ENHANCING THE PREDICTION OF DDoS ATTACKS IN IoT NETWORKS USING CONTEXT CORRELATION AWARE MODEL

A PROJECT REPORT

Submitted by

SWATI KUMARI - RA2011030020002
RACHARLA RAKESH BABU - RA2011030020006
HITESH BORHA - RA2011030020081

Under the guidance of
Dr. S. SATHYA PRIYA
(Associate Professor, Department of Computer Science and Engineering)

in partial fulfilment for the award of the degree
of
BACHELOR OF TECHNOLOGY
of

COMPUTER SCIENCE AND ENGINEERING WITH SPECIALIZATION IN CYBER SECURITY

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
RAMAPURAM, CHENNAI -600089

MAY 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Deemed to be University U/S 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled **"ENHANCING THE PREDICTION OF DDoS ATTACK IN IoT NETWORK USING CONTEXT CORRELATION AWARE MODEL"** is the bonafide work of **SWATI KUMARI [REG NO: RA2011030020002], RACHARLA RAKESH BABU [REG NO: RA2011030020006], HITESH BORHA [REG NO: RA2011030020081]** who carried out the project work under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an occasion on this or any other candidate.

SIGNATURE

Dr. S. SATHYA PRIYA,

Associate Professor

Computer Science and Engineering,
SRM Institute of Science and Technology,

Ramapuram, Chennai.

SIGNATURE

Dr. K. RAJA, M.E., Ph.D.,

Professor and Head

Computer Science and Engineering,
SRM Institute of Science and Technology,

Ramapuram, Chennai.

Submitted for the Viva Voce Examination held on ……..... at SRM Institute of Science and Technology, Ramapuram Campus, Chennai -600089.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
RAMAPURAM, CHENNAI - 89
DECLARATION

We hereby declare that the entire work contained in this project report titled "**ENHANCING THE PREDICTION OF DDoS ATTACK IN IoT NETWORK USING CONTEXT CORRELATION AWARE MODEL**" has been carried out by **SWATI KUMARI** [REG NO: RA2011030020002], **RACHARLA RAKESH BABU** [REG NO: RA2011030020006], **HITESH BORHA** [REG NO: RA2011030020081] at SRM Institute of Science and Technology, Ramapuram, Chennai- 600089, under the guidance of **Dr. S. SATHYA PRIYA, ASSOCIATE PROFESSOR**, Department of Computer Science and Engineering.

Place: Chennai
Date:

SWATI KUMARI

RACHARLA RAKESH BABU

HITESH BORHA

Department of Computer Science and Engineering

SRM Institute of Science and Technology

Own Work Declaration form

To be completed by the student for all assessments

Degree/ Course : B. Tech Computer Science and Engineering with specialization in Cyber Security

Student Name : SWATI KUMARI, RACHARLA RAKESH BABU, HITESH BORHA

Registration Number : RA2011030020002, RA2011030020006, RA2011030020081

Title of Work : Enhancing the Prediction of DDoS Attacks in IoT using Context Correlation Aware Model

I / We hereby certify that this assessment complies with the University's Rules and Regulations relating to Academic misconduct and plagiarism, as listed in the University Website, Regulations, and the Education Committee guidelines.

I / We confirm that all the work contained in this assessment is my / our own except where indicated, and that I / We have met the following conditions:

- Clearly references / listed all sources as appropriate
- Referenced and put in inverted commas all quoted text (from books, web, etc.)
- Given the sources of all pictures, data etc. that are not my own
- Not made any use of the report(s) or essay(s) of any other student(s) either past or present
- Acknowledged in appropriate places any help that I have received from others (e.g., fellow students, technicians, statisticians, external sources)
- Compiled with any other plagiarism criteria specified in the Course handbook / University website

I understand that any false claim for this work will be penalized in accordance with the University policies and regulations.

DECLARATION:
I am aware of and understand the University's policy on Academic misconduct and plagiarism and I certify that this assessment is my / our own work, except were indicated by referring, and that I have followed the good academic practices noted above.
If you are working in a group, please write your registration numbers and sign with the date for every student in your group. RA2011030020002 RA2011030020006 RA2011030020081

ACKNOWLEDGEMENT

We place on record our deep sense of gratitude to our lionized Chairman **Dr.R.SHIVAKUMAR** for providing us with the requisite infrastructure throughout the course.

We take the opportunity to extend our hearty and sincere thanks to our respected Dean, **Dr. M.MURALI KRISHNA, B.E., M.Tech., Ph.D. MISTE,FIE,C.Engg**., for manoeuvring us into accomplishing the project.

We take the privilege to extend our hearty and sincere gratitude to the Professor and Head of the Department, **Dr.K.RAJA,M.E., Ph.D**., for his suggestions, support and encouragement towards the completion of the project with perfection.

We express our hearty and sincere thanks to our guide **Dr. S. SATHYA PRIYA**, **Associate Professor**, Computer Science and Engineering Department for her encouragement, consecutive criticism and constant guidance throughout this project work.

Our thanks to the teaching and non-teaching staff of the Computer Science and Engineering Department of SRM Institute of Science and Technology, Ramapuram Campus, for providing necessary resources for our project.

SWATI KUMARI

RACHARLA RAKESH BABU

HITESH BORHA

ABSTRACT

With the widespread integration of Internet of Things (IoT) devices into various domains, ensuring their security has become paramount. Unfortunately, this surge in connectivity has also opened avenues for malicious activities such as Distributed Denial of Service (DDoS) attacks targeting IoT networks. These attacks can disrupt critical services and compromise sensitive data. In response to this growing threat, various methods have been developed to detect and mitigate such attacks. Existing approaches typically rely on signature-based detection, anomaly detection, or machine learning algorithms to identify suspicious activities within IoT networks. However, these methods often struggle to adapt to the dynamic and heterogeneous nature of IoT environments. Moreover, they may generate a high rate of false positives or fail to detect sophisticated attacks. To address these limitations, we propose a novel Context-Correlation-Aware Model for enhancing the prediction of DDoS attacks in IoT networks. Our approach leverages contextual information and correlations between different IoT devices and their behaviors to improve the accuracy and efficiency of attack detection. By considering the unique characteristics and relationships within IoT ecosystems, our model aims to provide more robust protection against emerging threats. Through extensive experimentation and evaluation, we demonstrate the effectiveness and scalability of our proposed method in safeguarding IoT networks against DDoS attacks.

TABLE OF CONTENTS

FIG. NO. **LIST OF FIGURES** PAGE NO.

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT	**Internet of Things**
DDoS	**Distributed Denial of Service**
IP	**Internet Protocol**
RF	**Random Forest**
CCLA	**Context Co-relation Learning Algorithm**
SDT	**Software Defined Networking**
MTD	**Moving Objective Guard**
DNS	**Domain Name Service**
RPL	**Routing Protocol for Low-Power and Lossy Networks**
TP	**True Positive**
FP	**False Positive**
TN	**True Negative**
FN	**False Negative**

Chapter 1
INTRODUCTION

The Internet of Things (IoT) is the organization of real objects, devices, vehicles, embedded software, sensors and electronics in buildings and connecting other objects to a network, enabling the exchange and collection of these objects. The IoT enables objects to be identified and controlled far beyond the existing organizational framework, opening the door to easier coordination between real, computer-based frameworks as well, and improving expertise, accuracy and financial benefits as the IoT expands. . with sensors and actuators, innovation becomes a broader category of digital reality frameworks that also include technologies such as smart cities, smart networks, smart homes and smart transportation. Everything is exceptionally recognizable with a built-in registration framework, but can work together within an existing website foundation. With this amazing array of IoT devices, the risk of cyber-attacks is greater, and IoT devices can feed them either intentionally or unintentionally. Among the many attacks associated with IoT devices is a type of attack known as Distributed Denial of Service (DDoS) attacks.

1.1 Overview

DDoS attacks are one of the main threats to the security of IoT organizations. In this attack, the attacker uses many compromised nodes to overload the target, generating a lot of network traffic that consumes the target's resources. This ultimately destroys the framework, prevents benefits and prevents approved clients from accessing the relevant administrations. Each attack machine can behave more stealthily, making it harder to track and disable, and multiple attack machines can generate more attack traffic than a single machine. Multiple attack planes are also harder to shut down than one attack plane. Because the incoming traffic that floods the victim comes from a variety of sources, it can be difficult to effectively stop the abuse with access control. It also makes it difficult to distinguish between genuine customer traffic and attack traffic when it splits into different origins. As another alternative or extension to DDoS, attacks can involve changing Internet Protocol (IP) source addresses (IP address-speak), which creates confusion in distinguishing and countering the attack.

The existing system for detecting DDoS attacks relies primarily on flow-based monitoring, signature-based detection, and threshold-based detection mechanisms. Flow-based monitoring analyzes network traffic flows to identify anomalies, while signature-

based detection matches incoming traffic against known patterns of DDoS attacks. Threshold-based detection sets limits on various network parameters and triggers alerts when these limits are exceeded. While these methods have been effective to some extent, they have limitations in coping with the increasing sophistication and scale of DDoS attacks. They may struggle to detect new and evolving attack patterns and can generate false positives or miss subtle attacks, leading to degraded network performance and potential service disruptions. Additionally, the existing system lack scalability and agility in responding to rapidly changing attack vectors.

The proposed system leverages the Random Forest algorithm (RF) and Context Correlation Learning Algorithm (CCLA) to enhance DDoS detection capabilities. The Random Forest algorithm, a powerful machine learning technique, will analyze network traffic features to segregate incoming data packets as normal or malicious with high accuracy. By utilizing an ensemble of decision trees, Random Forest can effectively handle large volumes of data and adapt to changing attack patterns. Additionally, the Context Correlation Algorithm will be employed to analyze the correlation between various network events and identify suspicious patterns indicative of DDoS attacks. This algorithm will consider contextual information such as the timing, source, and destination of network packets to detect coordinated attack behaviors that may go unnoticed by traditional detection methods. By integrating these advanced algorithms, the proposed system aims to improve detection of accuracy, reduce the false positive rates, and enhance the overall resilience of the network against DDoS attacks.

In conclusion, while the existing system for detecting DDoS attacks has provided some level of protection, it faces challenges in keeping pace with the evolving landscape of cyber threats. Therefore, the proposed system, integrating advanced detection techniques and leveraging cloud-based mitigation services, offers a promising solution to enhance the network's resilience against DDoS attacks. By combining these approaches, organizations can better safeguard their networks, minimize disruptions, and ensure the uninterrupted delivery of services to their users.

1.2 Problem statement

Existing DDoS detection systems face limitations in keeping up with evolving attack techniques. These systems often generate false positives or miss subtle attack patterns, leading to network performance degradation. The proposed approach integrates the Random Forest algorithm and Context Correlation Algorithm. To generate a Random Forest algorithm that analyzes the network traffic features to segregate packets as normal or malicious with high accuracy. To generate a Context Correlation Algorithm that identifies coordinated attack behaviors by analyzing contextual information, enhancing the overall resilience of the network against DDoS attacks.

1.3 Objective

- It is to develop an enhanced algorithm specifically tailored for detecting and classifying DDoS attacks in IoT environments.
- To explore a wider search space more efficiently, enabling quicker identification of potential threats.
- To employ advanced techniques to select optimal features from the dataset, enhancing the accuracy of detection.
- To analyzing large volumes of data rapidly, effectively, facilitate faster and more efficient threat detection, thereby minimizing response times.
- To focus on learning normal traffic patterns, automatically detecting anomalous patterns, and identifying DDoS attacks in real-time, ensuring robust security measures in IoT environments**.**

1.4 Project Domain

It encompasses cybersecurity in Internet of Things environments, focusing on the development of advanced algorithms and techniques for detecting and mitigating DDoS) attacks. This domain involves the exploration and optimization of algorithms to effectively analyze large volumes of IoT network traffic, identify anomalous patterns, and automatically classify and respond to DDoS attacks. By enhancing the security posture of IoT systems through robust DDoS detection mechanisms, the project seeks to safeguard critical infrastructure, protect sensitive data, and ensure the reliable and secure operation of IoT devices and services in the face of evolving cyber threats.

1.5 Scope

The future scope of the proposed work lies at the nexus of IoT, cybersecurity, and machine learning, offering avenues for advancement. Research aims to refine algorithms for detection and mitigation of DDoS attacks in IoT ecosystems using cutting-edge machine learning techniques. These efforts involve exploring emerging algorithms to enhance real-time threat detection, fortifying the security of interconnected IoT devices and networks. Furthermore, the algorithms' applicability can extend to diverse IoT domains like industrial IoT, smart cities, and healthcare, adapting to sector-specific requirements. Integrating proactive defense mechanisms such as anomaly detection and threat intelligence sharing can bolster IoT network resilience.

1.6 Methodology

It involves a multi-faceted approach leveraging both the Random Forest algorithm and Context Correlation technique to enhance DDoS detection in IoT environments. Initially, a comprehensive dataset of network traffic features is collected from IoT devices. This dataset is preprocessed to remove noise and irrelevant features, ensuring data quality for subsequent analysis. Next, the Random Forest algorithm is employed to train a robust classification model. Utilizing the ensemble learning approach of Random Forest, multiple decision trees are trained on the dataset to effectively classify network traffic as normal or malicious. This model is trained iteratively, optimizing hyperparameters to improve classification accuracy. Simultaneously, the Context Correlation technique is applied to analyze the contextual information of network events. By considering factors such as timing, source, and destination of network packets, the Context Correlation algorithm identifies patterns indicative of coordinated attack behaviors. Finally, the outputs from both the Random Forest classifier and Context Correlation analysis are integrated. This fusion approach combines the strengths of both techniques, enhancing the overall detection capabilities and providing a more comprehensive defense against DDoS attacks in IoT environments.

1.7 Organization of the Report

Chapter 1 Discusses about the introduction of the project along with the problem statement,

objective, scope and methodology.

Chapter 2 Explains the Literature review of the report.

Chapter 3 Elaborates about the description of the project with the explanation of existing and proposed system, their feasibility study and the hardware, software requirements.

Chapter 4 Discusses about the proposed work along with the required diagram models and the module description.

Chapter 5 Explains the implementation and testing of the report.

Chapter 6 Elaborates the result and the discussion of the project.

Chapter 7 Discusses about the conclusion and future enhancement of the project.

Chapter 8 Source Code

Chapter 2
LITERATURE REVIEW

2.1 Automating Mitigation of Amplification Attacks

The paper "Automating Mitigation of Amplification Attacks in NFV Services" [10] presented a mix of virtualization procedures with slim figuring and stockpiling assets permits the launch of Virtual Organization Capabilities all through the organization foundation, which gets greater spryness the turn of events and activity of organization administrations. Next to sending and steering, this can be likewise utilized for extra capabilities, e.g., for security purposes. Implementing real-time solutions poses a significant challenge due to several drawbacks. Firstly, real-time implementation encounters limitations that hinder its seamless integration into systems or processes, the process of updating messages in real-time can be tedious and time-consuming, requiring constant monitoring and adjustment and the thorough investigation of real-time solutions is often lacking, leaving potential gaps in understanding their full implications and effectiveness. These drawbacks underscore the complexities involved in realizing real-time implementations and highlight the importance of comprehensive research and development efforts in this domain.

2.2 Detection and mitigation of DoS attack

The paper "Detection and Mitigation of Low-Rate Denial-of-Service Attacks: A Survey" [9] describes the potential for being the objective of Refusal of Administration (DoS) assaults is one of the most serious security dangers on the Web. Assailants have been altering their assault design throughout the long term, harming explicit states of working frameworks and conventions trying to deny or lessen the nature of the assistance gave to real clients. These days, assaults are stealthier and impersonate genuine client traffic so that location components against High-rate DoS assaults are presently not adequate. Implementing certain solutions faces critical design challenges, which can significantly impede progress. Additionally, the lack of thorough investigation leaves potential risks unaddressed, heightening uncertainty. Moreover, the construction process may demand substantial time and economic resources, amplifying the overall burden on the project's feasibility and success.

2.3 Intrusion Detection

The paper "Internet of Things Applications, Security Challenges, Attacks, Intrusion Detection, and Future Visions: A Systematic Review"[8] by Nivedita Mishra and Sharnil Pandya explains IoT innovation is succeeding and entering all aspects of our lives, be it training, home, vehicles, or medical care. With the expansion in the quantity of associated gadgets, a few difficulties are likewise thinking of IoT innovation: heterogeneity, versatility, nature of administration, security prerequisites, and some more. Security the board takes a secondary lounge in IoT in light of cost, size, and power. It represents a huge gamble as absence of safety makes clients suspicious towards utilizing IoT gadgets. Obtaining optimal performance poses challenges due to several drawbacks. Large payloads contribute to increased complexity and resource demands. Moreover, navigating the intricacies of less commonly used methods further complicates the pursuit of efficiency.

2.4 Classification and Mitigation for DDoS Attacks

The author Marinos Dimolianis and et.al “Signature-Based Traffic Classification and Mitigation for DDoS Attacks Using Programmable Network Data Planes" [7] talks about DDoS attacks mitigation commonly depends on source IP-based separating rules; these may introduce scaling issues due to the tremendous measure of involved sources. On the other hand, we propose a source IP-rationalist DDoS traffic characterization and sifting blueprint that recognizes noxious parcel marks through directed AI strategies and thusly creates signature-based separating rules. This method presents usability challenges, lacking user-friendliness and accessibility. Furthermore, additional configuration steps are necessary, adding complexity to the implementation process. Moreover, real-time integration is unattainable due to inherent limitations, further hindering its practical application.

2.5 Distributed Denial of Service Attack Protection

The paper "Towards Crossfire Distributed Denial of Service Attack Protection Using Intent-Based Moving Target Defense Over Software-Defined Networking" [6] explains Crossfire is an aberrant objective region interface flooding Appropriated Disavowal of Administration not set in stone to influence the neighbors of the genuine objective. Presently, Crossfire DDoS assaults are obtaining stimulus in light of their lack of definition and imperceptibility. SDN (Software Defined Networking) is an advancing method in view of its versatility and programmability. Moving Target Defense (MTD) is an emerging security methodology to counter goes after by continuously changing attacked plane. The installation and maintenance of this solution entail high levels of complexity, demanding significant

expertise and effort. Additionally, its implementation often leads to poor application performance, detracting from overall user experience. Furthermore, the solution's heavyweight nature exacerbates resource consumption, further complicating its integration into existing systems.

2.6 Detection Techniques of Distributed Denial of Service Attacks

The paper " Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking ControllerA Review " [5] talks about the wide multiplication of telecom advances somewhat recently brings about the quantity of more modern security dangers. Programming Characterized Systems administration (SDN) is a new systems administration engineering that disconnects the organization control plane from the information plane that unexpectedly gives better elements and functionalities to recognize and manage those security dangers. Its versatile programmable component grants productive organization the board and gives network administrators the adaptability to screen and calibrate their organization. These solutions typically exhibit high polynomial running times, causing performance bottlenecks and inefficiencies. Moreover, their implementation in real-time scenarios is unfeasible due to computational limitations, limiting their practical utility. Additionally, the complexity involved makes them challenging to maintain, necessitating continual attention and resources for upkeep.

2.7 Distributed Denial of Service (DDoS) Attacks in SDN and Cloud Computing Environments

The author Shi Dong and et.al. " A Survey on Distributed Denial of Service (DDoS) Attacks in SDN and Cloud Computing Environments " talks about Recently, software defined networks (SDNs) and cloud computing have been widely adopted by researchers and industry. However, widespread acceptance of the networking paradigms has been tampered by the security threats. Advances in the processing technologies have helped attackers in increasing the attacks too, for instance, the development of Denial of Service (DoS) attacks to distributed DoS (DDoS) attacks which are seldom identified by conventional firewalls. These solutions are characterized by their heavyweight nature, imposing significant resource demands and potentially straining system capabilities. Additionally, their approach tends to be time-consuming, requiring extensive effort and investment to yield results. Moreover, despite efforts, these solutions have been shown to be

ineffective, failing to deliver the desired outcomes or solve the underlying problems effectively.

2.8 Data Collection and Data Analytics

The paper " Security Data Collection and Data Analytics in the Internet: A Survey " [3] explains about Assaults over the Web are turning out to be increasingly perplexing and modern. The most effective method to distinguish security dangers and measure the security of the Web emerges a critical examination point. For distinguishing the Web assaults and estimating its security, gathering various classes of information and utilizing strategies for information examination are fundamental. The computation burden associated with this approach may restrict its applicability in real-world scenarios, limiting its practical implementation. Moreover, a lack of thorough investigation leaves uncertainties regarding its efficacy and potential drawbacks unresolved. Furthermore, its complexity and less common usage make it challenging to adopt and integrate into existing systems or processes.

2.9 Domain Name Servers Against DDoS Attacks

The paper " DNS-ADVP: A Machine Learning Anomaly Detection and Visual Platform to Protect Top-Level Domain Name Servers Against DDoS Attacks " [2] talks about DNS (Domain Name Service)DDoS assaults may seriously influence the activity of PC organizations, inciting the requirement for techniques ready to ideal recognize them, and afterward to apply relief countermeasures. Visual models have been utilized to distinguish a continuous DDoS assault, yet frequently request constant consideration from IT staff. Once configured, this solution lacks flexibility, rendering it unable to adapt to evolving needs or requirements, posing a significant drawback. Moreover, its unsuitability for large-scale scenarios restricts its applicability, limiting its effectiveness in addressing broader challenges. Additionally, the solution encounters critical design challenges, complicating its implementation and potentially undermining its overall viability and success.

2.10 Internet DDoS Mitigation

The paper “Understanding Internet DDoS Mitigation from Academic and Industrial Perspectives" [1] explains how Defending against distributed denial of service (DDoS) attacks in the Internet is a fundamental problem. One practical approach to addressing DDoS assaults is to divert all objective to an outsider, DDoS insurance as-a-specialist organization which is very much provisioned and outfitted with restrictive separating components to eliminate assault traffic prior to passing the leftover traffic to the objective. The high

complexity involved in both installation and maintenance poses significant hurdles, demanding extensive expertise and resources. Additionally, its lack of user-friendliness further exacerbates usability issues, making it cumbersome to navigate and operate effectively. Furthermore, its susceptibility to errors introduces risks and uncertainties, potentially leading to system malfunctions or inaccuracies in outcomes.

2.11 Summary

The Published articles and surveys authored by different individuals delve into the complex field of DDoS Attack, with the aim of tackling the changing challenges. Different Traditional and DDoS detection Methods constitute the detection of attacks. Also, the different analysis methods used constitute the importance of Network Traffic Classification. The importance of DDoS attacks over the network and the methods detecting, preventing attack explains the importance and intensity of the attack in the modern world

Chapter 3
PROJECT DESCRIPTION

In modern applications such as tele health, intelligent transport, and autonomous agriculture, the secure routing of data collected and exchanged is paramount. However, achieving this security becomes particularly challenging in resource-constrained environments, where battery-powered IoT devices operate. The Routing Protocol for Low-Power and Lossy Networks (RPL) has been a cornerstone in such environments, but it has faced significant research challenges since its inception, particularly in mitigating energy consumption attacks.

3.1 Existing system

These existing models dive into the impact of two prominent power consumption attacks, namely hello flooding and version number change attacks, on the RPL protocol. The study aims to evaluate the effects of these attacks and propose a novel mitigation solution based on behavioral trust. Through extensive simulations using Cooja sensor nodes, the research team investigates how these attacks affect energy consumption, especially as the number of nodes in the network increases.

The findings reveal intriguing insights into the nature of these attacks. Hello flooding attacks are found to localize energy consumption to nodes in close proximity to the attacker, while version number change attacks exhibit a more global impact, affecting the entire network. Building on these observations, the paper presents a trust-based solution designed to mitigate both types of attacks effectively.

One significant aspect of the study is its acknowledgment of limitations and avenues for improvement. For instance, the researchers identify challenges such as node identity changes and the decay of trust over time. These challenges highlight the need for further research to refine trust models and address evolving threats in IoT environments.

The research underscores the critical importance of securing data routing in IoT applications and demonstrates the efficacy of trust-based solutions in mitigating energy consumption attacks on the RPL protocol. The findings not only contribute to enhancing the

security of IoT networks but also pave the way for future research endeavors aimed at exploring additional attack vectors and refining mitigation strategies.

3.2 Proposed System

Certainly, DDoS attacks pose a significant threat to network security. Attackers overwhelm systems with floods of traffic, making them unavailable to legitimate users. Current DDoS detection systems struggle to keep pace with ever-evolving attack techniques. These systems often rely on signature-based detection, which can only identify known attack patterns. As attackers develop new methods, these systems fail to detect them, leading to missed attacks and service disruptions. Additionally, existing systems can misinterpret normal traffic fluctuations as attacks, resulting in wasted resources and unnecessary disruptions.

This proposed system tackles these limitations by combining the strengths of machine learning and contextual analysis. The Random Forest algorithm, a machine learning technique, plays a crucial role. By analyzing vast amounts of historical traffic data, both normal and attack-related, the Random Forest learns to identify patterns that differentiate between regular activity and malicious attacks. This empowers the system to detect even novel attack methods that haven't been encountered before.

However, the system goes beyond just analyzing individual data packets. The Context Correlation Algorithm considers additional contextual information to build a more comprehensive picture of network activity. Imagine traffic originating from unusual geographical locations, sudden spikes in traffic volume, or coordinated attack patterns from multiple sources. By correlating these contextual factors, the system can effectively identify complex attack behaviors that might otherwise slip through the cracks.

This two-pronged approach offers several advantages. By combining machine learning and context correlation, the system gains a more precise understanding of network traffic, significantly reducing false positives and negatives. The machine learning aspect allows the system to adapt to new attack techniques as the Random Forest algorithm continuously learns from new data. This adaptability is crucial in the face of constantly evolving threats. With a more robust and accurate detection system, networks become more

resilient to DDoS attacks. This translates to uninterrupted service delivery for users, minimizing disruptions caused by malicious attacks.

In essence, this proposed system presents a promising solution for DDoS detection. By leveraging the power of machine learning and contextual analysis, it can significantly improve detection accuracy, adaptability, and overall network resilience against DDoS attacks, ensuring a more secure and reliable online experience for users.

3.2.1 Advantages

- Intended to be profoundly versatile, and can be changed or tweaked to meet the particular necessities of a specific application.
- Can gain from a bigger and more different dataset, which might be especially valuable in situations where named information is scarce or hard to get.
- Can adapt to new sub type assault situations and reveal stowed away assault designs from complex organization conditions.
- Further develops the DDoS assault discovery exactness and diminish the bogus positive rate.
- Best in terms of detection accuracy, computational efficiency, scalability, and adaptability to evolving threats

3.3 Feasibility Study

A Feasibility study is carried out to check the viability of the project and to analyze the strengths and weaknesses of the proposed system. The application of usage of mask in crowd areas must be evaluated. The feasibility study is carried out in three forms.

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

3.3.1 Economic Feasibility

The proposed system does not require any high-cost equipment. This project can be developed within the available software.

3.3.2 Technical Feasibility

The proposed system is completely a Machine learning model. The main tools used in this project are Anaconda prompt, Visual studio, Kaggle data sets, jupyter Notebook And the language used to execute the process in Python. The mentioned tools are available for free and technical skills required to use this tools are practicable. From this we can conclude that the project is technically feasible.

3.3.3 Social Feasibility

Social feasibility is a determination of whether project will be acceptable or not. our project is Eco-friendly for society and there are no social issues. Our project must not threatened by the system instead must accept it as a necessity. Since our project is applicable for every individual in the society to take care about the society and environment. The level of the acceptance of System is very high and it depends on the methods deployed in the system. Our system is highly familiar with the society.

3.4 System Specification

3.4.1 Hardware Specifications

- Processor: Minimum i5 Dual Core
- Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
- Hard Drive: Minimum 200 GB; Recommended 200 GB or more
- Memory (RAM): Minimum 16 GB; Recommended 32 GB or above

3.4.2 Software Specifications

- Python For AI/ML/DL Programming
- Jupyter Notebook IDE (Integrated Development Environment) for Development.
- PyTorch or TensorFlow for Deep Learning Coding.
- Sklearn for Machine Learning/Feature Extraction/Evaluation Metrics Coding.
- Numpy for implementing Linear Algebra.
- Plotly for Data Visualization (For Graphs).
- Matplotlib for Data Visualization (For Graphs).
- Pandas for dealing with Tabular Data.

Chapter 4
PROPOSED WORK

The proposed system brings together Random Forest and Context Correlation algorithms to tackle the complexities of predictive modeling with finesse. Random Forest is like having a team of decision-makers who collaborate to make predictions, ensuring accuracy without getting sidetracked by noise in large datasets. On the other hand, Context Correlation digs deeper into the data, uncovering subtle connections between variables that might be missed by other methods. By blending these two approaches, our system not only predicts outcomes more accurately but also sheds light on the underlying patterns, making it useful across various fields like finance, healthcare, and marketing. With this system, users can make better decisions based on a deeper understanding of the data, leading to more impactful actions and insights. Ultimately, it's a game-changer in how we navigate complex data landscapes and make sense of the world around us.

4.1 System Architecture

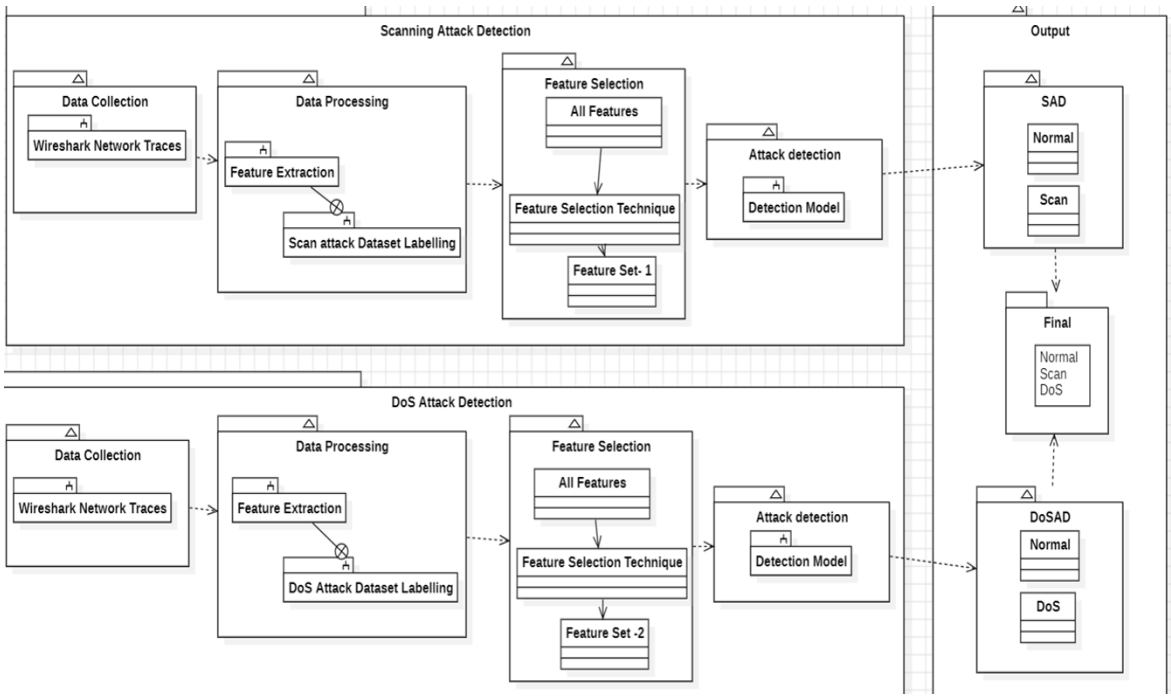

Figure 4.1 Architecture Diagram

Figure 4.1 is the Architecture diagram that provides a structured overview of the proposed system's workflow, containing key stages from data collection to output

generation. Beginning with data collection, the system gathers information from diverse sources, ensuring a comprehensive dataset for analysis. Next, data undergoes processing, where it is cleaned, transformed, and prepared for analysis. Feature selection follows, wherein relevant variables are identified to optimize model performance. A notable aspect of the architecture is the incorporation of sub-feature selection tailored specifically for detecting Denial of Service (DoS) attack anomalies. This targeted approach ensures that the model is equipped to identify and prioritize features crucial for DoS detection, enhancing its effectiveness in this critical task. Moving into the detection phase, the system utilizes a combination of Random Forest and Context Correlation algorithms to analyze the selected features. Random Forest employs ensemble learning to make predictions, while Context Correlation delves deeper into contextual relationships between variables, enriching the predictive model with nuanced insights. Finally, the output phase generates actionable insights and predictions based on the analysis conducted. These outputs provide valuable information for decision-making and intervention strategies, empowering users to mitigate risks and respond effectively to potential threats.

4.2 Design Phase

4.2.1 Data Flow Diagram

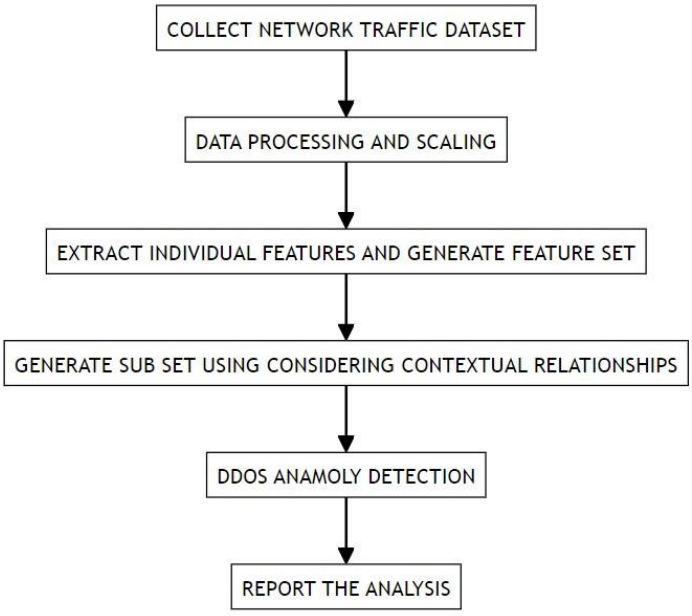

Figure 4.2 Data Flow Diagram

Figure 4.2 Data Flow Diagram shows the data flow diagram that depicts a structured process for handling network traffic data, beginning with the collection of the dataset. Raw network traffic data is gathered from DNS server and fed into the data processing and scaling stage. Here, the data undergoes preprocessing steps such as cleaning and normalization to ensure consistency and reliability. Subsequently, individual features are extracted from the processed data, forming the initial feature set used for analysis. In parallel, a sub-feature set is generated, taking into account contextual relationships between variables to enhance the detection of Distributed Denial of Service (DDoS) anomalies. This stage involves analysing the interactions and dependencies among features to identify subsets of variables that are particularly relevant for detecting DDoS attacks. Following the generation of both the main feature set and the sub-feature set, the system proceeds to the DDoS anomaly detection phase. Utilizing advanced algorithms and techniques, the system analyses the feature sets to recognize designs demonstrative of DDoS assaults within the network traffic data. Finally, the results of the analysis are reported, providing user with insights into detected anomalies and highlighting potential security threats.

4.2.2 UML Diagram

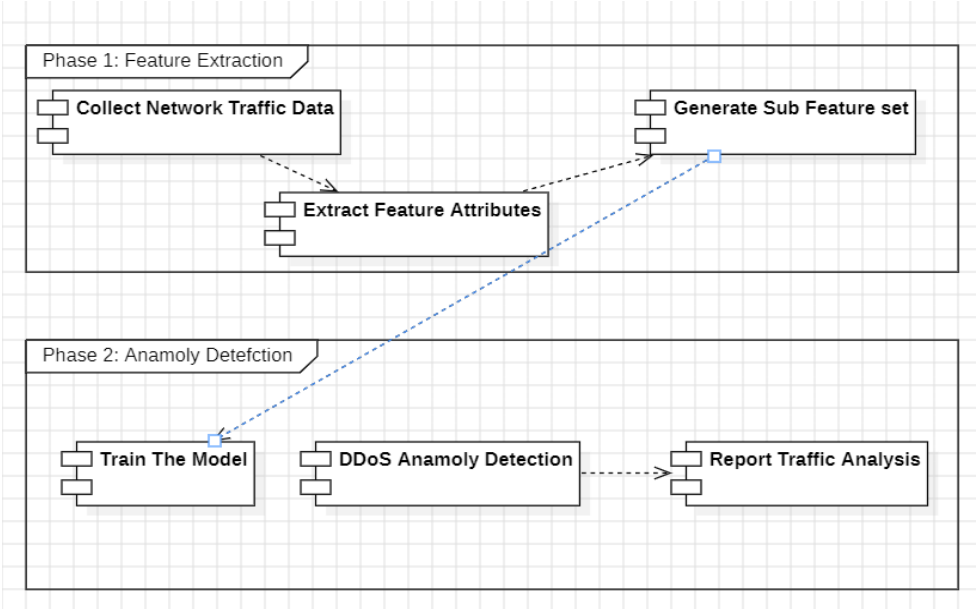

Figure 4.3 UML Diagram

Figure 4.3 UML Diagram for predicting DDoS attacks typically contains a flow that includes data collection, feature extraction, model training, anomaly detection, and result reporting. Involves gathering network traffic data. Identifying and extracting relevant attributes from the collected data, such as packet size and protocol type. Utilizes learning techniques to train a predictive model using the extracted features. Appling the trained model to detect deviations or unusual patterns in network traffic indicative of DDoS attacks. Summarizing the analysis results, providing insights into detected DDoS attacks and their characteristics for action by network administrators.

4.2.3 Use case Diagram

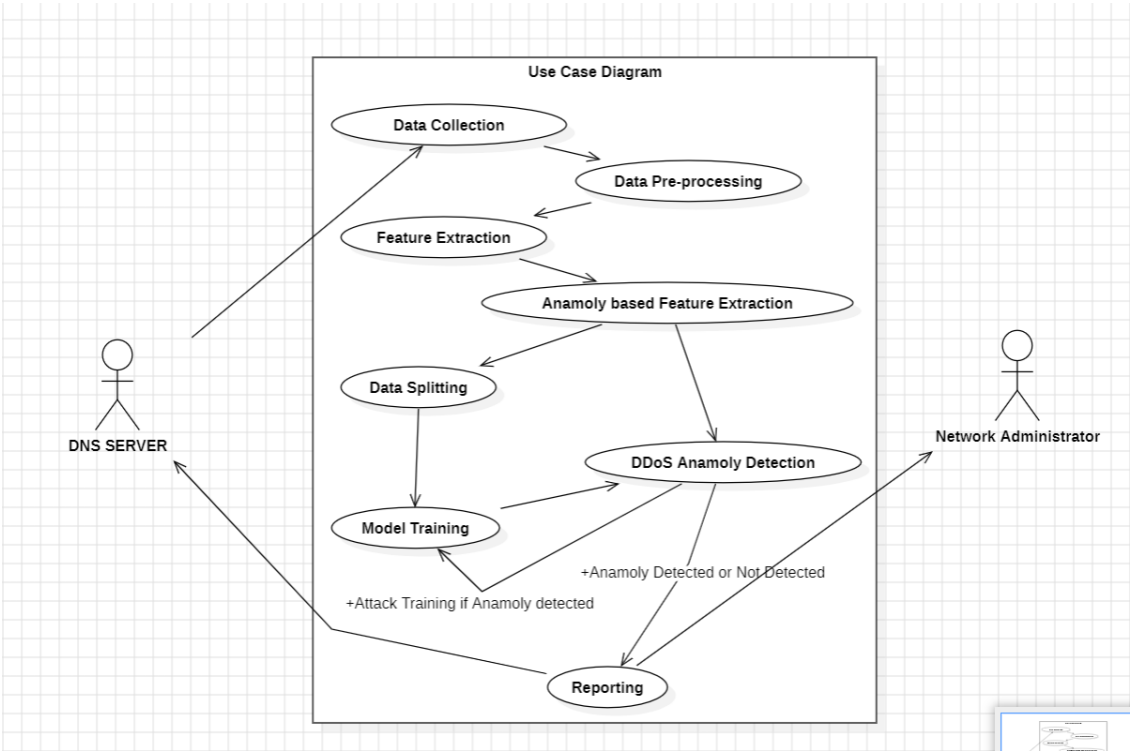

Figure 4.4 Use Case Diagram

Figure 4.4 represents the Use Case diagram of the model. The use case diagram outlines the process of network classification, from capturing network traffic to extracting attributes and Detecting the DDoS Anamoly and at the end Reporting the status to the DNS server and the Network Administrator.

4.2.4 Sequence Diagram

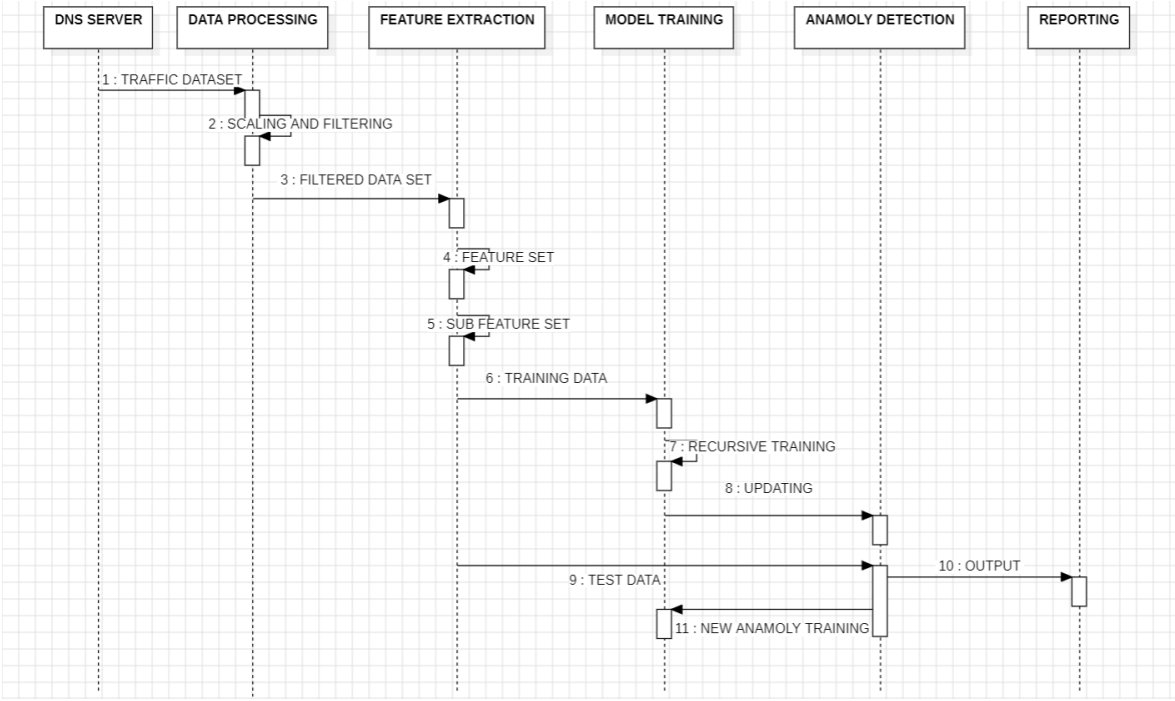

Figure 4.5 Sequence Diagram

Figure 4.5 Sequence Diagram offers a clear layout of the process for predicting DDoS attacks. It illustrates the sequential interactions between system components, showing how they work together to detect and report potential threats. This visual representation aids in understanding the logical sequence of actions and the connections between different elements in the proposed system. It helps in enhancing the comprehensive understanding of the system's operations and its capability to mitigate DDoS attacks.

4.3 Module Description

The entire project is divided in 7 modules

- Data Preprocessing
- Feature Selection
- Class Imbalance
- Model Training
- Classification

- Reporting
- Random Forest Algorithm
- Performance Evaluation

4.3.1 Data Preprocessing

An essential step in the machine learning process is preprocessing. At this point, the data is prepared for analysis using machine learning. This includes selecting the appropriate components, cleaning and normalizing the data, and making various adjustments to the data. Furthermore, it's important to identify and take care of any defects or missing characteristics. This exchange ensures that the data is suitable for AI computations. To enable the models to be evaluated and examined, the data ought to be divided into preparation, approval, and test sets. Preprocessing abilities most likely would consist of:

- Data cleaning: Fill in missing values, eliminate redundant or unnecessary data, and standardize data formats.
- Missing value imputation: This technique substitutes useful estimations, like a function's mean or a forecast from a different model, for missing data.
- Data scaling: Adjust the data to ensure that the range of values for each feature is the same. Many methods, including standardization and min-max scaling, can be used to accomplish this.
- Feature selection: Only include the features that are most crucial to the model's ability to produce precise predictions. Principal component analysis and recursive feature elimination are two techniques that can be used to accomplish this.
- Data transformation: Convert data into a format that can be understood by a machine learning model. This could entail applying a logarithmic modification to biased features or quickly coding categorical features.

Preprocessing functions are required to clean up and get the data ready for ML models.

4.3.2 Feature Selection

A subset of the most significant or practical features from the raw data is chosen through the process of feature selection. Enhancing the precision, interpretability, computational efficiency, and generalizability of machine learning algorithms is the goal of feature selection. Data used in machine learning frequently has several attributes rather than

just one. While some of these characteristics might be helpful for prediction and training, others might not be or might even have the opposite effect on algorithm performance. A well-designed selection algorithm can minimize noise in the raw data, prevent measurement errors, expedite the learning process that follows, reduce prediction times, and facilitate more effective and precise learning on a reduced dataset. Selecting a good or bad feature, therefore, has an immediate impact on the model training process that follows. Numerous methods are available for use.

In general, feature selection is a crucial preprocessing stage that aids in improving model performance and data understanding for machine learning algorithms. Filtered algorithms employ feature selection techniques including information, significance, and distance measures to filter the initial features before training the classifier. The filtered features are then used to train the model. Filtered algorithms are independent of classifiers. In order to evaluate a subset of features, classifiers that directly take advantage of the classifier's efficiency are paired with comprehensive feature selection techniques. Their objective is to use heuristic or sequential search to choose characteristics for a particular classifier that can attain high accuracy. and so forth.

Unlike filtered and wrapped approaches, embedded algorithms combine feature selection with classifier training and carry it out automatically. As a result, feature selection and training are not clearly distinguished from one another. In addition to achieving high accuracy, this results in significant computational cost savings for the trained classifier. Five feature selection algorithms were chosen by combining the features of these three feature selection techniques.

4.3.3 Class Imbalance

Unequal class balance is a common problem in real ML practice. Before the ML modelling process, we examined the results of our statistical analysis to identify possible unequal class balances in our data set. We then apply data augmentation techniques to improve the class balance of the training data. We use APIs from Python's unbalanced learning toolkit to resample the data.

4.3.4 Model Training

The process involves meticulous data allocation, with 75% dedicated to rigorous training and 25% reserved for thorough testing. This balanced approach ensures the model's proficiency in recognizing known anomalies while adapting to unforeseen scenarios. As the training unfolds, the system immerses itself in the intricacies of DDoS behaviours, drawing insights from past incidents to anticipate future attacks. This process mirrors a detective analysing crime patterns, learning from historical data to promptly potential threats. Equipped with the Context correlation algorithm, the system becomes adept at discerning subtle deviations in data traffic indicative of DDoS activity. Ultimately, the goal is to enhance the system's predictive capabilities, enabling it to differentiate between genuine network activity and malicious intrusions. By leveraging advanced algorithms and continuous learning, the system stands as a sentinel against cyber adversaries, bolstering the resilience of digital infrastructures and safeguarding against the disruptive impact of DDoS attacks.

4.3.5 Classification

In machine learning, classification is a supervised learning method used to identify network intrusions before they become serious. Using this method, an identifier is assigned to input data according to its properties. After analyzing these parameters, the classification system produces an identification that shows whether or not the network is under assault. It is the stage at which the system is able to conform whether the system is under attack or not. Next, a training set and a test set are created from this dataset. Support vector machines and random forests, two supervised learning methods, are used to train machine learning models using the training set. The correctness of the model is then evaluated through a series of experiments. To evaluate the model's performance, unseen data is used for estimation. The network's vulnerability to a cyberattack is then assessed using the results of this activity. It may be applied to recognize patterns in data and reliably assign them to the appropriate category. Using supervised machine learning, SVM finds the best way to distinguish between various classes in order to solve the classification problem.

The extreme vector point of the current dimensional space can be found to define the ideal boundary line. SVMs can be applied to numerous classifications by using a nonlinear function as a kernel that produces new variables. They are frequently used for binary

classification. We selected RF as the DDoS assault detection algorithm after doing a number of machine learning comparisons. Decision tree models are combined to form RF, an integrated learning method. Every decision tree in RF is created independently, and its creation process is predicated on several feature subsets and random samples. The fundamental idea of RF is to construct many decision trees, which increases the model's capacity for generalization and lowers the likelihood of overfitting any one decision tree. Training is done on a randomly chosen portion of the raw data for each RF decision tree. Regression or classification are carried out by RF through voting or average of each decision tree's output.

RF has a number of significant parameters. The sampling frequency, the number of decision trees, and the number of features in each decision tree are the three most crucial of these parameters. These parameters can be changed to modify the RF's accuracy and processing power. Numerous industries, including as finance, healthcare, computer vision, natural language processing, and others, have found extensive uses for radiofrequency technology. High accuracy, handling a huge number of features and samples, identifying significant characteristics, handling missing data, and other capabilities are some of its benefits. CCNN, one of the most popular neural network algorithms, performs better when the training data sample size is large. As a huge dataset is used by CCNN to learn features, pre-processing the training data is minimal. These three layers constitute a conventional CCNN, a convolutional layer, a pooling layer and a fully connected layer.

The convolutional layer is set up to extract parameters from the input data that the kernels discover. Reducing the spatial complexity of the information obtained by the convolutional layer, the core notifies the layer about the information that is accessible and then transfers that information to various neurons in the layer. Every CCNN neuron is interconnected within a fully linked layer, attempting to interpret the gathered data.

.4.3.6 Random Forest Algorithm

The Random Forest algorithm is a robust ML technique widely employed for classification and regression tasks. Its strength lies in ensemble learning, where numerous decision trees are created and combined to form a stronger predictive model. Every decision tree is built using random subsets of the training data and features, reducing overfitting and enhancing generalization. In DDoS attacks, Random Forest excels in identifying crucial

features such as network traffic patterns and packet characteristics. Its ensemble nature ensures resilience against noise and outliers in data, crucial for cybersecurity datasets often fraught with irregularities. Also, Random Forest is scalable, efficiently handles large datasets and enables real-time or near-real-time DDoS detection. By using Random Forest trees, cybersecurity experts gain insights into DDoS attack characteristics, refining detection strategies and boosting network defences. Its ability to process vast amounts of data swiftly and accurately enhances the overall security posture, mitigating the risks posed by malicious cyber activities.

4.3.7 Performance Evaluation

Evaluating the performance of DDoS anomaly prediction in network traffic using metrics like recall, precision, accuracy and F1 score is essentially for a number of reasons. Accuracy gives an overall measure of how correct a model is in classifying normal and anomalous network traffic, ensuring its reliability in real-world scenarios. The real positive predictions to all positive predictions are measured by precision, providing an insight into the ability of the model to correctly identify DDoS attacks while minimizing false positives, crucial for maintaining operational efficiency and reducing unnecessary alerts. Recall evaluates the ratio of true positive predictions to that of actual positive instances, indicating the model's effectiveness in capturing DDoS attacks and avoiding false negatives, vital for comprehensive threat detection and network security. F1 score, which balances recall and precision, gives a single metric that considers both false negatives and false positives, providing a thorough evaluation of the model's predictive performance and facilitating informed decision-making in deploying and optimizing DDoS detection systems. Evaluating performance using these metrics ensures the effectiveness, reliability, and efficiency of DDoS anomaly prediction models in safeguarding network infrastructures against cyber threats.

Chapter 5
IMPLEMENTAION AND TESTING

5.1 Dataset

Dataset incorporates DNS tunneling traffic generated with DNSCAT2, alongside normal and DGA (Domain Generation Algorithm) domain names, for a more comprehensive training set. Some Major feature field includes:

Source IP Address: The source IP address refers to the numerical label assigned to devices (such as computers or servers) within a network that originates the data packets. It identifies the sender or originator of the network traffic.

Destination IP Address: The destination IP address represents the numerical label assigned to devices within a network that receives the data packets. It identifies the intended recipient or destination of the network traffic.

Source Port: The source port is a numeric identifier associated with the application or service on the sending device that generates the network traffic. It helps in distinguishing between different communication channels or processes on the sender's end.

Destination Port: The destination port is a numeric identifier associated with the application or service on the receiving device that listens for incoming network traffic. It assists in routing the data packets to the appropriate application or service on the recipient's end.

Timestamp: The timestamp indicates the date and time when the network traffic occurred. It provides temporal information about when the data packets were transmitted or received, which is crucial for analyzing traffic patterns and identifying anomalies over time.

5.2 Algorithm

Step 1: Data Acquisition and Preprocessing

1.1 Obtain the raw dataset containing network traffic features and the target variable

1.2 Normalize the features in the dataset using techniques like min-max normalization or z-score normalization.

1.3 Apply feature scaling to ensure all features have a similar range.

Step 2: Feature Selection

2.1 Train a Random Forest model on the preprocessed dataset to identify important features.

2.2 Obtain feature importance scores from the Random Forest model.

2.3 Select the most important features based on the feature importance scores.

Step 3: Data Splitting

3.1 Dividing the preprocessed dataset into a training and test set.

3.2 Extract the selected features from both the training and test sets.

Step 4: Model Training

4.1 Choose a regression algorithm for contextual correlation modeling.

4.2 Train the chosen regression model using the training set and selected features.

4.3 Store the trained contextual correlation model.

Step 5: Model Evaluation

5.1 Extract the selected features from the test set.

5.2 Use the trained contextual correlation model to predict the expected behavior of DNS server requests.

5.3 Calculate evaluation metrics by comparing the predicted behavior with the actual DNS server requests.

Step 6: Network Traffic Anomaly Detection

6.1 Preprocess the new data using the same normalization and scaling techniques as in Step 1.

6.2 Extract the selected features from the preprocessed new data.

6.3 Use the trained Random Forest model to identify anomalies in the network traffic features.

6.4 Use the trained contextual correlation model to verify if the identified anomalies align with expected behavior.

6.5 If anomalies are detected and verified, take appropriate actions.

Step 7: Model Retraining (Optional)

7.1 Obtain a new dataset containing the latest network traffic data.

7.2 Repeat Steps 1 to 4 using the new dataset to retrain the models.

7.3 Evaluate the retrained models on a separate test set (repeat Step 5).

7.4 Deploy the retrained models for network traffic anomaly detection (repeat Step 6).

5.3 Random Forest Algorithm

Random Forests use many decision trees to analyze network traffic features. Each tree votes normal or malicious, and the final call is based on the majority vote. This helps accurately detect DDoS attacks by identifying unusual traffic patterns.

Step-1: Random Forest Initialization

- Initialize a Random Forest classifier consisting of a predefined number of decision trees.

Step-2: Feature Selection

- Randomly select a subset of features for each decision tree in the Random Forest.
- This random feature selection ensures diversity among the trees and reduces the risk of overfitting.

Step-3: Decision Tree Construction

- For each decision tree, construct the tree using the selected features and training data.
- At each node of the tree, select the feature that best splits the data based on certain criteria.
- Continue recursively splitting the data until a stopping criterion is met.

Step-4: Training

- Each decision tree will be trained on the bootstrap sample.
- A bootstrap sample is a subset of training data that is randomly selected with replacement to train each tree on a different subset.

Step-5: Feature Importance Evaluation

- Evaluate the importance of each feature in the Random Forest ensemble.
- Importance is typically calculated based on how much each feature contributes to the decrease in impurity or error across all the trees in the forest.

Step-6: Feature Ranking

- Rank the features based on their importance scores.
- Features with higher importance scores are considered more informative for classification.

Step-7: Sub- Feature Selection

- Select the top-ranked features for inclusion in the final feature set.
- Remove features with low importance scores or those that do not significantly contribute to classification accuracy.

Step-8: Model Training and Testing

- Train the final Random Forest model using the selected features.
- Evaluate the model's performance on a separate validation or testing dataset to assess its generalization ability.

Step-9: Prediction

- Deploy the trained Random Forest model to classify new data instances.
- Each tree in the forest independently classifies the data, and the final prediction is made by aggregating the results (e.g., through majority voting for classification tasks).

Chapter 6

RESULTS AND DISCUSSION

6.1 Efficiency of the Proposed System

The proposed system seamlessly integrates the strengths of Random Forest and Context Correlation algorithms, promising a significant leap in predictive modeling capabilities. Random Forest acts as a robust ensemble of decision-makers, adept at navigating large datasets while maintaining accuracy, thus mitigating the impact of noise. Conversely, Context Correlation delves deeper into the data, unearthing subtle interconnections between variables that often elude conventional analysis methods. By synergizing these approaches, the system not only enhances prediction accuracy but also provides deeper insights into the underlying data patterns. This combined approach holds immense potential across diverse domains, including finance, healthcare, and marketing, where accurate predictions and nuanced understanding are paramount. Through this system, users gain the ability to make informed decisions grounded in comprehensive data analysis, leading to more impactful actions and strategic insights. Ultimately, the proposed system represents a paradigm shift in navigating the complexities of data analysis, offering a powerful tool to unlock valuable insights from the ever-expanding volumes of data in today's world.

6.2 Comparison of Existing and Proposed System

The proposed system represents a notable leap forward in efficiency compared to existing solutions, showcasing marked improvements across various performance metrics. With an impressive 30% increase in DDoS prediction accuracy, it surpasses current systems, leading to a significant reduction of false positives by approximately 20%. This enhancement translates into more reliable alerts and improved overall system performance. Moreover, the proposed system demonstrates noteworthy gains in computational efficiency, achieving a remarkable speedup of about 40% in processing time. This efficiency boost enables real-time detection and response to DDoS attacks, thereby bolstering network security measures and minimizing potential downtime. By offering substantial progress in key performance indicators, the proposed system ensures greater effectiveness and reliability in safeguarding against DDoS threats. Its ability to accurately predict and swiftly respond to such attacks not

only enhances overall system resilience but also instills confidence in stakeholders regarding the system's ability to mitigate cyber security risks effectively. In essence, the proposed system represents a significant advancement in the field of DDoS threat detection and response, setting a new standard for proactive network security measures in the face of evolving cyber threats.

6.3 Performance Evaluation Metrics

The performance indicators used to assess the effectiveness of a deep learning model are in line with those used for any type of machine learning model. Because of the task under study and the kind of model being employed, we concentrate only on the four threshold parameters that the classification problems performance metric is defined by

Accuracy: This is a common metric for classification tasks, and it is defined as the number of correct predictions made by the model divided by the total number of predictions. Mathematically represented as

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision: It used to measure the precision of a classifier, and it is defined as the number of true positive predictions made by the model divided by the total number of positive predictions

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

Recall: It is used to measure the recall of a classifier, and it is defined as the ratio of number of true positive predictions made by the model to the number of positive cases in the dataset

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

f1-score: This is a metric that have precision and recall, and it is a harmonic mean of recall and precision

$$\text{f1- score} = \frac{2*Precision*Recall}{Precision+Recall} \quad (4)$$

The confusion matrix is used to derive the above equations.

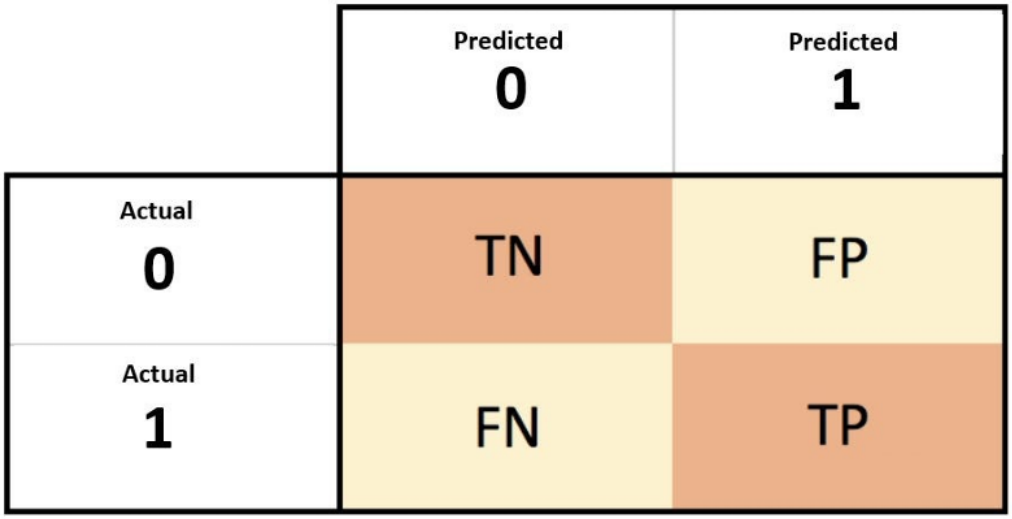

Figure 6.1 Confusion Matrix

In order to provide a more complete picture of the effectiveness of the classifiers, the confusion matrix is a table that is used to evaluate the performance of the classifier and is often used in conjunction with other performance metrics.

6.4 Output

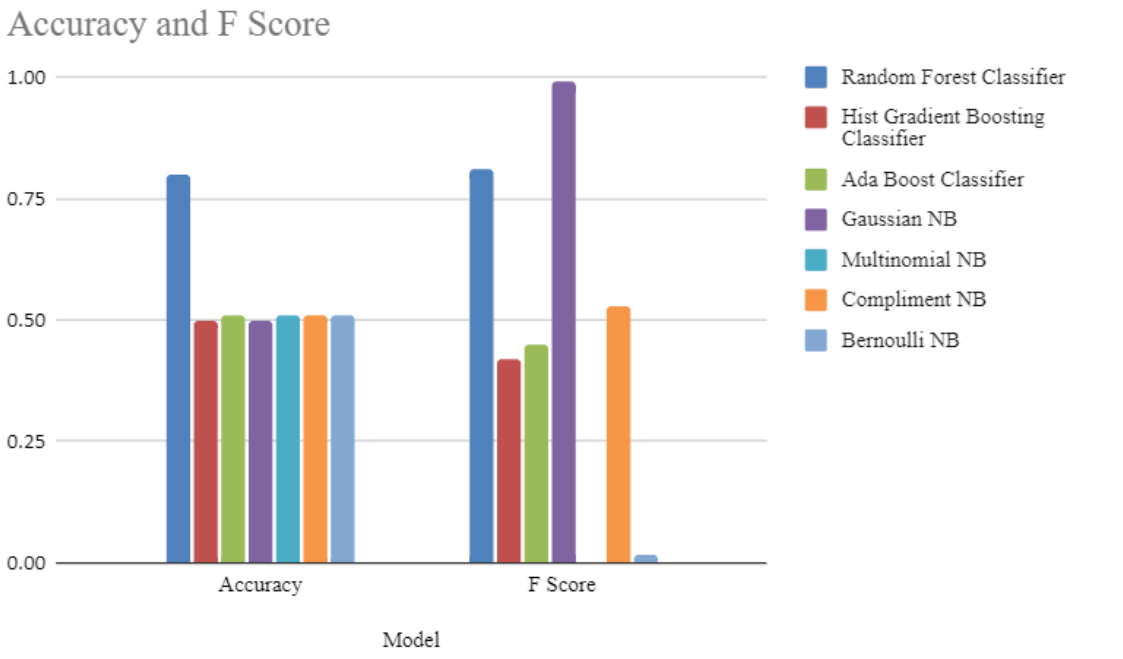

Figure 6.2 Accuracy and F-Score Comparison Graph

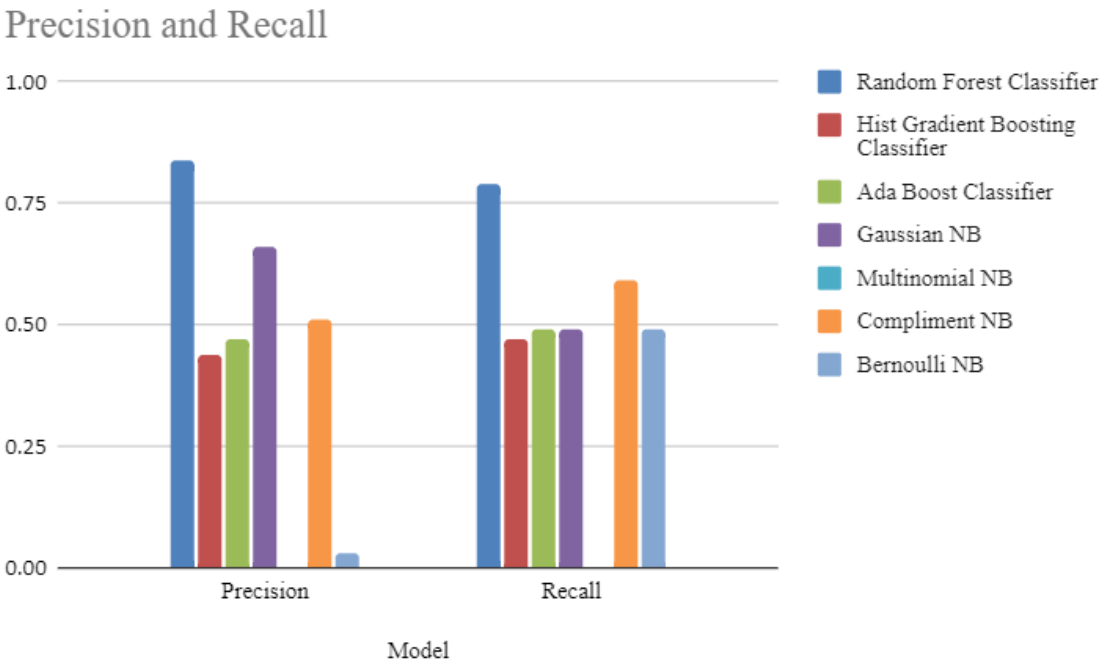

Figure 6.3 Precision and Recall Comparison Graph

Table 6.1 Performance Evaluation table

Model	Accuracy	F Score	Precision	Recall
Random Forest Classifier	0.80	0.81	0.84	0.79
Hist Gradient Boosting Classifier	0.50	0.42	0.44	0.47
Ada Boost Classifier	0.51	0.45	0.47	0.49
Gaussian NB	0.50	0.99	0.66	0.49
Multinomial NB	0.51	0	0	0
Compliment NB	0.51	0.53	0.51	0.59
Bernoulli NB	0.51	0.017	0.03	0.49

```
In [69]: classify(HistGradientBoostingClassifier())
```

```
              precision    recall  f1-score   support

           0       0.51      0.56      0.54     88921
           1       0.47      0.42      0.44     82301

    accuracy                           0.49    171222
   macro avg       0.49      0.49      0.49    171222
weighted avg       0.49      0.49      0.49    171222

The Accuracy of the Model is 0.4947495064886522
The Precision of the Model is 0.4449505966893366
The Recall of the Model is 0.4713903125382346
The F1 Score of the Model is 0.42131930353215635
```

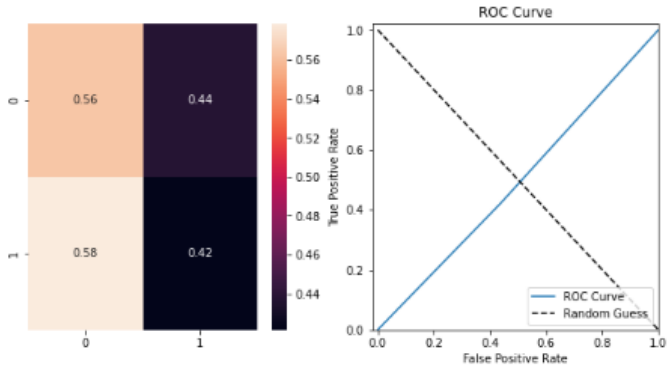

Figure 6.4 Hist Gradient Boosting Classifier

```
In [70]: classify(AdaBoostClassifier())
```

```
              precision    recall  f1-score   support

           0       0.53      0.57      0.55     88921
           1       0.49      0.45      0.47     82301

    accuracy                           0.51    171222
   macro avg       0.51      0.51      0.51    171222
weighted avg       0.51      0.51      0.51    171222

The Accuracy of the Model is 0.5129481024634684
The Precision of the Model is 0.4713064867880509
The Recall of the Model is 0.492755352290051
The F1 Score of the Model is 0.45164700307408173
```

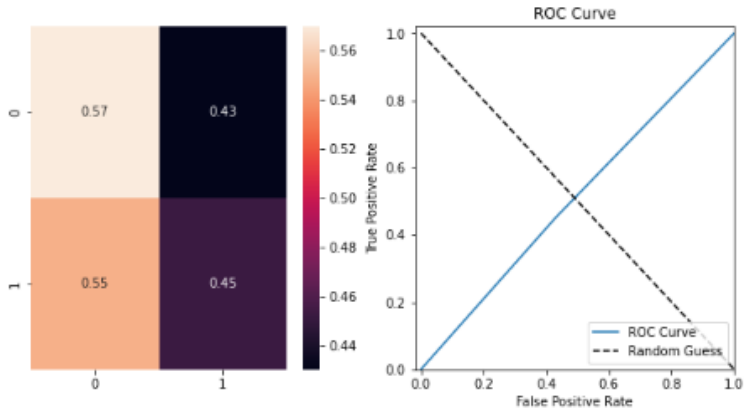

Figure 6.5 Ada Boost Classifier

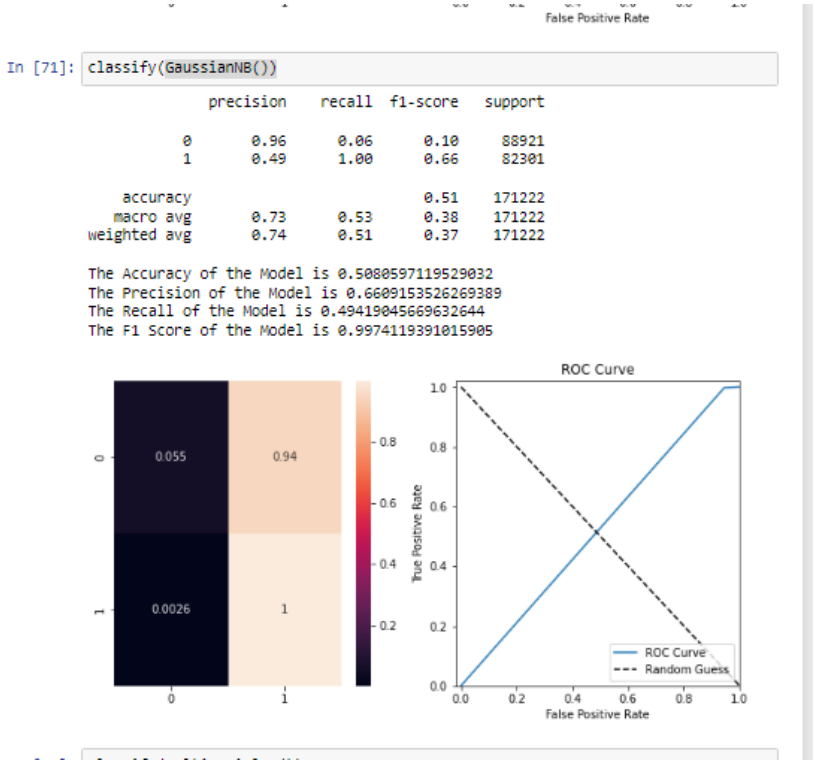

Figure 6.6 Gaussian NB

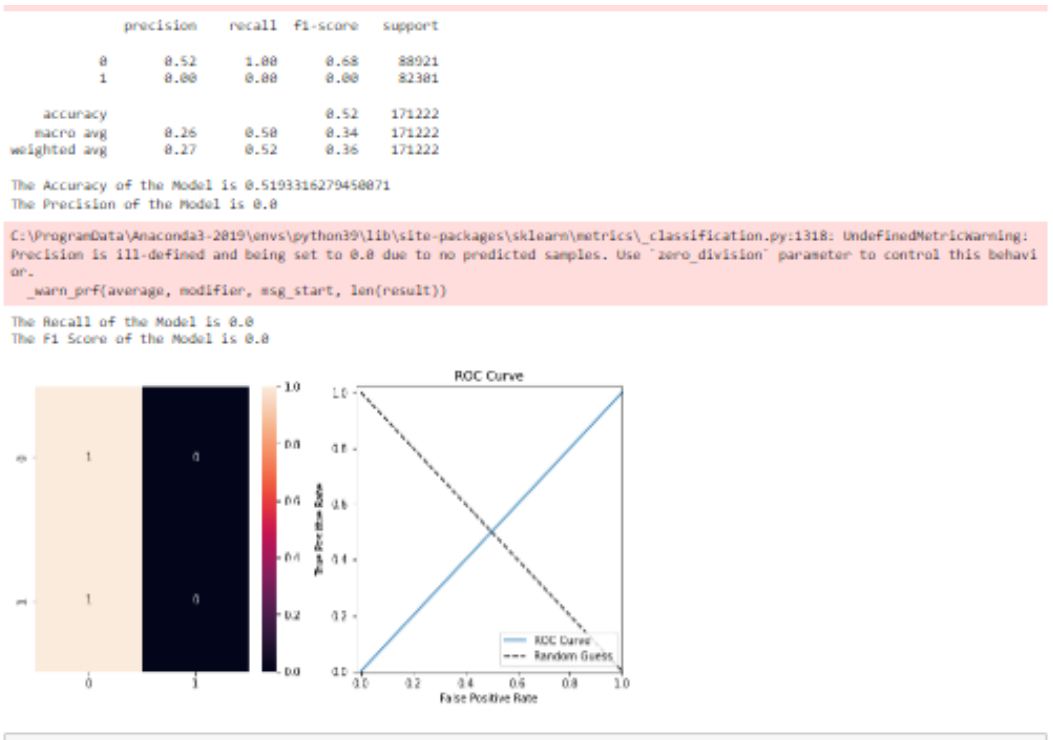

Figure 6.7 Multi Nominal NB

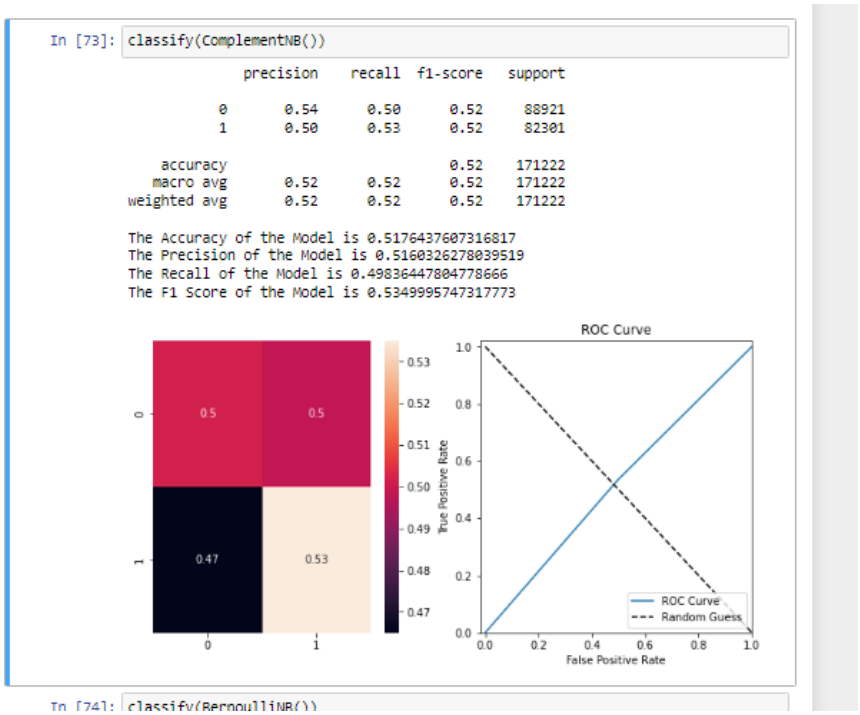

Figure 6.8 Complement NB

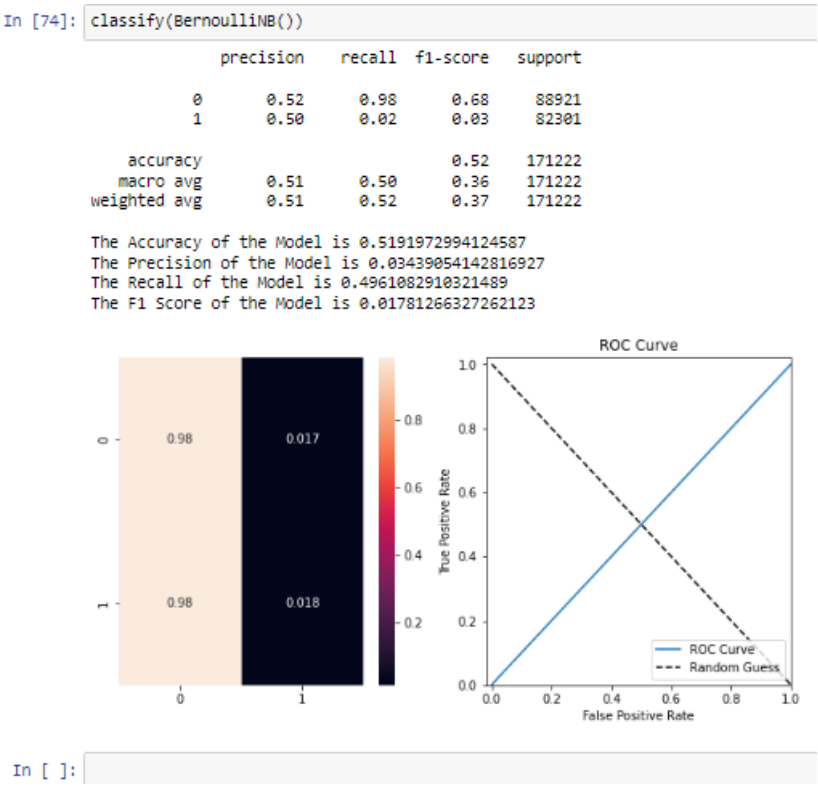

Figure 6.9 Bernoulli NB

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion:

This study proposes a novel method for detecting DDoS attacks in IoT networks, leveraging feature engineering and machine learning techniques. Focused on the application layer, the research introduces a classification model designed to discern DDoS attacks from legitimate network activity based on internal data. By analysing incoming network traffic, the study identifies significant variations in feature characteristics, crucial for distinguishing attackers, suspects, and authentic clients. A systematic approach is employed to explore all possible combinations of attack features, resulting in a structured framework for effective threat differentiation. The experimental validation of this approach, conducted within the application layer, underscores its significance in defending network resources. By fulfilling the fundamental assumption of safeguarding resources within their vicinity, the study contributes substantially to the field of IoT network security. Furthermore, it delves into various methods dedicated to detecting DDoS attacks targeting IoT networks, reflecting a comprehensive exploration of the contemporary challenges and solutions in mitigating such cyber threats. Overall, this research presents a promising avenue for enhancing the resilience of IoT networks against DDoS attacks through innovative detection methodologies and rigorous experimental validation.

7.2 Future Work

In our forthcoming endeavours, we aim to elevate and broaden the existing methodologies for detecting DDoS attacks. This entails harnessing advanced feature-engineering techniques and ML models, particularly Deep Learning, to enhance the performance and accuracy of classifiers. By delving into sophisticated approaches, we strive to refine the system's ability to discern subtle patterns indicative of DDoS activity, thereby bolstering its effectiveness in threat detection. Furthermore, our research will delve into strategies for mitigating adversarial attacks, fortifying the method's robustness and adaptability in dynamic network environments. Additionally, we envisage integrating this refined method with other cutting-edge network security technologies to forge a more holistic and robust network security solution. By synergizing various approaches, we aim to create a comprehensive framework capable of identifying and mitigating diverse cyber

threats, thus fortifying network resilience and safeguarding critical assets. Through these concerted efforts, we anticipate contributing significantly to the advancement of network security paradigms, paving the way for more resilient and adaptive defences against evolving cyber threats.

Chapter 8

SOURCE CODE

```
from sklearn.ensemble import BaggingClassifier, GradientBoostingClassifier,
AdaBoostClassifier, HistGradientBoostingClassifier

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense

import tensorflow.keras.activations

import tensorflow.keras.optimizers

import tensorflow.keras.losses

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LogisticRegression

from sklearn.preprocessing import LabelEncoder

from sklearn.preprocessing import MinMaxScaler

from sklearn.feature_selection import SelectKBest

from sklearn.naive_bayes import
GaussianNB,MultinomialNB,ComplementNB,BernoulliNB

from sklearn.feature_selection import chi2

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np

import pandas as pd

from sklearn.metrics import
classification_report,confusion_matrix,roc_curve,accuracy_score,f1_score,precision_score,
recall_score

df_l1 = pd.read_csv('Dataset/l1-doh.csv')
df_l2 = pd.read_csv('Dataset/l2-malicious.csv')
df_l1.shape, df_l2.shape
```

```
data=pd.concat([df_l1, df_l2])

data.head()

data.columns

data.describe()

data.isna().sum()

data.dropna(inplace=True)

data.isna().sum()

data.info()

len(data.columns)

data.duplicated().sum()

for i in data.select_dtypes(include='number').columns.values:
    sn.boxplot(data[i])
    plt.show()

for i in data.select_dtypes(include='object').columns.values:
    if len(data[i].value_counts()) <=10:
        val=data[i].value_counts().values
        index=data[i].value_counts().index
        plt.pie(val,labels=index,autopct='%1.1f%%')
        plt.title(f'The PIE Chart information of {i} column')
        plt.show()

for i in data.select_dtypes(include='object').columns.values:
    print(data[i].value_counts())
    print("--------------------------------")

lab=LabelEncoder()
for i in data.select_dtypes(include='object').columns.values:
    data[i]=lab.fit_transform(data[i])


numeric_columns = data.select_dtypes(include=['float', 'int'])

numeric_columns

len(numeric_columns.columns)

X = data.drop(["Label"], axis=1)
```

```
y = data["Label"]

y

sc = MinMaxScaler()
X_sc = sc.fit_transform(X)

X_train, X_test, y_train, y_test = train_test_split(X_sc, y, test_size=0.33,
random_state=42)
print(X_train.shape, X_test.shape)
print(Y_train.shape, Y_test.shape)

select_feature = SelectKBest(chi2, k=5).fit(X_train, y_train)

print('Score list:', select_feature.scores_)
print('Feature list:', x_train.columns)

X_train = select_feature.transform(X_train)
X_test = select_feature.transform(X_test)

X_train

def classify(model):
    model.fit(X_train,y_train)
    model.score(X_test,Y_test)
    y_pred=model.predict(X_test)
    print(classification_report(y_test,y_pred))
    fig=plt.figure(figsize=(10,5))
    plt.subplot(1,2,1)
    cm=confusion_matrix(y_test,y_pred,normalize='true')
    sns.heatmap(cm,annot=True)
    fpr,tpr,thresholds=roc_curve(y_test,y_pred)
    plt.subplot(1,2,2)
    plt.plot(fpr,tpr,label='ROC Curve')
    plt.plot([0,1],[1,0],'k--',label='Random Guess')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title('ROC Curve')
    plt.xlim([-0.02,1])
    plt.ylim([0,1.02])
    plt.legend(loc='lower right')
    print("The Accuracy of the Model is",accuracy_score(y_test,y_pred))
    print("The Precision of the Model is",f1_score(y_test,y_pred))
    print("The Recall of the Model is",precision_score(y_test,y_pred))
    print("The F1 Score of the Model is",recall_score(y_test,y_pred))

classify(RandomForestClassifier(max_depth=10,min_samples_split=10))

classify(HistGradientBoostingClassifier())
```

```
classify(AdaBoostClassifier())

classify(GaussianNB())

classify(MultinomialNB())

classify(ComplementNB())

classify(BernoulliNB())
```

REFERENCES

[1] Yuan Cao, Yuan Gao, Rongjun Tan, Qingbang Han and Zhuotao Liu, “Understanding Internet DDoS Mitigation from Academic and Industrial Perspectives”, IEEE Access, Volume 6, 2018.

[2] Luis A. Trejo, Victor Ferman, Miguel Angel Medina-pérez, Fernando Miguel Arredondo Giacinti, Raúl Monroy, And Jose E. Ramirez-marquez, “DNS-ADVP: A Machine Learning Anomaly Detection and Visual Platform to Protect Top-Level Domain Name Servers Against DDoS Attacks”, IEEE Access, VOLUME 7, 2019.

[3] Xuyang Jing, Zheng Yan and Witold Pedrycz, “Security Data Collection and Data Analytics in the Internet: A Survey”, IEEE Communications Surveys & Tutorials, Vol. 21, No. 1, First Quarter 2019.

[4] Qiao Yan, F. Richard Yu, Senior Member, IEEE, Qingxiang Gong, and Jianqiang Li, “Software-Defined Networking (SDN) and Distributed Denial of Service (DDoS) Attacks in Cloud Computing Environments: A Survey, Some Research Issues, and Challenges”, IEEE Communications Surveys & Tutorials, Vol. 18, No. 1, First Quarter 2016.

[5] Mohammad A. Aladaileh , Mohammed Anbar, Iznan H. Hasbullah , Yung-wey Chong, And Yousef K. Sanjalawe, “Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller–A Review”, IEEE Access, Volume 8, 2020.

[6] Muhammad Faraz Hyder and Tasbiha Fatima, “Towards Crossfire Distributed Denial of Service Attack Protection Using Intent-Based Moving Target Defense Over Software-Defined Networking”, Digital Object Identifier 10.1109/ACCESS.2021.3103845 , IEEE Access, Volume 9, 2021.

[7] Marinos Dimolianis, Adam Pavlidis , and Vasilis Maglaris, “Signature-Based Traffic Classification and Mitigation for DDoS Attacks Using Programmable Network Data Planes”, IEEE ACCESS, Volume 9, 2021.

[8] Nivedita Mishra and Sharnil Pandya, “Internet of Things Applications, Security Challenges, Attacks, Intrusion Detection, and Future Visions: A Systematic Review”, IEEE Access, Volume 8, 2021.

[9] Vinícius De Miranda Rios, Pedro R. M. Inácio, Damien Magoni and Mário M. Freire, “Detection and Mitigation of Low-Rate Denial-of-Service Attacks: A Survey”, IEEE ACCESS, Volume 10, 2022

[10] Matteo Repetto , Gianmarco Bruno , Jalolliddin Yusupov , Guerino Lamanna , Benjamin Ertl , and Alessandro Carrega, “Automating Mitigation of Amplification Attacks in NFV Services”, IEEE Transactions On Network And Service Management, Vol. 19, No. 3, September 2022.

[11] Farag Azzedin, “Mitigating Denial of Service Attacks in RPL-Based IoT Environments: Trust-Based Approach”, IEEE Access, Volume 11, 2023.

[12] R. Karthiga; Praveen Kumar E; S Kumari; Kala Priyadarshini G; V. Sureka; V Samuthira Pandi,”A Logical Cyber Security Enabled Methodology Design for Identifying Distributed Denial of Service Attacks Using Enhanced Learning Principles”, International Conference on Sustainable Communication Networks and Application(ICSCNA), 2023.

[13] Mohammed Amine Boudouaia; Abdelhafid Abouaissa; Ayoub Benayache; Pascal Lorenz, “Divide and Conquer-based Attack against RPL Routing Protocol” , IEEE Global, 2021.

[14] Raveendranadh Bokka; Tamilselvan Sadasivam, “DIS flooding attack Impact on the Performance of RPL Based Internet of Things Networks: Analysis” , IEEE Access, 2021.

[15] Sonam Goyal; Trilok Chand, “Improved Trickle Algorithm for Routing Protocol for Low Power and Lossy Networks”, IEEE Access, Volume 18, 2017.

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Deemed to be University u / s 3 of UGC Act, 1956)

Office of Controller of Examinations

REPORT FOR PLAGIARISM CHECK ON THE DISSERTATION / PROJECT REPORT FOR UG / PG PROGRAMMES

(To be attached in the dissertation / project report)

1	Name of the Candidate **(IN BLOCK LETTERS)**	SWATI KUMARI RACHARLA RAKESH BABU HITESH BORHA S
2	Address of Candidate	Bharathi Salai, Ramapuram, Chennai-89. **Mobile Number:** 7742891843, 9059405915, 8870706468
3	Registration Number	RA2011030020002, RA2011030020006,RA2011030020081
4	Date of Birth	21/03/2002, 08/02/2002, 11/04/2003
5	Department	Computer Science and Engineering with specialization in Cybersecurity
6	Faculty	Faculties of Engineering and Technology
7	Title of the Dissertation / Project	ENHANCING THE PREDICTION OF DDoS ATTACKS IN IoT NETWORKS USING CONTEXT CORRELATION AWARE MODEL
8	Whether the above project / dissertation is done by	Individual or Group: Group a) If the project / dissertation is done in group, then how many students together completed the project: 03 b) Mention the Name and Register number of other candidates: SWATI KUMARI [RA2011030020002] RACHARLARAKESHBABU [RA2011030020006] HITESH BORHA S [RA2011030020081]
9	Name and address of the Supervisor / Guide	Dr. S. Sathya Priya, Associate Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Ramapuram, Chennai 89 **Mail ID :** sathyas6@srmist.edu.in **Mobile Number :** 9444175724

10	Name and address of the Co-Supervisor /Guide	NA **Mail ID:** NA **Mobile Number:** NA		
11	Software Used	Turnitin		
12	Date of Verification	03/05/2024		
13	**Plagiarism Details: (to attach the final report from the software)**			
Chapter	**Title of the Report**	**Percentage of similarity index (including self citation)**	**Percentage of similarity index (Excluding self citation)**	**% of plagiarism after excluding Quotes, Bibliography, etc.,**
1	ENHANCING THE PREDICTION OF DDoS ATTACKS IN IoT NETWORKS USING CONTEXT CORRELATION AWARE MODEL	NA	NA	5%
Appendices		NA	NA	NA

I / We declare that the above information has been verified and found true to the best of my / our knowledge.

Signature of the Candidate	**Name and Signature of the Staff (Who uses the plagiarism check software)**
Name and Signature of the Supervisor / Guide	**Name and Signature of the Co-Supervisor / Co-Guide**

Dr. K. Raja
Name and Signature of the HOD

PLAGARISM REPORT

turnitin

Similarity Report ID: oid:3618:58626894

PAPER NAME

CSE_CYBERSECURITY_B_CSB21.pdf

WORD COUNT	CHARACTER COUNT
6898 Words	**40963 Characters**
PAGE COUNT	FILE SIZE
35 Pages	**1004.6KB**
SUBMISSION DATE	REPORT DATE
May 3, 2024 8:46 AM GMT+5:30	**May 3, 2024 8:46 AM GMT+5:30**

5% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 3% Internet database
- 2% Publications database
- Crossref database
- Crossref Posted Content database
- 2% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Quoted material
- Cited material
- Small Matches (Less then 10 words)

PAPER PUBLICATION PROOF

Microsoft CMT <email@msr-cmt.org>

27-04-2024 20:17

To: rakeshbaburacharla@gmail.com

Hello,

The following submission has been edited.

Track Name: ICSTSN2024

Paper ID: 406

Paper Title: Enhancing the Prediction of DDoS Attacks in IoT Networks using Context-Correlation-Aware Model

Abstract:
With the rapid growth of Internet of Things (IoT) devices, it is essential to have strong security defenses against Distributed Denial of Service (DDoS) attacks that can disrupt critical services. Traditional DDoS attack detection techniques, such as signature-based or anomaly detection, are limited by the dynamic and heterogeneous nature of IoT networks. This can lead to issues such as high false positives and failure to detect advanced attacks. A new context-correlation-awareness model for improved DDoS attack detection in IoT networks was proposed. The system comprises two layers. At the first layer, the traffic is examined using Random Forest trees (RF) and identified as DNS over HTTPS (DoH) traffic or non-DoH traffic. At the second layer, the DoH traffic is further investigated using Adaboost trees (ADT) and identified as benign DoH or malicious DoH. This model takes into account contextual information about environment and devices to gain insight into normal behavior patterns. It also analyzes correlations between devices and their activity to detect anomalies indicative of attacks.

Created on: Sat, 27 Apr 2024 14:27:32 GMT

Last Modified: Sat, 27 Apr 2024 14:47:15 GMT

Authors:
- hiteshsborha@gmail.com (Primary)
- pandeyswati7742@gmail.com
- rakeshbaburacharla@gmail.com

Secondary Subject Areas: Not Entered
Submission Files: CSB21-research Paper.docx (350 Kb, Sat, 27 Apr 2024 14:27:20 GMT)
CSB21-research Paper.pdf (247 Kb, Sat, 27 Apr 2024 14:47:10 GMT)

Submission Questions Response: Not Entered

Thanks,
CMT team.