DDOS DETECTION USING ENSEMBLE TECHNIQUES IN MACHINE LEARNING MODELS WITH A TAILORED COMPANION TOOL FOR CYBER SECURITY ANALYSTS

BY

Sidratul Muntaha Tuba
ID: 201-15-13585

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Dr. Sheak Rashed Haider Noori
Professor & Head
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised By

Dr. Ohidujjaman Tuhin
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

July 2024

APPROVAL

This Project titled "**DDoS Detection Using Ensemble Techniques in Machine Learning Models With a Tailored Companion Tool for Cyber Security Analysts**", submitted by **Sidratul Muntaha Tuba** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **13.07.2024**.

BOARD OF EXAMINERS

Narayan Ranjan Chakraborty (NRC) — **Board Chairman**
Associate Professor & Associate Head
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Md. Sadekur Rahman (SR) — **Internal Examiner 1**
Assistant Professor
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Mr. Tanvirul Islam (TI) — **Internal Examiner 2**
Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Manowarul Islam — **External Examiner**
Associate Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Professor & Head, Department of Computer Science and Engineering, Daffodil International University**. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Dr. Sheak Rashed Haider Noori
Professor & Head
Department of CSE
Daffodil International University

Co-Supervised by:

Dr. Ohidujjaman Tuhin
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

Sidratul Muntaha Tuba
ID: 201-15-13585
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, I express my heartiest thanks and gratefulness to almighty for His divine blessing making it possible for us to complete the final year project/internship successfully.

I am grateful and wish my profound indebtedness to **Dr. Sheak Rashed Haider Noori**, **Professor & Head**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of my supervisor in the field of "*Machine Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

I would like to express my heartiest gratitude to Dr. Sheak Rashed Haider Noori, **Professor & Head of the Department of CSE**, for his kind help in finishing my project and also to other faculty members and the staff of the Department of CSE, Daffodil International University.

I would like to thank my entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

With the increasing occurrence and complexity of cyber-threats, there is a need today to make use of highly performant and robust detection tools for advanced cyber-attacks such as Distributed Denial of Service (DDoS) attacks in the rapidly changing cyber-space-centric environment. In this paper, we propose a new ensemble approach that combines different machine learning models to improve the detection and classification of different DDoS attacks. We experimentally evaluate the proposed ensemble models for DDoS detection on a large and diverse but also imbalanced dataset of DDoS attack instances drawn from the CIC-DDoS2019. In this work, we analyze the influence of different ensemble voting classifiers to see the effects on the performance of the final models. Results show that 99.81% accurate detection class and 93.92% accurate classification class can be determined by our ensemble approach for identifying and classifying DDoS attacks which reflect the effectiveness and robustness of our method. We used evaluation metrics like validation accuracy and confusion matrices for the results revealed that our model is very interesting to detect cybersecurity problems such as DDoS. We are also writing a web app to plug in these performant detection, and recognition algorithms. By utilizing this platform, cyber security professionals around the world can have monitoring of their system logs and even be empowered to take necessary preemptive steps, thus creating an environment of heightened awareness and readiness for overall cyber security. In this paper, we are not only able to achieve a high detection and classification rate in DDoS attack detection and recognition by the proposed ensemble-based model, but also construct the premise of network security in academic research and network application especially when it comes to log file analysis.

TABLE OF CONTENTS

Contents	Page No
Board of Examiners	i
Declaration	ii
Acknowledgment	iii
Abstract	iv
CHAPTER	
CHAPTER 1: INTRODUCTION	**1-9**
1.1 Overview	1-2
1.2 Research Motivation	2-3
1.3 Problem Statement	3-4
1.4 Research Objectives	5
1.5 Rationale of the Study	5-6
1.6 Research Question	6
1.7 Expected Outcome	6-7
1.8 Report Layout	7-9
CHAPTER 2: BACKGROUND	**10-16**
2.1 Overview	10-11
2.2 Related Works	11-14
2.3 Scope of the Problem	14
2.4 Challenges	15-16
CHAPTER 3: RESEARCH METHODOLOGY	**17-33**
3.1 Proposed Methodology	17-18
3.2 Dataset Collection	18
3.3 Dataset Description	19-20
3.4 Data Cleaning	20-21
3.5 Data Pre-Processing	21-23
3.6 Feature Selection	23

3.7 Training Data and Testing Data 23
3.8 Target Columns 24-25
3.9 Machine Learning Algorithms Used 26-29
3.10 Ensemble Technique 29-31
3.11 Model Generation Process 31-32
3.12 Companion Tool 32-33

CHAPTER 4: EXPERIMENTAL RESULTS AND DISCUSSION 34-64

4.1 Experimental Setup 34
4.2 Experimental Results and Analysis 34-35
4.3 Outcome 35-55
4.4 Model Result Summary 56
4.5 Comparison with Previous Work 56-58
4.6 Companion Tool In Action 58-64

CHAPTER 5: IMPACT ON SOCIETY, ENVIRONMENT AND ETHICAL ASPECTS 65-66

5.1 Impact On Society 65
5.2 Impact On Environment 65
5.3 Ethical Aspects 65-66

CHAPTER 6: CONCLUSION AND FUTURE WORK 67-69

6.1 Summary of the Study 67
6.2 Conclusion 67-68
6.3 Recommendations 68
6.4 Future Work 68-69

APPENDIX 70-77

REFERENCES 78-80

LIST OF FIGURES

Figures	Page No
Figure 1.8: Report layout in flow diagram style	8
Figure 3.1: An Overview Diagram for DDoS Detection System	17
Figure 3.3: DDoS attacks classification based on [42]	20
Figure 3.5: Target Columns Encoded Results	22
Figure 3.8.1: Count plot of all Label Columns	24
Figure 3.8.2: Count plot of Selected Label Columns	25
Figure 3.8.3: Count plot of Class Column	25
Figure 4.3.1: Training and Validation Accuracy Curves for Random Forest Classifier for DDoS Classification	37
Figure 4.3.2: Training and Validation Accuracy Curves for Random Forest Classifier for DDoS Detection	37
Figure 4.3.3: Training and Validation Accuracy Curves for Decision Tree Classifier for DDoS Classification	38
Figure 4.3.4: Training and Validation Accuracy Curves for Decision Tree Classifier for DDoS Detection	38
Figure 4.3.5: Training and Validation Accuracy Curves for Support Vector Classifier for DDoS Classification	39
Figure 4.3.6 : Training and Validation Accuracy Curves for Support Vector Classifier for DDoS Detection	39
Figure 4.3.7: Training and Validation Accuracy Curves for Logistic Regression for DDoS Classification	40
Figure 4.3.8: Training and Validation Accuracy Curves for Logistic Regression for DDoS Detection	40
Figure 4.3.9: Training and Validation Accuracy Curves for K Neighbors Classifier for DDoS Classification	41

Figure 4.3.10: Training and Validation Accuracy Curves for K Neighbors Classifier for DDoS Detection	41
Figure 4.3.11: Training and Validation Accuracy Curves for Gradient Boosting Classifier for DDoS Classification	42
Figure 4.3.12: Training and Validation Accuracy Curves for Gradient Boosting Classifier for DDoS Detection	42
Figure 4.3.13: Training and Validation Accuracy Curves for AdaBoost Classifier for DDoS Classification	43
Figure 4.3.14: Training and Validation Accuracy Curves for AdaBoost Classifier for DDoS Detection	43
Figure 4.3.15: Training and Validation Accuracy Curves for Bagging Classifier for DDoS Classification	44
Figure 4.3.16: Training and Validation Accuracy Curves for Bagging Classifier for DDoS Detection	44
Figure 4.3.17: Training and Validation Accuracy Curves for Bernoulli Naive Bayes for DDoS Classification	45
Figure 4.3.18: Training and Validation Accuracy Curves for Bernoulli Naive Bayes for DDoS Detection	45
Figure 4.3.19: Training and Validation Accuracy Curves for Gaussian Naive Bayes for DDoS Classification	46
Figure 4.3.20: Training and Validation Accuracy Curves for Gaussian Naive Bayes for DDoS Detection	46
Figure 4.3.21: Training and Validation Accuracy Curves for Linear Discriminant Analysis for DDoS Classification	47
Figure 4.3.22: Training and Validation Accuracy Curves for Linear Discriminant Analysis for DDoS Detection	47
Figure 4.3.23: Training and Validation Accuracy Curves for Quadratic Discriminant Analysis for DDoS Classification	48

Figure 4.3.24: Training and Validation Accuracy Curves for Quadratic Discriminant Analysis for DDoS Detection	48
Figure 4.3.25: Training and Validation Accuracy Curves for Multilayer Perceptron Classifiers for DDoS Classification	49
Figure 4.3.26: Training and Validation Accuracy Curves for Multi-layer Perceptron Classifiers for DDoS Detection	49
Figure 4.3.27: Base ML Models Results for DDoS Detection	50
Figure 4.3.28: Base ML Models Results for DDoS Classification	50
Figure 4.3.29: Accuracy Scores For Final Ensemble Models	52
Figure 4.3.30: Ensemble Models Voting Accuracy Score	53
Figure 4.3.31: Confusion Matrix For DDoS Detection with final Detection Model	54
Figure 4.3.32: Classification Report For DDoS Detection with final Detection Model	54
Figure 4.3.33: Confusion Matrix For DDoS Classification with final Classification Model	55
Figure 4.3.34: Classification Report For DDoS Classification with final Classification Model	55
Figure 4.6.1: Companion Tool Website Guest View	58
Figure 4.6.2: Companion Tool Website Login Screen	59
Figure 4.6.3: Companion Tool Website Log File Upload Screen	59
Figure 4.6.4: Companion Tool Website Default Log File Screen	60
Figure 4.6.5: Companion Tool Website Demo Logs	60
Figure 4.6.6: Companion Tool Website Considering Feature Sets Announcement	61
Figure 4.6.7 : Companion Tool Website Log File Uploaded	62

Figure 4.6.8: Companion Tool Website Analyze Button For Log File Processing	62
Figure 4.6.9: Companion Tool Website Analyzing Log File and Waiting For Response	63
Figure 4.6.10: Companion Tool Website Analyzed Log File with Response	64

LIST OF TABLES

Tables	Page No
Table 2.2: Comparison of related works	13-14
Table 3.3.1: Label Target Column Unique Data Information	19-20
Table 3.3.2: Class Target Column Unique Data Information	20
Table 4.5: Comparison of proposed model accuracy results with previous models	57-58

CHAPTER 1
Introduction

1.1 Overview

Threat of cyber assaults has increased due to weaknesses in some internet-connected home devices that hackers may find an attractive target to conduct illegal activities. Due to the possibility of resulting in the leakage of confidential user data and harm to the essential infrastructure as well has been stated by Li et al. [1]. These types of major cyberattacks have been one of the greatest challenges to deal with and bring down by any other means compared to other malicious cyberattacks in the ecosystem. Quick growth and stealthy behavior make it difficult for scholars to identify these attacks as Vinayakumar et al. [2]. DDoS attacks also include many different kinds of distributed threats that try to flood a server that is internet-connected and a way of disrupting or completely disabling the server. Typically these attacks target specified applications such as web servers of banks, companies, online shopping sites, credit card payment networks etc. elaborated by Fu et al. [3]. To defend from any type of DDoS attack it is becoming quite essential to have sturdy systems in place. In order to build an efficient cyberattack detection system, it is important to design a robust machine learning model that minimizes false alarms while maintaining a high level of accuracy in identifying the presence of cyberattacks. With the growing threat landscape, the solution of the adoption of machine learning (ML) algorithms on DDoS attack detection and mitigation mechanisms has been established as a legitimate approach to improve cybersecurity defenses as mentioned by Priya et al. [4]. By using machine learning techniques stated by Shiaeles et al. [5], organizations can better identify and act upon malicious activity as it occurs and better secure their network architecture against future threats. Machine learning algorithms such as Random Forest, Decision Tree, Support Vector Classifiers, Logistic Regression, KNeighbors Classifiers, Gradient Boosting Classifier, AdaBoost Classifier, Bagging Classifier, Bernoulli NB, Gaussian NB, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Multi-layer Perceptron Classifier are necessary which will be used to automate the analysis of network traffic patterns and detect anomalies indicative of DDoS attacks. By doing systematic feature extraction, training and testing, ML-powered DDoS detection systems can separate ill-intentioned activities from legitimate user traffic, described by Rahman et al. [6]. This enables the possibility to intervene if needed. In this sense, the objective of this research is to evaluate how good machine learning techniques are for detecting and classifying DDoS attacks had been focused by Sarraf et al. [7].

Using a process of rigorous testing and comparing, this thesis aims to measure the accuracy of several machine learning tactics in detecting and classifying DDoS attacks accurately. To improve network security posture and resilience against evolving cyber threats by developing various ML driven DDoS detection techniques was elaborated by Idhammad et al. [8]. This will equip cybersecurity practitioners and decision makers to safeguard their digital assets and mitigate the impacts of DDoS attacks on critical infrastructure. Moreover, the use of emerging auxiliary tools not only improves the efficacy of the evaluation & response to the acquired insights but also encourages the evolution of machine learning algorithms in the cybersecurity space. By integrating visualization tools and supporting software during the examination of machine learning driven distributed denial-of-service (DDoS) detection systems, cybersecurity analysts can realize the status of the network as well as historical mitigation actions as stated by Wu et al. [9]. Another example is a custom application written to analyze log files and use machine learning models to determine whether network packets are indicative of DDoS attacks. To improve the accuracy and efficiency of threat detection, cyber intelligence analysts use cracked interfaces and interactive maps to detect irregularities, trends and commonalities has been emphasized by Choi et al. [10]. These are all things augmented by the statistical analysis capabilities of the companion application. It combines data from other machine learning based DDoS detection systems to produce actionable values that represent how well and reliably the system has been doing over time clarified by Santos et al. [11]. Incorporating visualization along with corresponding tools into machine-learning (ML) based cybersecurity systems is essential to secure critical infrastructure from emerging cyber threats. They serve to make well-considered decisions and actions, and they guarantee to secure the critical systems.

1.2 Research Motivation

In today's world of ever changing cybersecurity, Distributed Denial of Service (DDoS) attacks offer significant and ongoing threats. The growing sophistication of these attacks needs new detection methods, and machine learning (ML) has emerged as a vital component of this protection approach. Leveraging machine learning may dramatically improve DDoS attack detection and mitigation, allowing for adaptable and resilient responses to emerging threats. Furthermore, combining ensemble techniques which integrate many ML algorithms can improve detection reliability and resilience, outperforming solo methods. However, using ML in real-time cybersecurity scenarios presents several problems, such as data accessibility, model complexity, and scalability. To properly use ML for DDoS mitigation, it is necessary to create companion tools specifically for cybersecurity experts. These technologies

must offer extensive assistance in data interpretation and decision-making processes, strengthening network defenses.

i. **Detecting DDoS Attacks with Machine Learning:** This is the recurring problem addressed by machine learning in cybersecurity; new threats pose everyday disasters as machine learning attempts to overcome this. One way enterprises may hold hope for securing the network is by improving the detection capabilities using machine learning algorithms. To strengthen cyber defenses against ever changing threats, knowing the performance of these algorithms in detecting and preventing DDoS attacks is important.

ii. **Enhancing DDoS Detection through Ensemble Techniques:** While discrete machine learning algorithms have played a pivotal role in unraveling DDoS detection comprehension, using ensemble methods can further increase the reliability as well as the robustness of the detection mechanism. By using the collective intelligence and breadth of many algorithms, an organization can greatly enhance its ability to detect DDoS attacks. To improve the cyber defense strategies, more research is urgently needed to indicate the benefit of ensemble classifiers in the problem of DDoS detection.

iii. **Navigating Challenges in Implementing Machine Learning for DDoS Detection:** In real-time cyber-security environments, implementation of Machine Learning powered DDoS detection could potentially offer many benefits, but organizations face several challenges and constraints. Taking into account the principal challenges is essential for the successful implementation of a system, which includes data access, the complexity of models, the scalability of deployment, and interpretability. This will enable organizations to get the most out of machine learning in DDoS attack mitigation.

iv. **Empowering Cyber Security Analysts with Tailored Companion Tool:** Cybersecurity analysts work to interpret and act on the predictions with machine learning based DDoS detection systems helping to make them as accurate as possible. A custom companion tool with support from statistical analysis and visualization assists analysts in making informed decisions and effectively strengthening network security. It is important to understand how auxiliary tools help in the process of cyber defense, for machine learning to reduce the impact of DDoS attacks.

1.3 Problem Statement

The increase in the number and the intricacy of Distributed Denial of Service (DDoS) attacks is a major problem in the cybersecurity domain. For the organizations bearing the brunt of these risks, the challenges to protect their network infrastructure and services are formidable. With an ever-evolving set of threat vectors, most traditional

security measures fall short in effectively responding to altering tactics and counter-moves from malefactors, resulting in business disruptions and potential outages. Also, the complex and dynamic methods for orchestrating a DDoS, along with attackers using stealth methods, make it much more difficult to detect or classify such an attack.

i. **Dynamic Threat Landscape:** DDoS attacks are forever changing, hence new defense tools must be able to adapt to new threats on-the-fly. However, traditional security solutions are sometimes not able to keep pace with the rapidly evolving methodologies of attackers, and may therefore put the organizations at risk of being exploited.
ii. **Complexity and Variability of DDoS Tactics:** The wide range of attack types, coupled with the reality that DDoS tactics can be complex and variable make them very difficult to identify and address. In order to effectively handle a complete array of assaults - from ICMP flooding to downright complex application layer attacks, you must deploy the most advanced detection solutions, able to distinguish normal network traffic from the malicious side.
iii. **Stealth and Concealment Techniques:** Botnets and encryption have been used by the bad guys long enough to obfuscate their identity and the attack traffic making it hard to know who is attacking and who should receive the blame. Such tactics make it increasingly more difficult to accurately identify and thwart DDoS attacks.
iv. **Resource Limitations and Operational Challenges:** Due to resource constraints and limitations of operations, it is hard to have these measures in place to provide effective DDoS protection. The implementation of these measures may be beyond the human power and knowledge of the many organizations. Similarly, the lack of suitable network traffic visibility, shortage of trained security professionals, and prohibitive cost in time and money for special equipment and software collectively create the practical dilemma of fully comprehensive defense.

Machine learning offers several opportunities for incrementally enhancing DDoS detection and mitigation tools. That said, integrating machine learning into existing cybersecurity models poses its challenges. For machine learning based methods to be useful in DDoS defense, organizations need to overcome several challenges related to data availability, model complexity, deployment scalability, and interpretability. Legal and regulatory requirement compliance add a whole new layer of complexity to DDoS prevention solutions. Some important standards organizations may have to adhere to, while corrective action to fix compliance issues can be taken proactively. This is to save them from getting fines, penalties and damage to their image.

1.4 Research Objectives

To address the crucial problem of Distributed Denial of Service (DDoS) assaults, this study intends to use sophisticated machine learning (ML) approaches to improve detection and mitigation measures. The research focuses on how machine learning may not only predict and prevent DDoS assaults, but also enhance the practical applicability and efficacy of detection systems. The emphasis is on integrating ensemble approaches to use the combined capabilities of many ML models, resulting in enhanced DDoS detection performance. Furthermore, the work aims to improve the scalability and efficiency of these systems by comparing different dataset configurations and preprocessing methods. Finally, our research aims to enhance the area of cybersecurity by creating advanced ensemble models and approaches that dramatically improve defenses against DDoS assaults.

i. **Applying Machine Learning for DDoS Detection:** Research how machine learning could anticipate and prevent DDoS. Also, discover how machine learning algorithms could practically improve at detecting DDoS attacks.
ii. **Enhancing Ensemble Techniques:** However, treating the problem as using ensemble approaches to integrate results from multiple machine learning models to achieve better performance in DDoS detection. In the present dissertation, we investigate using ensemble techniques that can improve DDoS detection by merging the strengths of individual machine learning models.
iii. **Improving Scalability and Efficiency:** Evaluate different dataset setups along with classifications, and distinct preprocessing methods to improve the scalability and accuracy of DDoS detection. It focuses on an evaluation of how dataset configurations & pre-processing methods can be parameter tuned for the scalability & efficiency in DDoS detection systems.
iv. **Advancing Cybersecurity:** Improve the area of cybersecurity by developing more sophisticated ensemble models to detect DDoS attacks. The research discusses how machine learning improvements can reinforce cyber security against DDoS attacks.

1.5 Rationale of the Study

With Distributed Denial of Service (DDoS) attacks becoming more complex and frequent, organizations are now more vulnerable to the loss of integrity and availability of their critical online services. This demonstrates the necessity for novel methods of detection of DDoS attacks, as the advantages of traditional security solutions often fall short in the face of these ever adapting threats. It is because the study intends to solve mandating practical challenges such as limited resources and meeting legal compliances whilst implementing an effective DDoS protection.

The research aims to arm organizations with knowledge and technologies to enhance their defense systems, recognizing that being able to see cyber security threats through log files quickly and effectively is a critical component of combating DDoS attacks. This project targets reinforcing the immunity against DDoS attacks overall, and protects critical infrastructure against such challenges, specifically reducing potential interruptions of essential services, by identifying the relevant stumbling blocks and designing the appropriate solutions.

The research also suggests additional technologies necessary to do so beyond the standard cybersecurity measures. These complementary solutions, built to complement the tools existing security measures provide, are absolutely imperative when it comes to the rapid detection, investigation of a DDoS assault. Supporting tooling increases the effectiveness and efficiency of DDoS detection and classification operations, using data visualization and predictive modeling.

This study aims to sharpen DDoS prevention to better defend against an ever-evolving threat landscape and to increase digital ecosystem resilience. This is supposed to be a detailed research and consulting organization which helps enterprises from around the world to secure their digital assets, protect networks in time of DDoS attacks and ensure high availability of services.

1.6 Research Question

i. What is the extent to which the paper's machine learning algorithms can detect and counteract Distributed Denial of Service (DDoS) attacks?
ii. In what ways does the Ensemble Technique improve DDoS Detection?
iii. For DDoS detection, what function does the Cyber Security Analyst's Tailored Companion Tool fulfill?

1.7 Expected Outcome

The research provides a detailed evaluation of a number of machine learning algorithms including Random Forest, Decision Tree, Support Vector Classifiers, Logistic Regression, KNeighbors, Gradient Boosting, AdaBoost, Bagging Classifier, Bernoulli Naive Bayes, Gaussian Naive Bayes, Linear Discriminant Analysis, Quadratic Discriminant Analysis, and Multi-layer Perceptron Classifier, with a focus on performance as to the classification and detection of Distributed Denial of Service (DDoS) attacks. This research aims to evaluate each of these strategies in their individual contribution to effectively predicting DDoS attacks using extensive testing and comparison. The paper studies the effect of ensemble methods such as hard and soft voting integration in an effort to increase their dependability and robustness.

Besides evaluating machine learning models, the study also looks at how companion technologies can be integrated to improve DDoS detection, as well. Such additional tools, such as data visualization and predictive modeling and many more effectively detect fraud packets in log files. Furthermore, the research is intended to identify the best practices of feature selection, data sampling, and hyperparameter tuning in machine learning models to make detection more efficient and accurate. The report gives an understanding of real-world challenges to using machine learning driven DDoS detection algorithms to solve them, and provides guidelines for further work; even if this work is not currently present in literature as the focus of research. The study helps gain a detailed insight into the capabilities and constraints of machine learning techniques in detection and classifying DDoS attacks in log files. The work makes practical recommendations to improve the efficacy and efficiency of DDoS detection solutions when integrated with additional protection mechanisms during cybersecurity operations.

1.8 Report Layout

This presentation is to make people aware of the importance of DDoS detection in the modern cybersecurity scenario and will contain the details of our research on this topic. We aim to apply this approach towards an insightful study of the root problems of DDoS attacks and the suggestions of their consequent detection approaches which can be employed in protecting digital infrastructures from threats that DDoS pose. The report layout in a flow diagram can be seen in the following Figure 1.8,

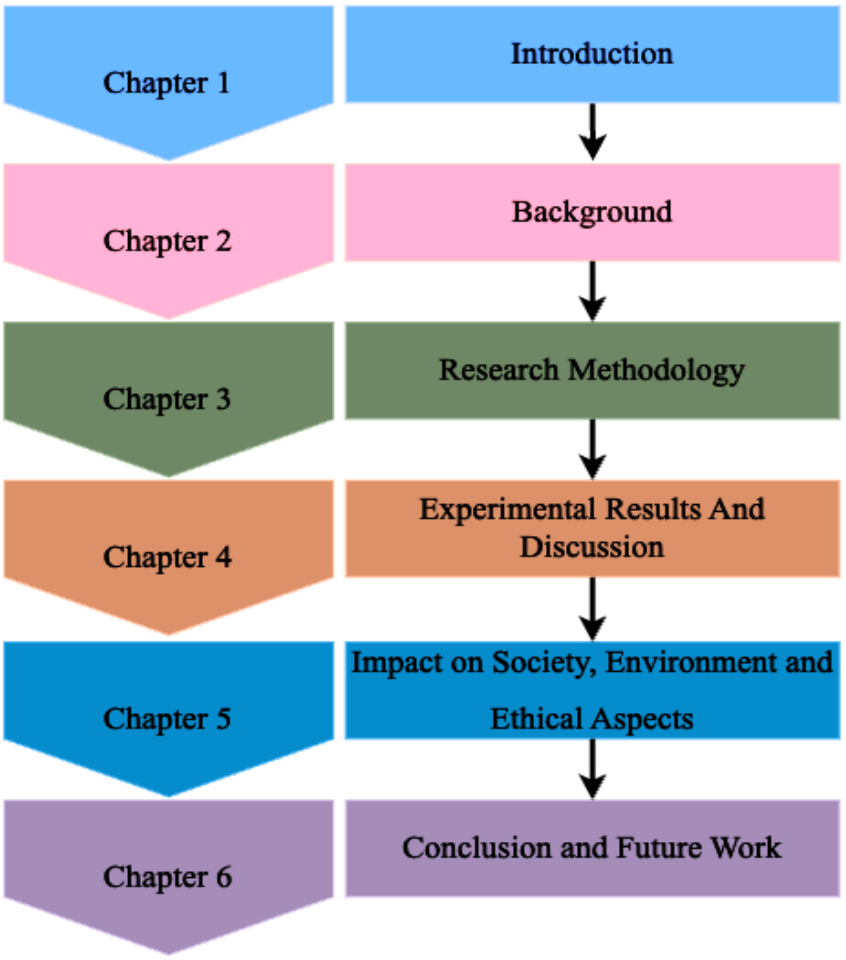


Figure 1.8: Report layout in flow diagram style

Chapter 01 outlines the research's objectives, purposes, and scope. It covers essential research topics and predicts outcomes. The emphasis is on understanding machine learning's strengths and limits in identifying and mitigating DDoS assaults.

Chapter 02 evaluates the existing research and literature on DDoS detection and categorization. This chapter examines existing industry solutions, methodologies, and technologies used to guard against DDoS assaults, laying the groundwork for our research.

Chapter 03 describes the approaches used to collect data, extract features, train models, and evaluate our DDoS detection platform. It also outlines how a companion online application tool for cybersecurity researchers was created, with an emphasis on log file uploads and DDoS detection parsing skills.

Chapter 04 shows the performance results of our DDoS detection algorithms, examining their effectiveness and consequences for cybersecurity. This chapter provides statistical assessments, strengths and shortcomings of our approaches, and demonstrations of how the companion tool helps identify DDoS assaults.

Chapter 05 compares our findings to existing DDoS detection systems, emphasizing the gains made. It stresses our approach's higher detection accuracy, efficiency, and applicability over earlier methodologies.
Chapter 06 examines the practical implications of our findings for a variety of stakeholders, including policymakers. This chapter also looks at future research directions, such as creating adaptive detection systems, integrating the companion tool with larger cybersecurity tools, and improving the tool's scalability and user experience.

The bottom line is, we take a deep dive into this field to conduct more studies and derive strategic recommendations from the study to enhance as much as possible the capacities of DDoS defense, and in general, the digital ecosystem resilience. Our goal is to combine state of the art machine learning models with practical tools to produce a holistic framework for DDoS threat detection and mitigation, in real world cybersecurity operations.

CHAPTER 2
Background

2.1 Overview

The critical nature of network resources and services has made the distributed denial of service attacks, also known as DDoS attacks, a major cybersecurity threat. This attack disrupts ordinary business processes by flooding target systems with an excessive number of requests, possibly causing the loss of valuable services as expressed by Shaar et al. [12]. Secondly, the result may be in the loss of vital services shown by Sachdeva et al. [13]. Due to the constant adaptation of these strategies by cybercriminals and the evolution of technology, the volume and sophistication of distributed denial of service attacks have increased exponentially. Hoque et al. [14] exploit network design flaws and adopt botnets as a service to launch attacks on a large scale, affecting all sorts of platforms, e.g. social networking, e-commerce, and financial institutions. The intent of these attacks is not to maximize strategic level disruption. The traditional defense solutions face serious challenges because the distributed denial of service assaults (DDoS) are elaborate and non-uniform. Volumetric attacks flood network bandwidth, and application-layer vulnerabilities target individual services as emphasized by Navruzov et al. [15]. Since it is constantly evolving, with the traditional systems more often than not almost always failing to meet these fluctuating threats head-on, it becomes important to have such proactive and dynamic cybersecurity measures in place. The result of this is the need for the exploration of new pathways to achieve detection and prevention of distributed denial of service attacks. Machine learning (ML) strategies for enhancing DDoS detection and response are a promising path expressed by Sofi et al. [16]. Machine Learning Machine learning algorithms can analyze vast amounts of network traffic data to identify the patterns typical of distributed denial of service attacks. This results in faster, more accurate and complete detection compared to prior configurations. Conversely, operationalizing machine learning (ML) based threat detection solutions within existing cybersecurity frameworks raises additional challenges to be addressed including scaling adaptation, percent correct detection rates, and the reduction of false positives. To counter these problems, employment of visualization techniques and companion monitoring systems has emerged as an important solution conveyed by Mansmann et al. [17]. These technologies offer immediate visibility into the network, and allow cybersecurity researchers to see anomalous behavior that occurs as part of a distributed denial of service attack earlier in its lifecycle. Visualization tools permit creation of graphical displays of network traffic patterns, the identification of probable threats, and tracing of attack trends over time, all which aid the improvement

of recognition of irregular behaviors and expedited response times by analysts stated by Bamasag et al. [18]. Companion monitoring systems can enhance the efficiency of machine learning-based DDoS detection by furnishing interactive dashboards and detailed analytics.

Due to the aggregation of data from many different sources, CIS and other systems like it can offer the most comprehensive view of network performance and attack tendencies. Companion tools could help analysts understand how effective past mitigation strategies were, or were not, with the use of statistical analysis and visualizations, and enable them to subsequently improve their defense in light of this. Studies have shown the detection and mitigation of distributed denial-of-service attacks (DDoS) can be significantly improved by the use of visualization and companion tools.

In conclusion, DDoS is becoming a bigger threat to network security and requires the use of advanced methods for detection and mitigation that are proactive. Integrating algorithms via machine learning techniques, visualization approaches and companion monitoring systems that provide a cocktail of these facets should offer a good blueprint for hardening the resilience of digital infrastructures against these nascent threats. Due to the dynamic nature of distributed denial of service attacks (DDoS), it is common to implement robust security measures and technologies in order to protect mission-critical systems and ensure the continued operation of network services.

2.2 Related Work

The approaches that have been presented by the researchers in the broad area of DDoS detection and classification are quite novel. Researchers have made many efforts to explore solutions and techniques. Through a detailed literature review spanning a wide variety of methods, ranging from ensemble learning approaches to anomaly-based detection. This body of work underscores the critical need to navigate a challenging and ever-changing cybersecurity landscape, where DDoS attacks remain significant challenges. Researchers have enforced machine learning techniques and Software Defined Networking (SDN) network topologies to improve the defensive measure against these malicious attacks, focusing on enhancing the accurate detection while reducing the false positives.

Hence, in this paper, they propose a universal machine learning (ML)-based model for detection in the varying landscape of distributed denial of service (DDoS) attacks. The model uses light gradient boosting machine (LGBM) algorithm to distinguish benign from malicious flows and combines filter and embedded feature selection

methods by a three-stage Integrated Feature Selection (IFS) process. This results in many measures increasing compared to most recent models which points to not only improved generalization of model but its usability pointed by Marvi et al. [19].

Novel approaches have proliferated in the field of network attack detection and prevention in recent years. In this spectrum the necessity of using vital security equipment such as Hybrid or Anomaly based or signature based Intrusion detection system (IDS) is exposed. Anomaly-based IDS detects the abnormality by comparing actions at the current traffic to the ordinary library of happenings at any stage; signature-based IDS, on the other hand, monitors the event by comparing it to an existing signature database. These tactics indicate that cybersecurity is evolving fields and calls for defensive frameworks stated by Aldweesh et al. [20].

Observed by them in every case, the decision to trigger an alert depends on finding corresponding parallels or spotting differences. Although signature-based intrusion detection systems are known for having low false alarm rates, they struggle to gather and keep up with a vast array of assault variants. As such, creating signatures for each possible attack variant presents a significant obstacle in the field of cybersecurity noted by Masdari et al. [21].

Yungaicela-Naula et al. [22] have employed deep and machine learning models to detect DDoS attacks. They performed a series of experiments with different deep learning and machine learning models on CICDoS2017 and CICDDoS2019 datasets, Gated Recurrent Units (an improvement of Recurrent Neural Networks), Random Forests, and Logistic Regression. Their models outperformed in the simulation network, with an accuracy score of 0.9977 in out-of-sample test data.

The EBF, empowered also with PCA feature selection, is an ensemble classifier that helps researchers identify rouge traffic denoted by Indrasiri et al. in [23] as did testing with their own IoTID20 and UNSW-NB15 datasets and a combination of them and got an accuracy of 0.984 and 0.985.

The rise of microservices, smart device usage, and researchers flour on the DDoS attacks. Manually, the traditional detection method fails for human fault and the strict rule follows. In this research, we have proposed a hybrid web intrusion detection system (HWIDS) to overcome this enthusiasm. It actually is an HTTP-DDoS attack and an unknown one, which are two most widely used and efficient feature sets, utilizing ensemble learning techniques. The results reveal the ensemble model attains 95.96% accuracy and 4.10% false negatives towards known attacks, as well as

93.48% accuracy and 6.52% false negatives towards unknown attacks. This demonstrates its self-detection and accuracy, which means great preemptive security benefits proposed by Potnis et al. [24].

Based on the above mentioned related works we can concise the discussion as of the following Table 2.2,

Table 2.2: Comparison of related works

Authors with Ref	Approach	Techniques	Dataset	Accuracy	Findings
Marvi et al. [19]	ML Model	Light Gradient Boosting Machine (LGBM), IFS	Not specified	High	Universal model for distinguishing benign vs. malicious flows with enhanced generalization and usability.
Aldweesh et al. [20]	Hybrid IDS	Anomaly based and Signature based IDS	Not specified	Not specified	Highlights need for IDS combining both anomaly and signature methods for evolving cybersecurity needs.
Masdari et al. [21]	Signature based IDS	Signature Comparison	Not specified	Not specified	Struggles with maintaining comprehensive and up-to-date signature databases.

Yungaicela-Naula et al. [22]	ML and Deep Learning Models	GRU, Random Forest, Logistic Regression	CICDoS2017 CICDDoS2019	99.77%	Superior performance in simulation networks with high accuracy
Indrasiri et al. [23]	Ensemble Classifier (EBF)	PCA Feature Selection, IoTID20, UNSW-NB15	IoTID20, UNSW-NB15	98.4%, 98.5%	High accuracy in identifying rogue traffic with ensemble classifier and feature selection.
Potnis et al. [24]	Hybrid Web Intrusion Detection System (HWIDS)	Ensemble Learning Techniques	Not specified	95.96% (known), 93.48% (unknown)	Effective in detecting both known and unknown HTTP-DDoS attacks, providing significant preemptive security.

2.3 Scope of the Problem

In this paper we aim to enhance the accuracy and efficiency of detecting and classifying Distributed Denial of Service (DDoS) attacks by integrating machine learning (ML) models via ensemble approach. For this we wish to exploit abilities of various classifiers across a single system. Furthermore, we are also supposed to create a sister website to aid cyber security analysts and the website will come in the packet tracking for the DDoS on log files. Our objective is to find the best possible model helping to enhance the detection performance optimizing both false positives as well as false negatives. This will allow us to design more effective, and more adaptable, security solutions to address the dynamic threat landscape. Our aim is that with the help of ensemble learning, the system log prediction against advanced attacks such as DDoS can be improved, in this way we redefine the periphery of DDoS Detection research and development. Addition of a DDoS companion system in our operations will help us greatly to deal with new threats and in the long run show the system health in the previous occurrences.

2.4 Challenges

Here we will talk about some of the problems that occur when using ensemble methods in machine learning (ML) to detect Distributed Denial of Service (DDoS) attacks. Problems like Data preprocessing, Feature selection, Model selection, Ensemble building, Accuracy thresholding, Detection accuracy improvement are covered under these kinds of problems. These issues are particularly important for the development of accurate and efficient DDoS detection systems because ensemble methods are one of the best types to combine multiple ML models. Similarly, the deployment of the packet wise DDoS Detection through log files for aiding the cyber security analysts is also a challenge that needs to be tackled.

i. **Data Processing:** Given, the dataset is with 431,371 entries it is cumbersome to manage the dataset due to processing speed & limited computing capacities. In order to create an effective ungulate data-processing pipeline, it is essential to develop useful data processing techniques that are going to maximize the intake and cleaning of the training of features from such data while simultaneously limiting the computational overhead.

ii. **Feature Selection:** To reduce the overall model training complexity we need to consider the best features for the process else there comes many issues and time is one of them. We are searching highly correlated features in the correlation matrix to remove the duplicate and multicollinearity features. We generally use a threshold of 0.98. This procedural method ensures a concise and effective set of features, highlighting the significance of feature detection on model improvement.

iii. **Model Selection:** There are a variety of criteria we have to consider, such as detection accuracy, the computational complexity and scalability while choosing the appropriate machine learning models for DDoS attack detection. Moreover, it is necessary to evaluate and compare the multiple models to find out which ones are suitable to the specific dataset and the detection requirements.

iv. **Threshold Setting:** It is the performance and efficiency at implementation that characterizes our model and requires appropriate threshold values for accuracy and other measures of evaluation. Balancing the trade-off between sensitivity and specificity has an important impact on detection rate. Setting threshold values appropriately would increase detection rate by reducing the number of false negatives that would be detected.

v. **Improving Accuracy:** One of the most necessary things to be done in model generation is to have the best accuracy possible. This needs to be done by proper tuning, considering proper aggregating methods like ensemble

techniques and most importantly understanding the proper situation and tuning parameters.

vi. **Resource Optimization:** One of the main focus in such contexts is optimization of memory and compute resources during model training and inference. Model compression, pruning and lightweight design as the average lower resource of assistance without detection performance, may be utilizable.

There are significant challenges to be addressed when it comes to creating the DDoS Detector Companion Tool intended to give predictions on log files and give the best user experience to the cybersecurity experts.

i. **Data Processing:** The very first thing is Dataset Processing. The strong file processing system must be able to remove columns that are repetitive after taking trained models into account while a cybersecurity analyst is putting input and log files. This is an extensive sanitization process to assure accurate identification and analysis while retaining only relevant data.

ii. **Model Running in Backend Server:** It is based on the fact that running the model on backend server is hard. These model files such as pkl or joblib files are large in size because the datasets used to create these models are of great scale. It is absolutely mandatory to think about the size of the model and the entire processing workflow in order to properly manage these mega models during prediction. The result is immediate in order to avoid any performance constraints.

iii. **Data Visualization:** We should never underestimate the importance of the concept that data visualization is vital. So, cybersecurity analysts need to make sure the data and findings are properly oriented and visualize to get systematic help to lookout and understand logs. It can provide visually intuitive and detailed images that allow the user to quickly, efficiently and accurately respond to the logs that have been identified as being obscured.

Addressing these aspects is crucial in order to develop a reliable and useful DDoS Detector Companion Tool. It will enlighten the expertise of cybersecurity experts to identify DDoS attacks.

CHAPTER 3
Research Methodology

3.1 Proposed Methodology

Our proposed methodology is to use the knowledge of Machine Learning and form the best base models we would ensemble them for generating the final model. Here we propose the working methodology to detect and classify DDoS attack with effective manner in addition to this the proposed model uses ensemble learning algorithms with popular machine learning classifiers such as Decision Tree Classifier, Random Forest Classifier, Logistic Regression, KNeighbors Classifier, Gradient Boosting Classifier, AdaBoost Classifier, Bagging Classifier, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Support Vector Classifier, Bernoulli Naive Bayes, Gaussian Naive Bayes, Multi-layer Perceptron Classifier. As illustrated in the Figure 3.1 below, we appropriately employ the overall methodology.

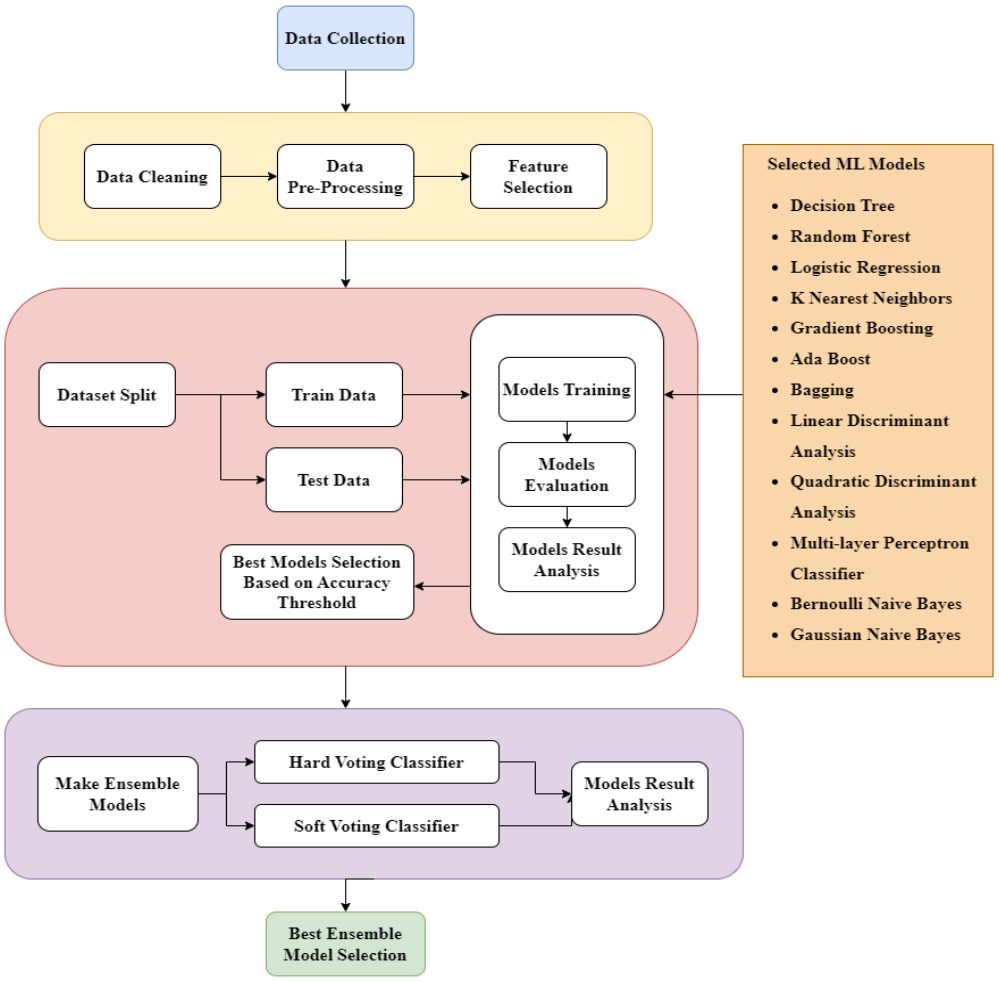


Figure 3.1: An Overview Diagram for DDoS Detection System

So the overall process for model generation is basically broken into the following section as per Figure 3.1 and after the data collection is done we are to clean the data, make some pre-processings and from the feature selection we consider the models and do the train test split and based on the model results we determine the acceptable models in the ensemblement and get the final best model.

3.2 Dataset Collection

The dataset was available at Mendeley Data Website and CIC-DDoS2019 Dataset (March 03, 2023) which was also deposited in Mendeley Data [43]. This dataset contains 80 attributes ('Unnamed: 0', 'Protocol', 'Flow Duration', 'Total Fwd Packets', 'Total Backward Packets', 'Fwd Packets Length Total', 'Bwd Packets Length Total', 'Fwd Packet Length Max', 'Fwd Packet Length Min', 'Fwd Packet Length Mean', 'Fwd Packet Length Std', 'Bwd Packet Length Max', 'Bwd Packet Length Min', 'Bwd Packet Length Mean', 'Bwd Packet Length Std', 'Flow Bytes/s', 'Flow Packets/s', 'Flow IAT Mean', 'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Total', 'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Max', 'Fwd IAT Min', 'Bwd IAT Total', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max', 'Bwd IAT Min', 'Fwd PSH Flags', 'Bwd PSH Flags', 'Fwd URG Flags', 'Bwd URG Flags', 'Fwd Header Length', 'Bwd Header Length', 'Fwd Packets/s', 'Bwd Packets/s', 'Packet Length Min', 'Packet Length Max', 'Packet Length Mean', 'Packet Length Std', 'Packet Length Variance', 'FIN Flag Count', 'SYN Flag Count', 'RST Flag Count', 'PSH Flag Count', 'ACK Flag Count', 'URG Flag Count', 'CWE Flag Count', 'ECE Flag Count', 'Down/Up Ratio', 'Avg Packet Size', 'Avg Fwd Segment Size', 'Avg Bwd Segment Size', 'Fwd Avg Bytes/Bulk', 'Fwd Avg Packets/Bulk', 'Fwd Avg Bulk Rate', 'Bwd Avg Bytes/Bulk', 'Bwd Avg Packets/Bulk', 'Bwd Avg Bulk Rate', 'Subflow Fwd Packets', 'Subflow Fwd Bytes', 'Subflow Bwd Packets', 'Subflow Bwd Bytes', 'Init Fwd Win Bytes', 'Init Bwd Win Bytes', 'Fwd Act Data Packets', 'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max', 'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min', 'Label', 'Class'). Through these features and target sets we can see a lot of columns contains numerical data where an independent column 'Protocol' contains categorical data. The ‘Label’ and ‘Class’ can be treated as the target columns for the section and rest who have categorical data as values points on the columns in the independent variables section. In fact, a lot of our packet information is already contained within this data and this is immensely helpful when we come to detect DDoS with such language models.

3.3 Dataset Description

The Canadian Institute for Cybersecurity (CIC) [44] has created the CIC-DDoS2019 dataset [45] for DDoS attack research, which includes a comprehensive collection of

network traffic data with a set of background traffic, as well as protocol flow, and time series information. By aggregating 431,371 data points, the dataset has been transformed into 80 feature columns which contain various characteristics of network traffic and can be used to investigate new cybersecurity risks, compare detection algorithms, and create and train intrusion detection systems.
Here one of the target columns is ‘Label’ and we can see the unique DDoS Attacks and their counts in the following Table 3.3.1,

Table 3.3.1: Label Target Column Unique Data Information

Label	**Count**
DrDoS_NTP	121368
TFTP	98917
Benign	97831
Syn	49373
UDP	18090
DrDoS_UDP	10420
UDP-lag	8872
MSSQL	8523
DrDoS_MSSQL	6212
DrDoS_DNS	3669
DrDoS_SNMP	2717
LDAP	1906
DrDoS_LDAP	1440
Portmap	685
NetBIOS	644
DrDoS_NetBIOS	598
UDPLag	55
WebDDoS	51

Total	**431371**

From the official page [42] we can also see the classification of the DDoS Attacks branched out as follows Figure 3.3 is just a better presentation on top of that.

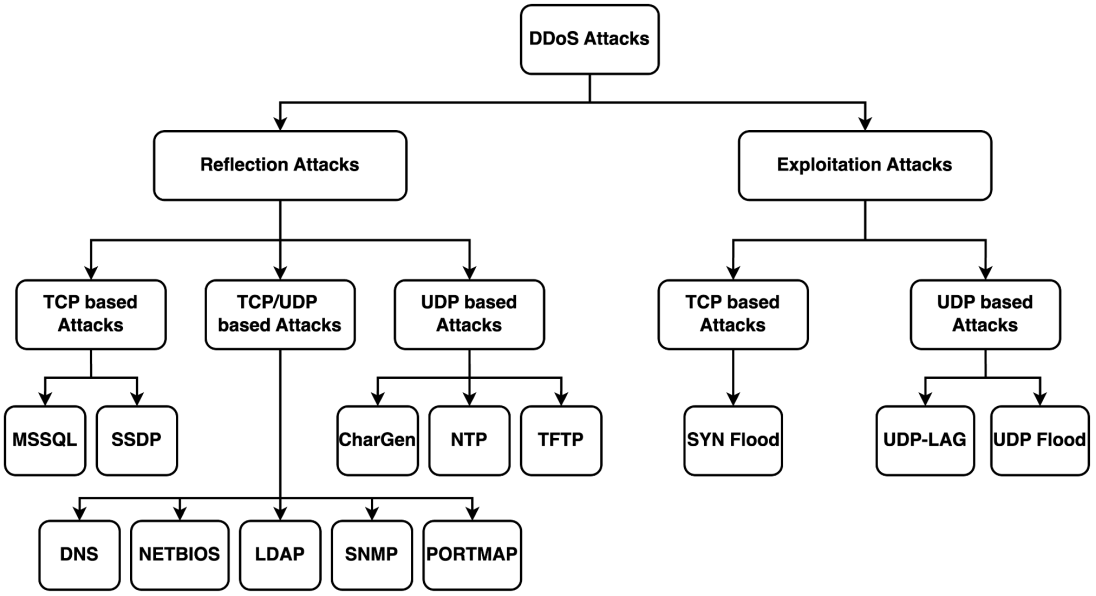


Figure 3.3: DDoS attacks classification based on [42]

The of the target columns is ‘Class’ and we can see the in-general DDoS Attacks and their counts in the following Table 3.3.2,

Table 3.3.2: Class Target Column Unique Data Information

Label	**Count**
Attack	336040
Benign	97831
Total	**431371**

3.4 Data Cleaning

We begin the data preparation by an initial step of cleaning the data rigorously to ensure that the dataset is of good quality and integrity which is then carried forward to the next stage. This step completes all the following important tasks that careful and necessary to do before we further analyze the data:

i. **Removing NA and Null Rows:** This means that the dataset is carefully examined for null and NA values, which are then removed. This mandatory

stage, erasing any possible error sources, ensures dataset purity and completeness across a model analysis and development scenario. Even though the dataset was nearly clean so no row got lost.

ii. **Removing Columns with Constant Values:** It inspects dataset in detail to find the columns that have the same value in every entry. By filtering out these unnecessary columns, we simplify the dataset, which reduces complexity and processing costs and ensures that we retain only the data required for analysis. We will obtain 12 columns after executing this, whose all values are constant across the entire dataset. They were (Fwd URG Flags, Bwd URG Flags, FIN Flag Count, PSH Flag Count, ECE Flag Count, Fwd Avg Bytes/Bulk, Fwd Avg Packets/Bulk, Fwd Avg Bulk Rate, Bwd Avg Bytes/Bulk, Bwd Avg Packets).

In this article, we applied strict data cleaning approaches to prepare the data set for further analysis. Hence these precautions are critical to ensure data credibility, relevance and sourcing for the robust and reliable development of Machine Learning based DDoS detection and mitigation tools.

3.5 Data Pre-Processing

Data preparation is one of the most critical parts of any machine learning pipeline that ensures the dataset is properly formatted and pre-processed before fitting the model. In the present work, we performed a data cleaning process in a meticulous way to enhance dataset quality and to make the data applicable for further analysis. The dataset was preprocessed, which included:

i. **Encoding Target Columns:** Since we have these target columns, "Label" and "Class" we have to encode them so that the training of the model could be easier. Where, Column 'Class' is differentiating between 'Attack' and 'Benign' traffic and Column 'Label' is indicating attack type('UDP', 'MSSQL', 'Benign', 'Portmap', 'Syn', 'NetBIOS', 'UDPLag', 'LDAP', 'DrDoS_DNS', 'UDP-lag', 'WebDDoS', 'TFTP', 'DrDoS_UDP', 'DrDoS_SNMP', 'DrDoS_NetBIOS', 'DrDoS_LDAP', 'DrDoS_MSSQL', 'DrDoS_NTP'). If these columns are encoded, then the target variables will be in a numerical format instead of a human readable feature which is most suitable for machine learning algorithms.

ii. **Dropping Irrelevant Columns:** We removed columns that were irrelevant for an analysis e.g 'Unnamed: 0' from the dataset. It removes any unnecessary columns in the dataset making it more streamlined and also reduces the computational cost during model training.

iii. **Instantaneous Encoding Category characteristics:** The one hot encoding was performed in the categorical features "Protocol". This makes binary columns as each category. One hot encoding is particularly useful in situations where no such ordinal relationship between categories can be assumed by the model and that we should be careful when doing it in order to avoid introducing multicollinearity.

iv. **Storing Encoding Information:** We kept encoding information like exact protocol mappings as well of each encoded feature so that things were done in a clear way and were easy to refer to later. We use this data, it actually connects the dots between the original merge values we have and the relevant one-hot encoded columns, which inturn gives us a kind of reference guide for us to follow and understand and interpret our model predictions. Also since we will be doing more tests so we should create a uniform information base and that information has to be shared to every model, as can be seen in Figure 3.5.

```
# for Label Feature
specificAttackEncoding = {
  'UDP': 14, 'MSSQL': 9, 'Benign': 0, 'Portmap': 11,
  'Syn': 12, 'NetBIOS': 10, 'UDPLag': 16, 'LDAP': 8,
  'DrDoS_DNS': 1, 'UDP-lag': 15, 'WebDDoS': 17, 'TFTP': 13,
  'DrDoS_UDP': 7, 'DrDoS_SNMP': 6, 'DrDoS_NetBIOS': 5,
  'DrDoS_LDAP': 2, 'DrDoS_MSSQL': 3, 'DrDoS_NTP': 4}

# for Class Feature
genarelAttackEncoding = {'Attack': 0, 'Benign': 1}
```

Figure 3.5: Target Columns Encoded Results

We then prepare this data by arranging it properly in an optimized format that makes it ready to feed to machine learning algorithms. In doing so, we prepare the ground for a solid and efficient model training process, whereby the precision and stability of our DDoS attack detection system over time will be increased.

3.6 Feature Selection

Feature selection was then conducted in all cases to (i) reduce the feature space keeping in mind computational complexity and (ii) maximize model performance. Taking the condition on the threshold point of 98% our correlation matrix result allowed us to drop 15 more columns. These are ('Flow Duration', 'Total Fwd Packets', 'Total Backward Packets', 'Fwd Packet Length Total', 'Bwd Packet Length Total', 'Fwd Packet Length Min', 'Fwd Packet Length Mean', 'Bwd Packet Length Mean', 'Flow Packets/s', 'Flow IAT Std', 'Flow IAT Max', 'Flow IAT Min', 'Fwd IAT Max', 'Fwd PSH Flags', 'Subflow Fwd Packets'). So taking out the ones that do not help we have the top 50 of the order gave us :

('Fwd Packet Length Max', 'Fwd Packet Length Std', 'Bwd Packet Length Max', 'Bwd Packet Length Min', 'Bwd Packet Length Std', 'Flow Bytes/s', 'Flow IAT Mean', 'Fwd IAT Total', 'Fwd IAT Mean', 'Fwd IAT Std', 'Fwd IAT Min', 'Bwd IAT Total', 'Bwd IAT Mean', 'Bwd IAT Std', 'Bwd IAT Max', 'Bwd IAT Min', 'Fwd Header Length', 'Bwd Header Length', 'Fwd Packets/s', 'Bwd Packets/s', 'Packet Length Min', 'Packet Length Max', 'Packet Length Mean', 'Packet Length Std', 'Packet Length Variance', 'SYN Flag Count', 'RST Flag Count', 'ACK Flag Count', 'URG Flag Count', 'CWE Flag Count', 'Down/Up Ratio', 'Avg Packet Size', 'Avg Fwd Segment Size', 'Avg Bwd Segment Size', 'Subflow Fwd Bytes', 'Subflow Bwd Packets', 'Subflow Bwd Bytes', 'Init Fwd Win Bytes', 'Init Bwd Win Bytes', 'Fwd Act Data Packets', 'Fwd Seg Size Min', 'Active Mean', 'Active Std', 'Active Max', 'Active Min', 'Idle Mean', 'Idle Std', 'Idle Max', 'Idle Min', 'Protocol')

3.7 Training Data and Testing Data

The original dataset was carefully broken down into training and test sets to prevent the model from being able to overfit the data and give a fair evaluation of the dataset. The remaining fell into the performance allocation, leaving enough data for the model to learn from in addition to having 80% training and 20% test data. Separated in this manner intentionally, to maintain the model generalization capability referred to as how the model could do well when seeing real data, and reflecting some truth on their predictive performance in the real world.

3.8 Target Columns

We get to work directly on each of the two target columns. The "Label" column is one that we will consider when we need our model to provide us with the DDoS attack type that took place with very high confidence. There were 18 classes being inducted

at max in the dataset with 17 from DDoS and 01 being the Benign one. Distribution for this column is presented in Figure 3.8.1, respectively with count.

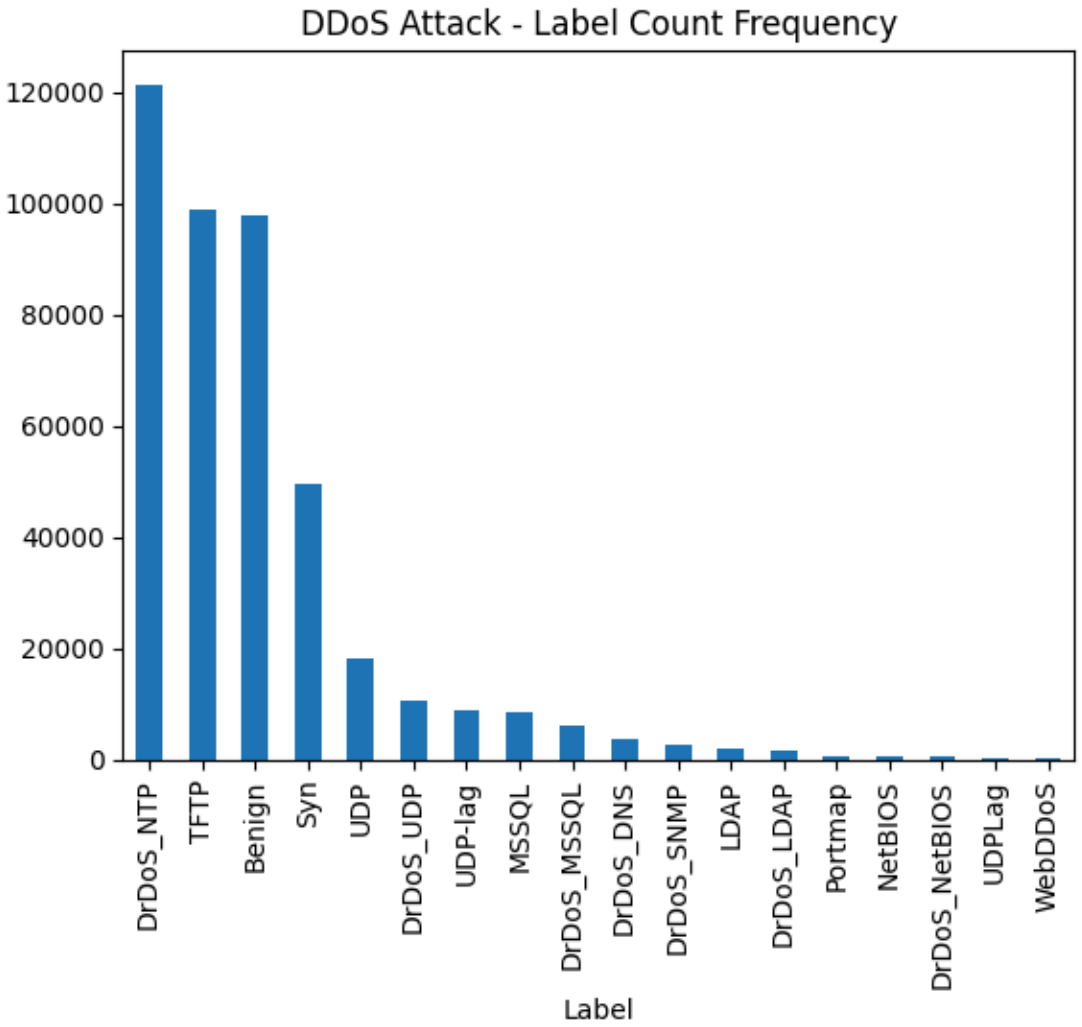


Figure 3.8.1: Count plot of all Label Columns

But we were really keen to improve the accuracy score for the final models therefore we kept a fine learning curve where some labels may have a very less occurrence, in this case the prediction will no longer be as accurate so we decided to consider only the top 13 labels where 12 of them are types of DDoS attacks and the rest one is the benign. We will then just focus on them and the count of their occurrence are represented as following Figure 3.8.2,

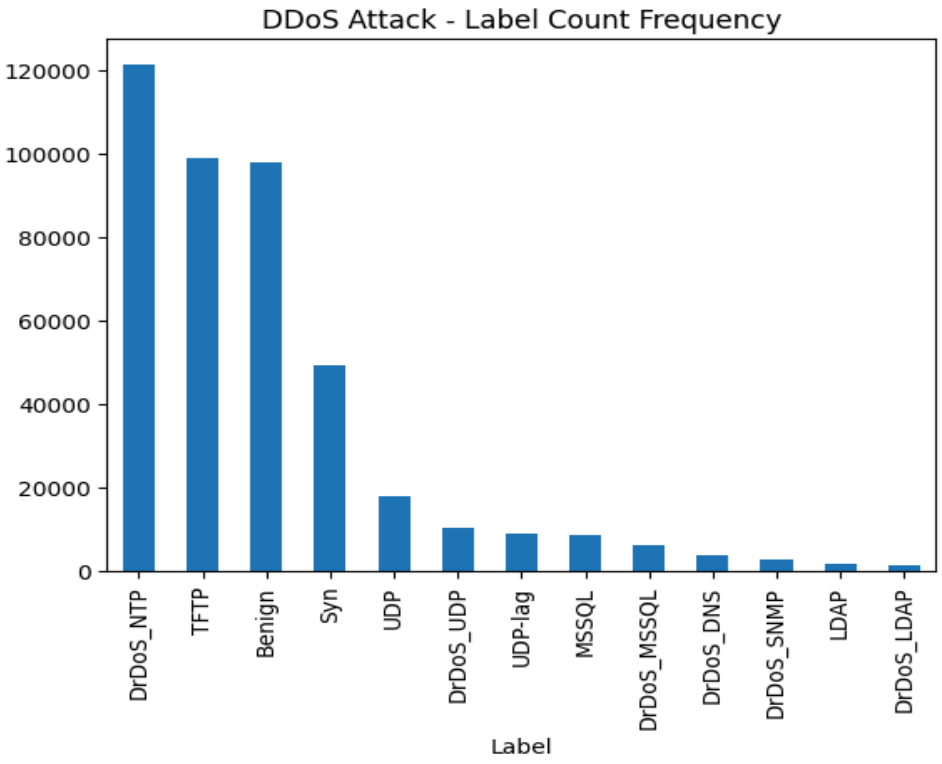


Figure 3.8.2: Count plot of Selected Label Columns

And since Figure 3.8.3 shows the distribution of the 'Class' column, we can readily evaluate it when we are merely interested in understanding if the packet information is DDoS or benign.

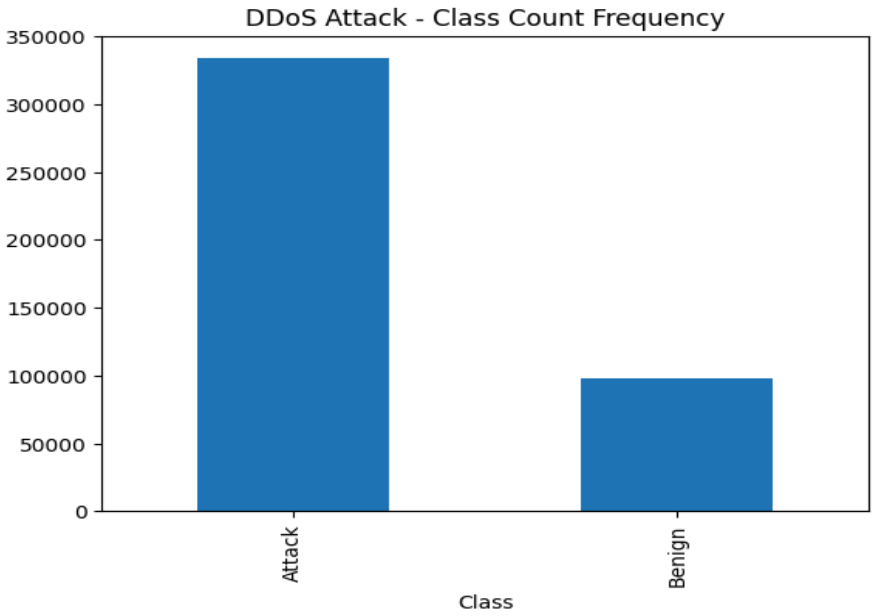


Figure 3.8.3: Count plot of Class Column

3.9 Machine Learning Algorithms Used

In this work, we performed a comprehensive analysis based on different ML classifiers to detect DDoS attacks at an early-stage reliably. We have considered the following ML models,

i. Random Forest Classifier
ii. Decision Tree Classifier
iii. Support Vector Classifier
iv. Logistic Regression
v. KNeighbors Classifier
vi. Gradient Boosting Classifier
vii. AdaBoost Classifier
viii. Bagging Classifier
ix. Linear Discriminant Analysis
x. Quadratic Discriminant Analysis
xi. Multi-layer Perceptron Classifier
xii. Bernoulli Naive Bayes
xiii. Gaussian Naive Bayes

Being the best classification models we had considered these in the first place. We did this by using data visualization techniques which allowed us to recognize the algorithms that could be used to achieve our objective of creating categories. They used test data to evaluate the model and train data to shape the data to find the patterns. The comparison of the forecasts against the ground truth was done during the DDoS detection performance evaluation to provide a robust assessment of the efficiency of the algorithm.

i. **Random Forest Classifier (RF):** RF builds a lot of decision trees during training and it stores the class mode as the prediction of each tree individually. RF is an ensemble learning technique. Given that RF can deal with high-dimensional data, and capture complex feature interactions, it is a good choice for our DDoS detection technology. RF is favored as it has the ability to provide accurate predictions in the presence of noisy data and is resistant to overfitting.
ii. **Decision Tree Classifier (DT):** Supervised learning algorithm DecisionTreeClassifier (DT) is a kind of tree-based model, and it is intended for classification and the feature space is divided into regions depending on the feature values. First of all, it needs to be clear that DT is a good candidate in our DDoS detection framework because of its transparent model interpretation and straightforward decision-making. Decision tree (DT) since DT can deal with both numeric and categorical data, which are required for

analyzing different types of network traffic data and hence will be used for this purpose.

iii. **Support Vector Classifier (SVC):** Support Vector Classifier (SVC) is a popular supervised learning algorithm which is primarily used for classification of instances of two distinct classes separated by a hyper-plane. As SVC can handle the highly complex decision boundaries and also works well on high-dimensional data, it is a very useful tool in the environment of DDoS detection. SVC is used as it is able to find and classify niche relationships between network traffic features.

iv. **Logistic Regression (LR):** Logistic Regression is a kind of linear classification algorithm used to find the probability of a dependent variable to one class among the two classes. Given LR possesses remarkable simplicity and interpretability, it becomes a favorable candidate for binary classification tasks within our DDoS detection framework. Furthermore, LR is quite effective in processing large-scale datasets, and with the ability of LR to produce probabilistic predictions that help in the risk assessment, and decision-making, it becomes a good candidate.

v. **KNeighbors Classifier (KNN):** It assigns any instance, the majority class in k-nearest neighbors, the label which is chosen. KNN is a practical choice for handling noisy input as well as non-linear decision limits due to their simplicity and flexibility for DDoS detection. KNN is selected to detect regional trends in the network traffic data for its ease to use while providing optimized results.

vi. **Gradient Boosting Classifier (GBC):** An iteratively built way to combine weak learners is this boosting ensemble technique which creates a series of weak learners, each fixing the mistakes of the one before it. One of the use cases of GBC in our DDoS detection system as explained earlier is that it combines many weak models into a single strong classifier. We introduce GBC to cope with overfitting and heterogeneity of features to learn subtle patterns in network traffic data.

vii. **AdaBoost Classifier (ABC)**: Firstly, the AdaBoost Classifier is an ensemble method that indeed combines many weak classifiers by giving greater weight to instances that are misclassified to produce a strong classifier. ABC is a promising candidate for DDoS detection due to its dynamic reduction of the influence of training samples and focusing on the hard samples. We chose ABC due to its ability to handle unbalanced datasets, as well as its good generalization properties on noisy data crucial aspects of DDoS attack detection.

viii. **Bagging Classifier (BC):** BaggingClassifier is an ensemble method which fits N base estimators on momo bootstrapped datasets and combines it using voting or averaging. BC can be an integral part of our DDoS detection methodology as it might decrease variance and make the model more robust by combining multiple base classifiers. It includes BC because it is robust overfitting and can handle many-numbered datasets which are necessary for prediction in order to be accurate and reliable.

ix. **Linear Discriminant Analysis (LDA):** It is used to choose linear feature combinations of a dataset that identifies the separate classes the best. LDA is an important tool for feature extraction and classification in our DDoS detection architecture as it can maximize the separation between classes but minimize the within class scatter. We choose LDA due to the interpretability and its ability to handle multicollinear features, which can help us to find out those most significant features related to DDoS attacks.

x. **Quadratic Discriminant Analysis (QDA):** Like LDA, Quadratic Discriminant Analysis (QDA) also assumes a Gaussian distribution, but unlike LDA it also assumes that each class has its own individual variance. As QDA can be used to represent complex relationships between features and classes it is well suited to capture nonlinear decision boundaries in the scope of DDoS detection. QDA is included due to its capability of handling non-Gaussian datasets and ability to identify fine-grained patterns in network traffic data which is required for accurate DDoS attack detection.

xi. **Multi-layer Perceptron Classifiers (MLPs):** MLP can learn non-linear relationships between input and output between classes. They are composed of many layers of connected nodes. Since MLP can model arbitrary complex dependencies of high-dimensional data it has a fitting location for unifying huge numbers of features of network traffic to detect DDoS as in our detection method. To maximize the accuracy of detection in dynamic network environments, the MLP is adopted, as the MLP is capable of handling large-scale datasets and enjoys flexibility in architecture design.

xii. **Bernoulli Naive Bayes (BNB):** It is a type of probabilistic classifier which is used for binary and discrete data. It is good especially in situations where the presence or absence of one property is essential, distinguished as with text classification or attributes on network packets, as it represents features as binary indicators. BNB is distinguished from its predecessors by finding patterns in binary feature vectors, as it is usually used to describe connection states or the packet flags in network packets, for instance in DDoS detection. The problem is that it is too much overhead and requires additional computation, which makes it unsuitable for real-time applications. This ability

allows BNB to detect even the minimal capacity of suspicious network traffic that can be detected as an early warning of DDoS attacks.

xiii. **Gaussian Naive Bayes (GNB):** GNB performs DDoS detection by simulating the probability distributions of these continuous features, from which the benign and malevolent traffic patterns could be discriminated against. Due to its stochastic nature, it can handle incoming data in dynamic network situations, which makes it very interpretable and easy to study. In addition, the elegance and scalability of GNB make it practical for tapping into large data, as in finding the right balance between detection speed and accuracy compared to potential DDoS threats.

These models are selected because of their various advantages and skills in addressing many facets of the DDoS detection issue, such as feature representation, dimensionality reduction, classification and prediction. Our DDoS detection system seeks to deliver robust and dependable performance across a range of network infrastructures and attack situations by utilizing the complimentary capabilities of these models.

3.10 Ensemble Technique

Ensemble methodologies, that combine several individual models to create a single prediction model that often performs better than any single component model on its own, are powerful machine learning tools. Ensemble approaches have several benefits for DDoS detection, making the detection more robust and performant. We researched voting classifiers as an ensemble of the selected models, using both Hard and Soft Voting Classifiers. Some of the key features for introducing ensemble techniques are as follows,

i. **Diverse Model Aggregation:** Ensemble methods combine multiple machine learning models with different strengths and weaknesses. Ensemble uses the wisdom of the ensemble of models like Random Forest, Support Vector Machine, or Gradient Boosting Classifier to give better, detailed, and more accurate detection outputs.

ii. **Reduction of Model Bias and Variance:** Ensemble methods are used to combine predictions of multiple different models which decreases the chance of both overfitting and underfitting. Ensemble methods give more stable and reliable predictions since they average the predictions of base learners that ideally have their individual biases, which reduces variance particularly in cases where one lacks an adequate amount of training data or the input attributes are just noisy.

iii. **Robustness to Model Uncertainty:** The key benefit of ensemble methods is their increased robustness and generalization mostly when attack patterns and group composition are constantly changing. Ensemble approaches might generalize well across different attack scenarios and sustain detection accuracy in the presence of emerging threats or adversary shaping by leveraging the collective knowledge of a variety of models.

iv. **Enhanced Sensitivity and Specificity of Detection:** The other half consists of complex significant patterns, even for skilled machine learning approaches and complex interrelationships which make accurate detection of those associations highly sensitive and specific, hence, employing ensemble approaches to achieve even higher precision detection is highly effective. An ensemble approach achieves a higher accuracy in widely recognizing known and unknown DDoS attack patterns and effectively decreasing false positive rates by combining the strengths of individual models. This decreases the risk of alert fatigue and decreases the rate at which RIOT can misidentify friendly behavior as enemy threats.

v. **Enhanced Interpretability of the Model:** Although ensemble models are complicated, they can offer important insights into the core features of DDoS attacks and the detection and classification analogy. Cybersecurity researchers may better analyze network risks by using ensemble approaches, which provide interpretable signs of harmful behavior using techniques like feature significance analysis and model visualization.

Classification after capturing network behaviors, the first step in detecting network anomalies and potential threats is the classification process in DDoS detection. In this phase, all incoming network traffic is divided into two classes; legitimate and malicious. Using labeled machine learning techniques, classification will analyze various aspects of network data and recognize any trend which may indicate an on-going attack. Over time, these algorithms are able to learn from labeled samples to differentiate typical and anomalous behavior, which provides the basis for subsequent processes to make detection decisions for threats.

In addition, voting classifiers act as an important method of aggregation that is vital in the construction of ensemble learning for DDoS detection. In machine learning models, hard voting is used to combine the predictions made by individual models to predict the most frequent class label. This technique uses strengths in multiple constituent models to evaluate malicious behavior holistically. If the confidence of the prediction is taken into account in soft voting, hard voting should be used all the time unless you are working with different but also complementary models. With hard

voting we force the detection system to provide strong detection capabilities by starting with simplicity and efficacy. This effectively reduces distortion and errors, and improves the reliability and accuracy of DDoS detectors.

Soft voting is ensemble learning which gives a weight to the models predictions based on the probability estimates of each model for every class. Soft voting, as it sounds, instead of picking a class label, it computes the average of predicted probabilities of the models. By examining the confidence levels of individual predictions, this approach allows the ensemble to take into account how sure each model is in its output. For DDoS detection, soft voting can be used to provide a better evaluation of possible dangers considering the probability of each class. Through doing so, adjusting the strengths of each model in estimating probabilities, often quite effective when the posterior models have drastically different levels of confidence or calibration.

3.11 Model Generation Process

The model building process is an orderly project to ensure the DDoS detection models produced are dependable to differentiate malicious network traffic from benign network traffic. This process contains several vital stages, all or the majority of which contribute to the efficiency of the detection system in general. This is followed by a careful cleaning and preparation step to make sure the dataset is clean and relevant. Ridding of duplicate or null rows and columns with constant values which cannot be providing any new information to detect is necessary. After the data is clean then selecting features is important. We try settings to get the maximum out of Feature selection that require score column requirements, different feature sets, different dataset size. How to achieve that by selecting the salient features which enhance DDoS detection efficiency is one of them.

Afterwards, the data set is split into two different data sets, one which the model will evaluate as it performs on unseen data and learns on another data set called Training data. Stratified sampling can be used in combination with a variety of partitioning strategies to ensure that classes of the data set are distributed across the train and test partitions. DDoS detection effectiveness evaluation of different machine learning classifiers like, RandomForest, DecisionTree, SVC, MLP and many more. Following this, each model is tested thoroughly and various performance metrics such as accuracy, precision, recall, F1-score are measured to determine how well it performs. There are lots of visualization methods like confusion matrices, model score tables, dataset size summaries etc. that are used to show the model evaluation results. These

graphs serve in informing decision-making by showcasing the differentiable pros and cons of different models.

Ensemble modeling comprises selecting the models that meet predefined accuracy thresholds and pooling the predictive power of multiple base classifiers. Ensemble methods increase the accuracy and reliability of the detection by combining the predictions of multiple models through hard voting and soft voting.

Finally, ensemble classifiers are formed by the models that are selected, with the combination of them created in elaboration trained with the models of the training set saved to be used in practice. This ensures that both computational efficiency and model accuracy can be achieved in a way that guarantees that the created DDoS detection system is reliable and can be scaled well. This methodical method of generating models builds effective DDoS detection models that can greatly decrease network security threats, protect against malicious assaults, help in prediction in log files and many more.

3.12 Companion Tool

For the companion tool we are considering a website that is basically meant for the cyber security analyst as this will be used for DDoS detection and classification on the packets' information. Basically the cyber security analyst needs to upload the log file in CSV format and the website will get the results based on the best model.

Tech Stacks of Companion Tools

For this companion website we had considered the following tech stacks,

i. **Frontend:** We have used React [32], a popular JavaScript Library which helps the total website to be built in component structure and is quite fast as well. This made the entire code structure very simple and basic html was done though jsx components. For the styling we have considered a popular styling library named, Tailwind CSS [33]. This has a very human language like styling syntax which makes the styling a breeze. For the components we have considered a UI library named, ShadCN [34]. The components look quite attractive and professional as well and the over aesthetic and usability helped to build the overall site in no time. So considering the Frontend of the website we entirely are working on the JavaScript [35] platform. This Frontend mainly does the talking with the Backend API and based on the overall request and response the cyber security analyst is assisted.

ii. **Backend:** As we are to consider making predictions with the final proposed model then we need to use such a backend that supports the concept of

integrating machine learning models with APIs. So this is the reason we chose Python [36]. And for the API section we used a popular python framework named, FastAPI [37]. This made the integration of machine learning models and API connection seamless and very convenient for the project. So when there is any request of Log Files in CSV format with proper data then the API predicts the result with the model and responds to the results which the Frontend then shows to the cyber security analyst.

iii. **Hosting Platforms:** As we are to make this system accessible to the cyber security analysts and other cyber security teams we need to make the website publicly available. In order to make fully hosted platforms the frontend and the backend must be hosted online. Now as per the hosting solutions we have considered two different platforms for the two parts as of their flexibility and use case. It is also needed to be mentioned that all of these platforms provide free tiers which we are using. The frontend React site is hosted in Vercel [38] where it is very easy to connect a Github [39] Repository and the updated codes or commits will start a new instance which will be reflected in Vercel which will have the updated codes reflected in the website. The backend Python FastAPI is hosted in Render [40] where the seamless integration of python projects especially backends can be done in no time. We have another Github Repository that is connected in the same manner which considers the latest codebase from the main branches repo and pushes that to the production server.

iv. **Codebase Storing Platforms:** Considering the publicly hosted DDoS Detection and Classification service we had to keep our codebase into a safe, secure and public place where the other systems can get the codebase all the time. For this we have used another online service named Github [39]. This is the place where we keep our code for public purposes.

CHAPTER 4
Experimental Results And Discussion

4.1 Experimental Setup

In this study, experimentations were done on the CIC-DDoS2019 dataset which was carefully constructed for DDoS attack detection and mitigation purposes. The data set consists of more than 431,371 flow samples and serves as a comprehensive network traffic dataset for designing and evaluating DDoS detection algorithms Benign network traffic and attack techniques such as TFTP, Syn, and UDP are included in the dataset, making it possible to learn the difference between malicious traffic and lawful traffic. The dataset underwent an extensive pre-processing step to remove unnecessary columns, missing values to improve the efficacy and the generalizability of the model, but at the same time making sure we do not hurt the continuity and reliability of the data. The data is split into the training and test datasets for proper model training and evaluation. A subset of this handpicked dataset has some significance in studying cyber-threat countermeasures and becomes the startpoint for designing DDoS detection systems.

4.2 Experimental Results and Analysis

This section provides a thorough breakdown of the performance indicators used to assess our DDoS detection algorithms. We prioritize four critical metrics: precision, recall, F1-score, and accuracy. Each indicator provides unique insights into the model's ability to discriminate between benign and malicious traffic. The metrics described in the following subsections are relevant in measuring the model's performance especially the accuracy as our main problem is a type of classification problem.

Precision:

One statistic used to assess a classification model's accuracy is precision. It calculates the percentage of all instances that the model categorized as positive like, true positives or false positives that were correctly recognized as positive cases, or true positives. Precision is a statistic used in machine learning to evaluate the effectiveness of a classification model, particularly in binary classification problems. The accuracy of the model is determined by the precision of its positive predictions. It is defined as the ratio of true positive predictions to the sum of true positive and false positive predictions. Precision is defined as the number of TP on the number of TP "+" number of FP. When a model that is actually negative is wrongly categorized as positive, this is known as a false positive. In mathematical terms, accuracy is determined using the following formula:

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

Recall:

Recall from a classification report is the number of true positives divided by the number of true positives plus the number of false negatives. This is the sensitivity or the true positive rate. This is the model's resistance to false negatives, or cases that have been falsely labeled as negative when they are actually positive. Recall simply refers to the number of correct positive true positive predictions made by the model to the number of actual positive true positives and false negatives. In other words, this is a measure of the model's ability to find all relevant observations in a dataset. The recall is defined as the number of genuine TP divided by the TP "+" FN. This is one of the definitions of the formula for recall;

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

F1-Score:

The F1 score is a measure that allows for the trade-off between precision and recall by calculating the harmonic mean of both. Handy for unbalanced datasets, in which the distribution of positive and negative to the dataset changes greatly. In machine learning F1 score is a widely used performance metric to compute the performance of a classification model. This is especially helpful in unbalanced datasets, where you have no samples of a class in contrast to others. F1 Score is a harmonic mean of precision & recall. Recall and Accuracy are two factors that highlight classification ability but focus on the different parts of that talent. F1 score should be used when you desire to seek a balance between Precision and Recall and there is an uneven class distribution. One definition for the recall formula is:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Accuracy:

Accuracy is one of the most common machine learning measures to define the performance of classification models. It is a binary classification evaluation metric that gives a general idea of the model performance. It is the number of correct predictions made from the dataset. The formula for accuracy is:

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

4.3 Outcome

In our experiment we considered the entire dataset, splitted into eighty-twenty ratio for train-test split, took the top fifty columns as the independent features and for the profound knowledge considered both “Label” and “Class” as the target features.

All thirteen base machine learning models ran for both the target features separately, and we selected the top models based on an 85% threshold value, as determined by the accuracy measure. Next we considered these best base models and ensemble them with both the Hard and Soft voting classifiers. So after the ensemble models we will get four total models where two variance is coming from Hard or Soft voting and two variance is coming from Label or Class target feature consideration.

After all of these we want to select the best one for “Label” feature and another best one for the “Class” feature so eventually we have one best model for DDoS Detection and another best model for DDoS Classification.

Machine Learning Models Results based on Accuracy

As it has been clarified already that we are having some variations to get the most accurate ensemble models so for that we have considered two target features and two methods of ensemblement. Firstly when we are concerned about DDoS Detection then we are targeting the “Class” feature and when it is for DDoS Classification we are on the “Label” column. Another aspect is how we ensemble the best models and that is using two methods like, Hard Voting and Soft Voting Classifiers for the ensemble technique.

In the following sections we are going to see the training accuracy curve and validation accuracy curve stacked against each other for each of the 13 models for both DDoS detection and classification analogy.

Random Forest Classifier

In the following Figure 4.3.1 we see the training and validation accuracy curve for DDoS Classification for RF

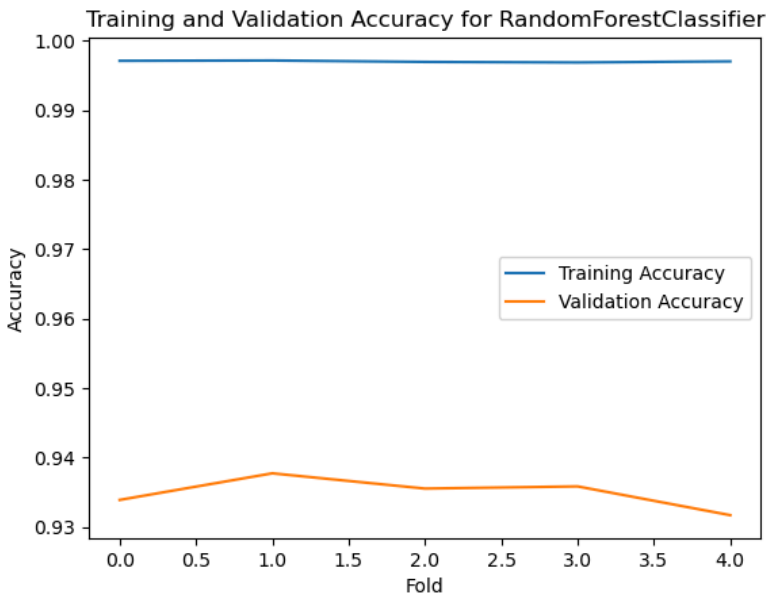


Figure 4.3.1: Training and Validation Accuracy Curves for Random Forest Classifier for DDoS Classification

In the following Figure 4.3.2 we see the training and validation accuracy curve for DDoS Detection for RF

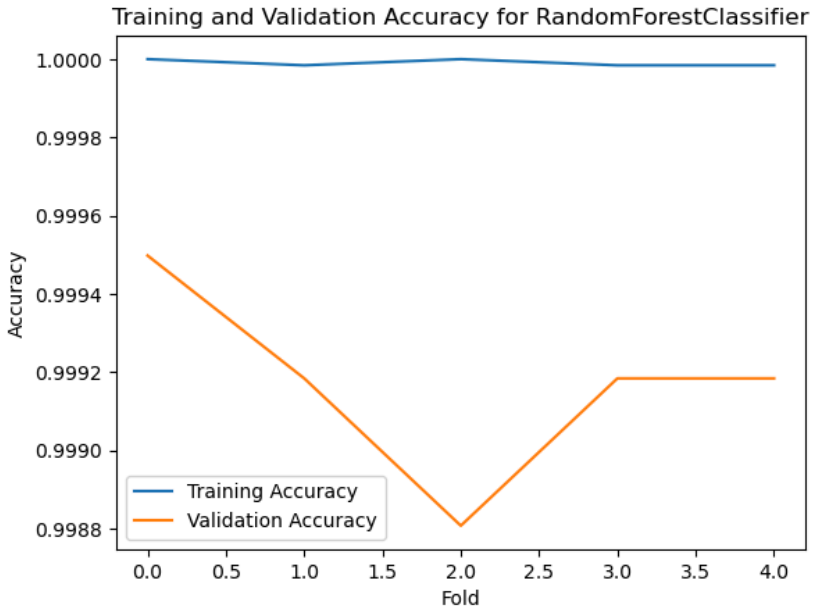


Figure 4.3.2: Training and Validation Accuracy Curves for Random Forest Classifier for DDoS Detection

Decision Tree Classifier

In the following Figure 4.3.3 we see the training and validation accuracy curve for DDoS Classification for DT

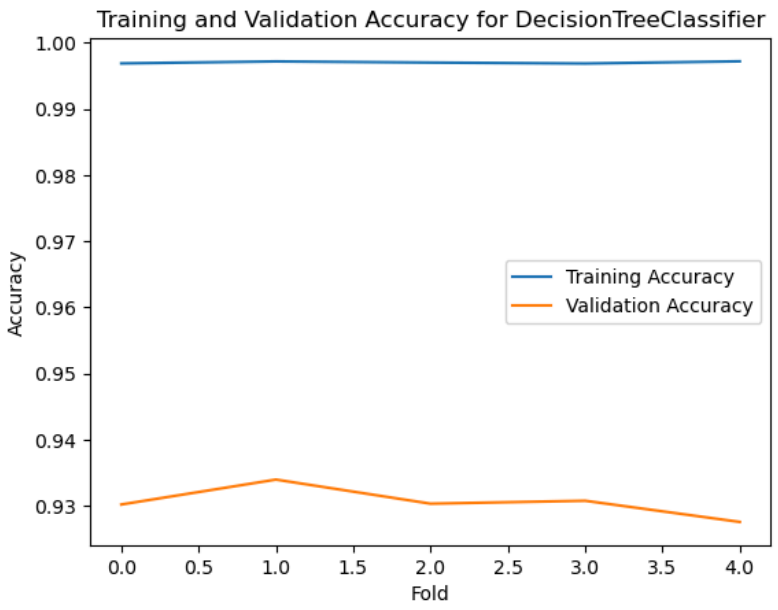


Figure 4.3.3: Training and Validation Accuracy Curves for Decision Tree Classifier for DDoS Classification

In the following Figure 4.3.4 we see the training and validation accuracy curve for DDoS Detection for DT

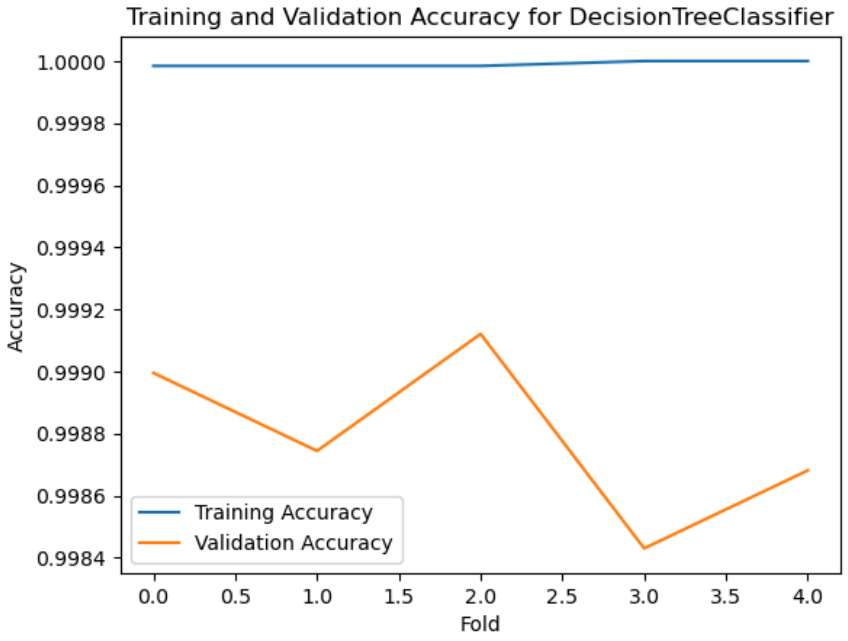


Figure 4.3.4: Training and Validation Accuracy Curves for Decision Tree Classifier for DDoS Detection

Support Vector Classifier

In the following Figure 4.3.5 we see the training and validation accuracy curve for DDoS Classification for SVC

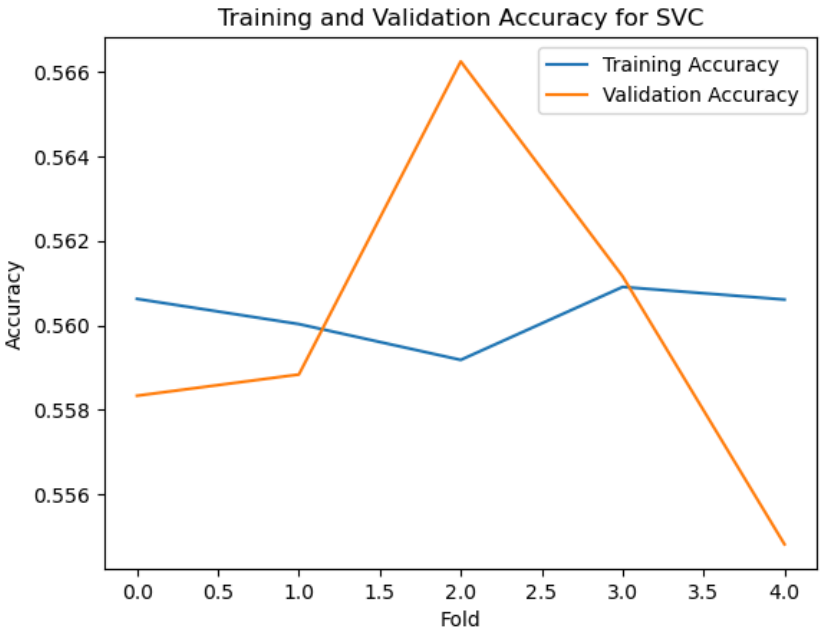


Figure 4.3.5: Training and Validation Accuracy Curves for Support Vector Classifier for DDoS Classification

In the following Figure 4.3.6 we see the training and validation accuracy curve for DDoS Detection for SVC

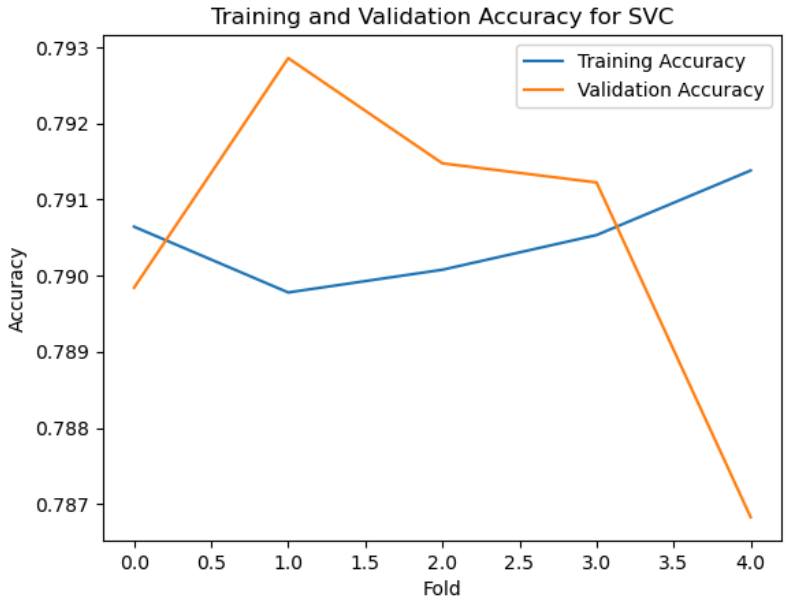


Figure 4.3.6 : Training and Validation Accuracy Curves for Support Vector Classifier for DDoS Detection

Logistic Regression

In the following Figure 4.3.7 we see the training and validation accuracy curve for DDoS Classification for LR

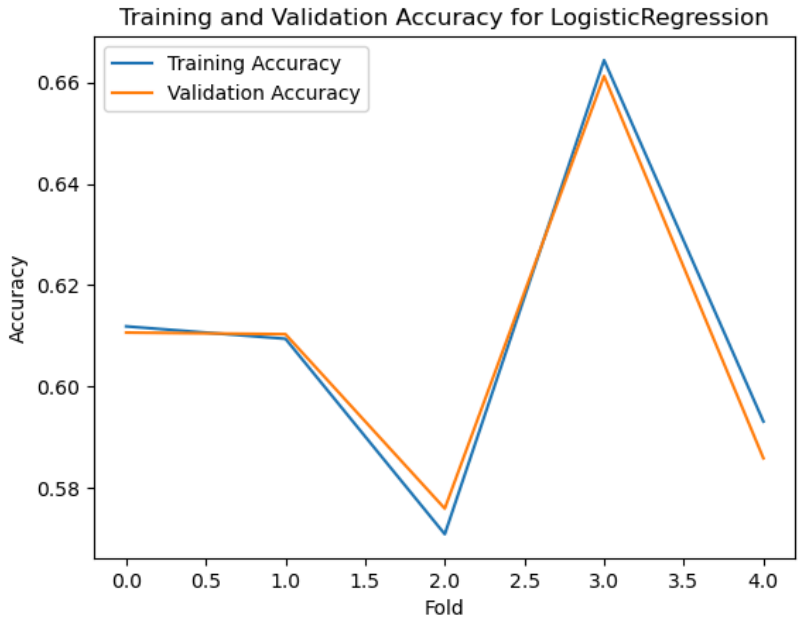


Figure 4.3.7: Training and Validation Accuracy Curves for Logistic Regression for DDoS Classification

In the following Figure 4.3.8 we see the training and validation accuracy curve for DDoS Detection for LR

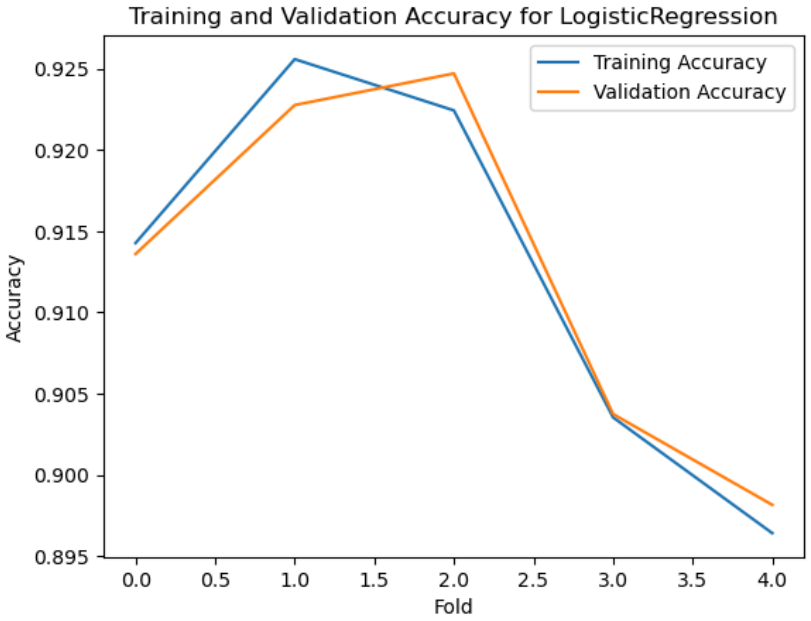


Figure 4.3.8: Training and Validation Accuracy Curves for Logistic Regression for DDoS Detection

K Neighbors Classifier

In the following Figure 4.3.9 we see the training and validation accuracy curve for DDoS Classification for KNN

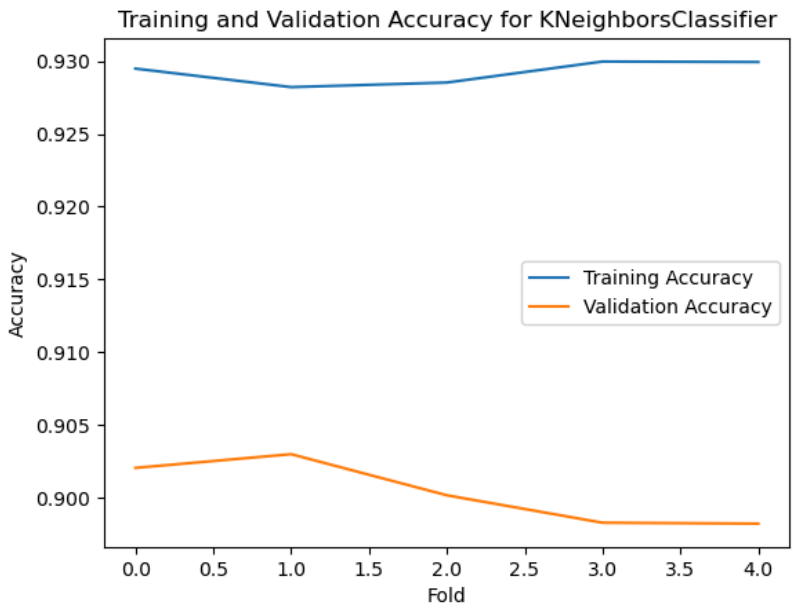


Figure 4.3.9: Training and Validation Accuracy Curves for K Neighbors Classifier for DDoS Classification

In the following Figure 4.3.10 we see the training and validation accuracy curve for DDoS Detection for KNN

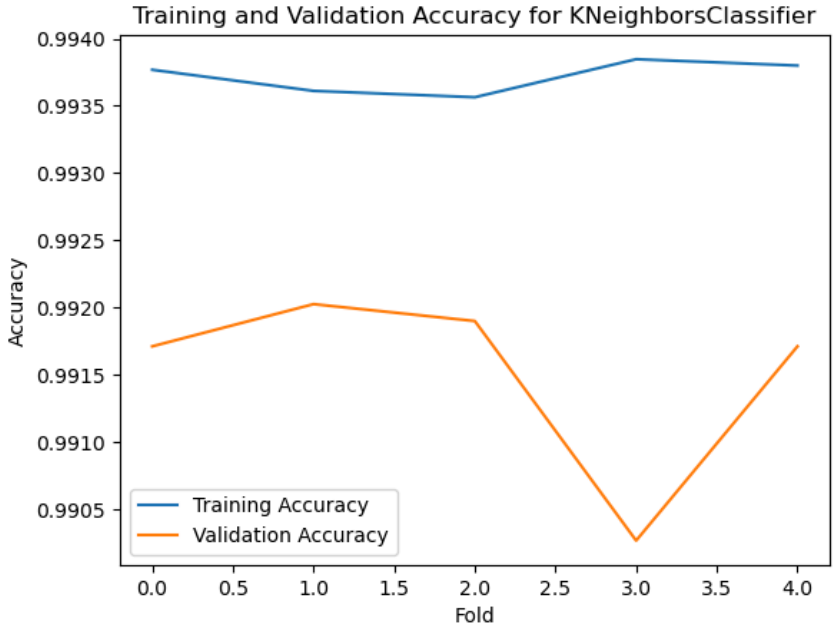


Figure 4.3.10: Training and Validation Accuracy Curves for K Neighbors Classifier for DDoS Detection

Gradient Boosting Classifier

In the following Figure 4.3.11 we see the training and validation accuracy curve for DDoS Classification for GBC

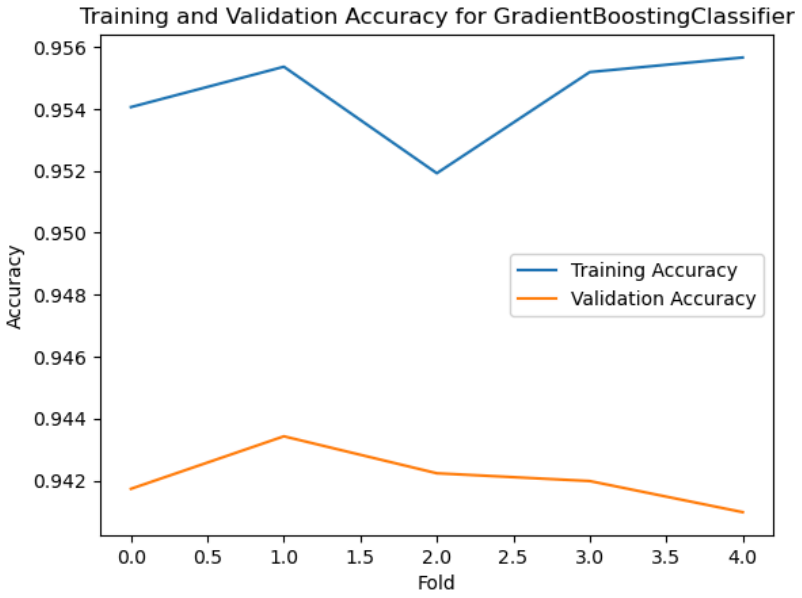


Figure 4.3.11: Training and Validation Accuracy Curves for Gradient Boosting Classifier for DDoS Classification

In the following Figure 4.3.12 we see the training and validation accuracy curve for DDoS Detection for GBC

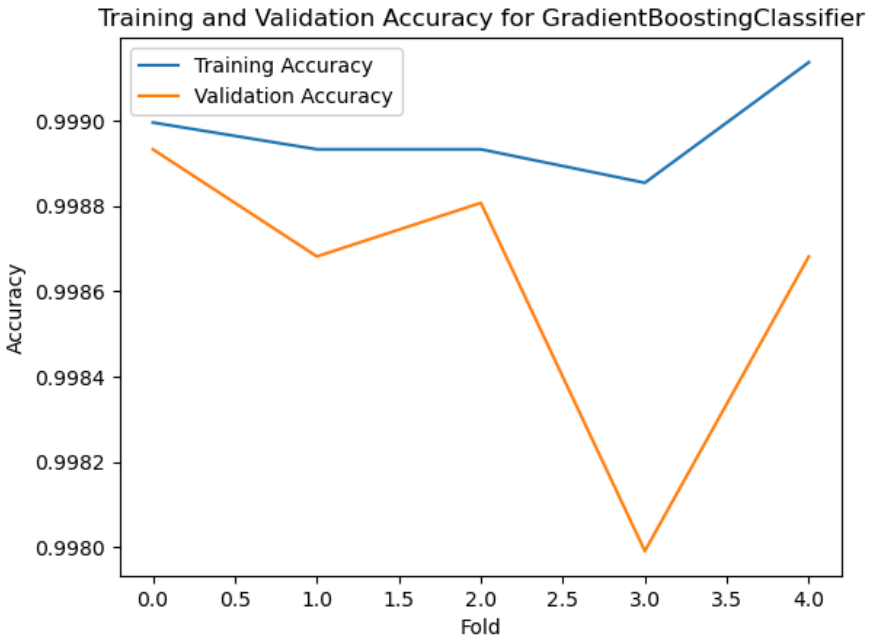


Figure 4.3.12: Training and Validation Accuracy Curves for Gradient Boosting Classifier for DDoS Detection

AdaBoost Classifier

In the following Figure 4.3.13 we see the training and validation accuracy curve for DDoS Classification for ABC

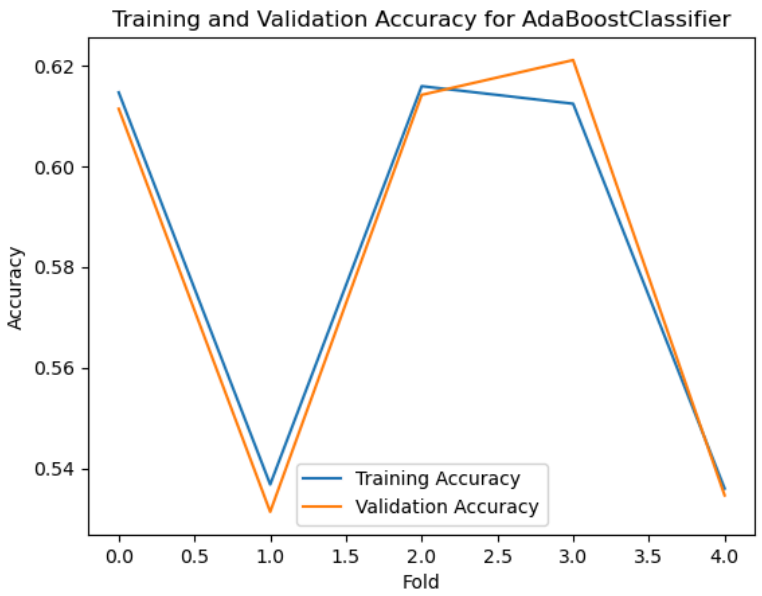


Figure 4.3.13: Training and Validation Accuracy Curves for AdaBoost Classifier for DDoS Classification

In the following Figure 4.3.14 we see the training and validation accuracy curve for DDoS Detection for ABC

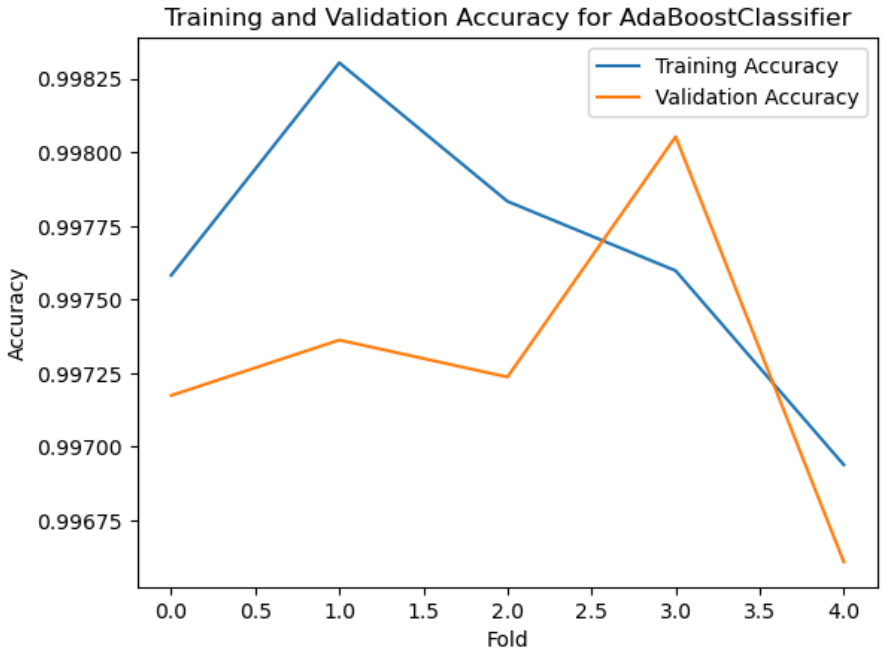


Figure 4.3.14: Training and Validation Accuracy Curves for AdaBoost Classifier for DDoS Detection

Bagging Classifier

In the following Figure 4.3.15 we see the training and validation accuracy curve for DDoS Classification for BC

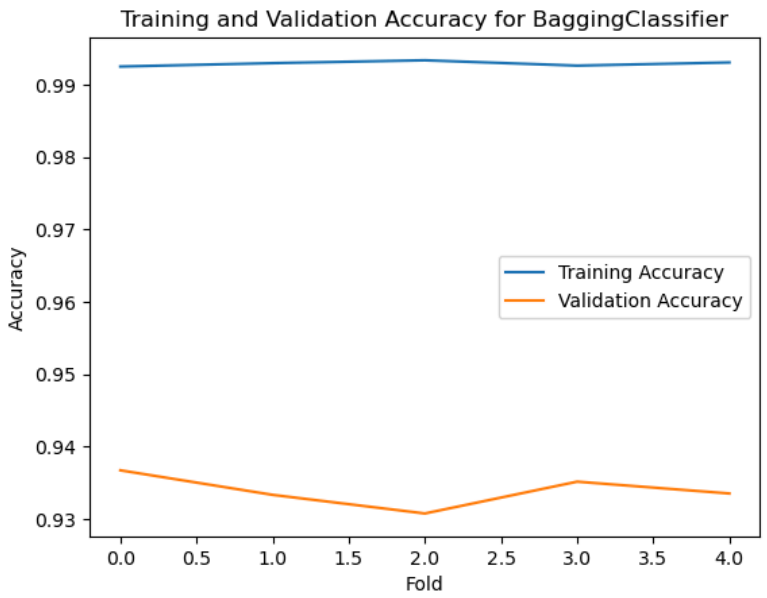


Figure 4.3.15: Training and Validation Accuracy Curves for Bagging Classifier for DDoS Classification

In the following Figure 4.3.16 we see the training and validation accuracy curve for DDoS Detection for BC

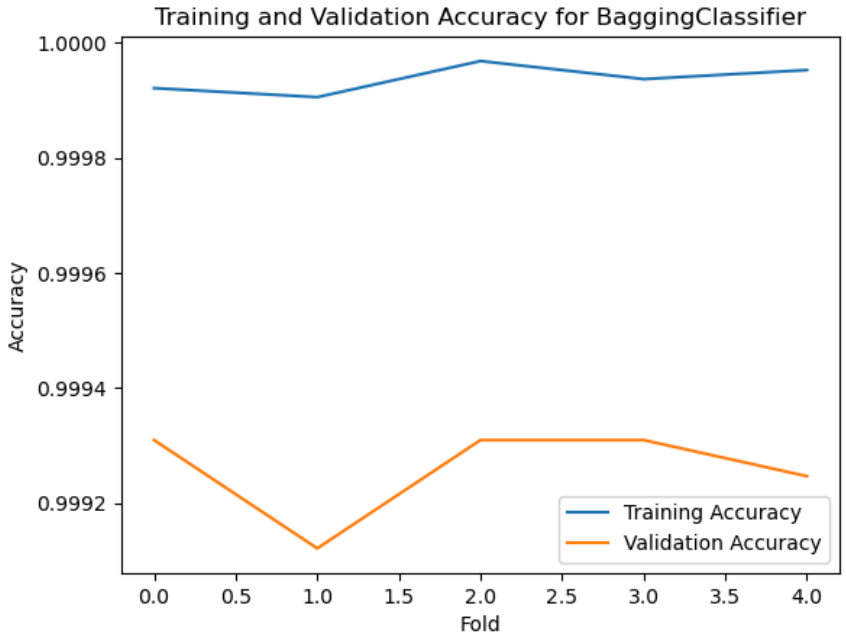


Figure 4.3.16: Training and Validation Accuracy Curves for Bagging Classifier for DDoS Detection

Bernoulli Naive Bayes

In the following Figure 4.3.17 we see the training and validation accuracy curve for DDoS Classification for BNB

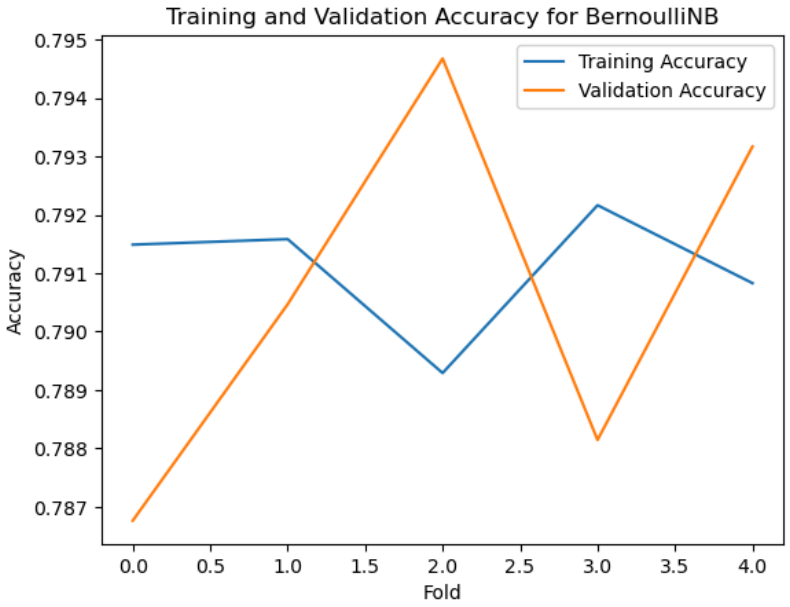


Figure 4.3.17: Training and Validation Accuracy Curves for Bernoulli Naive Bayes for DDoS Classification

In the following Figure 4.3.18 we see the training and validation accuracy curve for DDoS Detection for BNB

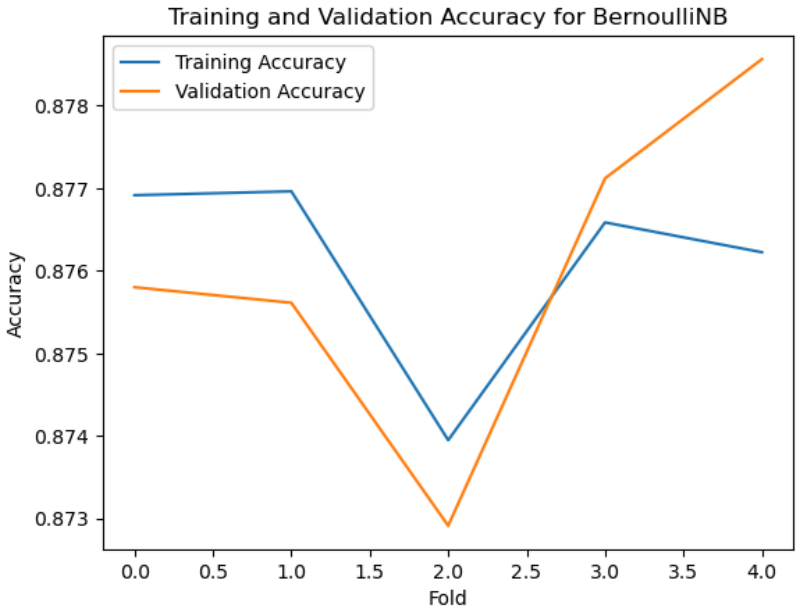


Figure 4.3.18: Training and Validation Accuracy Curves for Bernoulli Naive Bayes for DDoS Detection

Gaussian Naive Bayes

In the following Figure 4.3.19 we see the training and validation accuracy curve for DDoS Classification for GNB

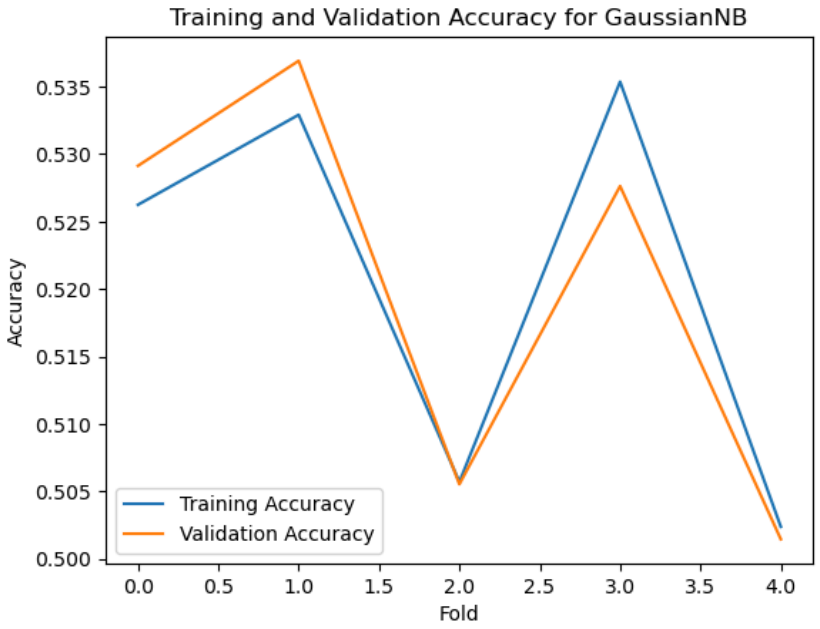


Figure 4.3.19: Training and Validation Accuracy Curves for Gaussian Naive Bayes for DDoS Classification

In the following Figure 4.3.20 we see the training and validation accuracy curve for DDoS Detection for GNB

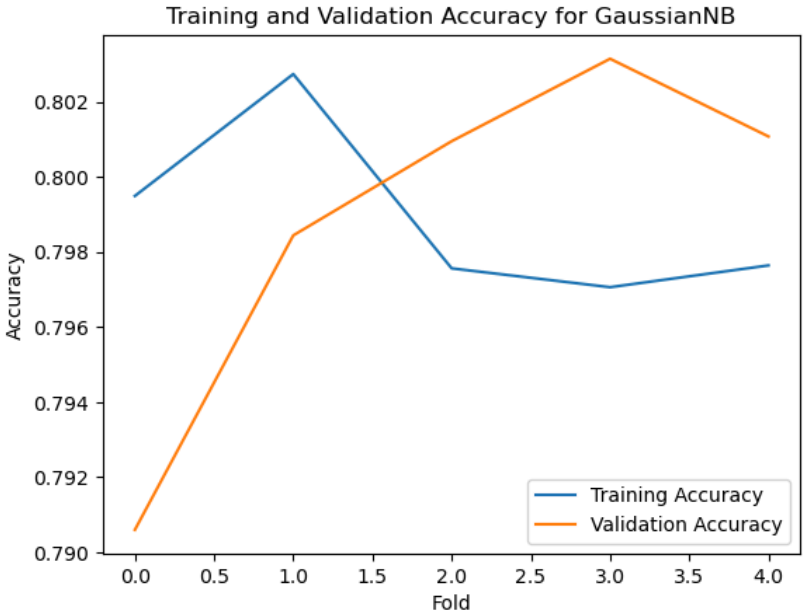


Figure 4.3.20: Training and Validation Accuracy Curves for Gaussian Naive Bayes for DDoS Detection

Linear Discriminant Analysis

In the following Figure 4.3.21 we see the training and validation accuracy curve for DDoS Classification for LDA

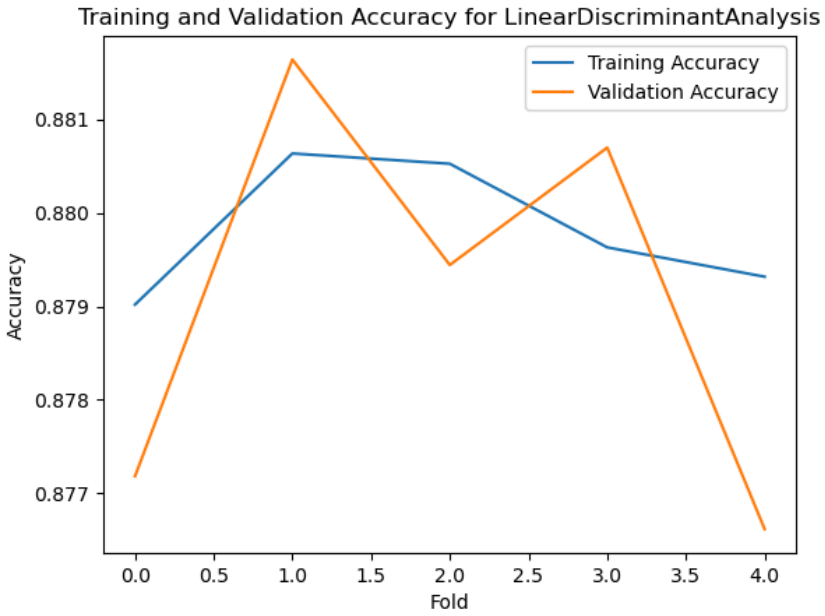


Figure 4.3.21: Training and Validation Accuracy Curves for Linear Discriminant Analysis for DDoS Classification

In the following Figure 4.3.22 we see the training and validation accuracy curve for DDoS Detection for LDA

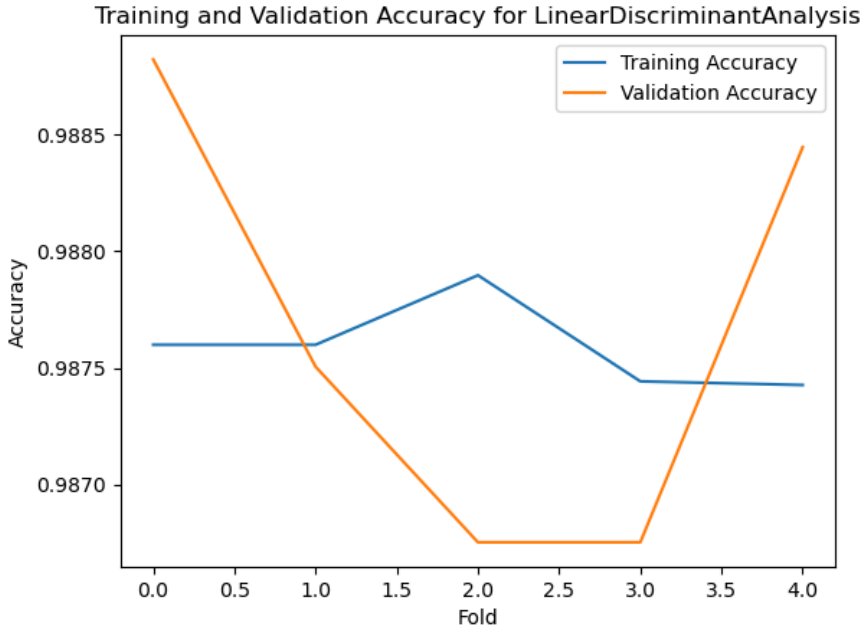


Figure 4.3.22: Training and Validation Accuracy Curves for Linear Discriminant Analysis for DDoS Detection

Quadratic Discriminant Analysis

In the following Figure 4.3.23 we see the training and validation accuracy curve for DDoS Classification for QDA

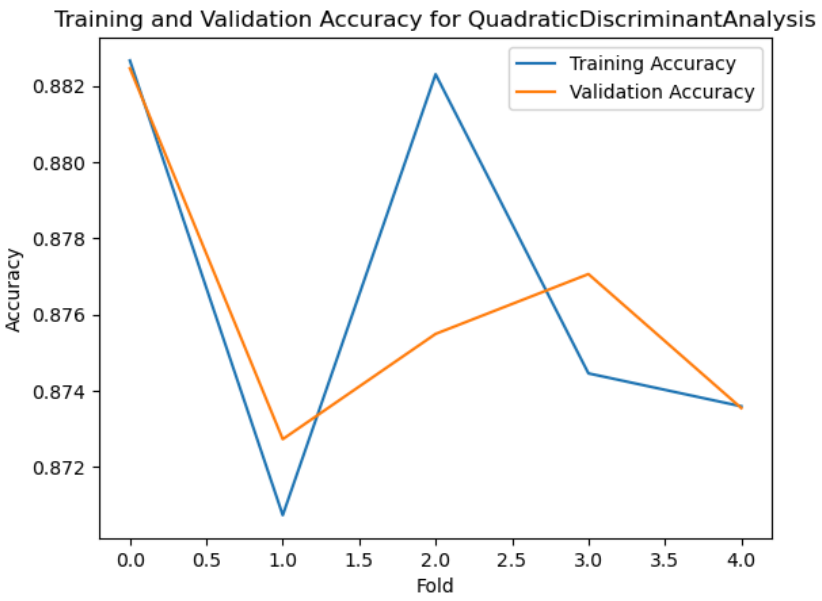


Figure 4.3.23: Training and Validation Accuracy Curves for Quadratic Discriminant Analysis for DDoS Classification

In the following Figure 4.3.24 we see the training and validation accuracy curve for DDoS Detection for QDA

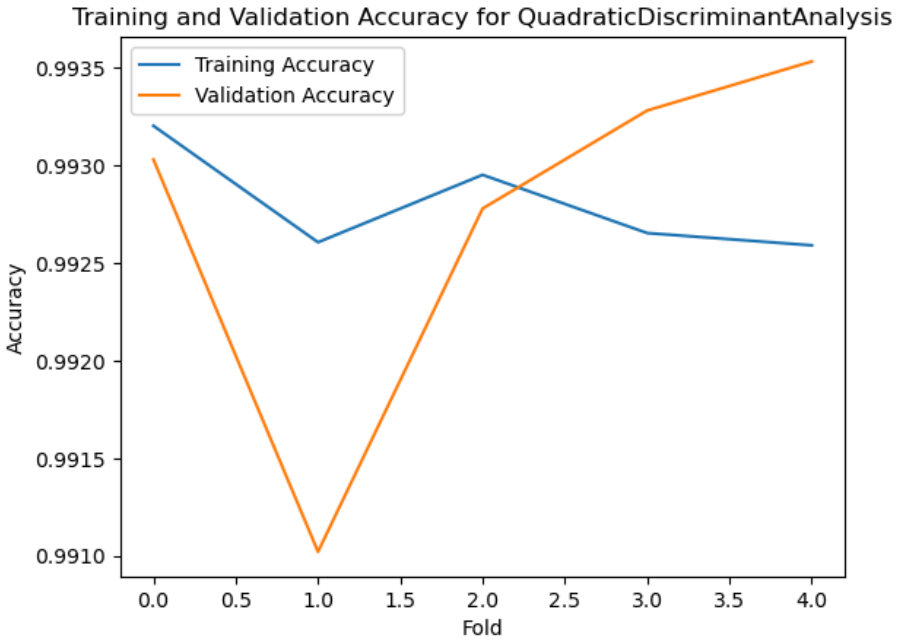


Figure 4.3.24: Training and Validation Accuracy Curves for Quadratic Discriminant Analysis for DDoS Detection

Multi-layer Perceptron Classifiers

In the following Figure 4.3.25 we see the training and validation accuracy curve for DDoS Classification for MLPs

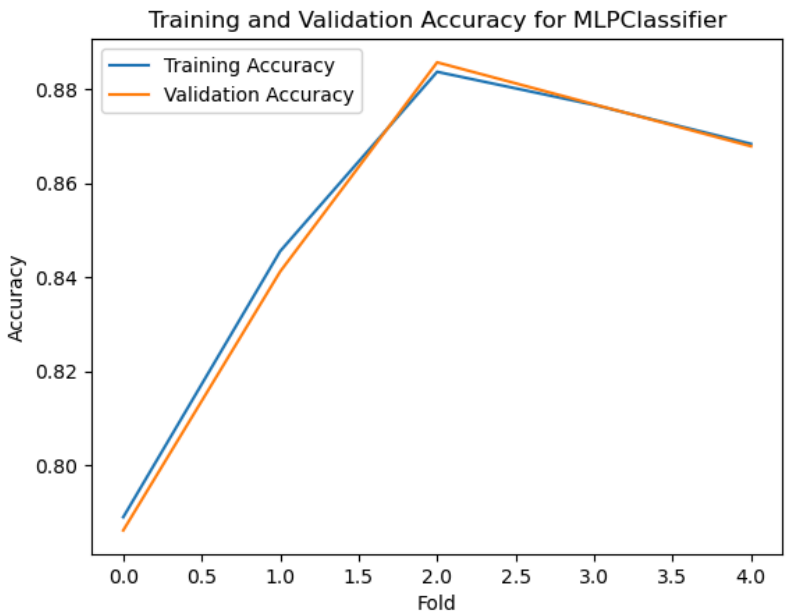


Figure 4.3.25: Training and Validation Accuracy Curves for Multilayer Perceptron Classifiers for DDoS Classification

In the following Figure 4.3.26 we see the training and validation accuracy curve for DDoS Detection for MLPs

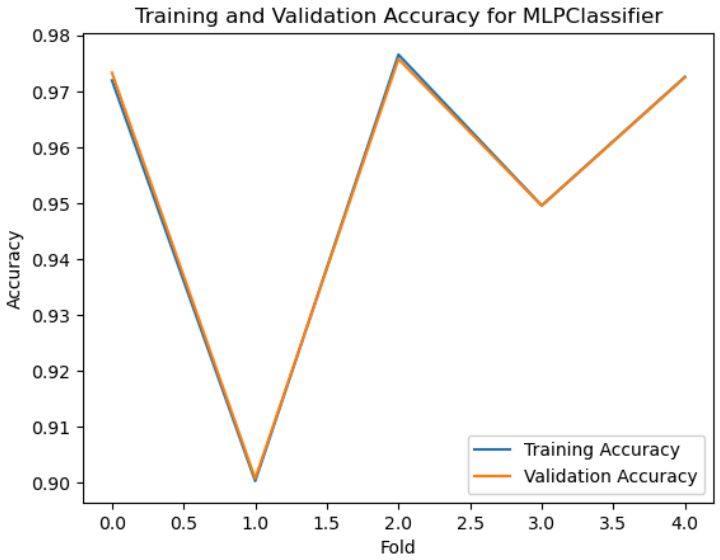


Figure 4.3.26: Training and Validation Accuracy Curves for Multi-layer Perceptron Classifiers for DDoS Detection

Now we are going to see the base models accuracy scores chart considering the target feature selections. So when we are concerned about the DDoS Detection then we need to consider the following Figure 4.3.27 where the target feature is set to “Class”. We see most of the base ML models are doing very good considering the 85% accuracy threshold mark.

	Model	Instance	Accuracy	Train Accuracy	Validation Accuracy
7	BaggingClassifier	(DecisionTreeClassifier(random_state=122086240...	0.999297	0.999937	0.999259
0	RandomForestClassifier	(DecisionTreeClassifier(max_features='sqrt', r...	0.999447	0.999991	0.999171
1	DecisionTreeClassifier	DecisionTreeClassifier()	0.999247	0.999991	0.998794
5	GradientBoostingClassifier	([DecisionTreeRegressor(criterion='friedman_ms...	0.998543	0.998970	0.998619
6	AdaBoostClassifier	(DecisionTreeClassifier(max_depth=1, random_st...	0.997589	0.997652	0.997287
11	QuadraticDiscriminantAnalysis	QuadraticDiscriminantAnalysis()	0.992315	0.992801	0.992729
4	KNeighborsClassifier	KNeighborsClassifier()	0.992516	0.993718	0.991523
10	LinearDiscriminantAnalysis	LinearDiscriminantAnalysis()	0.988145	0.987593	0.987655
12	MLPClassifier	MLPClassifier(max_iter=1000)	0.976140	0.954185	0.954364
3	LogisticRegression	LogisticRegression(max_iter=1000)	0.951979	0.912451	0.912596
8	BernoulliNB	BernoulliNB()	0.876833	0.876127	0.876002
9	GaussianNB	GaussianNB()	0.798573	0.798901	0.798845
2	SVC	SVC(probability=True)	0.792144	0.790481	0.790443

Figure 4.3.27: Base ML Models Results for DDoS Detection

And for the "Label" column being taken as target feature column we can see the base ML models performing in the following Figure 4.3.28. This result shows the impact that the models keep when the classification of DDoS attack types are the main focus.

	Model	Instance	Accuracy	Train Accuracy	Validation Accuracy
5	GradientBoostingClassifier	([DecisionTreeRegressor(criterion='friedman_ms...	0.945901	0.954433	0.942070
0	RandomForestClassifier	(DecisionTreeClassifier(max_features='sqrt', r...	0.936508	0.997043	0.934924
7	BaggingClassifier	(DecisionTreeClassifier(random_state=139239064...	0.935001	0.992902	0.933882
1	DecisionTreeClassifier	DecisionTreeClassifier()	0.931033	0.997036	0.930541
4	KNeighborsClassifier	KNeighborsClassifier()	0.901597	0.929229	0.900339
10	LinearDiscriminantAnalysis	LinearDiscriminantAnalysis()	0.882309	0.879825	0.879116
11	QuadraticDiscriminantAnalysis	QuadraticDiscriminantAnalysis()	0.876833	0.876746	0.876253
12	MLPClassifier	MLPClassifier(max_iter=1000)	0.491712	0.852628	0.851526
8	BernoulliNB	BernoulliNB()	0.791340	0.791071	0.790644
3	LogisticRegression	LogisticRegression(max_iter=1000)	0.590115	0.609921	0.608791
6	AdaBoostClassifier	(DecisionTreeClassifier(max_depth=1, random_st...	0.619701	0.583150	0.582532
2	SVC	SVC(probability=True)	0.726743	0.560269	0.559877
9	GaussianNB	GaussianNB()	0.555857	0.520520	0.520131

Figure 4.3.28: Base ML Models Results for DDoS Classification

Choosing the Best ML Models for Ensemble Technique

As per the Figure 4.3.27 and 4.3.28, the accuracy scores of the base ML models are very clear. Considering the two types of analogy DDoS Detection and Classification, we have chosen two ways to complete the ensemble classification and furthermore the final models generation.

When we are in the concept of DDoS Classification where we are classifying 13 types of end results, 12 being DDoS types and 01 for the Bening type we have gone for the best models that have the accuracy score of equal or greater than 85%. So based on Figure 4.3.28 we have ensembled the following base ML models.

i. Gradient Boosting Classifier
ii. Random Forest Classifier
iii. Bagging Classifier
iv. Decision Tree Classifier
v. K Neighbors Classifier
vi. Linear Discriminant Analysis
vii. Quadratic Discriminant Analysis
viii. Multi-layer Perceptron Classifiers

And when we are in the concept of DDoS Detection where we are detecting 02 types of end estimations, 01 being DDoS and 01 for the Bening we have gone for all the models. So based on Figure 4.3.27 we have ensembled the all the 13 base ML models.

i. Random Forest Classifier
ii. Decision Tree Classifier
iii. Support Vector Classifier
iv. Logistic Regression
v. K Neighbors Classifier
vi. Gradient Boosting Classifier
vii. AdaBoost Classifier
viii. Bagging Classifier
ix. Linear Discriminant Analysis
x. Quadratic Discriminant Analysis
xi. Multi-layer Perceptron Classifier
xii. Bernoulli Naive Bayes
xiii. Gaussian Naive Bayes

Ensemble Technique Results

Considering the best ML models we have performed both Hard and Soft voting

classifiers for the ensemble technique. In this way we were able to get the best results from our experiment and the results in the following Figure 4.3.29 shows the overall accuracy score for each of the configurations.

	Target Column	Hard Voting Accuracy	Soft Voting Accuracy
1	Class	0.998141	0.997941
0	Label	0.939220	0.935805

Figure 4.3.29: Accuracy Scores For Final Ensemble Models

So for DDoS Detection we are having an accuracy score of 99.81% and for DDoS Classification we are having an accuracy score of 93.92% when it comes to Hard Voting Classifiers. And when using Soft Voting Classifiers the accuracy was quite lower than Hard Voting in both the instances as in Detection analogy it is 93.92% and for classification 93.58%.

From this experiment we can also see that the accuracy scores tend to be higher in the detection spectrum rather than the classification spectrum as there are more items to be predicted correctly in the classification rather than only two in detection.

Here another common pattern we see is that the Hard Voting Classifier outperforms the Soft Voting Classifier every time and a graphical representation can be found in the following figure 4.3.30.

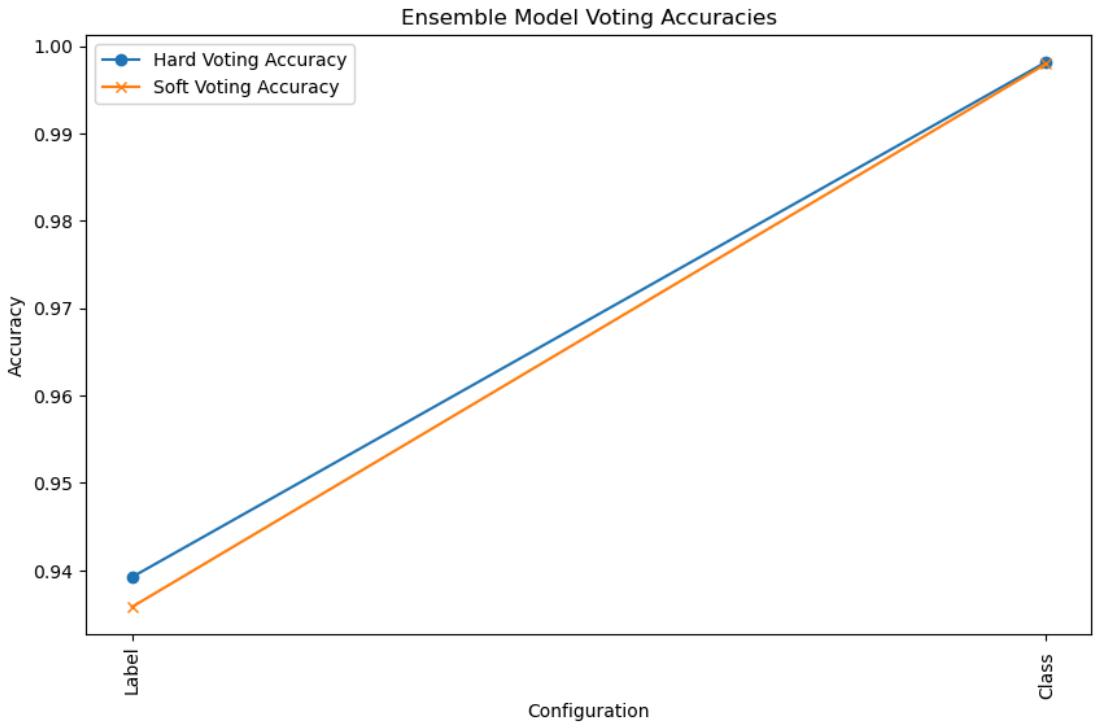


Figure 4.3.30: Ensemble Models Voting Accuracy Score

Except the Accuracy scores we have also considered the Confusion matrix and other parametric scores that show how the proposed models stack in the performance metrics.

Firstly when we are considering the DDoS Detection analogy then we can see the final model for detection, top 08 base ML Models ensembled with Hard Voting Classifier has the following confusion matrix as of the Figure 4.3.31.

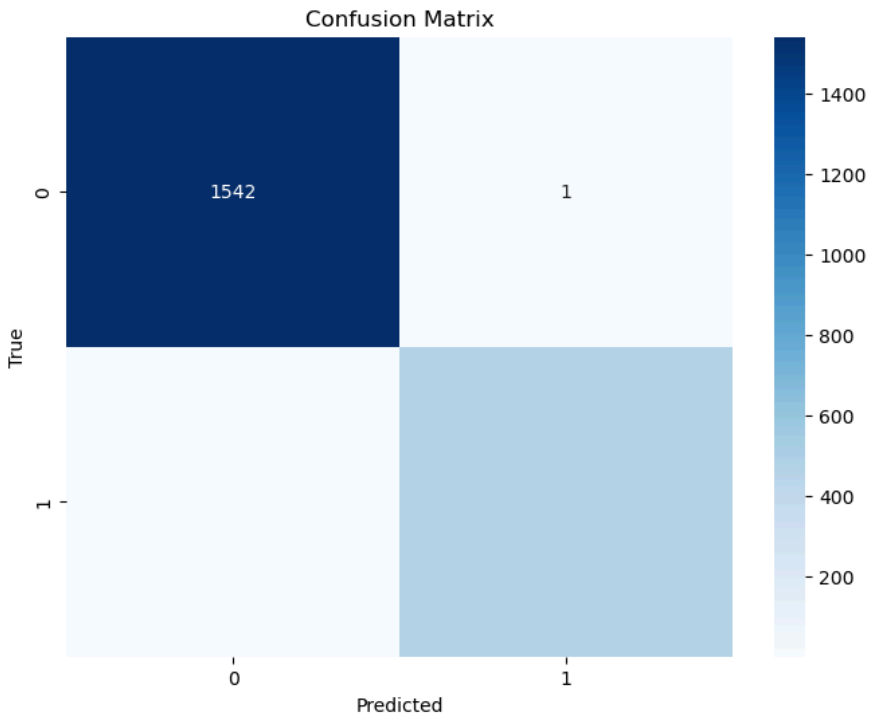


Figure 4.3.31: Confusion Matrix For DDoS Detection with final Detection Model

For the classification report we can see the following figure of Figure 4.3.32 for better understanding.

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	1543
1	1.00	1.00	1.00	456
accuracy			1.00	1999
macro avg	1.00	1.00	1.00	1999
weighted avg	1.00	1.00	1.00	1999

Figure 4.3.32: Classification Report For DDoS Detection with final Detection Model

Lastly when we are using DDoS Classification analogy then we can see the final model for classification, all 13 base ML Models ensembled with Hard Voting Classifier have the following confusion matrix as of the Figure 4.3.33.

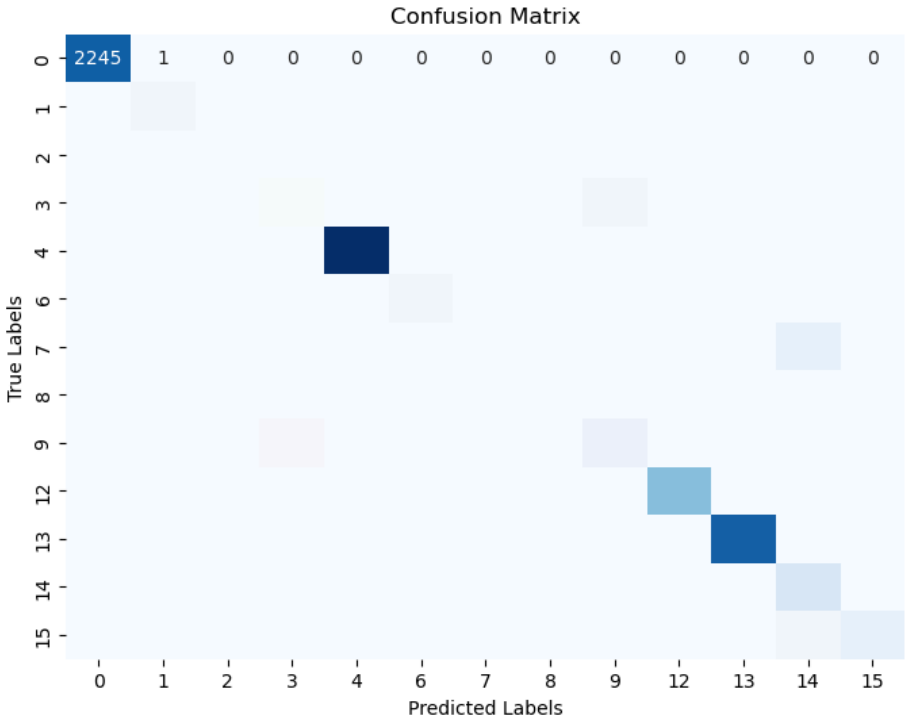


Figure 4.3.33: Confusion Matrix For DDoS Classification with final Classification Model

For the classification report we can see the following figure of Figure 4.3.34 for better understanding.

Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	2246
1	0.54	0.54	0.54	91
2	0.27	0.29	0.28	31
3	0.39	0.22	0.28	147
4	1.00	1.00	1.00	2758
6	0.55	0.84	0.67	64
7	0.36	0.02	0.04	249
8	0.47	0.18	0.26	50
9	0.52	0.74	0.61	216
12	1.00	0.99	0.99	1177
13	1.00	1.00	1.00	2250
14	0.58	0.94	0.72	422
15	0.92	0.71	0.80	245
accuracy			0.93	9946
macro avg	0.66	0.65	0.63	9946
weighted avg	0.93	0.93	0.92	9946

Figure 4.3.34: Classification Report For DDoS Classification with final Classification Model

4.4 Model Result Summary

Since we see from the analysis and all the testing that it has made the overall accuracy score really high because if we look at the detection accuracy it has been achieved by 99.81% and the classification accuracy be like 93.92%. On the other hand, with more classification reports, the confusion matrices also have a more appearance to detect DDoS packets with a high accuracy.

The best performing models in our analysis were Random Forest Classifier, Bagging Classifier, Decision Tree Classifier, Gradient Boosting Classifier, KNeighbors Classifier, and Linear Discriminant Analysis. These models resulted in very high accuracy in detecting distributed denial of service (DDoS) attacks. They were chosen based on several key factors, each of which makes them well-suited for the job. We have two powerful ensemble learning techniques, RandomForestClassifier and BaggingClassifier that provide robust solutions against the outliers and noisy data which let us provide consistent performance to a wide range of datasets including CIC-DDoS2019. Similarly, our DecisionTreeClassifier is naturally quite flexible, so will work well on different data sources and feature distributions, improving adaptability to the unique challenges of DDoS detection. An important tuning parameter for gradient boosting is the number of weak learners-that is the number of decision trees it should build, which can both help increase the accuracy, since we are not have a high variance and low bias model, and avoid to have a highly complex model, that may take to the overfit of problems. Moreover, the KNeighborsClassifier is excellent for spotting local anomalies and patterns in the data that would hint to DDoS attacks. Being able to use the collective knowledge of multiple models enables ensemble approaches to have superior generalization performance and resistance to overfitting, making them critical techniques in cyber security for reinforcing our defense against a continuously evolving threat landscape that DDoS attacks present.

All of the ensemble models, in particular the Hard Voting Classifiers, demonstrated acceptable DDoS detection and classification performances. It also became obvious a founding piece for the model behavior was the choice of columns for the target. More generalized classes of targets were also more robustly learned by the models, which learned to categorize benign and malicious activity together based on general features in the first case, whereas models trained for specific types of attacks might not always be able to tell one threat from the other the next moment.

4.5 Comparison with Previous Work

When it comes to the score comparison with other models then we can see a great achievement in front. We have collected many test results from different papers and

references that have shown their accuracy scores that they could come up with. So the following Table 4.5 will depict the model scores based on accuracy that are having the same focus of DDoS Detection and Classification.

Table 4.5: Comparison of proposed model accuracy results with previous models

Author	Dataset	Models	Prediction Type	Accuracy
Yungaicela-Naula et al. [22]	CICDDoS2019	ML	Detection	99.77%
Manikumaret al. [25]	CICDDoS2019	ML [KNN, DT, RF]	Detection	95.19%
Novaes et al. [26]	CICDDoS2019	ML	Detection	99.74%
Maranhão et al. [27]	CICDDoS2019	ML [DT, GB, RF]	Detection	99.76%
Chartuni et al. [28]	CICDDoS2019	DL [3 Dense Layer]	Classification - 10 classes	94%
Alghazzawi et al. [29]	CICDDoS2019	DL [CNN + BiLSTM]	Detection	94.52%
Haq et al. [30]	N-BaIoT	DL [DNNBoT1 & DNNBoT2]	Classification - 11 classes	90.71%
Haq et al. [30]	N-BaIoT	DL [DNNBoT1 & DNNBoT2]	Detection	91.44%
Kumar et al. [31]	CICDDoS2019	DL [LSTM]	Detection	98.6%
	CICDDoS2019	ML Ensemble [DT, RF, LR, KNN, GB, AB, BC, LDA, QDA, SVC, BNB, GNB, MLP]	Detection	99.81%

	CICDDoS2019	ML Ensemble [GB, RF, BC, DT, KNN, LDA, QDA, MLP]	Classification - 13 classes	93.92%

So with the comparison analysis we can see our final models are quite competitive and very much accurate from most other models which really means a lot given the fact that we have only considered the ML models rather than the addition of DL models.

4.6 Companion Tool In Action

As we have the best model selected we used this model to do all the predictions for DDoS Detection and Classification in our website or the companion tool for the cyber security analysts. The service can be found by going to the link [41].

If not logged in then we will see the guest-home section as the following Figure 4.6.1

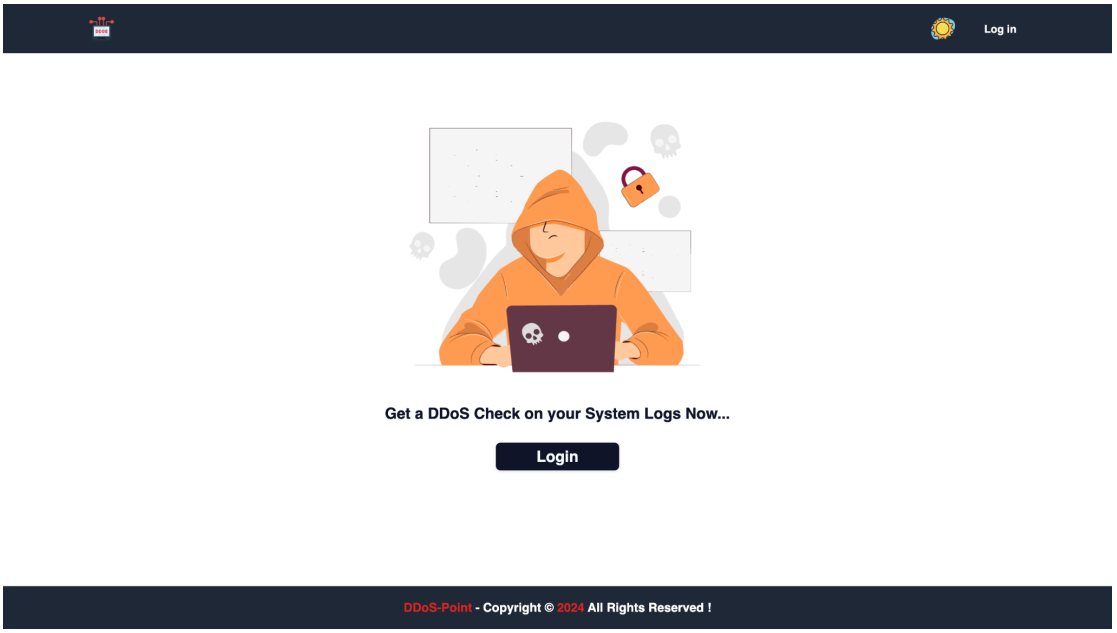


Figure 4.6.1: Companion Tool Website Guest View

It is required to login to the system with an administrative username and password which can be seen in the following Figure 4.6.2 for starting to use the system itself.

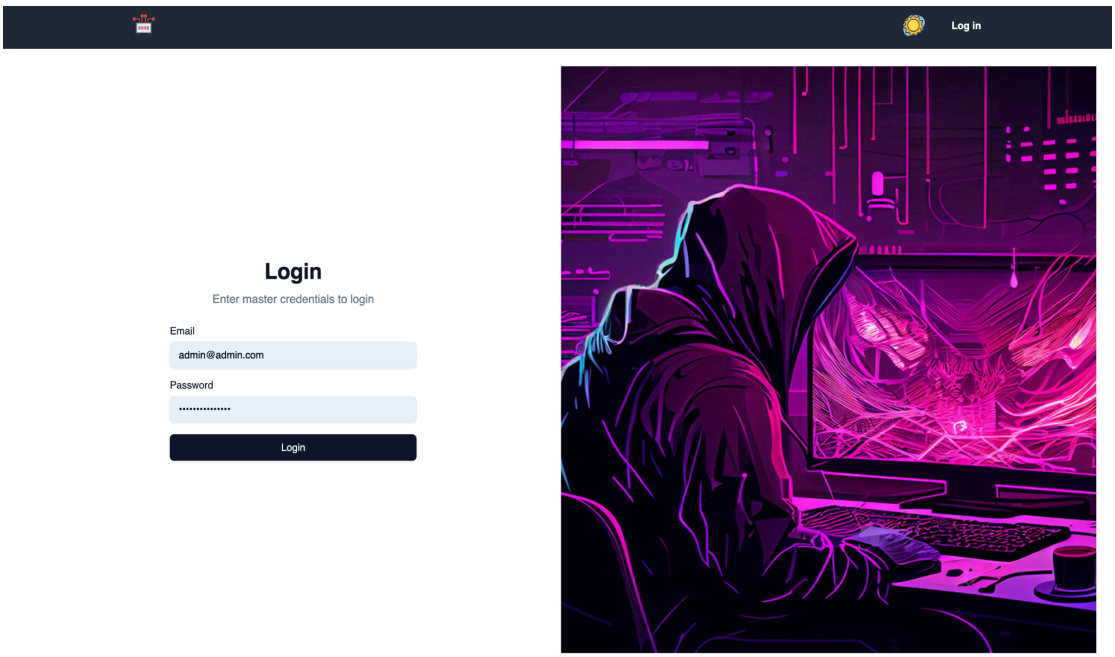


Figure 4.6.2: Companion Tool Website Login Screen

When logged in properly we can see the initial page where there is an option to upload the log file and consider the most important 50 independent feature sets or all of the independent features considered for the detection or classification mechanism as shown in the following Figure 4.6.3.

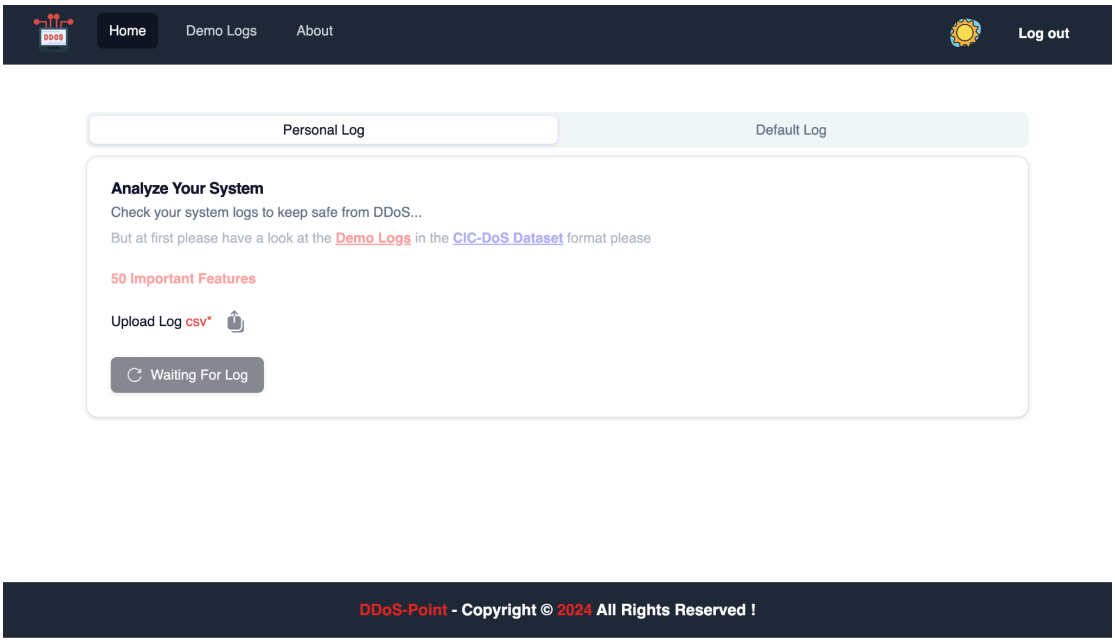
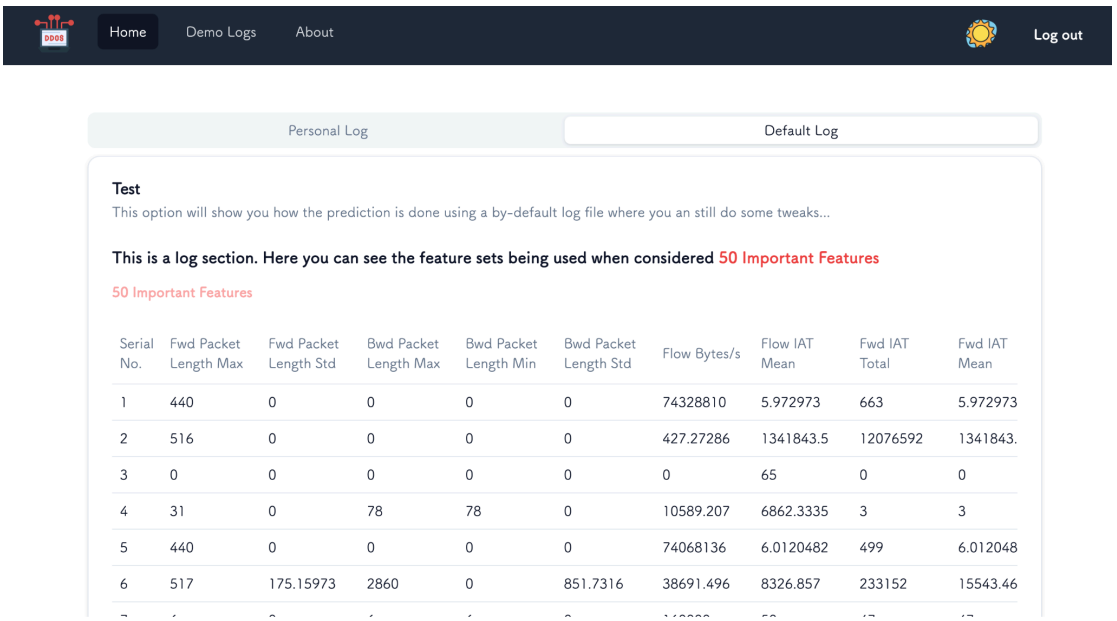


Figure 4.6.3: Companion Tool Website Log File Upload Screen

There is a tab and in that we have two options like, Personal Log which depicts that the user will add his/her own system logs for DDoS prediction and the other is the Default Log to see the overall process from the dataset to their prediction without

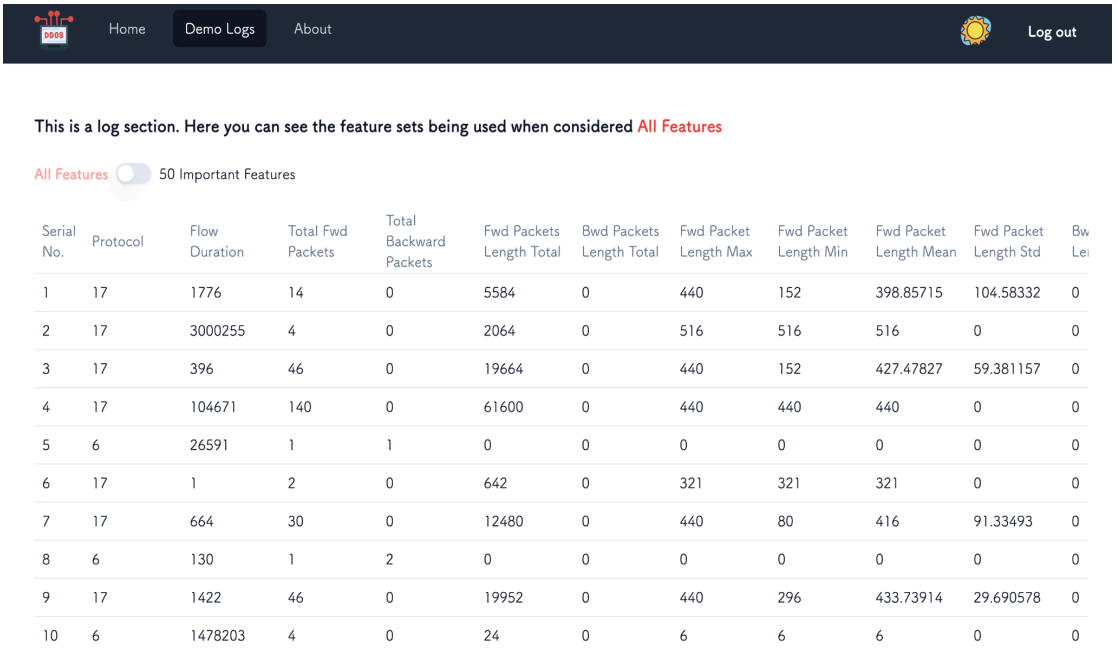
inserting any log files rather seeing an output with a ready log which is visible in the following Figure 4.6.4.



Serial No.	Fwd Packet Length Max	Fwd Packet Length Std	Bwd Packet Length Max	Bwd Packet Length Min	Bwd Packet Length Std	Flow Bytes/s	Flow IAT Mean	Fwd IAT Total	Fwd IAT Mean
1	440	0	0	0	0	74328810	5.972973	663	5.972973
2	516	0	0	0	0	427.27286	1341843.5	12076592	1341843.
3	0	0	0	0	0	0	65	0	0
4	31	0	78	78	0	10589.207	6862.3335	3	3
5	440	0	0	0	0	74068136	6.0120482	499	6.012048
6	517	175.15973	2860	0	851.7316	38691.496	8326.857	233152	15543.46

Figure 4.6.4: Companion Tool Website Default Log File Screen

Also before uploading any logs it is better to see the demo logs which shows all the features being considered and also a log file of 10 rows for a good reference. The following Figure 4.6.5 shows how the demo rows of logs are seen and the values along the corresponding columns as well.



Serial No.	Protocol	Flow Duration	Total Fwd Packets	Total Backward Packets	Fwd Packets Length Total	Bwd Packets Length Total	Fwd Packet Length Max	Fwd Packet Length Min	Fwd Packet Length Mean	Fwd Packet Length Std	Bw Le
1	17	1776	14	0	5584	0	440	152	398.85715	104.58332	0
2	17	3000255	4	0	2064	0	516	516	516	0	0
3	17	396	46	0	19664	0	440	152	427.47827	59.381157	0
4	17	104671	140	0	61600	0	440	440	440	0	0
5	6	26591	1	1	0	0	0	0	0	0	0
6	17	1	2	0	642	0	321	321	321	0	0
7	17	664	30	0	12480	0	440	80	416	91.33493	0
8	6	130	1	2	0	0	0	0	0	0	0
9	17	1422	46	0	19952	0	440	296	433.73914	29.690578	0
10	6	1478203	4	0	24	0	6	6	6	0	0

Figure 4.6.5: Companion Tool Website Demo Logs

For the considered metrics or feature sets on which the overall model prediction will be happening can be seen in the Figure 4.6.6 where they are addressed loudly.

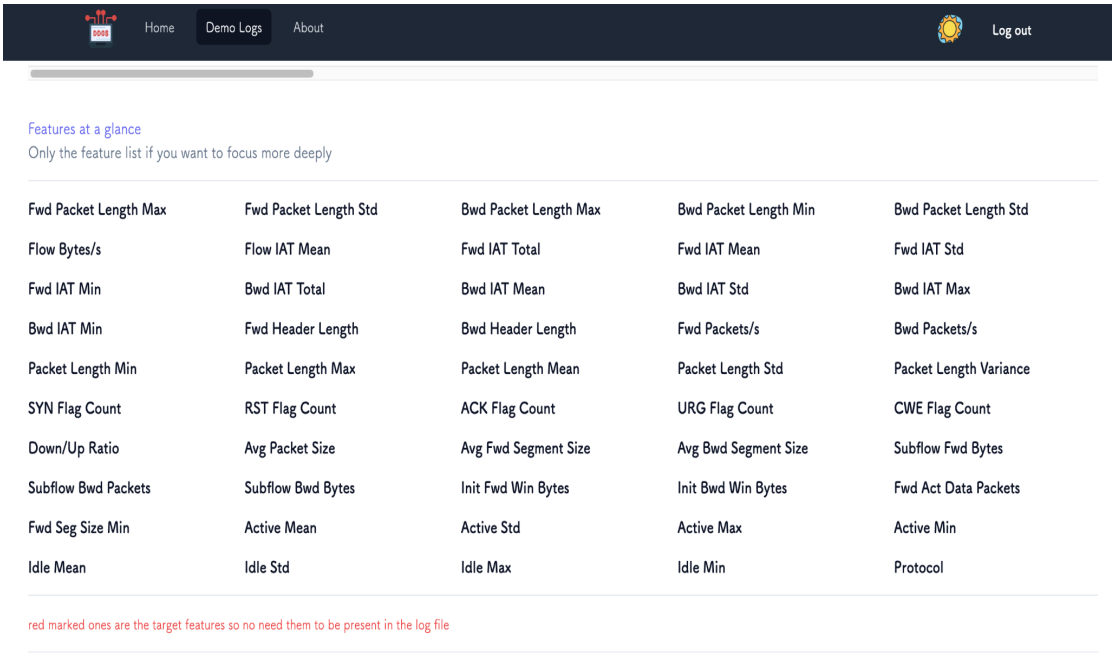
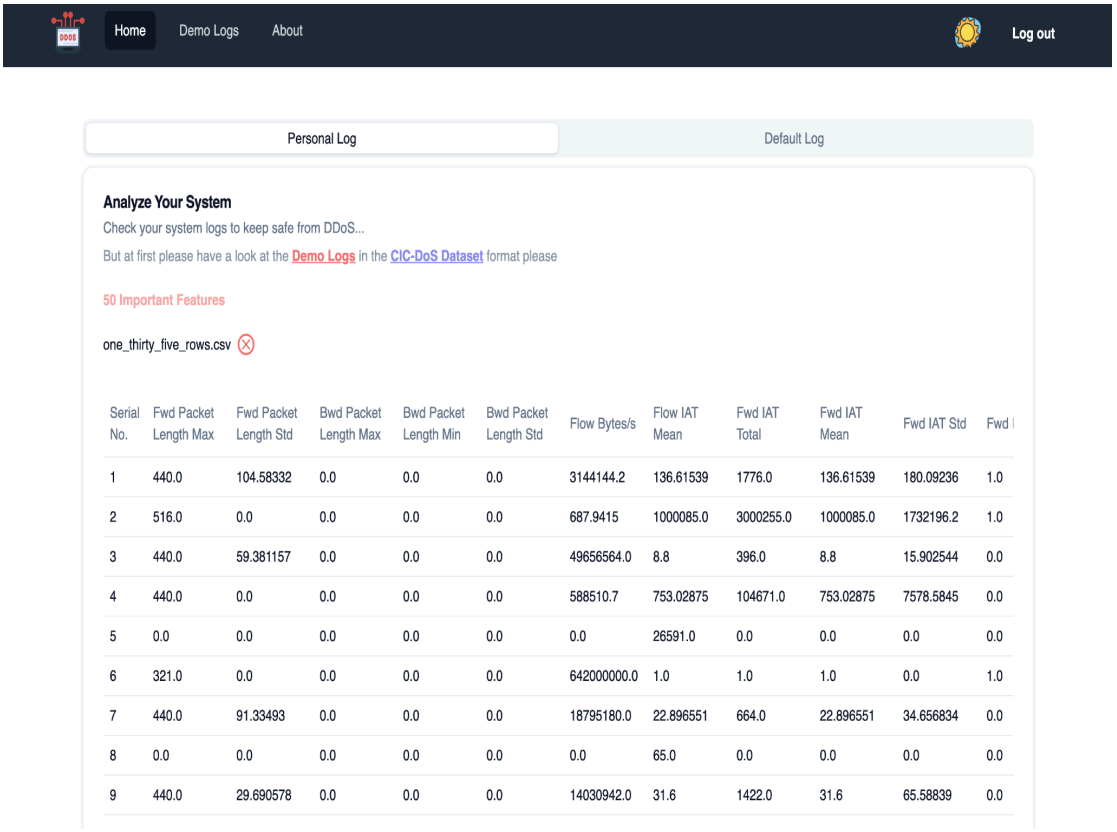


Figure 4.6.6: Companion Tool Website Considering Feature Sets Announcement

The same thing can be found for the most important features where not all the features are being considered. We can now focus on the main result part where we are uploading a log file and getting the result based on the prediction with the best ensemble model that we are getting.

Now after uploading the log file we will be able to see the entire log file being shown in the screen as like Figure 4.6.7.



Serial No.	Fwd Packet Length Max	Fwd Packet Length Std	Bwd Packet Length Max	Bwd Packet Length Min	Bwd Packet Length Std	Flow Bytes/s	Flow IAT Mean	Fwd IAT Total	Fwd IAT Mean	Fwd IAT Std	Fwd I
1	440.0	104.58332	0.0	0.0	0.0	3144144.2	136.61539	1776.0	136.61539	180.09236	1.0
2	516.0	0.0	0.0	0.0	0.0	687.9415	1000085.0	3000255.0	1000085.0	1732196.2	1.0
3	440.0	59.381157	0.0	0.0	0.0	49656564.0	8.8	396.0	8.8	15.902544	0.0
4	440.0	0.0	0.0	0.0	0.0	588510.7	753.02875	104671.0	753.02875	7578.5845	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	26591.0	0.0	0.0	0.0	0.0
6	321.0	0.0	0.0	0.0	0.0	642000000.0	1.0	1.0	1.0	0.0	1.0
7	440.0	91.33493	0.0	0.0	0.0	18795180.0	22.896551	664.0	22.896551	34.656834	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	65.0	0.0	0.0	0.0	0.0
9	440.0	29.690578	0.0	0.0	0.0	14030942.0	31.6	1422.0	31.6	65.58839	0.0

Figure 4.6.7 : Companion Tool Website Log File Uploaded

At the end of the page we can see the Analyze Log button which will eventually make the prediction process started as seen in Figure 4.6.8

121	17	0.0	3004462.0	6006889	3004462.0	6006889.0	0.0	0	1645053.1	516.0	516.0	516.0	516.0
122	17	0.0	3.0	3	3.0	3.0	0.0	0	0.0	1464.0	1464.0	1464.0	1464.0
123	17	0.0	110431.0	214357	110431.0	214357.0	0.0	0	58746.906	348.0	348.0	321.0	321.0
124	6	0.0	55.0	150	47.0	47.0	0.0	1	0.0	6.0	6.0	6.0	6.0
125	17	0.0	3012028.0	3012031	3012028.0	3012031.0	0.0	0	1738994.2	516.0	516.0	516.0	516.0
126	17	0.0	2.0	2	2.0	2.0	0.0	0	0.0	247.0	247.0	247.0	247.0
127	6	0.0	4223926.0	7316069	3.0	3.0	0.0	1	0.0	0.0	0.0	0.0	0.0
128	6	14543581.0	14543581.0	28543528	14543681.0	28543528.0	14271713.0	1	7819330.5	6.0	6.0	6.0	6.0
129	6	7999964.0	7999964.0	12000390	7999964.0	12000390.0	7999964.0	1	3577727.8	0.0	0.0	0.0	0.0
130	17	0.0	96.0	653	96.0	653.0	0.0	0	28.822205	440.0	440.0	440.0	440.0
131	6	0.0	4997630.0	6482413	119547.0	1484780.0	0.0	1	38695.082	95.4	95.4	0.0	0.0
132	17	0.0	305.0	754	305.0	754.0	0.0	0	40.084576	440.0	440.0	440.0	440.0
133	6	10031028.0	10031028.0	10103375	10031028.0	10072474.0	10031028.0	0	4481414.0	149.33333	149.33333	0.0	0.0
134	6	58556550.0	58556550.0	60437733	58582148.0	60437732.0	58556550.0	1	8042578.0	194.96297	194.96297	0.0	0.0
135	17	0.0	50.0	50	50.0	50.0	0.0	0	0.0	621.0	621.0	621.0	621.0

Figure 4.6.8: Companion Tool Website Analyze Button For Log File Processing

As the log file is being sent to the backend from the frontend we need to give it some time as the prediction will be done in the backend and then the final result will be sent back to the frontend. Until there are any results we can see a skeleton figure as per the following Figure 4.6.9.

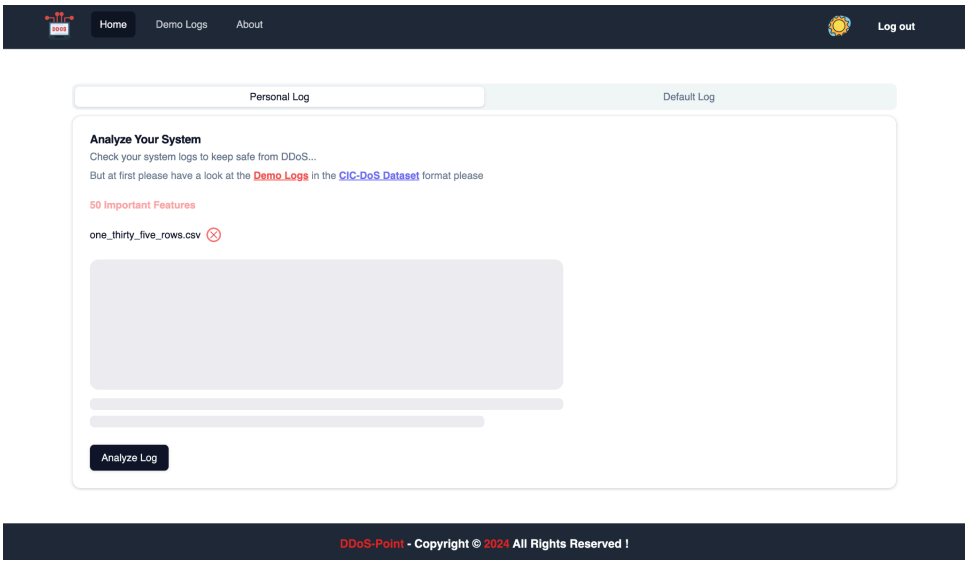
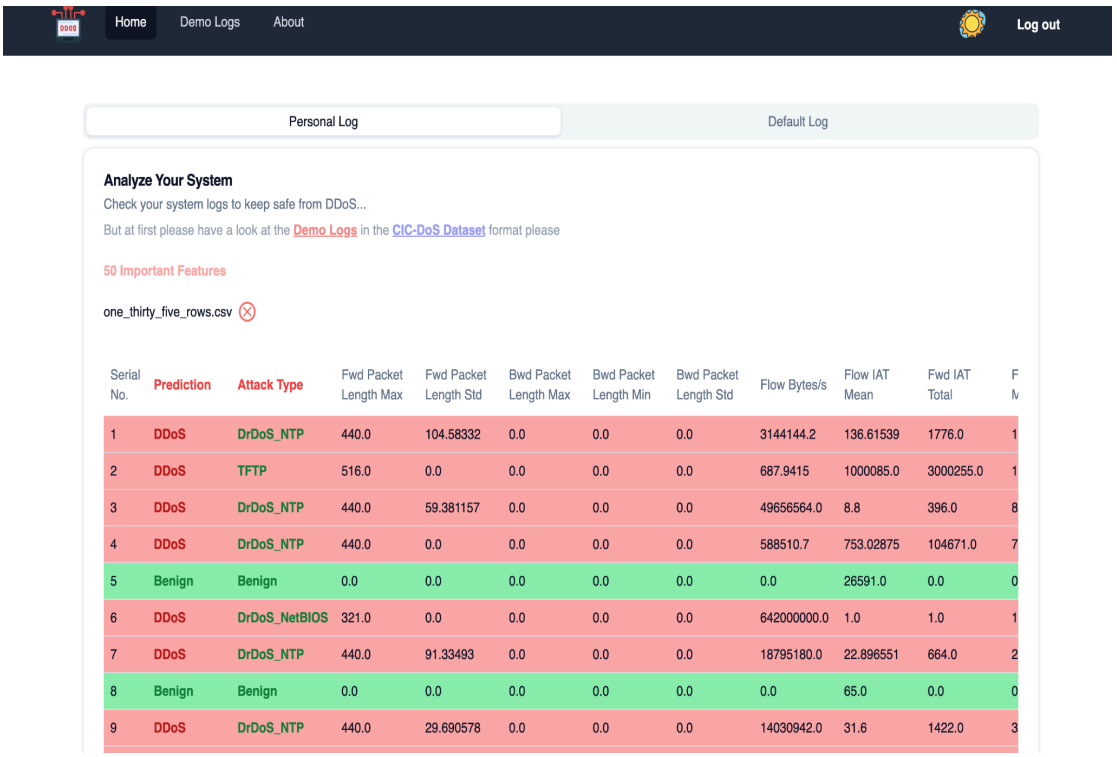


Figure 4.6.9: Companion Tool Website Analyzing Log File and Waiting For Response

As when the prediction is complete and the results can be seen in the frontend then the red marked rows mean the DDoS packets and the green ones depict Benign packets. Also for the DDoS attacked packets we can see the classification or the attack types and the detection as well in the result as per Figure 4.6.10.



Serial No.	Prediction	Attack Type	Fwd Packet Length Max	Fwd Packet Length Std	Bwd Packet Length Max	Bwd Packet Length Min	Bwd Packet Length Std	Flow Bytes/s	Flow IAT Mean	Fwd IAT Total	F M
1	DDoS	DrDoS_NTP	440.0	104.58332	0.0	0.0	0.0	3144144.2	136.61539	1776.0	1
2	DDoS	TFTP	516.0	0.0	0.0	0.0	0.0	687.9415	1000085.0	3000255.0	1
3	DDoS	DrDoS_NTP	440.0	59.381157	0.0	0.0	0.0	49656564.0	8.8	396.0	8
4	DDoS	DrDoS_NTP	440.0	0.0	0.0	0.0	0.0	588510.7	753.02875	104671.0	7
5	Benign	Benign	0.0	0.0	0.0	0.0	0.0	0.0	26591.0	0.0	0
6	DDoS	DrDoS_NetBIOS	321.0	0.0	0.0	0.0	0.0	642000000.0	1.0	1.0	1
7	DDoS	DrDoS_NTP	440.0	91.33493	0.0	0.0	0.0	18795180.0	22.896551	664.0	2
8	Benign	Benign	0.0	0.0	0.0	0.0	0.0	0.0	65.0	0.0	0
9	DDoS	DrDoS_NTP	440.0	29.690578	0.0	0.0	0.0	14030942.0	31.6	1422.0	3

Figure 4.6.10: Companion Tool Website Analyzed Log File with Response

This gives the Cybersecurity Analyst the overall understanding about the log file and the system as the packet wise prediction can be seen in a very prominent manner. This can always help in future decision making and testing.

CHAPTER 5
Impact On Society, Environment And Ethical Aspects

5.1 Impact On Society

This DDoS detection study has significant societal implications that extend far outside of the digital realm if one were to truly consider the effects of it. This enlightens the cyber analyst to get a better understanding and information about the system as the system log file is being predicted in the eyes of DDoS attack. Furthermore, this innovative approach not only addresses today's cyber threats but also lays the groundwork for a stronger, safer digital future. With these services, we are ultimately providing a safer and a more friendly digital environment that our children, and children of children, will inherit and will help advance. This is a very good resource for cyber security analysts in visualizing the system logs as penetrated with DDoS or not as this can make a lot of difference in the infra and future growth as well as decision making for the organization.

5.2 Impact On Environment

Considering the collateral effects on cybersecurity when confronted with the environmental implications of a study project is of utmost significance. This is an indirect way of saving natural resources by making defenses against DDoS attacks stronger. In what way? Add on to that the energy cost of running digital infrastructure - network operations centers and data centers such as those often targeted by DDoS attacks. Rather than optimizing for the ecological footprint, increasingly data centers are designed to throw more process power and bandwidth at an assault which is energy and carbon inducing. Nevertheless, our studies mitigates the overreliance on reactive measures by detecting future DDoS attacks and defense ways with tile learning techniques, to decrease the total environmental footprints related to cybersecurity attacks. In addition, our move enables healthy growth for digital ecosystems by providing for an eco-friendly and safe online space, ensuring that technology upgrades are compatible with the principles of environmental stewardship. Therefore, it encompasses an energy saving level, ultimately affecting the world; an environmental impact that is more like an idea of responsible digital citizenship, and a preservation of both the natural and digital ecosystems living in a peaceful harmony.

5.3 Ethical Aspects

The ethical considerations in this case reveal a complicated terrain in which ethical imperatives and social norms intersect with technological progress. The ethical foundation of this work is based on the idea that we should protect access to and the integrity of digital infrastructures against malicious attackers. However, the ethical

dimensions of cybersecurity research management demand a nuanced understanding of the potential impact and moral dilemmas that can arise in the process. For instance, although it seeks to build stronger security, an issue related to ethics is the use of data where the principle is to never break the promise of maintaining high confidentiality regardless of the case. Given the fact that it involves such voluminous data, and machine learning algorithms at play, it is paramount that all the ethical norms and data protection legislation be observed. In addition, the fair distribution of cybersecurity resources and protections raise social justice and inclusion concerns. How might advanced cybersecurity measures be useful to underprivileged and marginalized groups? The answer to this moral dilemma is in democratizing cybersecurity knowledge, tools, and resources, empower people, and communities to defend successfully against cyberattacks. In addition, the ethical implications of means of offensive cybersecurity such that there can be more, the more the attack and methods to the deterrent attackers, it will be necessary to consider with due care and to achieve compliance with the international norms on cyberwarfare by law and regulations. Our attempt through moral ethics is to cross further into cybersecurity through an open, answerable with hardliner scenarios & build a digital environment residing on ethical values which benefit the community.

CHAPTER 6
Conclusion And Future Work

6.1 Summary of the Study

Through this groundbreaking study, we provide a strong prediction analogy against one of the most threatening types of malevolent attacks: the Distributed Denial of Services (DDoS), using state-of-the-art Machine Learning, complex data analysis and an uncompromising will. We conducted an extensive exploration of machine learning models that we meticulously tested for their capability in DDoS identification and classification. The models we tried were Random Forest Classifier, Decision Tree Classifier, Support Vector Classifiers, Logistic Regression, KNeighbors Classifier, Gradient Boosting Classifier, AdaBoost Classifier, Bagging Classifier, Bernoulli Naive Bayes, Gaussian Naive Bayes, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Multi-layer Perceptron Classifier. We accomplish this through an exhaustive set of procedure tests in which we adjust our parameters by hand in order to combine the strengths of multiple algorithms into a single ensemble model that can be calibrated for utmost accuracy and performance. Our analysis relies on one of the most complete and widely cited empirical datasets available, the CIC-DDoS2019 dataset. We evaluated the performance of our models with various evaluation techniques such as accuracy, confusion matrix, etc to validate our models' performance to work in a real-world scenario. In other words, instead of focusing solely on the technological capabilities of cybersecurity research, our research promotes ethical behavior, inclusion, and openness by examining the ethical implications, social impact, and environmental consequences of cybersecurity research. As we close this chapter, we take the lessons we've learned and the hopes we've sparked, working towards a digital future marked by trust, security, and resilience. The companion tool also will have a greater role in the field of cyber security as it can detect DDoS attacks in log files within milliseconds.

6.2 Conclusion

In conclusion, our research into Distributed Denial of Service (DDoS) detection has been largely successful and very informative. To find the best ensemble model to detect DDoS, we conducted experiments with multiple machine learning models including Random Forest Classifier, Bagging Classifier, Decision Tree Classifier, Gradient Boosting Classifier, and KNeighbors Classifier. By using CIC-DDoS2019 Dataset with extensive feature selection and model validation our ensemble model is proven to be excellent in detection and accuracy score for the detection is 99.81% and accuracy score of the classification is 93.92%. Our top models are Gradient Boosting

Classifier, Random Forest Classifier, Bagging Classifier, Decision Tree Classifier, KNeighbors Classifier, and tend to perform more or less great across a number of setups. Their setups use target columns which single out individual DDoS assaults reflecting everything they have learned throughout a large amount of time of trial and error. Our paper illuminates the importance of ethical, social, and environmental considerations alongside technical functions in responsible cybersecurity research. Though we are closing this chapter, we remain unwavering in our commitment to relentless transparency, fairness, and openness in our work as we go forward to a future in which digital ecosystems are robust and trusted by the people who use them. The app that predicts DDoS packets from log files also has its presence in the cyber security turf.

6.3 Recommendations

Naturally, to improve the effectiveness and resilience of DDoS detection, we urge a more indepth exploration of the family of the ensemble learning methods as indicated by the results of this study. Focus first on creating models that make sense and are transparent for trust and understanding between stakeholders, but in the meantime develop guidelines and consent to protect the privacy and rights of users. Promote collaboration among cyber-stakeholders to share expertise and best practices, encourage standardization, effective regulation and capacity building, and nurture a cyber-security culture. These recommendations can be used by stakeholders to strengthen the resilience of digital infrastructure from cyber attacks and to extend DDoS detection capabilities.

6.4 Future Work

It provides the stepping stone for further research in different directions of DDoS detection. To guarantee the performance of models in practical scenarios, one needs to examine the ability of a model for transferring over diverse network topologies and also various kinds of attacks. Future work should focus on empirical comparisons between new deep learning models and existing machine learning techniques, which may be useful to fully understand the pros and cons of different DDoS detection methods. In addition, policy makers and cybersecurity professionals can use it to analyze the socio-economic implications of DDoS attacks and the effectiveness of mitigation strategies. The ethical considerations related to model interpretability, fairness and transitivity should also be taken into account in order to build more verifiable and robust DDoS detection methods. After all, this might promote more reliable and ethically-correct cybersecurity solutions thanks to a deeper study in these areas of knowledge.

Also, it provides an opportunity to build a cyber specific dashboard for the security professional to consume the research results. In order to make source the best use offered to cyber security professionals using the trained DDoS detection models deliverable introduced in the earlier part and integrating that model with this dashboard, cybersecurity experts can possibly deploy the DDoS detection models and monitor them in real-time. For example, the dashboard could possibly display real-time warnings of potential DDoS assaults as models are being updated in real-time as network traffic data streams in. This leads to faster response and mitigation actions. Additionally, the dashboard could offer interactive tools for exploring model predictions, viewing archetypes of previous attacks, and adjusting model parameters as the threat landscape evolves. Such a technology would greatly benefit cyber security teams and enable them to more effectively fend off DDoS attacks.

APPENDIX
Course Outcomes, Complex Engineering Problems (EP) and Complex Engineering Activities (EA) Addressing

Title:
DDOS DETECTION USING ENSEMBLE TECHNIQUES IN MACHINE LEARNING MODELS WITH A TAILORED COMPANION TOOL FOR CYBER SECURITY ANALYSTS

Student ID: 201-15-13585

Course Outcomes Description

CO	CO Description
Phase - I	
CO1	Integrate recently gained and previously acquired knowledge to identify a **DDoS Detection** problem for the Final Year Design Project (FYDP)
CO2	Analyze different aspects of the goals in designing a solution for this FYDP
CO3	Explore diverse problem domains through a literature review, delineate the issues, and establish this goals for the FYDP
CO4	Perform economic evaluation and cost estimation and employ suitable project management procedures throughout the development life cycle of the FYDP
Phase - II	
CO5	Design and develop technical solutions and system components or processes that meet specified requirements, ensuring compliance with public health and safety standards, as well as considering cultural, socioeconomic, and environmental factors in this FYDP
CO6	Choose and apply appropriate methodologies, resources, and contemporary engineering and IT technologies to address complex engineering processes, encompassing prediction and modeling, while adhering to relevant constraints in this FYDP

CO7	Analyze societal, health, safety, legal, and cultural considerations, along with associated responsibilities, in the context of professional engineering practice and the resolution of this problem, employing logical reasoning guided by contextual understanding.
CO8	Comprehend and evaluate the enduring sustainability and impact of professional engineering endeavors in addressing intricate engineering challenges within social and environmental frameworks.
CO9	Implement ethical principles and adhere to professional standards and norms in this FYDP
CO10	Capable of operating proficiently both individually and as a team member or leader across diverse teams and interdisciplinary settings in this FYDP
CO11	Proficiently communicate with the engineering community and broader society regarding complex engineering endeavors, including the ability to comprehend and generate comprehensive reports and design documentation, as well as provide and receive clear instructions throughout this FYDP
CO12	Acknowledge the importance of self-directed and life-long learning within the evolving landscape of technology, and possess the readiness and capability to engage in lifelong learning endeavors

Program Outcomes Description

POs	Category	Program Outcomes
PO1	**Engineering Knowledge**	Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
PO2	**Problem Analysis**	Identify, formulate, research the literature and analyze complex engineering problems and reach substantiated conclusions using first principles of mathematics, the natural sciences and the engineering sciences.

PO3	**Design/Development of Solutions**	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety as well as cultural, societal and environmental concerns.
PO4	**Investigations**	Conduct investigations of complex problems, considering design of experiments, analysis and interpretation of data and synthesis of information to provide valid conclusions.
PO5	**Modern tool usage**	Create, select and apply appropriate techniques, resources and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
PO6	**The engineer and society**	Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to professional engineering practice.
PO7	**Environment and sustainability**	Understand the impact of professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of, and need for sustainable development.
PO8	**Ethics**	Apply ethical principles and commit to professional ethics, responsibilities and the norms of the engineering practice.
PO9	**Individual work and teamwork**	Function effectively as an individual and as a member or leader of diverse teams as well as in multidisciplinary settings.
PO10	**Communication**	Communicate effectively about complex engineering activities with the engineering community and with society at large. Be able to comprehend and write effective reports, design documentation, make effective presentations and give and receive clear instructions.

PO11	**Project management and finance**	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work as a member or a leader of a team to manage projects in multidisciplinary environments
PO12	**Life Long Learning**	Recognize the need for and have the preparation and ability to engage in independent, life-long learning in the broadest context of technological change.

CO-PO Mapping

PO's / CO's	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	✓											
CO2		✓										
CO3				✓								
CO4												
CO5			✓									
CO6					✓							
CO7												
CO8							✓					
CO9								✓				
CO10									✓			
CO11										✓		
CO12												✓

Addressing CO (1 to 8), Knowledge Profile (K), Attainment of Complex Engineering Problems (EP), and Attainment of Complex Engineering Activities (EA)

Addressing CO (1 to 8), Knowledge Profile (K), Attainment of Complex Engineering Problems (EP):

EP	EP Definition	Attainment	CO	Justification (with Knowledge Profile)
EP1	**Depth of Knowledge Required**	Yes	CO1, CO2, CO3, CO5, CO6, CO8	This project highlights **fundamental engineering (K3)** principles by using machine learning and a variety of base models to detect and classify DDoS attacks. The project demonstrates **specialist knowledge (K4)** by performing ensemble learning with various ML models such as RF, DT, SVC, LR, KNN, GBC, ABC, BC, LDA, QDA, MLPs, BNB, and GNB, thereby improving DDoS detection accuracy in data primarily monitored in log files, which is critical for computer-aided DDoS attack detection systems.
				The project uses **engineering practice & design (K5)** through the figure of the process of experimentation. The project targets **engineering practice & technology (K6)** by applying ML models such as RF, DT, SVC, LR, KNN, GBC, ABC, BC, LDA, QDA, MLPs, BNB, and GNB via ensemble learning.
				This study contributes to the **research literature (K8)** by

				integrating insights from recent studies to advance DDoS detection and classification using machine learning, demonstrating a thorough awareness of current approaches.
EP2	**Range of Conflicting Requirements**	Yes	CO2	This study addresses **EP-2** by identifying the challenges in DDoS detection, such as the complications of combining ensemble learning and machine learning. It uses comparative analysis to try to develop the best model with the highest possible accuracy.
EP3	**Depth of Analysis Required**	Yes	CO2, CO6	This study solves **EP-3** by methodically analyzing experimental results, indicating Ensemble Learning as the most significant answer for improving DDoS detection among a variety of viable methodologies.
EP4	**Familiarity of Issues**	No	CO8	N/A
EP5	**Extends of Application Code**	Yes	CO5	This project went beyond the theoretical approach and attempted to get involved at the application level, as noted in **EP-5**, which includes a specific companion tool that will help cyber security experts advance DDoS detection and categorization through the system's log files.
EP6	**Extends of Stakeholders Involved and Conflicting Requirements**	No		N/A

EP7	**Interdependence**	Yes	CO5	This project's comprehensive strategy addresses high-level issues by integrating numerous components such as data collecting, statistical analysis, and proposed methodology, resulting in a holistic solution to challenging obstacles in medical diagnostics and assuring **EP-7**.

Addressing CO11 with Complex Engineering Activities (EA):

EA	EA Definition	Attainment	CO	Justification
EA1	**Range of Resources**	Yes	**CO11**	Our study makes use of a variety of resources, including high-performance computing infrastructure, GPUs, large datasets, and ethical considerations, to assure systematic research and contribute to advances in DDoS detection using ensemble learning on machine learning base models.
EA2	**Level of Interaction**	No		N/A
EA3	**Innovation**	No		N/A
EA4	**Consequences for society and the environment**	Yes		This project benefits society by enhancing cyber-security through DDoS detection methods, while simultaneously encouraging environmental sustainability by using efficient computational resources and detecting attacks in system logs.
EA5	**Familiarity**	No		N/A

Addressing CO (CO4, CO7, CO9, CO10, CO12):

SN	COs	Attainment	Justification
1	CO4	NO	N/A
2	CO7	NO	N/A
3	CO9	Yes	The project demonstrates adherence to ethical principles by prioritizing patient privacy, obtaining informed consent, and transparently documenting the research process. It also ensures responsible knowledge dissemination and societal well-being through the ethical use of advanced cyber-security technologies that comply with **CO9**.
4	CO10	Yes	This project was completed as individual labor as a solo player, which resulted in the understanding of many concepts and interdisciplinary setups that ultimately fulfilled the **CO10**.
5	CO12	Yes	The project's commitment to continuous learning **(CO12)** and adaptation within the dynamic technological landscape is reflected in its comprehensive data collection, rigorous statistical analysis, meticulous methodology development, and thorough experimental results and analysis, demonstrating a commitment to staying current and refining techniques to address modern challenges.

REFERENCES

[1] K. Li, H. Zhou, Z. Tu, W. Wang, and H. Zhang, "Distributed network intrusion detection system in satellite-terrestrial integrated networks using federated learning," IEEE Access, vol. 8, pp. 214852-214865, 2020.

[2] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," IEEE Access, vol. 7, pp. 41525-41550, 2019.

[3] Y. Fu, F. Lou, F. Meng, Z. Tian, H. Zhang, and F. Jiang, "An intelligent network attack detection method based on RNN," in Proc. 2018 IEEE Third Int. Conf. Data Sci. Cyberspace (DSC), pp. 483-489, 2018.

[4] S. S. Priya, M. Sivaram, D. Yuvaraj, and A. Jayanthiladevi, "Machine learning based DDoS detection," in Proc. 2020 Int. Conf. Emerging Smart Comput. Informat. (ESCI), pp. 234-237, 2020.

[5] S. N. Shiaeles, V. Katos, A. S. Karakos, and B. K. Papadopoulos, "Real time DDoS detection using fuzzy estimators," Computers & Security, vol. 31, no. 6, pp. 782-790, 2012.

[6] O. Rahman, M. A. G. Quraishi, and C.-H. Lung, "DDoS attacks detection and mitigation in SDN using machine learning," in Proc. 2019 IEEE World Congr. Services (SERVICES), vol. 2642, pp. 184-189, 2019.

[7] S. Sarraf, "Analysis and detection of DDoS attacks using machine learning techniques," Am. Sci. Res. J. Eng. Technol. Sci., vol. 66, no. 1, pp. 95-104, 2020.

[8] M. Idhammad, K. Afdel, and M. Belouch, "Semi-supervised machine learning approach for DDoS detection," Applied Intelligence, vol. 48, pp. 3193-3208, 2018.

[9] C. Wu, S. Sheng, and X. Dong, "Research on visualization systems for DDoS attack detection," in Proc. 2018 IEEE Int. Conf. Syst., Man, Cybern. (SMC), pp. 2986-2991, 2018.

[10] H. Choi, H. Lee, and H. Kim, "Fast detection and visualization of network attacks on parallel coordinates," Computers & Security, vol. 28, no. 5, pp. 276-288, 2009.

[11] R. Santos, D. Souza, W. Santo, A. Ribeiro, and E. Moreno, "Machine learning algorithms to detect DDoS attacks in SDN," Concurrency Comput.: Pract. Exper., vol. 32, no. 16, p. e5402, 2020.

[12] F. Shaar and A. Efe, "DDoS attacks and impacts on various cloud computing components," Int. J. Inf. Security Sci., vol. 7, no. 1, pp. 26-48, 2018.

[13] M. Sachdeva, G. Singh, K. Kumar, and K. Singh, "DDoS Incidents and their Impact: A Review," Int. Arab J. Inf. Technol., vol. 7, no. 1, pp. 14-20, 2010.

[14] N. Hoque, D. K. Bhattacharyya, and J. K. Kalita, "Botnet in DDoS attacks: trends and challenges," IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2242-2270, 2015.

[15] E. Navruzov and A. Kabulov, "Detection and analysis types of DDoS attack," in Proc. 2022 IEEE Int. IOT, Electron. Mechatronics Conf. (IEMTRONICS), pp. 1-7, 2022.

[16] I. Sofi, A. Mahajan, and V. Mansotra, "Machine learning techniques used for the detection and analysis of modern types of DDoS attacks," Int. Res. J. Eng. Technol., vol. 4, no. 6, pp. 1085-1092, 2017.

[17] F. Mansmann, "Visual analysis of network traffic: Interactive monitoring, detection, and interpretation of security threats," Ph.D. dissertation, 2008.

[18] O. Bamasag, A. Alsaeedi, A. Munshi, D. Alghazzawi, S. Alshehri, and A. Jamjoom, "Real-time DDoS flood attack monitoring and detection (RT-AMD) model for cloud computing," PeerJ Comput. Sci., vol. 7, p. e814, 2022.

[19] M. Marvi, A. Arfeen, and R. Uddin, "A generalized machine learning-based model for the detection of DDoS attacks," Int. J. Network Manage., vol. 31, no. 6, p. e2152, 2021.

[20] A. Aldweesh, A. Derhab, and A. Z. Emam, "Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," Knowl.-Based Syst., vol. 189, p. 105124, 2020.
[21] M. Masdari and H. Khezri, "A survey and taxonomy of the fuzzy signature-based intrusion detection systems," Appl. Soft Comput., vol. 92, p. 106301, 2020.
[22] N. M. Yungaicela-Naula, C. Vargas-Rosales, and J. A. Perez-Diaz, "SDN-based architecture for transport and application layer DDoS attack detection by using machine and deep learning," IEEE Access, vol. 9, pp. 108495-108512, 2021.
[23] P. L. Indrasiri, E. Lee, V. Rupapara, F. Rustam, and I. Ashraf, "Malicious traffic detection in IoT and local networks using stacked ensemble classifier," Comput., Mater. Contin., vol. 71, no. 1, pp. 489-515, 2022.
[24] M. S. Potnis, S. K. Sathe, P. G. Tugaonkar, G. L. Kulkarni, and S. S. Deshpande, "Hybrid intrusion detection system for detecting DDoS attacks on web applications using machine learning," in ICT Anal. Appl., Singapore: Springer Nature Singapore, pp. 797-805, 2022.
[25] D. V. V. S. Manikumar and B. U. Maheswari, "Blockchain based DDoS mitigation using machine learning techniques," in Proc. 2nd Int. Conf. Inventive Res. Comput. Appl. (ICIRCA), Jul. 2020, pp. 794-800.
[26] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proenca, "Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment," IEEE Access, vol. 8, pp. 83765-83781, 2020.
[27] J. P. A. Maranhão, J. P. C. L. da Costa, E. P. de Freitas, E. Javidi, and R. T. de Sousa Júnior, "Error-robust distributed denial of service attack detection based on an average common feature extraction technique," Sensors, vol. 20, no. 20, p. 5845, Oct. 2020.
[28] A. Chartuni and J. Márquez, "Multi-classifier of DDoS attacks in computer networks built on neural networks," Appl. Sci., vol. 11, no. 22, p. 10609, 2021.
[29] D. Alghazzawi, O. Bamasag, H. Ullah, and M. Z. Asghar, "Efficient detection of DDoS attacks using a hybrid deep learning model with improved feature selection," Appl. Sci., vol. 11, no. 24, p. 11634, 2021.
[30] M. A. Haq and M. A. R. Khan, "DNNBoT: Deep neural network-based botnet detection and classification," Comput., Mater. Contin., vol. 71, no. 1, pp. 1-18, 2022.
[31] D. Kumar, R. K. Pateriya, R. K. Gupta, V. Dehalwar, and A. Sharma, "DDoS detection using deep learning," Procedia Comput. Sci., vol. 218, pp. 2420-2429, 2023.
[32] Learn about React, available at <<https://react.dev/>>, last accessed on 26-06-2024 at 12:00 PM.
[33] Explore Tailwind CSS, available at <<https://tailwindcss.com/>>, last accessed on 26-06-2024 at 12:02 PM.
[34] Discover Shadcn UI, available at <<https://ui.shadcn.com/>>, last accessed on 26-06-2024 at 12:03 PM.
[35] Learn about JavaScript, available at <<https://en.wikipedia.org/wiki/JavaScript>>, last accessed on 26-06-2024 at 12:05 PM.
[36] Visit the Python homepage, available at <<https://www.python.org/>>, last accessed on 26-06-2024 at 12:06 PM.
[37] Discover FastAPI, available at <<https://fastapi.tiangolo.com/>>, last accessed on 26-06-2024 at 12:08 PM.
[38] Explore Vercel's services, available at <<https://vercel.com/>>, last accessed on 26-06-2024 at 12:09 PM.
[39] Access GitHub, available at <<https://github.com/>>, last accessed on 26-06-2024 at 12:11 PM.

[40] Learn about Render, available at <<https://render.com/>>, last accessed on 26-06-2024 at 12:12 PM.
[41] Visit the DDoS Point Frontend, available at <<https://ddos-point-fe.vercel.app/>>, last accessed on 26-06-2024 at 12:14 PM.
[42] View the DDoS taxonomy image, available at <<https://www.unb.ca/cic/_assets/images/ddostaxonomy.png>>, last accessed on 26-06-2024 at 12:15 PM.
[43] Access the DDoS dataset on Mendeley, available at <<https://data.mendeley.com/datasets/ssnc74xm6r/1>>, last accessed on 26-06-2024 at 12:17 PM.
[44] Visit the Canadian Institute for Cybersecurity, available at <<https://www.unb.ca/cic/>>, last accessed on 26-06-2024 at 12:18 PM.
[45] Explore the CICDDoS2019 dataset, available at <<https://www.unb.ca/cic/datasets/ddos-2019.html>>, last accessed on 26-06-2024 at 12:20 PM.

DDOS DETECTION USING ENSEMBLE TECHNIQUES IN MACHINE LEARNING MODELS WITH A TAILORED COMPANION TOOL FOR CYBER SECURITY ANALYSTS

ORIGINALITY REPORT

23%	18%	15%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
2	Submitted to Daffodil International University Student Paper	2%
3	Submitted to Montclair State University Student Paper	1%
4	Rocha, Silvia Jorge Moreira da. "Human Partner Understanding: Recognition and Prediction of Human Action Sequences for Safe Human-Robot Collaboration", Universidade do Porto (Portugal), 2024 Publication	1%
5	www.techscience.com Internet Source	1%
6	Submitted to University of Mines and Technology Student Paper	1%