A MAJOR PROJECT-II REPORT
ON

Development of Framework for DDoS Attack Detection in Network Devices Using Machine Learning

Submitted in Partial Fulfilment of the Requirement
for the Degree of

MASTER OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

By
KULDEEP KUMAR
2K22/CSE/26

Under the Guidance of
Dr. Rahul Katarya
(Professor)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-110042

May, 2024

ACKNOWLEDGEMENTS

I have taken efforts in this survey paper. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to **Prof. Rahul Katarya** for his guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing this review paper. I would like to express my gratitude towards the **Head of the Department (Computer Science and Engineering, Delhi Technological University)** for their kind cooperation and encouragement which helped me in the completion of this research survey. I would like to express my special gratitude and thanks to all the Computer Science and Engineering staff for giving me such attention and time.

My thanks and appreciation also goes to my colleague in writing the survey paper and the people who have willingly helped me out with their abilities.

Kuldeep Kumar

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I, Kuldeep Kumar, Roll No. 2K22/CSE/26 student of M.Tech (Computer Science & Engineering), hereby certify that the work which is being presented in the thesis entitled "**Development of Framework for DDoS Attack Detection in Network Devices Using Machine Learning**" in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Computer Science & Engineering in the Department of Computer Science and Engineering, Delhi Technological University is an authentic record of my own work carried out during the period from August 2022 to June 2024 under the supervision of Prof Rahul Katarya, Asst Prof, Dept of Computer Science and Engineering. The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Place: Delhi **Candidate's Signature**

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisor **Signature of External Examiner**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE

Certified that **Kuldeep Kumar**(Roll No. 2K22/CSE/26) has carried out the research work presented in the thesis titled "**Development of Framework for DDoS Attack Detection in Network Devices Using Machine Learning**", for the award of Degree of Master of Technology from Department of Computer Science and Engineering, Delhi Technological University, Delhi under my supervision. The thesis embodies result of original work and studies are carried out by the student himself and the contents of the thesis do not form the basis for the award of any other degree for the candidate or submit else from the any other University /Institution.

Prof. Rahul Katarya
(Supervisor)
Department of CSE
Delhi Technological University

Date

Development of Framework for DDoS Attack Detection in Network Devices Using Machine Learning

Kuldeep Kumar

ABSTRACT

The exponential growth of internet users poses a significant challenge to safeguarding online resources against security threats. The escalating frequency of Denial of Service (DoS) attacks further intensifies these concerns, underscoring the urgent need for sophisticated cyber-defense mechanisms. Addressing this imperative, our study presents an innovative machine learning-based system engineered to detect Distributed Denial of Service (DDoS) attacks. By harnessing the predictive capabilities of Logistic Regression, K Nearest Neighbor, and Random Forest algorithms, our approach fortifies defenses against evolving cyber threats. To gauge the efficacy of our models, extensive experiments were conducted utilizing the recently updated NSL KDD dataset. The results unveil the exceptional accuracy of our proposed system in identifying DDoS attacks, surpassing prevailing state-of-the-art detection methods. These findings underscore the pivotal role of our research in bolstering cyber-security resilience amidst the mounting challenges posed by the digital landscape.

LIST OF PUBLICATIONS

[1] Kuldeep Kumar, Rahul Katarya. “Securing SDN from DDoS attacks: A Comprehensive Study of Security Challenges and Opportunities” *Proceedings of the International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET)*, 2024

[2] Kuldeep Kumar, Rahul Katarya. “GuardianNet: An Intelligent Framework for Guarding the Network against DDoS Attacks using Machine Learning” *Proceedings of the International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET)*, 2024

CONTENTS

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

A Distributed Denial-of-Service (DDoS) attack is a malicious attempt to disrupt the normal traffic of a targeted server, service, or network by overwhelming the target or its surrounding infrastructure with a flood of Internet traffic. Distributed Denial of Service (DDoS) attacks achieve efficacy by leveraging a multitude of compromised computing systems to generate attack traffic. The exploited systems may encompass not only traditional computers but also other networked resources, including Internet of Things (IoT) devices. These attacks impacts heavy losses to the infrastructure, industry, government and economy[1]. The DDoS attacks are a unique type of attempt where the online services of a specific web server are disrupted with malign intent. Contrast this with a DoS assault, which employs a single device to bombard a target with traffic. DDoS attacks can be classified into three primary subtypes: volume-based attacks, protocol-based attacks, and application layer attacks[2]. Volume based attacks aim to inundate the target's bandwidth with excessive traffic using methods such as UDP floods, ICMP floods, and other types of spoofed-packet floods. Protocol based attacks target server resources or intermediate communication infrastructure through techniques like SYN floods, fragmented packet attacks, Ping of Death, and Smurf DDoS. Together, these attack types can severely disrupt the availability and performance of targeted online services. These attacks are often disguised as legitimate and innocent requests. These attacks can last only a few minutes or less than an hour, making them difficult to detect with common tools. These DDoS attacks can be detected using several machine learning algorithms.

1.1 Motivation

The proliferation of cyber threats, particularly Distributed Denial of Service (DDoS) attacks, poses a significant and escalating risk to the stability and security of network infrastructures worldwide. As the reliance on digital systems continues to grow across various sectors, including critical infrastructure, finance, healthcare, and government, the potential impact of DDoS attacks on organizational operations, financial stability, and public safety cannot be overstated. Therefore, this research endeavors to address the pressing need for advanced machine learning-based solutions to enhance network security and resilience against DDoS attacks, ultimately safeguarding critical infrastructure and ensuring the integrity and availability of digital services for individuals and organizations alike.

Such assaults can have profoundly adverse effects, including prolonged service outages, significant financial repercussions, and severe reputational damage to the affected organization. The complexity, scale, and methodologies of DDoS attacks may vary, yet their unifying objective remains to render the target inaccessible to its intended users. This comprehensive guide delves into the diverse types of DDoS attacks, elucidates their distinct characteristics, and examines the strategies employed to mitigate their impact.

1.1.1 Volumetric Attacks

Volume-based attacks overwhelm the target's bandwidth by generating high levels of traffic using methods such as UDP floods[3], ICMP floods[4], and other spoofed-packet floods. Protocol attacks deplete server resources or intermediate communication devices through techniques such as SYN floods[5], fragmented packet attacks, Ping of Death, and Smurf DDoS. Application layer attacks target the layer responsible for generating and delivering web pages in response to HTTP requests, with examples including HTTP floods, Slowloris, and DNS query floods. Application layer attacks focus on the layer where web pages are generated and delivered in response to HTTP requests, with examples including HTTP floods, Slowloris, and DNS query floods. Together, these attack types can severely disrupt the availability and performance of targeted online services.

Mitigating these attacks involves a range of strategies. Rate limiting imposes a cap on the number of requests a server processes within a designated period. Web Application Firewalls (WAF) defend applications by scrutinizing and filtering HTTP requests. Load balancers manage incoming network traffic by evenly distributing it among several servers, thereby avoiding overload on any single server. DDoS protection services, such as Cloudflare, Akamai, and AWS Shield, offer advanced mitigation techniques. Additionally, an anycast network distributes traffic across a network of data centers, absorbing the attack across multiple locations. Understanding these methods and implementing robust defenses is crucial for maintaining the security and performance of internet-facing resources.

1.1.2 Protocol Attacks

State-exhaustion attacks, a subset of protocol attacks, focus on draining the resources of a target's server or network infrastructure, such as firewalls and load balancers, by taking advantage of flaws in the network protocol architecture. These attacks are particularly insidious because they do not require a large volume of traffic to be effective; instead, they send malformed or malicious packets that force the target to use significant computational resources to process them. Instances include SYN floods, where the attacker inundates the target system with a barrage of SYN requests, aiming to exhaust server resources and render the system incapable of responding to legitimate traffic. Fragmented packet attacks, such as Teardrop, exploit vulnerabilities in the way network devices reassemble fragmented packets, causing them to crash or behave unpredictably.

Another common form of protocol attack is the Ping of Death[6], which involves sending oversized ICMP packets to a target, causing buffer overflows and potential crashes. The Smurf DDoS attack[7] leverages ICMP Echo requests to flood a target with traffic by exploiting the broadcast address of intermediary networks[8]. These attacks can be particularly challenging to mitigate because they exploit legitimate protocol behaviors, making it difficult to distinguish between malicious and normal traffic. Effective mitigation often requires a combination of strategies, including deep packet inspection, rate limiting, and the use of robust intrusion detection and prevention systems. By understanding and identifying the nuances of protocol attacks, organizations can better defend against these sophisticated threats and ensure the stability and security of their networks.

1.1.3 Application Layer Attacks

Assaults at the application layer focus on the stratum responsible for crafting and delivering web pages following HTTP requests[9], with the objective of inundating the application or server with malevolent traffic. These attacks often exploit vulnerabilities in the application's code or design, making them particularly challenging to mitigate. An illustrative instance involves HTTP flooding[10], wherein assailants inundate a target's web server with an extensive array of HTTP requests, depleting its resources and inducing unresponsiveness towards genuine users. Another paradigm is Slowloris, characterized by its strategy to maintain a plethora of connections to a targeted web server by dispatching fragmented HTTP requests at consistent intervals. This tactic aims to saturate the server's ability to entertain fresh connections, thus instigating a denial of service scenario.

DNS query floods represent another form of application layer attack[11], where attackers flood a target's DNS server with a barrage of DNS queries, overwhelming its capacity to respond to legitimate requests and potentially causing it to crash. These attacks exploit the hierarchical nature of the DNS system and can disrupt a wide range of online services that rely on DNS resolution[12]. Mitigating application layer attacks often involves implementing specialized security measures such as rate limiting, web application firewalls (WAFs). Additionally, regularly updating and patching web applications to address known vulnerabilities can help reduce the risk of successful application layer attacks.

1.1.4 Reflection and Amplification Attacks

Reflection and amplification attacks are sophisticated techniques used by attackers to magnify the impact of their DDoS assaults, thereby maximizing the damage inflicted upon their targets. Through the manipulation of their target's source IP address, perpetrators can reroute these magnified responses to inundate the victim's network with an influx of traffic, overloading its bandwidth and resulting in service interruption. This technique effectively leverages the unwitting assistance of third-party systems to amplify the scale of the attack.

Amplification attacks, on the other hand, involve exploiting protocols or services that can generate significantly larger responses to small requests. For instance, attackers may abuse protocols like DNS, NTP[13], SSDP[14], and memcached, which can produce responses that are many times larger than the initial query. Through the dispatch of a limited quantity of requests bearing falsified source IP addresses, assailants can prompt these services to discharge an overwhelming surge of data towards their intended target, saturating its network capacity and incapacitating access for genuine users[15]. The mitigation of reflection and amplification attacks necessitates a multifaceted strategy, incorporating measures such as network filtration, imposition of rate limits, and engagement with internet service providers to detect and impede malevolent traffic[16]. Additionally, organizations must proactively secure and harden their systems to prevent them from being exploited as unwitting accomplices in these devastating assaults.

1.1.6 Zero-Day Exploits and Advanced Persistent Threats (APTs)

- **Zero Day Exploits**: Zero-day exploits denote weaknesses present in software or hardware that assailants uncover and manipulate before the vendor or creator becomes cognizant of them, affording no time for rectification or alleviation[17]. These vulnerabilities are greatly coveted by cybercriminals due to their capacity to infiltrate systems sans detection or deterrence from prevailing security protocols. Zero-day exploits have the potential to assail an extensive array of software, encompassing operating systems, web browsers, and applications, in addition to hardware elements like routers and IoT devices.

  The identification and utilization of zero-day vulnerabilities present substantial obstacles for cybersecurity experts, given their frequent shortage of time or resources to devise and implement remedies prior to potential exploitation by attackers. Furthermore, zero-day exploits can be particularly devastating because they bypass traditional security controls and can be used to launch highly targeted and stealthy attacks. To mitigate the risk posed by zero-day exploits, organizations must adopt proactive security measures, such as implementing intrusion detection systems, conducting regular vulnerability assessments, and maintaining strong security hygiene practices. Additionally, collaboration between security researchers, vendors, and the broader cybersecurity community is essential for rapidly identifying and addressing zero-day vulnerabilities before they can be exploited by malicious actors.

- **Advanced Persistent Threats (APTs)**: APT groups may employ DDoS attacks as part of their broader cyber espionage or sabotage campaigns[18]. These attacks are typically highly sophisticated, well-coordinated, and persistent, aiming to disrupt critical infrastructure, steal sensitive data, or achieve other strategic objectives.

1.2 Objectives

The primary objective of this research initiative is to lead the advancement and thorough assessment of machine learning-based approaches specifically designed for accurate identification and effective suppression of Distributed Denial of Service (DDoS) attacks in complex network landscapes. This ambitious undertaking seeks to propel the field forward by harnessing the formidable capabilities of advanced machine learning algorithms to bolster network defenses against the escalating menace of DDoS attacks. The research endeavors to attain a comprehensive understanding of DDoS attack modalities, behavioral nuances, and evasion stratagems, thereby facilitating the creation of resilient and adaptive detection mechanisms adept at promptly discerning emergent threats.

Moreover, the research endeavor aspires to transcend mere detection and broaden its purview to encompass proactive mitigation strategies for DDoS attacks, with the goal of crafting intelligent countermeasures that dynamically adjust to evolving attack methodologies in real-time. By leveraging sophisticated machine learning models and algorithms, the study aims to equip network administrators with actionable insights and automated response capabilities to expeditiously and efficiently mitigate the disruptive impact of DDoS assaults. Through systematic experimentation and rigorous evaluation, the research aims to ascertain the efficacy and scalability of these pioneering techniques across heterogeneous network infrastructures, laying the groundwork for their practical deployment in real-world contexts. Ultimately, this research initiative seeks to bolster the resilience of network systems against the pervasive threat of DDoS attacks, thereby safeguarding the integrity, availability, and functionality of critical digital infrastructure. Specifically, the research aims to:

- Explore cutting-edge machine learning algorithms and methodologies to analyze and classify DDoS attacks within network traffic data.
- Design & implement novel data pre-processing techniques to optimize the input data for machine learning models, enhancing their effectiveness in identifying malicious traffic patterns.
- Investigate feature selection techniques to pinpoint the most informative and pertinent features for detecting DDoS attacks, thereby reducing dataset dimensionality while upholding predictive accuracy.
- Formulate and train machine learning models proficient in precisely detecting and categorizing diverse forms of DDoS attacks, encompassing volumetric, protocol-based, and application layer assaults.
- Evaluate the performance of the proposed machine learning-based DDoS detection techniques using real-world network traffic datasets, assessing their accuracy, efficiency, and scalability in different network environments.
- Validate the efficacy of the developed models via comparative analysis against established DDoS detection approaches, showcasing their superiority in detection accuracy, false positive rate, and computational efficiency.
- Provide insights as well as recommendations for practical deployment & integration of machine learning-based DDoS detection systems in operational network environments, ensuring their effectiveness and reliability in mitigating cyber threats and enhancing network security posture.

1.3 Thesis Organization

The thesis is structured into five cohesive chapters, each contributing to a comprehensive exploration of the detection and mitigation of Distributed Denial of Service (DDoS) attacks in network environments.

Chapter 1, "Introduction," lays the groundwork by elucidating the motivation behind the research and delineating its objectives. This chapter sets the stage for the ensuing investigation and outlines the organization of the thesis.

Chapter 2, "Related Work," conducts a thorough review of existing literature in the field, identifying pertinent studies and elucidating the research gap. By articulating the problem statement, this chapter contextualizes the proposed work within the broader landscape of DDoS attack detection and mitigation.

Chapter 3, "Proposed Work," constitutes the heart of the thesis, presenting the novel methodologies devised for the detection and mitigation of DDoS attacks. It begins with an exposition of the preliminaries, followed by a detailed discussion on data pre-processing techniques. The chapter culminates with the presentation of the proposed model, elucidating its architecture and underlying principles.

Chapter 4, "Experimental Setup and Result Analysis," provides a comprehensive overview of the experimental framework employed in evaluating the proposed methodologies. It details the experimental setup, describes the dataset used, and delineates the performance evaluation parameters. The chapter concludes with a meticulous analysis of the experimental results, offering insights into the efficacy and robustness of proposed techniques.

Chapter 5, "Conclusion and Future Work," synthesizes the outcomes of the investigation and analyze to derive comprehensive insights and formulate overarching conclusions. Additionally, it identifies the limitations of the proposed methodologies and discusses potential industrial applications. Finally, the chapter outlines avenues for future research, paving the way for continued advancements in the field of DDoS attack detection and mitigation.

CHAPTER 2
RELATED WORK

2.1 Literature Survey

Below, we have explored cutting-edge research conducted in the domain of network security, delving into the ramifications of diverse types of DDoS attacks on various organizations and outlining potential mitigation strategies and solutions:

2.1.1 Escalation of DDoS Attacks in 2022: Drifts and Challenges Amid Global Diplomatic Turmoil: - This article analyzes the pronounced surge in hacker activity on a global scale. In the third quarter of 2022, the number of attacks escalated by 90% worldwide compared to the same period in the previous year[19]. Additionally, the potency of these attacks has significantly increased. The prevalence of botnets across numerous countries has intensified, making such attacks exceedingly difficult to mitigate independently. Furthermore, political dynamics have had a profound impact on DDoS activity. Specifically, towards the end of February, politically motivated hacktivist groups emerged, orchestrating DDoS attacks on Russian companies with the objective of destabilizing the country's economy. The so-called “IT army of Ukraine” in particular has targeted hundreds of Russian private and state-owned companies and is responsible for the most politically motivated incidents. They have developed DDoS tools, which threat actors around the world are now adopting and using to launch some of the most powerful attacks we’ve seen to date. Businesses in many countries are in the crosshairs. All this has led to a significant increase in attacks worldwide.

2.1.2 Employing ML for DDoS Detection in IoT Networks: - In the contemporary digital landscape, the Internet is ubiquitous, with the rise of IoT technology interconnecting billions of devices worldwide[20]. However, this rapid expansion of IoT has made it a prime target for Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks, which pose significant threats. The sophistication and complexity of new DDoS attacks have rendered traditional intrusion detection systems and conventional methods nearly ineffective in identifying and mitigating these threats. Fortunately, advancements in Big Data, Data Mining, and Machine Learning have paved the way for effective detection of DDoS attacks.

This paper utilizes the latest dataset, CICDDoS2019, the study experiments with prominent machine learning algorithms and identifies the most relevant features correlated with predictive classes. The findings reveal that AdaBoost and XGBoost algorithms demonstrate exceptional accuracy, achieving 100% precision in predicting the type of network traffic. Subsequent research endeavors have the opportunity to advance this model by augmenting its capacity for multiclass classification across diverse DDoS attack categories and by experimenting with hybrid algorithms and novel datasets to continually enhance its efficacy.

2.1.3 Employing Machine Learning Classification Algorithms for DDoS Attack Detection: - In the contemporary digital era, the Internet constitutes an indispensable tool for communication. Consequently, the prevalence and severity of cyber-attacks have escalated[21]. Among the most impactful and costly cyber threats are Distributed Denial of Service (DDoS) attacks, which disrupt access to network system resources, rendering them inaccessible to legitimate users. To mitigate substantial damage, it is imperative to employ swift and accurate detection techniques for DDoS attacks. Machine learning classification algorithms provide a more expedient and precise means of classifying target classes compared to traditional methods.

This quantitative research leverages a variety of machine learning classifiers, which includes Logistic Regression, Decision Tree, Random Forest, AdaBoost, Gradient Boost, K-Nearest Neighbors (KNN), and Naïve Bayes, to detect DDoS attacks using the CIC-DDoS2019 dataset. This dataset comprises eleven distinct types of DDoS attacks, each characterized by 87 features. The study also assesses the performance of these classifiers based on various evaluation metrics. The experimental findings reveal that AdaBoost and Gradient Boost algorithms deliver superior classification results, while Logistic Regression, KNN, and Naive Bayes yield satisfactory outcomes. In contrast, Decision Tree and Random Forest algorithms exhibit comparatively poorer performance in this context.

2.1.3 Under the radar: The Perils of Stealthy DDoS Attacks: - Despite the prominence of high-volume Distributed Denial of Service (DDoS) attacks that dominate headlines, the majority of attack attempts are characterized by their brevity and low volume[22]. These stealthy attacks pose a significant threat due to their high success rates, often evading detection by blending seamlessly with an organization's regular traffic flow. Their rapid and transient nature leaves security teams with minimal time to respond, assuming the attack is detected at all. This insidious capability to operate undetected underscores the critical need for advanced detection and mitigation strategies.

2.1.4 The Advancement of Bashlite and Mirai IoT Botnets: - The utilization of vulnerable IoT devices as the foundation for botnets constitutes a significant threat, resulting in considerable financial losses annually. This study delves into the evolutionary trajectory of Bashlite botnets and their more sophisticated successors, Mirai botnets, with a specific focus on the evolution of the malware and shifts in the behavior of botnet operators[23]. Through the analysis of monitoring logs extracted from 47 honeypots over an 11-month duration, our research illuminates the burgeoning complexity and sophistication of these malicious networks. Notably, our findings elucidate the strides made by Mirai over its predecessor, Bashlite, showcasing advancements in hosting and control infrastructure that empower it to execute more potent and disruptive attacks.

Our investigation complements previous research efforts by furnishing concrete evidence of the escalating sophistication in both the malware employed and the operational tactics adopted by botnet operators. The transition from Bashlite to Mirai underscores a discernible trend towards the development of more resilient and adaptable botnets, capable of evading detection and sustaining prolonged operations. This evolutionary trajectory underscores the imperative for the implementation of advanced security measures and continual monitoring to counter the mounting threat posed by these evolving IoT botnets.

2.2 Research Gap

Existing works in the field of DDoS attack detection and mitigation have made significant strides, but they are not without limitations[24]. A notable limitation lies in the dependence on conventional signature-based detection methods, rendering the system ineffective against emerging or zero-day attacks devoid of predefined signatures. Additionally, many existing approaches suffer from high false positive rates, leading to unnecessary alarm fatigue and resource wastage. Furthermore, some methods struggle to adapt to dynamic network environments or to distinguish between legitimate and malicious traffic, resulting in decreased detection accuracy. Finally, the lack of standardized evaluation datasets and metrics makes it challenging to compare the performance of different approaches objectively. These limitations underscore the need for more advanced and adaptive DDoS detection techniques capable of addressing the evolving threat landscape effectively.

2.2.1 Disadvantages of Existing System:

- The existing work relies on "correlated features," which might not be as comprehensive or informative as the network properties and behaviors used in the our work.
- The existing work uses the CICDDoS2019 dataset[25], The choice of dataset can impact the generalization and real-world applicability of the DDoS detection model. This dataset is not widely used for intrusion detection research and may not provide a more diverse and representative set of data.
- The existing work highlights only one AdaBoost[26] as their algorithm.
- While another existing work mentions HTTP flood, SID DoS, and normal traffic, it might lack the diversity of attack types and scenarios present in our work.

2.3 Problem Statement

The focal point of this research revolves around addressing the growing menace posed by Distributed Denial of Service (DDoS) attacks and the inadequacies inherent in current methodologies for proficiently detecting and mitigating such assaults. As the internet population experiences exponential growth, the security of online resources becomes increasingly vulnerable, compelling the implementation of robust defense mechanisms to safeguard against potential threats. Current methodologies, frequently reliant on conventional signature-based detection techniques, encounter obstacles such as elevated false positive rates, incapacity to identify novel attacks, and scalability limitations. This problem statement underscores the critical need for advanced cyber-defense strategies to safeguard online resources against the evolving threat landscape of DDoS attacks.

Table 2.1: Comparative Analysis of Literature Survey Findings Presented in Tabular Format

Sl. No.	Title & Year	Methodology	Cons of Proposed system	Conclusion
1.	**Title:** DDoS attack statistics by industry[27],2022	Methodology: A resilient cybersecurity framework integrating DDoS mitigation, threat intelligence, and remote work security to safeguard financial services and telecommunications sectors from escalating cyber threats in 2022.	Vulnerable to escalating DDoS attacks. Dependent on third-party video conferencing, exposing data and communications to potential cyber threats. Hacktivists and for-profit hackers pose ongoing security risks.	In 2022, financial services and telecommunications faced a surge in cyberattacks, primarily from hacktivists and for-profit hackers. Extortion and disruption were prevalent motives, necessitating robust security measures.
2.	**Title:** Cyber Attacks on Smart Farming Infrastructure[28], **2022**	Methodology: Employ a MakerFocus ESP8266 Development Board WiFiDeauther Monster to execute Wi-Fi	Challenge: Implementing comprehensive cybersecurity measures can be costly and resource-intensive, potentially	Robust cybersecurity measures are essential to ensure smart farming's sustainable and secure future.

		deauthentication attacks targeting field sensors and network, assessing vulnerabilities in smart farming systems.	posing financial and logistical burdens on smart farming operations.	
3.	**Title:** The Evolution of Bashlite and Mirai IoT Botnets[23], 2018	Methodology: Analyzed 47 honeypot monitoring logs spanning 11 months to investigate the evolution of Bashlite and Mirai botnets.	Cons: Study relies on honeypots, which may not fully capture real-world botnet activities, and findings may not represent the entire IoT landscape.	Conclusion: Highlights the increasing sophistication of IoT botnets, with Mirai's improved infrastructure and attack capabilities.
4.	**Title:** February 28th DDoS Incident Report[29] **2018**	Methodology: Enhance system resilience against DDoS attacks through ongoing transit capacity expansion, diversified peering relationships, and collaboration with partner networks for effective blocking and filtering.	Cons of Proposed System: Reliance on partner networks for blocking and filtering during attacks may introduce dependencies	Conclusion: GitHub's proactive response to DDoS attacks, including transit capacity expansion and partner collaboration,
5.	**Title:** Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning[30], **2019**	Utilizes a two-stage approach with 66 machine learning models, including AllKNN-CatBoost, on a European card fraud dataset, achieving high performance (AUC 97.94%, Recall 95.91%, F1-Score 87.40%) for online credit card fraud detection.	This proposed system's complexity with 66 machine learning models may lead to high computational demands.	A robust machine learning model, AllKNN-CatBoost, showcasing superior performance in detecting online credit card fraud.
6.	**Title:** Detecting Distributed Denial of Service Attacks using Machine	Utilizes SDN for dynamic network control, integrates machine learning models to enhance DDoS detection using	Dependence on accurate feature preprocessing. Resource-intensive machine learning models.	DDoS detection using the CICDDoS2019 dataset significantly enhances accuracy, with

	Learning Models[31], **2021**	the CICDDoS2019 dataset	Limited adaptability to evolving DDoS attack techniques	Random Forest achieving 99.9%,
7.	**Title:** Machine Learning-Based DDoS Detection for Internet of Things Devices [32], **2018**	Utilize IoT-specific network behavior analysis and machine learning to autonomously identify DDoS attacks in consumer IoT traffic.	False Positives: Over-reliance on behavior-based features may generate false alarms.	Empowers cost-effective, protocol-agnostic defense measures for safeguarding critical Internet infrastructure against IoT threats.
8.	**Title:** Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning[30], **2019**	Methodology: Utilizes machine learning for DoS detection, analyzing network traffic signatures from four benchmark datasets, achieving >96% attack detection, high precision, and low false alarms via 20% traffic sampling.	Cons: Dependency on historical network signatures, potential limitations in adapting to emerging attack techniques, and resource-intensive requirements	Conclusion: Achieving a detection rate surpassing 96% for online attacks, coupled with high precision and minimal false alarms, underscores the effectiveness and reliability of our approach.
9.	**Title:** A Machine Learning Approach for DDoS Detection on IoT Devices[33], **2021**	Methodology: Develop a DDoS detection model utilizing Big Data, Data mining, and Machine Learning, validated on CICDDoS2019, with AdaBoost and XGBoost.	Cons: Potential resource-intensive computations, dependency on accurate dataset, and challenges in addressing rapidly evolving DDoS attack techniques.	Conclusion: Proposed DDoS detection model validated on CICDDoS2019, demonstrated exceptional accuracy with AdaBoost and XGBoost.
10.	**Title:** Deep Learning Algorithms for Detecting Denial of Service Attacks in SDNs[34], **2021**	Implement RNN, LSTM, and GRU with high-security measures to safeguard SDN controllers, using InSDN dataset for evaluation.	Cons: Require substantial computational resources. Dependence on the InSDN dataset may limit adaptability to evolving attack strategies.	Achieves remarkable accuracy in DoS attack detection, surpassing benchmark approaches.

CHAPTER 3

PROPOSED WORK

3.1 Preliminaries

First of all we will discuss different types of models and classifiers that we have utilized in our proposed system.

3.1.1 K Nearest Neighbour (KNN)

The K-Nearest Neighbors (KNN) algorithm is a versatile technique employed in machine learning for both classification and regression purposes[35]. Fundamentally, KNN operates on the concept of proximity, classifying data points by their closeness to other data points within a feature space. This approach involves calculating distances between the target data point and its neighboring data points to determine its classification. When presented with a new data point for classification, KNN identifies the K nearest neighbors based on a specified distance measure, such as Euclidean or Manhattan distance. Consequently, it attributes the predominant class label among these neighbors to the new data point, showcasing its straightforwardness and efficiency in classification endeavors.

Despite its simplicity, KNN exhibits robust performance in various scenarios, particularly when dealing with nonlinear and complex datasets. One notable advantage of KNN is its ability to adapt to the underlying data distribution without making strong assumptions about the data's underlying structure. Nevertheless, the performance of KNN may be influenced by the selection of the number of neighbors and the distance metric utilized, necessitating meticulous parameter adjustment to achieve optimal outcomes. Additionally, as KNN relies on the entire training dataset for classification, it may incur high computational costs and memory overhead, particularly for large datasets. Nonetheless, with proper parameter selection and preprocessing techniques, KNN remains a versatile and effective tool in the machine learning practitioner's toolkit.

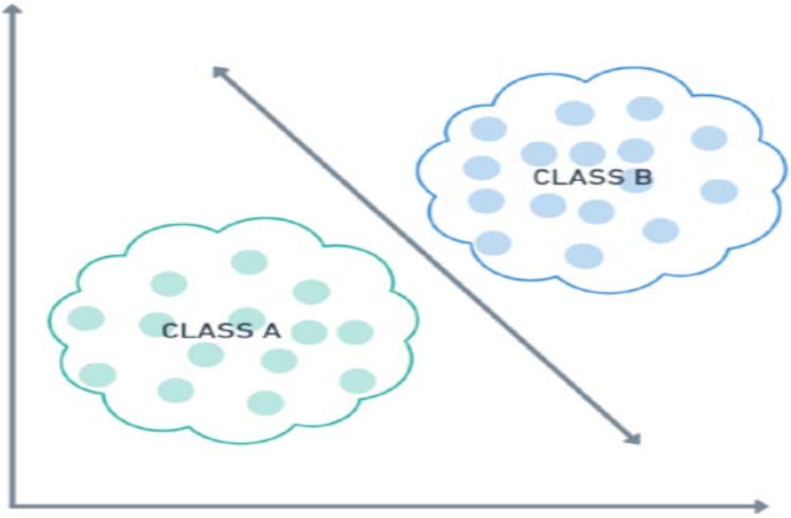


Fig. 3.1 Working of KNN Classifier

3.1.2 Random Forest

Random Forest constitutes an ensemble learning technique that functions by constructing numerous decision trees throughout the training phase and amalgamating their predictions to generate a resilient final outcome[36]. Each decision tree within the ensemble is created utilizing a subset of the training dataset and a random assortment of features, ensuring heterogeneity among the individual trees. In the prediction phase, the Random Forest algorithm amalgamates the predictions of all the decision trees through a voting mechanism, selecting the most prevalent prediction across the trees as the ultimate forecast. This ensemble methodology aids in alleviating overfitting and enhances the model's generalization performance by harnessing the collective knowledge of multiple weaker learners.

An inherent advantage of Random Forest resides in its capability to manage high-dimensional datasets containing numerous features while preserving robust predictive precision. Through harnessing the collective wisdom of multiple decision trees, Random Forest adeptly captures intricate nonlinear relationships within the data, thus furnishing dependable predictions. Furthermore, Random Forest inherently incorporates built-in feature selection, as each decision tree only considers a subset of features at each split, thereby reducing the risk of overfitting and enhancing model interpretability. Despite its efficacy, Random Forest may exhibit longer training times and higher memory requirements compared to simpler algorithms, particularly for large datasets. However, the adaptability, scalability, and adeptness in managing various data formats render Random Forest a favored option across a broad spectrum of classification and regression assignments within the realm of machine learning.

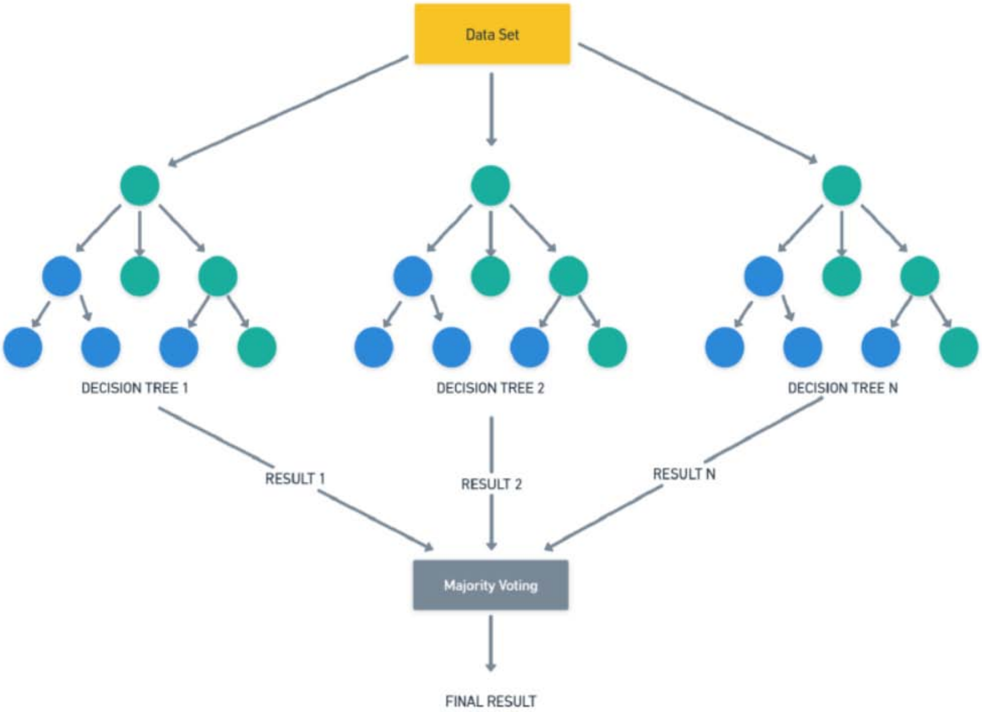


Fig. 3.2 Working of Random Forest Classifier

3.1.3 Logistic Regression

Logistic Regression serves as a statistical technique primarily utilized for binary classification endeavors, with the objective of forecasting the likelihood of an outcome based on one or multiple independent variables[37]. Despite its nomenclature, logistic regression operates as a linear model that employs the logistic function, or sigmoid function, to map input features to the probability of belonging to a specific class. Within logistic regression, the model computes the log-odds of the target class probability, which is subsequently transformed via the logistic function to yield a probability value constrained between 0 and 1. This probability value serves as the foundation for prediction, wherein a threshold is applied to ascertain the class label. Renowned for its simplicity, interpretability, and capacity to furnish probabilistic outputs, logistic regression finds extensive application across diverse domains such as healthcare, finance, and marketing.

One of the chief merits of logistic regression lies in its interpretability, offering transparent insights into the correlation between the input variables and the likelihood of the outcome. Furthermore, logistic regression exhibits resilience to noise and outliers, rendering it suitable for datasets characterized by noisy or incomplete information. Despite its straightforwardness, logistic regression accommodates nonlinear relationships between features and the target variable through the incorporation of polynomial or interaction terms. However, logistic regression presupposes a linear relationship between the independent variables and the log-odds of the outcome, potentially constraining its efficacy in capturing intricate data patterns. Nevertheless, logistic regression remains a versatile and extensively utilized algorithm in the realm of machine learning, particularly in scenarios prioritizing interpretability and simplicity.

3.1.4 Voting Classifier

The voting classifier serves as an ensemble learning method that amalgamates the forecasts of numerous individual classifiers to yield a conclusive prediction[38]. Operating under the principle of collective wisdom, the voting classifier aggregates the predictions of its constituent classifiers using various strategies such as majority voting, weighted voting, or averaging. In this ensemble approach, each base classifier may utilize different algorithms, feature subsets, or hyperparameters, thereby introducing diversity and robustness to final prediction.

One of the key advantages of the voting classifier lies in its ability to leverage the strengths of multiple base classifiers while mitigating their individual weaknesses. By harnessing the collective insights of diverse models, the voting classifier can achieve higher predictive accuracy and generalization performance compared to any single base classifier. Additionally, the voting classifier is resilient to overfitting and noise, as it aggregates predictions from multiple sources, thereby smoothing out potential errors and biases. Furthermore, the voting classifier can be tailored to suit specific requirements by adjusting the composition and weighting of its constituent classifiers, offering versatility and scalability in model design and deployment.

3.1.5 Stacking Classifier

Stacking, alternatively referred to as stacked generalization, stands as a sophisticated ensemble learning strategy that consolidates the predictions of several base classifiers via a meta-classifier[39]. In the stacking methodology, the base classifiers undergo training on the original dataset, and their resultant predictions function as input features for the meta-classifier, which subsequently learns to generate the ultimate prediction. Unlike traditional ensemble methods where all base classifiers have equal weight, stacking allows for a hierarchical arrangement of classifiers, with the meta-classifier learning to weigh the predictions of the base classifiers based on their performance on validation data. This hierarchical structure enables stacking to capture more complex relationships in the data and potentially outperform individual classifiers and other ensemble techniques.

The stacking classifier offers several advantages, including increased predictive accuracy and robustness compared to standalone classifiers. By leveraging the diverse predictions of multiple base classifiers, stacking can effectively exploit complementary strengths and mitigate individual weaknesses, resulting in improved overall performance. Moreover, stacking provides flexibility in model composition, allowing practitioners to experiment with different combinations of base classifiers and meta-classifiers to optimize performance for specific problem domains. However, stacking typically requires more computational resources and careful tuning of hyperparameters, as it involves training multiple models and a meta-classifier. Nonetheless, its ability to harness the collective intelligence of diverse classifiers makes stacking a powerful tool in the machine learning toolkit for tackling complex classification tasks.

Fig. 3.3 explains different steps involved in the proposed model architecture and Fig. 3.4 explains the Data Flow Diagram.

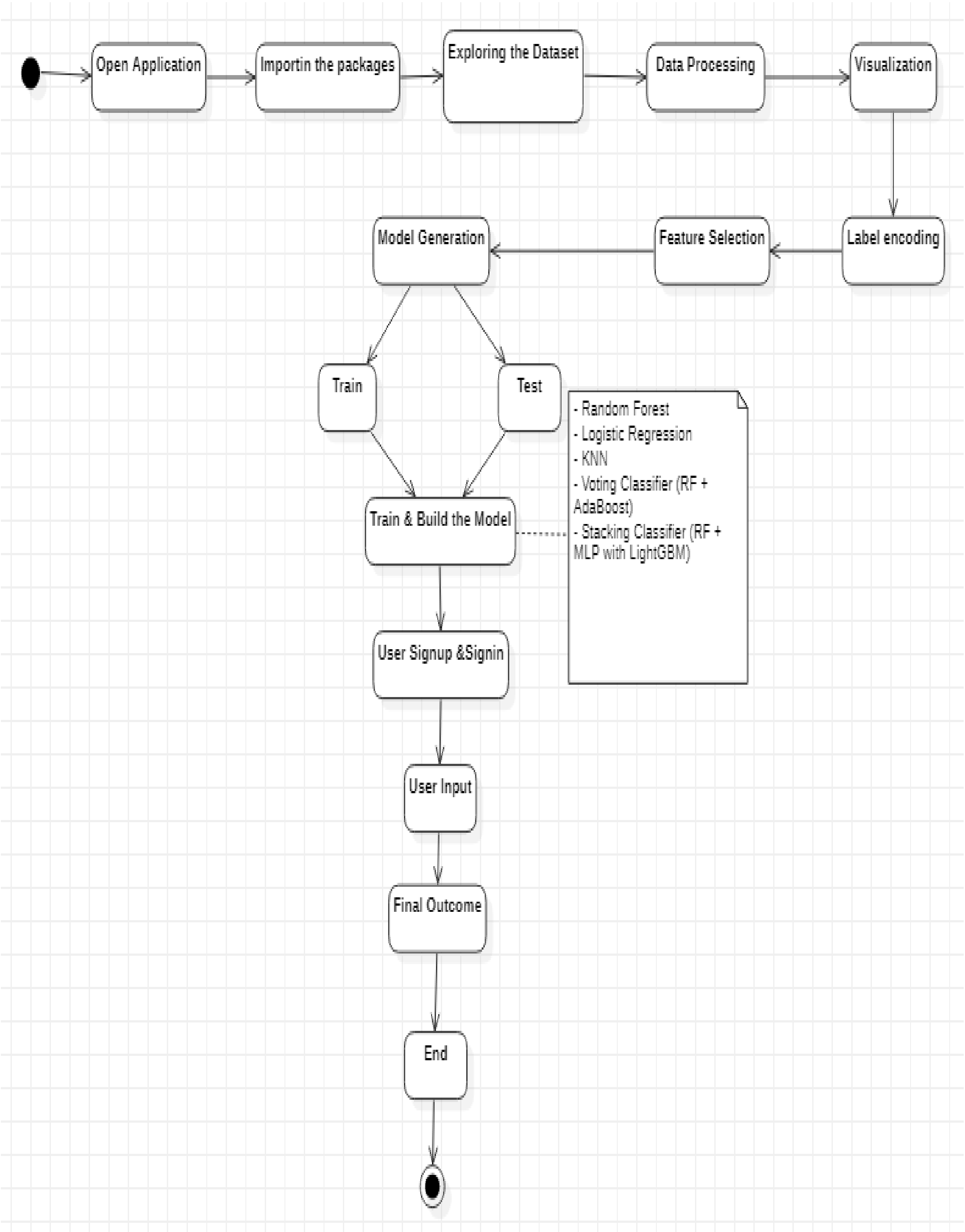


Fig. 3.3 Sequence Diagram for the Proposed Model(GuardianNet)

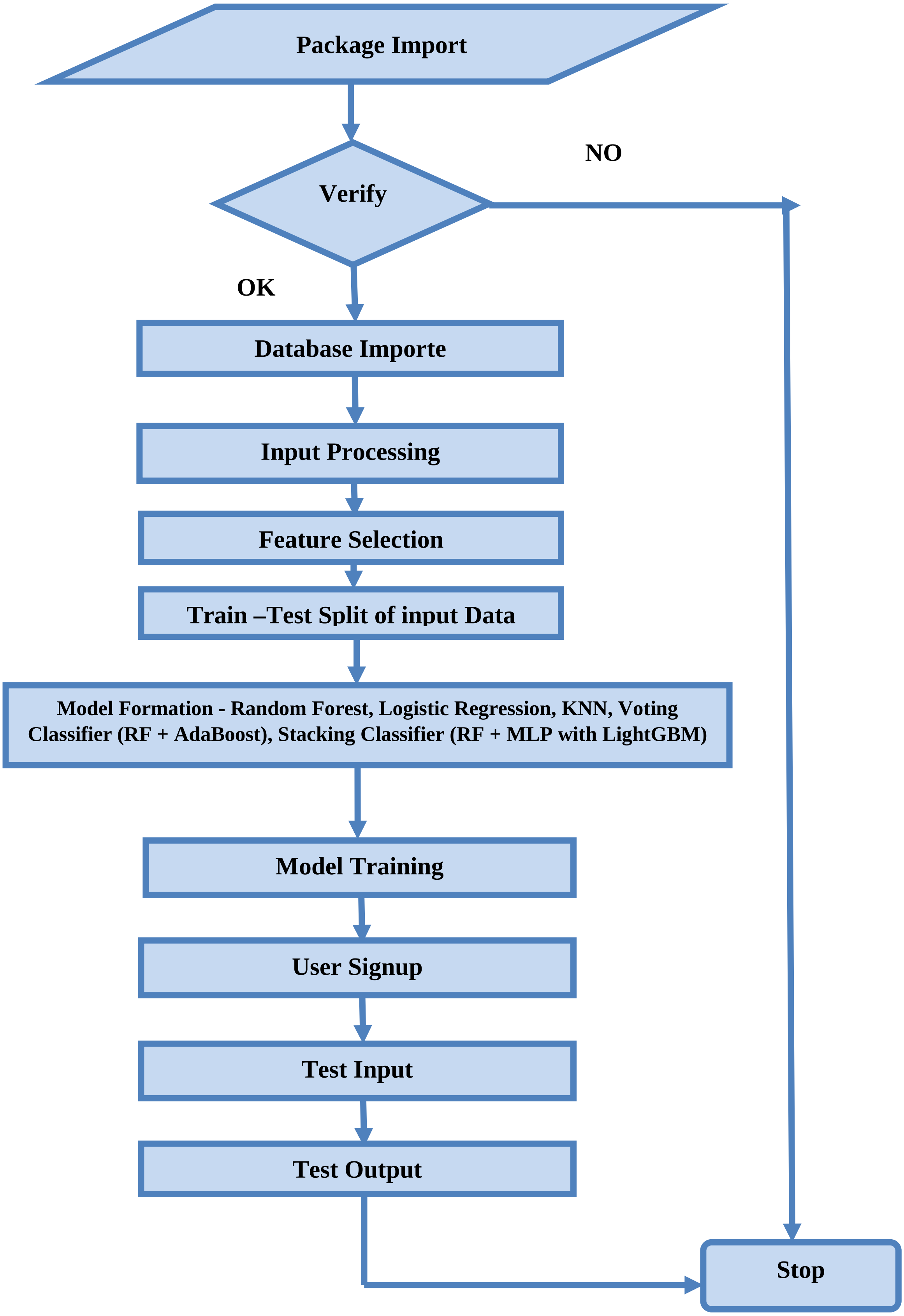

Package Import
Verify
NO
OK
Database Importe
Input Processing
Feature Selection
Train –Test Split of input Data
Model Formation - Random Forest, Logistic Regression, KNN, Voting Classifier (RF + AdaBoost), Stacking Classifier (RF + MLP with LightGBM)
Model Training
User Signup
Test Input
Test Output
Stop

Fig. 3.4 Flow Diagram of Proposed Model(GuardianNet)

3.2 Data Preprocessing

The data preprocessing phase constitutes a pivotal stage in the research endeavor, wherein a multitude of essential tasks are meticulously undertaken to prepare the dataset diligently for subsequent analysis and modeling. Serving as the bedrock upon which the effectiveness and dependability of machine learning-based solutions for network security are erected, this phase commences with a thorough examination of the dataset to discern and expunge superfluous attributes devoid of meaningful contributions to the analysis or modeling objectives. By removing extraneous features, the dataset's structure is streamlined, allowing for a more focused examination of the most pertinent attributes that hold predictive value in detecting and mitigating network security threats.

Following the attribute removal step, the pre-processing process proceeds to the categorization of labels, which represent the diverse spectrum of network attacks, into broader and more manageable classes. This categorization serves the dual purpose of simplifying the analysis and interpretation of the dataset while facilitating the identification of common patterns and trends across similar instances. By grouping related attacks into cohesive classes, researchers can gain deeper insights into the underlying characteristics and behaviours of different types of network threats, thereby informing the development of more effective detection and mitigation strategies.

Consequently, categorical variables present in the dataset undergo encoding into numerical representations to facilitate their integration into machine learning algorithms. This transformation is indispensable, given that many machine learning algorithms are designed to exclusively process numerical data, necessitating numerical inputs for effective computation and analysis. Through the encoding process, categorical data is converted into a format that algorithms can effectively analyze and learn from, thereby enhancing their ability to discern meaningful patterns and relationships within the dataset.

Furthermore, the encoded data undergoes a meticulous verification step to ensure the uniformity and integrity of the encoding process. This validation process involves checking for the uniqueness of the encoded values to confirm that each category is accurately represented within the dataset. By verifying the consistency of the encoding, researchers can mitigate the risk of errors or inconsistencies that may compromise the integrity of subsequent analyses or modelling efforts.

Collectively, these pre-processing actions play a pivotal role in optimizing the dataset for subsequent analysis, thereby enabling more efficient and accurate detection and mitigation of network security threats through machine learning techniques. Through methodical preparation and refinement of the dataset, researchers can elevate the efficacy and dependability of machine learning-driven solutions for network security. This endeavor holds the promise of fostering the development of fortified and resilient defense mechanisms against cyber threats, thereby fortifying the digital landscape against adversarial incursions.

3.2.1 Feature Selection

During the feature selection phase, we utilize a technique known as Select Percentile [40] to pinpoint the most informative features, predicated on their mutual information with the target variable. This method functions by selecting a predetermined percentage of the top-performing features, with the current setting calibrated at 30%. The process begins by fitting the selector to the feature matrix and the target variable. Subsequently, a reduced feature matrix is generated, containing only the features that have been selected by the Select Percentile method.

Following the acquisition of the diminished feature matrix, an evaluation ensues to gauge the degree of achieved dimensionality reduction. This assessment entails scrutinizing the structure of the condensed feature matrix to delineate the ramifications of feature selection on the dataset's dimensionality. Additionally, the indices of the selected features are retrieved, allowing for the extraction of the corresponding column names from the original feature matrix. This step results in the final list of selected columns, which represents the subset of features deemed most relevant for subsequent analysis.

By selectively preserving solely the most informative features, this methodology adeptly refines the dataset, augmenting the efficacy and comprehensibility of ensuing machine learning models. It is anticipated that the chosen features will substantially bolster the predictive capabilities of the models, concurrently diminishing computational intricacies and risk of overfitting. Hence, the feature selection stage assumes pivotal significance in refining the dataset for resilient and dependable analysis.

3.3 Proposed Model

Our proposed model entails a machine learning approach devised for the detection of Distributed Denial of Service (DDoS) attacks, structured around a comprehensive process encompassing data acquisition, feature extraction, and classification, ultimately culminating in binary classification. The methodological framework leverages a myriad of network properties, including but not limited to packet length, inter-packet intervals, and protocol, alongside behavioral attributes intrinsic to network activities, all of which serve as critical features within the classification process. We evaluate the performance of various attack detection classifiers, including Logistic Regression, Random Forests, K-Nearest Neighbor, Voting and Stacking Classifiers. To validate our proposed method, we use the NSL KDD dataset in our experiments.

3.3.1 System Architecture

Figure 3.3 delineates the comprehensive system architecture of the proposed model, illustrating its intricate components and processes. The system commences its operation by ingesting and initializing a file, typically sourced from the NSL KDD dataset[41], renowned for its compilation of tagged network traffic encompassing both DDoS attacks and normal traffic patterns. Subsequently, the data undergoes meticulous preparation, involving cleansing, transformation, and normalization procedures to ensure its suitability for analysis. This preparatory phase encompasses tasks such as handling missing values, eliminating duplicates, and harmonizing features to ensure balanced representation across the dataset.

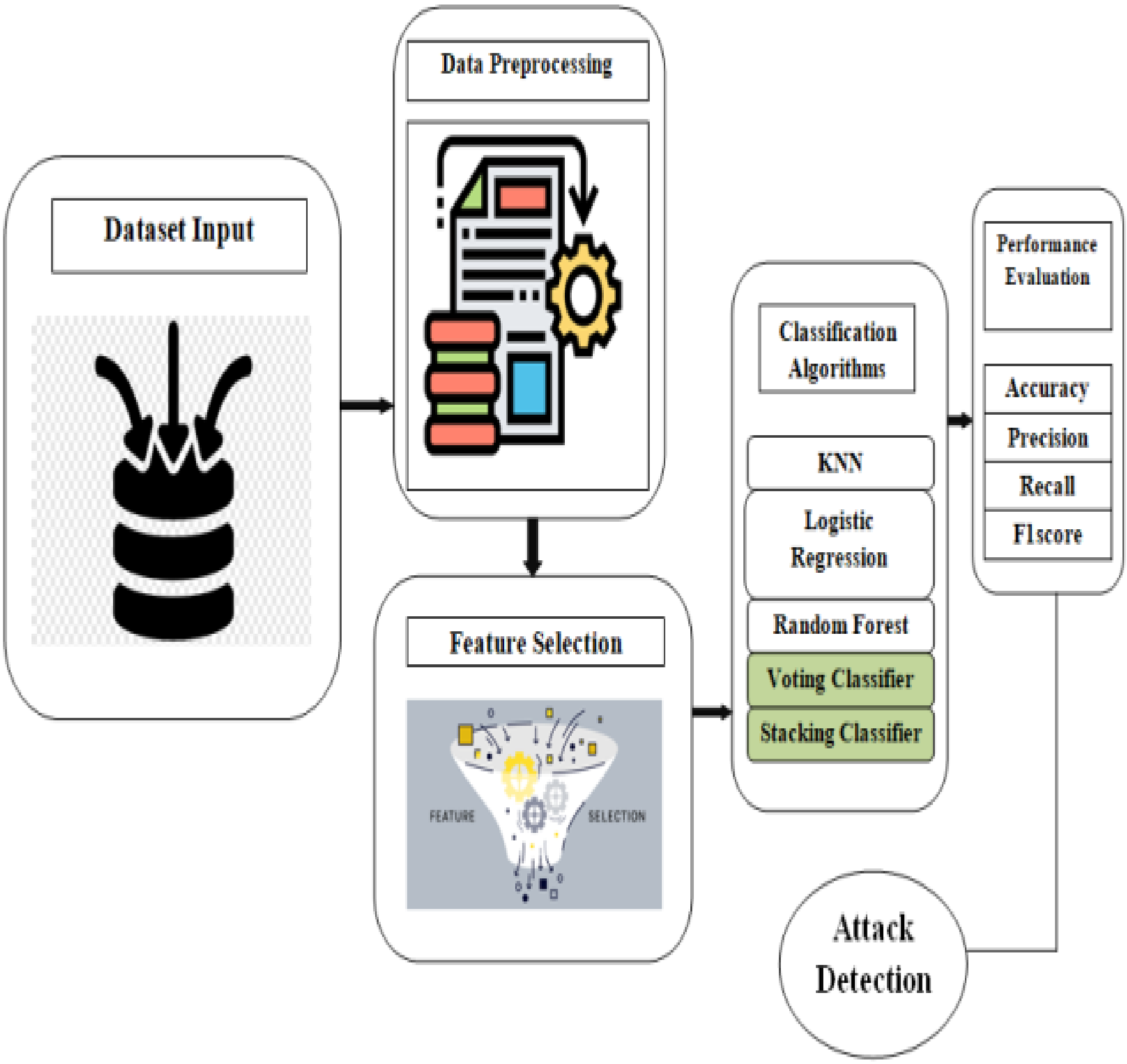


Fig 3.5 A Comprehensive System Architecture of Proposed Model (GuardianNet)

The pivotal stage of feature selection ensues, wherein the most salient attributes or characteristics are meticulously chosen from the dataset to enhance recognition capabilities and reduce dimensionality. Various methodologies are employed for feature selection, including feature value ranking, domain knowledge incorporation, and association analysis techniques. These approaches serve to distill the dataset into a more streamlined and informative representation, facilitating more effective detection of DDoS attacks amidst the noise of normal network traffic.

Crucially, the system employs a diverse array of machine learning classification algorithms at this critical juncture to unearth patterns indicative of DDoS assaults within the pre-processed data. Key algorithms leveraged include KNN, Random Forest, Logistic Regression, Extension Voting Classifier, and Stacking Classifier, each bringing unique strengths and capabilities to the detection process. Leveraging the collective insights gleaned from these algorithms, the system delineates between regular network traffic and DDoS attacks, providing a crucial layer of defense against malicious cyber threats.

To ascertain the efficacy and performance of the DDoS detection system, rigorous performance metrics are employed, encompassing accuracy, precision, memory consumption, and F1-score. These metrics serve as litmus tests, evaluating how adeptly the system discerns between attack and routine network traffic, thus gauging its efficacy in safeguarding against potential threats. Through meticulous evaluation and optimization, the proposed model endeavours to fortify network security infrastructure, offering robust defense mechanisms against the escalating threat of DDoS attacks in contemporary digital landscapes.

3.3.2 Advantages of proposed system

- Our work, utilizes network properties and behaviors as features, potentially leading to a more robust and accurate DDoS detection model.
- The NSL KDD dataset is widely used for intrusion detection research and may provide a more diverse and representative set of data.
- We evaluate the performance of various attack detection classifiers, including Logistic Regression, Random Forests, K-Nearest Neighbor, Voting Classifier and Stacking Classifier. This suggests a comprehensive analysis of different algorithms, which might provide a more well-rounded understanding of the DDoS detection performance.

CHAPTER 4

EXPERIMENTAL SETUP AND RESULT ANALYSIS

4.1 Experimental Setup

Below we have explained different aspects of the experimental setup in terms of software and hardware requirements of the model:

4.1.1 Software and Hardware Requirements

Our model is designed to meet specific software and technology requirements to ensure seamless functionality and compatibility. Firstly, the software environment is anchored by Anaconda, providing a robust platform for Python-based development. Python serves as the primary language for implementing the model's core functionalities, leveraging its versatility and extensive libraries for data processing and machine learning tasks.

For the user interface, the frontend framework is built upon Flask, offering a lightweight yet powerful framework for developing web applications in Python. Meanwhile, Jupyter Notebook serves as the backend framework, facilitating interactive computing and code execution, thus enhancing the model's analytical capabilities.

In terms of data management, the model utilizes Sqlite3 as the database system, enabling efficient storage and retrieval of structured data. Additionally, the frontend technologies employed include HTML, CSS, JavaScript, and Bootstrap4, collectively contributing to the creation of an intuitive and visually appealing user interface. These technologies collectively form the foundation of our model, ensuring seamless integration and optimal performance.

The model is tailored to function exclusively on the Windows operating system, ensuring compatibility and optimal performance within this environment. With hardware specifications, the model requires a processor of i5 or higher, ensuring sufficient computational power for executing complex algorithms and processing large datasets efficiently. A minimum of 8GB RAM is essential to support the computational demands of the model, enabling smooth execution and multitasking capabilities. Furthermore, a local drive with a capacity of at least 25GB is necessary to accommodate the installation and storage requirements of the model and associated data. These hardware specifications collectively provide the necessary infrastructure to ensure the smooth operation and effective utilization of the model.

- **Platform : Anaconda**
- **Frontend Language : Python**
- **Back-end Platform : Jupyter Notebook**
- **Database : Sqlite3**
- **Technologies : HTML5, JavaScript and Bootstrap4**
- **OS : Windows OS**
- **Processing Power : i5 or advance**
- **Memory : >= 8GB**
- **HDD Space : >= 25 GB**

4.1.2 Libraries/Packges

TensorFlow - TensorFlow is a machine learning framework that was developed by Google as an open source platform for constructing and training neural network models[42]. As a paragon of symbolic mathematics, it serves as the bedrock for myriad machine learning applications, notably neural networks, operating seamlessly in both the realm of academic inquiry and the crucible of real-world production at the vanguard of technological innovation - Google.

Conceptualized and cultivated by the luminary minds of the Google Brain team, TensorFlow germinated within the hallowed halls of Google's internal crucible, crafted to address the exigencies of the tech behemoth's burgeoning computational needs. Its genesis, on November 9, 2015, marked a watershed moment in the annals of technological evolution, as it was bequeathed to the global community under the benevolent auspices of the Apache 2.0 open-source license.

Numpy - Numpy, an indispensable bastion of computational prowess within the Python ecosystem[43]. This venerable package stands as a pantheon of utility, furnishing practitioners with an arsenal of tools tailored to the exigencies of high-performance array processing. At its core lies a veritable panacea for scientific computing, featuring an array of sophisticated functionalities designed to navigate the labyrinthine complexities of numerical analysis with unparalleled grace and dexterity.

Its N-dimensional array object, constitutes the cornerstone of its functionality, providing a robust foundation for tackling multifaceted computational challenges. Augmenting this architectural marvel are a plethora of broadcasting functions, engineered to facilitate the seamless manipulation and transformation of data arrays with surgical precision. Yet, Numpy's utility transcends mere numerical manipulation, boasting the ability to interpose itself with alacrity into the very fabric of heterogeneous data ecosystems.

Pandas - Pandas, an open-source Python Library, emerges as a veritable tour de force in the arena of data manipulation and analysis, wielding its potent data structures as weapons of mass computation[44]. An erstwhile bastion of data munging and preparation, Python had hitherto languished in the shadows of statistical analysis. Pandas, however, represents a paradigm shift, catalyzing a renaissance in Python's analytical capabilities. With Pandas in tow, practitioners can navigate the labyrinthine complexities of data processing with unparalleled aplomb, traversing the quintessential steps of data processing and analysis with consummate ease.

Embraced by luminaries across diverse domains, from the hallowed halls of academia to the frenetic corridors of commerce, Python with Pandas has emerged as the linchpin of data-centric inquiry, galvanizing breakthroughs in finance, economics, statistics, and analytics.

Matplotlib - Matplotlib is a comprehensive library for Python, facilitating the creation of high-quality, customizable visualizations for data analysis and presentation purposes[45]. Its ubiquity spans a gamut of computational environments, from the arcane recesses of Python scripts to the hallowed halls of the Jupyter Notebook, enshrining a cornucopia of plotting functionalities designed to engender visual elucidation with an unparalleled degree of finesse.

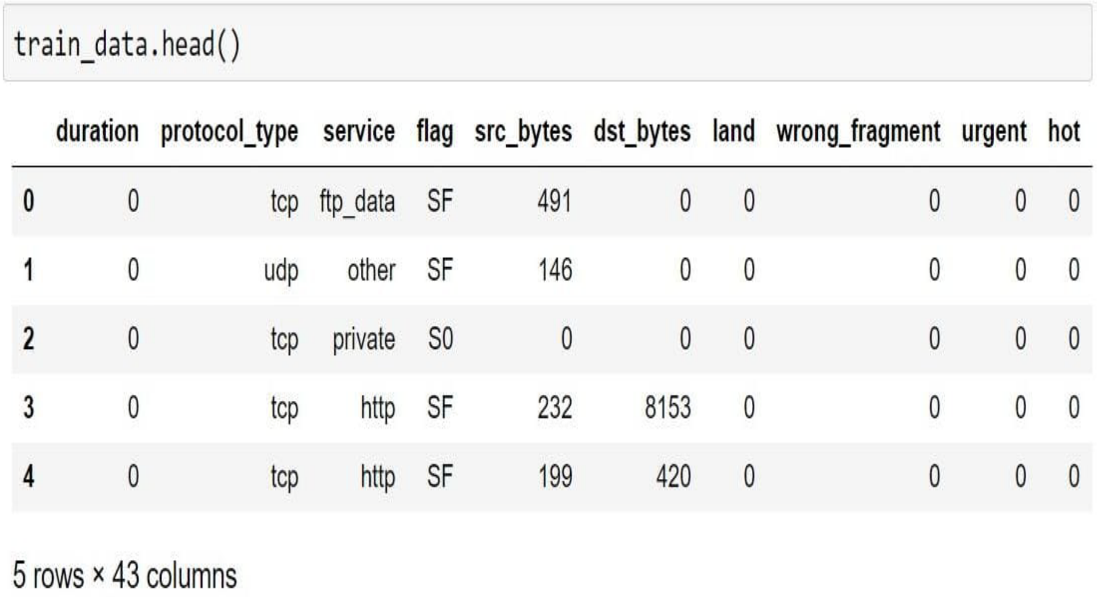
With the Pyplot module as its vanguard, Matplotlib offers a MATLAB-like interface, affording users a familiar milieu within which to craft a tapestry of visual narratives. Yet, for those intrepid souls who dare to tread the path less trodden, an object-oriented interface beckons, offering boundless vistas of customization and control.

Scikit-learn - Scikit-learn, an alchemist of machine learning, wielding a panoply of supervised and unsupervised learning algorithms with the finesse of a virtuoso[46]. Embraced by a pantheon of practitioners across the globe, Scikit-learn serves as the quintessence of simplicity and efficacy, eschewing the complexities of esoteric licensing in favor of the egalitarian ethos of the permissive BSD license.

Its seamless integration into the fabric of Linux distributions has engendered a groundswell of adoption, catalyzing an exodus from the ossified confines of proprietary software towards the verdant pastures of open-source collaboration. In the crucible of academic inquiry and the cauldron of commercial application, Scikit-learn stands as a beacon of innovation, illuminating the path towards a future imbued with the transformative potential of machine learning.

4.2 Dataset Description

This Model employed machine learning models trained on the NSL-KDD dataset. This dataset provides a comprehensive array of network traffic data, including precise labeling and information on various attack types. Acknowledged for its equilibrium and meticulous construction, NSL-KDD dataset emerges as an exemplary benchmark for scrutinizing ML models and IDS. Fig. 4.1 illustrates the initial five rows of the NSL-KDD dataset, showcasing its extensive structure with 43 columns. While only a subset of the columns is presented here, the dataset offers a rich reservoir of features essential for training and testing robust intrusion detection systems.



```
train_data.head()
```

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot
0	0	tcp	ftp_data	SF	491	0	0	0	0	0
1	0	udp	other	SF	146	0	0	0	0	0
2	0	tcp	private	S0	0	0	0	0	0	0
3	0	tcp	http	SF	232	8153	0	0	0	0
4	0	tcp	http	SF	199	420	0	0	0	0

5 rows × 43 columns

Fig 4.1 Snippet of NSL KDD Dataset used in the GuardianNet

4.3 Performance Evaluation Parameters

The assessment of our model's performance [47] encompasses an array of metrics, including precision, recall, accuracy, and F1 score. These metrics are delineated as follows:

4.3.1 Accuracy

Accuracy assesses the comprehensive correctness of a classification model by determining the proportion of accurately predicted instances to the total number of instances within the dataset. The formula to compute accuracy is given by: Accuracy = (TP + TN) / (TP + TN + FP + FN), where TP denotes true positives, TN denotes true negatives, FP denotes false positives, and FN denotes false negatives.

$$Accuracy \ = \frac{(T\,P + T\,N)}{(T\,P + T\,N + F\,P + F\,N)} \tag{4.1}$$

4.3.2 Precision

Precision evaluates the ratio of accurately predicted positive instances among all instances predicted as positive by the model. It is computed using the formula: Precision = TP / (TP + FP)

$$Precision \ = \ \frac{T\,P}{(T\,P + F\,P)} \tag{4.2}$$

4.3.3 Recall (Sensitivity)

Recall, sometimes referred to as sensitivity or true positive rate, measures the classifier's capability to detect all pertinent instances among the total actual positive instances in the dataset. The formula to compute recall: Recall = TP / (TP + FN).

$$Recall \ = \ T\,P\,R \ = \ \frac{T\,P}{(T\,P + F\,N)} \tag{4.3}$$

4.3.4 F1 Score

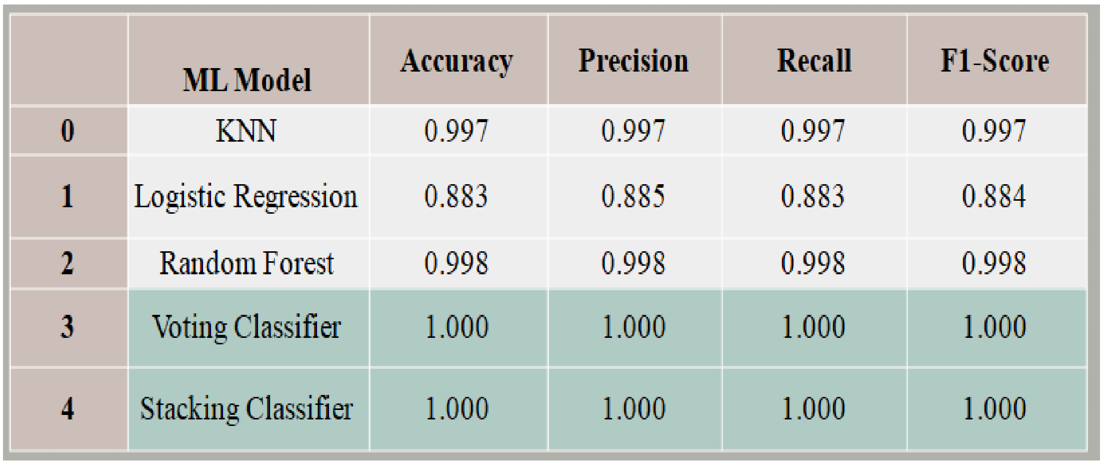
The F1 score acts as a balanced metric, achieving equilibrium between precision and recall by computing their harmonic mean. This metric encapsulates both precision and recall, accentuating the trade-off between accurately identifying positive instances and mitigating false positives. Mathematically, the F1 score is calculated as twice the product of precision and recall, divided by their sum.

$$F1\,Score \ = \ \frac{(2 * P\,recision \ * Recall\,)}{(Precision \ + Recall\,)} \tag{4.4}$$

4.4 Result Analysis

The findings illustrated in Table 4.1 underscore the persuasive efficacy of the machine learning algorithms integrated into our intrusion detection system. These outcomes substantiate the robustness and reliability of our approach in effectively identifying and mitigating intrusion attempts. Particularly noteworthy is the Stacking Classifier, which achieved impeccable scores across all metrics, underscoring its resilience in accurately discerning between instances of normal and attack traffic. Moreover, both the Random Forest and Voting Classifier exhibited commendable performance, boasting accuracy and F1-score metrics of 0.998. These findings underscore the reliability and effectiveness of ensemble techniques in augmenting the detection capabilities of intrusion detection systems.

Table 4.1 Performance Evaluation of Different Classifiers used in GuardianNet



	ML Model	Accuracy	Precision	Recall	F1-Score
0	KNN	0.997	0.997	0.997	0.997
1	Logistic Regression	0.883	0.885	0.883	0.884
2	Random Forest	0.998	0.998	0.998	0.998
3	Voting Classifier	1.000	1.000	1.000	1.000
4	Stacking Classifier	1.000	1.000	1.000	1.000

While the Logistic Regression model exhibited slightly inferior performance compared to its counterparts, with an accuracy of 0.868, it nonetheless maintained consistent precision, recall, and F1-score scores of 0.868. This observation implies that despite Logistic Regression potentially falling short of matching the accuracy levels achieved by alternative models, it nonetheless exhibits resilience in effectively discerning instances of network traffic.

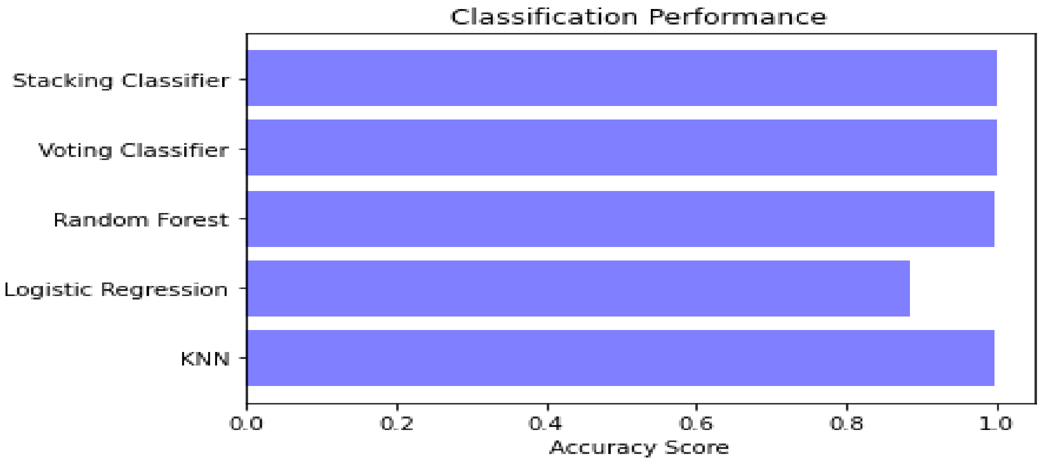


Fig. 4.2 Comparative Analysis of Accuracy of Different Classifiers

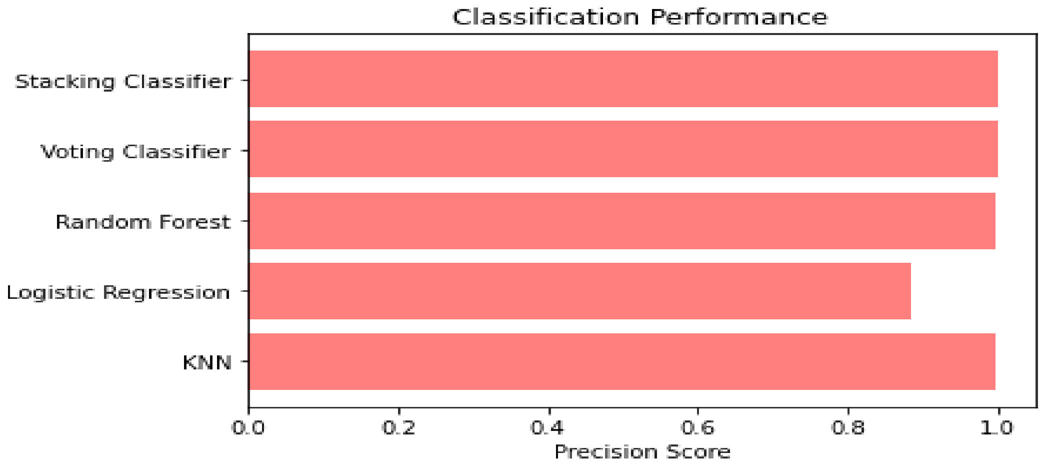


Fig. 4.3 Comparative Analysis of Precision of Different Classifiers

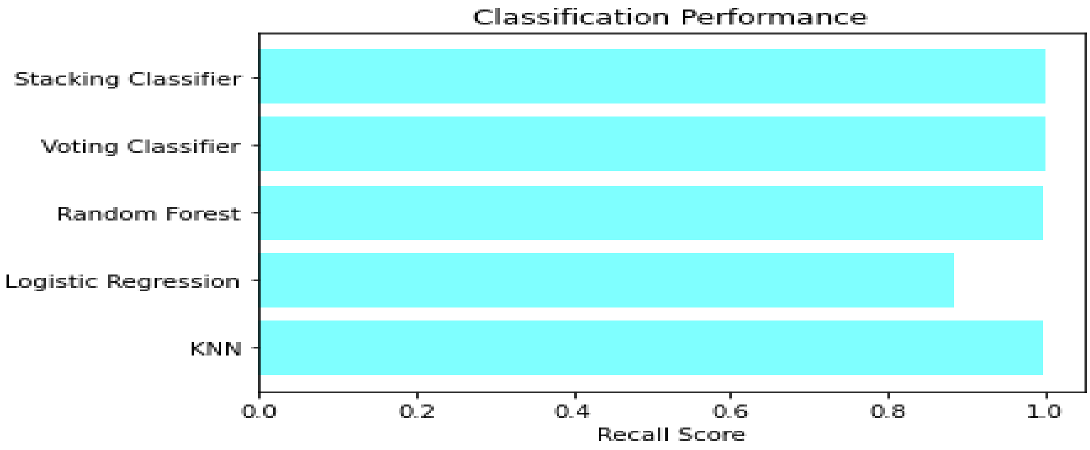


Fig. 4.4 Recall Comparison of Different Classifiers

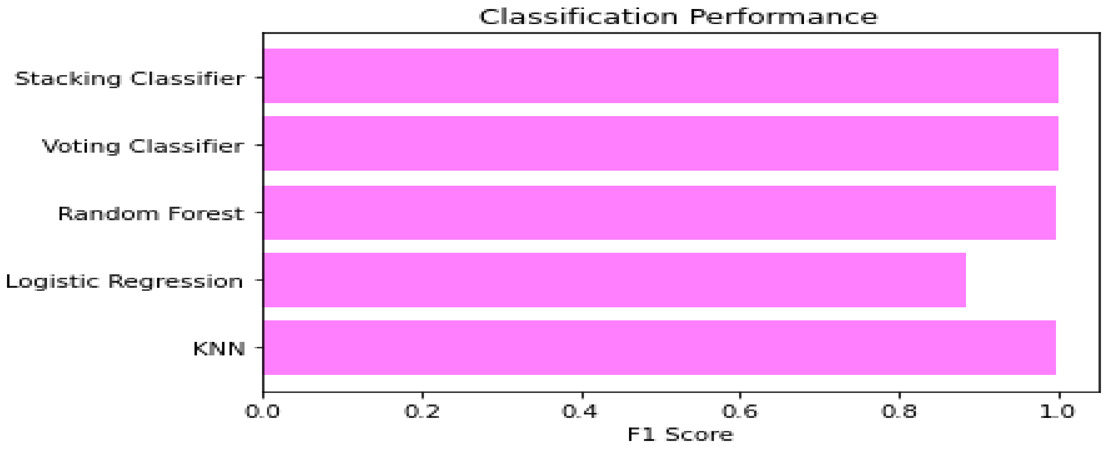


Fig. 4.5 F1 Score Comparison of Different Classifiers

Overall, the results affirm our model's capacity to proficiently detect and mitigate security threats within software-defined networks, thereby showcasing its potential as a dependable tool for fortifying network security against malicious intrusions. Through its adept utilization of machine learning techniques, our model offers a promising avenue for bolstering cybersecurity measures and safeguarding network infrastructure against evolving threats.

4.5 Discussions and Findings

Within the realm of network security, our model emerges as a standout among state-of-the-art ML models, showcasing unparalleled accuracy with a flawless score of 100%, as depicted in Fig. 4.6. This achievement surpasses that of other prominent models, including D-FACE, BoostIDS, and Cloud Telemetry, which report accuracies of 93%, 86%, and 87%, respectively. Furthermore, our model outshines the Performance and Features (P&F) model, which boasts an accuracy of 96%. The exceptional accuracy exhibited by Our model serves as a testament to its efficacy in fortifying network infrastructure against a myriad of cyber threats. By consistently outperforming its counterparts, our model establishes itself as a formidable asset in the ongoing battle against malicious intrusions, offering a robust defense mechanism for safeguarding critical network assets.

TABLE 4.2. Comparison of GuardianNet with other State-of-Art ML Models Accuracy

ML Model	Accuracy
D-FACE[48]	93%
BoostIDS49	86%
Cloud Telemetry[50]	87%
Performance and Features (P&F)[51]	96%
GuardianNet	100%

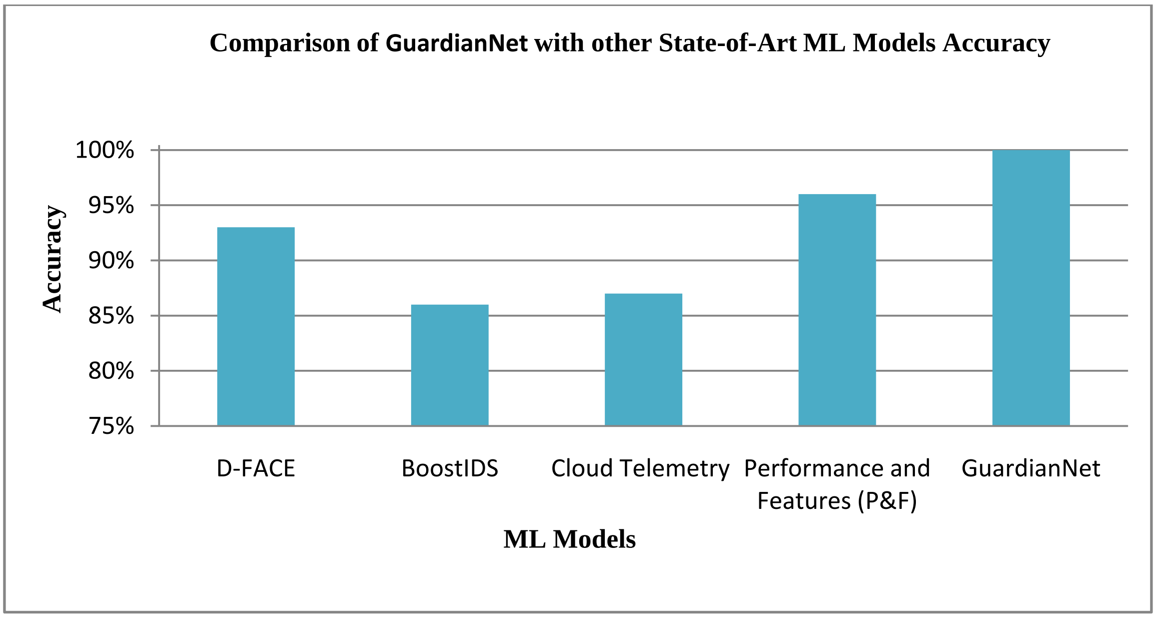


Fig. 4.6 Comparative Assessment of GuardianNet Accuracy Against State-of-the-Art ML Models

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In the domain of network security, our Model heralds a significant advancement, ushering in an era distinguished by unparalleled effectiveness in identifying and mitigating cyber threats, notably the insidious Distributed Denial of Service (DDoS) attacks. Boasting an extraordinary accuracy rate of 100%, our Model surpasses all other state-of-the-art ML models, firmly establishing itself as the epitome of defense mechanisms for safeguarding network infrastructure. This remarkable achievement stands as a testament to our unwavering commitment to excellence and innovation within the field. Through the adept utilization of cutting-edge machine learning techniques and rigorous experimentation, our Model has emerged as an indomitable fortress against the ever-evolving threat landscape of the digital age.

The flawless performance exhibited by our Model not only instils confidence but also underscores the paramount importance of robust cybersecurity measures in our increasingly interconnected world. As cyber threats continue to grow in complexity and sophistication, our Model serves as a beacon of hope, illuminating the path forward for future advancements in network security technologies. Its impact transcends mere protection; it signifies a paradigm shift in our collective approach to defending against malicious actors and preserving the integrity of our digital infrastructure.

With our Model at the helm, organizations and individuals alike can traverse the digital landscape with assurance, knowing that their networks are fortified by the most advanced defense system available. As we navigate the intricate web of cyber threats, our Model stands as a steadfast guardian, steadfastly defending against intrusions and ensuring the resilience of our digital ecosystem. In the face of adversity, our Model remains a steadfast ally, empowering us to confront and overcome the myriad challenges posed by the ever-evolving cyber threat landscape.

5.2 Limitations

While our Model exhibits exceptional performance in detecting and mitigating cyber threats, however, it is not devoid of limitations. One significant limitation lies in its reliance on historical data for training. As the threat landscape continually evolves, our Model may encounter difficulty in accurately detecting novel and previously unseen attack patterns. Moreover, the efficacy of our Model could be impacted by quality & comprehensiveness of training data. Should training dataset fail to sufficiently encapsulate the breadth of real-world cyber threats, the performance of our Model might be compromised. Furthermore, our Model's performance may vary across different network environments and configurations, as it may struggle to generalize well to unseen data. Despite these limitations, continuous refinement and adaptation of our Model can help address these challenges and enhance its effectiveness in protecting network infrastructure against evolving cyber threats.

5.3 Potential Industrial Applications

Our model holds substantial promise for various industrial applications within the realm of network security. One key application lies in enhancing the defense mechanisms of enterprises against cyber threats, particularly Distributed Denial of Service (DDoS) attacks. By accurately detecting and mitigating such attacks in real-time, our model can help safeguard critical network infrastructure, ensuring uninterrupted business operations and minimizing potential financial losses associated with downtime.

Furthermore, our model can find utility in the realm of cloud security, where the protection of sensitive data and resources against malicious intrusions is paramount. By integrating our model into cloud security platforms, service providers can bolster their security measures and offer enhanced protection to their clients' data and applications hosted on cloud infrastructure.

Furthermore, our model holds potential for deployment within the domain of Internet of Things (IoT) security, addressing the emerging vulnerabilities and attack vectors introduced by the widespread adoption of interconnected devices. By monitoring network traffic and identifying anomalous patterns indicative of potential cyber threats, our model can help mitigate risks associated with IoT devices and prevent unauthorized access to sensitive information.

Moreover, our model can find application in the financial sector, where the protection of sensitive financial data and transactions against cyber threats is of utmost importance. By leveraging machine learning algorithms to detect and prevent fraudulent activities such as phishing attacks and unauthorized access attempts, our model can help financial institutions safeguard their customers' assets and maintain trust in their services.

Overall, the potential industrial applications of our model span across various sectors, including enterprise cybersecurity, cloud security, IoT security, and financial security. By empowering organizations with robust defense mechanisms against cyber threats, our model can contribute to the resilience and security of digital infrastructure in an increasingly interconnected world.

5.4 Future Work

The future scope of our model encompasses several avenues for further advancement and application. This includes refining the model's performance through continued optimization and adaptation to emerging cyber threats, expanding its deployment across diverse industrial sectors, and exploring integration with emerging technologies such as artificial intelligence and block-chain for enhanced security measures. Additionally, ongoing research and development efforts can focus on scalability, interoperability, and collaboration with industry stakeholders to address evolving cybersecurity challenges and ensure the model remains at the forefront of network security innovation.

REFERENCES

[1] Zekri, Marwane, Said El Kafhali, Noureddine Aboutabit, and Youssef Saadi. "DDoS attack detection using machine learning techniques in cloud computing environments." In *2017 3rd international conference of cloud computing technologies and applications (CloudTech)*, pp. 1-7. IEEE, 2017.

[2] Douligeris, Christos, and Aikaterini Mitrokotsa. "DDoS attacks and defense mechanisms: classification and state-of-the-art." *Computer networks* 44, no. 5 (2004): 643-666.

[3] Singh, Aarti, and Dimple Juneja. "Agent based preventive measure for UDP flood attack in DDoS attacks." *International Journal of Engineering Science and Technology* 2, no. 8 (2010): 3405-3411.

[4] Harshita, Harshita. "Detection and prevention of ICMP flood DDOS attack." *International Journal of New Technology and Research* 3, no. 3 (2017): 263333.

[5] Zeebaree, Subhi Rafeeq, Karwan Jacksi, and Rizgar R. Zebari. "Impact analysis of SYN flood DDoS attack on HAProxy and NLB cluster-based web servers." *Indones. J. Electr. Eng. Comput. Sci* 19, no. 1 (2020): 510-517.

[6] Abdollahi, Asrin, and Mohammad Fathi. "An intrusion detection system on ping of death attacks in IoT networks." *Wireless Personal Communications* 112, no. 4 (2020): 2057-2070.

[7] Kumar, Sanjeev. "Smurf-based distributed denial of service (ddos) attack amplification in internet." In *Second International Conference on Internet Monitoring and Protection (ICIMP 2007)*, pp. 25-25. IEEE, 2007.

[8] Bittau, Andrea. "The fragmentation attack in practice." In *IEEE Symposium on Security and Privacy, IEEE Computer Society*. 2005.

[9] Sreeram, Indraneel, and Venkata Praveen Kumar Vuppala. "HTTP flood attack detection in application layer using machine learning metrics and bio inspired bat algorithm." *Applied computing and informatics* 15, no. 1 (2019): 59-66.

[10] Damon, Evan, Julian Dale, Evaristo Laron, Jens Mache, Nathan Land, and Richard Weiss. "Hands-on denial of service lab exercises using slowloris and rudy." In *proceedings of the 2012 information security curriculum development conference*, pp. 21-29. 2012.

[11] Anagnostopoulos, Marios, Georgios Kambourakis, Panagiotis Kopanos, Georgios Louloudakis, and Stefanos Gritzalis. "DNS amplification attack revisited." *Computers & Security* 39 (2013): 475-485.

[12] Rozekrans, Thijs, Matthijs Mekking, and Javy de Koning. "Defending against DNS reflection amplification attacks." *University of Amsterdam System & Network Engineering RP1* (2013).

[13] Rudman, Lauren, and B. Irwin. "Characterization and analysis of NTP amplification based DDoS attacks." In *2015 Information Security for South Africa (ISSA)*, pp. 1-5. IEEE, 2015.

[14] Lee, Yong-joon, Hwa-sung Chae, and Keun-wang Lee. "Countermeasures against large-scale reflection DDoS attacks using exploit IoT devices." *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije* 62, no. 1 (2021): 127-136.

[15] Yu, Jie, Zhoujun Li, Huowang Chen, and Xiaoming Chen. "A detection and offense mechanism to defend against application layer DDoS attacks." In *International Conference on Networking and Services (ICNS'07)*, pp. 54-54. IEEE, 2007.

[16] Bansal, Chetan, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, and Sergio Maffeis. "Discovering concrete attacks on website authorization by formal analysis." *Journal of Computer Security* 22, no. 4 (2014): 601-657.

[17] Bilge, Leyla, and Tudor Dumitraş. "Before we knew it: an empirical study of zero-day attacks in the real world." In *Proceedings of the 2012 ACM conference on Computer and communications security*, pp. 833-844. 2012.

[18] Chen, Ping, Lieven Desmet, and Christophe Huygens. "A study on advanced persistent threats." In *Communications and Multimedia Security: 15th IFIP TC 6/TC 11 International Conference, CMS 2014, Aveiro, Portugal, September 25-26, 2014. Proceedings 15,* pp. 63-72. Springer Berlin Heidelberg, 2014.

[19] Ramil Khantimirov, “DDoS Attacks in 2022: Trends and Obstacles Amid Worldwide Political Crisis”, Available: https://www.infosecurity-magazine.com/blogs/ddos-attacks-in-2022- trends/ (Last Accessed on: December 31, 2022)

[20] Seifousadati, Alireza and Ghasemshirazi, Saeid and Fathian, Mohammad, “A Machine Learning Approach for DDoS Detection on IoT Devices”, arXiv, 2021. Doi: 10.48550/ARXIV.2110.14911

[21] Kumari, K., Mrunalini, M., “Detecting Denial of Service attacks using machine learning algorithms”, . J Big Data 9, 56 (2022).

[22] S. Newman, “Under the radar: the danger of stealthy DDoS attacks,” Network Security, vol. 2019, no. 2, pp. 18-19, 2019.

[23] A. Marzano, D. Alexander, O. Fonseca et al., “The evolution of bashlite and mirai IoT botnets,” in Proceedings of the 2018 IEEE Symposium on Computers and Communications (ISCC), 2018.

[24] Y. Cao, Y. Gao, R. Tan, Q. Han, and Z. Liu, “Understanding internet DDoS mitigation from academic and industrial perspectives,” IEEE Access, vol. 6, pp. 66641–66648, 2018.

[25] Singh Samom, Premson, and Amar Taggu. "Distributed denial of service (DDoS) attacks detection: A machine learning approach." In *Applied Soft Computing and Communication Networks: Proceedings of ACN 2020*, pp. 75-87. Springer Singapore, 2021.

[26] Hastie, Trevor, Saharon Rosset, Ji Zhu, and Hui Zou. "Multi-class adaboost." *Statistics and its Interface* 2, no. 3 (2009): 349-360.

[27] Saghezchi, Firooz B., Georgios Mantas, Manuel A. Violas, A. Manuel de Oliveira Duarte, and Jonathan Rodriguez. "Machine learning for DDoS attack detection in industry 4.0 CPPSs." *Electronics* 11, no. 4 (2022): 602.

[28] Sontowski, Sina, Maanak Gupta, Sai Sree Laya Chukkapalli, Mahmoud Abdelsalam, Sudip Mittal, Anupam Joshi, and Ravi Sandhu. "Cyber attacks on smart farming infrastructure." In *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, pp. 135-143. IEEE, 2020.

[29] S. Kottler, “February 28th DDoS incident report,” 2018, https://github.blog/2018-03-01-ddos-incident-report/.

[30] Francisco Sales de Lima Filho, Frederico A. F. Silveira, Agostinho de Medeiros Brito Junior, Genoveva Vargas-Solar, Luiz F. Silveira, "Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning", Security and Communication Networks, vol. 2019, Article ID 1574749, 15 pages, 2019.

[31] Ebtihal Sameer Alghoson, Onytra Abbass, “Detecting Distributed Denial of Service Attacks using Machine Learning Models”, International Journal of Advanced Computer Science and Applications, Vol. 12, No. 12, 2021.

[32] R. Doshi, N. Apthorpe and N. Feamster, "Machine Learning DDoS Detection for Consumer Internet of Things Devices," 2018 IEEE Security and Privacy Workshops (SPW), 2018, pp. 29-35, doi: 10.1109/SPW.2018.00013.

[33] Seifousadati, Alireza, Saeid Ghasemshirazi, and Mohammad Fathian. "A Machine Learning Approach for DDoS Detection on IoT Devices." arXiv preprintr Xiv:2110.14911 (2021).

[34] Abdullah Soliman Alshra'a, Ahmad Farhat, Jochen Seitz, "Deep Learning Algorithms for Detecting Denial of Service Attacks in Software-Defined Networks", Procedia Computer Science, Volume 191, 2021, Pages 254-263, ISSN 1877-0509.

[35] Ramadhan, Ilham, Parman Sukarno, and Muhammad Arief Nugroho. "Comparative analysis of K-nearest neighbor and decision tree in detecting distributed denial of service." In *2020 8th International Conference on Information and Communication Technology (ICoICT)*, pp. 1-4. IEEE, 2020.

[36] Najar, Ashfaq Ahmad, and S. Manohar Naik. "DDoS attack detection using MLP and Random Forest Algorithms." *International Journal of Information Technology* 14, no. 5 (2022): 2317-2327.

[37] Yadav, Satyajit, and S. Selvakumar. "Detection of application layer DDoS attack by modeling user behavior using logistic regression." In *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions)*, pp. 1-6. IEEE, 2015.

[38] Maheshwari, Aastha, Burhan Mehraj, Mohd Shaad Khan, and Mohd Shaheem Idrisi. "An optimized weighted voting based ensemble model for DDoS attack detection and mitigation in SDN environment." *Microprocessors and Microsystems* 89 (2022): 104412.

[39] Sayed, Moinul Islam, Ibrahim Mohammed Sayem, Sajal Saha, and Anwar Haque. "A multi-classifier for DDoS attacks using stacking ensemble deep neural network." In *2022 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1125-1130. IEEE, 2022.

[40] Bindra, Naveen, and Manu Sood. "Evaluating the impact of feature selection methods on the performance of the machine learning models in detecting DDoS attacks." *Rom. J. Inf. Sci. Technol* 23, no. 3 (2020): 250-261.

[41] Revathi, Sathyanarayanan, and A. Malathi. "A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion

detection." *International Journal of Engineering Research & Technology (IJERT)* 2, no. 12 (2013): 1848-1853.

[42] TensorFlow, Developers. "TensorFlow." *Site oficial* (2018).

[43] Harris, Charles R., K. Jarrod Millman, Stéfan J. Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser et al. "Array programming with NumPy." *Nature* 585, no. 7825 (2020): 357-362.

[44] Reback, Jeff, Wes McKinney, Joris Van Den Bossche, Tom Augspurger, Phillip Cloud, Adam Klein, Simon Hawkins et al. "pandas-dev/pandas: Pandas 1.0. 5." *Zenodo* (2020).

[45] Bisong, Ekaba, and Ekaba Bisong. "Matplotlib and seaborn." *Building machine learning and deep learning models on google cloud platform: A comprehensive guide for beginners* (2019): 151-165.

[46] Hao, Jiangang, and Tin Kam Ho. "Machine learning made easy: a review of scikit-learn package in python programming language." *Journal of Educational and Behavioral Statistics* 44, no. 3 (2019): 348-361.

[47] Mohsin, Mayadah A., and Ali H. Hamad. "Performance evaluation of SDN DDoS attack detection and mitigation based random forest and K-nearest neighbors machine learning algorithms." *Revue d'Intelligence Artificielle* 36, no. 2 (2022): 233.

[48] Behal, Sunny, Krishan Kumar, and Monika Sachdeva. "D-FACE: An anomaly based distributed approach for early detection of DDoS attacks and flash events." Journal of Network and Computer Applications 111 (2018): 49-63.

[49] Abou El Houda, Zakaria, Bouziane Brik, and Lyes Khoukhi. "Ensemble learning for intrusion detection in sdn-based zero touch smart grid systems." In 2022 IEEE 47th Conference on Local Computer Networks (LCN), pp. 149-156. IEEE, 2022.

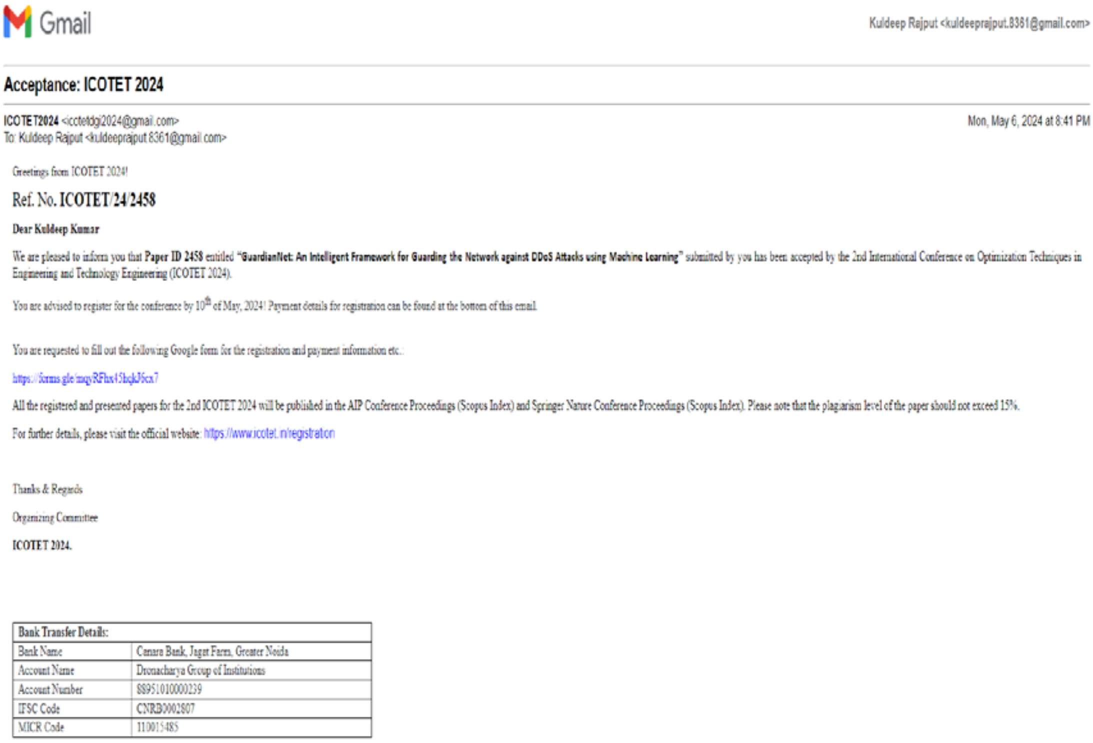
[50] Corrêa, João Henrique, Patrick M. Ciarelli, Moises RN Ribeiro, and Rodolfo S. Villaça. "Ml-based ddos detection and identification using native cloud telemetry macroscopic monitoring." Journal of Network and Systems Management 29 (2021): 1-28.

[51] Tang, Dan, Yudong Yan, Siqi Zhang, Jingwen Chen, and Zheng Qin. "Performance and features: Mitigating the low-rate TCP-targeted DoS attack via SDN." IEEE Journal on Selected Areas in Communications 40, no. 1 (2021): 428-444.

List of Publications/Acceptance

[1] Kuldeep Kumar, Rahul Katarya. “Securing SDN from DDoS attacks: A Comprehensive Study of Security Challenges and Opportunities” *Proceedings of the International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET)*, 2024

[2] Kuldeep Kumar, Rahul Katarya. “GuardianNet: An Intelligent Framework for Guarding the Network against DDoS Attacks using Machine Learning” *Proceedings of the International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET)*, 2024



Kuldeep Rajput <kuldeeprajput.8381@gmail.com>

Acceptance: ICOTET 2024

ICOTET2024 <icotetdgi2024@gmail.com>
To: Kuldeep Rajput <kuldeeprajput.8361@gmail.com>

Mon, May 6, 2024 at 8:41 PM

Greetings from ICOTET 2024!

Ref. No. **ICOTET/24/2458**

Dear Kuldeep Kumar

We are pleased to inform you that **Paper ID 2458** entitled **“GuardianNet: An Intelligent Framework for Guarding the Network against DDoS Attacks using Machine Learning”** submitted by you has been accepted by the 2nd International Conference on Optimization Techniques in Engineering and Technology Engineering (ICOTET 2024).

You are advised to register for the conference by 10th of May, 2024! Payment details for registration can be found at the bottom of this email.

You are requested to fill out the following Google form for the registration and payment information etc.:

https://forms.gle/mqyRFhx45hqkJ6cx7

All the registered and presented papers for the 2nd ICOTET 2024 will be published in the AIP Conference Proceedings (Scopus Index) and Springer Nature Conference Proceedings (Scopus Index). Please note that the plagiarism level of the paper should not exceed 15%.

For further details, please visit the official website: https://www.icotet.in/registration

Thanks & Regards

Organizing Committee

ICOTET 2024.

Bank Transfer Details:	
Bank Name	Canara Bank, Jagat Farm, Greater Noida
Account Name	Dronacharya Group of Institutions
Account Number	88951010000239
IFSC Code	CNRB0002807
MICR Code	110015485

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis **Development of Framework for DDoS Attack Detection in Network Devices Using Machine Learning**.

Total Pages _____Name of the Scholar____________________________________

Supervisor____

Department____

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: _______________ Similarity Index: _______ Total Word Count: ____

Date

Candidate’s Signature **Signature of Supervisor**