CLASSIFICATION OF CYBER ATTACKS ON IoT USING MACHINE LEARNING

By

HAMZA ISHTIAQ

NATIONAL UNIVERSITY OF MODERN LANGUAGES,

ISLAMABAD

January 2024

CLASSIFICATION OF CYBER ATTACKS ON IoT USING MACHINE LEARNING

By

HAMZA ISHTIAQ

BE EE, National University of Modern Languages, Islamabad 2019

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

Electrical Engineering

TO

A FACULTY OF ENGINEERING AND COMPUTING

NATIONAL UNIVERSITY OF MODERN LANGUAGES ISLAMABAD

THESIS AND DEFENCE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defense, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering and Computer science for acceptance.

Thesis Title: Classification of Cyber Attacks on IoT Using Machine Learning

Submitted By**:** Hamza Ishtiaq Registration #: MS/EE/F20/002

Master of Science in Electrical Engineering

Electrical Engineering

Discipline

Dr. Sajid Saleem _____

Research Supervisor Signature of Research Supervisor

Dr. Noman Malik _____

Dean (FEC) Signature of Dean (FEC)

Brig. Shahzad Munir _____

Director General Signature of Director General

January 29th, 2024

Date

AUTHOR'S DECLARATION

I **Hamza Ishtiaq**

Son of **Ishtiaq Ahmed**

Registration # **MS/EE/F20/002**

Discipline **Electrical Engineering**

Candidate of **Master of Science in Electrical Engineering (MSEE)** at the National University of Modern Languages do hereby declare that the thesis **Classification of Cyber Attacks on IoT using Machine Learning** submitted by me in partial fulfillment of MSEE degree, is my original work, and has not been submitted or published earlier. I also solemnly declare that it shall not, in the future, be submitted by me for obtaining any other degree from this or any other university or institution. I also understand that if evidence of plagiarism is found in my thesis/dissertation at any stage, even after the award of a degree, the work may be canceled, and the degree revoked.

Signature of Candidate

Hamza Ishtiaq

Name of Candidate

JANUARY 29th , 2024

Date

ABSTRACT

Classification of Cyber Attacks on IoT using Machine Learning

Internet of Things (IoT) devices are growing rapidly, which are raising concerns regarding data security and privacy. With the increase in volume of data generated by IoT devices, secure transmission and processing has become essential. Protecting IoT infrastructure and devices from cyber-attacks is an active research area now-a-days. The detection and classification of attacks on IoT require dynamic and automatic techniques, based on Machine Learning (ML). This thesis focuses on the classification of cyber-attacks on IoT using ML. It explores the potential of ensemble learning to enhance the accuracy of ML. Two different publicly available datasets are used in this thesis. These datasets are UNSW NB 15 and NSL KDD. The ensemble learning is implemented by combining the strengths of Extreme Gradient Boosting (XGBoost) and Deep Neural Network (DNN). The proposed ensemble learner is named as X-DNN.

The performance evaluation of X-DNN is carried out using, accuracy, precision, recall, and F1 scores. The experimental results show that X-DNN demonstrates an accuracy of 85.36% in classifying cyber-attacks on UNSW NB 15 dataset whereas it achieves accuracy of 99.81% on NSL KDD dataset. X-DNN outperforms state of the art methods such as Hierarchical Clustering Decision Tree Twin Support Vector Machine, Gated Recurrent Unit, and Deep Neural Networks. The proposed method offers relatively better performance which highlights its significance in classifying cyber-security attacks on IoT.a

TABLE OF CONTENT

List of Tables

TABLE NO. **TITLE** **PAGE**

List of Figures

FIGURE NO. **TITLE** **PAGE**

LIST OF ABBREVIATIONS

ACPS	Aerospace Cyber-Physical Systems
AD	Anomaly Detection
Adam	Adaptive Moment Estimation
AdaMax	Adaptive Movement Estimation
AMoF	Accumulated Measure of Fluctuation
ANN	Artificial Neural Network
AODE	Averaged one-dependence estimators
AUC	Area under the curve
CAS	Cognitive Adaptive System
CNN	Convolutional Neural Networks
CPS	Cyber-Physical System
DCNN	Deep Convolutional Neural Network
DDoS	Distributed Denial of services
DFEL	Deep Feature Embedding Learning
DL	Deep Learning
DNN	Deep Neural Networks
DoS	Denial of Service
DT	Decision Tree
EASH	Energy Aware Smart Home
ECASP	Ensemble classifier with Stacking Process
FN	False Negative
FP	False Positive

FPR	False Positive Rate
GNN	Graph Neural Network
GRU	Gated Recurrent Unit
HC-DTTSVM	Hierarchical Clustering and Decision Tree Twin Support Vector Machine
IDSs	Intrusion detection systems
IIoT	Industrial IoT
IoT	Internet of Things
IoV	Internet of Vehicles
LDA	linear discriminant analysis
LR	Logistic Regression
LSTM	Long Short-Term Memory
MCC	Mathew Correlation Coefficient
MFO- RELM	Mayfly Optimization-Regularized Extreme Learning Machine
ML	Machine Learning
MLP	Multilayer Perceptron
MQTT	Message Queuing Telemetry Transport
Nadam	Nesterov-accelerated Adaptive Moment Estimation
NIDS	Network Intrusion Detection system
NLIF	Non-Leaky Integrate and Fire
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
R2L	Remote to Local
RF	Random Forests
RFID	Radio Frequency Identification
RL	Reinforcement Learning

ROC	Receiver Operating Characteristic
SDF	Symbolic Dynamic Filtering
SDN	Software-Defined Networking
TN	True Negative
TP	True Positive
TPR	True Positive Rate
U2R	User to Root
WoT	Web of Things
WSNs	Wireless Sensor Networks
XGBoost	Extreme Gradient Boosting

ACKNOWLEDGEMENT

First and foremost, I am deeply grateful to Allah Almighty for bestowing His blessings upon me. I would like to extend my heartfelt appreciation to my parents for their unwavering support throughout my research journey. Their love, encouragement, and unwavering belief in my abilities have been instrumental in my accomplishments.

I am also indebted to my research supervisor, Dr. Sajid Saleem, for his persistent dedication and invaluable guidance in completing this research. His unwavering enthusiasm and expertise in the field of research have continuously inspired me. I am truly grateful for his invaluable advice, comprehensive support, and unwavering involvement in every aspect of my research. His exceptional guidance has played a crucial role in shaping the outcome of this thesis.

Once again, I extend my heartfelt gratitude to Allah Almighty, my parents, my friend Awais Abbas, and my esteemed supervisors for their invaluable contributions to my academic journey. Their guidance, support, and friendship have been invaluable, and I am truly grateful for their presence in my life.

DEDICATION

This thesis is dedicated to my father, who taught me that the best kind of knowledge to have is that which is learned for its own sake. It is also dedicated to my mother, who taught me that even the largest task can be accomplished if it is done one step at a time.

CHAPTER 1

INTRODUCTION

1.1 Introduction

Internet of Things (IoT) has a wide range of applications in different domains [1]. Cyber-attacks on IoT have considerably increased with the passage of time which are raising concerns for IoT applications such as smart homes [2], energy [3], healthcare [4], industrial processes [5], agriculture [6], and advanced manufacturing [7] sectors. IoT devices generate huge amounts of data and require authentication, security, protection and transmission [8] mechanisms.

Figure 1.1 IoT Application

The use of Machine learning (ML) techniques in this regard have considerably increased to detect and identify cyber-attacks[9]. In Figure 1.1, an example of IoT application is shown. In the given IoT architecture diagram, various components are interconnected to enable

seamless communication and functionality. A brief explanation of each component is as follows:

i. Home: This represents the physical residence or living space where the IoT system is deployed. It serves as the central hub for connecting and controlling the interconnected devices.
ii. Mic: The microphone component captures audio input and allows voice-based interaction with the IoT system. It enables users to give voice commands or communicate with other devices.
iii. Maintenance: Maintenance component represents a set of crucial activities essential for ensuring the health and performance of the IoT system. These activities encompass tasks such as software updates, device troubleshooting, and maintaining the system's smooth operation, contributing to the overall efficiency of the IoT ecosystem.
iv. Music: This component refers to the integration of music streaming services or audio playback devices within the IoT architecture. It allows users to stream and control music wirelessly through connected devices.
v. Setting: The setting component refers to the configuration or customization options available for the IoT system. It allows users to adjust various parameters or preferences according to their requirements.
vi. LED: The LED component represents light-emitting diodes, which can be integrated into the IoT system for various purposes. These LEDs can serve as indicators, status displays, or even interactive elements that respond to user commands.
vii. Wireless Fidelty: WiFi represents the wireless network connectivity that enables communication between different IoT devices within the home environment. It serves as the communication medium for data transmission.
viii. Persons: The persons component represents individuals or users who interact with the IoT system. They can engage with the system through voice commands, mobile applications, or other interfaces to control and access connected devices.
ix. Cloud: The cloud component represents the cloud computing infrastructure that provides storage, processing, and analysis capabilities for the IoT system. It facilitates data storage, advanced analytics, and remote access to the system.

x. Games: The games component indicates the integration of gaming devices or applications within the IoT system. It enables users to engage in interactive gaming experiences using connected devices.

xi. Home Appliances: This component includes various household appliances that are connected to the IoT system. Examples may include smart thermostats, smart lighting systems, smart kitchen appliances, and other automated devices that can be remotely controlled or monitored.

The interconnected components in the IoT architecture aim to create a smart and automated environment where users can control various devices, access services, and personalize their living spaces according to their preferences.

The rapid development of communication and information technologies has surpassed conventional methods for recognizing trivial network settings. The evolution of smart home systems has been aided by IoT based technologies which have enhanced quality of life but have increased the security concerns as well due to cyber-attacks. Furthermore, IoT allows millions of devices to communicate and collaborate [10]. Sensors, internet, Radio Frequency Identification (RFID), and similar innovation have enabled transforming of conventional items into smart objects to interact with one another [3]. The ability to anticipate future needs and the capacity to execute the requirements successfully using IoT is a significant societal and industrial revolution. Sensory devices are connected with the IoT to sense the environments [11].

The IoT consists of three primary stages: data collection, data processing/aggregation, and data transmission [3]. Data is collected from various nodes having limited storage, limited range, and low computational power. Then the data is analyzed or aggregated and then finally transmitted.

An IoT architecture consists of sensors, actuators, protocols, cloud services, and layers that make up the organizing framework. The platform is typically made up of distinct layers that allow administrators to maintain, monitor, and evaluate consistency. The application, transport, and perception layers are part of an IoT architecture. The first layer utilizes numerous smart technologies to gather data by combining facts and connecting users and IoT via

distributed computing or advanced data management methods. The second layer is responsible for network activities, whereas the third level gathers data. The absence of usage layer security is a disadvantage [12].

Various approaches provide IoT security, some of those were used to secure network access control procedures that concentrate on information confidentiality and authentication for proper network access. As a result, the IoT-based network necessitates a security function that can be utilized as a guard to identify cyber-attacks [13]. As a result, researchers are looking for innovative and automatic methods to improve security using Machine Learning (ML) algorithms. These systems protect data by examining the entire network activity and generating an alert if a security breach is discovered [14].

The IoT security has evolved over time, but attacks have also grown broad and distracting [15]. Intrusion detection systems (IDSs) based on ML have frequently explored by the researchers in attempts to defend the IoT networks. Traditional IDS techniques are less efficient owing to their simple and typical characteristics such as monitoring diversity, limited bandwidth size, and worldwide connectivity [14]. The most recent ML methods have brought revolution in IoT, video surveillance, medical IoT, and others because they are capable of identifying unusual and new patterns and detect attacks initiated toward IoT networks. Numerous surveys have been conducted on the security and attacks on IoT with focus on recently emerging cyber-attacks on IoT [16]. These needs revisiting the state-of-the-art ML techniques and requires combining their strengths in an ensemble way to enhance security and defend the IoT against cyber-attacks.

1.2 Problem Statement

IoT devices requires seamless connectivity beside a robust security framework to ensure prevention of malicious traffic and cyber-attacks without compromising the uninterrupted connectivity and data transmission [17]. Protecting IoT against cyber attacks, hacking attempts and malicious traffic is of an important research area now a days [4]. Numerous methods have been proposed to mitigate cyber-attacks [18]. However, there is still a need of new and robust

methods to mitigate emerging cyber attacks by using methods especially based on ML and deep learning techniques.

1.3 Research Question

The main research questions are as follows:

RQ1: Which Machine learning approaches perform best against which types of cyber-attacks?

RQ2: How to combine different machine and deep learning techniques in an ensemble way to identify the new cyber-attacks efficiently?

1.4 Objectives

Main objective are as follows:

i. To find the best ML algorithms to identify cyber security attacks on IoT
ii. To perform a comparative analysis of different ML approaches in an ensemble way to mitigate new cyber-attacks

1.5 Scope of Study

Cybersecurity is one of the most challenging problems that IoT devices confront. Sensors, equipment, and networks connected to the internet are subjected to regular internet-based examination, theft, ransom, espionage, and even destruction [19]. An IoT device is the most vulnerable to severe attacks since because it has multiple online nodes spread across large areas. IoT attacks could cause chaos and significant economic damages [20].

Security is often overlooked and users are put at risk as vendors compete to bring new goods to the market, making IoT device protection even more difficult [21]. The lack of a single approach for protecting already deployed gadgets makes the problem even more complex [6]. Often IoT devices are unable to upgrade software after purchase by the consumer because they do not provide such a functionality. Therefore, it is essential to develop a technique for detecting vulnerabilities in IoT.

With the increasing number and complexity of attacks on IoT, protecting sensitive personal and commercial information, and safeguarding security, requires constant attention. The adoption of technology has led to including most countries' essential services in cyberspace networks. Because of this, the internet has become a rather appealing and effective platform for attack. After land, sea, air and space warfare, cyber warfare has been recognized as the fifth modern domain war [22]. The accessibility to a global network allows attackers to roam the internet freely and target systems in places that would otherwise be out of reach, making attacks more straightforward to execute and faster than traditional warfare[23]. So, it is essential to develop techniques for detecting anomalies in the networks.

This thesis focuses on the identification and mitigation of cyber-attacks in the context of IoT systems. The primary objective of our thesis is to develop effective methods for identifying cyber-attacks targeting IoT systems. Examples of attacks are Reconnaissance, Backdoor, Denial of Service (DoS), Exploits, Fuzzers, Probe, Distributed Denial of Service (DDoS), R2L, U2R, and unauthorized access attempts.

To train ML for the identification of cyber-attacks depends upon the dataset. The dataset should reflect real-world scenarios and characteristics of cyber-attacks. It should contain instances or examples that are similar to the actual attacks that will be occurring in the targeted environment, such as IoT systems. The dataset should be sufficiently large to provide an adequate number of samples for learning. A larger dataset enables robust and reliable results. The dataset should cover a wide range of attack types, including both known and emerging ones. For instance, DoS, DDoS, Reconnaissance, Backdoor, Exploits, Fuzzers, Probe, R2L, U2R, etc.

1.6 Contribution and Significance

This research applies ML to enhance the security aspect of the IoT. It has applications in IoT systems used in home, energy, agriculture, smart homes, advanced manufacturing, and industries. The focus is on emerging attacks. The automatic classification of the attacks are investigated. The study of cyber-attacks and countermeasures are rapidly growing fields, as evidenced by several recent research works [24]. Every day, new types of attacks emerge. So, ML algorithms applied on the most recent data sets, combining even more information with the essential IoT features help them to work efficiently.

1.7 Organization of Thesis

The thesis is organized into five main sections: introduction, literature review, methodology, results and discussion, and limitations and conclusion.

The introduction provides an overview of the study's objectives, the research problem background, research questions, and objectives, while also discussing the study's significance. The literature review delves into relevant literature in the cybersecurity, machine learning (ML), and deep learning (DL) fields, covering the history of cybersecurity, ML and DL evolution, their applications in cybersecurity, and existing research on ML in cybersecurity. The methodology chapter outlines the research approach, data collection and preprocessing, feature selection, data augmentation, proposed model development, and dataset descriptions. The results and discussion chapter details the experimental setup, main findings, and comparisons among various ML and DL techniques, including the proposed method. The conclusion and future work chapter summarizes research outcomes, identifies constraints, suggests future research directions, and underscores the study's contribution to cybersecurity and ML.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

Cyber-attacks on IoT devices pose severe threats, compromise on data integrity, privacy, and the overall functionality of these interconnected systems. Timely identification and mitigation of cyber-attacks are vital for safeguarding data, maintaining operational efficiency, and protecting user trust in IoT technologies [25]. This chapter provides insights into the state of the art methods, highlights research gaps and challenges, and identifies the effective ML approaches which can be employed in the cyber-attacks.

2.2 Internet of Things Security

A framework is presented in [1] to classify attacks of botnets for nine devices and calculate various performance metrics for different algorithms. The study showed impressive results with high attack detection rates of above 99%.

A systematic review of the security challenges faced by the IoT technology is presented in [14]. In IoT, there is an increase in connected devices, which leads to a decline in security management, leaving it open to security attacks that can cause significant financial and reputational losses. There are various security issues that exist at IoT layers, with a particular emphasis on DDoS attacks, which pose a noteworthy threat to the cyber world. Furthermore, various anomaly detection techniques have been explored along with future research challenges and potential solutions [26].

Several research papers have addressed the security challenges posed by the Internet of Things (IoT) and its large-scale deployment of connected devices. In [27], an unsupervised anomaly detection system was proposed for large-scale smart grids using symbolic dynamic filtering (SDF) for feature extraction, achieving a high accuracy of 99%. Ref [28] introduced a two-stage detection method for Mobile Ad Hoc Network (MANETs), wireless Sensor Networks (WSNs), and IoT networks, which demonstrated detection rates exceeding 90% for node velocity scenarios. The importance of enhancing the security of IoT networks due to the large scale of IoT devices being developed and deployed is challenging due to the limited computational ability of these devices and the difficulty in applying encryption and authentication to prevent malicious cyber-attacks [29]. Various studies have focused on security concerns of IoT networks with new and conventional ML-based approaches. Different studies have focused on different safety concerns, although others provide a variety of security methods dependent on various models and strategies.

The challenges of privacy, safety, and security in the IoT systems are discussed in [30]. Traditional techniques and tools are not efficient in preventing and protecting against new types of cyber threats, and therefore, up-to-date, dynamic, and robust, security measures are reported essential to secure IoT systems. Regarding security in IoT development, a study [31] highlights the security issues in IoT development and emphasizes the need to link high-risk network activities with entities to promote industrial development. It investigates the use of deep learning for intelligent system defense and automated monitoring, along with the contribution of IoT in detecting and responding to cyber-attacks and securing edge data transmission. Lastly, research in [32] investigated intrusion detection and security in the growing networks of the IoT, emphasizing the use of advanced deep learning techniques for detecting intrusions and abnormal behavior in large-scale IoT networks [33].

In the context of the growing networks of the IoT and their vulnerability to security threats, the research [34] study effect of intrusion detection and security in the growing networks of the IoT, which generate huge amounts of vulnerable data to security threats. Conventional intrusion detection systems (IDSs) are insufficient for advanced attacks. So it investigates the utilization of sophisticated deep learning methods for detecting intrusions automatically and identifying abnormal behavior in a precise manner. Public network-based datasets of IDSs are used and the performance of DL techniques are based on different metrics.

2.2.1 Machine Learning

This section provides an overview of different ML techniques used for the identification and detection of cyber-attacks.

In response to the escalating challenges in securing IoT systems within smart cities due to the exponential growth of network traffic, researchers proposed a Random Forest Based method for Anomaly Detection in IoT systems [35]. The exponential growth of network traffic through IoT systems in smart cities has introduced new cybersecurity challenges. Infected IoT devices can be detected at distributed fog nodes using the Random Forest Based method, which achieves a classification accuracy of 99.34%. The authors in [36] argue that their approach is effective in addressing IoT cybersecurity threats in smart cities. It evaluate the performance of multiple ML models to achieve accurate prediction of attacks and anomalies in IoT systems. It is shown that the Random Forest based algorithm gives accuracy of 99.4%. In the realm of IoT security against denial of service (DoS) attacks, a study [37] deploy ML classifiers to secure IoT against denial of service (DoS) attacks. A comprehensive study conducted on classifiers by the authors to develop IDSs and evaluated their performance using popular datasets NSL-KDD, UNSW-NB15, and CIDDS-001. They also analyzed the substantial distinction among classifiers statistically using Friedman and Nemenyi tests and evaluated the response time of classifiers on IoT.

A framework on Authentic and protocol-compliant DoS attack cases is proposed in [38]. The framework is compare with three classifiers, C4.5, Averaged one-dependence estimators (AODE), and Multilayer Perceptron (MLP) using different features, and the results are reported in the form of false positive rate, true positive rate, training time, accuracy and error. Considering the obtained results, the framework appears to be effective in detecting attacks. The AODE and C45 classifiers, when used with complete features without feature selection, yield the highest accuracy rates [39]. In contrast, the MLP classifier exhibited the highest false positive rate and training time. A novel approach is presented to detect multi-layer DDoS attacks in IoT devices using ML techniques [40]. The network system includes Software-Defined Networking (SDN) switches, cloud servers, and IoT devices, IoT gateways. The sensory data is collected from eight smart poles in the system and processed to extract features to identify DDoS attacks [41]. The collected data is split into training and testing samples,

achieving over 97% test accuracy using DT based methods. An ML-based framework [25] is used for identifying botnet attacks on IoT devices. The framework is a lightweight system with less computational power requirement for detecting botnet-based attacks. The framework uses sequential detection architecture and an efficient feature selection procedure. It achieves a high level of accuracy, around 99%, using several ML algorithms which are Naïve Bayes, artificial neural network (ANN), and decision tree [42].

According to [43] the increasing popularity of smart homes and the potential security risks has come with the devices and sensors being connected to the internet. A hybrid IDS incorporating RF, XGBoost, DT, KNN methods are used for the detection of cyber-attacks. An intrusion detection in Aerospace Cyber-Physical Systems (ACPS) is proposed [44] for the system equipped with sensors, data analysis tools and wireless standards. These systems are prone to cyber threats that compromise their safety, efficiency, and reliability. ML approaches have demonstrated encouraging outcomes in the cybersecurity applications such as identifying intrusion [45]. However, the lack of available data on IoT attacks is a significant problem for applying machine learning techniques to this field. The study compares different machine learning models and network architectures for efficiency and accuracy, and achieves over 99% accuracy with low energy consumption and memory usage. A two-stage hybrid framework is employed for enhancing the accuracy of intrusion detection systems in IoT networks [46]. The method utilizes a genetic algorithm to select appropriate features and ML algorithms, such as ensemble classifier, SVM, and DT, for classification. The study reports 99.8% accuracy with 10-fold cross-validation on a multiclass NSL-KDD database.

In light of the significant challenges encountered in IoT security, researchers in [30] highlight the significant challenges faced by IoT security and how ML has emerged as a powerful method to recognize threats, and suspicious activities in networks. A comprehensive literature review and comparison of various ML algorithms are carried out in detecting attacks and anomalies. It also highlights the potential of ML-based techniques in providing an effective solution to the security issues faced by the IoT networks [47]. ML algorithms are widely used in diverse areas, including healthcare and IoT. However, there is a high possibility of manipulating the training datasets, which can produce biased results. To tackle this issue, a solution is proposed [48] that uses a private cloud and blockchain technology. The system allows dataset owners to securely store and manage their data, preventing tampering and

ensuring the integrity of the data. A Framework discusses the increasing concern for attacks on IoT systems and the enormous types of attacks that can occur [49]. They compare various ML algorithms, including ANN, LR, RF, SVM, and DT, using evaluation metrics such as accuracy, f1 score, recall, and precision and area under the ROC Curve. The results suggest that the RF algorithm performs relatively well, compared to other methods, although all methods have similar accuracy.

Investigating the application of machine learning algorithms to IoT datasets, including the RF classifier, SVM, and LR, is the focal point of the study in [50]. This study extensively compares the outcomes concerning various metrics, such as the F1 score, accuracy, area, recall, precision, and error rate. The results show that RF and SVM algorithms perform almost similarly in terms of all parameters, achieving high accuracy rates of 96.85% and 96.74%, respectively. RF gives a true negative rate of 95.65%, while SVM yields a true negative rate of 95.87%. The precision and F-measure scores are also high for both algorithms. It also emphasizes the importance of detecting and preventing phishing attacks in IoT devices to safeguard online transactions. An ML-based IDS for an Energy Aware Smart Home (EASH) framework [51] is designed to address service failures, security issues, and abnormality in the Cyber-Physical System (CPS). The system analyzes network attacks and communication failures using ML techniques to differentiate abnormality sources. The framework achieves an 85% accuracy rate and shows potential for further development to enhance accuracy. In [52] an analysis of Bot-IoT is carried out, which contains about 73 million instances and is used to detect numerous botnet attacks in IoT networks using ML. The paper discusses the dataset's features, pitfalls, and data cleaning procedures, as well as its use in published research. In their overview of machine learning (ML) approaches to IoT security, the authors in [3] delve into the security challenges associated with the Internet of Things (IoT). Security issues related to IoT are discussed and six research questions are addressed and identified as emerging research directions shaping future investigations in this area [53]. The work emphasizes the significance of developing models that can incorporate the latest advancements and technologies from big data and ML to identify IoT attacks with high accuracy and efficiency in real or nearly real-time.

Exploring security issues in the Web of Things (WoT), [9] highlights the vulnerabilities of this expansive network of services and applications to attacks such as sinkhole, overhang

dropping, and denial of service (DoS). To address these concerns, the use of an IDS based on ML algorithms is suggested. It emphasizes the importance of a properly designed IDS to protect the future network infrastructure of smart cities. It also draws attention to the constraints of the widely used cloud-based IDS, which deals with high latency, resulting in slow detection of malicious users. The study in [11] investigate the effect of intrusion detection in the face of increasing web organization and the need to protect against data theft and information security risks. It presents two different approaches to intrusion detection, one using association rule data mining and the other using a ML algorithm SVM, and comparing their outcomes on the KDD Cup '99 dataset. The results show that SVM outperforms in terms of precision. It is reported that the model achieves excellent accuracy in detecting the four types of attacks, in contrast to the existing methodologies. Research in [54] underscore the importance of data security measurement and privacy in the IoT and the need for dynamic methods to address the nature of attacks. It emphasizes the use of ML techniques for IDS in IoT networks and explores various architectures and methods for identifying compromised IoT devices. It provides an analysis of the contemporary literature on attack types and ML-based IDS, focusing on IoT attacks.

In [55] ML techniques for detecting and identifying IoT devices are compared, with a focus on non-cryptographic approaches. Rogue IoT devices pose a serious security risk to the IoT ecosystem, and detecting and identifying them is a crucial first step in securing the network. It discusses various ML-related enabling technologies, such as learning algorithms, feature engineering, incremental learning, and abnormality detection. A system called ANTE is proposed for early detection of botnets in the IoT using ML algorithms [56]. The evaluation of the system is assessed by testing it on four typical datasets. Similarly, a framework [57] is proposed for a ML-IDS for detecting IoT network attacks, due to the growing security risks and privacy posed by the proliferation of IoT devices. The framework uses feature scaling, dimensionality reduction, and six different ML models to analyze the UNSW-NB15 dataset. The experimental evaluation shows that the proposed ML-IDS achieved competitive accuracy, recall, F1-score, precision, Mathew Correlation Coefficient (MCC), and kappa compared to existing works [58]. The framework suggests that ML-supervised algorithm-based IDS for IoT can help enhance the privacy and security challenges of IoT A framework by [59] uses a feature selection and ML-based approach to optimize security in the IoT network. The presence of cyberattacks and security risks represents a substantial threat to the integrity and stability of IoT systems. Consequently, the development and implementation of a machine learning framework

are imperative to effectively mitigate these risks and ensure the robust security of IoT environments. The proposed approach in [60] involves dimension reduction using linear discriminant analysis (LDA) to reduce the size of network data, followed by training machine learning predictors using the reduced dataset. The efficiency of the approach is assessed using various criteria, and the results show the impact of the ML approach in enhancing security in the IoT network.

Addressing the challenge of imbalanced multiclass output data in IoT datasets, the study presented in [61] introduces an Intrusion Detection System (IDS) for the IoT that employs the XGBoost model. The accuracy of the proposed method is assessed using two IoT imbalanced datasets, and the XGBoost model demonstrated exceptional attack detection capabilities. It emphasizes the need for high security levels to protect IoT networks, which are easily attacked by cybercriminals, and ML-based IDSs as a crucial security solution to monitor safe network traffic and detect network's abnormal behavior to avoid attacks Dissertation [62] present a method in which SVM is used to detect port sweep attacks and compared it with an ensemble-based hybrid classification approach that used weak and strong classifiers. The results evaluate that the ensembled approach performed better, with a Random Forest accuracy of 99.82%, ANN accuracy of 99.70%, and SVM accuracy of 93.29% on intrusion detection rather than attack type classification. Within the context of the banking sector, which faces elevated cybersecurity risks due to the sensitive nature of its data, [20] presents a novel strategy for identifying Distributed Denial of Service (DDoS) attacks in IoT-based monitoring systems. The banking industry is particularly vulnerable to cyberattacks due to the high value of the data they hold. The method uses a Banking Dataset and applies multiple classification models, including SVM, KNN, and RF algorithms. The SVM algorithm achieves the highest accuracy score of 99.5%. It is concluded that the SVM algorithm is more resilient in detecting DDoS attacks than KNN, RF, and other existing ML and DL techniques.

Addressing the escalating cybersecurity threats posed by the proliferation of the IoT in institutions and companies, the study in [63] discusses the use of AI in cybersecurity to combat the increasing cybersecurity threats faced by institutions and companies due to the rise in the use of the IoT. A system is proposed that combines a CNN with a LSTM algorithm to detect BASHLITE and Mirai attacks on four types of security cameras, which are common and severe attacks in the IoT. The suggested system achieves optimal performance, with high F1 scores,

recall and precision, for detecting botnet attacks on IoT devices, specifically cameras. The system's success highlights the potential of AI in cybersecurity to detect and mitigate sophisticated cyber threats. In the context of network traffic anomaly detection, [64] propose an ML based IDS to detect anomalies in network traffic. The CICIDS2017 data set is used, which includes both current and historical attacks, and implemented multiple classifiers such as DT, NB, RF and LR after preprocessing the data set through cleaning, normalization, oversampling, and feature selection. It highlights the importance of using machine learning-based models in detecting sophisticated and undetected threats quickly and efficiently.

A Framework by [24] used for botnets detection using ML techniques. The existing techniques for botnet detection are not effective for detecting botnets at an early stage. Honeypots are utilized to lure botnet infiltrations and reveal the identities and activities of the attackers, causing attackers to avoid them. ML techniques are used to support the detection and prevention of bot attacks. The proposed framework employs an Ensemble classifier with Stacking Process (ECASP) to identify the most suitable features for the machine learning classifiers to use as input. The proposed method achieves an accuracy of 94.08%, sensitivity of 86.5%, specificity of 85.68%, and F-measure of 78.24%. In response to the escalating cyber threats and vulnerabilities in the healthcare ecosystem, [65] proposes a ML approach to analyze cyber threats and exposure in the healthcare ecosystem. The healthcare industry has become more vulnerable to cyber-attacks, owing mostly to the increasing deployment of electronic health records and the integration of connected medical devices. In the healthcare system, the approach can be used to manage risks to identify potential patterns of security issues.

In their survey [66] focuses on privacy protection techniques for IoT, based on ML and DL. They highlight the need for data privacy in IoT applications, which are extensively adopted in the eHealth environment, industries, smart cities, and autonomous vehicles. They analyze the current privacy threats and attacks, present various ML and DL architectures proposed for privacy protection in IoT, and provide implementation details and publish results of each method. They conclude by identifying the most efficient solutions to address various privacy threats and attacks in IoT. Introducing a novel approach for cybersecurity threat identification and classification in the Internet of Things (IoT) environment, [67] harnesses machine learning and artificial intelligence (AI) tools.

The approach, called Mayfly Optimization-Regularized Extreme Learning Machine (MFO-RELM), preprocesses the IoT data and carries out classification using the RELM model. The proposed study utilizes the Manta Ray Foraging Optimization (MFO) algorithm to improve the productivity to identify attacks with the RELM model. Presenting a machine learning-based approach in [68] the study focuses on the detection of network fault severity and the identification of IoT devices based on multiple attributes. The proposed framework captures different features from each network flow to identify various network attacks. Experiments with different ML algorithms such as SVM, DT, and RF, and found that RF and DT classifiers are carried out, the best performing ML algorithms with an accuracy of about 96.98%. Whereas, Gaussian Naive Bayes is found to be the least performing model with an accuracy of about 54.41%.

Introducing an innovative edge-cloud architecture called LocKedge for detecting cyberattacks in IoT networks, [94] with low complexity and high accuracy. The authors implemented the proposed mechanism in two ways, namely centralized and federated learning, and evaluated its performance by comparing it with other ML and DL methods. The evaluation was accomplished using the BoT-IoT dataset. The results indicate that LocKedge exhibits superior performance in accuracy and complexity, and can detect attacks quickly at the edge layer, reducing the workload of the cloud. In their discussion, [95] highlight the importance of detecting and monitoring emerging security risks in IoT systems, given the increasing elegance and variety of security threats. They classify IoT security risks and issues and review current Network Intrusion Detection Systems (NIDS) methods, data sets, and sniffing applications for IoT networks. Proposing a general framework for intrusion detection in IoT using Graph Neural Network (GNN) technologies, [96] relies on a network embedding feature representation to address labeled data that is high-dimensional, redundant, and rare within IoT. They use a network embedding feature representation as a solution for handling labeled data that is high dimensional, redundant, and rare in the context of IoT. This approach is motivated by the challenge of obtaining labeled samples and the imbalanced distribution of sample categories. The suggested method was tested using public datasets and outperformed several contemporary algorithms in many evaluation metrics.

Addressing the susceptibility of IoT to cyberattacks due to its distributed nature, [97] introduces a mechanism tested with the latest dataset and benchmark algorithms and achieves

a 99.94% detection rate with 0.066(ms) time, showing an auspicious outcome in terms of accuracy and speed efficiency. The architecture can efficiently detect sophisticated bot attacks, making it a reliable solution for IIoT security against botnet attacks. Based on misuse or anomaly detection, a representative survey [7] evaluated the security challenges faced by smart devices and networks. An analysis of ML approaches for IoT security, different types of attacks, and IoT architecture is provided, along with proposed solutions based on ML. Providing a survey of IDS for IoT from 2015 to 2019, [98] explores various IDS deployment and analysis strategies within IoT architecture. They explore diverse IDS deployment and analysis strategies in IoT architecture, while emphasizing the security challenges and issues involved in IoT. They also examine various intrusions in IoT and review the recent updates, challenges related to ML and DL techniques, and security issues overall, they provide a valuable reference for researchers and practitioners working on IoT security and IDS. In their overview of eighty studies published between 2016 and 2021,

2.2.2 Deep Learning

A framework called Deep Feature Embedding Learning (DFEL) is proposed by [69] for detecting cyberattacks in IoT environments. The framework uses deep learning to improve accuracy and reduce detection time. DFEL successfully detects internet intrusion and meets detection performance and speed. Emphasizing the critical role of security in the success of IoT networks and the significant losses incurred due to Distributed Denial of Service (DDoS) attacks, [70] employs the CICIDS2017 dataset to evaluate their proposed deep learning models for DDoS attack detection. The authors use CICIDS2017 dataset to evaluate their proposed deep learning models for DDoS attack detection, achieving an accuracy of 97.16%. They also compare their models to traditional ML algorithms and identify open research challenges in the use of DL for IoT cyber security.

In their proposal [71] introduce a methodology to turn network traffic data into an image form and train a ResNet CNN model to efficiently detect these attacks. Also they suggest integrating AI-based techniques like deep learning models can improve the reliability and effectiveness of traditional security solutions in detecting complex attacks. Providing a

comprehensive taxonomy of notable deep learning (DL) techniques used in designing network-based Intrusion Detection Systems (IDS), [72] reviews recent articles on Network Intrusion Detection systems (NIDS), emphasizing their strengths and limitations. The study also highlights research challenges and potential future improvements in ML and DL-based NIDS. Introducing a novel security framework for IoT attack detection based on a deep learning model, [73] conducts experimental evaluations using a dataset derived from twenty infected IoT devices running on Raspberry Pi. The experimental evaluation employs a dataset from twenty infected IoT devices running on Raspberry Pi, and the proposed model achieves an attack detection accuracy of 96%. It emphasizes the importance of machine learning models in utilizing the enormous amount of data originated by IoT devices to ensure their security.

Dissertation [74] propose a DL-based anomaly detection (AD) system for cellular IoT devices to address the security challenges and device management of existing IoT networks. The system is demonstrated to be effective in identifying anomalies in IoT devices with potential applications in smart logistics. The author in [75] validate a proposed model using various intrusion detection datasets and compare their results with existing deep learning implementations. According to the results, DL methods, especially CNN models, are effective in identifying anomalies accurately in IoT networks and can be used for intrusion detection systems. Research [76] discuss the use of DL-based models for identifying anomalies in time-series data. Experiments with different look-back values and timestamps are reported by the authors. According to the authors, anomaly detection techniques should be accurate and efficient in IoT and cybersecurity contexts. Exploring the end-to-end security challenges within IoT ecosystems [77] the unique security issues that IoT solutions face and the need to address them to ensure the security of IoT products and services. They provide an overview of IoT and identify potential threats to IoT security, along with their attack surfaces and associated hazards. They also discuss various DL approaches for IoT security, their strengths, weaknesses, and opportunities for future research. Proposing an ensemble deep learning model for cyber threat hunting in the IoT,[78] tackles the imbalanced nature of IoT datasets by extracting new balanced data. Their model achieves high accuracy, surpassing other models in the comparison

The importance of securing smart electronic gadgets, and how traditional techniques may not be suitable for devices with resource constraints [64]. The research evaluates and compares the efficiency of CNN, DNN and LSTM algorithms. The Deep Convolutional Neural

Network (DCNN) model [79] consists of two convolutional layers and three fully connected dense layers. The model is evaluated on the IoTID20 dataset, and its performance is analyzed using different performance metrics. Several optimization techniques, such as Nesterov-accelerated Adaptive Moment Estimation (Nadam), Adaptive Movement Estimation (AdaMax), and Adaptive Moment (Adam), are applied to the proposed model, which outperforms various advanced DL and traditional ML techniques. A Cognitive Adaptive System (CAS) is designed for industrial production systems [80]. The system utilizes Reinforcement Learning (RL) in conjunction with IoT edges to overcome a variety of challenges and enhance productivity.

A DL-based IDS framework [81] is proposed for Agriculture 4.0 networks to detect DDoS attacks. The proposed system combines LSTM and CNN architectures to achieve high F1-score, precision, accuracy and recall, in identifying different types of DDoS attacks using the CIC-DDoS2019 dataset. The proposed model outperforms existing systems and achieves a high degree of precision. It is concluded that the proposed model offers the highest level of protection against cyber threats to Agriculture 4.0. A DL-powered anomaly detection model for IoT systems is proposed to identify malicious data and ensure their exclusion from IoT-driven decision support systems [82]. The proposed DL model uses a denoising autoencoder to obtain robust features that are not significantly affected by the unstable environment of IoT. The proposed model's effectiveness in improving the accuracy of identifying attacks in IoT models is demonstrated through experimental results. The DL-based intrusion detection model named DIDS for IoT networks [83] is proposed that includes the prediction of unknown attacks to reduce computational overhead, increase throughput, and achieve a low false alarm rate. The proposed method outperforms standard algorithms in detecting attacks earlier with reduced computational time, achieving 99% accuracy in detecting attacks. It highlights the significance of AI techniques in IoT networks, making life easier and more connected. The challenges of providing security for IoT devices is discussed and a DL algorithm is proposed for IDS [84]. The evaluation of the proposed model is assessed using a commonly used dataset, and compared against the efficacy of established intrusion detection methods. [85] proposes a model for IoT networks to detect DDoS attacks, in which systems are restricted by their limited resources and dynamic communication, leading to security limitations. The model combines three DL algorithms, RNN, LSTM-RNN, and CNN, to develop a bidirectional CNN-BiLSTM DDoS

detection model that accurately detects and distinguishes DDoS from legitimate traffic. The four models were evaluated using the CICIDS2017.

Underlining the pivotal role of Intrusion Detection Systems (IDS) in safeguarding the IoT environment, [86] discuss the importance of IDS in securing the IoT environment. They highlight the significance of efficient security management techniques and the secure functioning of IoT devices. They also propose a new IDS based on deep learning, called fuzzy CNN, to improve the security of IoT communication. The proposed IDS is found to be beneficial in optimizing the accuracy of detection, detecting DoS attacks efficiently, and reducing false positive rates. They provide a thorough examination of IoT network security using quality-of-service metrics, and compare them with currently existing security metrics, leading to a detailed analysis. In their Study [87] present a DL approach for IDS in the IoT using the focal loss function to tackle the problem of imbalanced datasets. The proposed approach is implemented using two established neural network architectures and assessed with three datasets from various IoT domains by the authors. Compared to existing intrusion detection models, the proposed method outperforms them in terms of precision, recall, accuracy, and MCC. They conclude that their approach using the focal loss function is effective for training DL models for IDS in IoT.

2.2.3 Ensemble Methods

The growing risks of cybersecurity threats and attacks in smart cities due to the widespread deployment of IoT applications are discussed in [88]. Stacking, bagging, and boosting are among the methods the authors explore to improve the detection system. The authors further explore the integration of multi-class classification, cross-validation, and feature selection techniques to enhance the accuracy of the system. The experimental outcomes suggest that the proposed methodology is efficient in identifying cyberattack. In their research [89] propose an ensemble method and deep learning model to classify the type of attack and protect IoT devices. Their method significantly enhances network-level security in IoT devices that utilize the Message Queuing Telemetry Transport (MQTT) protocol by employing ML techniques to effectively diagnose and mitigate attacks. The system allows dataset owners to

securely store and manage their data, preventing tampering and ensuring the integrity of the data. Botnet attacks, in particular, are a serious and widespread threat to IoT devices. A study by [90] use the real N-BaIoT dataset, which includes both malicious and benign patterns, to train and test the CNN with a Long Short-Term Memory (LSTM) model. Overall, the CNN-LSTM model detects botnet attacks from a wide range of IoT devices with high accuracy.

Introducing an innovative approach for identifying Distributed Denial of Service (DDoS) attacks, [91] combines supervised and unsupervised machine learning algorithms. By using an advanced data processing framework, this method separates anomalous traffic from normal data using a clustering algorithm based on the CICIDS2017 dataset. In their proposal [3] present a new technique called IDS based on Spike Neural Network and Decision Tree (SNNDT) for IoT devices which uses Decision Tree (DT) to select optimal samples and SNN to identify cyber-attacks. The SNN uses Non-Leaky Integrate and Fire (NLIF) neurons and a Random Order Code (ROC) technique to minimize devices' power usage and reduce latency. Simulation results show that IDS-SNNDT has high detection accuracy and less latency with low power usage as compared to the other two methods. Proposing an efficient NIDS named DCNNBiLSTM [92] based on DL techniques. The system has been trained and tested using real-time traffic datasets, achieving an excellent accuracy rate on CICIDS2018 and Edge_IoT datasets.

The authors highlight the exceptional performance of DL in numerous identification tasks. With the increasing reliance on Internet technology and the subsequent surge in data generation, ensuring network security has become a pressing concern. The research paper [93] proposes a novel XGBoost-DNN model for network intrusion detection, which combines the XGBoost technique for feature selection and deep neural network (DNN) for classification. The experiments conducted on the NSL-KDD dataset demonstrate the superiority of the proposed model over existing shallow machine learning algorithms such as logistic regression, SVM, and naive Bayes in terms of accuracy, precision, recall, and F1-score. The XGBoost-DNN model outperforms these algorithms, achieving a consistent classification accuracy of 97%. Notably, the paper introduces a unique feature selection method using the importance score generated by XGBoost. The research highlights the potential of deep learning models in intrusion detection and suggests further exploration for multiclass classification in the future.

In [13] the utilization of DL and ML techniques in IoT security and their performance in identifying attacks is discussed. The study also identifies smart IDS with intelligent architectural frameworks. SVM and RF are among the most used methods, Dissertation [15] presents a systematic review of botnet detection techniques based on ML in the context of software-defined networking (SDN). The authors highlight the significant threat that botnets pose to organizations and the advantages of SDN in network management. The survey evaluates several dissertations published from 2006 to 2015 for botnet attack detection and SDN, respectively. The authors follow Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines for their evaluation and address current research challenges in the area while proposing directions for future research. The article [99] provides a summary of the security challenges associated with IoT systems and highlights the insufficiency of traditional security techniques in addressing them. The authors suggest that utilizing ML and DL solutions is essential for dynamically improve the security of IoT devices. They explore the potential of AI expertise in providing an up-to-date security system for upcoming IoT systems. In their discussion [100] address the security concerns associated with the widespread use of IoT technologies and the potential use of ML to address these issues. The authors provide an overview of the major security challenges encountered by IoT systems, and review the contemporary ML-based solutions that have been proposed to mitigate them.

Emphasizing the importance of protecting digital systems from potential threats and attacks, [101] explores how machine learning (ML) models can be employed to predict and detect these threats. They focus on comparing the accuracy of different linear ML algorithms for detecting cyber-attacks. They also mention the use of balance procedures in the presentation and how the classifiers' accuracy is compared. Presenting a detailed survey of existing research on malware detection, [102] highlights the strengths and weaknesses of different ML algorithms. The findings of the study suggest that ML can be an effective approach for identifying malware attacks. The paper provides valuable insights for researchers and practitioners seeking to implement ML algorithms for detecting malware in enterprise information systems based on IoT. Proposing the Decision Tree Twin Support Vector Machine and Hierarchical Clustering (HC-DTTWSVM) method for the effective detection of various types of network intrusion, [103] demonstrates comparable evaluation in terms of detection capability when compared to a recently proposed Network Intrusion Detection (NID) method. The study's findings demonstrate that Generic samples have the most elevated levels of

Precision, Recall, F1-score, and G-mean. In general, the comparative analysis suggests that the HC-DTTWSVM algorithm is an efficient approach for NIDS, surpassing the detection accuracy of several recently proposed methods. The analysis of the literature shows that the HC-DTTWSVM algorithm improve detection accuracy with 81.21% and 85.95%on the UNSW NB 15 and NSL KDD datasets respectively.

2.3 Summary:

In conclusion, this literature review has highlighted the prominence of DL-based methods, such as CNN and LSTM, in achieving high accuracy in detecting a variety of network attacks. The integration of feature selection and extraction methods has shown substantial enhancements in IDS performance. Future research directions should emphasize multi-modal approaches, real-time detection, adaptability, resource-efficient solutions, data diversity, scalability, user-friendly interfaces, and standardized benchmarks to propel advancements in the field of cyber-attack detection and identification. That requires the need of further investigation

CHAPTER 3

METHODOLOGY

3.1 Overview

The proposed methodology for identifying cyber-attacks on IoT devices is shown in Figure 3.1. It consists of employing different ML techniques to accurately identify and classify cyber-attacks. The proposed method combines the strength of XG-Boost and DNN, named as X-DNN. Each block in the diagram represents a key step in the process and is explained in detail in the subsequent sections.

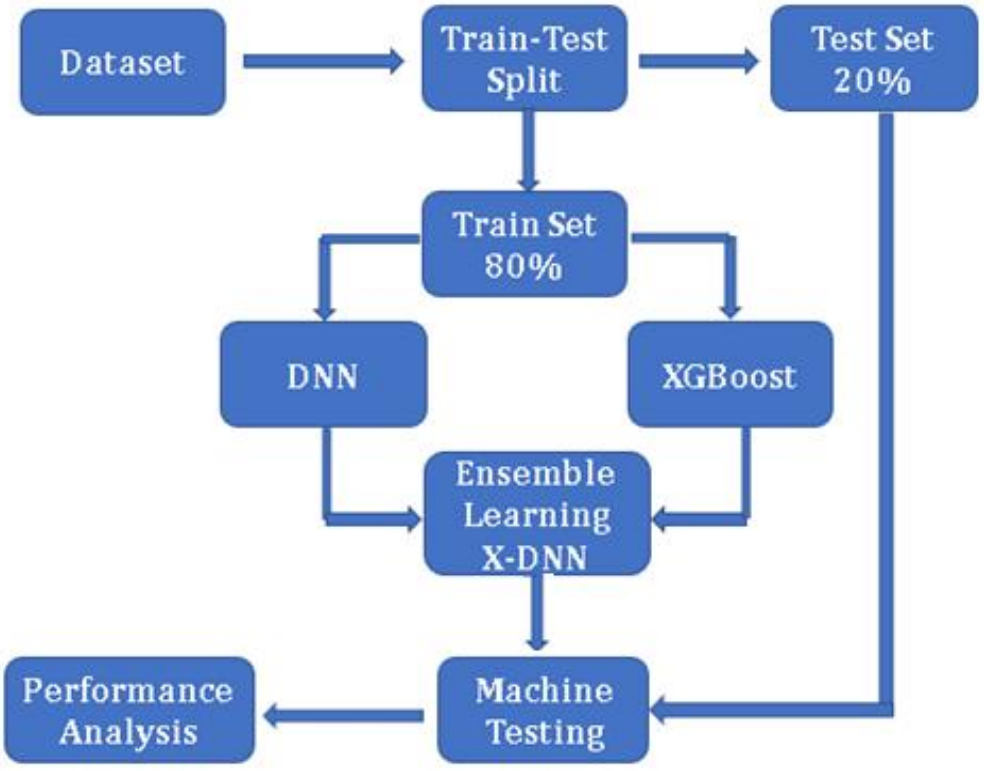

Figure 3.1: Steps used in the proposed method to identify cyber attacks

3.2 Proposed Method

Ensemble learning [116] is a ML approach that integrates multiple models to enhance the predictive model's precision, accuracy, robustness and resilience. This technique involves aggregating the outputs of several individual models to generate a consolidated prediction. There are various techniques for ensemble learning such as stacking, boosting, and bagging and others. These techniques aim to combine different models in a way that compensates for the weaknesses of individual models and leverages their strengths to produce a more accurate prediction. Stacking is a powerful ensemble technique that can combine the predictions of multiple ML models, in order to enhance the overall performance. It can be used to leverage the strengths of different models, while mitigating their weaknesses. In this thesis the strength of XGboost and DNN are combined, which result in a new method that is called X-DNN.

The X-DNN consists of DNN and XGBoost, both are trained on the UNSW NB15 and NSL KDD datasets to classify the various types of cyber-attacks. Both DNN and XGBoost models are combined together to enhance the overall accuracy of X-DNN. The model consists of two separate inputs, one for the DNN model and the other for the XGBoost model. Three dense layers with 64, 32, and 10 units each are included in the DNN model, and the hidden layers are activated using ReLU, while the output layer is activated using softmax function.

Using both models DNN and XGBoost together to improve classification performance, an ensemble technique known as model averaging is employed. This technique involves obtaining prediction scores from both the DNN and XGBoost models and computing the average probabilities for each data point. Based on the combined predictions of the two models, we leveraged the complementary capabilities of the DNN in capturing hidden patterns and the apply XGBoost in handling data analysis. Ensembling of the DNN and XGBoost models through averaging yields significant benefits over using individual models alone. The ensemble approach enable us to achieve improved accuracy and robustness in classification compared to using either model in isolation or other ensemble techniques. The experimental results show the effectiveness of ensemble technique in enhancing classification accuracy and effectively addressing the challenges presented by the datasets

3.3 Datasets

The selection of an appropriate dataset is crucial for the successful implementation of any ML-based approach. To select a dataset for this research, several factors were taken into consideration, including the type, quality, size and the diversity of cyber-attack types present in the dataset. It is also vital to consider the representativeness of the data with respect to the problem being addressed, and the availability of labeled data.

Two different datasets are used which are, UNSW-NB15 [104] and NSL KDD [105] datasets. UNSW-NB15 dataset contains 257,673 instances of network traffic records with different types of cyber-attacks, i.e., Fuzzers, DoS, Reconnaissance, Backdoor, Exploits, Analysis, Worms, Shellcode, Generic, and normal traffic. Moreover, the UNSW-NB15 dataset serves as a benchmark dataset, which has been widely used in various researches [3] [8] [11] [46] [57] [103] [106]. The dataset also includes a ground truth, which is important for the evaluation of ML models.

Fuzzers are attack methods that involve sending a large amount of random or unexpected data to applications or systems to identify vulnerabilities. Denial of Service attacks aims to disrupt the availability of network services by overwhelming them with a flood of requests, rendering them unavailable for legitimate users. Reconnaissance attacks are exploratory in nature, involving an attacker probing a network to gather information about potential vulnerabilities or targets. Backdoor attacks aim to create unauthorized access points in a system, allowing attackers to bypass normal security controls and gain persistent access. Exploits involve taking advantage of known vulnerabilities in software or systems to gain unauthorized access or execute malicious code. Analysis attacks include efforts to scrutinize a target system's behavior and characteristics to gain insights for further exploitation. Worms are self-replicating malware that spread across networks, often causing widespread damage and data loss. Shellcode attacks involve injecting malicious code directly into a system's memory to execute unauthorized actions or provide remote control. Generic attacks are broad and less specific in nature, often used as a catch-all category for unclassified or generalized cyber threats. Normal traffic represents legitimate network activity without malicious intent, serving as a baseline for comparison in cybersecurity research and threat detection.

The NSL KDD dataset comprises a total of 125,972 instances of network traffic records. These records encompass various types of cyber-attacks, which are primarily classify into five attack types: Probe, DoS, Remote to Local (R2L), User to Root (U2R), and Normal. Each instance in the dataset comprises of 44 different features that capture relevant information about the network traffic.

Probe attacks involve the systematic exploration of a network to gather information about its configuration, services, and vulnerabilities. Attackers aim to identify potential weaknesses without directly causing harm. Denial of Service attacks disrupt network services by overwhelming them with excessive traffic, rendering them inaccessible to legitimate users. This type of attack can disrupt normal network operations. R2L attacks attempt to exploit vulnerabilities in remote systems to gain unauthorized local access. Attackers aim to escalate their privileges and gain control over the target system. U2R attacks involve a regular user trying to exploit vulnerabilities to gain superuser privileges on a system. Attackers aim to elevate their access level to the highest possible. Normal traffic represents legitimate network activities and serves as a baseline for comparison in cybersecurity research. It includes routine, non-malicious network communication.

3.3.1 Preprocessing of dataset:

The preprocessing steps are applied to the UNSW NB15 and NSL KDD dataset before using them for the training and testing of X-DNN.

Data pre-processing, or data preparation, is the first step in the data processing operation. The extraction of flow-based features from raw data is the initial stage in data processing. This stage aims to transform the data into a format that ML algorithms can learn. The dataset progresses through subsequent stages encompassing data cleaning, feature engineering, and feature scaling.

Data cleansing requires an understanding of the significance of each feature, its anticipated data type, and its possible range of values. Without this knowledge, it becomes challenging to differentiate between acceptable and unacceptable values. In the data cleaning

stage, the rows in the data frame that contain missing values, also known as "NaN and Null" values, are eliminated to ensure that the dataset used for analysis is complete and accurate.

The UNSW NB 15 and NSL KDD datasets are carefully checked for any missing values and anomalies in data. The missing data is handled by deleting the instances with missing values using dropna function of Python. The anomalies in the data are also checked and corrected manually where necessary.

Feature engineering [107] involve selecting, transforming, and creating features (variables) that can be used as inputs for a ML algorithm to enhance its performance in making predictions or classifications. In label encoding, categorical variables are transformed into numerical values that can be used in ML models.

In the process of feature engineering, one of the employed data preprocessing techniques is called label encoding. This technique is utilized to transform textual data into numeric format. The label encoder assigns a unique numerical value to each category in the feature, which is then used as input to ML algorithms.

Feature scaling [108] is another important practice in data preprocessing. Standard scaling transforms the data to have a mean of 0 and a standard deviation of 1, which normalizes the data and improves the efficacy of ML algorithms. Feature scaling is the process of transforming the data such that all the features or variables are on the same scale. Standard Scaler is one of the widely used scaling techniques in ML, which ensures that the data has zero mean and unit variance. The scalar object is fitted on the training data to learn the mean and standard deviation of each feature, and the learned parameters are used to scale datasets. Standard Scaler scales each feature, ensuring that each feature has a mean of zero and variance of one.

3.3.2 Selection of top-ranking features

After applying these data preprocessing techniques to the UNSW NB 15 and NSL KDD dataset, a transformed dataset with encoded categorical features and standardized numerical features are obtained. This transformed dataset was then used in further analysis, including feature selection, to recognize the most important features for detecting network intrusions.

Table 3.1: Selection of Top ranked Features from the UNSW NB 15 Dataset

Sr. No.	Feature	Description
1	**Dpkts**	Total packets sent to the destination port
2	**Rate**	Packet rate, indicating the speed of data transmission
3	**Sttl**	Source Time to Live, remaining lifespan of a packet before it's discarded
4	**Sload**	Load in packets per second for the source
5	**Dload**	Load in packets per second for the destination
6	**Sinpkt**	Total input packets from the source
7	**Swin**	Window sizes at the source, indicating the data quantity that can be sent before acknowledgment
8	**Dwin**	Window sizes at the destination, indicating the data quantity that can be sent before acknowledgment
9	**Stcpb**	Source TCP sequence number base values
10	**Dtcpb**	Destination TCP sequence number base values
11	**dmean**	Mean value for the destination across various attributes
12	**ct_srv_src**	Count of connections sharing the same service source
13	**ct_state_ttl**	Count of connections with the same state TTL
14	**ct_dst_ltm**	Number of connections with the same destination Last Time Modified (LTM)
15	**ct_src_dport_ltm**	Count of connections sharing the same source destination port LTM
16	**ct_dst_sport_ltm**	Count of connections sharing the same destination source port LTM, respectively
17	**ct_dst_src_ltm**	Count of connections with the same destination source LTM
18	**ct_src_ltm**	Count of connections with the same source LTM
19	**ct_srv_dst**	Count of connections with the same service destination
20	**is_sm_ips_ports**	Indicates whether the IP or port is small, potentially signaling unusual or malicious traffic

UNSW NB 15 contains 43 features, whereas NSL KDD contains 40 features. To decrease dataset dimensionality and improve learning algorithms' efficiency, a feature selection process is applied. The process of feature selection, particularly using the SelectKBest technique, plays a crucial role in refining the models' ability to accurately detect network intrusions. By focusing on the 20 most significant features, as identified by their ANOVA F-values, we ensure that the models are not burdened by redundant or irrelevant data, which can often lead to overfitting or decreased performance. This careful pruning of the feature set not only enhances the efficiency of the learning algorithms but also contributes to a more interpretable model.

The selected features from the UNSW NB 15 dataset provide essential insights into network traffic and behavior. SelectKBest function is employed to select the 20 most significant features according to their respective importance scores. The selected features are then used to train the ensemble model. Table 3.1 and 3.2 presents the 20 most significant features achieved through the selectKBest feature selection methodology on both datasets.

Table 3.2: Selection of Top Ranked Features from the NSL KDD Dataset

Sr. No.	Feature	Description
1	**num_failed_logins**	Counts the number of failed login attempts, indicative of unauthorized access or potential security breaches
2	**num_compromised**	Counts the number of compromised conditions, with an elevated count indicating security vulnerabilities or unauthorized access
3	**is_host_login**	Binary feature determining if the login is by a host, distinguishing between regular user logins and potential host-level logins
4	**is_guest_login**	Binary feature identifying guest logins, crucial in understanding the nature of login attempts
5	**srv_count**	Represents the number of services accessed during the connection, offering insights into the diversity of services used
6	**serror_rate**	Indicates the percentage of connections that encountered errors from a general perspective
7	**srv_serror_rate**	Indicates the percentage of connections that encountered errors specifically during service access
8	**rerror_rate**	Indicates the percentage of connections that encountered general errors

9	**srv_rerror_rate**	Indicates the percentage of connections that encountered errors specifically during service access
10	**same_srv_rate**	Calculates the percentage of connections using the same service, helping to identify patterns of repeated service usage
11	**srv_diff_host_rate**	Focuses on the percentage of connections to different hosts, revealing the diversity of host connections
12	**dst_host_count**	Represents the number of destination hosts, providing insights into the overall network structure
13	**dst_host_srv_count**	Represents the number of services on destination hosts, offering insights into the diversity of services on the network
14	**dst_host_same_srv_rate**	Calculates the percentage of connections to hosts offering the same service
15	**dst_host_diff_srv_rate**	Gauges diversity in service usage by representing the percentage of connections to hosts offering different services
16	**dst_host_same_src_port_rate**	Provides insights into source port patterns
17	**dst_host_srv_diff_host_rate**	Offers insights into destination port patterns
18	**dst_host_serror_rate**	Indicates the error rate for connections to destination hosts from a general perspective
19	**dst_host_srv_serror_rate**	Indicates the error rate for connections to destination hosts specifically during service access
20	**dst_host_rerror_rate**	Indicates the error rate for connections to destination hosts from a general perspective

3.3.3 Train Test Split

The train-test split is a common practice in machine learning and data analysis to assess the generalization ability of a model. It involves randomly partitioning the dataset into two separate subsets: 80 percent the training set and 20 percent testing set.

3.4 Machine and Deep Learning Algorithms

The study explores a range of algorithms employed in this research. The focus is on various algorithms, revealing how machines understand and tackle complex tasks. This encompasses the exploration of computational processes designed to discern patterns from data and make informed decisions. Various algorithms are scrutinized, providing a comprehensive insight into their applications and implications within classification of cyber attacks on IoT. The ML and DL algorithms used are as follows:

3.4.1 Support Vector Machine

SVM [109] is a well-known and powerful ML technique that provides high accuracy while requiring computational power. SVM technique utilizes the features to create a decision boundary or hyperplane, followed by applying classification methods to differentiate among multiple groups. When the data size is not substantial, SVM outperforms neural networks. The classifier functions perform well in distinguishing harmful IoT traffic from safe traffic. This transforms a low-dimensional feature space into a high-dimensional feature space by employing sophisticated kernel functions. SVM has several kernel functions that are utilized to improve model learning and optimize performance.

3.4.2 Decision Tree

A DT [110] is a classifier that makes use of a series of if-then-else decisions to identify the label for a given observation. The benefit of using DT is that they are easy to interpret and can be used to model non-linear relationships. The DT model is applied on the train set and its performance is evaluated using the test set.

3.4.3 Random Forest

The Random Forest (RF) algorithm [111] is another well-known ML method. This technique uses many unique decision trees to find the best results, with their findings combined (referred to as bagging) to get good outcomes. The fact that random forest splits features into tiny samples instead of relying on feature correlation makes it different from traditional. After partitioning the preprocessed dataset into train and test sets, the RF algorithm is applied.

3.4.4 Logistic Regression

In LR [111] the input features are compared to the probability of belonging to a class based on their relationship. Based on a logistic function, LR estimates the likelihood of an instance belonging to a particular class (sigmoid function). It calculates the weighted sum of the input features and their corresponding coefficients, and then applies the logistic function to transform this linear combination into a probability value. LR optimizes the coefficients (weights) during the training phase using techniques such as maximum likelihood estimation or gradient descent to find the optimal values that minimize the difference between the predicted probabilities and the actual class labels.

3.4.5 Deep Neural Network

DNN [112] are complex neural networks with input and multiple hidden neurons at output layers. The hidden number of neurons is primarily responsible for parameter optimization. Dropout regularization is a technique used for reducing overfitting while improving deep neural network generalization.

At a high level, a DNN model can be viewed as a series of layers, where each layer performs computation on the input data, passing the output to the next layer until the final layer produces the model's prediction. The layers themselves are composed of neurons that take in

the output from the previous layer and apply some transformation to it, typically in the form of a nonlinear activation function.

The DNN model used in this thesis is a feedforward neural network architecture consisting of multiple densely connected layers. The DNN model incorporates three dense layers with 64, 32, and 10 units (neurons) with reLu as activation function and softmax as action function at the output neuron for the UNSW NB 15 dataset whereas for NSL KDD dataset three dense layers with 64, 32 and 5 units are used. The dense layers are followed by Rectified Linear Units (ReLUs) activation functions, introducing nonlinearity. The ReLU activation function is applied to the output of each dense layer, except for the output layer.

The input data is fed into the DNN model, and it undergoes forward propagation through the layers. During forward propagation, the inputs are multiplied by the weight matrices associated with each layer, and biases are added. The outputs from the previous layer serve as inputs to the subsequent layer, and this process continues until the final layer is reached.

To train the DNN model, a backpropagation algorithm with stochastic gradient descent optimization is used. The model parameters are updated iteratively to minimize a defined loss function, such as categorical cross-entropy, which measures the discrepancy between the predicted probabilities and the true labels. DNNs are capable of learning complex features and patterns with in the data, which can be helpful in identifying cyber-attacks and anomalies in network traffic. It can automatically extract hierarchical representations of data, allowing them to effectively capture non-linear relationships and dependencies between input features. DNNs can learn and derive hierarchical representations of data features, enabling them to capture complex relationships between input features and output labels. The DNN model is applied on the training set and subsequently evaluated.

Based on the UNSW-NB15 and NSL KDD datasets, a DNN model is trained using the Keras sequential API, The DNN model comprises three dense layers, each consisting of 64, 32, and 10 units, respectively. Each hidden layer is activated using a ReLU activation function, which has 20 input dimensions. Multi-class classification is achieved using the softmax activation function in the output layer. Nevertheless, the DNN architecture used for the NSL KDD dataset is similar but with modified layers comprising 64, 32, and 5 dense layers.

A DNN involves progressively learning complex representations of data through multiple hidden layers, leveraging non-linear activation functions and optimizing the weights using training data. This enables DNNs to handle more sophisticated tasks and achieve higher levels of accuracy and performance compared to traditional shallow neural networks. A DNN model is constructed using the categorical_crossentropy loss function, using the Adam optimizer with 0.001 learning rate, and evaluating their performance by accuracy metric. Overfitting is prevented by using an early stop callback.

The DNN model trained on UNSW NB 15 and NSL KDD datasets consist of a total of 3,754 and trainable parameters. With a batch size of 32, the DNN model is trained for 50 epochs or early as mentioned above. As the model is being trained, it was monitored for performance on both the training and validation datasets.

In the UNSW NB 15 dataset, the training process is halted prematurely at the 38th epoch in accordance with the early stopping callback function. This is due to the model's performance on the validation dataset not improving for a certain number of epochs (in this case, 5 epochs) based on the validation loss. After the training process on the UNSW NB 15 dataset, the DNN model training accuracy is approximately 83.25% and a validation accuracy of approximately 83.07%. These results indicate that the model successfully captured patterns in the dataset and exhibited good generalization to unseen data and did not have overfitting problems. In contrast, when evaluated on the NSL KDD dataset, an accuracy of approximately 99.67% and a validation accuracy of approximately 99.62% after completing the training process is achieved with DNN model. These high accuracy values suggest that the model effectively learned the underlying patterns in the NSL KDD dataset.

The loss values for the training and validation datasets of UNSW NB 15 are calculated to be 0.4414 and 0.4456, respectively. Similarly, for the NSL KDD dataset, the corresponding loss values for the training and validation sets are 0.0109 and 0.0162, respectively. These loss values represent the average discrepancy between the predicted and actual values, with lower values indicating better performance of the model in minimizing the prediction errors.

3.4.6 EXtreme Gradient Boosting

EXtreme Gradient Boosting (XGBoost) [113] is a powerful and efficient execution of the gradient boosting algorithm. XGBoost is particularly well-suited for tabular data and is often used due to its high performance and scalability. It involves iteratively adding weak models to the ensemble, with each consecutive model attempting to correct the mistakes made by the previous models. By learning from the residuals of the previous models, XGBoost gradually improves the overall predictive capability of the ensemble.XGBoost utilizes decision trees as base learners. Decision trees are simple yet effective models that make predictions based on a series of hierarchical if-else conditions. XGBoost combines multiple DT to form an ensemble, where each tree contributes to the final prediction based on its individual strength.

The XGBoost algorithm optimizes a specific objective function by considering both the model's accuracy and its complexity. This objective function incorporates a loss function that measures the deviation between predicted and actual labels, as well as regularization terms that control the complexity of the model. By finding the optimal balance between accuracy and complexity, XGBoost delivers robust and generalizable classification models. XGBoost is an ML algorithm that utilizes a gradient boosting framework for classification tasks. It is known for high accuracy and ability to handle complex datasets. In this thesis XGBoost is trained on two distinct datasets: NSL KDD and UNSW NB 15.

The architecture of the XGBoost model involves ensembling of decision trees. This technique involves sequentially training each tree in an effort to correct the mistakes made by the previous trees. This iterative process aims to optimize the model's performance by minimizing the loss function. XGBoost incorporates several hyperparameters that can be tuned to enhance its predictive capabilities, such as the number of trees, tree depth, learning rate, and regularization parameters. The XGBoost model accuracy is 85.14% when applied to the UNSW NB 15 dataset. Although slightly lower than the accuracy on the NSL KDD dataset, this result still demonstrates the model's effectiveness in classifying network connections in a different dataset. The performance may vary due to the differences in the data characteristics, such as the distribution of features and the complexity of the underlying patterns.

The XGBoost model is 99.71% accurate when applied to the NSL KDD dataset. This indicates that the model performed exceptionally well in accurately classifying the network connections in this dataset. The high accuracy can be attributed to XGBoost ability to capture intricate patterns and relationships in the data, thereby enabling accurate predictions. The results are obtained by individually training the XGBoost model on both datasets.

3.4.7 Gated Recurrent Unit

The GRU [114] is a type of recurrent neural network designed to process sequential data by preserving information from the previous time step. The GRU model is trained on the train set to learn patterns in the input data and forecast the corresponding target variable. The accuracy metric is used to assess the GRU model's performance on the train set.

3.4.8 Multi-Layer Perceptron

MLP [115] is an artificial feedforward neural network. It comprises multiple node layers, each node is a basic processing unit that receives one or more inputs, applies a weighted sum function, and passes the result through a non-linear activation function to produce an output. After applying the MLP model, the results are evaluated based on the accuracy metric. Accuracy measures the proportion of instances correctly classified in the test set.

3.5 Performance Criteria

In the intricate landscape of machine learning, deciphering the performance of classification models is akin to navigating through a labyrinth of insights. Imagine it as the compass guiding us through the dense terrain of algorithms. Precision, recall, accuracy, and the F1 score act as our North Star, illuminating the path toward a model's prowess. It's not just about metrics; it's about sculpting models that stand resilient in the face of real-world

challenges. Let's embark on this analytical journey to refine, optimize, and ensure our models are not just algorithms but trusted allies in the realm of intelligent decision-making.

3.5.1 Confusion matrix

The Confusion Matrix is an evaluation matrix for classifier performance. The matrix is commonly used to express a classification model's competence on a UNSW NB 15 and NSL KDD test dataset where actual values are known. The confusion matrix gives False Negative (FN), True Positive (TP), False Positive (FP), and True Negative (TN) for the classes. Following are the definitions for TP, FP, FN, and TN for a Ci class. where

- TP(C_i) = Instances that belong to class C_i and that are correctly classified as C_i.
- FP (C_i) = Instances that do not belong to class C_i but are classified as C_i
- FN (C_i) = Instances that belong to class C_i and that are misclassified as non-C_i.
- TN (C_i) = Instances that do not belong to class C_i and are classified as not belonging to class C_i.

3.5.2 Accuracy

The metric of accuracy is employed to assess the effectiveness of a classification model applied on UNSW NB 15 and NSL KDD. The accuracy of a model is only a single facet of its comprehensive performance. Eq. (1) depicts the measurement of single class accuracy.

$$Accuracy = \frac{TP + TN}{FP + TP + TN + FN} \tag{3.1}$$

3.5.3 Precision

Precision is defined as a positive predictive value. It measures the proportion of genuine positive classifications within all positive classifications. In other words, it indicates how many

of the instances that the model classified as positive are actually positive in UNSW NB 15 and NSL KDD. The following Eq. (2) gives the precision value for a single class:

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

3.5.4 Recall

Recall is the percentage of individuals in a population associated with one or more anomalies in the system. True positive rate is also known as Recall, which is how many times out of every 100 individuals determined initially to have a problem did not have one. The recall value for a single category can be found using Eq., which has the

$$Recall = \frac{TP}{TP+FN} \tag{3.3}$$

3.5.5 F1 Score

It is important to note that the F1 score measures both precision and recall, and provides an overall summary of the algorithm's performance. It is particularly useful when one class has significantly more instances than the other class. It may not be possible to use accuracy as a reliable metric in these circumstances, as the algorithm may simply predict the majority class for all instances. As a result of Eq. (4), the F1 Score value for a single class can be determined.

$$F1\ Score = \frac{2 * TP}{2 * TP + FP + FN} \tag{3.4}$$

3.5.6 ROC curve

For multiclass classification, a one vs all approach is used to calculate the ROC curve and AUC metric. The positive classes are treated individually, while the negative classes are grouped together. After calculating the area under the ROC curve for each class, the true positive rate (TPR) and false positive rate (FPR) for each threshold are plotted against the ROC curve.

CHAPTER 4

EXPERIMENTAL RESULTS AND ANALYSIS

4.1 Overview

The results of the proposed algorithm X-DNN for identification of Cyber-attacks on IoT are presented and discussed in this chapter. All the algorithms are trained and tested using Google Colab.

This thesis evaluates several ML approaches, including DT, LR, random forest, SVM, and ensemble models based on DNN. The performance of these approaches varied depending on the type of cyber-attack. For example, decision trees performed well for DOS attacks, while RF performed well for probing attacks. Our ensemble model X-DNN performed well across a range of cyber-attack types

The results also indicate that ML techniques are highly effective in identifying new cyber-attacks on IoT. Also explored the use of ensemble models that integrate different ML and DL techniques for identifying cyber-attacks. X-DNN outperforms the other approaches, achieving high accuracy. This suggests that combining different ML and DL techniques in an ensemble way is highly effective approach for identifying cyber-attacks.

4.2 Performance Comparison of Machine Learning Models

ML models stand at the forefront of this endeavor, providing the computational power to discern patterns and anomalies indicative of such security breaches. This section delves into a comparative analysis of several ML models to discern their efficacy in the classification of cyber-attacks on IoT devices. To ensure a comprehensive evaluation, we employed a range of

models, each with its theoretical strengths and practical applications. The models under scrutiny include LR, known for its simplicity and interpretability; DT, prized for their decision-making transparency; MLP, which leverage the power of neural networks; RF, which combines multiple decision trees to improve predictive accuracy; and XGBoost, recognized for its efficiency and performance in structured datasets.

The comparison is based on a set of established performance metrics: Accuracy, Precision, Recall, and the F1-Score. These metrics collectively offer a multifaceted view of model performance, each providing unique insights into the models' predictive capabilities. Accuracy measures the overall correctness of the model, Precision assesses the model's exactness in predicting positive instances, Recall evaluates the model's completeness in identifying all positive cases, and the F1-Score harmonizes Precision and Recall, offering a single measure of the model's balanced performance.

The models were tested on two distinct datasets: the NSL KDD dataset, a benchmark in the field of network intrusion detection, and the UNSW NB 15 dataset, which represents a more contemporary and nuanced set of network interactions. The ensuing analysis offers a rigorous comparison of the models' performances, revealing their strengths and weaknesses in detecting cyber-attacks within different data environments.

4.2.1 Comparison on NSL KDD Dataset

The NSL KDD dataset, a benchmark dataset in network intrusion detection, was first employed to gauge the performance of our chosen ML models. LR demonstrated commendable accuracy at 84%, with precision, recall, and F1-score metrics following suit, indicating reliable predictive power. DT offered slightly higher accuracy at 86%, suggesting good fit for the data, but with a marginally lower recall. MLP showed improved accuracy at 89%, though the F1-score indicated a need for balance between precision and recall. RF achieved an accuracy of 87%, with consistent performance across all metrics, reflecting its robustness through ensemble learning. However, it was the XGBoost model that delivered a standout performance with an accuracy nearing perfection at 99.78%, and equally high precision and recall scores, solidifying

its dominance in handling the NSL KDD dataset. This comparative performance analysis of the models is visually summarized in Figure 4.1, which provides a clear graphical representation of the accuracy levels achieved by each ML model on the NSL KDD dataset.

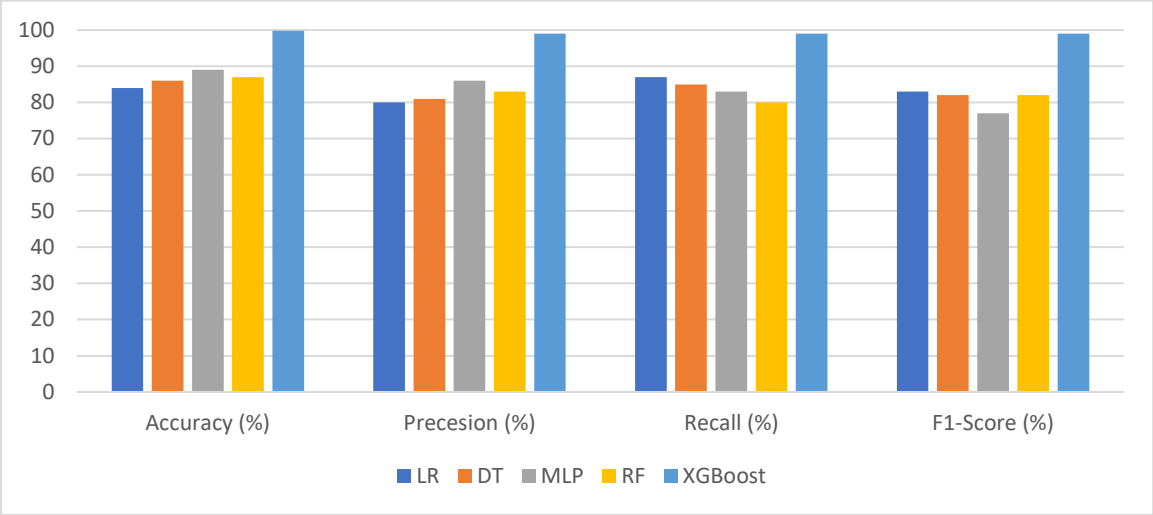

Figure 4.1: Comparison of ML models on NSL KDD dataset

4.2.2 Comparison on UNSW NB 15 Dataset

Turning our attention to the UNSW NB 15 dataset, designed to represent modern network traffic and attacks, the models were subjected to a more challenging classification task. Here, LR accuracy dipped to 73.92%, underscoring the model's limitations in adapting to more complex data structures.

DT and MLP shared an identical accuracy of 80.95%, with DT showing superior recall, indicative of its ability to capture a higher rate of true positive attacks. RF presented a modest increase in accuracy at 81.8%, reflecting the strength of the ensemble approach in a more diverse dataset. Yet again, XGBoost prevailed with the highest accuracy of 82.81%, demonstrating a consistent ability to adapt and maintain performance even in the face of more intricate and varied data. This comparative performance analysis of the ML 2wmodels is visually summarized in Figure 4.2, which provides a clear graphical representation of the accuracy levels achieved by each ML model on the UNSW NB 15 dataset.

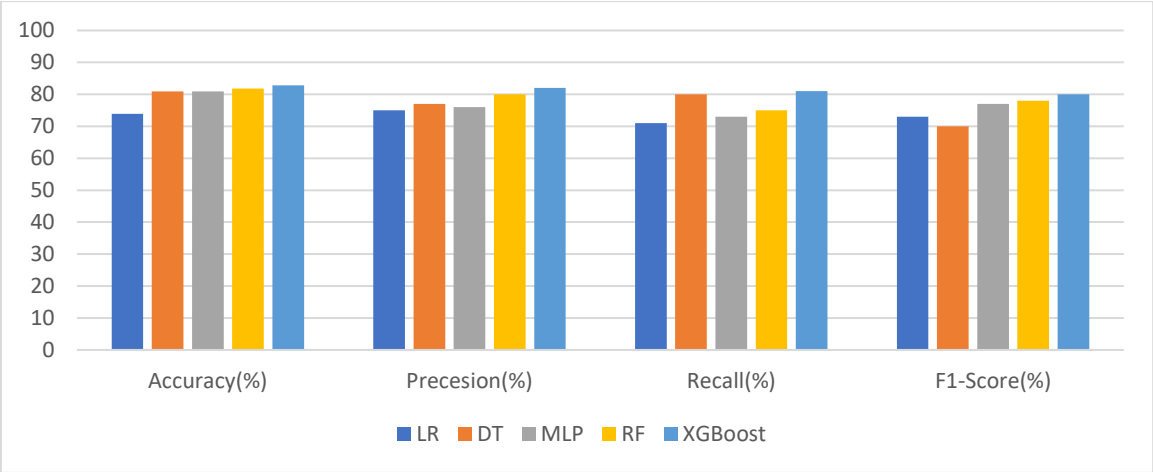

Figure 4.2: Comparison of ML models on UNSW NB 15 dataset

The comparative analysis revealed a clear stratification among the ML models when applied to cybersecurity threat classification in IoT environments. The XGBoost algorithm's superior performance across both datasets is noteworthy, suggesting that its sophisticated boosting and tree-pruning techniques are well-suited for the complex and high-dimensional nature of network traffic data.

Table 4.1: Performance Comparison of Machine Learning Methods

Model Name	UNSW NB 15	NSL KDD
LR	73.92%	83.73%
DT	80.95%	85.95%
MLP	80.95%	89.06%
XGBoost	81.30%	99.78%
RF	82.81%	87.37%

The detection accuracy of the XGBoost algorithm, along with other selected ML algorithms, was evaluated on the NSL-KDD and UNSW NB 15 datasets. These comparative accuracy results are detailed in Table 4.1. The consistency of XGBoost high precision and recall suggests it is a robust choice for minimizing both false positives and false negatives a critical consideration in cybersecurity where the cost of either can be high. In contrast, simpler models like LR struggled with the more complex UNSW NB 15 dataset, indicating that while such models may offer speed and interpretability, they may not always be suitable for the nuanced

requirements of modern intrusion detection. The performance of MLP and DT on both datasets suggests these models can offer a middle ground, balancing complexity and interpretability. However, the ensemble approaches, as seen in RF and particularly XGBoost, appear to provide a more adaptable framework for this domain.

4.3 Performance Analysis of X-DNN

Accuracy, a widely employed metric, measures the overall correctness of predictions during model training. For instance, the accuracy of UNSW NB 15 is 85.3%, while NSL KDD exhibits a notably high accuracy of 99.7%. However, when faced with imbalanced datasets, relying solely on accuracy may not provide a complete picture.

The detection accuracy of the X-DNN algorithm, along with other selected DL algorithms, was evaluated on the NSL-KDD and UNSW NB 15 datasets. These comparative accuracy results are detailed in Table 4.2. The results, presented in Table 4.2, show the performance of the deep belief network (DBN)[118], stacked non-symmetric deep auto-encoder (S-NDAE) [118], differential evolution and extreme learning machine (DE-ELM) [119], HC-DTTSVM [103], and X-DNN algorithms. The analysis reveals that the X-DNN algorithm achieves a detection accuracy of 99.79% for normal samples, which is comparable to the performance of HC-DTTSVM 96.69%, and higher than that of DBN 95.64%, S-NDAE 97.73% and DE-ELM 96.80% algorithms. In terms of detecting Probe samples, X-DNN demonstrate a detection accuracy of 84.35 %, outperforming HC-DTTSVM 78.75%, DBN 72.97% and DE-ELM 62.74%, but slightly trailing behind S-NDAE 94.67%. For DoS samples, the X-DNN algorithm achieves a detection accuracy 97.78%, which is higher than HC-DTTSVM 94.20%, DBN 87.96% and DE-ELM 91.50%, and S-NDAE 94.58%.

Notably, the X-DNN algorithm exhibits promising results in detecting U2R and R2L samples, achieving detection accuracies of 26.82% and 36.60% respectively. Although these accuracies are significantly higher than those obtained by DBN and DE-ELM (0.00% for both U2R and R2L), they are still lower than S-NDAE (2.70% for U2R and 3.82% for R2L). The XDNN algorithm, proposed in this study, demonstrates exceptional performance across all

categories. It achieves a detection accuracy of 99.79% for Normal samples, outperforming all other algorithms. For Probe samples, XDNN achieves an accuracy of 84.35%, which is higher than DBN, DE-ELM, HC-DTTSVM, and S-NDAE. The XDNN algorithm also showcases remarkable accuracy for DoS samples 97.78%. The detection accuracy of different intrusion detection algorithms on the UNSW NB 15 dataset is presented in Table 4.3. The comparison includes the DT, LR, SVM, HC-DTTSVM algorithm, and the proposed X-DNN model.

Table 4.2: Comparative Analysis of Class-Wise Accuracy on NSL-KDD Dataset

Category	**DBN** [118]	**S-NDAE** [118]	**DE-ELM** [119]	**HC-DTTSVM [103]**	**X-DNN**
Normal	95.64%	97.73%	96.80%	96.69%	99.79%
Probe	72.97%	94.67%	62.74%	78.75%	84.35%
DoS	87.96%	94.58%	91.50%	94.20%	97.78%
U2R	0.00%	2.70%	0.00%	24.32%	26.82%
R2L	0.00%	3.82%	11.90%	21.60%	36.60%

Table 4.3: Comparative Analysis of Class-Wise Accuracy on UNSW-NB15 dataset

Category	**DT**	**LR**	**SVM**	**HC-DTTSVM**	**X-DNN**
Normal	74.93%	64.54%	75.60%	88.84%	77.38%
Reconnaissance	80.77%	49.57%	78.30%	52.83%	67.03%
Backdoor	4.97%	22.47%	3.30%	8.23%	41.28%
DoS	8.83%	0.00%	4.60%	64.25%	64.53%
Exploits	90.08%	24.97%	92.70%	80.17%	83.27%
Analysis	0.00%	0.00%	0.00%	12.56%	98.54%
Fuzzers	55.24%	36.28%	47.10%	29.23%	99.23%
Worms	72.72%	38.64%	9.10%	25.00%	62.57%
Shellcode	60.84%	1.32%	53.20%	38.62%	99.79%
Generic	96.96%	96.29%	95.80%	98.18%	71.24%

Among the algorithms evaluated, the X-DNN model achieves the highest accuracy in several categories. For Normal samples, X-DNN achieves an accuracy of 77.38%, surpassing the performance of Decision Tree 74.93%, LR 64.54%, SVM 75.60%, and HC-DTTSVM

88.84% algorithms. In the Reconnaissance category, X-DNN obtains an accuracy of 67.03%, outperforming Decision Tree 80.77%, LR 49.57%, SVM 78.30%, and HC-DTTSVM 52.83%. Remarkably, the X-DNN model showcases significant improvements in the Backdoor category, achieving an accuracy of 41.28%, compared to Decision Tree 4.97%, LR 22.47%, SVM 3.30%, and HC-DTTSVM 8.23%.

For DoS samples, X-DNN achieves an accuracy of 64.53%, surpassing Decision Tree 8.83%, LR 0.00%, SVM 4.60%, and HC-DTTSVM 64.25%. In the Exploits category, X-DNN achieves an accuracy of 83.27%, outperforming Decision Tree 90.08%, LR 24.97%, SVM 92.70%, and HC-DTTSVM 80.17%. Notably, the X-DNN model demonstrates exceptional accuracy in the Analysis category with 87.52%, surpassing the performance of Decision Tree 0.00%, LR 0.00%, SVM 0.00%, and HC-DTTSVM 12.56%.

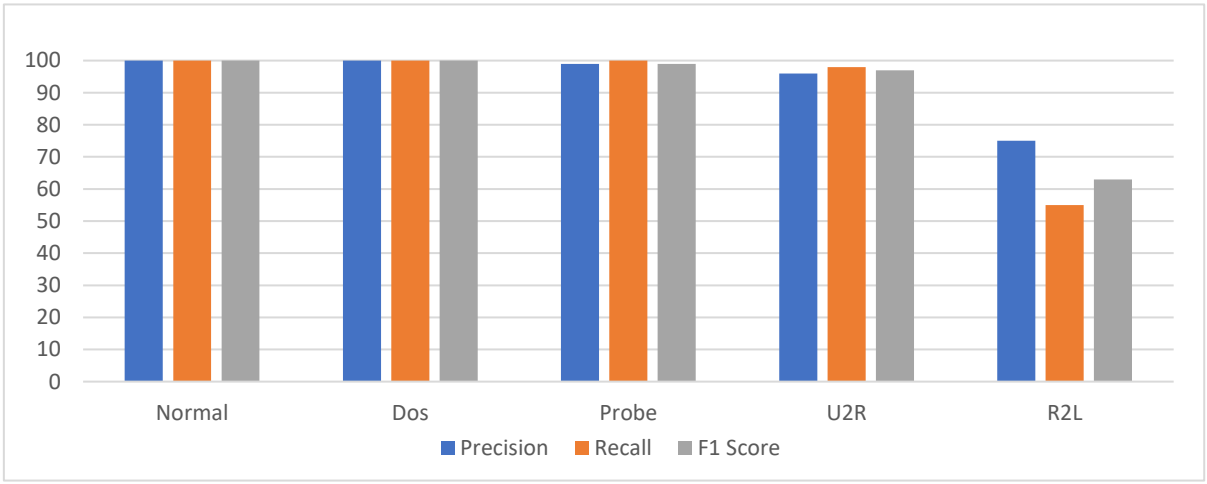

Figure 4.3: X-DNN class-wise Classification Performance on NSL KDD Dataset

For Fuzzers and Shellcode samples, X-DNN achieves superior accuracy compared to other algorithms, obtaining 99.23% and 99.79% respectively. In contrast, DT, LR, SVM, and HC-DTTSVM algorithms show relatively lower accuracies in these categories. For the Generic category, X-DNN achieves an accuracy of 71.24%, comparable to Decision Tree 96.96%, LR 96.29%, and SVM 95.80%, but slightly lower than HC-DTTSVM 98.18%. For comparison, several individual models as discussed above and compared with the proposed method. The results are shown in Figure 4.3 and 4.4 for NSL KDD and UNSW NB 15. The GRU model achieved an accuracy of 73.95%, LR achieved 73.92% accuracy, DT model achieved 80.9%

accuracy, and the MLP model achieved 80.95% accuracy, Random Forest achieved accuracy of 81.8% and, the HC-DTTSVM model achieved 81.21% accuracy, and CNN achieved 81.6% accuracy on the test set. All the accuracies of these models are achieved on the test set of UNSW NB 15 dataset.

Precision becomes essential when the accuracy of positive predictions is of paramount importance. For UNSW NB 15, precision values vary across classes, emphasizing the need to scrutinize specific class performances. The precision for class 0 (Normal) is 0.77, indicating that 77% of instances predicted as class 0 were true positives. Conversely, NSL KDD demonstrates precision values of 1 for classes 0 (Normal) and 1 (Probe), reflecting high accuracy in positive predictions. Precision becomes essential when the accuracy of positive predictions is of paramount importance. For UNSW NB 15, precision values vary across classes, emphasizing the need to scrutinize specific class performances. The precision for class 0 (Normal) is 0.77, indicating that 77% of instances predicted as class 0 were true positives. Conversely, NSL KDD demonstrates precision values of 1 for classes 0 (Normal) and 1 (Probe), reflecting high accuracy in positive predictions.

Table 4.4: Classification Performance of X-DNN Model on NSL KDD Dataset Classes

Class	Attack type	precision	recall	f1-score
0	Normal	1.00	1.00	1.00
1	Probe	1.00	1.00	1.00
2	DoS	0.99	1.00	0.99
3	R2L	0.96	0.98	0.97
4	U2R	0.75	0.55	0.63

Similarly, in Table 4.4, the precision of NSL KDD is shown and for class 0 (Normal) & 1 (Probe) is 1.00, indicating that the model's predictions for this class have high accuracy. The weighted average precision is 1, which means that on average the model’s positive predictions were correct 100% of the time. Precision for the NSL KDD and UNSW NB 15 datasets are depicted in Figure 4.5 and 4.6 respectively. In situations where false positives (FP) carry significant consequences, precision values are carefully examined. Precision, calculated as TP / (TP + FP), is a key indicator of the model's positive prediction accuracy. Meanwhile, recall, also known as sensitivity, gauges the model's ability to identify all relevant instances of a class.

The recall values for UNSW NB 15 range from 0.00 to 1.00, with an overall weighted average recall of 0.85. It's observed that certain classes exhibit lower recall values, suggesting challenges in correctly identifying instances. However, classes 0 (normal), 1 (reconnaissance), 2 (Backdoor), 4 (exploits), 8 (shellcode), and 9 (Generic) have relatively low recall values, suggesting that the model struggles to correctly identify some instances of these classes. For the NSL KDD dataset, Class 4 demonstrates a relatively low recall value. However, overall, the recall value for the NSL KDD dataset is exceptionally high, reaching a value of 1. Recall for the NSL KDD and UNSW NB 15 datasets are depicted in Figure 4.3 and 4.4 respectively.

Table 4.5: Classification Performance of X-DNN Model on UNSW NB15 Dataset Classes

Class	Attack Type	precision	recall	f1-score
0	Normal	0.77	0.18	0.29
1	Reconnaissance	0.67	0.01	0.02
2	Backdoor	0.41	0.03	0.06
3	DoS	0.60	0.88	0.71
4	Exploits	0.83	0.81	0.82
5	Analysis	0.98	0.97	0.97
6	Fuzzers	1.00	1.00	1.00
7	Worms	0.61	0.69	0.65
8	Shellcode	1.00	0.00	0.01
9	Generic	0.70	0.27	0.39

Incorporating both precision and recall, the F1-Score emerges as a comprehensive metric, particularly useful in scenarios with imbalanced class distributions. Calculated as the harmonic mean of precision and recall, the F1-Score offers a balanced assessment of a model's ability to handle both false positives and false negatives. Its application becomes imperative for a nuanced understanding of a classification model's overall performance in real-world scenarios. The F1-scores for other classes of UNSW NB 15 dataset range from 0.00 to 0.97, which indicates the level of accuracy of the model for each class. The weighted average F1-score of 0.83 suggests that the overall accuracy of the model is decent. In contrast, the NSL KDD dataset, the weighted average F1-score of 1.00 indicates that the model achieves a commendable overall accuracy. F1-Score for the NSL KDD and UNSW NB 15 datasets are depicted in Figure 4.3 and 4.4.

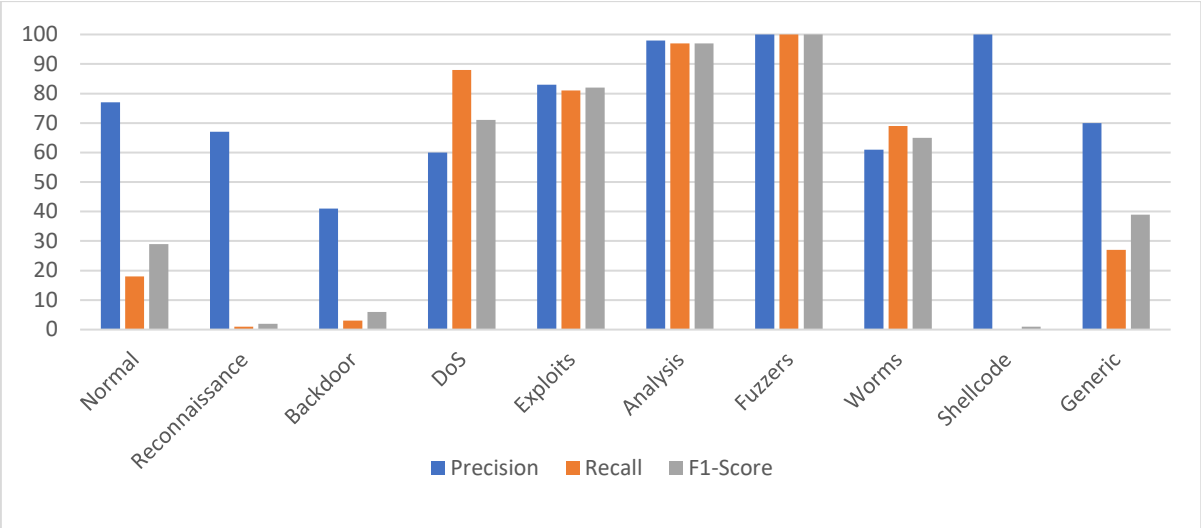

Figure 4.4: X-DNN class-wise Classification Performance on UNSW NB 15 Dataset

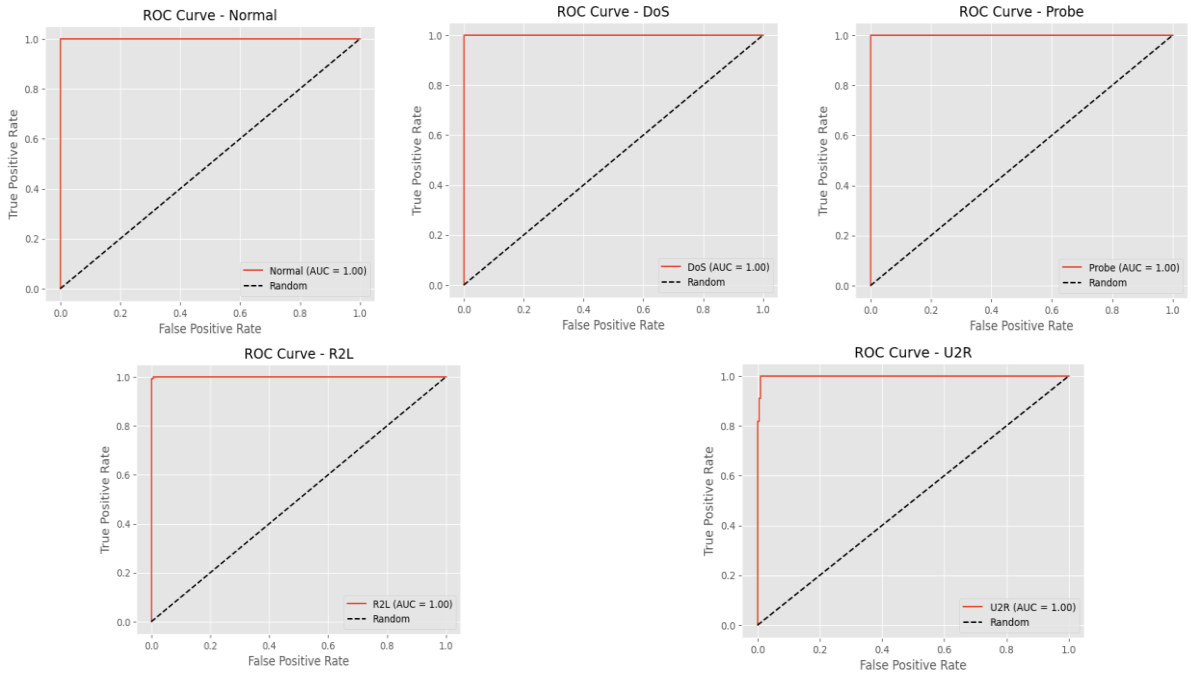

Figure 4.5: ROC Curves for NSL KDD dataset Classes using X-DNN

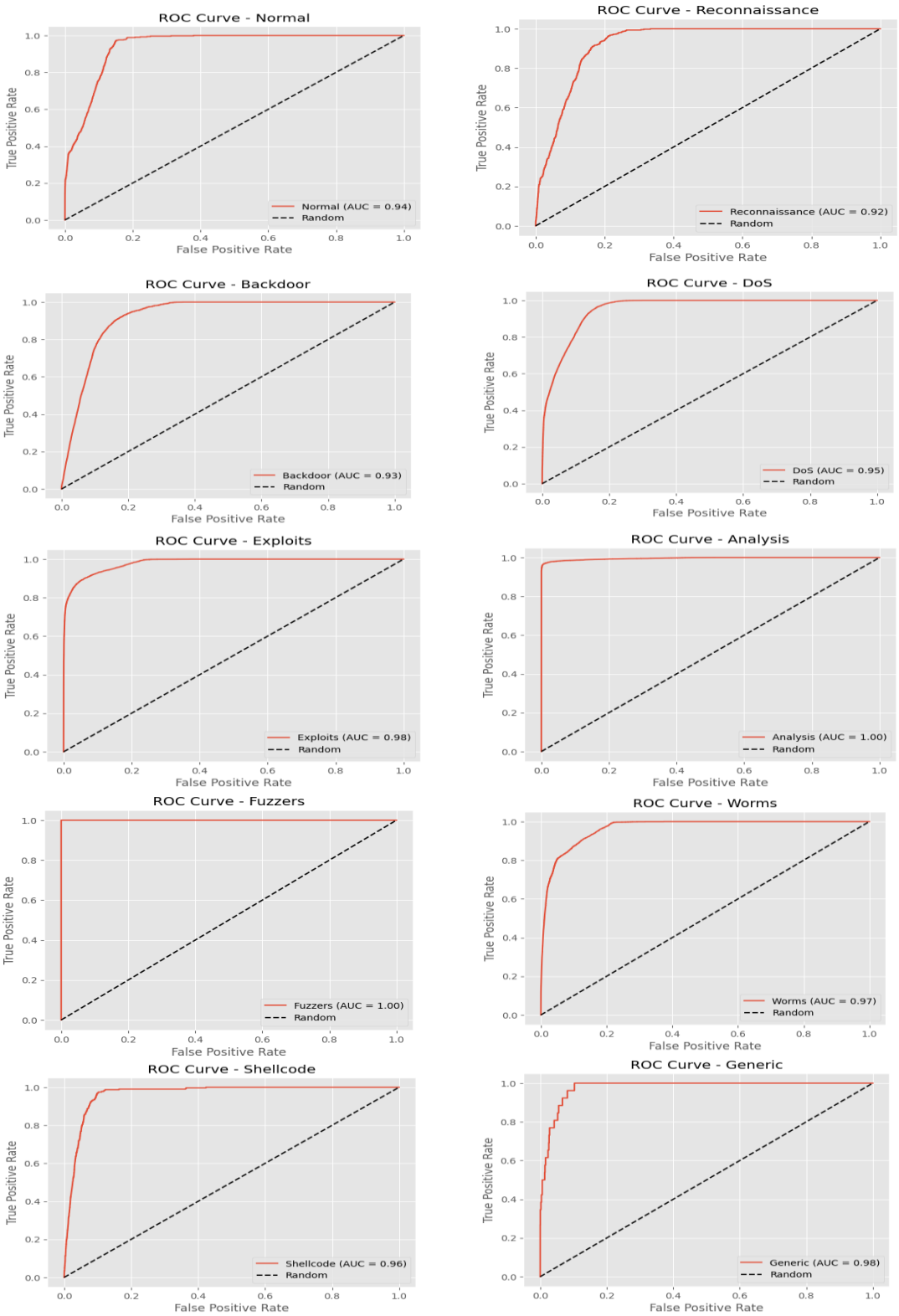

Figure 4.6: ROC Curves for UNSW NB 15 dataset Classes using X-DNN

ROC (Receiver Operating Characteristic) and AUC (Area Under the Curve) are assessment metrics. AUC, on the other hand, represents the area under the ROC curve. AUC is a scalar value between 0 and 1, where 0 represents a classifier that predicts all samples wrong, and 1 represents a perfect classifier that predicts all samples correctly. Table. 4.6 and 4.7 shows that the proposed model X-DNN is able to perfectly distinguish between different classes. Fig.4.5 and 4.6 provide the ROC curve subsequently for each specific attack class of the UNSW NB 15 and NSL KDD datasets.

Table 4.6: AUC Metrics for Classes in NSL KDD Dataset

Attack type	**AUC Value**
Normal	1.00
Probe	1.00
DoS	1.00
R2L	1.00
U2R	1.00

Table 4.7: AUC Metrics for Classes in UNSW NB15 Dataset

Attack Type	**AUC Value**
Normal	0.94
Reconnaissance	0.92
Backdoor	0.93
DoS	0.95
Exploits	0.98
Analysis	1.00
Fuzzers	1.00
Worms	0.97
Shellcode	0.96
Generic	0.98

The confusion matrix demonstrates the model's competency by displaying the number of true positive, true negative, false positive, and false negative predictions. As a measure of the model's performance on the test dataset, confusion matrices are computed for each class. In the confusion matrix, each class is represented by the number of instances that are correctly and incorrectly classified.

Normal	95	0	11	331	67	28	0	6	0	0
Reconnaissance	0	4	12	332	77	23	0	23	0	0
Backdoor	1	0	103	2816	132	58	0	183	0	0
DoS	20	1	90	7800	301	90	0	549	0	2
Exploits	4	0	9	604	3843	56	0	226	0	0
Analysis	1	0	2	279	71	11441	0	54	0	1
Fuzzers	0	0	0	0	0	0	18675	0	0	0
Worms	2	1	26	712	110	8	0	1911	0	0
Shellcode	0	0	0	105	22	0	0	190	1	0
Generic	0	0	0	14	0	0	0	5	0	7
	Normal	**Reconnaissance**	**Backdoor**	**DoS**	**Exploits**	**Analysis**	**Fuzzers**	**Worms**	**Shellcode**	**Generic**

Figure 4.7: X-DNN Model Evaluation with Confusion Matrix for UNSW NB 15 Dataset

Normal	9215	7	2	0	0
Probe	6	13361	13	4	2
DoS	4	3	2363	3	0
R2L	2	2	1	196	0
U2R	1	1	1	2	6
	Normal	**Probe**	**DoS**	**R2L**	**U2R**

Figure 4.8: X-DNN Model Evaluation with Confusion Matrix for NSL KDD Dataset

From Fig 4.7 and 4.8, it can be seen that the model performed well for some classes, such as class Fuzzers and Probe, which is correctly classified for all instances. However, the

model performed poorly for some other classes, such as class DOS, which has a high number of misclassified instances on both datasets.

4.4 Comparison of Proposed Method with DL Models

The advent of DL in the field of cybersecurity has provided sophisticated tools capable of detecting and classifying intricate cyber threats in IoT environments. In this section, we present a comparative analysis between established DL models and our proposed ensemble model, termed X-DNN, which integrates the strengths of XGBoost and DNN.

To benchmark the efficacy of these models, we utilized a set of performance metrics Accuracy, Precision, Recall, and F1-Score across two key datasets in cybersecurity: NSL KDD and UNSW NB 15. Additionally, for our proposed X-DNN model, we examined detailed performance indicators such as the confusion matrix, ROC curves, and AUC values for each class, enabling a granular analysis of its classification prowess.

4.4.1 Comparison on NSL KDD Dataset

The NSL KDD dataset results demonstrate that the traditional DL models, such as GRU and HC-DTTSVM, show respectable performance with accuracies of 83% and 86% respectively. However, they fall short when compared to the standalone XGBoost and DNN models, which showcase near-perfect accuracy, precision, and recall metrics. Our proposed X-DNN model surpasses these figures marginally, achieving an accuracy of 99.81%. While there is a slight decrease in precision and recall to 94% and 90% respectively, the F1-Score of 92% indicates a strong balance between precision and recall, crucial for reliable cyber threat detection. This comparative performance analysis of the models is visually summarized in Figure 4.9, which provides a clear graphical representation of the accuracy levels achieved by each DL models on the NSL KDD dataset.

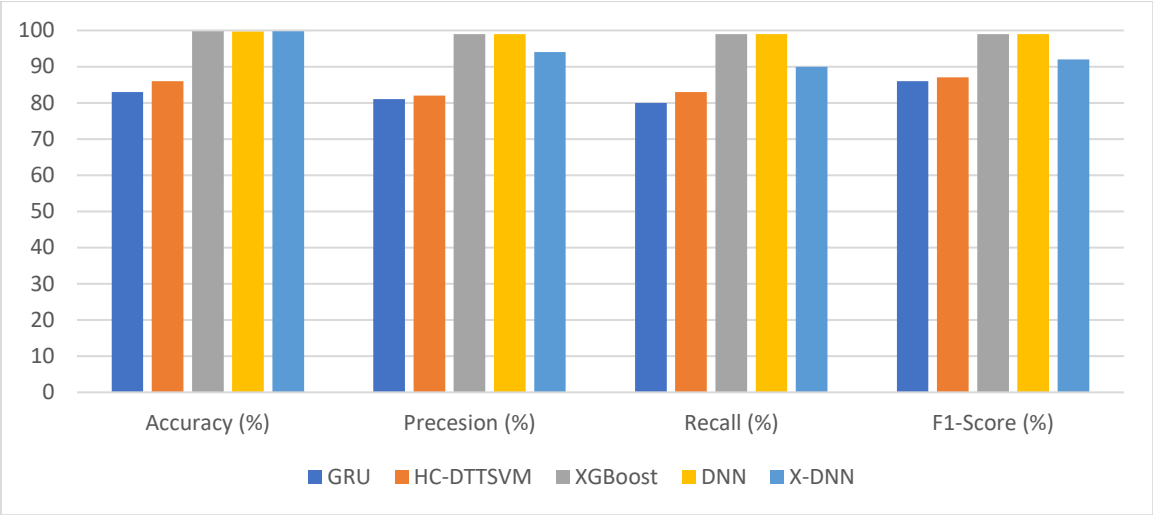

Figure 4.9: Comparison of DL models on NSL KDD dataset

4.4.2 Comparison on UNSW NB 15 Dataset

When applied to the more challenging UNSW NB 15 dataset, the traditional DL models maintain a consistent performance, with GRU achieving an accuracy of 73.95%. The HC-DTTSVM model shows a significant improvement with an accuracy of 81.3%. However, our proposed X-DNN model demonstrates superior performance, with an accuracy of 85.36%, precision at 85%, and recall also at 85%.

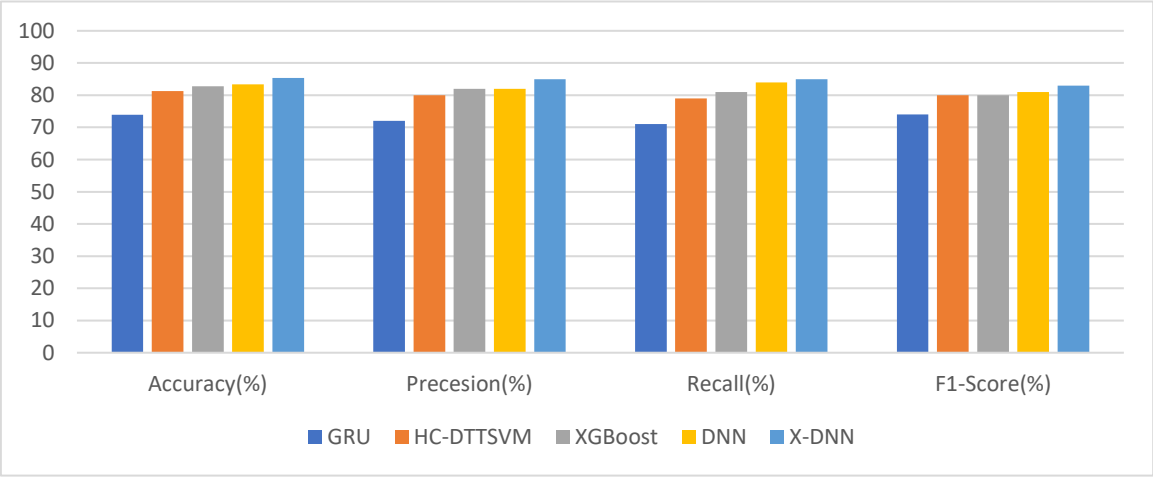

Figure 4.10: Comparison of DL models on UNSW NB 15 dataset

The F1-Score of 83% is noteworthy, as it suggests that X-DNN maintains high performance in precision and recall even when dealing with more complex and nuanced data types found in modern network traffic. This comparative performance analysis of the models is visually summarized in Figure 4.10, which provides a clear graphical representation of the accuracy levels achieved by each DL model on the UNSW NB 15 dataset. Our proposed X-DNN model's performance warrants a detailed examination beyond conventional metrics. The confusion matrix for each class will reveal the model's specific strengths in identifying each type of attack, while the ROC curves and AUC values will provide insights into the model's true positive rate against the false positive rate at various threshold settings. This analysis will shed light on the model's robustness and reliability in operational settings.

Table 4.8: Performance Comparison of the Proposed Method with Deep Learning Methods

Model Name	UNSW NB 15	NSL KDD
GRU	73.95%	83.05%
HC-DTTSVM	81.21%	85.95%
XGBoost	81.30%	99.78%
DNN	83.31%	99.71%
X-DNN (Proposed)	85.36%	99.81%

4.5 Discussion

The experimental results show that the X-DNN ensemble model represents an efficient approach for detecting cyber-attacks in the IoT environments. This ensemble model exhibited superior performance compared to both the individual DNN and XGBoost models, not only in terms of accuracy but also in various evaluation metrics. These results suggest that combining the strengths of DNN and XGBoost allows for a more comprehensive capture of the intricate patterns inherent in the data.

Based on the assessment, the ensemble model appears to perform well, although there is room for improvement, particularly for classes with low accuracy. The ROC curve analysis further shows the ensemble model's efficiency, as it exhibits a notably high true positive rate across all the attack classes. This signifies its ability in accurately identifying each specific type

of attack. These outcomes serve as evidence for the working of ensemble learning techniques in the scenario of cyber-security applications. Moreover, they provide valuable insights into the performance characteristics of diverse ML approaches employed for the identification of cyber-attacks in the IoT environments.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Thesis presents effectiveness of machine learning techniques, specifically the ensemble of XGBoost and DNN, in identifying cyber-attacks on IoT, achieving accuracy of 85.36% and 99.81% in the UNSW NB 15 and NSL KDD datasets, respectively. These results demonstrated that X-DNN exhibits exceptional performance in classifying cyber-attacks. The selection of the UNSW NB 15 and NSL KDD datasets is based on comprehensive coverage of cyber-attack instances in IoT environments. While these datasets provided valuable insights and served as benchmarking tools. The proposed method utilizes an ensemble approach, which is stemmed from the complementary strengths of XGBoost and DNN algorithms. This combination resulted in high accuracy in identifying diverse cyber threats within IoT environments.

In the comparative analysis, X-DNN's performance was evaluated against other selected DL algorithms, employing the NSL-KDD and UNSW NB 15 datasets. The analysis revealed that for normal samples, X-DNN achieved a detection accuracy of 99.79%, surpassing HC-DTTSVM and other algorithms. Furthermore, in detecting Probe and DoS samples, X-DNN demonstrated superior accuracy, significantly outperforming the compared algorithms. This highlights X-DNN effectiveness in differentiating between various types of cyber threats. One of the remarkable aspects of X-DNN's performance is its capability to detect U2R and R2L samples, where it achieved higher accuracies than other sophisticated models. The class-wise comparative analysis further emphasizes X-DNN efficacy. On the NSL KDD dataset, X-DNN outperformed other algorithms in almost all categories, demonstrating particularly strong performance in normal, probe, and DoS categories. Similarly, on the UNSW NB 15 dataset, X-DNN achieved the highest accuracy in several categories, notably improving the detection in the backdoor and analysis categories.

Precision and recall metrics provided deeper insights into X-DNN's performance. On the UNSW NB 15 dataset, precision values varied across classes, highlighting X-DNN’s ability to correctly predict positive instances with high reliability. In contrast, the NSL KDD dataset showed precision values of 1 for normal and probe classes, indicating the model's exceptional accuracy in these classifications. The recall values, representing the model's ability to identify all relevant instances of a class, also favored X-DNN, particularly in the NSL KDD dataset. The F1-Score, a critical metric that combines precision and recall, reinforced the balanced performance of X-DNN. The model demonstrated a high F1-Score across various classes, signifying its capability to effectively manage false positives and negatives, a crucial aspect in cybersecurity threat detection.

In a comparative evaluation with other ML and DL models, X-DNN maintained its superiority, particularly on the more challenging UNSW NB 15 dataset. It showed higher accuracy, precision, recall, and F1-Scores than traditional ML and DL models. This comparative analysis underscored X-DNN effectiveness in handling complex and nuanced data types found in modern network traffic.

5.2 Limitation

While the results of X-DNN ensemble model are promising, it is imperative to acknowledge the existing limitations that impact the accuracy of cyber-attack identification in IoT devices. One critical limitation is the dependency on labeled data for training our machine learning models. The reliance on labeled data poses challenges as it restricts the model's ability to adapt swiftly to emerging cyber threats without retraining and updating, potentially affecting its accuracy in real-time scenarios. Additionally, the datasets utilized, namely the UNSW NB 15 and NSL KDD datasets, while valuable, might not encompass the entirety of potential cyber-attack variations observed in IoT environments.

This limitation could potentially hinder the model's capability to detect and categorize new, unconventional attack patterns not present in the training data, thereby impacting its overall accuracy in detecting novel threats. In terms of dataset size, the relatively small nature

of the UNSW NB 15 dataset might limit the model's capacity to capture the diverse spectrum of cyber-attacks prevalent in IoT ecosystems. Their limitations regarding newer attack patterns open avenues for future research to expand and enhance dataset representativeness. While the model achieves commendable accuracy rates based on these datasets, it's essential to recognize that generalization to unseen or diverse cyber threats could be hindered due to the dataset's limitations. Moving forward, addressing these limitations requires exploring advanced deep learning methodologies and potentially incorporating diverse data sources beyond network traffic data. The inclusion of metadata or contextual information could significantly enhance the model's capacity to discern nuanced cyber threats and improve accuracy.

5.3 Future Work

Future endeavors in this domain should prioritize the acquisition of larger and more diverse datasets that encapsulate the evolving landscape of cyber threats within IoT environments. The pursuit of extensive datasets, inclusive of emerging cyber-attack patterns and variations, will significantly augment the model's capabilities. Evaluating our model's performance across diverse datasets will be pivotal. It will not only bolster our understanding of its adaptability but also fortify its generalization capacities, paving the way for a more resilient and precise cyber-attack identification system.

Moreover, exploration into cutting-edge generalization techniques is crucial. This exploration will aim to fortify the model's resilience against newer, unforeseen cyber threats. Continual updates and expansion of the dataset to incorporate the latest cybersecurity threats will be instrumental. This continual update mechanism ensures our model remains relevant and effective in combating the ever-evolving cyber landscape. Looking ahead, the development of a real-time detection system emerges as a priority. A system that swiftly identifies and responds to cyber-attacks in real-time can significantly mitigate their impact. This proactive stance ensures a rapid response, minimizing potential damages within IoT environments.

REFERENCES

[1] S. Bagui, X. Wang, and S. Bagui, “Machine Learning Based Intrusion Detection for IoT Botnet,” *Int. J. Mach. Learn. Comput.*, vol. 11, no. 6, pp. 399–406, Nov. 2021, doi: 10.18178/ijmlc.2021.11.6.1068.

[2] P. M. Shakeel, S. Baskar, H. Fouad, G. Manogaran, V. Saravanan, and C. E. Montenegro-Marin, “Internet of things forensic data analysis using machine learning to identify roots of data scavenging,” *Futur. Gener. Comput. Syst.*, vol. 115, pp. 756–768, Feb. 2021, doi: 10.1016/j.future.2020.10.001.

[3] G. E. I. Selim *et al.*, “Towards Machine Learning Based Intrusion Detection in IoT Networks,” *IEEE Access*, vol. 9, no. 1, pp. 1801–1821, Jan. 2021, doi: 10.32604/cmc.2021.018466.

[4] K. Geetha and S. H. Brahmananda, “Network traffic analysis through deep learning for detection of an army of bots in health IoT network,” *Int. J. Pervasive Comput. Commun.*, vol. ahead-of-p, no. ahead-of-print, Jan. 2022, doi: 10.1108/IJPCC-10-2021-0259.

[5] M. M. N. Aboelwafa, K. G. Seddik, M. H. Eldefrawy, Y. Gadallah, and M. Gidlund, “A Machine-Learning-Based Technique for False Data Injection Attacks Detection in Industrial IoT,” *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8462–8471, Sep. 2020, doi: 10.1109/JIOT.2020.2991693.

[6] M. A. Ferrag, L. Shu, O. Friha, and X. Yang, “Cyber Security Intrusion Detection for Agriculture 4.0: Machine Learning-Based Solutions, Datasets, and Future Directions,” *IEEE/CAA J. Autom. Sin.*, vol. 9, no. 3, pp. 407–436, 2022, doi: 10.1109/JAS.2021.1004344.

[7] S. M. Tahsien, H. Karimipour, and P. Spachos, “Machine learning based solutions for security of Internet of Things (IoT): A survey,” *J. Netw. Comput. Appl.*, vol. 161, no. November 2019, p. 102630, Jul. 2020, doi: 10.1016/j.jnca.2020.102630.

[8] N. Islam *et al.*, “Towards Machine Learning Based Intrusion Detection in IoT Networks,” *Comput. Mater. Contin.*, vol. 69, no. 2, pp. 1801–1821, 2021, doi: 10.32604/cmc.2021.018466.

[9] R. Al-Amri, R. K. Murugesan, M. Man, A. F. Abdulateef, M. A. Al-Sharafi, and A. A. Alkahtani, “A review of machine learning and deep learning techniques for anomaly detection in iot data,” *Appl. Sci.*, vol. 11, no. 12, p. 5320, Jun. 2021, doi: 10.3390/app11125320.

[10] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. K. A. A. Khan, “Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review,” *Procedia Comput. Sci.*, vol. 171, no. 2019, pp. 1251–1260, 2020, doi:

10.1016/j.procs.2020.04.133.

[11] N. Raghavendra Sai, J. Bhargav, M. Aneesh, G. Vinay Sahit, and A. Nikhil, "Discovering network intrusion using machine learning and data analytics approach," in *Proceedings of the 3rd International Conference on Intelligent Communication Technologies and Virtual Mobile Networks, ICICV 2021*, IEEE, Feb. 2021, pp. 118–123. doi: 10.1109/ICICV50876.2021.9388552.

[12] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 1, p. e4150, Jan. 2021, doi: 10.1002/ett.4150.

[13] M. Abdullahi *et al.*, "Detecting Cybersecurity Attacks in Internet of Things Using Artificial Intelligence Methods: A Systematic Literature Review," *Electron.*, vol. 11, no. 2, p. 198, Jan. 2022, doi: 10.3390/electronics11020198.

[14] R. Ahmad and I. Alsmadi, "Machine learning approaches to IoT security: A systematic literature review[Formula presented]," *Internet of Things (Netherlands)*, vol. 14, p. 100365, Jun. 2021, doi: 10.1016/j.iot.2021.100365.

[15] K. Shinan, K. Alsubhi, A. Alzahrani, and M. U. Ashraf, "Machine learning-based botnet detection in software-defined network: A systematic review," *Symmetry (Basel).*, vol. 13, no. 5, p. 866, May 2021, doi: 10.3390/sym13050866.

[16] U. Urooj, B. A. S. Al-Rimy, A. Zainal, F. A. Ghaleb, and M. A. Rassam, "Ransomware Detection Using the Dynamic Analysis and Machine Learning: A Survey and Research Directions," *Appl. Sci.*, vol. 12, no. 1, p. 172, Dec. 2022, doi: 10.3390/app12010172.

[17] S. Ullah *et al.*, "A New Intrusion Detection System for the Internet of Things via Deep Convolutional Neural Network and Feature Engineering," *Sensors*, vol. 22, no. 10, 2022, doi: 10.3390/s22103607.

[18] S. Mahdavifar *et al.*, "Lightweight Hybrid Detection of Data Exfiltration using DNS based on Machine Learning," in *ACM International Conference Proceeding Series*, The 11th IEEE International Conference on Communication and Network Security (ICCNS), Dec. 3-5, 2021, Beijing Jiaotong University, Weihai, China., 2021, pp. 80–86. doi: 10.1145/3507509.3507520.

[19] A. K. Sahu, S. Sharma, M. Tanveer, and R. Raja, "Internet of Things attack detection using hybrid Deep Learning Model," *Comput. Commun.*, vol. 176, pp. 146–154, Aug. 2021, doi: 10.1016/j.comcom.2021.05.024.

[20] U. Islam *et al.*, "Detection of Distributed Denial of Service (DDoS) Attacks in IOT Based Monitoring System of Banking Sector Using Machine Learning Models," *Sustain.*, vol. 14, no. 14, 2022, doi: 10.3390/su14148374.

[21] G. E. I. Selim, E. E. D. Hemdan, A. M. Shehata, and N. A. El-Fishawy, “Anomaly events classification and detection system in critical industrial internet of things infrastructure using machine learning algorithms,” *Multimed. Tools Appl.*, vol. 80, no. 8, pp. 12619–12640, Mar. 2021, doi: 10.1007/s11042-020-10354-1.

[22] S. Mahdavifar, N. Maleki, A. H. Lashkari, M. Broda, and A. H. Razavi, “Classifying Malicious Domains using DNS Traffic Analysis,” *Proc. - 2021 IEEE Int. Conf. Dependable, Auton. Secur. Comput. Int. Conf. Pervasive Intell. Comput. Int. Conf. Cloud Big Data Comput. Int. Conf. Cybe*, no. March 2022, pp. 60–67, 2021, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech52372.2021.00024.

[23] T. Tatarnikova and P. Bogdanov, “Intrusion detection in internet of things networks based on machine learning methods,” *Inf. Control Syst.*, no. 6, pp. 42–52, Dec. 2021, doi: 10.31799/1684-8853-2021-6-42-52.

[24] M. Y. Alzahrani and A. M. Bamhdi, “Hybrid deep-learning model to detect botnet attacks over internet of things environments,” *Soft Comput.*, vol. 26, no. 16, pp. 7721–7735, 2022, doi: 10.1007/s00500-022-06750-4.

[25] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, “Machine learning-based IoT-botnet attack detection with sequential architecture,” *Sensors (Switzerland)*, vol. 20, no. 16, pp. 1–15, 2020, doi: 10.3390/s20164372.

[26] S. K. Pani, S. S. Rautaray, and M. Pandey, “IoT: the theoretical fundamentals and practical applications,” *Internet Things Enabling Technol. Secur. Soc. Implic.*, pp. 1–16, 2021.

[27] H. Karimipour, A. Dehghantanha, R. M. Parizi, K. K. R. Choo, and H. Leung, “A Deep and Scalable Unsupervised Machine Learning System for Cyber-Attack Detection in Large-Scale Smart Grids,” *IEEE Access*, vol. 7, pp. 80778–80788, 2019, doi: 10.1109/ACCESS.2019.2920326.

[28] A. Amouri, V. T. Alaparthy, and S. D. Morgera, “A machine learning based intrusion detection system for mobile internet of things,” *Sensors (Switzerland)*, vol. 20, no. 2, p. 461, Jan. 2020, doi: 10.3390/s20020461.

[29] Z. Liu, N. Thapa, A. Shaver, K. Roy, X. Yuan, and S. Khorsandroo, “Anomaly detection on lot network intrusion using machine learning,” in *2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems, icABCD 2020 - Proceedings*, IEEE, Aug. 2020, pp. 1–5. doi: 10.1109/icABCD49160.2020.9183842.

[30] S. H. Haji and S. Y. Ameen, “Attack and Anomaly Detection in IoT Networks using Machine Learning Techniques: A Review,” *Asian J. Res. Comput. Sci.*, no. June, pp. 30–46, Jun. 2021, doi: 10.9734/ajrcos/2021/v9i230218.

[31] Y. Li, Y. Zuo, H. Song, and Z. Lv, “Deep Learning in Security of Internet of Things,”

IEEE Internet Things J., vol. 9, no. 22, pp. 22133–22146, 2022, doi: 10.1109/JIOT.2021.3106898.

[32] A. R. Khan, M. Kashif, R. H. Jhaveri, R. Raut, T. Saba, and S. A. Bahaj, "Deep Learning for Intrusion Detection and Security of Internet of Things (IoT): Current Analysis, Challenges, and Possible Solutions," *Secur. Commun. Networks*, vol. 2022, 2022, doi: 10.1155/2022/4016073.

[33] H. V. Poor, M. Goldenbaum, and W. Yang, "Fundamentals for IoT networks: Secure and low-latency communications," in *Proceedings of the 20th International Conference on Distributed Computing and Networking*, 2019, pp. 362–364.

[34] S. Srivastava, A. Verma, and P. Verma, "Fundamentals of internet of things," in *Futuristic Research Trends and Applications of Internet of Things*, CRC Press, 2022, pp. 1–30.

[35] I. Alrashdi, A. Alqazzaz, E. Aloufi, R. Alharthi, M. Zohdy, and H. Ming, "AD-IoT: Anomaly detection of IoT cyberattacks in smart city using machine learning," in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019*, IEEE, Jan. 2019, pp. 305–310. doi: 10.1109/CCWC.2019.8666450.

[36] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, "A sequential scheme for detecting cyber attacks in IoT environment," in *2019 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, IEEE, 2019, pp. 238–244.

[37] A. Verma and V. Ranga, "Machine Learning Based Intrusion Detection Systems for IoT Applications," *Wirel. Pers. Commun.*, vol. 111, no. 4, pp. 2287–2310, Apr. 2020, doi: 10.1007/s11277-019-06986-8.

[38] N. F. Syed, Z. Baig, A. Ibrahim, and C. Valli, "Denial of service attack detection through machine learning for the IoT," *J. Inf. Telecommun.*, vol. 4, no. 4, pp. 482–503, Oct. 2020, doi: 10.1080/24751839.2020.1767484.

[39] F. Firouzi, B. Farahani, M. Weinberger, G. DePace, and F. S. Aliee, "Iot fundamentals: Definitions, architectures, challenges, and promises," *Intell. Internet Things From Device to Fog Cloud*, pp. 3–50, 2020.

[40] J. Li and Y. Cheng, "Design and implementation of voice-controled intelligent fan system based on machine learning," in *Proceedings of 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications, AEECA 2020*, IEEE, Aug. 2020, pp. 548–552. doi: 10.1109/AEECA49918.2020.9213552.

[41] D. Rani, N. S. Gill, and P. Gulia, "Classification of Security Issues and Cyber Attacks in Layered Internet of Things," *J. Theor. Appl. Inf. Technol.*, vol. 100, no. 13, pp. 4895–4913, 2022.

[42] S. Boopathi, “Securing Healthcare Systems Integrated With IoT: Fundamentals, Applications, and Future Trends,” in *Dynamics of Swarm Intelligence Health Analysis for the Next Generation*, IGI Global, 2023, pp. 186–209.

[43] F. Alghayadh and D. Debnath, “A Hybrid Intrusion Detection System for Smart Home Security,” in *IEEE International Conference on Electro Information Technology*, IEEE, Jul. 2020, pp. 319–323. doi: 10.1109/EIT48999.2020.9208296.

[44] Y. Maleh, “Machine learning techniques for IoT intrusions detection in aerospace cyber-physical systems,” in *Studies in Computational Intelligence*, Springer, 2020, pp. 205–232. doi: 10.1007/978-3-030-20212-5_11.

[45] M. K. Kagita, N. Thilakarathne, T. R. Gadekallu, P. K. R. Maddikunta, and S. Singh, “A review on cyber crimes on the Internet of Things,” *Deep Learn. Secur. Priv. Preserv. IoT*, pp. 83–98, 2022.

[46] T. Saba, T. Sadad, A. Rehman, Z. Mehmood, and Q. Javaid, “Intrusion Detection System through Advance Machine Learning for the Internet of Things Networks,” *IT Prof.*, vol. 23, no. 2, pp. 58–64, Mar. 2021, doi: 10.1109/MITP.2020.2992710.

[47] A. Djenna and D. E. Saïdouni, “Cyber attacks classification in IoT-based-healthcare infrastructure,” in *2018 2nd Cyber Security in Networking Conference (CSNet)*, IEEE, 2018, pp. 1–4.

[48] T. R. Gadekallu, M. M K, S. K. S, N. Kumar, S. Hakak, and S. Bhattacharya, “Blockchain-Based Attack Detection on Machine Learning Algorithms for IoT-Based e-Health Applications,” *IEEE Internet Things Mag.*, vol. 4, no. 3, pp. 30–33, Sep. 2021, doi: 10.1109/iotm.1021.2000160.

[49] P. Bedi *et al.*, “Detection of attacks in IoT sensors networks using machine learning algorithm,” *Microprocess. Microsyst.*, vol. 82, p. 103814, Apr. 2021, doi: 10.1016/j.micpro.2020.103814.

[50] M. Al-Sarem, F. Saeed, E. H. Alkhammash, and N. S. Alghamdi, “An aggregated mutual information based feature selection with machine learning methods for enhancing iot botnet attack detection,” *Sensors*, vol. 22, no. 1, p. 185, Dec. 2022, doi: 10.3390/s22010185.

[51] D. J. Atul, R. Kamalraj, G. Ramesh, K. Sakthidasan Sankaran, S. Sharma, and S. Khasim, “A machine learning based IoT for providing an intrusion detection system for security,” *Microprocess. Microsyst.*, vol. 82, p. 103741, Apr. 2021, doi: 10.1016/j.micpro.2020.103741.

[52] J. M. Peterson, J. L. Leevy, and T. M. Khoshgoftaar, “A Review and Analysis of the Bot-IoT Dataset,” in *Proceedings - 15th IEEE International Conference on Service-Oriented System Engineering, SOSE 2021*, IEEE, Aug. 2021, pp. 20–27. doi: 10.1109/SOSE52839.2021.00007.

[53] S. Pattar, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, “Searching for the IoT resources: Fundamentals, requirements, comprehensive review, and future directions,” *IEEE Commun. Surv. Tutorials*, vol. 20, no. 3, pp. 2101–2132, 2018.

[54] E. Rehman *et al.*, “Intrusion detection based on machine learning in the internet of things, attacks and counter measures,” *J. Supercomput.*, vol. 78, no. 6, pp. 8890–8924, Jan. 2022, doi: 10.1007/s11227-021-04188-3.

[55] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, “Machine Learning for the Detection and Identification of Internet of Things Devices: A SurveyKhan, A. R., Kashif, M., Jhaveri, R. H., Raut, R., Saba, T., & Bahaj, S. A. (2022). Deep Learning for Intrusion Detection and Security of Internet of Things (IoT),” *IEEE Internet Things J.*, vol. 9, no. 1, pp. 298–320, Jan. 2022, doi: 10.1109/JIOT.2021.3099028.

[56] A. M. Araujo, A. Bergamini de Neira, and M. Nogueira, “Autonomous machine learning for early bot detection in the internet of things,” *Digit. Commun. Networks*, 2022, doi: 10.1016/j.dcan.2022.05.011.

[57] Y. Kayode Saheed, A. Idris Abiodun, S. Misra, M. Kristiansen Holone, and R. Colomo-Palacios, “A machine learning-based intrusion detection for detecting internet of things network attacks,” *Alexandria Eng. J.*, vol. 61, no. 12, pp. 9395–9409, 2022, doi: 10.1016/j.aej.2022.02.063.

[58] U. Inayat, M. F. Zia, S. Mahmood, H. M. Khalid, and M. Benbouzid, “Learning-based methods for cyber attacks detection in IoT systems: A survey on methods, analysis, and future prospects,” *Electronics*, vol. 11, no. 9, p. 1502, 2022.

[59] M. Paricherla *et al.*, “Towards Development of Machine Learning Framework for Enhancing Security in Internet of Things,” *Secur. Commun. Networks*, vol. 2022, 2022, doi: 10.1155/2022/4477507.

[60] M. Shahin, F. F. Chen, A. Hosseinzadeh, H. Bouzary, and R. Rashidifar, “A deep hybrid learning model for detection of cyber attacks in industrial IoT devices,” *Int. J. Adv. Manuf. Technol.*, vol. 123, no. 5–6, pp. 1973–1983, 2022.

[61] T. T. H. Le, Y. E. Oktian, and H. Kim, “XGBoost for Imbalanced Multiclass Classification-Based Industrial Internet of Things Intrusion Detection Systems,” *Sustain.*, vol. 14, no. 14, pp. 1–21, 2022, doi: 10.3390/su14148707.

[62] SaiKira N, Pradeep Naidu P D S, Harshini K, Venkateswarlu M, and Surya Narayana Reddy V, “Detection of cyber attacks in network using machine learning techniques,” *South Asian J. Eng. Technol.*, vol. 12, no. 3, pp. 138–145, 2022, doi: 10.26524/sajet.2022.12.51.

[63] D. Rathee and S. Mann, “Detection of E-Mail Phishing Attacks – using Machine Learning and Deep Learning,” *Int. J. Comput. Appl.*, vol. 183, no. 47, pp. 1–7, Jan. 2022, doi: 10.5120/ijca2022921868.

[64] E. Y. Guven, S. Gulgun, C. Manav, B. Bakir, and Z. G. Aydin, “Multiple Classification of Cyber Attacks Using Machine Learning,” *Electrica*, vol. 22, no. 2, pp. 313–320, 2022, doi: 10.54614/electrica.2022.22031.

[65] S. Silvestri, S. Islam, S. Papastergiou, C. Tzagkarakis, and M. Ciampi, “A Machine Learning Approach for the NLP-Based Analysis of Cyber Threats and Vulnerabilities of the Healthcare Ecosystem †,” *Sensors*, vol. 23, no. 2, pp. 1–26, 2023, doi: 10.3390/s23020651.

[66] E. Rodríguez, B. Otero, and R. Canal, “A Survey of Machine and Deep Learning Methods for Privacy Protection in the Internet of Things,” *Sensors*, vol. 23, no. 3, 2023, doi: 10.3390/s23031252.

[67] J. Jose and D. V. Jose, “Deep learning algorithms for intrusion detection systems in internet of things using CIC-IDS 2017 dataset,” *Int. J. Electr. Comput. Eng.*, vol. 13, no. 1, pp. 1134–1141, 2023, doi: 10.11591/ijece.v13i1.pp1134-1141.

[68] B. Madhu and M. V. Gopalachari, “Classification of the Severity of Attacks on Internet of Things Networks,” in *Sentiment Analysis and Deep Learning: Proceedings of ICSADL 2022*, Springer, 2023, pp. 411–424.

[69] Y. Zhou, M. Han, L. Liu, J. S. He, and Y. Wang, “Deep learning approach for cyberattack detection,” in *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, IEEE, Apr. 2018, pp. 262–267. doi: 10.1109/INFCOMW.2018.8407032.

[70] M. Roopak, G. Yun Tian, and J. Chambers, “Deep learning models for cyber security in IoT networks,” in *2019 IEEE 9th Annual Computing and Communication Workshop and Conference, CCWC 2019*, IEEE, Jan. 2019, pp. 452–457. doi: 10.1109/CCWC.2019.8666588.

[71] F. Hussain, S. G. Abbas, M. Husnain, U. U. Fayyaz, F. Shahzad, and G. A. Shah, “IoT DoS and DDoS Attack Detection using ResNet,” in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, IEEE, Nov. 2020, pp. 1–6. doi: 10.1109/INMIC50486.2020.9318216.

[72] A. Tabassum and W. Lebda, “Security framework for iot devices against cyber-attacks,” *arXiv Prepr. arXiv1912.01712*, 2019.

[73] S. Jain, P. Choudhari, and A. Srivastava, “The fundamentals of Internet of Things: architectures, enabling technologies, and applications,” in *Healthcare Paradigms in the Internet of Things Ecosystem*, Elsevier, 2021, pp. 1–20.

[74] M. Savic *et al.*, “Deep Learning Anomaly Detection for Cellular IoT with Applications in Smart Logistics,” *IEEE Access*, vol. 9, pp. 59406–59419, 2021, doi: 10.1109/ACCESS.2021.3072916.

[75] I. Ullah and Q. H. Mahmoud, “Design and Development of a Deep Learning-Based Model for Anomaly Detection in IoT Networks,” *IEEE Access*, vol. 9, pp. 103906–103926, 2021, doi: 10.1109/ACCESS.2021.3094024.

[76] S. Gopali and A. Siami Namin, “Deep Learning-Based Time-Series Analysis for Detecting Anomalies in Internet of Things,” *Electron.*, vol. 11, no. 19, 2022, doi: 10.3390/electronics11193205.

[77] K. C. Ravikumar, P. Chiranjeevi, N. Manikanda Devarajan, C. Kaur, and A. I. Taloba, “Challenges in internet of things towards the security using deep learning techniques,” *Meas. Sensors*, vol. 24, no. June, p. 100473, 2022, doi: 10.1016/j.measen.2022.100473.

[78] A. Yazdinejad, M. Kazemi, R. M. Parizi, A. Dehghantanha, and H. Karimipour, “An ensemble deep learning model for cyber threat hunting in industrial internet of things,” *Digit. Commun. Networks*, vol. 9, no. 1, pp. 101–110, 2022, doi: 10.1016/j.dcan.2022.09.008.

[79] J. Alsamiri and K. Alsubhi, “Internet of things cyber attacks detection using machine learning,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 12, 2019.

[80] A. S. Rajawat, S. B. Goyal, C. Chauhan, P. Bedi, M. Prasad, and T. Jan, “Cognitive Adaptive Systems for Industrial Internet of Things Using Reinforcement Algorithm,” pp. 1–16, 2023.

[81] A. R. Zarzoor, N. A. S. Al-Jamali, and D. A. Abdul Qader, “Intrusion detection method for internet of things based on the spiking neural network and decision tree method,” *Int. J. Electr. Comput. Eng.*, vol. 13, no. 2, pp. 2278–2288, 2023, doi: 10.11591/ijece.v13i2.pp2278-2288.

[82] T. Yi, X. Chen, Y. Zhu, W. Ge, and Z. Han, “Review on the application of deep learning in network attack detection,” *J. Netw. Comput. Appl.*, vol. 212, no. June 2022, p. 103580, 2023, doi: 10.1016/j.jnca.2022.103580.

[83] A. Abusitta, G. H. S. de Carvalho, O. A. Wahab, T. Halabi, B. C. M. Fung, and S. Al Mamoori, “Deep learning-enabled anomaly detection for IoT systems,” *Internet of Things (Netherlands)*, vol. 21, no. October 2022, p. 100656, 2023, doi: 10.1016/j.iot.2022.100656.

[84] N. A. Bajao and J. Sarucam, “Threats Detection in the Internet of Things Using Convolutional neural networks, long short-term memory, and gated recurrent units,” *Mesopotamian J. Cyber Secur.*, vol. 2023, pp. 22–29, 2023, doi: 10.58496/mjcs/2023/005.

[85] F. M. Aswad, A. M. S. Ahmed, N. A. M. Alhammadi, B. A. Khalaf, and S. A. Mostafa, “Deep learning in distributed denial-of-service attacks detection method for Internet of Things networks,” *J. Intell. Syst.*, vol. 32, no. 1, 2023, doi: 10.1515/jisys-2022-0155.

[86] S. V. N. Santhosh Kumar, M. Selvi, and A. Kannan, “A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things,” *Comput. Intell. Neurosci.*, vol. 2023, pp. 1–24, 2023, doi: 10.1155/2023/8981988.

[87] A. S. Dina, A. B. Siddique, and D. Manivannan, “A deep learning approach for intrusion detection in Internet of Things using focal loss function,” *Internet of Things*, p. 100699, 2023.

[88] M. M. Rashid, J. Kamruzzaman, M. M. Hassan, T. Imam, and S. Gordon, “Cyberattacks detection in iot-based smart city applications using machine learning techniques,” *Int. J. Environ. Res. Public Health*, vol. 17, no. 24, p. 9347, 2020.

[89] T. Ghrib, M. Benmohammed, and P. S. Pandey, “Automated diagnosis of attacks in internet of things using machine learning and frequency distribution techniques,” *Bull. Electr. Eng. Informatics*, vol. 10, no. 2, pp. 950–961, Apr. 2021, doi: 10.11591/eei.v10i2.2766.

[90] S. Bi *et al.*, “A survey on artificial intelligence aided Internet-of-Things technologies in emerging smart libraries,” *Sensors*, vol. 22, no. 8, p. 2991, 2022.

[91] M. Najafimehr, S. Zarifzadeh, and S. Mostafavi, “A hybrid machine learning approach for detecting unprecedented DDoS attacks,” *J. Supercomput.*, vol. 78, no. 6, pp. 8106–8136, Jan. 2022, doi: 10.1007/s11227-021-04253-x.

[92] V. Hnamte and J. Hussain, “DCNNBiLSTM: An Efficient Hybrid Deep Learning-Based Intrusion Detection System,” *Telemat. Informatics Reports*, vol. 10, no. February, p. 100053, 2023, doi: 10.1016/j.teler.2023.100053.

[93] P. Devan and N. Khare, “An efficient XGBoost–DNN-based classification model for network intrusion detection system,” *Neural Comput. Appl.*, vol. 32, no. 16, pp. 12499–12514, 2020, doi: 10.1007/s00521-020-04708-x.

[94] T. T. Huong *et al.*, “LocKedge: Low-Complexity Cyberattack Detection in IoT Edge Computing,” *IEEE Access*, vol. 9, pp. 29696–29710, 2021, doi: 10.1109/ACCESS.2021.3058528.

[95] L. Saravanan, Himanshu Sharma, K. . Sreenivasulu, and M. Deivakani, “WITHDRAWN: Detection of software intrusion based on machine learning techniques for IOT systems,” *Mater. Today Proc.*, Apr. 2021, doi: 10.1016/j.matpr.2021.03.138.

[96] Y. Zhang, C. Yang, K. Huang, and Y. Li, “Intrusion Detection of Industrial Internet-of-Things Based on Reconstructed Graph Neural Networks,” *IEEE Trans. Netw. Sci. Eng.*, pp. 1–12, 2022, doi: 10.1109/TNSE.2022.3184975.

[97] T. Hasan *et al.*, “Securing Industrial Internet of Things Against Botnet Attacks Using

Hybrid Deep Learning Approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 09, no. 00, pp. 1–12, 2022, doi: 10.1109/TNSE.2022.3168533.

[98] A. Thakkar and R. Lohiya, "A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges," *Arch. Comput. Methods Eng.*, vol. 28, no. 4, pp. 3211–3243, Jun. 2021, doi: 10.1007/s11831-020-09496-0.

[99] I. H. Sarker, A. I. Khan, Y. B. Abushark, and F. Alsolami, "Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions," *Mob. Networks Appl.*, pp. 1–17, 2022.

[100] U. Farooq, N. Tariq, M. Asim, T. Baker, and A. Al-Shamma'a, "Machine learning and the Internet of Things security: Solutions and open challenges," *J. Parallel Distrib. Comput.*, vol. 162, pp. 89–104, 2022.

[101] M. S. Akhtar and T. Feng, "Comparison of Classification Model for the Detection of Cyber-attack using Ensemble Learning Models," *EAI Endorsed Trans. Scalable Inf. Syst.*, vol. 9, no. 5, pp. 1–11, 2022, doi: 10.4108/eai.1-2-2022.173293.

[102] A. Gaurav, B. B. Gupta, and P. K. Panigrahi, "A comprehensive survey on machine learning approaches for malware detection in IoT-based enterprise information system," *Enterp. Inf. Syst.*, vol. 17, no. 3, pp. 1–25, Jan. 2023, doi: 10.1080/17517575.2021.2023764.

[103] L. Zou, X. Luo, Y. Zhang, X. Yang, and X. Wang, "HC-DTTSVM: A Network Intrusion Detection Method Based on Decision Tree Twin Support Vector Machine and Hierarchical Clustering," *IEEE Access*, vol. 11, no. February, pp. 21404–21416, 2023, doi: 10.1109/ACCESS.2023.3251354.

[104] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," *2015 Mil. Commun. Inf. Syst. Conf. MilCIS 2015 - Proc.*, 2015, doi: 10.1109/MilCIS.2015.7348942.

[105] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set in Computational Intelligence for Security and Defense Applications," *Comput. Intell. Secur. Def. Appl.*, no. Cisda, pp. 1–6, 2009.

[106] M. Ahmad, Q. Riaz, M. Zeeshan, H. Tahir, S. A. Haider, and M. S. Khan, "Intrusion detection in internet of things using supervised machine learning based on application and transport layer features using UNSW-NB15 data-set," *Eurasip J. Wirel. Commun. Netw.*, vol. 2021, no. 1, p. 10, Dec. 2021, doi: 10.1186/s13638-021-01893-8.

[107] T. Verdonck, B. Baesens, M. Óskarsdóttir, and S. vanden Broucke, "Special issue on feature engineering editorial," *Mach. Learn.*, no. 0123456789, 2021, doi: 10.1007/s10994-021-06042-2.

[108] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, “Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance,” *Technologies*, vol. 9, no. 3, pp. 5–9, 2021, doi: 10.3390/technologies9030052.

[109] X.-D. Zhang, “Support Vector Machines (SVM) Support Vector Machines (SVM),” *Gesture*, vol. 23, no. 6, pp. 349–361, 2001.

[110] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, and S. D. Brown, “An introduction to decision tree modeling,” *J. Chemom.*, vol. 18, no. 6, pp. 275–285, 2004, doi: 10.1002/cem.873.

[111] M. R. Segal, “Machine Learning Benchmarks and Random Forest Regression,” *Biostatistics*, pp. 1–14, 2004, [Online]. Available: http://escholarship.org/uc/item/35x3v9t4.pdf

[112] H. Larochelle, Y. Bengio, J. Louradour, and P. Lamblin, “Exploring strategies for training deep neural networks,” *J. Mach. Learn. Res.*, vol. 10, no. 1, pp. 1–40, 2009.

[113] T. Chen and T. He, “xgboost: Extreme Gradient Boosting,” *R Lect.*, no. 2016, pp. 1–84, 2014.

[114] R. Dey and F. M. Salem, “Gate-Variants of Gated Recurrent Unit (GRU),” *Midwest Symp. Circuits Syst. Inst. Electr. Electron. Eng. Inc.*, vol. 784, no. 2017, pp. 1597–1600, 2017.

[115] A. Botalb, M. Moinuddin, U. M. Al-Saggaf, and S. S. A. Ali, “Contrasting Convolutional Neural Network (CNN) with Multi-Layer Perceptron (MLP) for Big Data Analysis,” *Int. Conf. Intell. Adv. Syst. ICIAS 2018*, pp. 1–5, 2018, doi: 10.1109/ICIAS.2018.8540626.

[116] F. Huang, G. Xie, and R. Xiao, “Research on ensemble learning,” *2009 Int. Conf. Artif. Intell. Comput. Intell. AICI 2009*, vol. 3, pp. 249–252, 2009, doi: 10.1109/AICI.2009.235.

[117] N. Pilnenskiy and I. Smetannikov, “Modern Implementations of Feature Selection Algorithms and Their Perspectives,” *Conf. Open Innov. Assoc. Fruct*, no. November 2019, pp. 250–256, 2019, doi: 10.23919/FRUCT48121.2019.8981498.

[118] S. Moraboena, G. Ketepalli, and P. Ragam, “A deep learning approach to network intrusion detection using deep autoencoder,” *Rev. d’Intelligence Artif.*, vol. 34, no. 4, pp. 457–463, 2020, doi: 10.18280/ria.340410.

[119] W. L. Al-Yaseen, A. K. Idrees, and F. H. Almasoudy, “Wrapper feature selection method based differential evolution and extreme learning machine for intrusion detection system,” *Pattern Recognit.*, vol. 132, p. 108912, 2022.