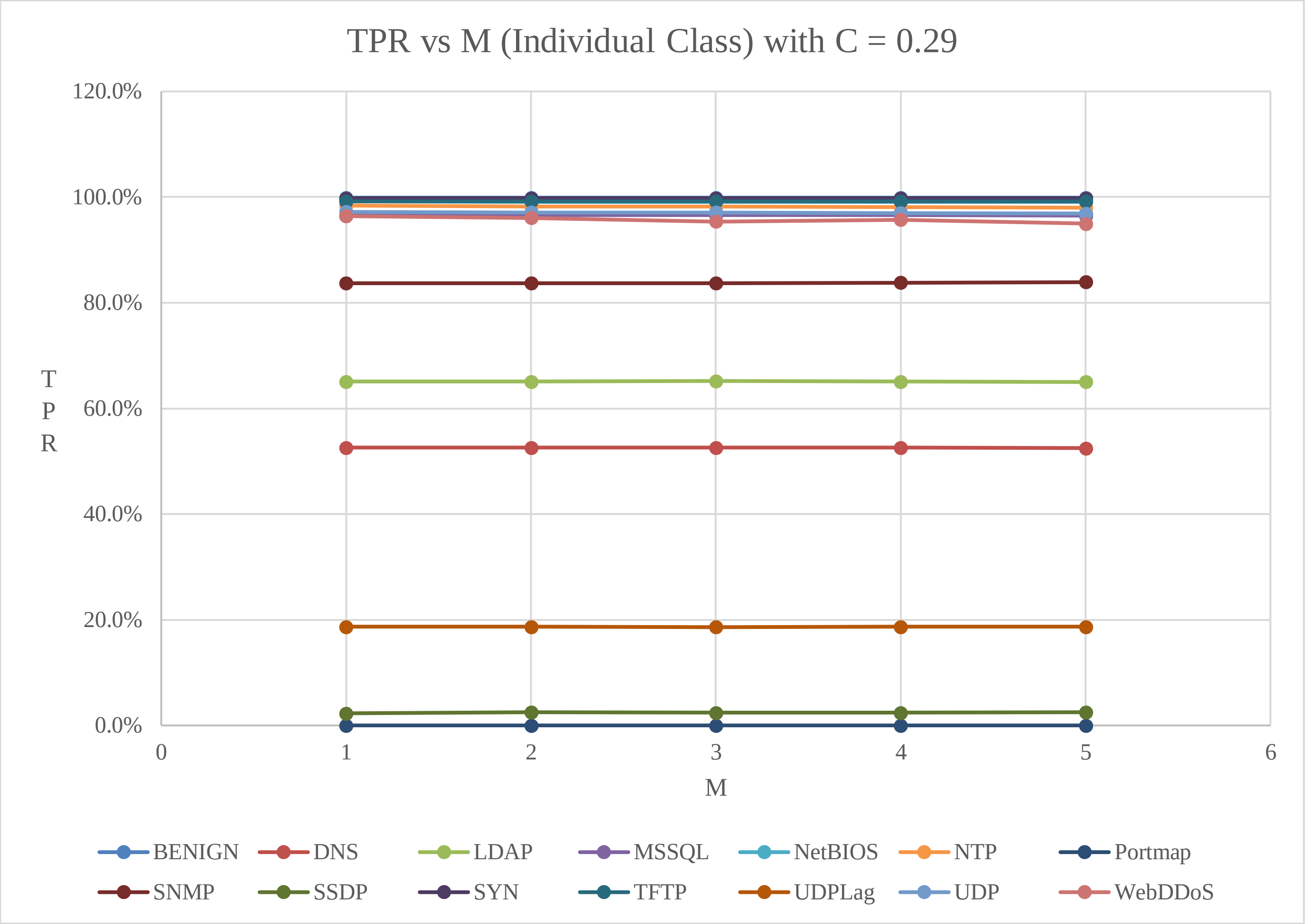


Figure A-14: Plot of TPR values for each individual class for values of M tested (with C = 0.29) using WEKA for Preliminary Ungrouped Classification.

Table A-11: Table of Precision (PREC) values obtained for each class for each value of M tested (with C = 0.29) in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value of M tested	PREC for Each Class														Net Change for PREC (%)
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
1	99.9	75.7	57.0	95.4	94.6	97.8	0.0	73.1	42.7	94.3	99.9	91.2	72.6	92.0	-4.5
2 (Default Value)	**99.9**	**75.7**	**57.0**	**95.4**	**94.6**	**97.8**	**0.0**	**73.2**	**43.8**	**94.2**	**99.9**	**94.2**	**72.7**	**92.3**	**0.0**
3	99.9	75.8	57.0	95.3	94.6	97.9	0.0	73.2	44.9	94.3	99.9	92.9	72.7	92.1	-0.2
4	99.9	75.8	57.1	95.3	94.6	97.6	■	73.2	41.9	94.3	99.9	91.2	72.7	92.1	-5.1
5	99.8	75.7	57.0	95.3	94.6	97.9	■	73.1	41.9	94.2	99.9	91.2	72.7	92.1	-5.3

*Black cells denote no values of PREC for Portmap class obtained with M = 4 and 5.

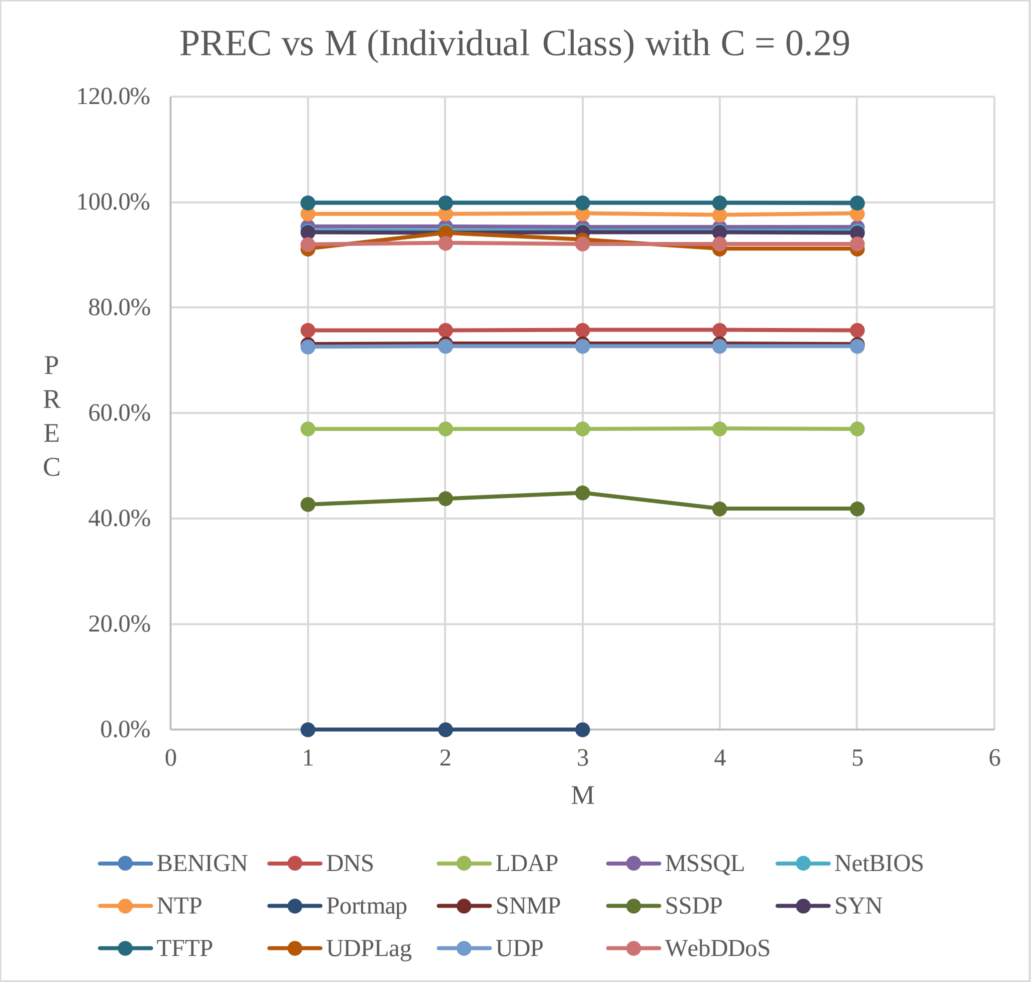


Figure A-15: Plot of PREC values for each individual class for values of M tested (with C = 0.29) using WEKA for Preliminary Ungrouped Classification.

Table A-12: Table of F-Measure (F1) values obtained for each class for each value of M tested (with C = 0.29) in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value of M tested	F1 values for Each Class														Net Change for F1
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
1	0.999	0.621	0.608	0.960	0.968	0.981	0.000	0.781	0.044	0.970	0.996	0.311	0.831	0.941	-0.004
2 (Default Value)	**0.999**	**0.621**	**0.608**	**0.960**	**0.968**	**0.980**	**0.000**	**0.781**	**0.047**	**0.969**	**0.996**	**0.313**	**0.831**	**0.942**	**0.000**
3	0.999	0.621	0.608	0.960	0.968	0.980	0.000	0.781	0.045	0.970	0.996	0.311	0.831	0.937	-0.008
4	0.999	0.621	0.608	0.959	0.968	0.979	■	0.781	0.045	0.970	0.996	0.311	0.831	0.939	-0.008
5	0.999	0.620	0.608	0.959	0.968	0.979	■	0.781	0.047	0.969	0.996	0.311	0.831	0.935	-0.012

*Black cells denote no values of F1 for Portmap class obtained with M = 4 and 5.

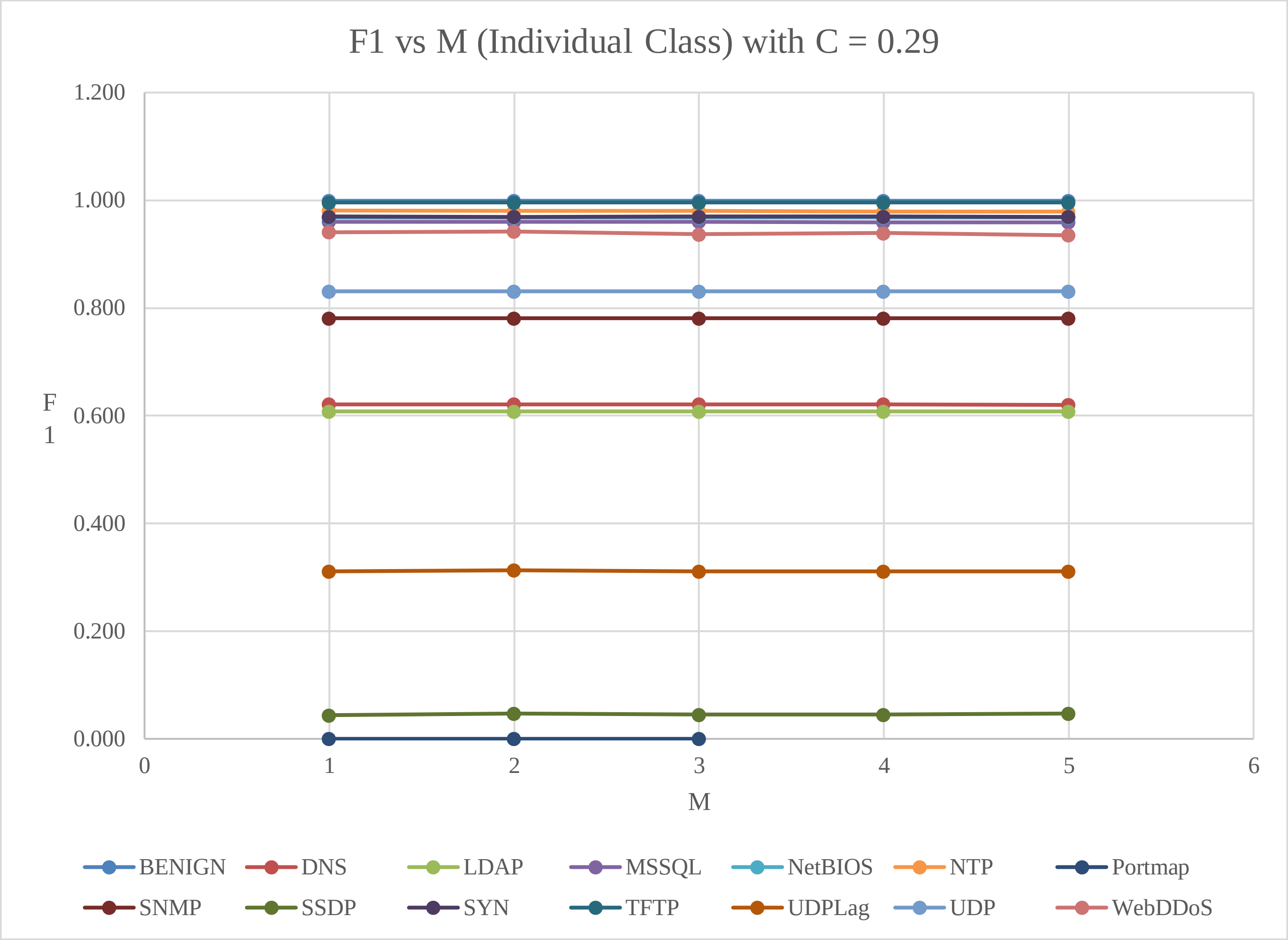


Figure A-16: Plot of F1 values for each individual class for values of M tested (with C = 0.29) using WEKA for Preliminary Ungrouped Classification.

Table A-13: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of M tested (with C = 0.29) in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value of M tested	AUC Values for Each Class														Net Change for AUC
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
1	1.000	0.971	0.972	0.995	0.996	0.998	0.959	0.979	0.961	0.999	1.000	0.973	0.985	0.990	-0.004
2 (Default Value)	**1.000**	**0.972**	**0.972**	**0.995**	**0.996**	**0.998**	**0.959**	**0.980**	**0.961**	**0.999**	**1.000**	**0.974**	**0.985**	**0.991**	**0.000**
3	1.000	0.973	0.973	0.995	0.997	0.999	0.959	0.981	0.961	0.999	1.000	0.975	0.986	0.994	+0.010
4	1.000	0.974	0.973	0.995	0.997	0.999	0.959	0.981	0.962	0.999	1.000	0.976	0.986	0.994	+0.013
5	1.000	0.974	0.974	0.995	0.996	0.999	0.959	0.981	0.963	0.999	1.000	0.976	0.986	0.994	+0.014

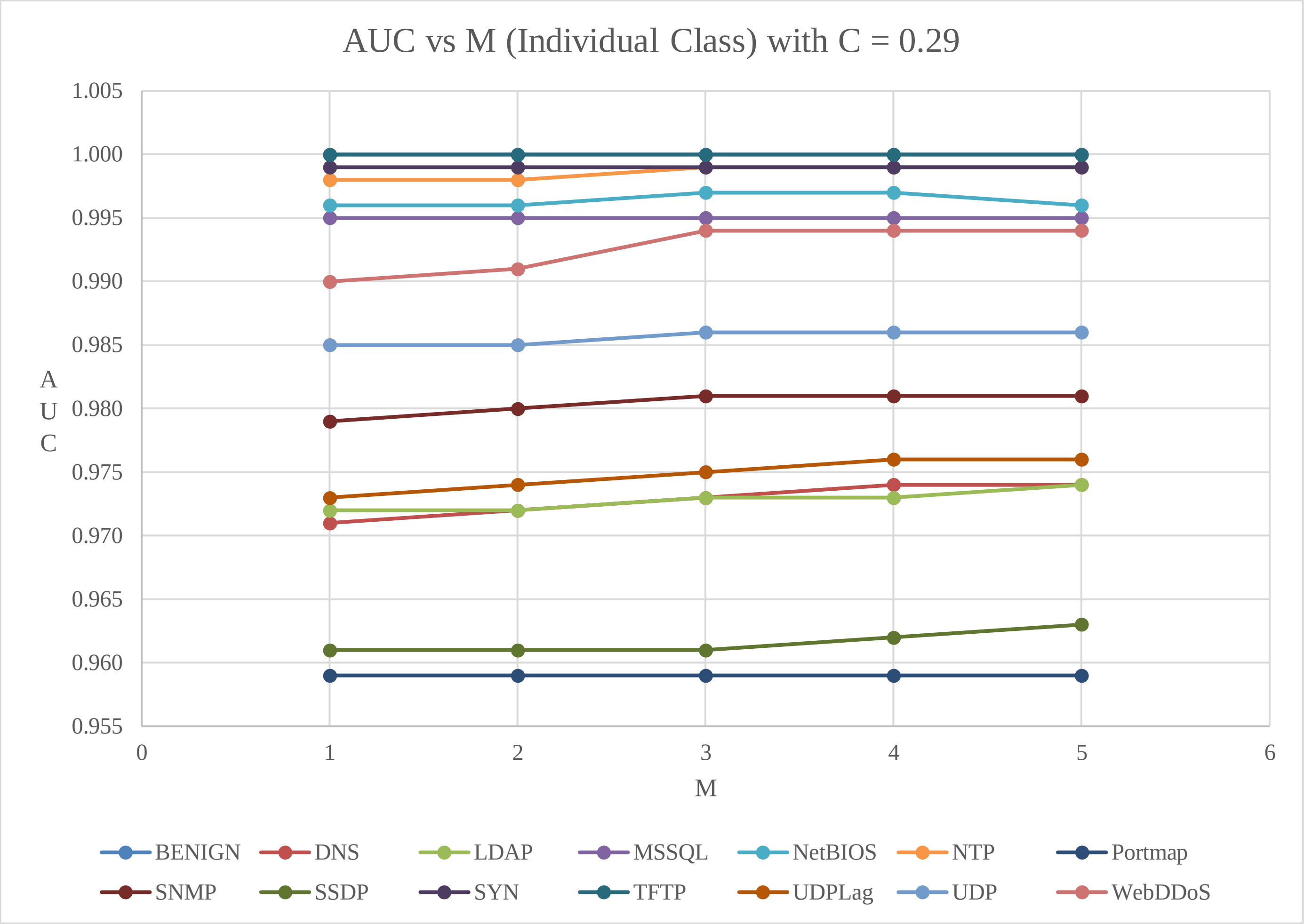


Figure A-17: Plot of AUC values for each individual class for values of M tested (with C = 0.29) using WEKA for Preliminary Ungrouped Classification.

Table A-14: Table of False Positive Rate (FPR) values obtained for each class for each value of M tested (with C = 0.29) in WEKA for Preliminary Ungrouped Classification. (Light green highlight denotes lower than value obtained with default value of M = 2, while light red denotes higher than default value obtained)

Value of M tested	FPR for Each Class													
	BENIGN	DNS	LDAP	MSSQL	NetBIOS	NTP	Portmap	SNMP	SSDP	SYN	TFTP	UDPLag	UDP	WebDDoS
1	0.0%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	3.0%	0.0%
2 (Default Value)	**0.0%**	**0.9%**	**2.3%**	**0.6%**	**0.5%**	**0.0%**	**0.0%**	**1.8%**	**0.1%**	**0.4%**	**0.0%**	**0.0%**	**2.9%**	**0.0%**
3	0.1%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
4	0.1%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%
5	0.1%	0.9%	2.3%	0.6%	0.5%	0.0%	0.0%	1.8%	0.1%	0.4%	0.0%	0.0%	2.9%	0.0%

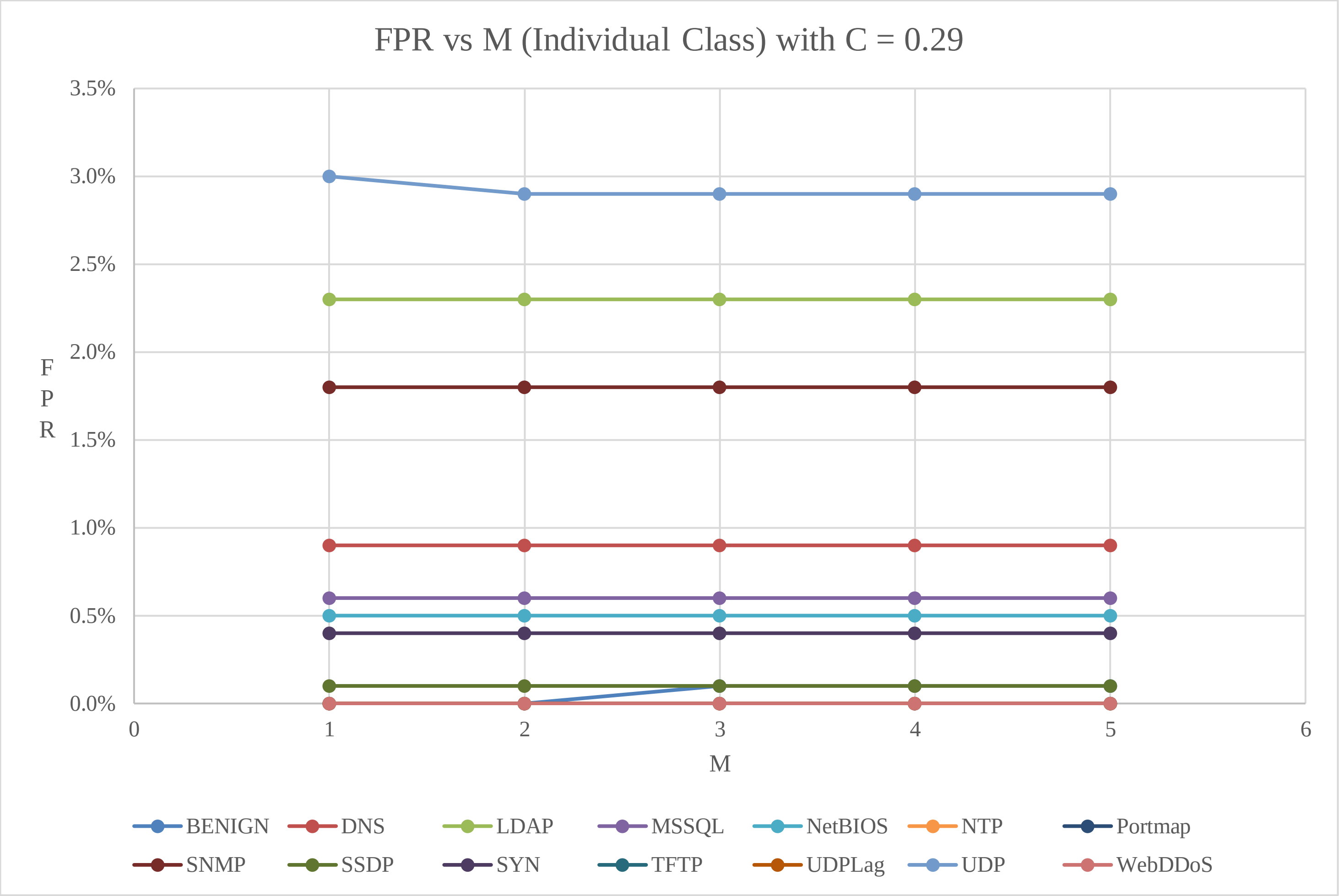


Figure A-18: Plot of FPR values for each individual class for values of M tested (with C = 0.29) using WEKA for Preliminary Ungrouped Classification.

Table A-15: Table of G-Mean (GMEAN) values obtained for each class for each value of M tested (with C = 0.29) for Preliminary Ungrouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value of M tested	GMEAN Values for Each Class														Net Change for GMEAN
	BENIGN	DNS	LDAP	MSSQL	Net BIOS	NTP	Port map	SNMP	SSDP	SYN	TFTP	UDP Lag	UDP	Web DDoS	
1	0.9995	0.7220	0.7975	0.9794	0.9935	0.9920	0.0000	0.9066	0.1516	0.9970	0.9960	0.4324	0.9710	0.9818	-0.0039
2	**0.9995**	**0.7220**	**0.7975**	**0.9799**	**0.9930**	**0.9910**	**0.0000**	**0.9066**	**0.1580**	**0.9970**	**0.9960**	**0.4324**	**0.9710**	**0.9803**	**0.0000**
3	0.9990	0.7220	0.7981	0.9799	0.9930	0.9910	0.0000	0.9066	0.1548	0.9970	0.9960	0.4313	0.9710	0.9767	-0.0078
4	0.9990	0.7220	0.7975	0.9799	0.9930	0.9905	0.0000	0.9071	0.1548	0.9970	0.9960	0.4324	0.9705	0.9783	-0.0062
5	0.9990	0.7213	0.7969	0.9794	0.9930	0.9899	0.0000	0.9077	0.1580	0.9970	0.9960	0.4324	0.9700	0.9747	-0.0089

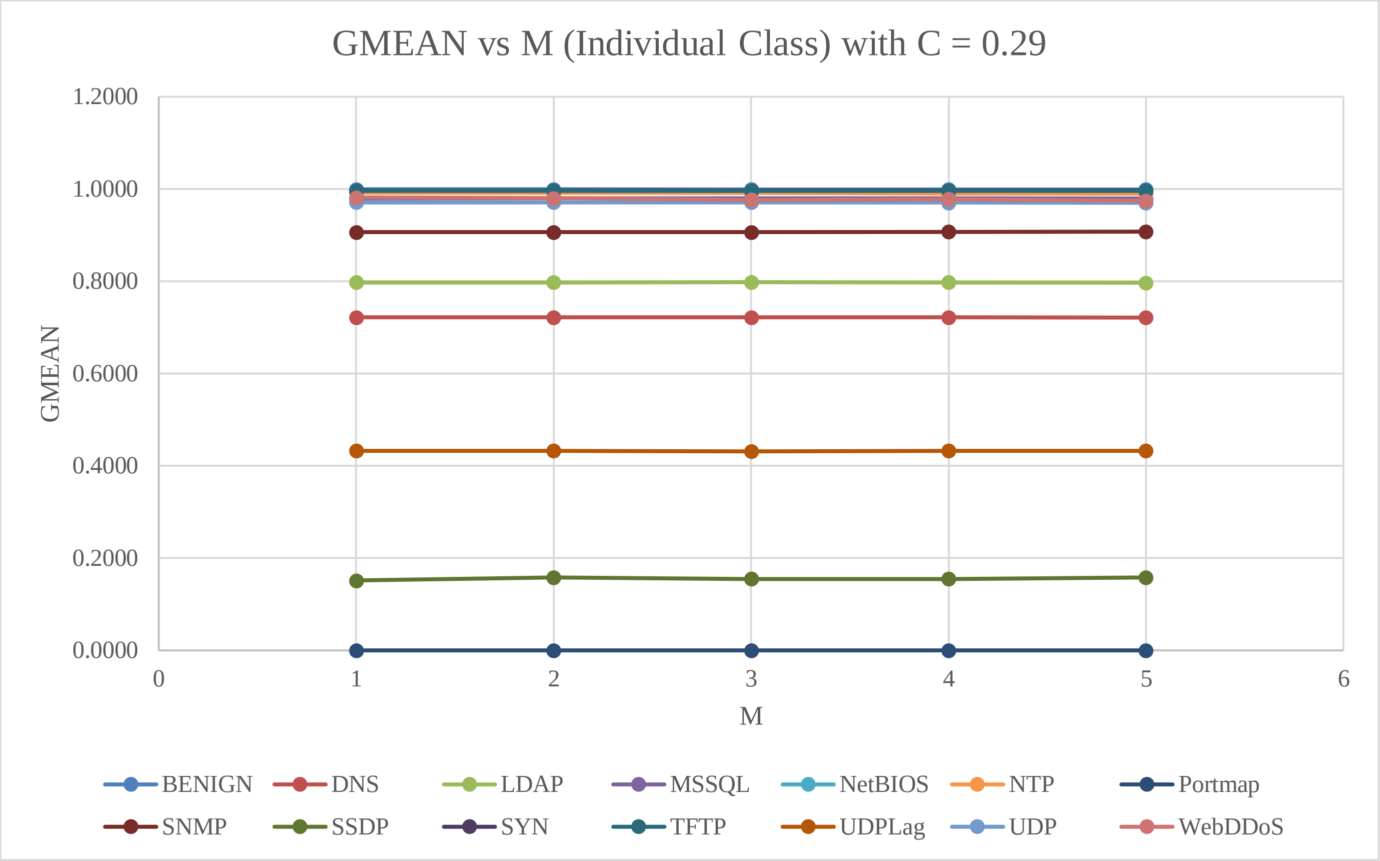


Figure A-19: Plot of GMEAN values for each individual class for Preliminary Classification by values of M tested (with C = 0.29) for Preliminary Ungrouped Classification.

Table A-16: Table of Net Change values for every evaluation metric used by value of M tested (using C = 0.29).

Value Of M Tested	Net Change				
	TPR	PREC	F1	AUC	GMEAN
1	0.4%	-4.5%	-0.004	-0.004	-0.0039
2 (Default & Selected Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
3	-0.8%	-0.2%	-0.008	+0.010	-0.0078
4	-0.6%	-5.1%	-0.008	+0.013	-0.0062
5	-1.6%	-5.3%	-0.012	+0.014	-0.0089

APPENDIX B: Hierarchical Grouped Classification Results

Level 0 Grouped Classification

Table B-1: Table of General Evaluation Metrics (Weighted Average) Obtained from Level 0 Grouped Classification and Optimisation for each value of C tested in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value of 0.25, while light red denotes lower than default value)

Value Tested For C	Evaluation Metric (Weighted Average)							Standard Deviation					
	ACC	TPR	PREC	F1	AUC	TNR	GMEAN	ACC	TPR	PREC	F1	AUC	TNR
0.1	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
0.2	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
0.25 (Default Value)	**99.96%**	**100%**	**100%**	**1.00**	**1.00**	**100%**	**1.0000**	**0.01**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
0.3	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
0.4	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
0.5	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00

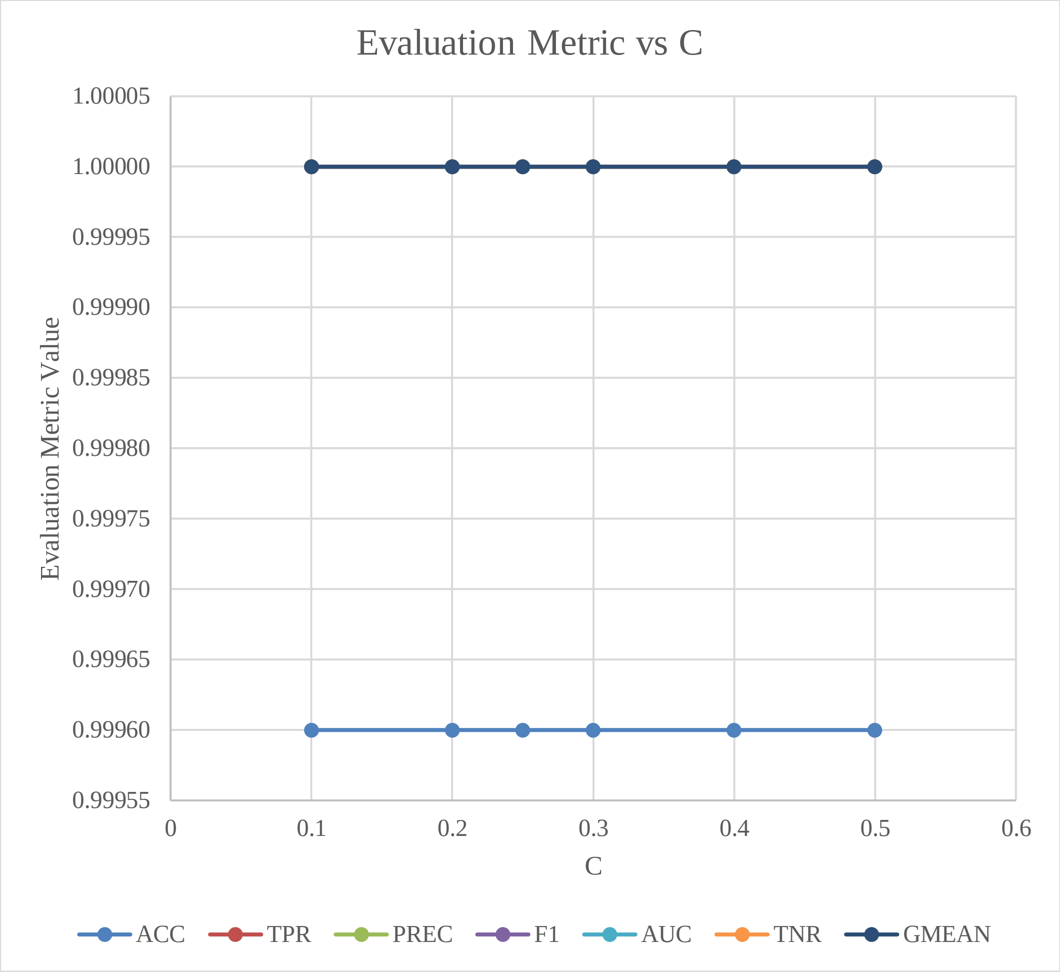


Figure B-1: Graph of General Weighted Average Evaluation Metrics value for Level 0 Grouped Classification by value of C tested in WEKA.

Table B-2: Table of True Positive Rates (TPR) obtained for each class for each value of C tested in WEKA for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	TPR for Class		Net Change (TPR)
	BENIGN	DDOS	
0.1	99.9%	100.0%	0.0%
0.2	99.9%	100.0%	0.0%
0.25 (Default Value)	**99.9%**	**100.0%**	**0.0%**
0.3	99.9%	100.0%	0.0%
0.4	99.9%	100.0%	0.0%
0.5	99.9%	100.0%	0.0%

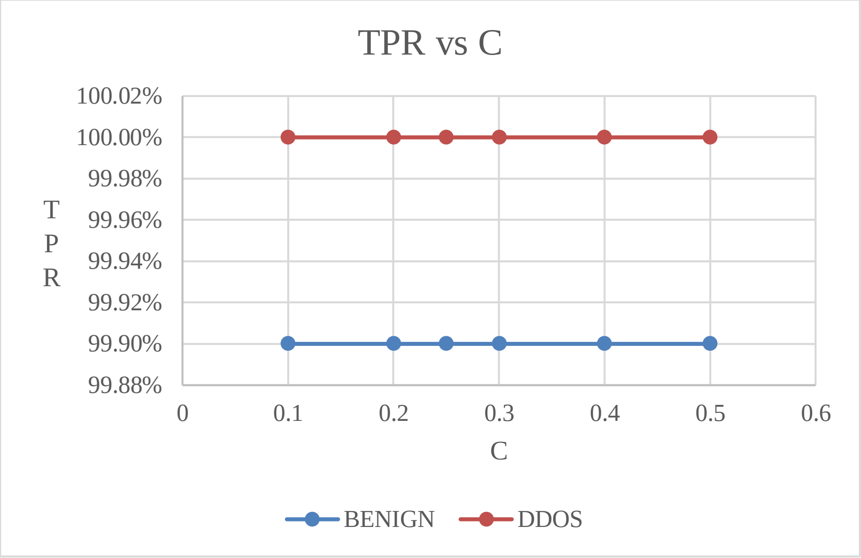


Figure B-2: Plot of TPR values for each individual class for values of C tested using WEKA for Level 0 Grouped Classification.

Table B-3: Table of Precision (PREC) values obtained for each class for each value of C tested in WEKA for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	PREC for Class		Net Change (PREC)
	BENIGN	DDOS	
0.1	99.9%	100.0%	0.0%
0.2	99.9%	100.0%	0.0%
0.25 (Default Value)	**99.9%**	**100.0%**	**0.0%**
0.3	99.9%	100.0%	0.0%
0.4	99.9%	100.0%	0.0%
0.5	99.9%	100.0%	0.0%

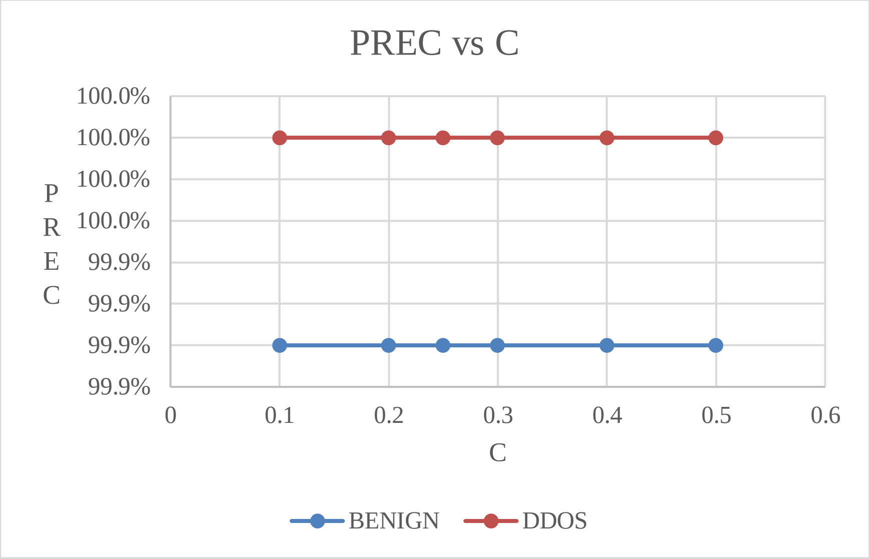


Figure B-3: Plot of PREC values for each individual class for values of C tested using WEKA for Level 0 Grouped Classification.

Table B-4: Table of F1-Measure (F1) values obtained for each class for each value of C tested in WEKA for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	F1 value for Class		Net Change (F1)
	BENIGN	DDOS	
0.1	0.999	1.000	0.000
0.2	0.999	1.000	0.000
0.25 (Default Value)	**0.999**	**1.000**	**0.000**
0.3	0.999	1.000	0.000
0.4	0.999	1.000	0.000
0.5	0.999	1.000	0.000

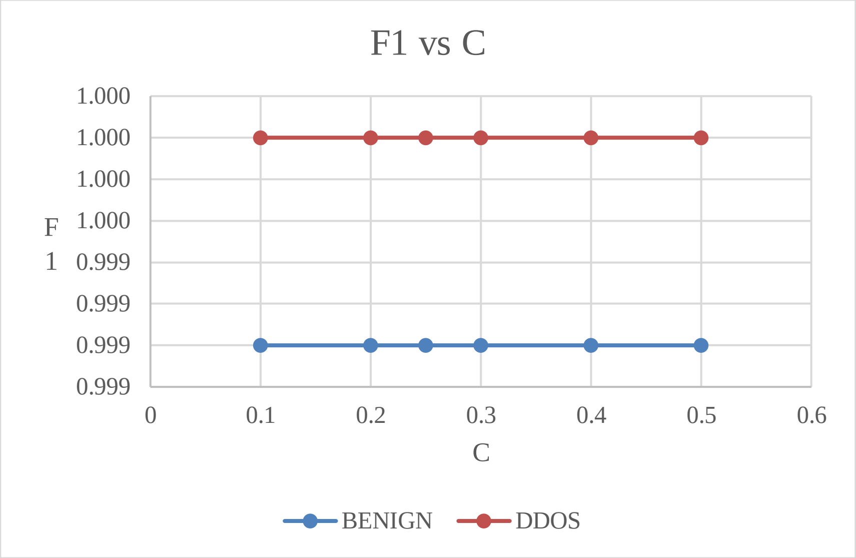


Figure B-4: Plot of F1 values for each individual class for values of C tested using WEKA for Level 0 Grouped Classification.

Table B-5: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of C tested in WEKA for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	AUC value for Class		Net Change (AUC)
	BENIGN	DDOS	
0.1	1.000	1.000	0.000
0.2	1.000	1.000	0.000
0.25 (Default Value)	**1.000**	**1.000**	**0.000**
0.3	1.000	1.000	0.000
0.4	1.000	1.000	0.000
0.5	1.000	1.000	0.000

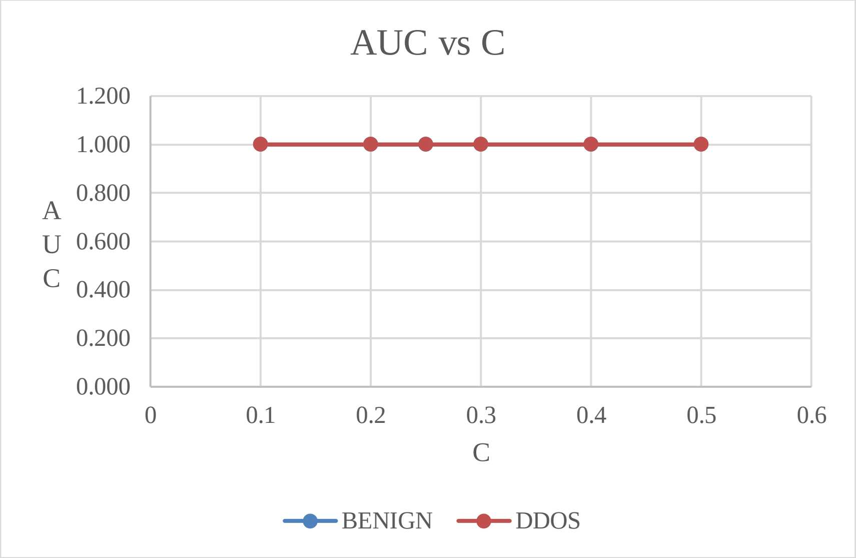


Figure B-5: Plot of AUC values for each individual class for values of C tested using WEKA for Level 0 Grouped Classification.

Table B-6: Table of False Positive Rates (FPR) obtained for each class for each value of C tested in WEKA for Level 0 Grouped Classification. (Light green highlight denotes lower than value obtained with default value from C = 0.25, while light red denotes higher than default value obtained)

Value Tested for C	FPR for Class	
	BENIGN	DDOS
0.1	0.0%	0.1%
0.2	0.0%	0.1%
0.25 (Default Value)	**0.0%**	**0.1%**
0.3	0.0%	0.1%
0.4	0.0%	0.1%
0.5	0.0%	0.1%

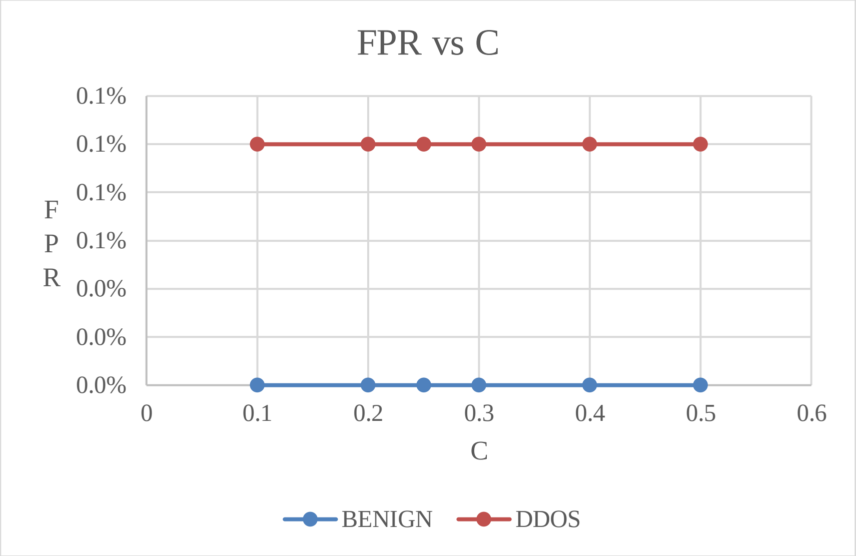


Figure B-6: Plot of FPR values for each individual class for values of C tested using WEKA for Level 0 Grouped Classification.

Table B-7: Table of G-Mean (GMEAN) values obtained for each class for each value of C tested for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested for C	GMEAN for Class		Net Change (GMEAN)
	BENIGN	DDOS	
0.1	0.9995	0.9995	0.0000
0.2	0.9995	0.9995	0.0000
0.25 (Default Value)	**0.9995**	**0.9995**	**0.0000**
0.3	0.9995	0.9995	0.0000
0.4	0.9995	0.9995	0.0000
0.5	0.9995	0.9995	0.0000

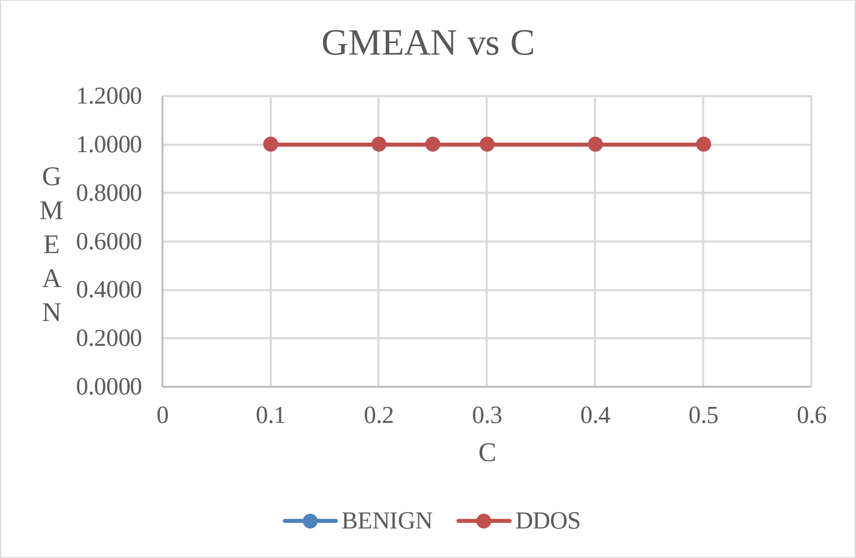


Figure B-7: Plot of GMEAN values for each individual class for values of C tested for Level 0 Grouped Classification.

Table B-8: Table of Net Change values for every evaluation metric used by value of C tested for Level 0 Grouped Classification.

Value Tested for C	Net Change				
	TPR	PREC	F1	AUC	GMEAN
0.1	0.0%	0.0%	0.000	0.000	0.0000
0.2	0.0%	0.0%	0.000	0.000	0.0000
0.25 (Default & Selected Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
0.3	0.0%	0.0%	0.000	0.000	0.0000
0.4	0.0%	0.0%	0.000	0.000	0.0000
0.5	0.0%	0.0%	0.000	0.000	0.0000

Table B-9: Table of General Evaluation Metrics (Weighted Average) Obtained from Level 0 Grouped Classification and Optimisation for each value of C tested in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value from C = 0.25, while light red denotes lower than default value)

Value Tested For C	Evaluation Metric (Weighted Average)							Standard Deviation					
	ACC	TPR	PREC	F1	AUC	TNR	GMEAN	ACC	TPR	PREC	F1	AUC	TNR
1	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
2 (Default Value)	**99.96%**	**100%**	**100%**	**1.00**	**1.00**	**100%**	**1.0000**	**0.01**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
3	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
4	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00
5	99.96%	100%	100%	1.00	1.00	100%	1.0000	0.01	0.00	0.00	0.00	0.00	0.00

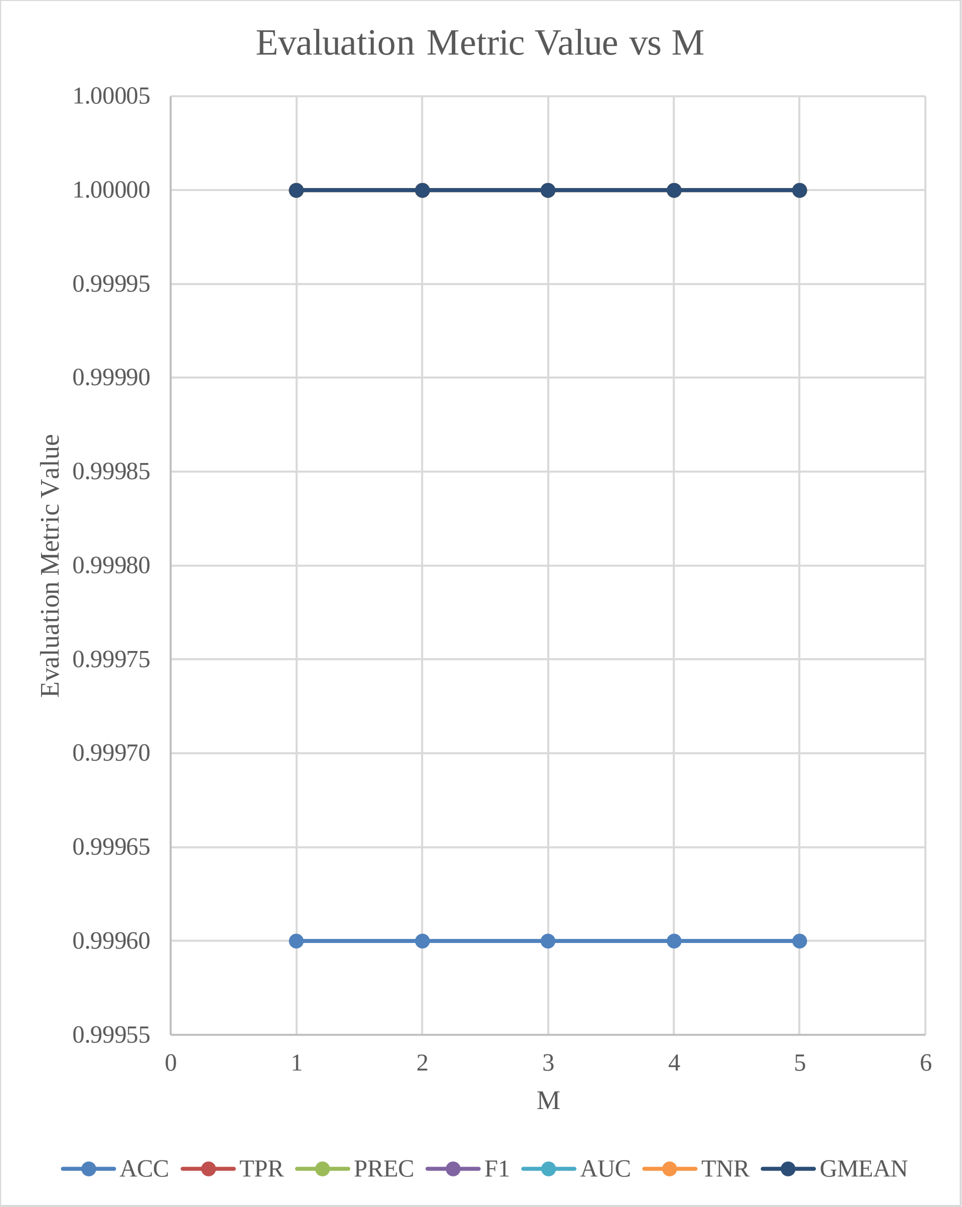


Figure B-8: Graph of General Weighted Average Evaluation Metrics value for Level 0 Grouped Classification by value of M tested in WEKA.

Table B-10: Table of True Positive Rates (TPR) obtained for each class for each value of M tested in WEKA (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value Tested for M	TPR for Class		Net Change (TPR)
	BENIGN	DDOS	
1	99.9%	100.0%	0.0%
2 (Default Value)	**99.9%**	**100.0%**	**0.0%**
3	99.9%	100.0%	0.0%
4	99.9%	100.0%	0.0%
5	99.9%	100.0%	0.0%

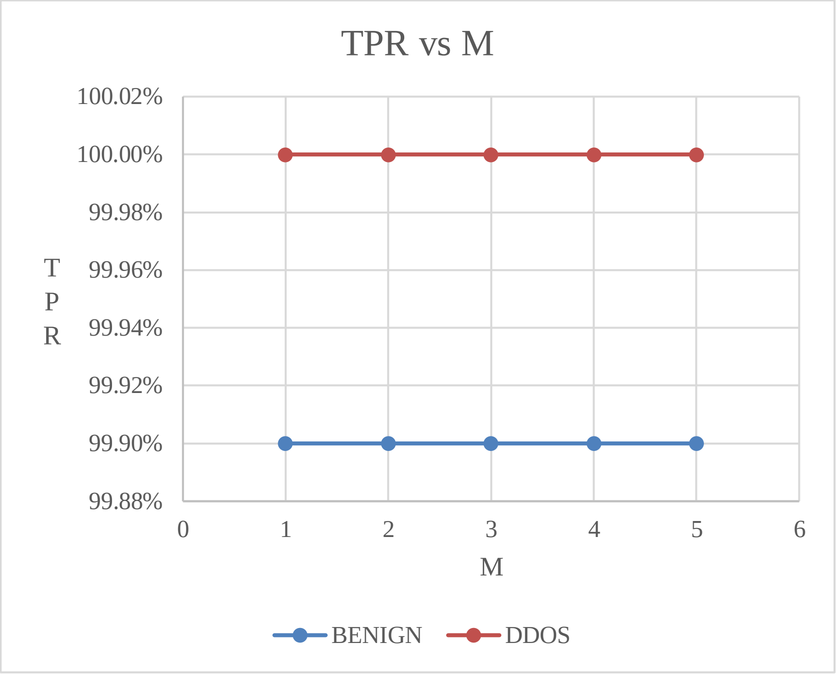


Figure B-9: Plot of TPR values for each individual class for values of M tested using WEKA (with C = 0.25) for Level 0 Grouped Classification.

Table B-11: Table of Precision (PREC) values obtained for each class for each value of M tested in WEKA (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value Tested for M	PREC for Class		Net Change (PREC)
	BENIGN	DDOS	
1	99.9%	100.0%	0.0%
2 (Default Value)	**99.9%**	**100.0%**	**0.0%**
3	99.9%	100.0%	0.0%
4	99.9%	100.0%	0.0%
5	99.9%	100.0%	0.0%

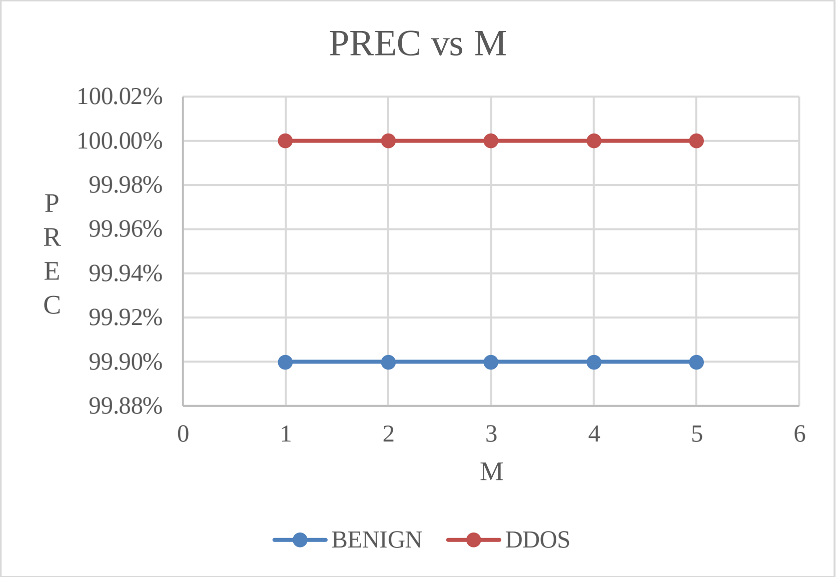


Figure B-10: Plot of PREC values for each individual class for values of M tested using WEKA (with C = 0.25) for Level 0 Grouped Classification.

Table B-12: Table of F1-Measure (F1) values obtained for each class for each value of M tested in WEKA (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value Tested for M	F1 for Class		Net Change (F1)
	BENIGN	DDOS	
1	0.999	1.000	0.000
2 (Default Value)	**0.999**	**1.000**	**0.000**
3	0.999	1.000	0.000
4	0.999	1.000	0.000
5	0.999	1.000	0.000

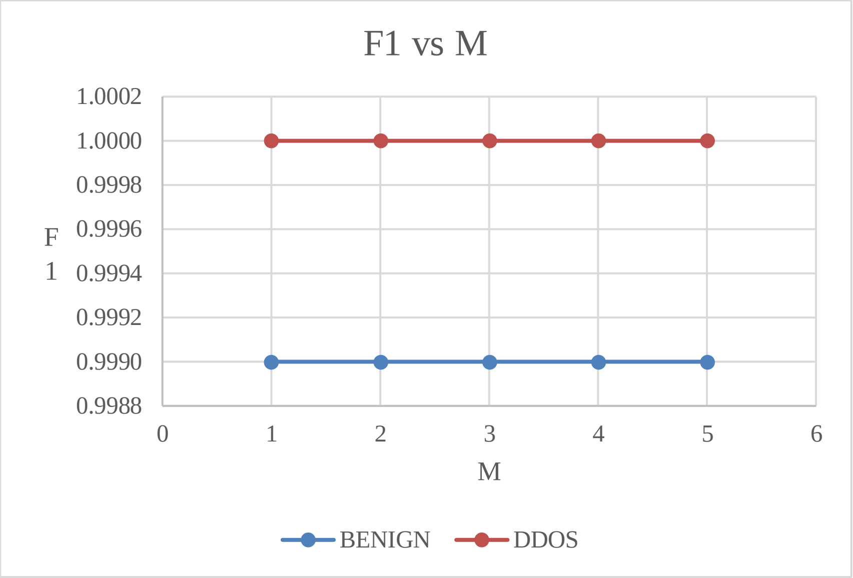


Figure B-11: Plot of F1 values for each individual class for values of M tested using WEKA (with C = 0.25) for Level 0 Grouped Classification.

Table B-13: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of M tested in WEKA (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value Tested for M	AUC for Class		Net Change (AUC)
	BENIGN	DDOS	
1	1.000	1.000	0.000
2 (Default Value)	**1.000**	**1.000**	**0.000**
3	1.000	1.000	0.000
4	1.000	1.000	0.000
5	1.000	1.000	0.000

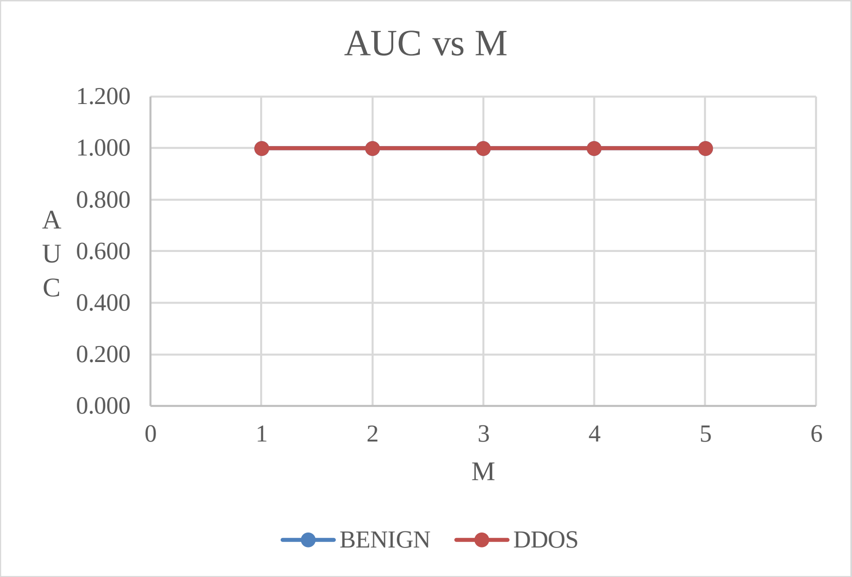


Figure B-12: Plot of AUC values for each individual class for values of M tested using WEKA (with C = 0.25) for Level 0 Grouped Classification.

Table B-14: Table of False Positive Rates (FPR) obtained for each class for each value of M tested in WEKA (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes lower than value obtained with default value of M = 2, while light red denotes higher than default value obtained)

Value Tested for M	FPR for Class	
	BENIGN	DDOS
1	0.0%	0.1%
2 (Default Value)	**0.0%**	**0.1%**
3	0.0%	0.1%
4	0.0%	0.1%
5	0.0%	0.1%

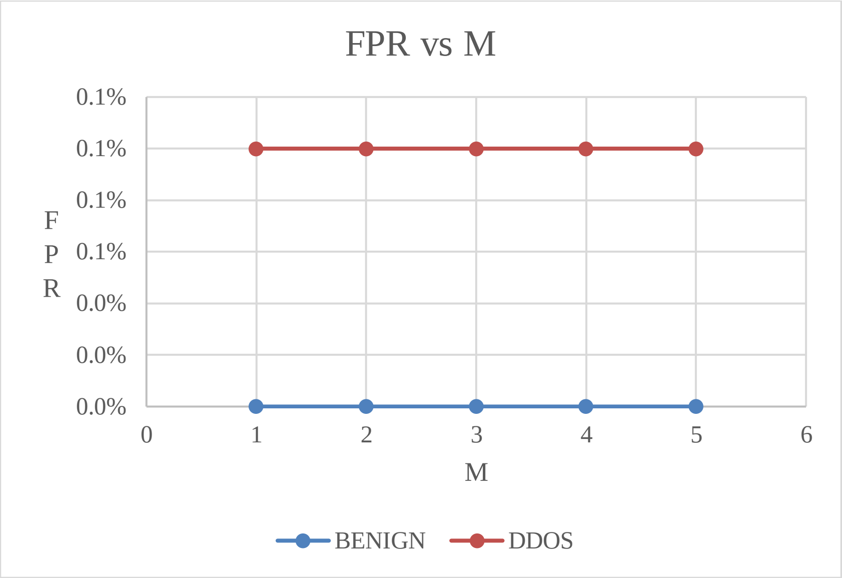


Figure B-13: Plot of FPR values for each individual class for values of M tested using WEKA (with C = 0.25) for Level 0 Grouped Classification.

Table B-15: Table of G-Mean (GMEAN) values obtained for each class for each value of M tested (with C = 0.25) for Level 0 Grouped Classification. (Light green highlight denotes higher than value obtained with default value of M = 2, while light red denotes lower than default value obtained)

Value Tested for M	GMEAN for Class		Net Change (GMEAN)
	BENIGN	DDOS	
1	0.9995	0.9995	0.0000
2 (Default Value)	**0.9995**	**0.9995**	**0.0000**
3	0.9995	0.9995	0.0000
4	0.9995	0.9995	0.0000
5	0.9995	0.9995	0.0000

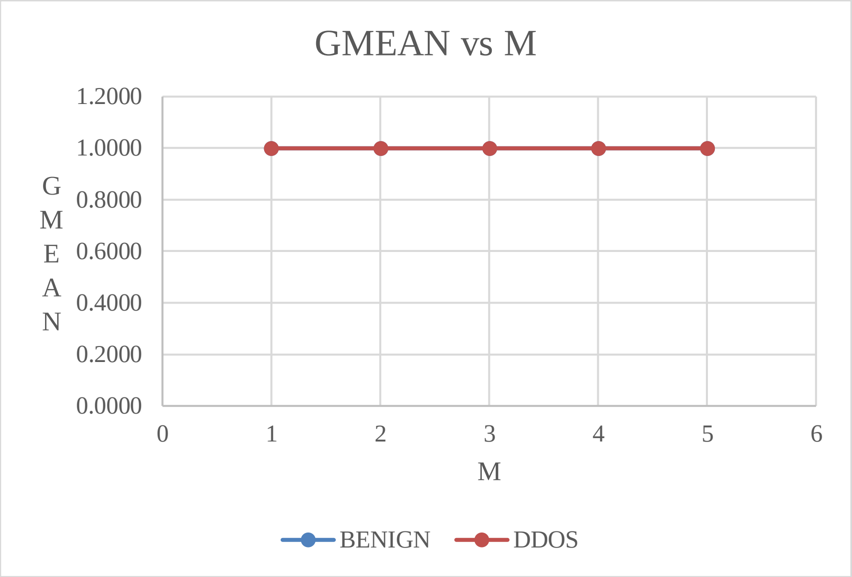


Figure B-14: Plot of GMEAN values for each individual class for values of M tested (with C = 0.25) for Level 0 Grouped Classification.

Table B-16: Table of Net Change values for every evaluation metric used by value of M tested (with C = 0.25) for Level 0 Grouped Classification.

Value Tested for M	Net Change				
	TPR	PREC	F1	AUC	GMEAN
1	0.0%	0.0%	0.000	0.000	0.0000
2 (Default & Selected Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
3	0.0%	0.0%	0.000	0.000	0.0000
4	0.0%	0.0%	0.000	0.000	0.0000
5	0.0%	0.0%	0.000	0.000	0.0000

Level 1 Grouped Classification

Table B-17: Table of General Evaluation Metrics Obtained from Level 1 Grouped Classification and Optimisation for each value of C tested in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value from C = 0.25, while light red denotes lower than default value)

Value of C Tested	Evaluation Metric (Weighted Average)							Standard Deviation					
	ACC	TPR	PREC	F1	AUC	TNR	GMEAN	ACC	TPR	PREC	F1	AUC	TNR
0.1	96.84%	97%	97%	0.97	0.99	99%	0.9799	0.10	0.00	0.00	0.00	0.00	0.00
0.2	96.85%	97%	97%	0.97	0.99	99%	0.9799	0.10	0.00	0.00	0.00	0.00	0.00
0.21	96.85%	97%	97%	0.97	0.99	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.22	96.84%	97%	97%	0.97	0.99	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.23	96.84%	97%	97%	0.97	0.99	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.24	96.84%	97%	97%	0.97	0.99	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.25 (Default value)	**96.84%**	**97%**	**97%**	**0.97**	**1.00**	**99%**	**0.9799**	**0.09**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
0.26	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.27	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.28	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.29	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.3	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
0.4	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.10	0.00	0.00	0.00	0.00	0.00
0.5	96.84%	97%	97%	0.97	1.00	99%	0.9799	0.10	0.00	0.00	0.00	0.00	0.00

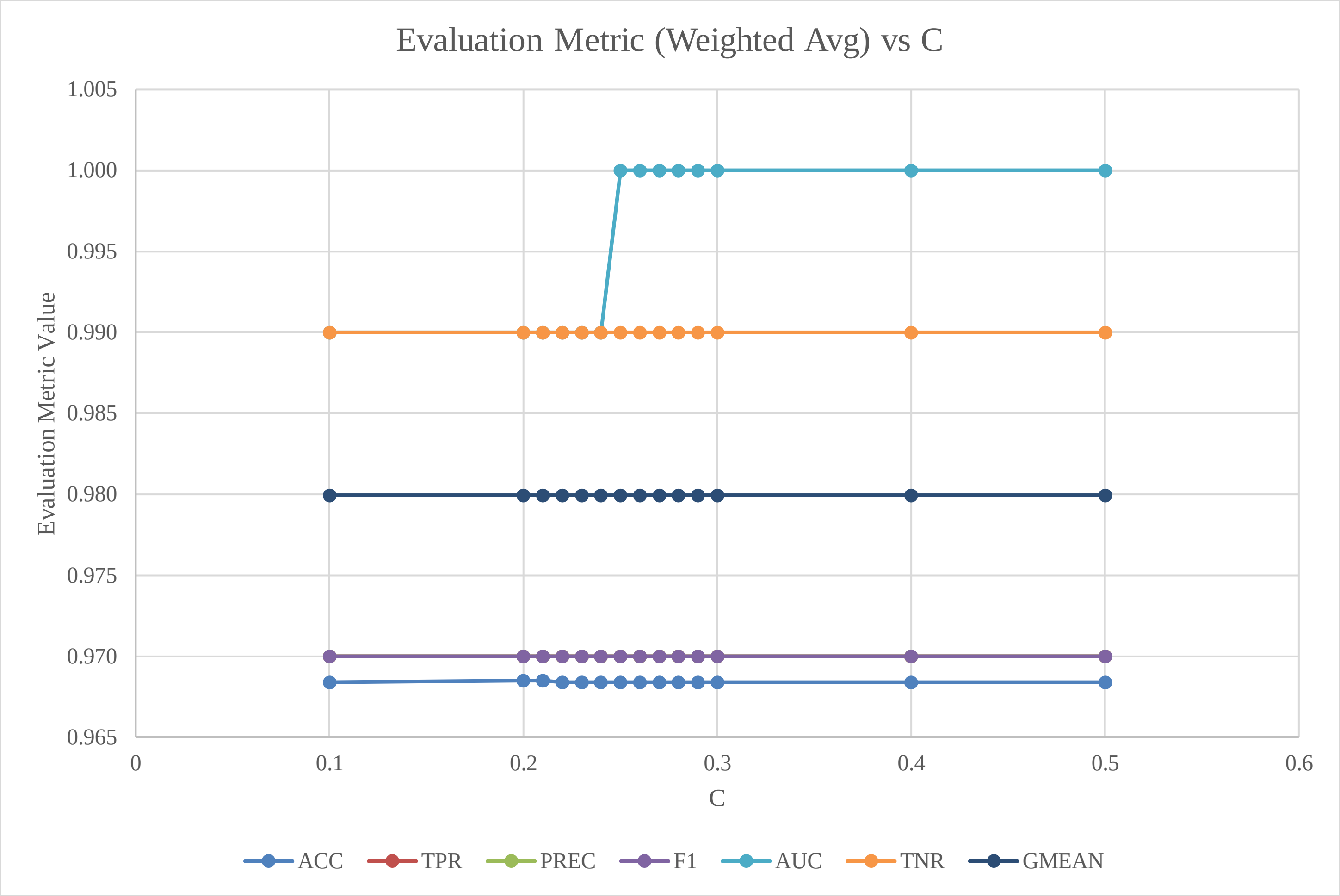


Figure B-15: Graph of General Evaluation Metrics value for Level 1 Grouped Classification by value of C tested in WEKA.

Table B-18: Table of True Positive Rates (TPR) obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value of C Tested	TPR for Class				Net Change (TPR)
	Normal	Reflection	Exploitation	HTTP/WebDDoS	
0.1	99.9%	95.1%	98.6%	95.4%	-0.4%
0.2	99.9%	95.1%	98.5%	96.1%	+0.2%
0.21	99.9%	95.1%	98.5%	95.4%	-0.5%
0.22	99.9%	95.1%	98.5%	95.4%	-0.5%
0.23	99.9%	95.1%	98.5%	96.1%	+0.2%
0.24	99.9%	95.1%	98.4%	96.1%	+0.1%
0.25 (Default value)	**99.9%**	**95.1%**	**98.3%**	**96.1%**	**0.0%**
0.26	99.9%	95.2%	98.2%	96.1%	0.0%
0.27	99.9%	95.2%	98.2%	96.1%	0.0%
0.28	99.9%	95.2%	98.2%	96.1%	0.0%
0.29	99.9%	95.2%	98.0%	96.1%	-0.2%
0.3	99.9%	95.2%	97.9%	96.1%	-0.3%
0.4	99.9%	95.3%	97.8%	96.1%	-0.3%
0.5	99.9%	95.3%	97.7%	95.7%	-0.8%

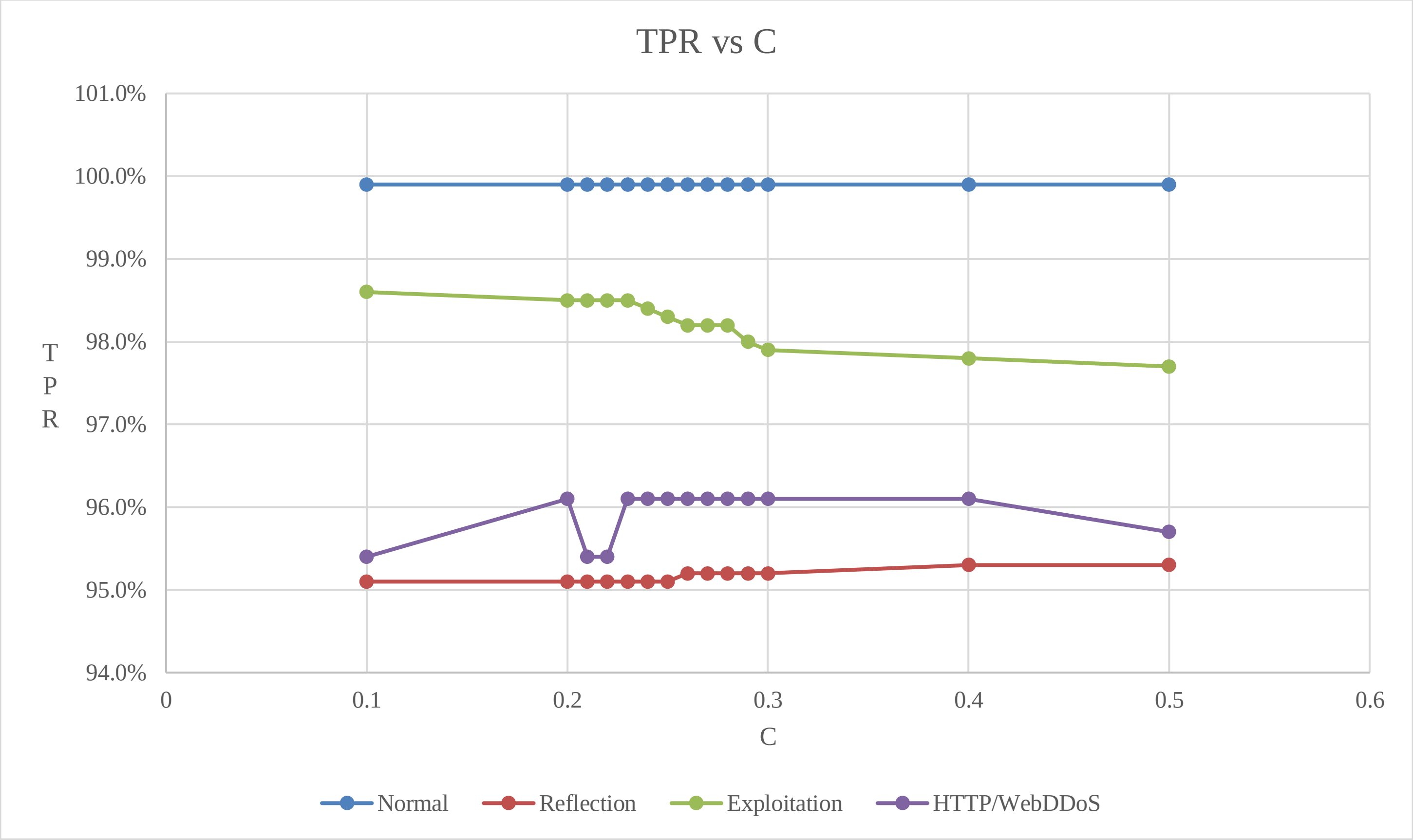


Figure B-16: Plot of TPR values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-19: Table of Precision (PREC) values obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value of C Tested	PREC for Class				Net Change (PREC)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
0.1	99.9%	99.6%	82.9%	95.4%	+2.8%
0.2	99.9%	99.6%	82.9%	92.3%	-0.3%
0.21	99.9%	99.6%	83.0%	92.1%	-0.4%
0.22	99.9%	99.6%	82.9%	92.5%	-0.1%
0.23	99.9%	99.6%	82.9%	92.5%	-0.1%
0.24	99.9%	99.6%	83.0%	92.5%	0.0%
0.25 (Default value)	**99.9%**	**99.6%**	**83.0%**	**92.5%**	**0.0%**
0.26	99.9%	99.6%	83.0%	92.1%	-0.4%
0.27	99.9%	99.6%	83.0%	92.1%	-0.4%
0.28	99.9%	99.5%	83.1%	92.1%	-0.4%
0.29	99.9%	99.5%	83.2%	91.9%	-0.5%
0.3	99.9%	99.5%	83.2%	91.9%	-0.5%
0.4	99.9%	99.4%	83.3%	91.9%	-0.5%
0.5	99.9%	99.4%	83.3%	91.9%	-0.5%

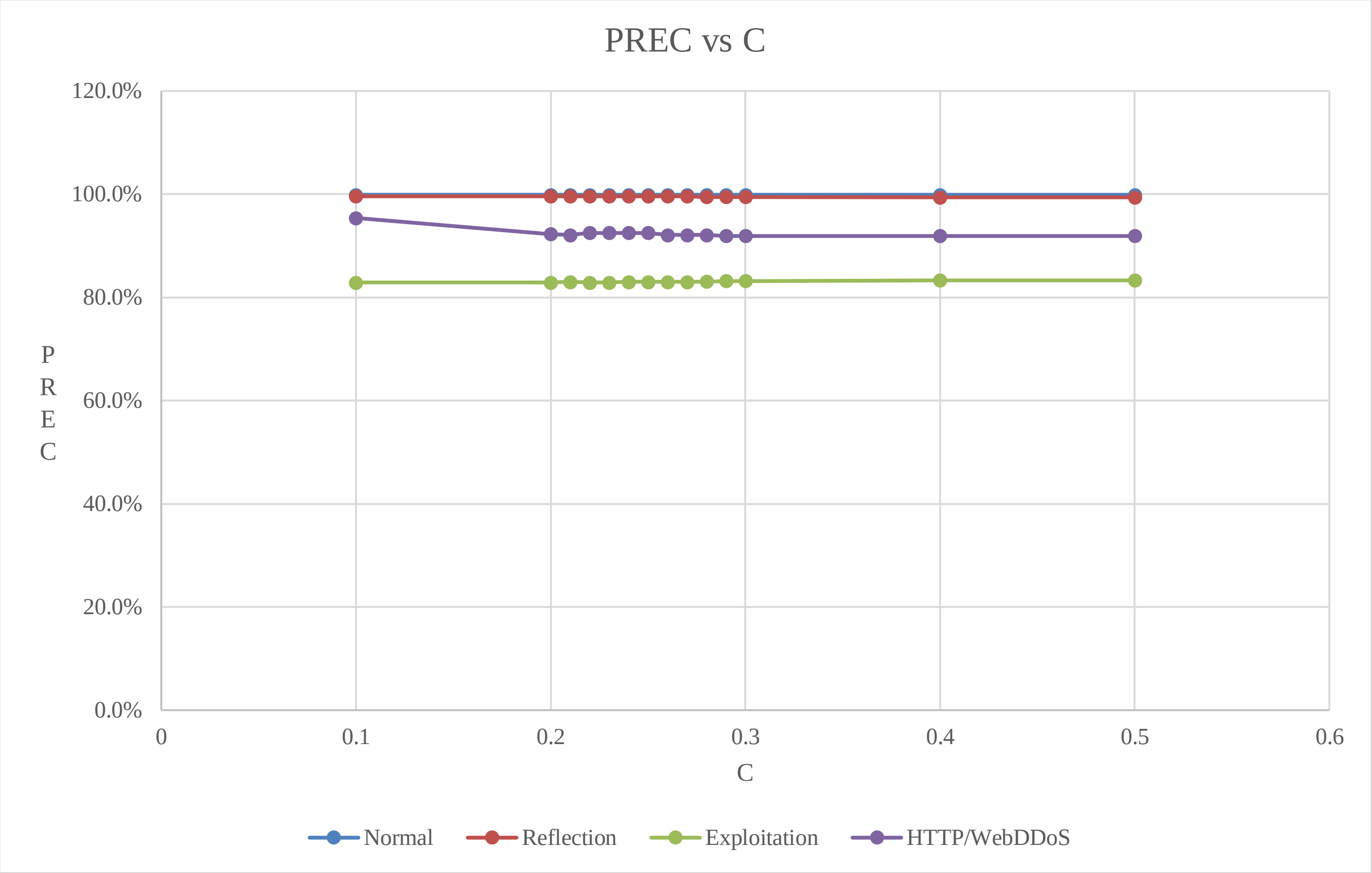


Figure B-17: Plot of PREC values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-20: Table of F-Measure (F1) values obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value of C Tested	F1 for Class				Net Change (F1)
	Normal	Reflection	Exploitation	HTTP/WebDDoS	
0.1	0.999	0.973	0.901	0.954	+0.097
0.2	0.999	0.973	0.901	0.942	+0.000
0.21	0.999	0.973	0.901	0.937	-0.005
0.22	0.999	0.973	0.900	0.943	0.000
0.23	0.999	0.973	0.900	0.943	0.000
0.24	0.999	0.973	0.900	0.943	0.000
0.25 (Default value)	**0.999**	**0.973**	**0.900**	**0.943**	**0.000**
0.26	0.999	0.973	0.900	0.941	-0.002
0.27	0.999	0.973	0.900	0.941	-0.002
0.28	0.999	0.973	0.900	0.941	-0.002
0.29	0.999	0.973	0.900	0.940	-0.003
0.3	0.999	0.973	0.900	0.940	-0.003
0.4	0.999	0.973	0.899	0.940	-0.004
0.5	0.999	0.973	0.899	0.938	-0.006

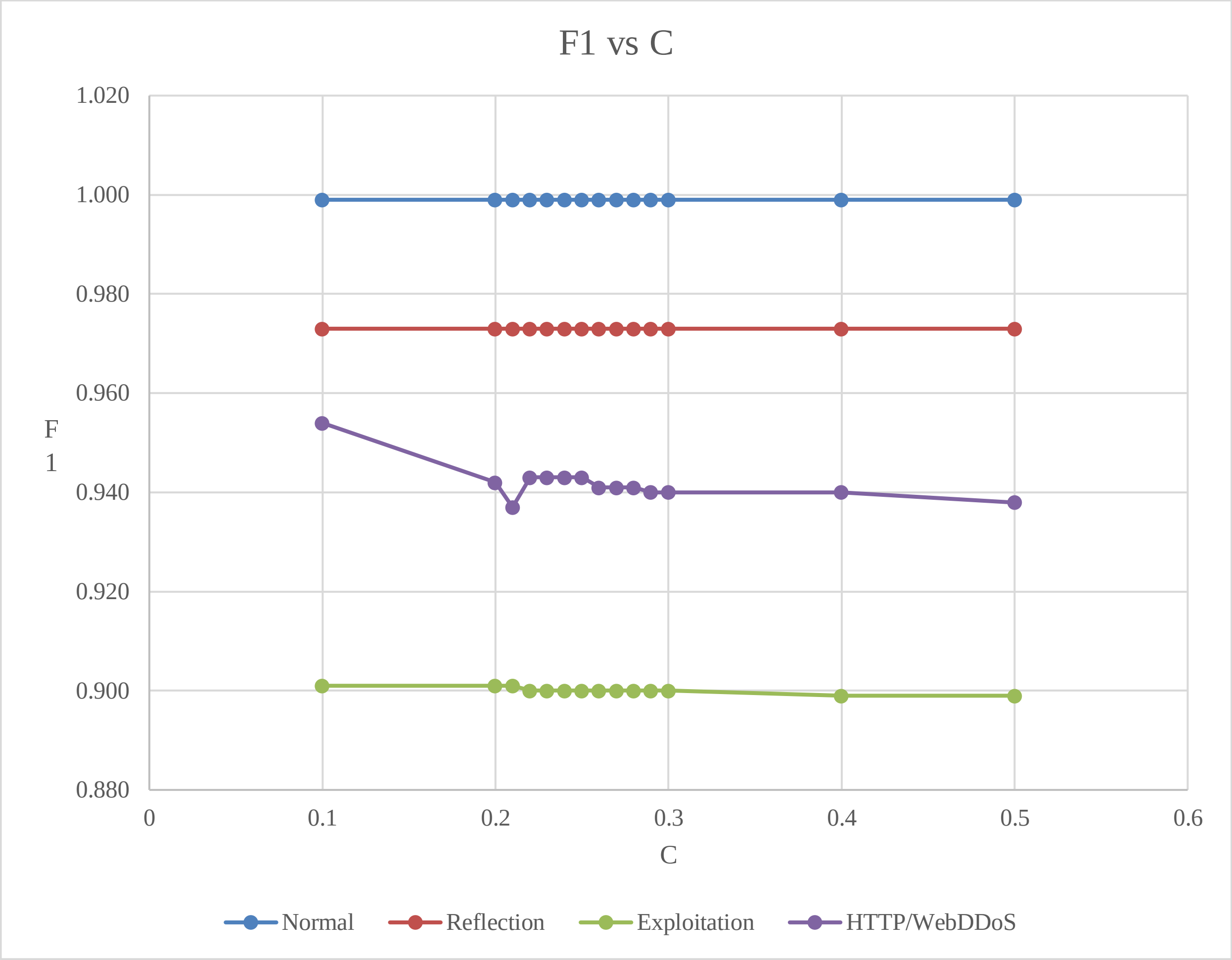


Figure B-18: Plot of F1 values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-21: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value of C Tested	AUC for Class				Net Change (AUC)
	Normal	Reflection	Exploitation	HTTP/WebDDoS	
0.1	1.000	0.994	0.988	0.995	0.000
0.2	1.000	0.994	0.989	0.994	0.000
0.21	1.000	0.994	0.989	0.988	-0.006
0.22	1.000	0.994	0.989	0.994	0.000
0.23	1.000	0.994	0.989	0.994	0.000
0.24	1.000	0.994	0.989	0.994	0.000
0.25 (Default value)	**1.000**	**0.994**	**0.989**	**0.994**	**0.000**
0.26	1.000	0.995	0.989	0.994	+0.001
0.27	1.000	0.995	0.989	0.994	+0.001
0.28	1.000	0.995	0.989	0.994	+0.001
0.29	1.000	0.995	0.990	0.993	+0.001
0.3	1.000	0.995	0.990	0.993	+0.001
0.4	1.000	0.995	0.990	0.993	+0.001
0.5	1.000	0.995	0.990	0.988	-0.004

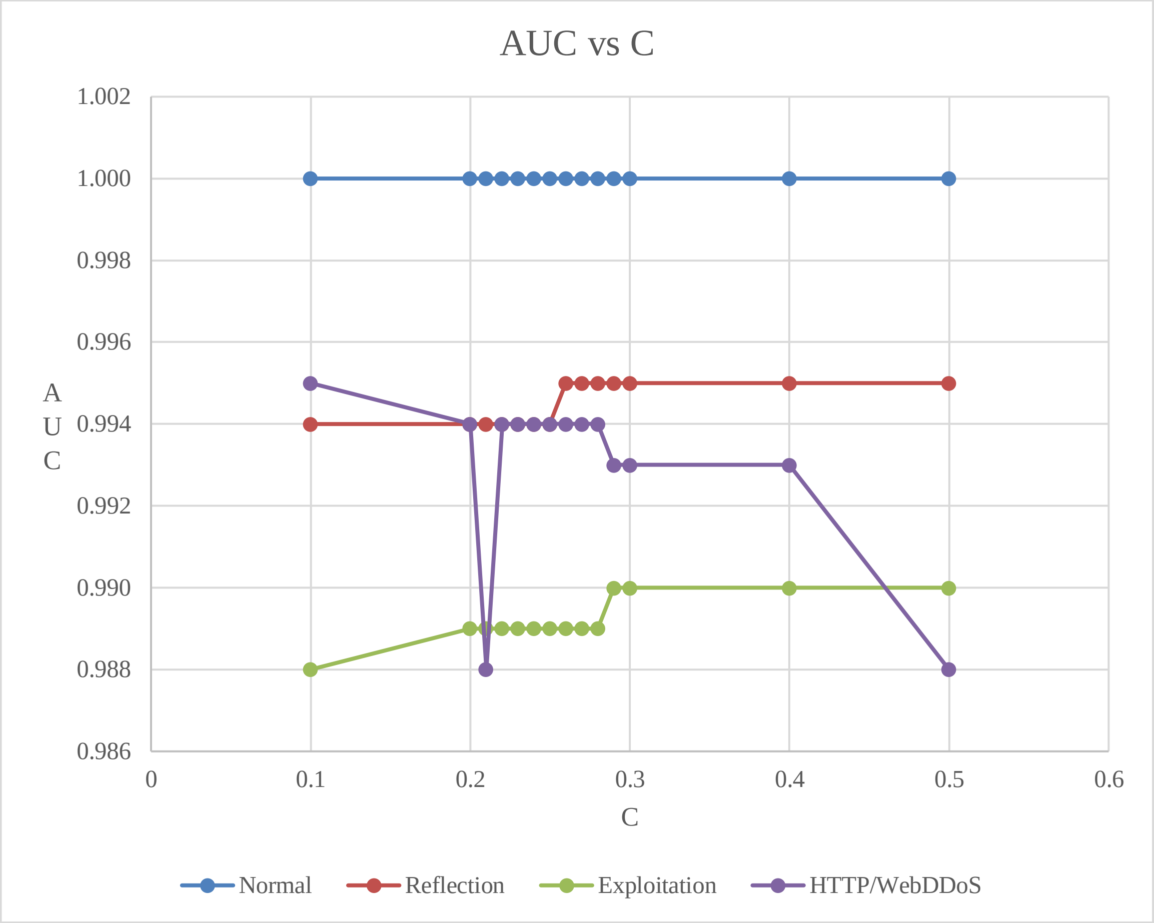


Figure B-19: Plot of AUC values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-22: Table of False Positive Rates (FPR) obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes lower than value obtained with default value from C = 0.25, while light red denotes higher than default value obtained)

Value of C Tested	Class			
	Normal	Reflection	Exploitation	HTTP/WebDDoS
0.1	0.0%	0.5%	3.4%	0.0%
0.2	0.0%	0.5%	3.4%	0.0%
0.21	0.0%	0.5%	3.4%	0.0%
0.22	0.0%	0.5%	3.4%	0.0%
0.23	0.0%	0.5%	3.4%	0.0%
0.24	0.0%	0.6%	3.4%	0.0%
0.25 (Default value)	**0.0%**	**0.6%**	**3.4%**	**0.0%**
0.26	0.0%	0.6%	3.3%	0.0%
0.27	0.0%	0.6%	3.3%	0.0%
0.28	0.0%	0.6%	3.3%	0.0%
0.29	0.0%	0.7%	3.3%	0.0%
0.3	0.0%	0.7%	3.3%	0.0%
0.4	0.0%	0.8%	3.3%	0.0%
0.5	0.0%	0.8%	3.3%	0.0%

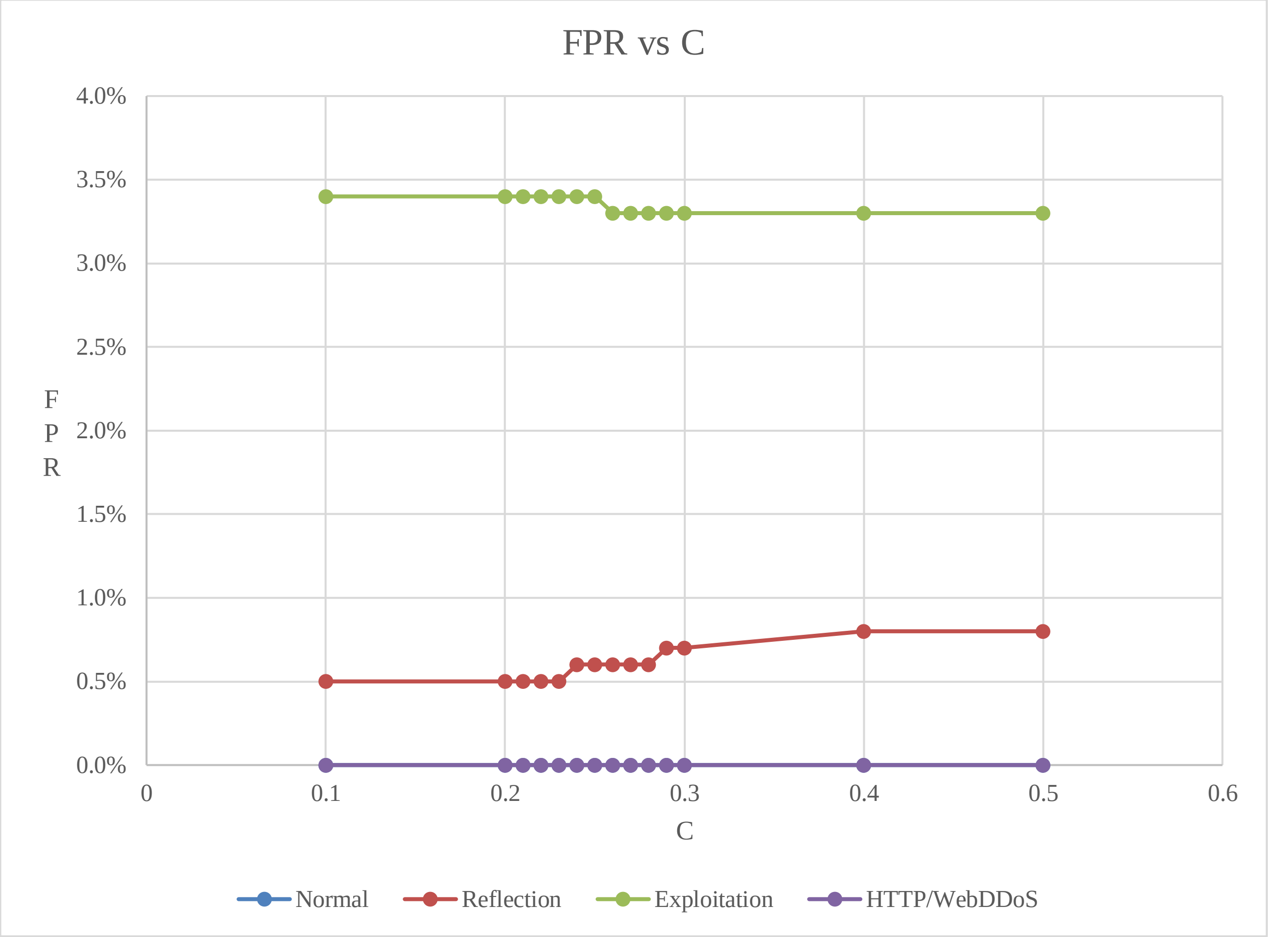


Figure B-20: Plot of FPR values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-23: Table of G-Mean (GMEAN) values obtained for each class for each value of C tested in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested	Class				Net Change (GMEAN)
	Normal	Reflection	Exploitation	HTTP/WebDDoS	
0.1	0.9995	0.9728	0.9759	0.9767	-0.002
0.2	0.9995	0.9728	0.9755	0.9803	+0.001
0.21	0.9995	0.9728	0.9755	0.9767	-0.002
0.22	0.9995	0.9728	0.9755	0.9767	-0.002
0.23	0.9995	0.9728	0.9755	0.9803	+0.001
0.24	0.9995	0.9723	0.9750	0.9803	0.000
0.25 (Default Value)	**0.9995**	**0.9723**	**0.9745**	**0.9803**	**0.000**
0.26	0.9995	0.9728	0.9745	0.9803	+0.001
0.27	0.9995	0.9728	0.9745	0.9803	+0.001
0.28	0.9995	0.9728	0.9745	0.9803	+0.001
0.29	0.9995	0.9723	0.9735	0.9803	-0.001
0.3	0.9995	0.9723	0.9730	0.9803	-0.001
0.4	0.9995	0.9723	0.9725	0.9803	-0.002
0.5	0.9995	0.9723	0.9720	0.9783	-0.004

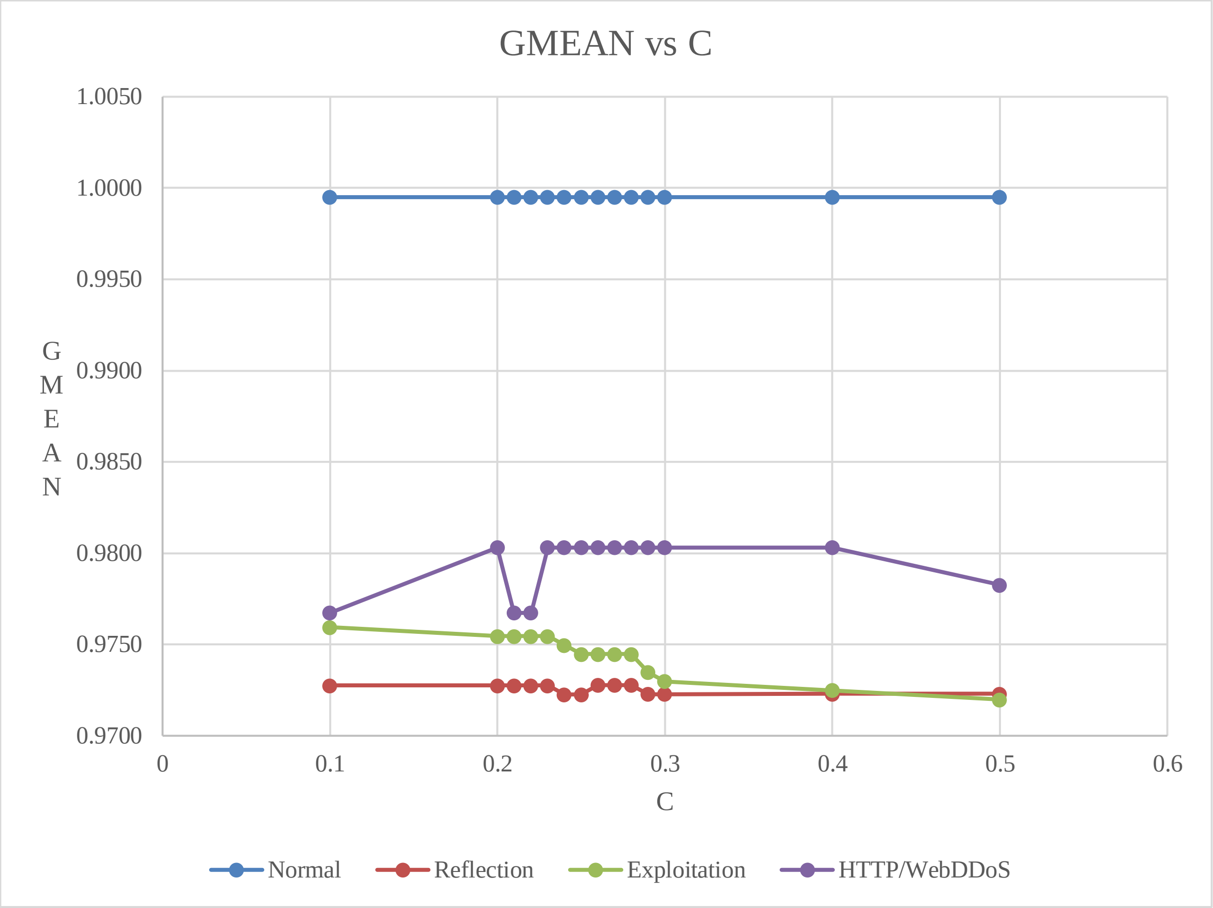


Figure B-21: Plot of GMEAN values for each individual class for values of C tested using WEKA for Level 1 Grouped Classification.

Table B-24: Table of Net Change values for every evaluation metric used by value of C tested for Level 1 Grouped Classification.

Value Tested for C	Net Change				
	TPR	PREC	F1	AUC	GMEAN
0.1	-0.4%	+2.8%	+0.097	0.000	-0.002
0.2	+0.2%	-0.3%	+0.000	0.000	+0.001
0.21	-0.5%	-0.4%	-0.005	-0.006	-0.002
0.22	-0.5%	-0.1%	0.000	0.000	-0.002
0.23 (Selected Value)	**+0.2%**	**-0.1%**	**0.000**	**0.000**	**+0.001**
0.24	+0.1%	0.0%	0.000	0.000	0.000
0.25 (Default Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.000**
0.26	0.0%	-0.4%	-0.002	+0.001	+0.001
0.27	0.0%	-0.4%	-0.002	+0.001	+0.001
0.28	0.0%	-0.4%	-0.002	+0.001	+0.001
0.29	-0.2%	-0.5%	-0.003	+0.001	-0.001
0.3	-0.3%	-0.5%	-0.003	+0.001	-0.001
0.4	-0.3%	-0.5%	-0.004	+0.001	-0.002
0.5	-0.8%	-0.5%	-0.006	-0.004	-0.004

Table B-25: Table of General Evaluation Metrics from Level 1 Grouped Classification and Optimisation for each value of M tested (with C = 0.23) in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value from M = 2, while light red denotes lower than default value)

Value of M Tested	Evaluation Metric (Weighted Average)							Standard Deviation					
	ACC (%)	TPR (%)	PREC (%)	F1	AUC	TNR (%)	GMEAN	ACC (%)	TPR (%)	PREC (%)	F1	AUC	TNR (%)
1	96.84	97	97	0.97	0.99	99	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
2 (Default value)	**96.84**	**97**	**97**	**0.97**	**0.99**	**99**	**0.9799**	**0.09**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
3	96.84	97	97	0.97	1.00	99	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
4	96.84	97	97	0.97	1.00	99	0.9799	0.09	0.00	0.00	0.00	0.00	0.00
5	96.84	97	97	0.97	1.00	99	0.9799	0.09	0.00	0.00	0.00	0.00	0.00

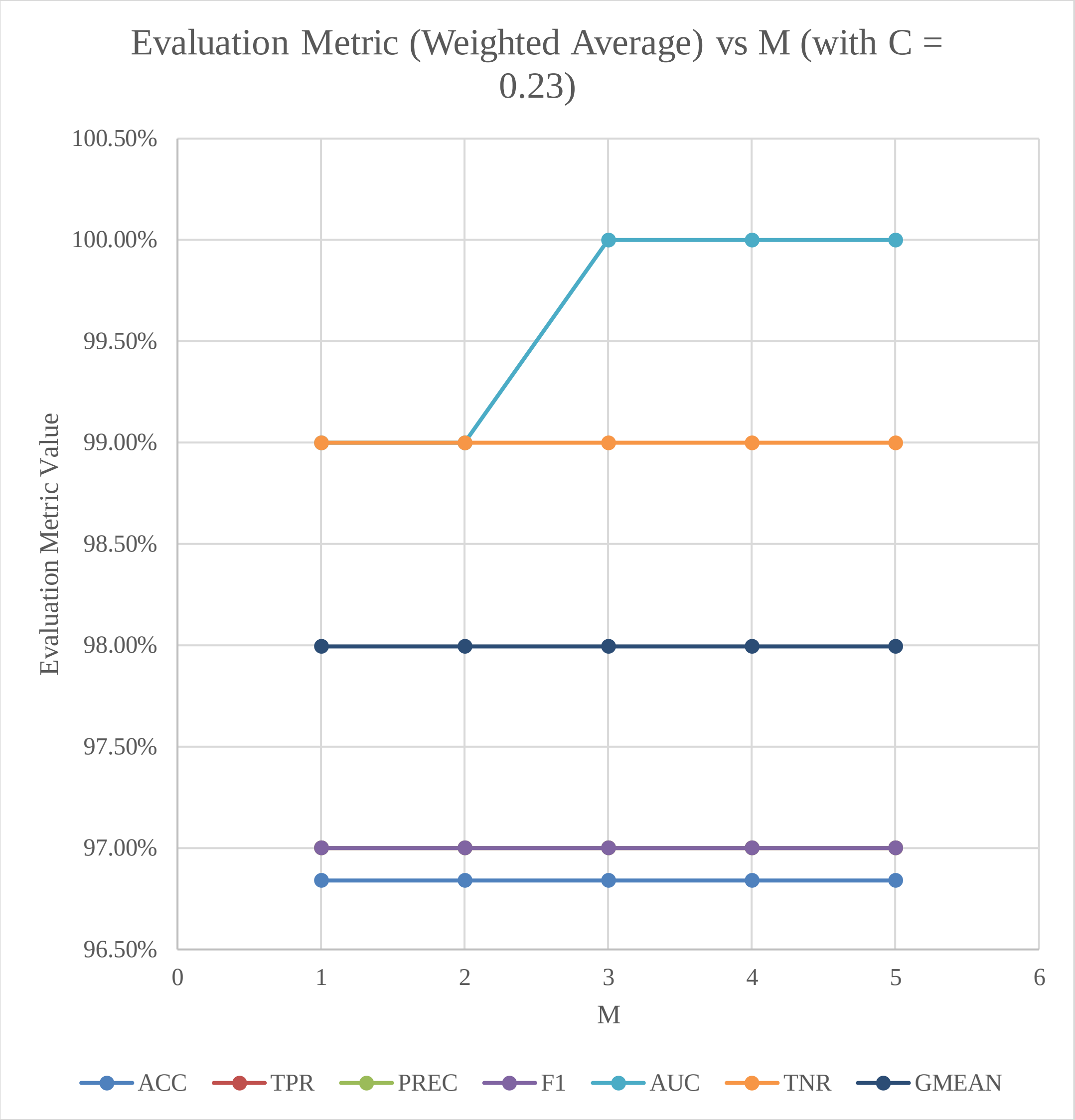


Figure B-22: Graph of General Evaluation Metrics value for Level 1 Grouped Classification by value of M tested (with C = 0.23) in WEKA.

Table B-26: Table of True Positive Rates (TPR) obtained for each class for each value of M tested (with C = 0.23) in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	TPR for Class				Net Change (TPR)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
1	99.9%	95.1%	98.5%	96.4%	+0.4%
2 (Default Value)	**99.9%**	**95.1%**	**98.4%**	**96.1%**	**0.0%**
3	99.9%	95.1%	98.5%	96.1%	+0.1%
4	99.9%	95.1%	98.5%	96.1%	+0.1%
5	99.9%	95.1%	98.4%	95.7%	-0.4%

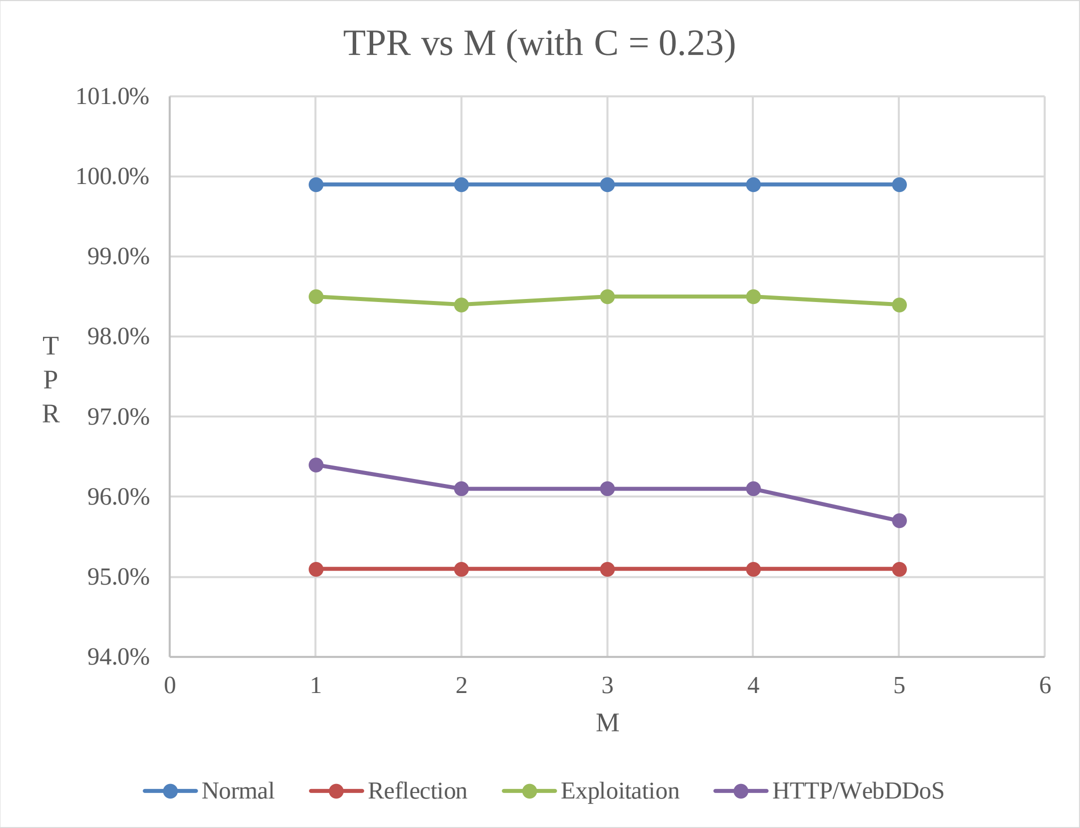


Figure B-23: Plot of TPR values for each individual class for values of M tested (with C = 0.23) using WEKA for Level 1 Grouped Classification.

Table B-27: Table of Precision (PREC) values obtained for each class for each value of M tested (with C = 0.23) in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	PREC for Class				Net Change (PREC)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
1	99.9%	99.6%	82.9%	92.8%	+0.2%
2 (Default value)	**99.9%**	**99.6%**	**83.0%**	**92.5%**	**0.0%**
3	99.9%	99.6%	83.0%	92.7%	+0.2%
4	99.9%	99.6%	83.0%	92.7%	+0.2%
5	99.9%	99.6%	83.0%	92.7%	+0.2%

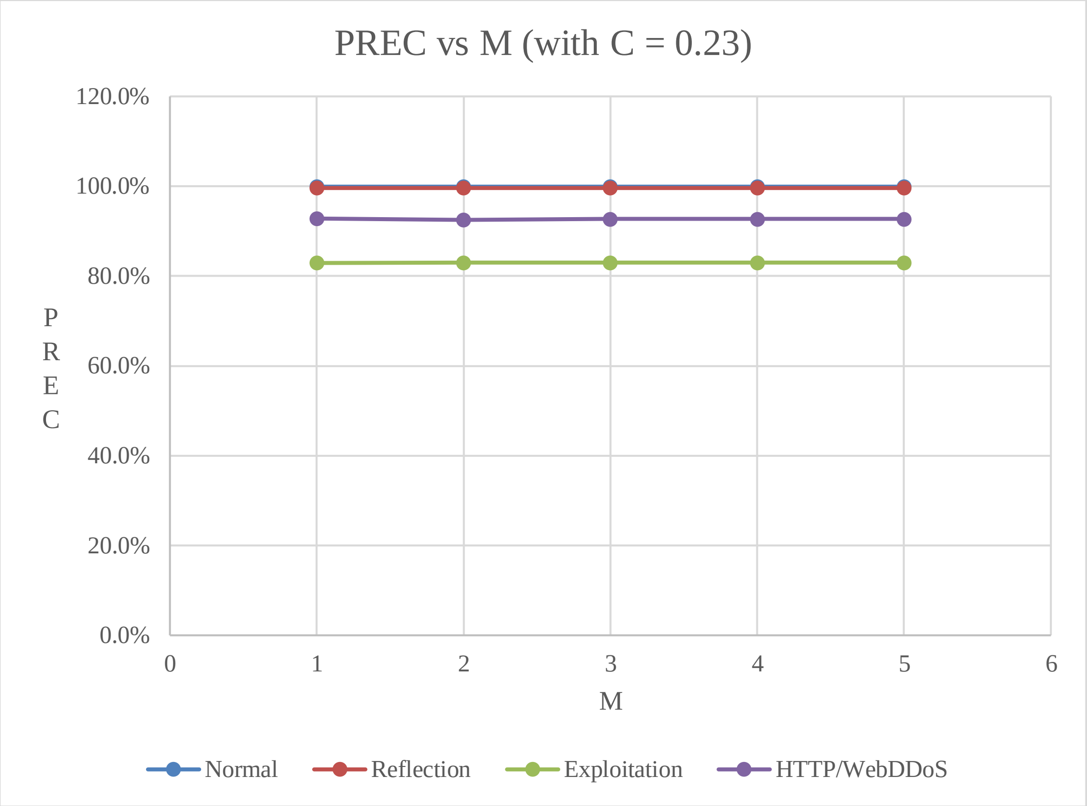


Figure B-24: Plot of PREC values for each individual class for values of M tested (with C = 0.23) using WEKA for Level 1 Grouped Classification.

Table B-28: Table of F-Measure (F1) obtained for each class for each value of M tested (with C = 0.23) in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	F1 for Class				Net Change (F1)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
1	0.999	0.973	0.900	0.945	+0.002
2 (Default value)	**0.999**	**0.973**	**0.900**	**0.943**	**0.000**
3	0.999	0.973	0.900	0.944	+0.001
4	0.999	0.973	0.901	0.944	+0.002
5	0.999	0.973	0.900	0.942	-0.001

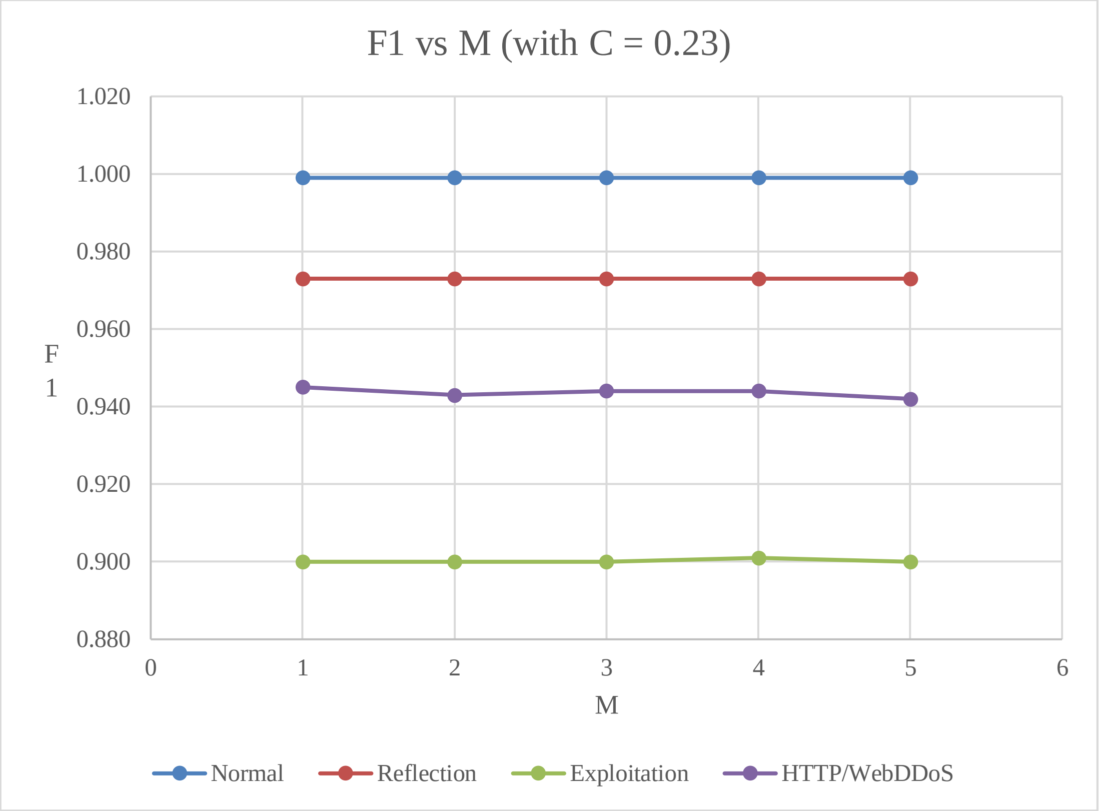


Figure B-25: Plot of F1 values for each individual class for values of M tested (with C = 0.23) using WEKA for Level 1 Grouped Classification.

Table B-29: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of M tested (with C = 0.23) in WEKA for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	AUC for Class				Net Change (AUC)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
1	1.000	0.994	0.989	0.991	-0.003
2 (Default value)	**1.000**	**0.994**	**0.989**	**0.994**	**0.000**
3	1.000	0.995	0.989	0.995	+0.002
4	1.000	0.995	0.989	0.997	+0.004
5	1.000	0.995	0.989	0.997	+0.004

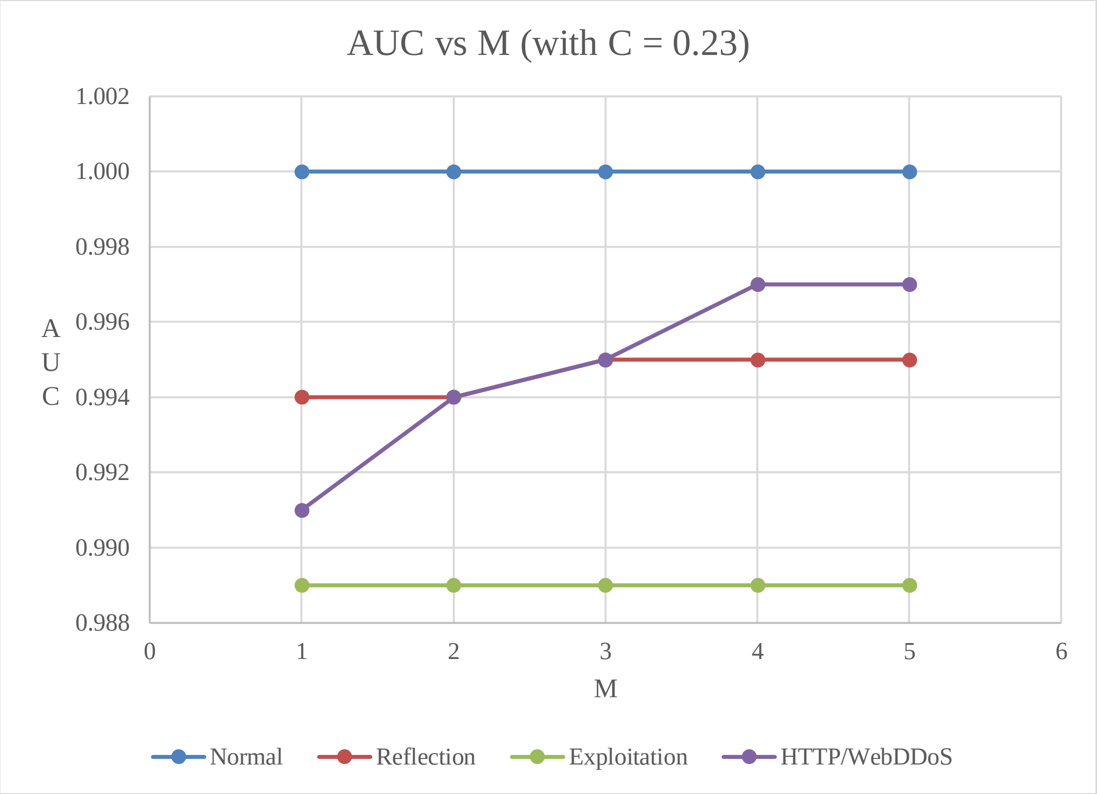


Figure B-26: Plot of AUC values for each individual class for values of M tested (with C = 0.23) using WEKA for Level 1 Grouped Classification.

Table B-30: Table of False Positive Rates (FPR) obtained for each class for each value of M tested (with C = 0.23) in WEKA for Level 1 Grouped Classification. (Light green highlight denotes lower than value obtained with default value from M = 2, while light red denotes higher than default value obtained)

Value Tested For M	FPR for Class			
	Normal	Reflection	Exploitation	HTTP/WebDDoS
1	0.0%	0.5%	3.4%	0.0%
2 (Default value)	**0.0%**	**0.6%**	**3.4%**	**0.0%**
3	0.0%	0.5%	3.4%	0.0%
4	0.0%	0.6%	3.4%	0.0%
5	0.0%	0.6%	3.4%	0.0%

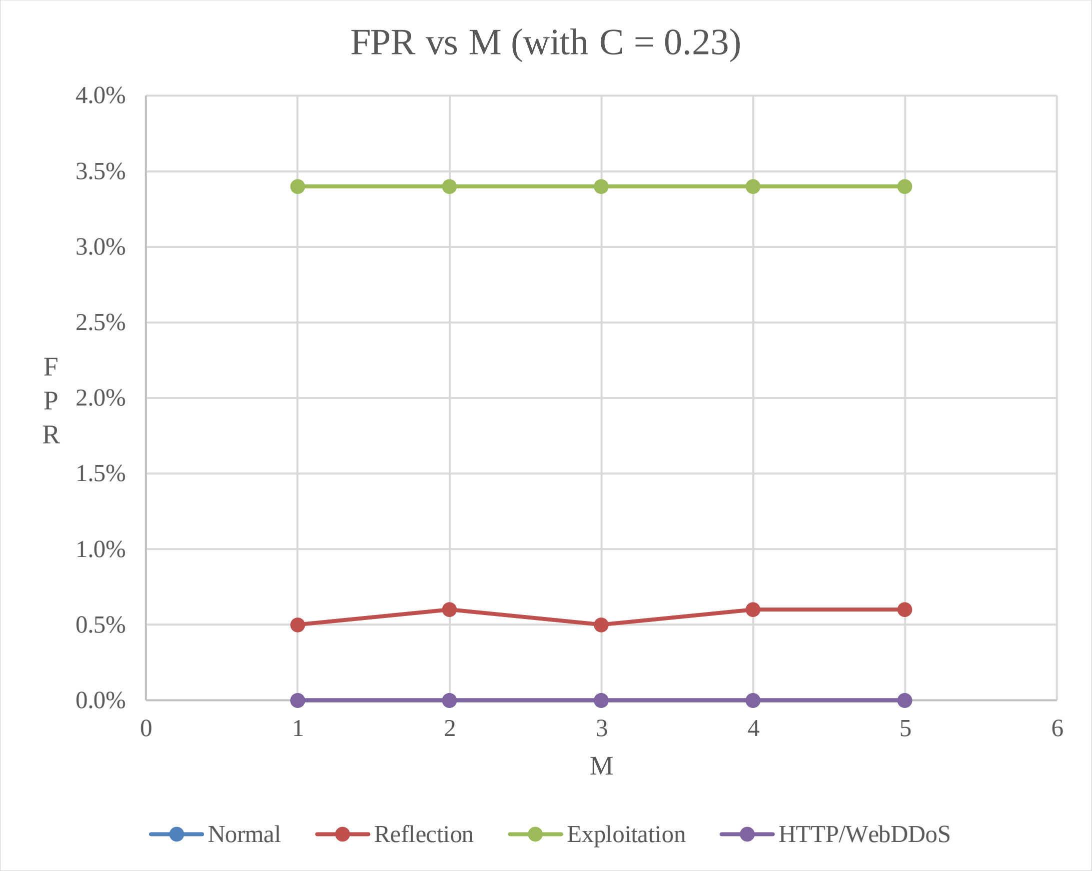


Figure B-27: Plot of FPR for each individual class for values of M tested (with C = 0.23) using WEKA for Level 1 Grouped Classification.

Table B-31: Table of G-Mean (GMEAN) values obtained for each class for each value of M tested (with C = 0.23) for Level 1 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	GMEAN for Class				Net Change (GMEAN)
	Normal	Reflection	Exploitation	HTTP/ WebDDoS	
1	0.9995	0.9728	0.9755	0.9818	+0.0025
2 (Default value)	**0.9995**	**0.9723**	**0.9750**	**0.9803**	**0.0000**
3	0.9995	0.9728	0.9755	0.9803	+0.0010
4	0.9995	0.9723	0.9755	0.9803	+0.0005
5	0.9995	0.9723	0.9750	0.9783	-0.0020

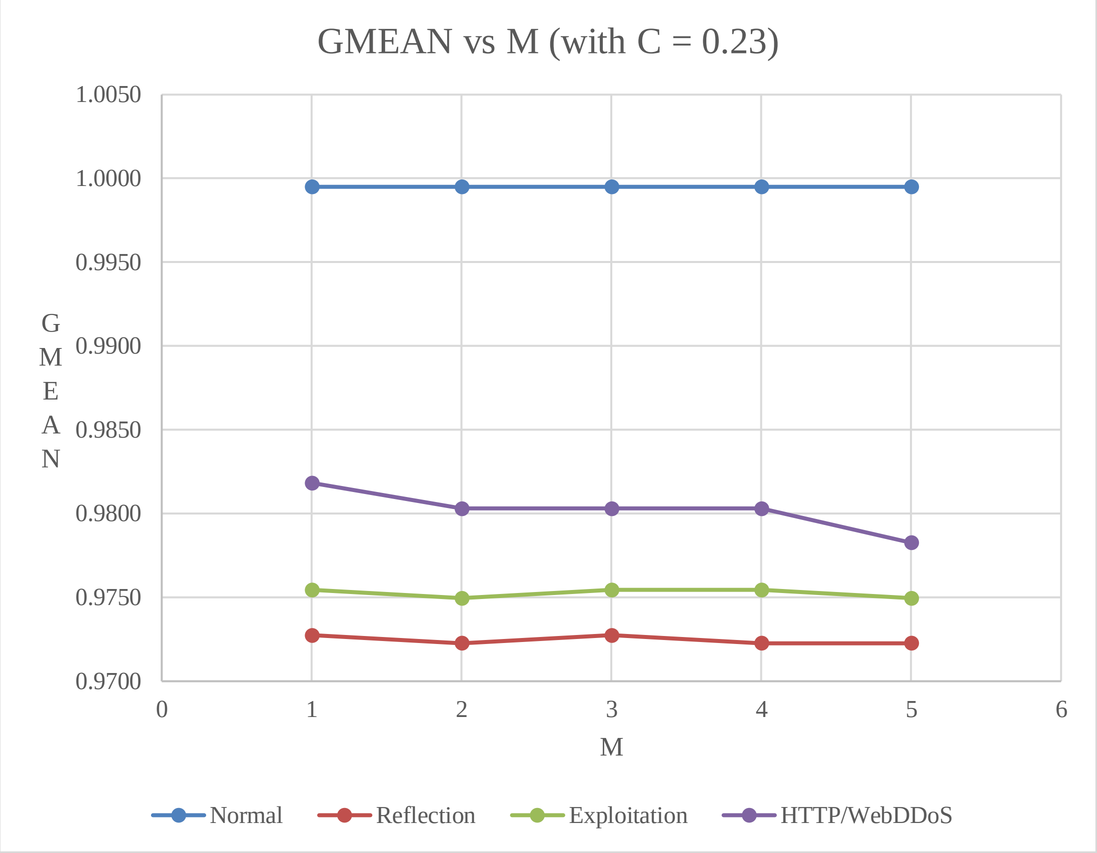


Figure B-28: Plot of GMEAN values for each individual class for values of M tested (with C = 0.23) for Level 1 Grouped Classification.

Table B-32: Table of Net Change values for every evaluation metric used by value of M tested (with C = 0.23) for Level 1 Grouped Classification.

Value Tested for M	Net Change				
	TPR	PREC	F1	AUC	GMEAN
1	+0.4%	+0.2%	+0.002	-0.003	+0.0025
2 (Default Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
3 (Selected value)	**+0.1%**	**+0.2%**	**+0.001**	**+0.002**	**+0.0010**
4	+0.1%	+0.2%	+0.002	+0.004	+0.0005
5	-0.4%	+0.2%	-0.001	+0.004	-0.0020

Level 2 Grouped Classification

Table B-33: Table of General Evaluation Metrics Obtained from Level 2 Grouped Classification and Optimisation for each value of C tested in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value from C = 0.25, while light red denotes lower than default value)

Value of C Tested	Evaluation Metric (Weighted Average)							Standard Deviation					
	ACC	TPR	PREC	F1	AUC	TNR	GMEAN	ACC	TPR	PREC	F1	AUC	TNR
0.1	95.90%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.2	95.92%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.21	95.92%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.22	95.93%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.23	95.93%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.24	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.25 (Default value)	**95.94%**	**96%**	**96%**	**0.96**	**1.00**	**100%**	**0.9798**	**0.10**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
0.26	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.27	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.28	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.29	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.3	95.94%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.4	95.92%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
0.5	95.90%	96%	96%	0.96	1.00	100%	0.9798	0.10	0.00	0.00	0.00	0.00	0.00

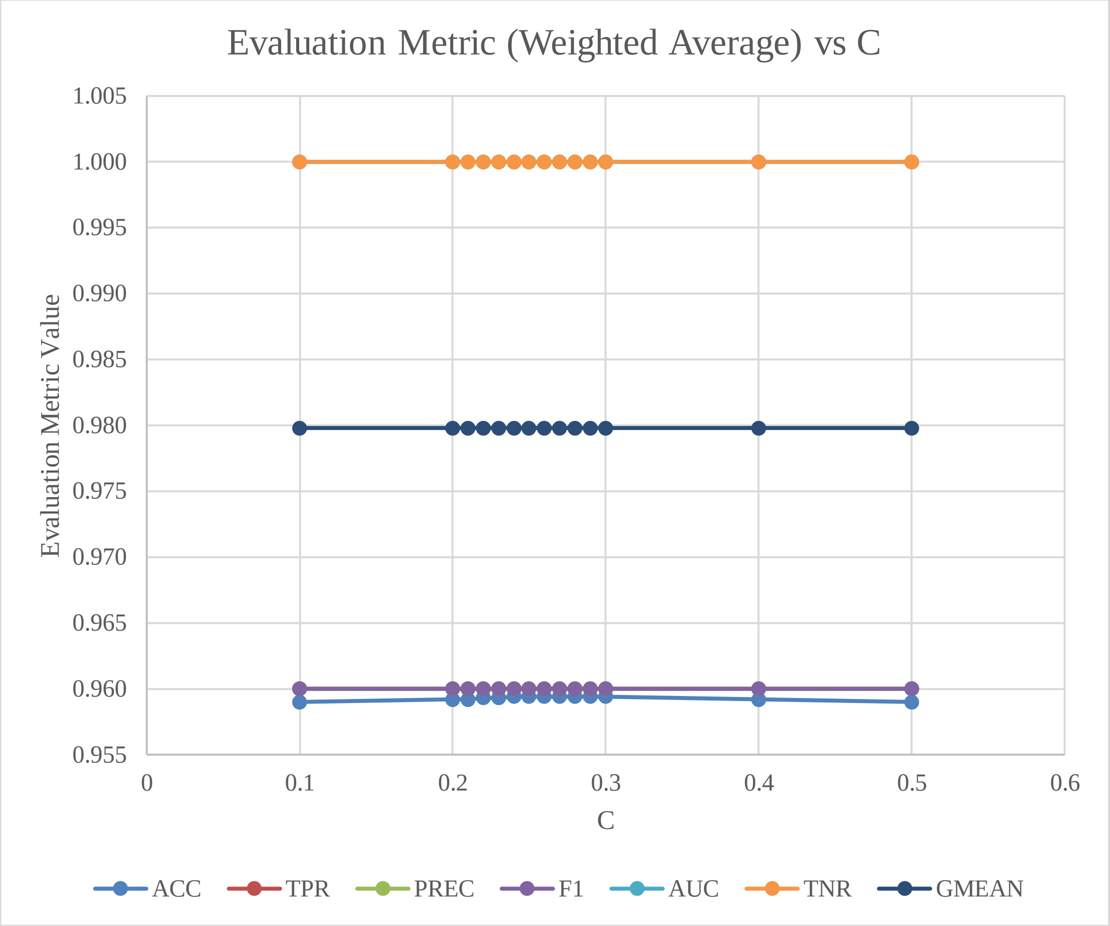


Figure B-29: Graph of General Evaluation Metrics value for Level 2 Grouped Classification by value of C tested in WEKA.

Table B-34: Table of True Positive Rates (TPR) obtained for each class for each value of C tested in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested For C	TPR for Class							Net change (TPR)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
0.1	99.9%	98.6%	77.0%	99.2%	99.8%	94.6%	95.2%	-1.8%
0.2	99.9%	98.6%	77.3%	99.2%	99.8%	94.5%	95.9%	-0.9%
0.21	99.9%	98.6%	77.3%	99.2%	99.8%	94.5%	95.7%	-1.1%
0.22	99.9%	98.6%	77.3%	99.2%	99.8%	94.5%	95.7%	-1.1%
0.23	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
0.24	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
0.25 (Default value)	**100.0%**	**98.5%**	**77.4%**	**99.2%**	**99.8%**	**94.6%**	**96.6%**	**0.0%**
0.26	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
0.27	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
0.28	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
0.29	99.9%	98.6%	77.5%	99.2%	99.8%	94.3%	95.7%	-1.1%
0.3	99.9%	98.6%	77.6%	99.2%	99.8%	94.2%	95.7%	-1.1%
0.4	99.9%	98.6%	77.8%	99.2%	99.8%	93.7%	95.4%	-1.7%
0.5	99.9%	98.6%	77.9%	99.2%	99.7%	93.2%	95.7%	-1.9%

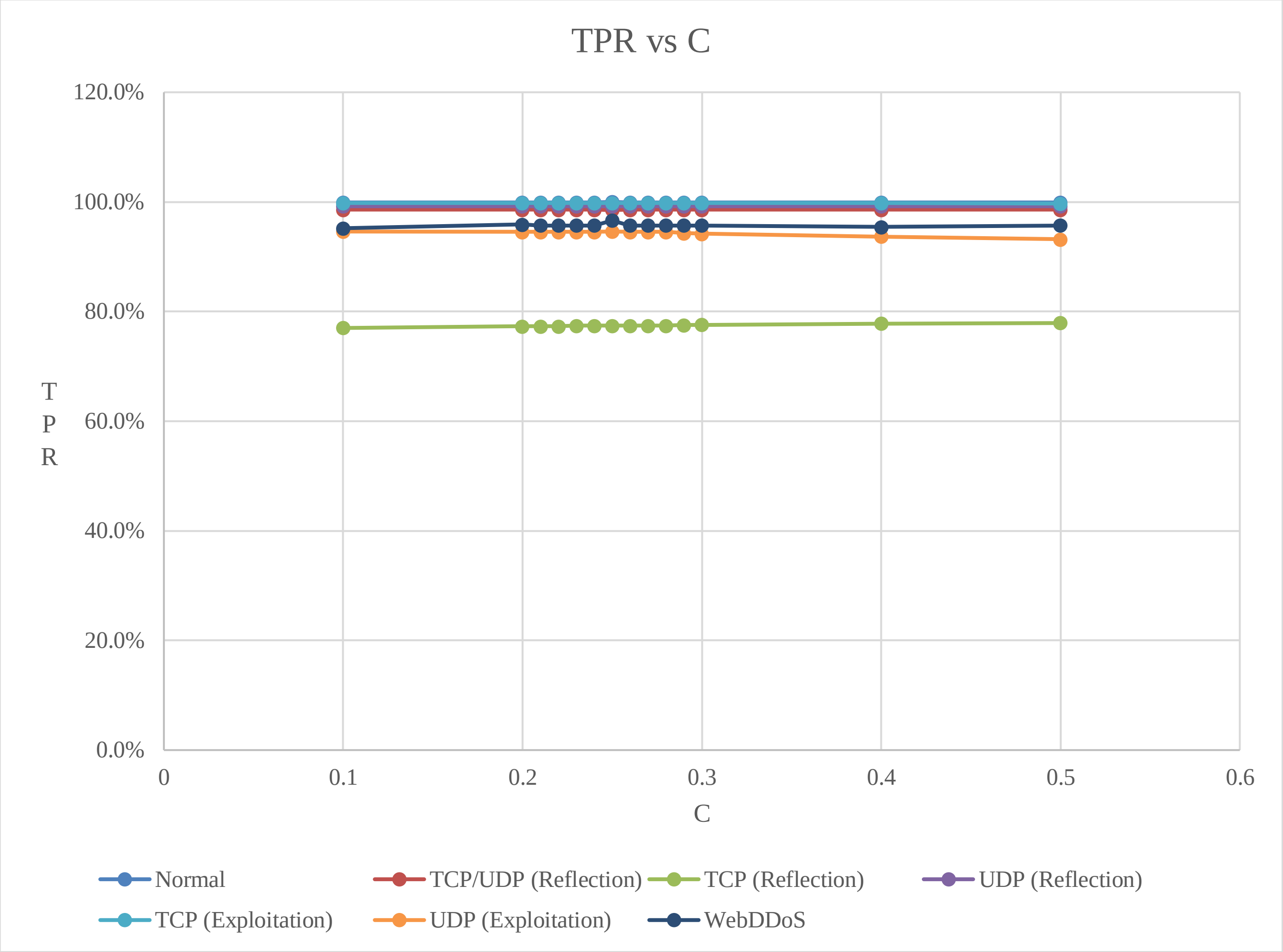


Figure B-30: Plot of TPR values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-35: Table of Precision (PREC) obtained for each class for each value of C tested in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested For C	PREC for Class							Net change (PREC)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	WebDDoS	
0.1	99.8%	98.4%	95.8%	99.7%	94.2%	72.8%	91.1%	-2.4%
0.2	99.9%	98.5%	95.8%	99.8%	94.3%	73.0%	91.3%	-1.6%
0.21	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.3%	-1.7%
0.22	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.3%	-1.7%
0.23	99.9%	98.6%	95.7%	99.8%	94.3%	73.0%	91.3%	-1.6%
0.24	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.3%	-1.7%
0.25 (Default value)	**99.9%**	**98.5%**	**95.8%**	**99.8%**	**94.3%**	**73.1%**	**92.8%**	**0.0%**
0.26	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.5%	-1.5%
0.27	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.1%	-1.9%
0.28	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.1%	-1.9%
0.29	99.9%	98.5%	95.6%	99.8%	94.3%	73.1%	91.1%	-1.9%
0.3	99.9%	98.5%	95.5%	99.8%	94.3%	73.1%	91.1%	-2.0%
0.4	99.9%	98.5%	95.1%	99.8%	94.3%	73.3%	91.3%	-2.0%
0.5	99.9%	98.5%	94.9%	99.7%	94.3%	73.3%	91.5%	-2.1%

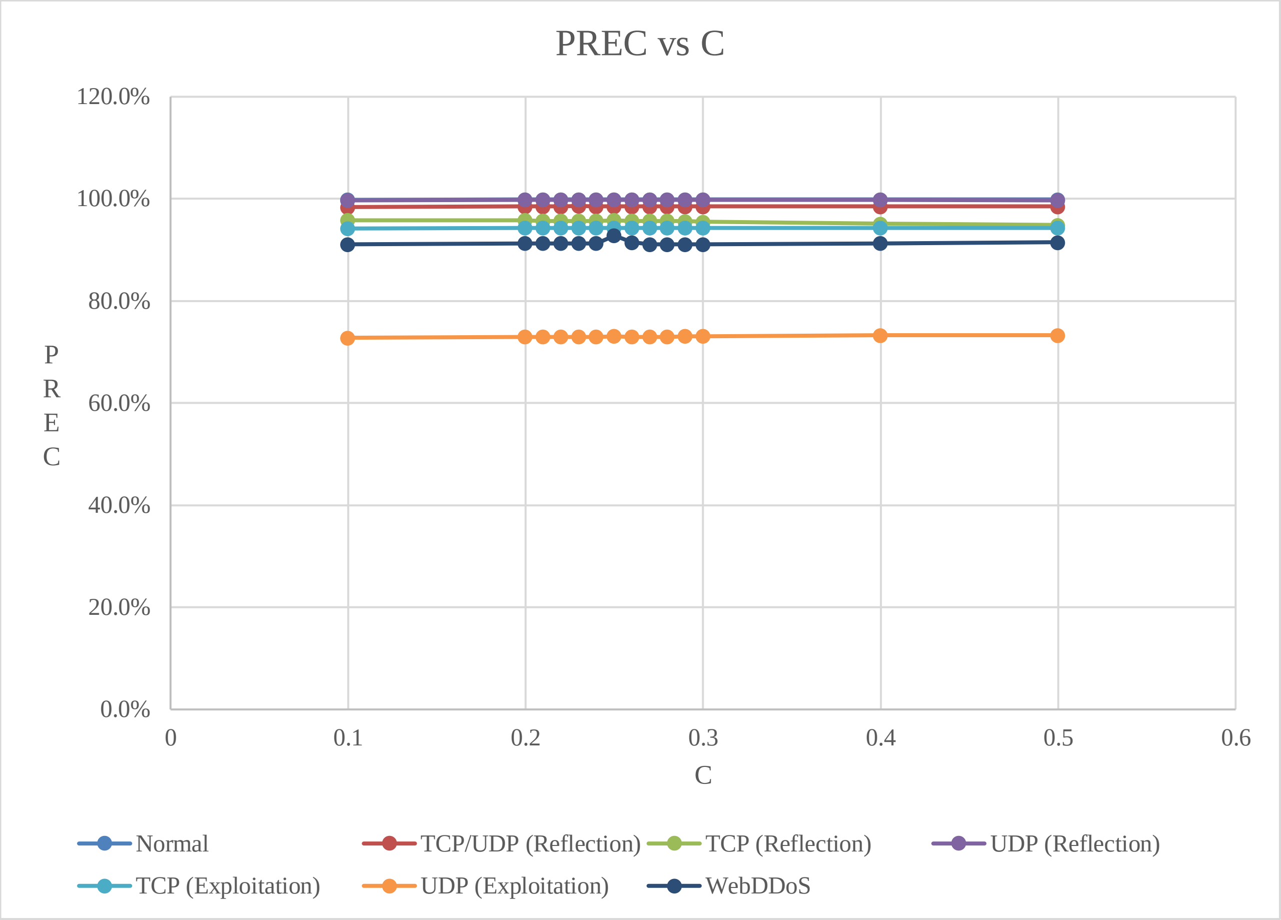


Figure B-31: Plot of PREC values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-36: Table of F1-Measure (F1) values obtained for each class for each value of C tested in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested For C	F1 for Class							Net change (F1)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
0.1	0.999	0.985	0.854	0.995	0.969	0.823	0.931	-0.020
0.2	0.999	0.985	0.855	0.995	0.969	0.824	0.936	-0.013
0.21	0.999	0.985	0.855	0.995	0.969	0.824	0.934	-0.015
0.22	0.999	0.985	0.855	0.995	0.969	0.824	0.934	-0.015
0.23	0.999	0.985	0.856	0.995	0.969	0.824	0.934	-0.014
0.24	0.999	0.985	0.856	0.995	0.969	0.824	0.934	-0.014
0.25 (Default value)	**0.999**	**0.985**	**0.856**	**0.995**	**0.970**	**0.825**	**0.946**	**0.000**
0.26	0.999	0.985	0.856	0.995	0.969	0.824	0.935	-0.013
0.27	0.999	0.985	0.856	0.995	0.969	0.824	0.933	-0.015
0.28	0.999	0.985	0.856	0.995	0.969	0.824	0.933	-0.015
0.29	0.999	0.985	0.856	0.995	0.969	0.824	0.933	-0.015
0.3	0.999	0.985	0.856	0.995	0.969	0.823	0.933	-0.016
0.4	0.999	0.985	0.856	0.995	0.969	0.822	0.933	-0.017
0.5	0.999	0.985	0.856	0.995	0.969	0.821	0.935	-0.016

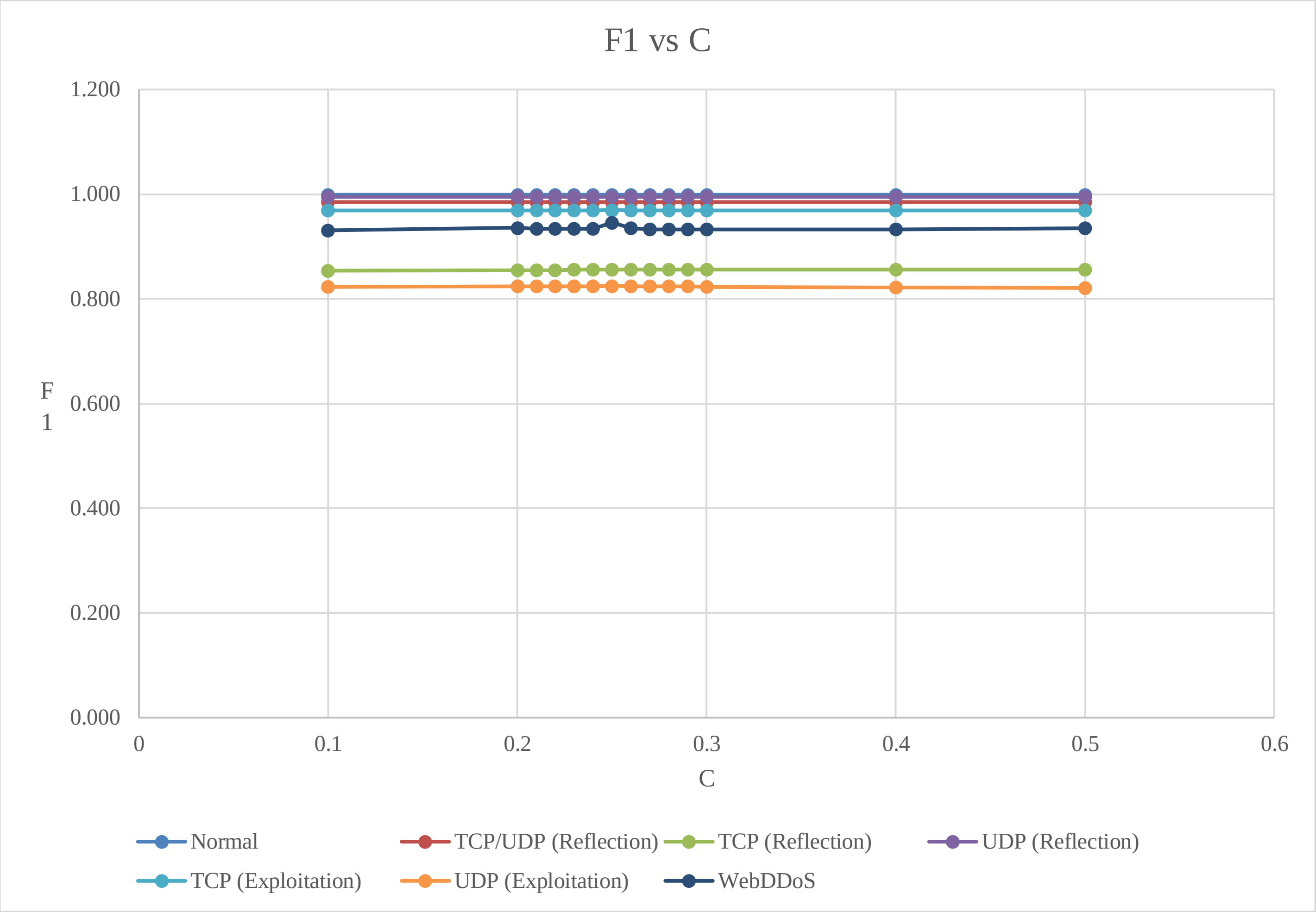


Figure B-32: Plot of F1 values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-37: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of C tested in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested For C	AUC for Class							Net change (AUC)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
0.1	1.000	0.997	0.984	0.999	0.999	0.982	0.995	+0.001
0.2	1.000	0.997	0.984	0.999	0.999	0.983	0.991	-0.002
0.21	1.000	0.997	0.985	0.999	0.999	0.983	0.990	-0.002
0.22	1.000	0.997	0.985	0.999	0.999	0.983	0.990	-0.002
0.23	1.000	0.997	0.985	0.999	0.999	0.983	0.990	-0.002
0.24	1.000	0.997	0.985	0.999	0.999	0.983	0.990	-0.002
0.25 (Default value)	**1.000**	**0.997**	**0.984**	**0.999**	**0.999**	**0.983**	**0.993**	**0.000**
0.26	1.000	0.997	0.985	0.999	0.999	0.983	0.990	-0.002
0.27	1.000	0.997	0.985	0.999	0.999	0.983	0.988	-0.004
0.28	1.000	0.997	0.984	0.999	0.999	0.983	0.988	-0.005
0.29	1.000	0.997	0.985	0.999	0.999	0.983	0.988	-0.004
0.3	1.000	0.997	0.985	0.999	0.999	0.983	0.988	-0.004
0.4	1.000	0.997	0.986	0.999	0.999	0.984	0.989	-0.001
0.5	1.000	0.997	0.986	0.999	0.999	0.985	0.987	-0.002

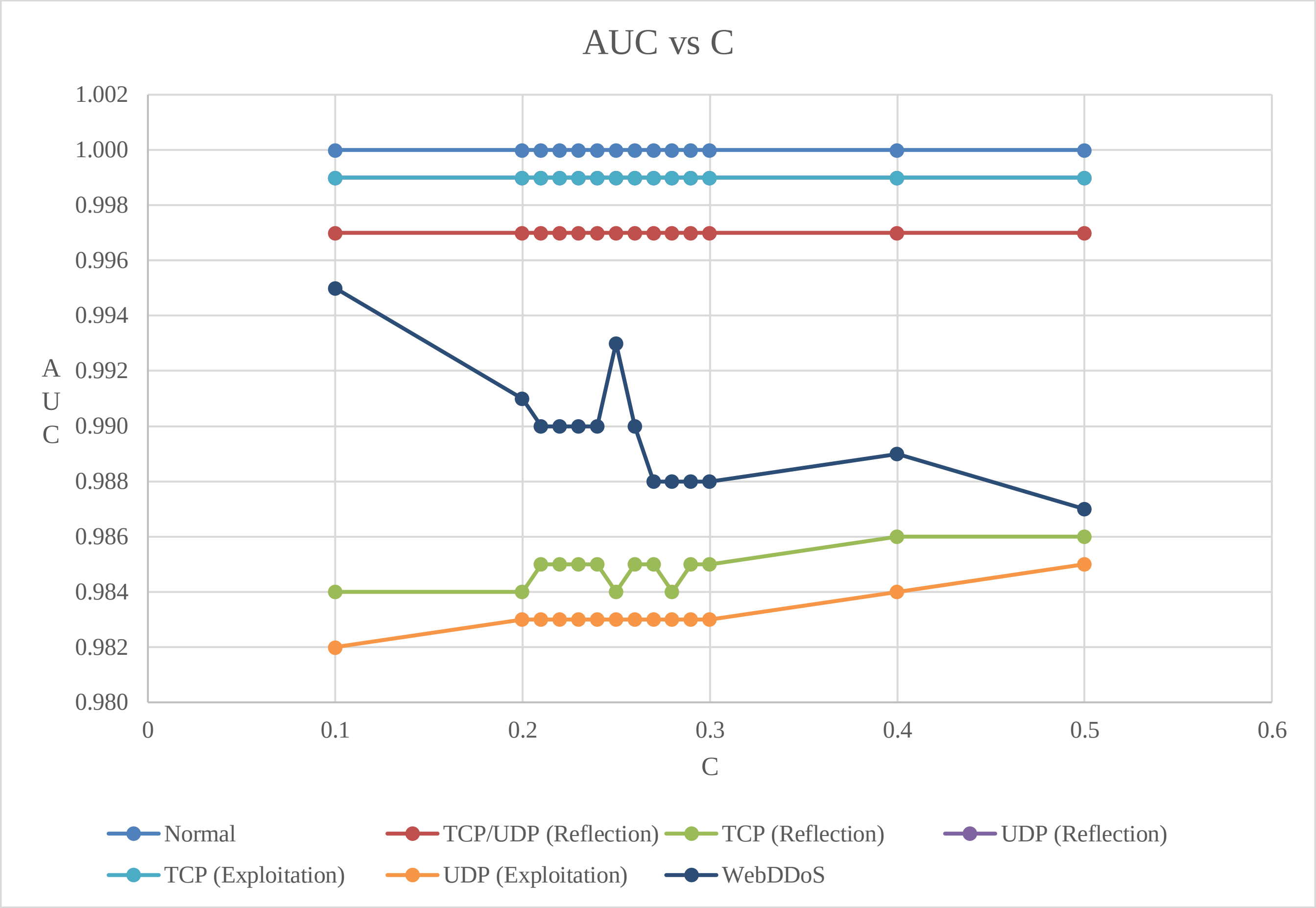


Figure B-33: Plot of AUC values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-38: Table of False Positive Rates (FPR) obtained for each class for each value of C tested in WEKA for Level 2 Grouped Classification. (Light green highlight denotes lower than value obtained with default value from C = 0.25, while light red denotes higher than default value obtained)

Value Tested For C	Class						
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS
0.1	0.1%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.2	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.21	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.22	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.23	0.0%	0.5%	0.6%	0.1%	0.4%	3.0%	0.0%
0.24	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.25 (Default value)	**0.0%**	**0.5%**	**0.5%**	**0.1%**	**0.4%**	**3.0%**	**0.0%**
0.26	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.27	0.0%	0.5%	0.6%	0.1%	0.4%	3.0%	0.0%
0.28	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
0.29	0.0%	0.5%	0.6%	0.1%	0.4%	3.0%	0.0%
0.3	0.0%	0.5%	0.6%	0.1%	0.4%	2.9%	0.0%
0.4	0.0%	0.5%	0.6%	0.1%	0.4%	2.9%	0.0%
0.5	0.0%	0.5%	0.7%	0.1%	0.4%	2.9%	0.0%

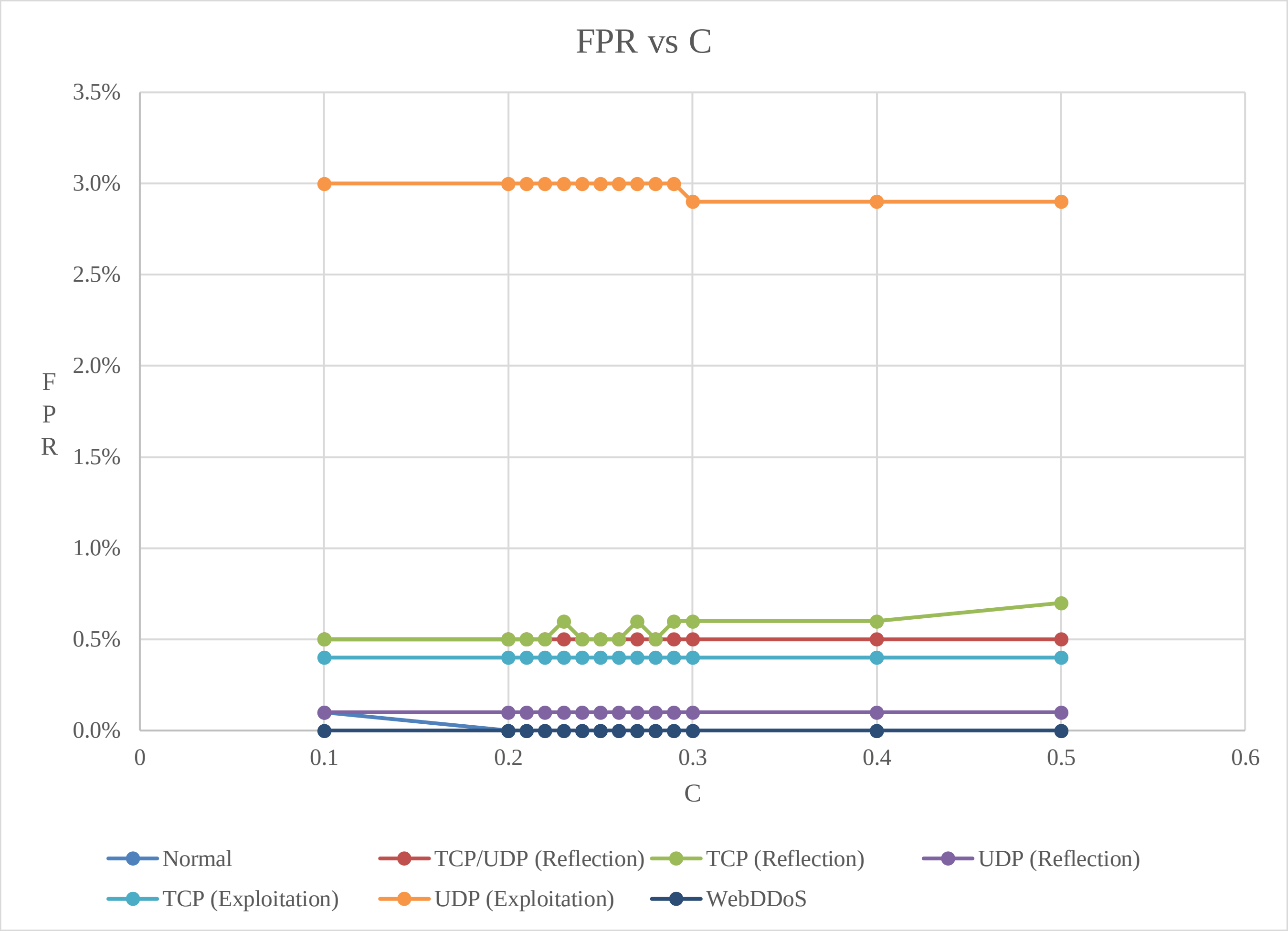


Figure B-34: Plot of FPR values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-39: Table of G-Mean (GMEAN) values obtained for each class for each value of C tested in for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from C = 0.25, while light red denotes lower than default value obtained)

Value Tested For C	Class							Net change (GMEAN)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
0.1	0.9990	0.9905	0.8753	0.9955	0.9970	0.9579	0.9757	-0.0099
0.2	0.9995	0.9905	0.8770	0.9955	0.9970	0.9574	0.9793	-0.0046
0.21	0.9995	0.9905	0.8770	0.9955	0.9970	0.9574	0.9783	-0.0057
0.22	0.9995	0.9905	0.8770	0.9955	0.9970	0.9574	0.9783	-0.0057
0.23	0.9995	0.9905	0.8771	0.9955	0.9970	0.9574	0.9783	-0.0055
0.24	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9783	-0.0051
0.25 (Default value)	**1.0000**	**0.9900**	**0.8776**	**0.9955**	**0.9970**	**0.9579**	**0.9829**	**0.0000**
0.26	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9783	-0.0051
0.27	0.9995	0.9905	0.8771	0.9955	0.9970	0.9574	0.9783	-0.0055
0.28	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9783	-0.0051
0.29	0.9995	0.9905	0.8777	0.9955	0.9970	0.9564	0.9783	-0.0060
0.3	0.9995	0.9905	0.8783	0.9955	0.9970	0.9564	0.9783	-0.0054
0.4	0.9995	0.9905	0.8794	0.9955	0.9970	0.9538	0.9767	-0.0084
0.5	0.9995	0.9905	0.8795	0.9955	0.9965	0.9513	0.9783	-0.0098

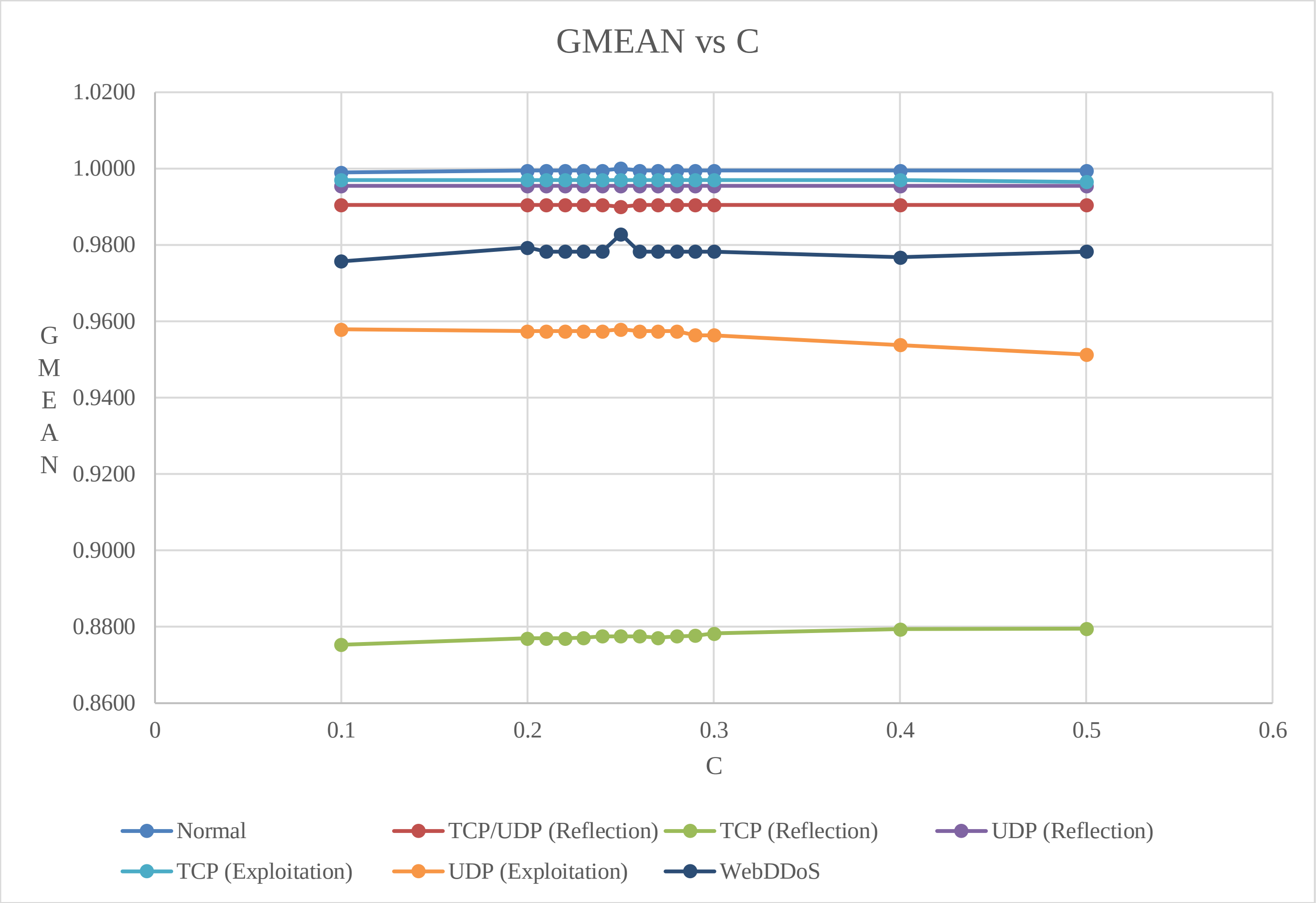


Figure B-35: Plot of GMEAN values for each individual class for values of C tested using WEKA for Level 2 Grouped Classification.

Table B-40: Table of Net Change values for every evaluation metric used by value of C tested for Level 2 Grouped Classification.

Value Tested For C	Net Change				
	TPR	PREC	F1	AUC	GMEAN
0.1	-1.8%	-2.4%	-0.02	+0.001	-0.0099
0.2	-0.9%	-1.6%	-0.013	-0.002	-0.0046
0.21	-1.1%	-1.7%	-0.015	-0.002	-0.0057
0.22	-1.1%	-1.7%	-0.015	-0.002	-0.0057
0.23	-1.0%	-1.6%	-0.014	-0.002	-0.0055
0.24	-1.0%	-1.7%	-0.014	-0.002	-0.0051
0.25 (Default & Selected value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
0.26	-1.0%	-1.5%	-0.013	-0.002	-0.0051
0.27	-1.0%	-1.9%	-0.015	-0.004	-0.0055
0.28	-1.0%	-1.9%	-0.015	-0.005	-0.0051
0.29	-1.1%	-1.9%	-0.015	-0.004	-0.0060
0.3	-1.1%	-2.0%	-0.016	-0.004	-0.0054
0.4	-1.7%	-2.0%	-0.017	-0.001	-0.0084
0.5	-1.9%	-2.1%	-0.016	-0.002	-0.0098

Table B-41: Table of General Evaluation Metrics Obtained from Level 2 Grouped Classification and Optimisation for each value of M tested (with C = 0.25) in WEKA for 5 Repetitions and Two-Tailed T-Testing with Confidence Factor of 0.05. (Light green highlight denotes higher value than obtained from default value from C = 0.25, while light red denotes lower than default value)

Value of M Tested	Evaluation Metric							Standard Deviation					
	ACC (%)	TPR (%)	PREC	F1	AUC	TNR (%)	GMEAN	ACC (%)	TPR (%)	PREC	F1	AUC	TNR (%)
1	95.94	96	96	0.96	1.00	100	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
2 (Default Value)	**95.94**	**96**	**96**	**0.96**	**1.00**	**100**	**0.9798**	**0.10**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
3	95.94	96	96	0.96	1.00	100	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
4	95.93	96	96	0.96	1.00	100	0.9798	0.10	0.00	0.00	0.00	0.00	0.00
5	95.92	96	96	0.96	1.00	100	0.9798	0.10	0.00	0.00	0.00	0.00	0.00

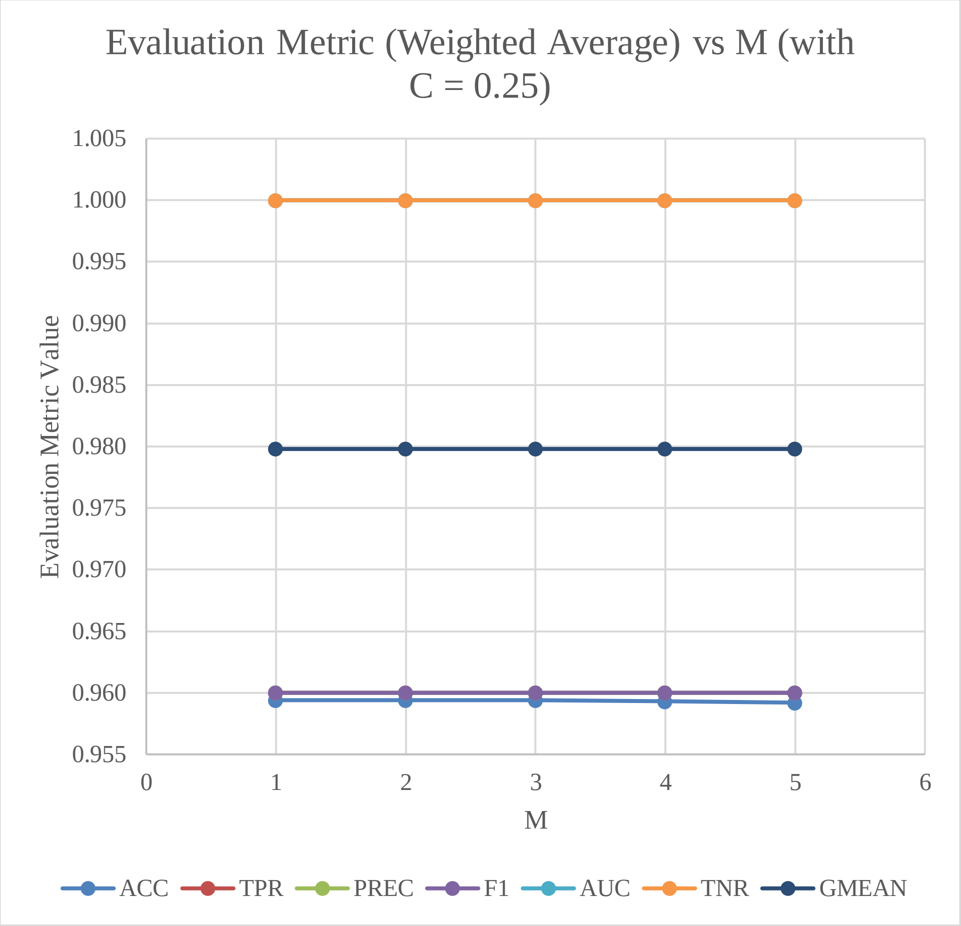


Figure B-36: Graph of General Evaluation Metrics value for Level 2 Grouped Classification by value of M tested (with C = 0.25) in WEKA.

Table B-42: Table of True Positive Rates (TPR) obtained for each class for each value of M tested (with C = 0.25) in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value of M Tested	TPR for Class							Net Change (TPR)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
1	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.9%	-0.8%
2 (Default Value)	**100.0%**	**98.5%**	**77.4%**	**99.2%**	**99.8%**	**94.6%**	**96.6%**	**0.0%**
3	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
4	99.9%	98.6%	77.4%	99.2%	99.8%	94.5%	95.7%	-1.0%
5	99.9%	98.5%	77.4%	99.2%	99.8%	94.6%	95.2%	-1.5%

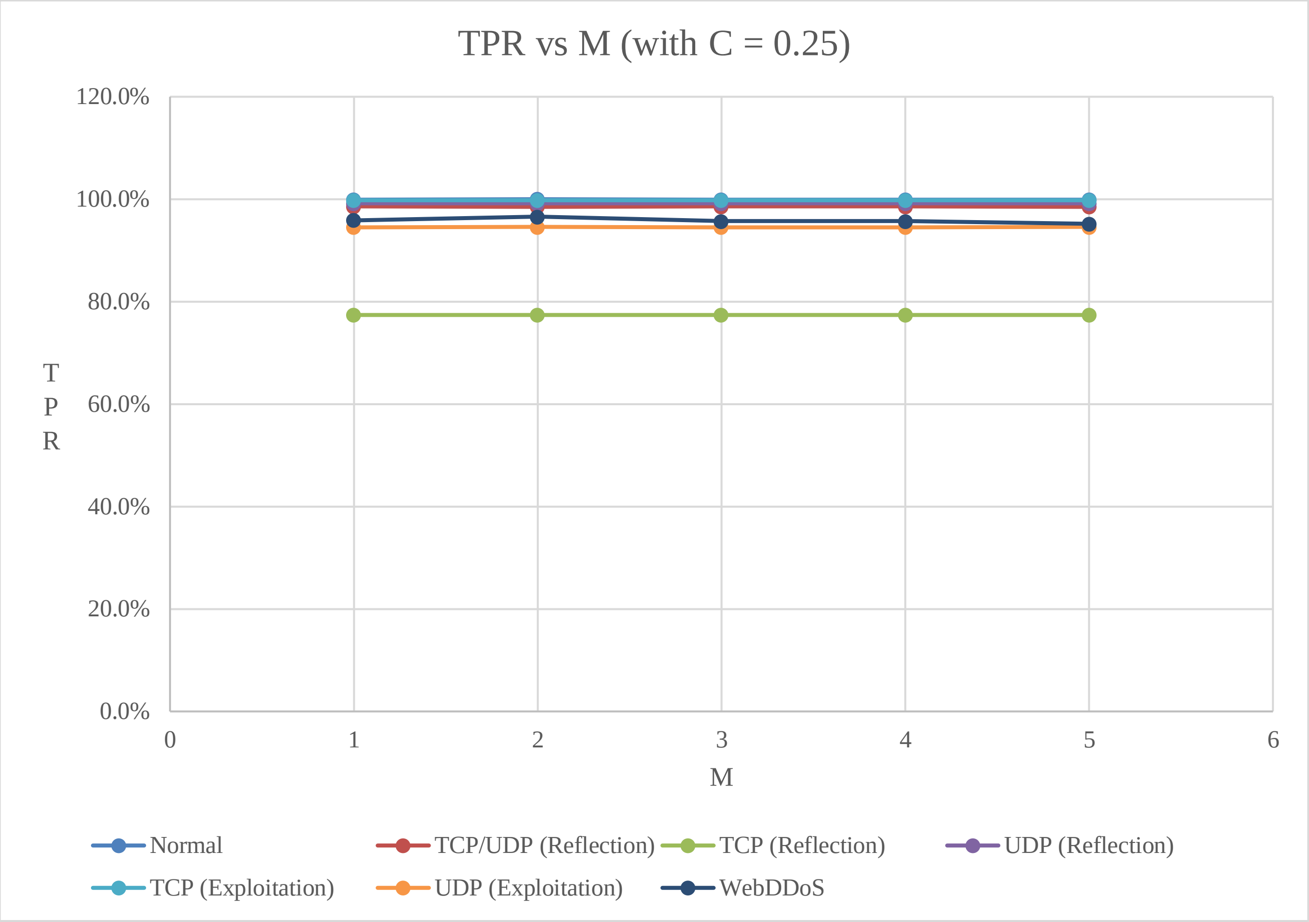


Figure B-37: Plot of TPR values for each individual class for values of M tested (with C = 0.25) using WEKA for Level 2 Grouped Classification.

Table B-43: Table of Precision (PREC) values obtained for each class for each value of M tested (with C = 0.25) in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	PREC for Class							Net change (PREC)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
1	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.5%	-1.5%
2 (Default Value)	**99.9%**	**98.5%**	**95.8%**	**99.8%**	**94.3%**	**73.1%**	**92.8%**	**0.0%**
3	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.5%	-1.5%
4	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.5%	-1.5%
5	99.9%	98.5%	95.7%	99.8%	94.3%	73.0%	91.1%	-1.9%

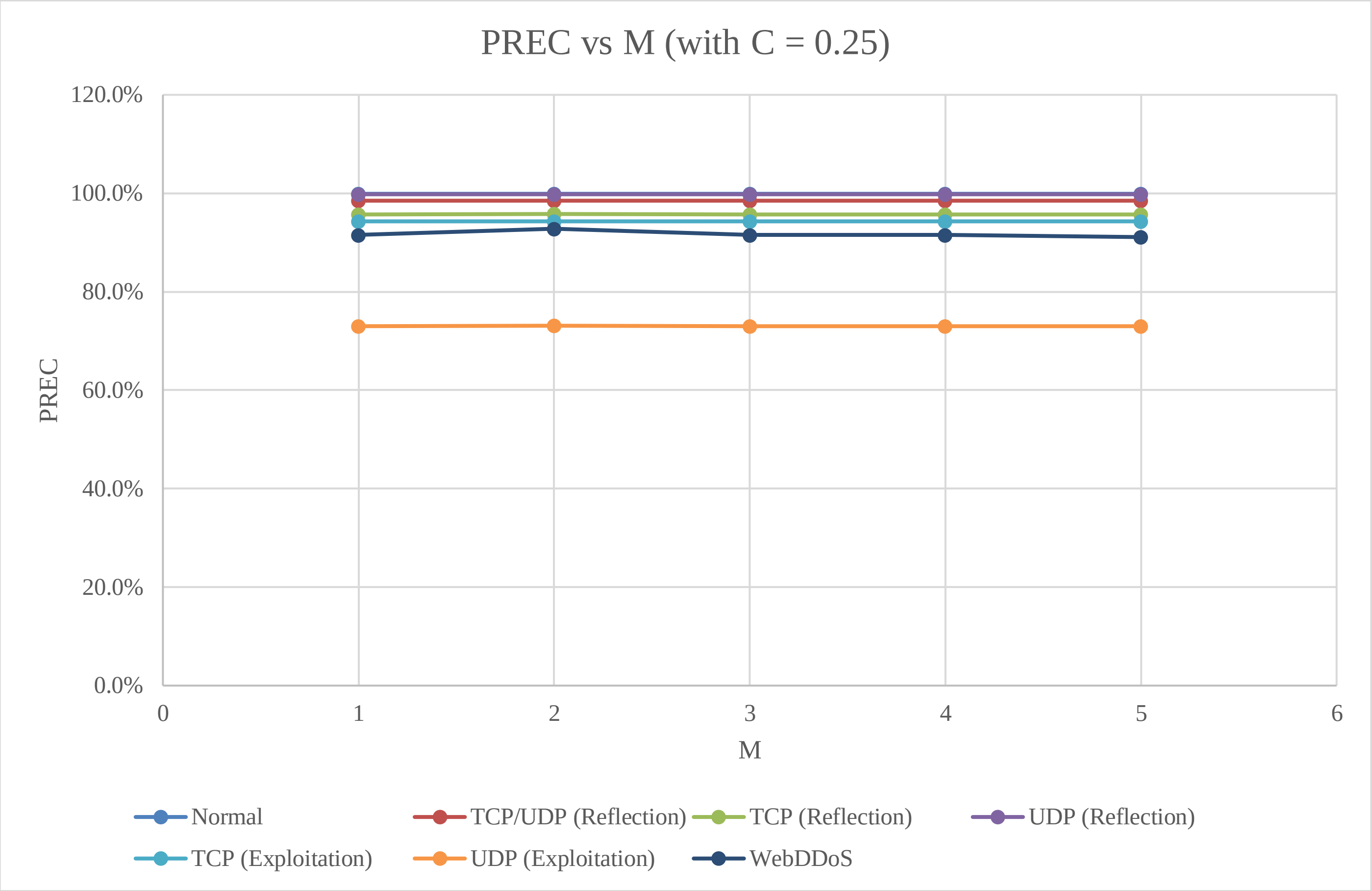


Figure B-38: Plot of PREC values for each individual class for values of M tested (with C = 0.25) using WEKA for Level 2 Grouped Classification.

Table B-44: Table of F-Measure (F1) values obtained for each class for each value of M tested (with C = 0.25) in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	F1 for Class							Net change (F1)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
1	0.999	0.985	0.856	0.995	0.969	0.824	0.937	-0.011
2 (Default Value)	**0.999**	**0.985**	**0.856**	**0.995**	**0.970**	**0.825**	**0.946**	**0.000**
3	0.999	0.985	0.856	0.995	0.969	0.824	0.935	-0.013
4	0.999	0.985	0.856	0.995	0.969	0.824	0.935	-0.013
5	0.999	0.985	0.856	0.995	0.969	0.824	0.931	-0.017

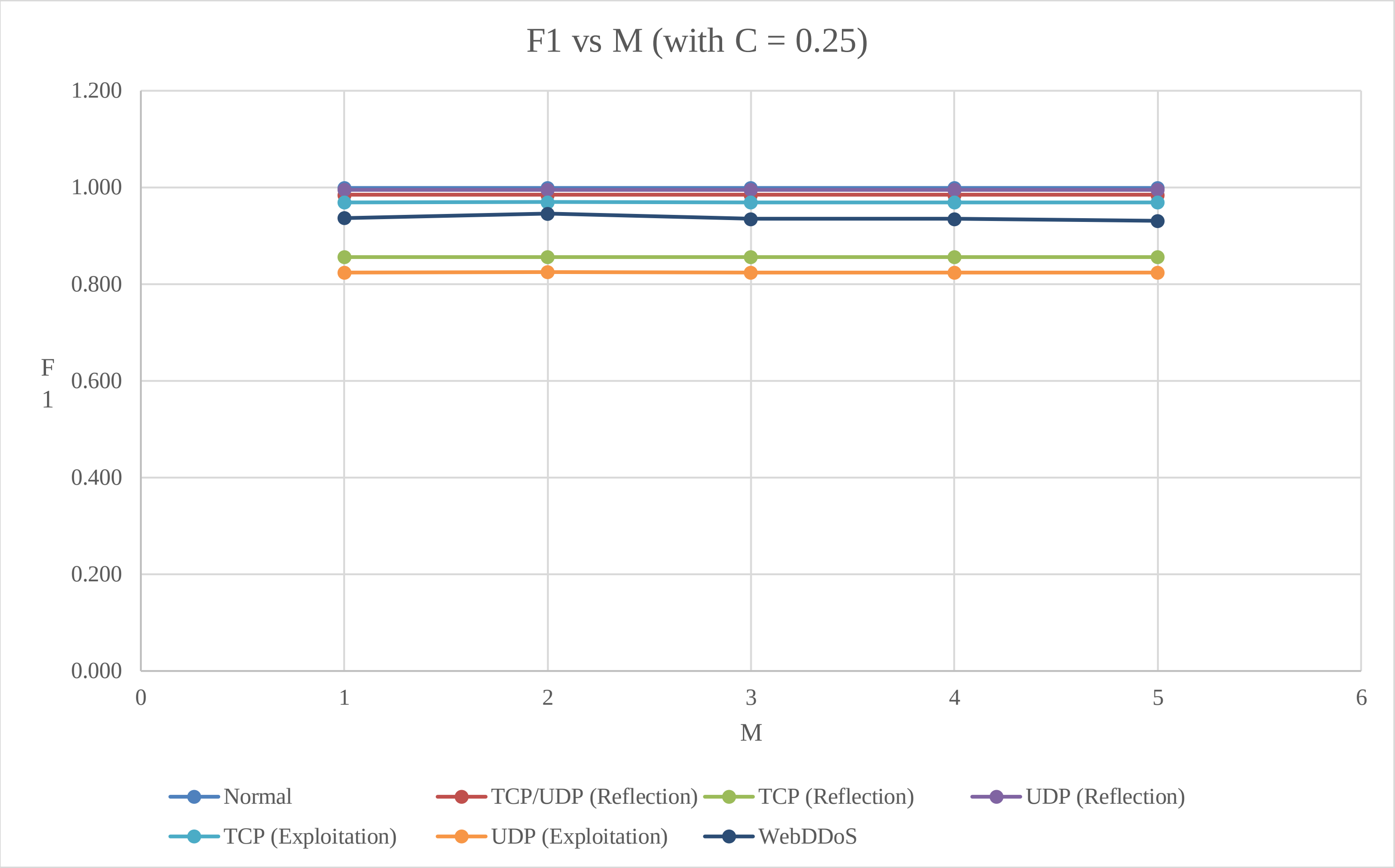


Figure B-39: Plot of F1 values for each individual class for values of M tested (with C = 0.25) using WEKA for Level 2 Grouped Classification.

Table B-45: Table of Area Under ROC Curve (AUC) values obtained for each class for each value of M tested (with C = 0.25) in WEKA for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	AUC for Class							Net change (AUC)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
1	1.000	0.997	0.984	0.999	0.999	0.982	0.988	-0.006
2 (Default value)	**1.000**	**0.997**	**0.984**	**0.999**	**0.999**	**0.983**	**0.993**	**0.000**
3	1.000	0.997	0.985	1.000	0.999	0.983	0.994	+0.003
4	1.000	0.998	0.985	1.000	0.999	0.983	0.994	+0.004
5	1.000	0.998	0.985	1.000	0.999	0.983	0.994	+0.004

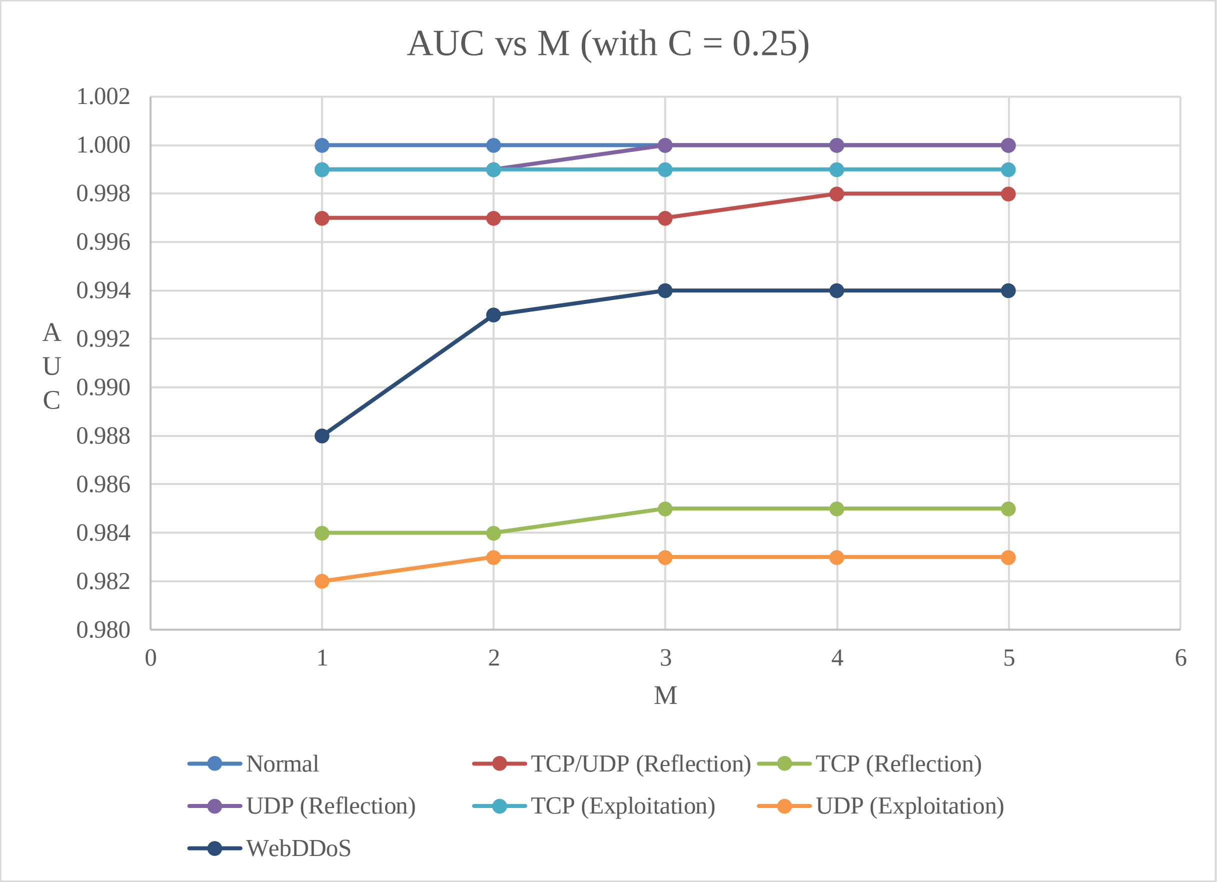


Figure B-40: Plot of AUC values for each individual class for values of M tested (with C = 0.25) using WEKA for Level 2 Grouped Classification.

Table B-46: Table of False Positive Rates (FPR) obtained for each class for each value of M tested (with C = 0.25) in WEKA for Level 2 Grouped Classification. (Light green highlight denotes lower than value obtained with default value from M = 2, while light red denotes higher than default value obtained)

Value Tested For M	FPR for Class						
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS
1	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
2 (Default Value)	**0.0%**	**0.5%**	**0.5%**	**0.1%**	**0.4%**	**3.0%**	**0.0%**
3	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
4	0.0%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%
5	0.1%	0.5%	0.5%	0.1%	0.4%	3.0%	0.0%

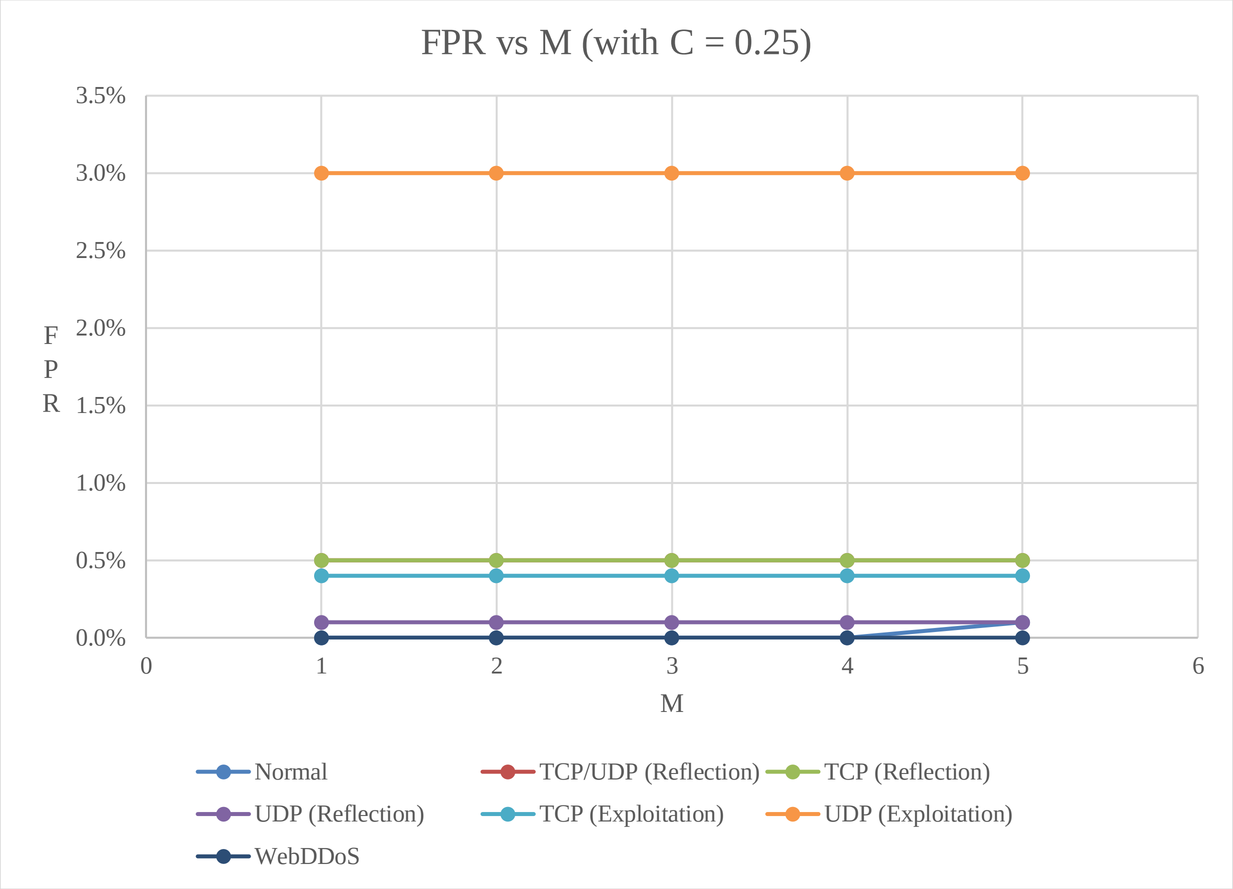


Figure B-41: Plot of FPR values for each individual class for values of M tested (with C = 0.25) using WEKA for Level 2 Grouped Classification.

Table B-47: Table of G-Mean (GMEAN) values obtained for each class for each value of M tested (with C = 0.25) for Level 2 Grouped Classification. (Light green highlight denotes higher than value obtained with default value from M = 2, while light red denotes lower than default value obtained)

Value Tested For M	GMEAN for Class							Net change (GMEAN)
	Normal	TCP/UDP (Reflection)	TCP (Reflection)	UDP (Reflection)	TCP (Exploitation)	UDP (Exploitation)	Web DDoS	
1	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9793	-0.0041
2 (Default value)	**1.0000**	**0.9900**	**0.8776**	**0.9955**	**0.9970**	**0.9579**	**0.9829**	**0.0000**
3	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9783	-0.0051
4	0.9995	0.9905	0.8776	0.9955	0.9970	0.9574	0.9783	-0.0051
5	0.9990	0.9900	0.8776	0.9955	0.9970	0.9579	0.9757	-0.0081

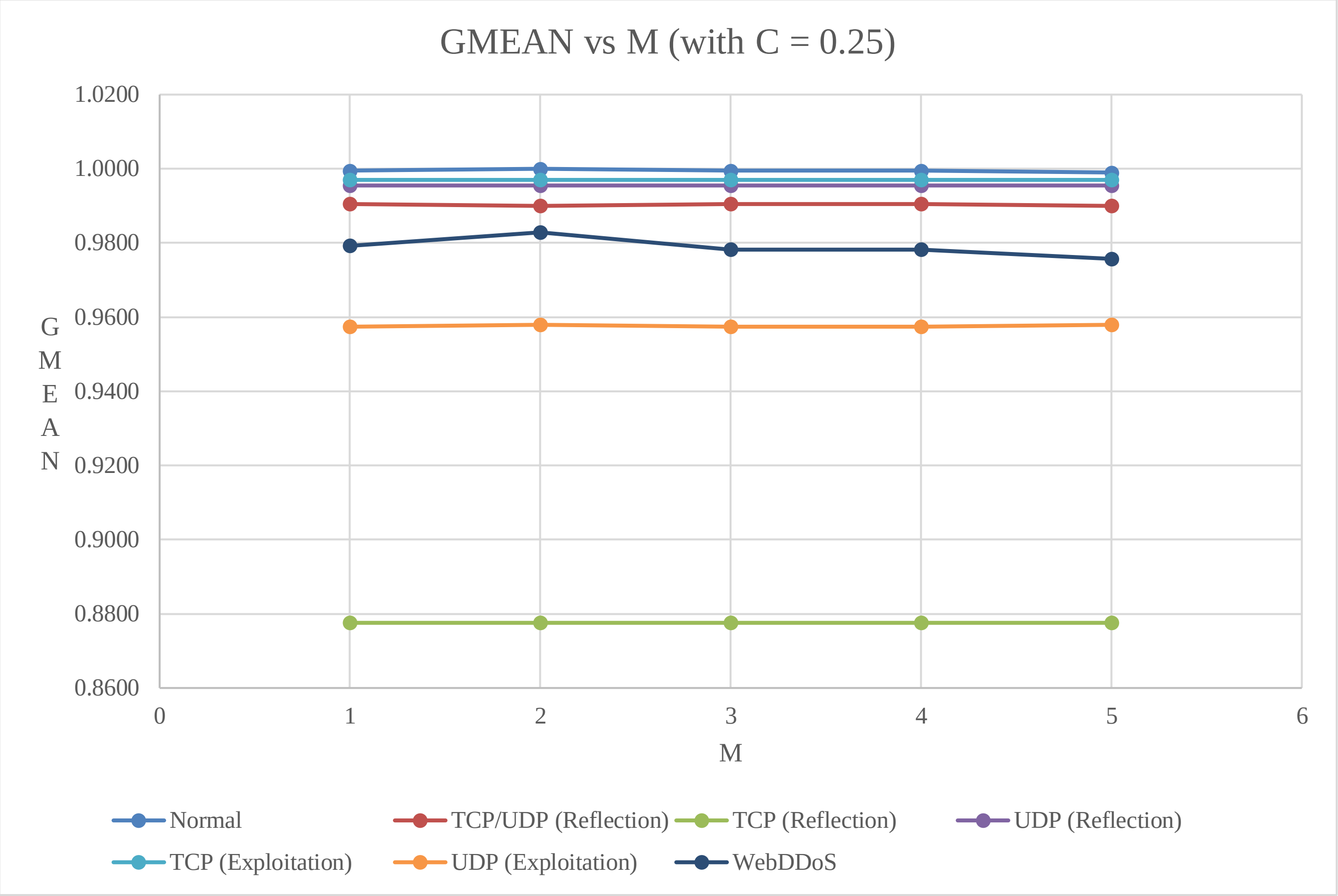


Figure B-42: Plot of GMEAN values for each individual class for values of M tested (with C = 0.25) for Level 2 Grouped Classification.

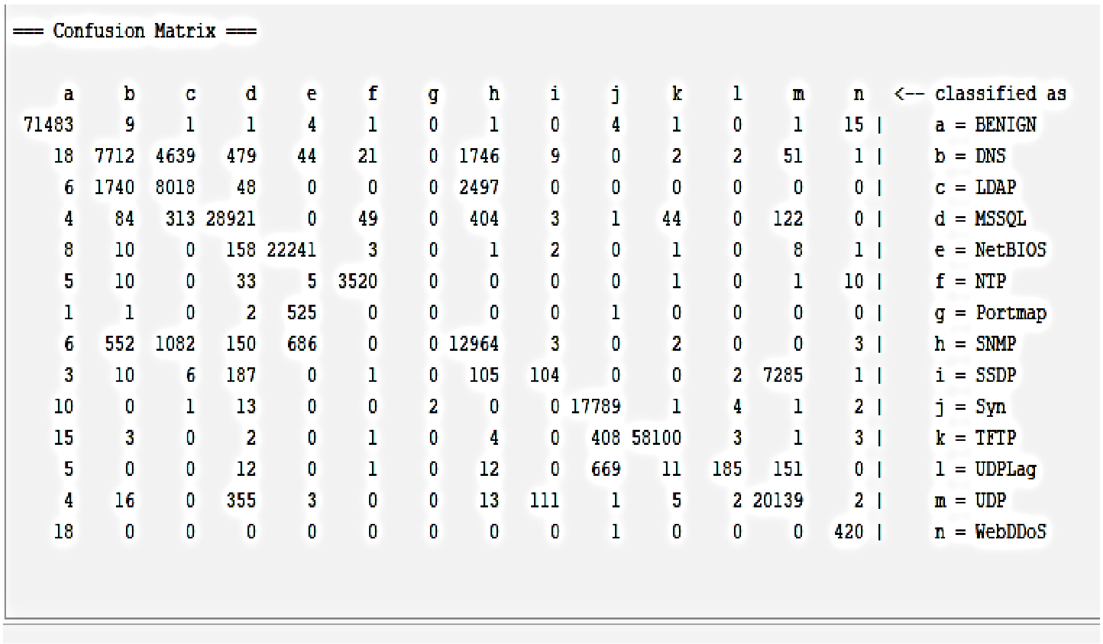
Table B-48: Table of Net Change values for every evaluation metric used by value of M tested (with C = 0.25) for Level 2 Grouped Classification.

Value Tested for M	Net Change				
	TPR	PREC	F1	AUC	GMEAN
1	-0.8%	-1.5%	-0.011	-0.006	-0.0041
2 (Default & Selected Value)	**0.0%**	**0.0%**	**0.000**	**0.000**	**0.0000**
3	-1.0%	-1.5%	-0.013	+0.003	-0.0051
4	-1.0%	-1.5%	-0.013	+0.004	-0.0051
5	-1.5%	-1.9%	-0.017	+0.004	-0.0081

APPENDIX C: Miscellaneous Classification Results

Table C-1: Detailed Evaluation Metric table for Individual Class using parameters C = 0.23 and M = 3 for J48 Classifier in Preliminary Ungrouped Classification under 10-fold Cross Validation using WEKA, for comparison with Level 1 Grouped Classification Results.

Class	Evaluation Metric (Individual Class)						
	TPR (%)	PREC (%)	F1	AUC	FPR (%)	TNR (%)	GMEAN
BENIGN	99.9	99.9	0.999	1.000	0.1	99.9	0.999 000
DNS	52.4	76.0	0.620	0.974	0.9	99.1	0.720 614
LDAP	65.1	57.0	0.608	0.973	2.3	97.7	0.746 251
MSSQL	96.6	95.4	0.959	0.995	0.6	99.4	0.979 900
NetBIOS	99.1	94.6	0.968	0.997	0.5	99.5	0.992 998
NTP	98.2	97.9	0.980	0.999	0.0	100.0	0.990 959
Portmap	0.0	0.0	0.000	0.959	0.0	100.0	0.000 000
SNMP	83.9	73.0	0.781	0.981	1.8	98.2	0.907 688
SSDP	1.3	44.8	0.026	0.958	0.4	99.6	0.113 789
SYN	99.8	94.3	0.970	0.999	0.4	99.6	0.996 999
TFTP	99.2	99.9	0.996	1.000	0.0	100.0	0.995 992
UDPLag	17.7	93.4	0.297	0.976	0.0	100.0	0.420 714
UDP	97.5	72.5	0.832	0.984	3.0	97.0	0.972 497
WebDDoS	95.7	91.7	0.936	0.994	0.0	100.0	0.978 264



```
=== Confusion Matrix ===

     a     b     c     d     e     f     g     h     i     j     k     l     m     n   <-- classified as
 71483     9     1     1     4     1     0     1     0     4     1     0     1    15 |     a = BENIGN
    18  7712  4639   479    44    21     0  1746     9     0     2     2    51     1 |     b = DNS
     6  1740  8018    48     0     0     0  2497     0     0     0     0     0     0 |     c = LDAP
     4    84   313 28921     0    49     0   404     3     1    44     0   122     0 |     d = MSSQL
     8    10     0   158 22241     3     0     1     2     0     1     0     8     1 |     e = NetBIOS
     5    10     0    33     5  3520     0     0     0     0     1     0     1    10 |     f = NTP
     1     1     0     2   525     0     0     0     0     1     0     0     0     0 |     g = Portmap
     6   552  1082   150   686     0     0 12964     3     0     2     0     0     3 |     h = SNMP
     3    10     6   187     0     1     0   105   104     0     0     2  7285     1 |     i = SSDP
    10     0     1    13     0     0     2     0     0 17789     1     4     1     2 |     j = Syn
    15     3     0     2     0     1     0     4     0   408 58100     3     1     3 |     k = TFTP
     5     0     0    12     0     1     0    12     0   669    11   185   151     0 |     l = UDPLag
     4    16     0   355     3     0     0    13   111     1     5     2 20139     2 |     m = UDP
    18     0     0     0     0     0     0     0     0     1     0     0     0   420 |     n = WebDDoS
```

Figure C-1: Confusion Matrix produced by WEKA using parameters C = 0.23 and M = 3 for J48 Classifier in Preliminary Ungrouped Classification under 10-fold Cross Validation, for comparison with Level 1 Grouped Classification Results.

Table C-2: Summarised Confusion Matrix generated in WEKA for Preliminary Ungrouped Classification (with C = 0.23 and M = 3) from Figure C-1 after accounting for 24 961 DDoS attack labels incorrectly classified as other DDoS attack labels and the 37 BENIGN labels classified as DDoS attack classes, for comparison with Level 1 Grouped Classification Results.

Actual/Predicted	BENIGN	DDoS (Any class)
BENIGN	71 483	38
DDoS (Any class)	103	205 074