Republic of Iraq
Ministry of Higher Education
and Scientific Research
University of Anbar
College of Computer Science
and Information Technology
Department of Computer Science

Intrusion Detection Approaches for Internet of Things

A Thesis

Submitted to the Department of Computer Science , College of Computer Science and Information Technology, University of Anbar as a Partial Fulfillment of the Requirements for Master Degree of Science in Computer Sciences

By:
Maha Majid Mohammed

Supervised by:
Asst Prof. Dr. Khattab M. Ali Alheeti

2021A.D **1443 A.H**

بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ

﴿ وَلَقَدْ آتَيْنَا دَاوُودَ وَسُلَيْمَانَ عِلْمًا ۖ وَقَالَا الْحَمْدُ لِلَّهِ
الَّذِي فَضَّلَنَا عَلَىٰ كَثِيرٍ مِّنْ عِبَادِهِ الْمُؤْمِنِينَ ﴾

صدق الله العظيم

(النمل- ١٥)

اسم الطالب: مها ماجد محمد
كلية علوم الحاسوب وتكنولوجيا المعلومات ـ قسم علوم الحاسبات
عنوان الرسالة: مناهج كشف التسلل لأنترنت الاشياء

طبقا لقانون حماية حق المؤلف رقم 3 لسنة 1971 المعدل العراقي فأن للمؤلف حق منع اي حذف او تغيير للرسالة او الأطروحة بعد إقرارها وهي الحقوق الخاصة بالمؤلف وحده والتي لا يجوز الاعتداء عليها. فلا يحق لأحد ان يقرر نشر مصنف أحجم مؤلفه عن نشره او اعادة نشر مؤلف لم يقر مؤلفه بذلك، فإذا قام بذلك اعتبر عمله غير مشروع لأنه إستعمل سلطة لا يملكها قانونيا.

كلية علوم الحاسوب
وتكنولوجيا المعلومات

عنوان البحث			
Deep Learning Model for IDS in the Internet of Things			
جهة النشر	7th International Conference on Contemporary Information Technology and Mathematics, 2021 (ICCITM)	نوع المؤتمر	Scopus
حالة النشر	Accepted	رقم المؤتمر	
رابط البحث			

عنوان البحث			
Evaluating Machine Learning Algorithms to Detect and Classify Attacks in IoT			
جهة النشر	2021 International Conference on Communication and Information Technology (ICICT21)	نوع المؤتمر	Scopus
حالة النشر	Accepted	رقم المؤتمر	
رابط البحث			

اسم وتوقيع المشرف

أ.م. د خطاب معجل علي

اسم وتوقيع رئيس القسم

أ.م.د. وسام محمد جاسم

Supervisor Certification

*I certify that I read this thesis entitled “*Intrusion Detection Approaches for Internet of Things*” was prepared under my supervision at the department of computer science – College of Computer Science and Information Technology – University of Anbar, by “*Maha Majid Mohammed*” as partial fulfillment of the requirements of the degree of Master of Science in Computer Science.*

Signature:

Name: Asst Prof. Dr. Khattab M. Ali Alheeti

Date: /   /2021

Certification of the Examination Committee

We the examination committee certify that we have read this thesis entitled " Intrusion Detection Approaches for Internet of Things *" and have examined the student "* Maha Majid Mohammed *", in its contents and what is related to it, and that in our option it is adequate to fulfill the requirements for the degree of* Master of Computer Science.

Signature:
Name: **Prof. Dr. Sufyan Taih Faraj Al-Janabi (Chairman)**
Date: / / 2021

Signature:
Name: **Asst. Prof. Dr. Khalid Shaker Jasim (Member)**
Date: / / 2021

Signature:
Name: **Asst. Prof. Dr. Yousra Abdul alsahib S. Aldeen (Member)**
Date: / / 2021

Signature:
Name: **Asst. Prof. Dr. Khattab M. Ali Alheeti (Supervisor)**
Date: / / 2021

Approved by the Dean of the College of Computer Science and Information Technology, University of Anbar.

Signature:
Name: **Prof. Dr. Salah Awad Salman**
Title: **Dean of the College**
Date: / / 2021

Acknowledgments

First of all, praise be to Allah, the most Beneficent, and the most Merciful for his blessings and help that he has bestowed on me throughout my life’s journey and his guidance that has enabled me to complete my project.

Then, I would like to express my sincere gratitude to my supervisor, Asst. Prof. Dr. Khattab M. Ali Alheeti for the continuous support of my master's study and research for his motivation, encouragement, and advice that he gave to me throughout the progress of this project.

Last but not least, I would like to thank all my family who has supported me during my life stages. I wish to express my sincere gratitude to my friends for their continuous support that helped me overcome some difficult moments. I also convey my thankfulness towards those that don’t seem to be listed here. So many people have helped me during my Master's study.

Maha Majid

2021

Dedication

I dedicate my work to...

The woman without whom I would never be who I am, who gives me love, strength, and courage, the person to whom I am still indebted, The greatest mother of all(mom).

For my first teacher, the great man who offers his life for us to get to where we are now (dad).

To my lovely sisters and my friendly brothers who always supported me

To my best friend, Jory

Thanks to all my friends for their moral support and encouragement.

Maha Majid

2021

Student Name: Maha Majid Mohammed

Thesis Title: Intrusion Detection Approaches for Internet of Things

Abstract

The Internet of Things (IoT) links "everything" to the internet and enables "things" to communicate with one another over wired or wireless networks. The number of IoT applications has significantly increased, such as smart cities, smart homes, wearable's, and healthcare. Security becomes more important as the number of devices connected to the IoT increases due to the types of devices, the volume of data that is sent over the network, the nature of the structure, and the different communication methods (primarily wireless). This fundamental nature of the IoT architecture intensifies the number of attack targets that may affect the sustainable growth of the IoT. Hence, security issues become a critical factor that must be addressed. Therefore, it became necessary to develop an attack detection system to keep pace with the current development of the IoT, as it deals with sensitive information that must be protected and secured.

In this thesis, a deep learning approach based on convolutional neural networks is proposed to perform real-time detection of attack behaviors in IoT systems. The UNSW-NB15 dataset was used to train and test the proposed approach. The approach uses binary classification to distinguish attack and normal patterns. The proposed intrusion detection approach passes through two stages. The first stage: After loading the dataset, pre-processing is performed to obtain more accurate results. The second stage is classification by CNN classifier where the experimental result shows the efficiency of the presented approach concerning precision, recall, and f-measure as the detection precision reached 100%. The result of experiments is highly efficient for intrusion detection to distinguishing between normal and attack behaviors that provides a research approach.

Keywords: Security, Internet of Things, Machine Learning, Deep Learning, Convolutional Neural Networks.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviations	Details
AI	Artificial Intelligence
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DDoS	Distributed Denial of Service
DES	Data Encryption Standard
DL	Deep Learning
DNN	Deep Neural Network
DoS	Denial of Service
DT	Decision Tree
IDSs	Intrusion Detection Systems
IoT	Internet of Things
LR	Logistic Regression
MDP	Markov Decision Process
M-G	Middle box-Guard
ML	Machine Learning
NB	Naïve Bayes
PSNR	Peak Signal-to-Noise Ratio
QoS	Quality of Service
RAT	Radio Access Technology
RF	Random Forest
RNNs	Recurrent Neural Networks
SDN	Software Defined Networks
SJF	Short Job First
SNN	Spiking Neural Network
SVM	Support- Vector Machine
UAKMP	User Authenticated Key Management Protocol

CHAPTER ONE

Introduction

Chapter One

Introduction

1.1 Introduction

The dizzying expansion in the use of technology and modern electronic applications has led to tremendous developments and an emerging telecommunication model in communication networks called the Internet of Things (IoT). It provides communication between everyone and everything in order to exchange information and allows the inclusion of a kind of intelligence in the object that enables it to sense in its environment [1]. The IoT consists of real physical things integrated with electronics, software, and sensors, providing them with intelligence to sense and process information about the object and control it remotely through the network infrastructure, which significantly helps enhance efficiency, accuracy, and independent decision-making [2]. The number of devices connected to the internet has highly increased. As there will be more than 50 billion connected things to the IoT network in 2023 and 75 billion in 2025, according to research conducted by Statista, as illustrated in Figure 1.1 [3][4].

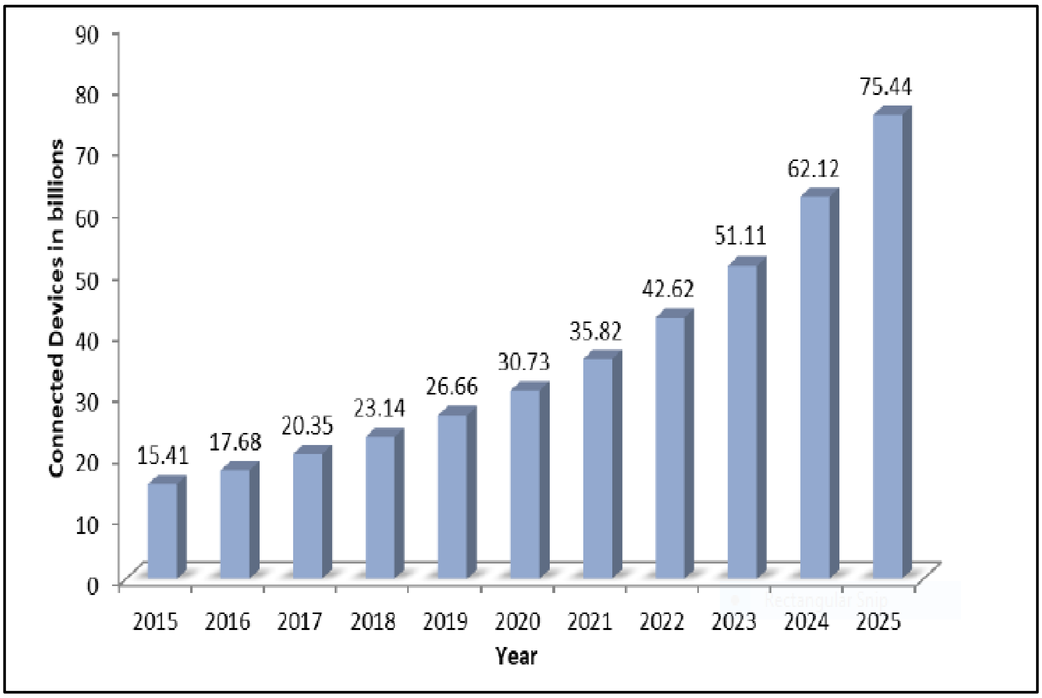


Figure (1.1): Number of devices connected in IoT [3]

IoT devices have diversified and expanded from millions to billions of connected things, where sensors collect data processed on a cloud, remote server, or network edge [5]. The data collected is sensitive to some extent, especially for the individuals because it is part of their personal life. Therefore, data security and privacy protection are the major issues in IoT networks that must be addressed in all these smart environments and Prevent an unauthorized entity to access them [6]. Interventions are known as the set of procedures that endanger security objectives to obtain information, intrude privacy, or control a system by exploiting some security weaknesses. Therefore, stringent security standards must be implemented to protect data from mitigating security attacks related to the IoT [7]. Exploiting the security vulnerabilities of some systems in the United States led to the launch of Distributed Denial of Service (DDoS) attacks that affected IoT devices and some websites and online services such as PayPal, Netflix, and Twitter [6]. In addition, one of the biggest breaches that occurred in the US company Target led to the theft of about 40 million registered clients due to the installation of malicious programs on the company's selling points devices, including financial data like credit card and debit information [7]. Various kinds of these attacks, as a Denial of Service (DoS) attack and (DDoS) attack that can seriously harm IoT services, so designing effective new solutions to maintain privacy and ensure data security against a wide range of attackers is a major concern [6]. The challenge is further complicated by the increasing size and diversity of devices. Therefore, traditional privacy and security politick cannot be enforced on IoT technologies directly. The IoT security issue is dangerous, thus it is necessary to understand the security vulnerabilities and develop corresponding security approaches [1].

In this thesis, the primary goal is to propose a deep learning-based security model for IoT. The proposal provides a practical method to applying the Deep Learning (DL) approach and improving its efficiency in detecting intrusions in the IoT with high detection accuracy and low false alarm rates in professional manner. Two models of Machine Learning (ML) techniques were applied, which are Naïve Bayes (NB) and Bayes Net (BN). Then, the results of these techniques will be compared with approach of DL.

1.2 Literatures Review

Many researchers have introduced various methods and approaches to enhancing security systems for IoT. This section focuses on previous studies, some of which were related to this work. The literature review reviewed below:

Prachi. et al. (2017) designed three new IoT Intrusion Detection Systems (IDSs) which are decision tree-based supervised IDS, K-means clustering unsupervised learning based IDS in addition to K-means, and decision tree learning methods are combined in a hybrid two-stage IDS [8]. All three IDS systems are centralized and scalable. The K-means method achieves a detection rate of 70-93 % for various sizes of random IoT networks. Decision tree-based IDS achieves a detection rate of 71-80 % for the same network sizes, whereas the hybrid method achieves a detection rate of 71-75 %. Although the hybrid IDS has a lower detection rate than the other two methods, it is more reliable.

Yanbing. et al. (2018) designed a safe use model Software Defined Networks (SDN) based data transfer security model Middle box-Guard (M-G) for data transfer in IoT systems Where (M-G) aims to ensure safe network operation by reducing network latency and properly managing data flow [9]. First, using data-flow abstraction and a heuristic algorithm, the middle boxes for specific safety policies are placed in the most convenient locations. Then, to address the limitations of the switch size, an offline Integer Linear Program (ILP) pruning algorithm in M-G was proposed to avoid any medium square becoming a hotspot. With the data flow management protocol, network routing is flexibly resolved. The results show that the system improves security performance and effectively manages data flow in SDN.

Mohammad. et al. (2018) designed the User Authenticated Key Management Protocol (UAKMP) which is a new, secure, and lightweight Hierarchical IoT Network (HIoTN) authentication system [10]. UAKMP used three factors: password, the user's smart card and personal biometrics. Compared to other relevant current schemes, the UAKMP system has provided many functional features, including the possibility of biometric update and freely password facility, user anonymity, and sensing node anonymity in addition to offline sensing node registration.

Amjad. et al. (2018) presented an advanced intrusion detection system and protect IoT infrastructure from distributed denial-of-service attacks caused by hackers [11]. The method uses the Naïve Bayes (NB) classification algorithm and multiple agents to detect attacks. It provides better performance compared to the traditionally used IDS systems. It aimed to secure the IoT network layer from DDoS attacks simulated by malicious objects, and attack detection and prevention are reported very quickly.

Mohamed. et al. (2018) presented more progress in medical image security in the IoT through the use of optimization strategies with an innovative encryption model [12]. The optimal key is selected using Hybrid Swarm Enhancement (HSE) to increase the security level of the encoding and decoding process, i.e., improving the grasshopper and improving the particle swarm in the elliptic curve coding. By analyzing and comparing the experimental results, which indicated control image quality against all tests, the researchers used the necessary measures such as Peak Signal-to-Noise Ratio (PSNR) and Supplemental Security Income (SSI). The proposed algorithm also takes less time for the encoding and decoding process.

Jadel. et al. (2019) presented seven different Machine Learning (ML) algorithms evaluated to contribute in the literature that can be used to quickly and effectively detect IoT network attacks [13]. Bot-IoT is a recent dataset that is being used to test various detection algorithm. The implementation of new features derived from the Bot-IoT dataset was compared to studies from the literature, and the new features produced better results. According to the F-measure, these algorithms achieved output ratios ranging from 0 to 1.

Sana. et al. (2019) Suggested a method for detecting an attempt to flood the network with large amounts of unnecessary data by an adversary to prevent the nodes from using services in the IoT [14]. A Support-based Vector Machine learning (SVM) was used to develop a detection strategy. According to experimental results, the proposed system can work satisfactorily from where of detection time and classification accuracy with the help of a combination of two or three uncomplicated features.

Liu. et al. (2019) proposed a deep learning model to effectively solve the network intrusion classification problem with high-dimensional and complex features [15]. Testing on invisible data demonstrates high accuracy and low false alarm rate as an experiment was conducted to determine the activation function and optimal features. The evaluation results showed that the proposed open new dimensions to research in network penetration detection through the superiority of the classifier over other machine learning models achieved 99% accuracy with FAR drop.

Haowei. et al. (2019) presented a triple data encryption approach to develop the CIA Triad aspects (data confidentiality, availability, and integrity) that included failover system, triple data encryption, and the Short Job First (SJF) scheduling algorithm [16]. The presented system combines digital watermarks Rivest–Shamir–Adleman (RSA) and Data Encryption Standard (DES). The experimental results showed that to encode the RGB array the cipher text takes a long time, but it may be too complex that no one wants to have it decoded. The Standard deviation is higher than the full system as indicated by the load test, and the failover approach works with write type only.

Samir. et al. (2019) developed a lightweight intrusion detection system (IDS) based on two machine learning techniques, feature selection, and feature classification, to improve IoT protection [17]. Because of its low computational cost, the filter-based approach was used to select the features. The feature classification algorithm for the system was identified through comparison between different classification algorithms such as NB, Random Forest (RF), logistic Regression (LR), Decision Tree (DT). Due to its outstanding performance on several datasets the DT algorithm was selected for the system.

Geethapriya. et al. (2019) presented an intelligent intrusion detection system designed specifically for cyber-attacks on the IoT [18]. To detect harmful traffic in networks, use a DL algorithm to facilitate interoperability between the various network communication protocols used in the IoT. Using both real network tracking to provide proof of concept, the proposed disclosure framework, as well as simulations, were evaluated to provide evidence of its scalability. From an analysis of the experiment results, it appears that an accuracy rate of 95% and a recovery rate of 97% for different attack scenarios.

Mohamed. et al. (2020) presented a system that combines different classification methods based on decision tree and rule-based concepts, namely the REP tree, Forest PA, and JRip algorithm aimed at Detecting Intrusion (IDS) of IoT networks [19]. The dataset takes methods one and two as input features and classifies network traffic as a benign attack. Compared to the latest current schemes, the results of the experiment showed that the proposed IDS are superior in detection rate, reliability, false alarm rate, and overtime.

Arul. et al. (2019) proposed a multi-layered perspective-based ID system (MLP) in Artificial Neural Network (ANN) to detect attacks in an IoT environment initiated by a destination-directed dashed graph information request attack and release attack [20]. Use Contiki O.S / Cooja Simulator 3.0 to simulate IoT. The experiment showed maximum values of true positive rate, accuracy, and recall.

Sarika. et al. (2020) used the Deep Neural Network (DNN) to identify attacks in the Internet of Things [21]. Model performance was evaluated to identify the attack on three data sets correctly. The results showed that the system was able to successfully identify the attacking behavior and the accuracy rate of the proposed method using DNN indicates that the accuracy rate is higher than 90%.

S Malliga. et al. (2020) the naïve bayes classification algorithm is used in this method for intrusion detection systems in IoT networks. The findings are compared to those obtained by using a DL model [22]. When tested on test data, the result shows that the accuracy of detecting the attack improved when the DL algorithm is used instead of the machine learning algorithm for network data classification into normal/attack with an accuracy rate of more than 99%.

Sahar. et al. (2020) proposed a new hybrid infiltration detection system for deep learning and Dendritic Cell Algorithm(DeepDCA) [23]. This mechanism aims to classify IoT interference and reduce the generation of false alarms based on the DCA and the autonomous Spiking Neural Network (SNN). In addition to improving classification performance by automating and facilitating the signal extraction stage. Results show that the system has more than 98.73% reliability and a low false-positive ratio in detecting IoT attacks.

Bomin. et al. (2020) proposed an adaptive security specialization model for 6G IoT networks based on Artificial Intelligence (AI) [24]. To predict future harvest strength, the Extended Kalman Filtration (EKF) method was adopted. Then layout a mathematical paradigm to account for the energy required for various safety strategies in each perceptible energy cycle and select the higher level supported protection that the service can meet requirements and Avoid depleting energy. The experimental results showed that the proposal adjusts security protection to avoid energy depletion as well as provides security protection satisfied for various services.

Zaid. et al. (2021) proposed a Full-Duplex (FD) Cooperative Jamming(CJ) scheme to provide a higher level of security for multi-branch IoT networks in the presence of random eavesdropping devices [25]. It is equipped with only two antennas. By two Adjacent nodes, one of which is an FD receiver, each information code is protected by two jamming signals. The experimental results showed If the receiving node works with high accuracy and allocates both antennas to receive data, higher confidentiality performance is obtained when the SNR is reduced while leaving one jammer active. The system has shown great flexibility in facing multi-stage networks wherewith the increasing number of hops, only marginal loss in performance occurs.

Muhammad. et al. (2021) presented a system for detecting and removing harmful packets from entering the Internet of Things by applying supervised ML algorithms such as Random Forest (RF), Support Vector Machine, and Artificial Neural Networks [26]. It suggested feature sets in terms of flow Transmission Control Protocol (TCP) and Message Queuing Telemetry Transfer (MQTT) using the features in the UNSW-NB15 dataset. Eliminate the problems of imbalance, overfitting, and the curse of dimensionality in the dataset. The model achieved 98.67% and 97.37% accuracy in dual- and multi-category classification and also required less training time compared to the newer methods based on ML-supervised.

Md Arafatur. et al. (2021) presented an approach that adopts a centralized intrusion detection system, using classification, deep feature abstraction, and feature selection to train the model for detect malicious and anomalous actions in IoT [27]. To create more features for traffic, it uses DL techniques of an artificial neural network in the form of an unsupervised automatic encoder to strip deep features. Then select highly rated features by various coat-based feature selection techniques ranging from Naive Bayes, SVM, and Decision Tree. Analytical results revealed that the system achieves high accuracy, up to 99.95%.

1.3 Problem Statement

The IoT plays an important role in both military and civilian applications. There have been rapid developments in smart systems, but still face many challenges some of it list below :

1. One of the most important problems facing these environments is the exposure of networks to different types of attacks
2. IoT is being deployed in an open medium and unprotected environment. Thus, detection and elimination of attacks is a serious problem that requires effective techniques to secure against different types of attacks.
3. A lot of research focus on the importance of increasing the accuracy of the system regardless of other performance measures.
4. The problem is that most current machine learning models either have low detection accuracy or can achieve high detection accuracy but have high False Alarm Rates (FAR), resulting in reduced detection efficiency.
5. The main goal is to examine and provide an effective method for applying the DL approach and improving its performance in detecting intrusions in IoT network infrastructure professionally and efficiently, with a high degree of detection accuracy and low FAR.
6. The Convolutional Neural Network(CNN) is the DL model that will be used in the detection system.
7. Some machine learning algorithms, i.e., Naive Bayes and Bayes Net, will be used to compare the proposed deep approach with machine learning performance.

1.4 Thesis Objectives

This thesis goal is to proposed a security approach for the detection of intrusions for the IoT network layer. The objectives of this thesis are as follow:

1. A novel deep neural network (CNN) model will be constructed. CNN approach training and evaluation to detecting intrusions in IoT networks.
2. Determine the most likely optimal hyper-parameters that the proposed approach would take to achieve the maximum detection precision and a low FAR.

1.5 The Main Contributions

The main contribution of this work is to achieve high precision performance and low FAR by implemented the proposed detection approach to classify intrusion in IoT network layer although the imbalance between normal and attack behavior in dataset.

1.6 Thesis Outline

This study is presented in five chapters. The first chapter presents the introduction, scope, and objectives of the research.

Chapter Two: presents the theoretical background and basic concept of IoT and its applications, details on security issues as well as security challenges and the types of threats, explains machine learning and deep learning and clarifies the difference between them.

Chapter Three: explains the steps of the suggested approach and techniques that are used. A whole description of the suggested approach is given.

Chapter Four: presents the results of tests that have been implemented to estimate the approach performance. The results of the experiment are discussed.

Chapter Five: summaries the conclusions, recommendations for future works, and limitations of this research.

CHAPTER TWO

Theoretical Background

Chapter Two

Theoretical Background

2.1 Introduction

Data security and privacy are the most difficult challenges that networks face in IoT, and it is a basic requirement for all types of computer systems. Interoperability, mixing, and independent decision-making will deliver the potential risks associated with loT to new complexity and possible vulnerability levels. Among the challenges of data in loT systems is that a lot of information is linked to the personal information, such as location, date of birth, budgets, etc. The sources of threats may be individuals, groups, or organizations, some of which affect the integrity and confidentiality of information, and other threats impact the availability of the system. The network should be designed so that less private data cannot be used easily to extract more private data as well as to protect user data from outright theft [28].

This chapter explains IoT systems and the main layers that the system consists of, with a description of the Cloud in IoT. There are details on security issues as well as security challenges and the types of threats each layer faces. The chapter provides some details on intrusion detection systems in IoT environments, in addition to artificial intelligence. Finally, it explains machine learning and deep learning and clarifies the difference between them.

2.2 The Internet of Things

The IoT is a significant technology in a variety of fields such as sensor networks, pervasive information systems, and embedded computing, which has made it a marketing trend and a popular news item [28]. IoT does not require human or human-computer intervention to operate by using a unique identification based on Radio-Frequency Identification (RFID) communication. For a huge address space, the IoT devices use IPv6 unique addresses and protocols standard communication system as the IoT technology evolves incredibly rapidly and through. The heterogeneous IoT network integrates large number of technologies. It is a hot topic of research due to the high interest in this technology and the large global investments devoted to it [29]. An example of the spread of computing devices is the smart refrigerator that includes many built-in computers and allows users to enter information about the contents of their fridge to plan the menu and take care of data entry the conceptual devices automatically scan the contents of the refrigerator [28].

The architecture of IoT is viewed as data at rest and data at motion state, where the data begins in the first layer of physical devices that pass through the communication and processing units. Then the third layer provides knowledge by showing hidden information from the structured data for an intelligent action at the end [29].

While the IoT tries to produce a large amount of data and link things in a single network to obtain actionable information, IoT is everywhere in daily life. They are used in hospitals, homes, widespread overseas control, reporting changes in the environment, preventing fires, and other issues [30]. However, it faces many challenges that must be addressed, including data integration from multiple sources such as social networks, sensors, and mobile devices. All in different contexts plus scalability because IoT deals with large amounts of data of various sizes, accuracy, and truthfulness, which is challenging for working in real-time and processing data efficiently. In addition to significant risks related to loss of privacy and security issues, all that's is shown in Figure 2.1.

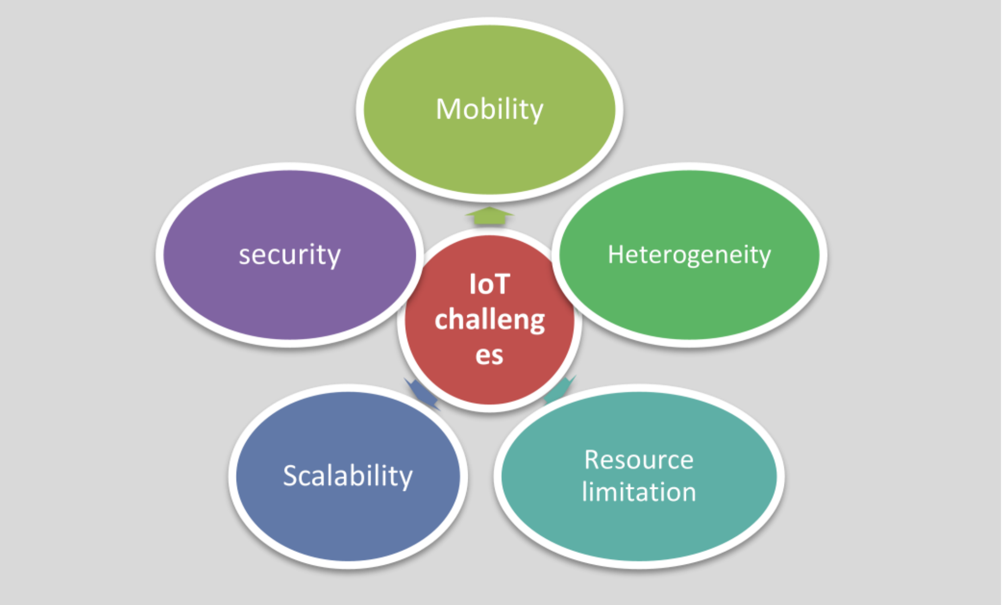


Figure (2.1): The IoT challenges.

To make the IoT environment more reasonable for the customer who use it in an intelligent connected environment, it must be able to address fundamental issues such as control, independence, monitoring, and improvement [29]**.** the IoT consists of several stages, from data collection until end-user distribution outside of demand or on-demand. Figure 2.2 shows the five different phases of IoT systems [31].

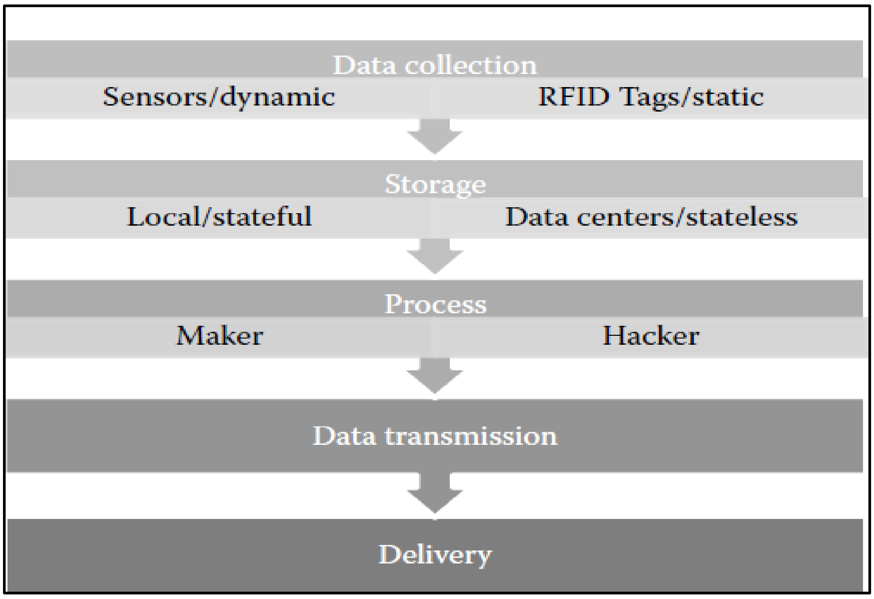


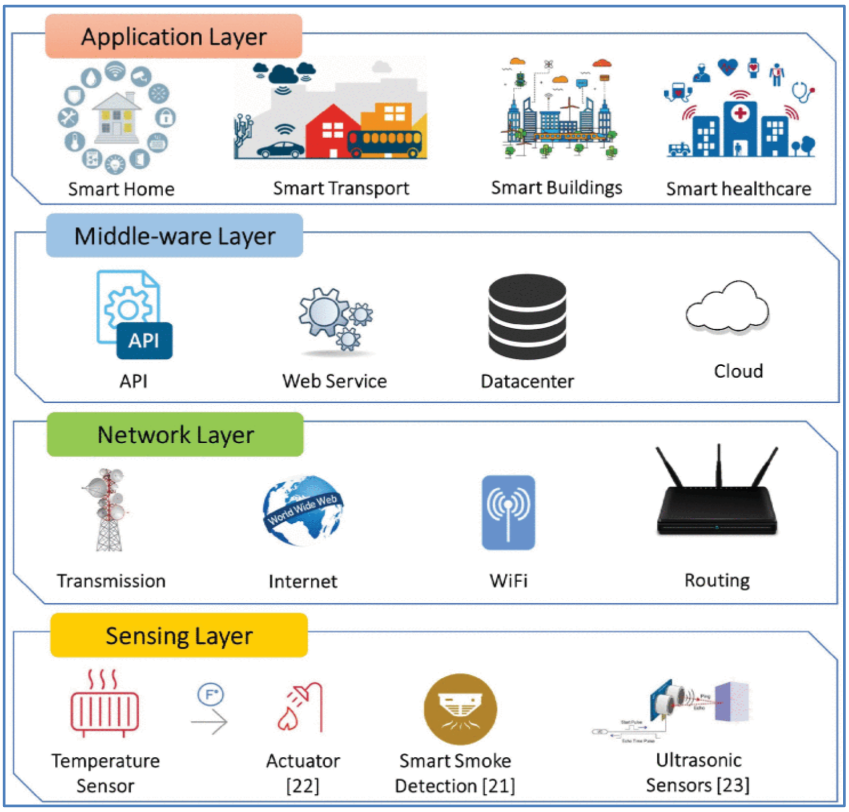
Figure (2.2): Block diagram phases of IoT system [31].

2.2.1 Internet of Things Architectures

IoT architecture addresses basic factors such as reliability, Quality of Service (QoS), integrity, confidentiality, etc. [32]. In this part, will take a brief look at the general architectural framework of the IoT which is divided into several layers shown in Figure 2.3. The IoT is not yet restricted to a specific architecture, as many vendors and applications adopt their layers to suit the needs of different smart environments [31].

- ❖ The sensing layer consists of physical objects of various shapes, sensors, and controllers (RFID tags, digital IDs, motors, infrared ZigBee, etc.). These elements provide information gathering, processing, storage, and transmission to the network layer [33][6].

- Network Layer is responsible for transmitting data from the perception layer's physical objects to the processing device over the internet using various technologies with a secure communication system (such as 5G, Zigbee, WiFi, LTE, Bluetooth, etc.). The connection can be wired or wireless and relies on the physical object or ingredients of The sensor. The data is collected and filtered through network gateways and then transit to the next middleware layer [32][6].



Figure(2.3): IoT basic layered architecture [34].

- The middleware layer functions are service management, which allows IoT devices that provide the same service to communicate with one another. In addition to storing information coming from the lower layer in the database for processing and thus facilitating decision making [32][6].

- The application layer manages IoT applications and is dependent on the data processed by the middleware layer. Furthermore, this layer relies on the characteristics of various IoT applications such as smart health, smart transportation, etc [32][6].

- The business layer creates a business model based on the information processed in the previous layer. It serves as the system's boss, overseeing business models, privacy, and application management [6][32].

2.3 IoT Applications

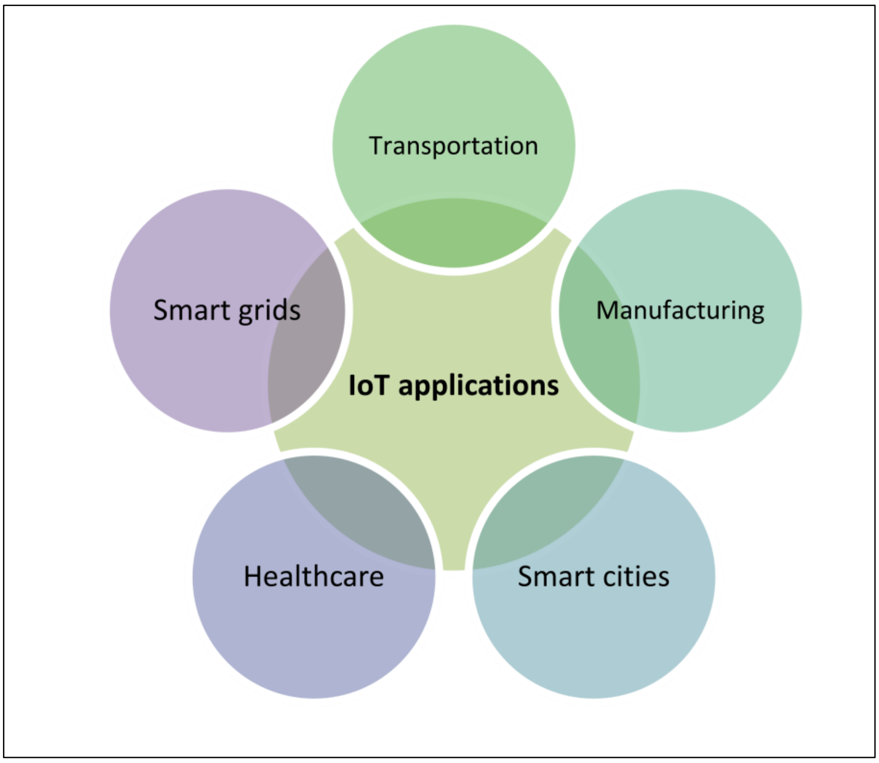
The IoT enables the advancement of a wide range of applications in various fields, including smart cities, healthcare, smart homes, in addition to other industrial applications. In this section, will highlight some key IoT applications, as shown in Figure (2.4).

1) Smart Grids

Electrical energy is a valuable resource with a high industrial value that plays an important role in economic growth. Nowadays, uses cutting-edge IT technology to maximize electricity generation by factoring in consumer demands across the entire electricity delivery line. This distribution line is driven by smart grid technology, also known as advanced metering infrastructure (AMI). It consists of an interconnected network installed between electricity production centers and end-users, with the primary purpose of coordinating electricity production with end customers' usage. One of the most appealing fields of IoT is smart grids. The key aim is to increase the efficiency of the final customer's experience while also optimizing the electricity supply [35].

2) Smart Cities

One of the most significant IoT implementations is smart cities, which is a modern evolving concept that seeks to improve the efficiency of public services and the quality of service provided to people. Sensors are installed in buildings, streets, smart vehicles, and other structures. To help control traffic, respond to conditions, lighting that matches the sun's direction; domestic incidents can be avoided with alarms, etc [36].



Figure(2.4): Internet of things applications.

3) Healthcare

In healthcare, IoT is used to monitor a patient's physiological conditions. Embedding sensors and actuators in patients' bodies for sensing and recording purposes can gather data directly from the patients and relay it to the physician. This technology has the ability to fully isolate the patient from the hospital's centralized structure while keeping constant communication with the doctor. Healthcare-based IoT solutions are currently one of the most promising innovations that have a significant effect on society. There is a lot of interest in using emerging IoT-based technology to track patients in real-time. Indeed, in France, the proportion of people over 60 hit about 24% of the population in 2015, which is expected to increase to 32% by 2060 [37].

4) Transportation Systems

Intelligent Transportation Systems (ITS) are the next generation of transportation systems that use embedded systems and networking technology to link people, highways, and intelligent vehicles. That will make transportation cleaner, greener, and more efficient by linking and distributing intelligent processors within vehicles as well as transportation infrastructure. Connected cars are gaining in popularity as a way to make driving more dependable, fun, and effective. In vehicular networks, there are three categories of communications: V2I (Vehicle to Infrastructure), V2V (Vehicle to Vehicle), and V2P (Vehicle to Pedestrian). However, a new method of connectivity called V2G (Vehicle to Grid) has recently appeared, with the primary purpose of ensuring electric vehicle charging focused on the energy of smart grid electricity distribution [37][38].

5) Manufacturing

IoT now plays a significant part in the industry. It is regarded as a promising approach for automating the manufacturing process and controlling the production chain. To build an intelligent manufacturing environment, the Industrial Internet of Things (IIoT) employs emerging technology such as Wireless Sensor Networks (WSN), big data, Machine-to-Machine (M2M) networking, and automation technologies. The key goal of the IIoT is to improve end-product productivity, performance, reliability, and power [39].

2.4 Cloud Computing and The IoT

IoT systems connect a massive number of sensors and devices to provide support and services and exchange huge amounts of data. Therefore, this large amount of data for analysis, management, and storage imposes certain requirements such as high-speed network capabilities, powerful processing, calculation, and mass storage. Cloud computing helps meet these requirements, including huge storage capacity with virtual capabilities to process large amounts of collected data in addition to configurable resources and high computational power for data created from smart environments based on the Internet. The integration to smart environments with cloud computing systems helped access data and smart things quickly and easily and managed them at any time and place, thus providing better services. Their challenges and solutions provided is discussed in Figure (2.5).

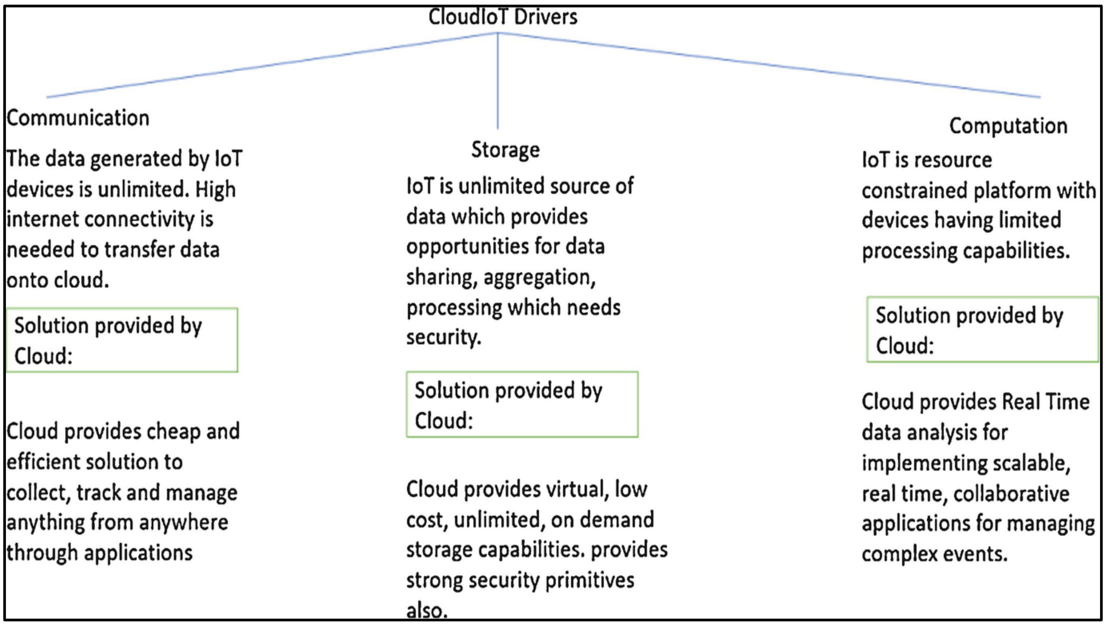

CloudIoT Drivers
Communication
The data generated by IoT devices is unlimited. High internet connectivity is needed to transfer data onto cloud.
Solution provided by Cloud:
Cloud provides cheap and efficient solution to collect, track and manage anything from anywhere through applications
Storage
IoT is unlimited source of data which provides opportunities for data sharing, aggregation, processing which needs security.
Solution provided by Cloud:
Cloud provides virtual, low cost, unlimited, on demand storage capabilities. provides strong security primitives also.
Computation
IoT is resource constrained platform with devices having limited processing capabilities.
Solution provided by Cloud:
Cloud provides Real Time data analysis for implementing scalable, real time, collaborative applications for managing complex events.

Figure (2.5): Cloud computing for IoT[40].

One of the most significant obstacles in using a cloud computing system is to meet the real-time requirements of many IoT applications as well as synchronize between various cloud vendors. vital requirement for meeting IoT needs in cloud computing platform is thinking about the necessary security restrictions. Therefore, the major factor obstructing the adoption of cloud computing by government institutions and companies is security challenges [6][41].

2.5 Security Issues in IoT

The ever-increasing number of network IoT-connected devices and the complexity because it relies on various technologies that do not share a global language and consist of different products and different companies, which has created more security holes associated with the service that affect the security of these devices. Where interoperability, mixing, and independent decision-making begin to embed potential complexity and vulnerability. To enhance security interoperability and functionality of all devices in the IoT system by creating a unified standard architecture for all manufacturers and vendors, as this is one of the important aspects of the security of the smart systems. Standardization enhances the security interoperability and functionality of all objects and sensors in the IoT system [6].

In smart IoT-based environments used in areas such as medicine and industry that do not have robust security systems, vulnerabilities will have severe impacts on and endanger applications and services. Security-related issues for any IoT device can be classified into the following types: confidentiality risks, privacy and authentication issues, physical threats, and data integrity issues [6]. They threaten the integrity of the messages and the QoS [24].

With an endless array of IoT applications, new security challenges will emerge, for example, industrial security concerns, Programmable Logic Controllers (PLCs), smart sensors, and robotic systems, which are usually integrated with storage infrastructure. IoT may include several hybrid systems, and providing protection and security controls for these systems is crucial. Endpoint security for IoT and how final nodes receive security patches or software updates without compromising functional safety and promptly [40].

2.6 The Security Challenges in IoT

The IoT faces many limitations related to IoT devices and components, i.e., its computational strength, resources, and even the inconsistency of devices have led to new security problems and challenges, as seen in Figure (2.6). Here are some of the challenges listed below [33].

2.6.1 Authentication

The perception layer includes many devices and objects; every object must be able to authenticate other nodes in the IoT. A universal authentication mechanism is required to authenticate devices in these environments. Due to the heterogeneous nature of IoT devices, the process is not simple and poses a great challenge [33]. Security must be considered while designing itself to be tamper-resistant, and required software updates must be made frequently [29].

2.6.2 Confidentiality

Within an IoT device, stored data must be protected and secured during transmission over wireless networks as confidentiality risks arise between IoT devices and gateways. By taking into account the level of security, power consumption, key size, and time required to implement algorithmic encryption while using it for secure process monitoring applications. So data is protected from unauthorized controls to not allow attackers to capture information that is transmitted over the Internet easily and to ensure targeted customers that the data are tamperproof [6][31].

2.6.3 Privacy

Data privacy is another serious part of security in IoT systems. Devices use various types of object identification techniques, and therefore each element has its identification card that carries personal information, location, and movement. Privacy must be imposed on personal information and restrictions placed on storing and disclosing information to others since any unauthorized access to the administration system threatens the privacy of IoT users' data [6][42].

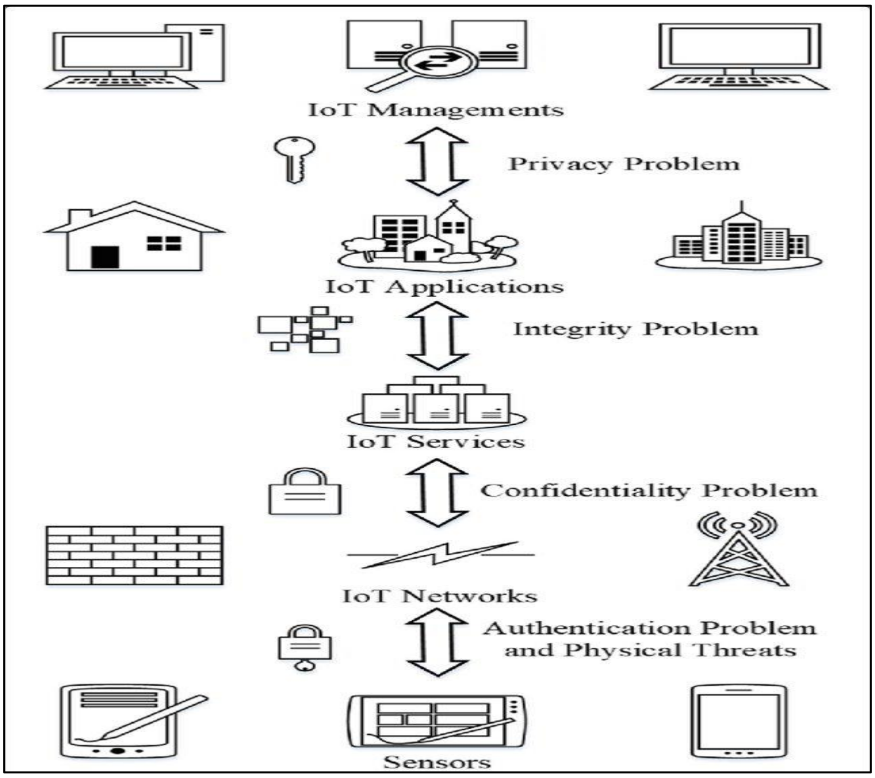


Figure (2.6): The security challenges in IoT layers[6].

2.6.4 Integrity

The fourth challenge includes the integrity of data during transmission over wireless networks, i.e., preserving it, ensuring accuracy and completeness, and ensuring that it is not changed or dropped over transmission. When spoofing, noise, and other attacks affect the system, data integrity problems arise. Hence it is necessary to secure plant information and secure process equipment from attackers who harm IoT applications [6][42].

2.7 Typical attacks in IoT

- **Packet dumping:** In this type of attack, the entire network is flooded, where the intruder can create a storm of deceptive packets or frequently duplicated legal packets. These attacks result in battery depletion and network node stores overflowing in addition to overloading channels that will cause some problems regarding implementation. However, in all cases, the system's goal was to deplete battery power and increase the traffic density, such as the DoS attack [43][14].
- **Vulnerabilities attacks:** Some of the popular gaps that systems possess and are exploited by the vulnerability attack. To mislead a protocol or application running under it, some distorted packets are sent to the target. The vulnerability allows an attacker to perform an unauthorized action (for example, accessing data without appropriate privileges) [43][14].
- **Blackhole attack:** It constitutes the most critical attack as the malicious node can pull whole packets by demand a new and deceptive path to the destination and then accept them without redirecting them to the destination. This attack leads to huge energy losses, congestion, and other problems, thus significantly affecting network performance [44][14].
- **Jamming:** It is one of the most destructive security attacks in the WSN-based IoT. The attack confuses and reduces the traffic in the network by blocking the channel, so a group of nodes do not generate traffic and become isolated due to the attack. The intruder transmits a signal and a stack of network operating frequencies in a way that reduces the signal-to-noise ratio to a level where the wireless nodes cannot operate [14]. Jamming attacks exhibit behavior with some differences depending on their type, such as reactive jamming attacks and deceptive jamming devices [40].

- **Selective redirection:** In this type of attack, the malicious nodes drop a portion of the packets by refusing to redirect them, ensuring that they will not spread any further and thus the integrity of the information degradation. Intrusion is quickly detected, and then the traffic density decreases if an intruder does not replace legal packages with fraudulent ones [14][45].
- **Sybil attack:** Sybil is defined as "a malicious device that illegally takes multiple identities." In this attack, the adversary obtains multiple identities and pretends to be a distinct user, and then tries to establish a relationship with an authentic user to be hacked. The attack could lead to two traffic-altering scenarios depending on the intruder's target: either the dummy nodes generate additional traffic or block the legal nodes from passing through. Sybil attacks also pose a significant threat to geo-routing protocols [14][45][31].
- **Sinkhole attack:** The compromised node attracts network traffic after announcing false routing information. Then uses it to initiate other attacks such as changing or dropping packets, selective redirecting, or identifying spoofing. Sensors left unattended for long periods in the network are vulnerable to sinkhole attacks [14][31].
- **Clone attack:** These types of attacks are very dangerous on IoT networks. An adversary can obtain confidential information of the nodes and insert a replica or clone of this information into the network. Replicated nodes can carry out a variety of attacks such as a black hole, wrong data injection, etc [14][46].
- **Wormhole attack:** is very strange and difficult to recognize. This attack is generally used with selective redirection or eavesdropping. The IoT connects both static and dynamic things; the connection may be wired or wireless and depends on geographic location ranging from wristwatches and refrigerators to vehicles. The attack allows an opponent to capture data, forward it to another node, and forward it from that node [14][31].

2.8 Architecture Layer Wise Threats

There are various kinds of security attacks against different layers of the IoT, some of which are briefed below:

2.8.1 Security thread in the Physical layer

The physical layer of IoT platforms is the layer where data is generated by objects, sensors, and actuators. Furthermore, sensors are the most insecure interface of IoT systems because they are the instruments that gather data directly and can easily be hacked by attacks, such as:

Eavesdropping: Attackers can eavesdrop and capture data during different stages of data movement or authentication in IoT applications since they often consist of multiple nodes publishing in an open medium. In order to gain some useful information, the attacker establishes a connection with both victims involved in a conversation, which think they're talking to each other explicitly, but they're actually infecting the conversation [34][47][31].

Battery drain attack: An attack that does not allow application from entry into sleep or energy-saving mode is executed by sending persistent trusted requests to implement the attack on IoT devices with limited resources [47].

Malign Data Injection: involves the attacker injecting a fake device into an IoT system that floods the wireless channel with fake messages or can sniff out wireless traffic to make the system inaccessible to regular users [47].

Node cloning: IoT devices consist of various low-power nodes such as motors and sensors; due to the lack of standardization, these nodes are subject to a diverse of attacks by opponents, i.e., node cloning can easily be performed or attempted to replace it with a malicious node. This can be done in the production and operational phases. It may seem that the new node is a portion of the system but is managed by the attacker, thus endangering the safety of the entire IoT application [47][34].

2.8.2 Security thread in the network layer

Attacks such as DoS and DDoS can cause excessive network congestion, resulting in connection failure. Here are some of the security issues encountered in the network layer.

Phishing site attack: the network layer of the IoT is very vulnerable to phishing Website attacks. They are attacks in which one can target many devices with minimal effort from the attacker. There is a potential of facing phishing sites while users visit web pages on the Internet as the IoT devices used by the user become vulnerable to attacks once the user account and password are hacked [34].

Access attack: is also considered an Advanced Persistent Threat (APT). In this kind, the opponent can access the system's network, and he can remain in it for a long period without being detected. Usually, the goal of APT is to theft data or valuable information instead of causing network harm. APT attacks demand a complex and progress process as well as a high degree of stealth during a cyber-attack and are generally supported by great organizations or countries. Stuxnet and Black Energy are some examples of APT [34][48].

DDoS / DoS attack: it aims to increase the chaos in the system network infrastructure by filling the target servers with undesirable traffic packets of massive size for the sake of capture and completely deplete the memory resources, which leads to disrupting the target server and thus disrupting the services for the real users. It is called a DDoS attack if the attacker uses many resources to damage the target server [34][48][31].

Data Transfer Attacks: There is a significant amount of data traffic among drives, sensors, the cloud, etc. Valuable data is stored and exchanged between IoT applications and thus is vulnerable to hacker attacks. Data stored in the cloud or on local servers carries security risks, but data transferred from one site to another is more amenable to cyber-attacks [34].

Routing Attacks: in these attacks, the malicious nodes attempt to redirect the routing paths through data transfer in the IoT application. One type of routing attack is a Sinkhole attack in which an opponent declares the shortest technical directional path and attracts a node to steer the c path through it [34].

2.8.3 Security Thread in Application Layer

There are many security issues related to IoT applications. Instead of focusing on security, application developers focus on efficiency and reliability in serving the product, and each application has a different authentication method. Therefore, it is challenging to guarantee authentication and privacy. Some threats in the applications layer include the following:

Malicious code: attackers generally choose a straightforward or the easiest way to storming a system or network by exploiting vulnerabilities. The first login method an attacker chooses is whether the system is vulnerable to malicious scripts and erroneous directives due to inappropriate code scans. A successful attack can lead to Internet paralysis in IoT and account hijacking [47][34].

Poor application security: among the consequences of a weak authentication and authorization mechanism are data tampering, unnecessary data disclosure, brute force, and escalated privileges. Moreover, IoT systems accessed via websites are at risk. An attacker can operate devices abnormally by entering the IoT application system [47].

Man in the middle attack (MITM): an attack aims to eavesdrop on the communication between devices because the Message Queuing Telemetry Transport (MQTT) protocol uses a subscription model for publishing to communicate between subscribers and clients using an MQTT broker. This helps separate publishing clients and subscribing clients from each other. By enabling the attacker to intercept communication between two nodes and control the middleman and become a man in the middle, the attacker can recognize and modify the data transmission on the network without the knowledge of the clients [34][48][49].

2.9 Intrusion Detection System in IoT

Intrusion detection is a security mechanism that detects the actions that hackers take against information systems and operate primarily in the network layer of the IoT. The purpose of intrusive actions is unauthorized access to a computer system. Foreign users attempting to obtain unauthorized access to device information when outside the target network and inner intruders with a certain level of legal access inside the network attempting to misuse unauthorized privileges by increasing their access privileges are the two types of intruders [6][50]. Typical IDS system consists of an analysis engine, reporting system, and sensors. A smart IoT-based environment requires a device that can run under a variety of conditions, including low computing capacity, quick response, and massive data processing. IDS systems can be categorized in IDS development strategies to Distributed, Centralized, or Hybrid.

In the distributed position, the nodes may also be responsible to monitoring their neighbors. This node is referred to as the guard, meaning that IoT devices may be in charge of inspecting other IoT devices. [50][51].

In the IDS central site, which is located at the centralized devices; for example, it is placed in the border router or the specified host. Where the IDS system located in a border switch can verify the packets that have been switched among the IoT devices and the network. However, analyzing the traffic that crosses the border router is insufficient to identify anomalies and has difficulty monitoring nodes during an attack threatening the part of the network. To take advantage of the strengths and avoid defects in central and distributed placement, a hybrid IDS mode combines both approaches [50]. There are many techniques used to detect intrusion, including: classification, mining data streams, frequent pattern mining and clustering [52].

2.10 Artificial Intelligence in IoT

As move towards the highly connected digital world and because the IoT collects a huge amount of data from these networked devices using lightweight, distributed intelligence and small sensors. Where anything and everything becomes smart as accidents and crimes can be predicted through the use of traffic data. Traffic produced by the IoT and many more. The data collected from the devices connected to the network is so large that the real-time and accuracy is a challenge (even if a sample is taken for processing) in dealing with them to obtain better insights and take corrective actions about it. This kind of real-time processing poses challenges on the IoT. It is conceivable that using traditional methods of dealing with such an enormous amount of data is a time-consuming process. Here, artificial intelligence methods can be used side by side and integrated into IoT data to address the overall problem and improve speed and accuracy in order to make smart decisions [29].

AI focuses on simulating intelligent dynamics behaviors that occur in nature and human style of thinking and is an active field of research that provides efficient and accurate solutions to real-world based problems. Integrating specific functions in the IoT allows to benefit from the existing technology assets of artificial intelligence and does not require the disposal of existing IoT systems to start over. Thus, maximizing the utilization of the current systems to rapidly build an IoT system. Sensors collect data, but data is only useful if it is meaningful and actionable.

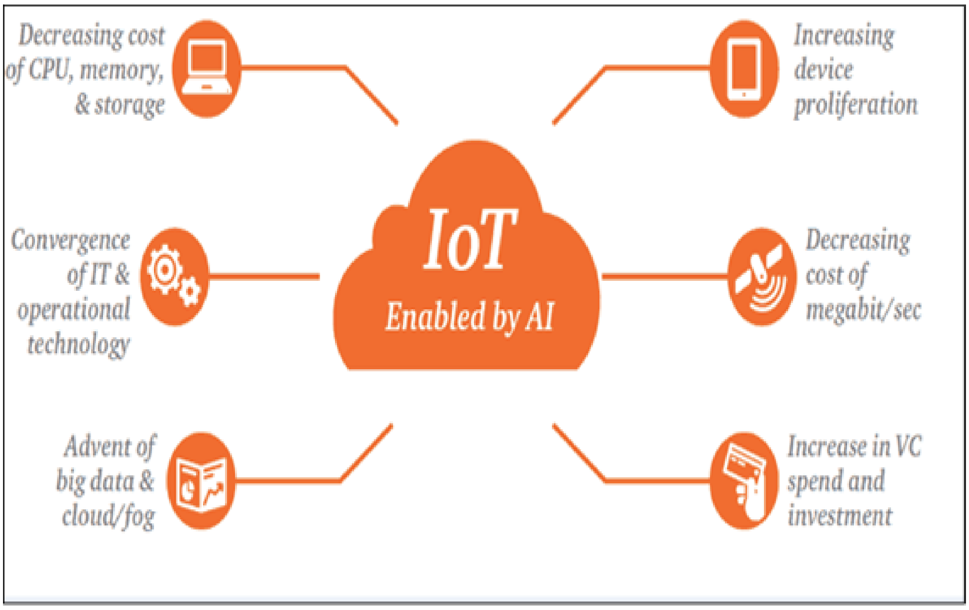

Decreasing cost of CPU, memory, & storage
Increasing device proliferation
IoT
Enabled by AI
Convergence of IT & operational technology
Decreasing cost of megabit/sec
Advent of big data & cloud/fog
Increase in VC spend and investment

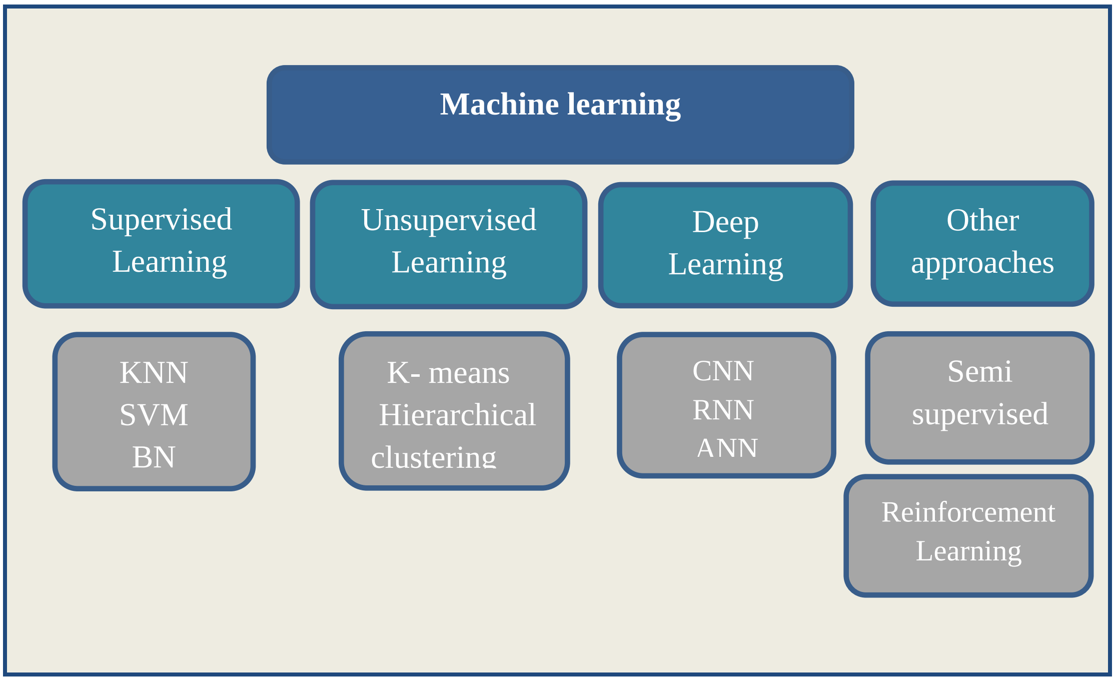
Figure (2.7): Drivers of IoT growth (IoT enabled by AI)[53].

For determining how to use data in productive ways, IoT and AI go hand in hand. IoT can inform AI, and AI can provide context and creativity to help people make better decisions.

One of the best practice would be to "enable intelligence" for the artificial intelligence of new IoT systems shown in Figure 2.7. This empowerment results in a series of "building blocks" that can be used to provide services. The building blocks are turned into security assets that are expressed in key security elements in the IoT architecture [5].

2.11 Machine Learning

Machine learning (ML) that was submitting in the year 1950, is a subfield of technology within artificial intelligence. With access to the enormous capabilities of data, it is becoming more prevalent. It is widely employed in industry and academia to solve different tasks initially aiming at robust and viable algorithms for many applications such as intrusion detection, spam detection and prediction, bioinformatics, and the smart grid, to name a few [29]. ML techniques are being exploited within IoT systems to take advantage of their potential and desire continuous improvement, where ML is a powerful tool for analyzing IoT data sets. ML technologies are designed to automatically take advantage in a previous experience at work in the future without explicit reprogramming. Current ML approaches are often categorized as supervised, unsupervised, or deep learning have been widely applied to improve network security, as seen in Figure 2.6 [40].



Figure(2.8): Classification of machine learning techniques.

2.11.1 Unsupervised Machine learning

Unsupervised learning algorithms work on datasets that don't have labeled answers and only have input data (X) with no output variables. In such algorithms, inferences are made by classifying unlabeled input data into groups called clusters based on the statistical properties of the data. Some of the most commonly used clustering algorithms are kmeans clustering, association rules, and hierarchical clustering [54].

2.11.2 Supervised Machine Learning

In supervised learning, the input and target output data are both labeled for classification. The learner is provided with two separate sets of data. Input variables (x) include the training set while the test set is the other being the output variable (Y), and an algorithm is used to learn the assignment function from input to output. Through the training set, the learner can "learn" patterns of classified cases, and can identify unlabeled samples in the test data set with the highest possible accuracy [40]. Supervised learning is distinguished by the fact that before fitting a model, the label/response variable Y is known. [54]. This section comprehensively and briefly explores using some of the different Supervised ML algorithms in the IoT and WSN security context to perform classification [40].

2.11.2.1 Bayesian Network

Bayesian networks are a graphical representation of probability relationships between a set of random variables that allows them to consider the underlying probabilistic classification mechanism. In the fields of machine learning, Bayesian statistics is gaining popularity. Their power comes from the relatively simple theoretical underpinning while being powerful and applicable in various contexts. Among the most commonly used Bayesian algorithms are Bayes Network and Naïve Bayes. In regression and classification problems, Bayesian algorithms use Bayes' theorem. It takes into account the conditional likelihood of a possible cause for a certain result [54]. The Bayes Theorem equation is:

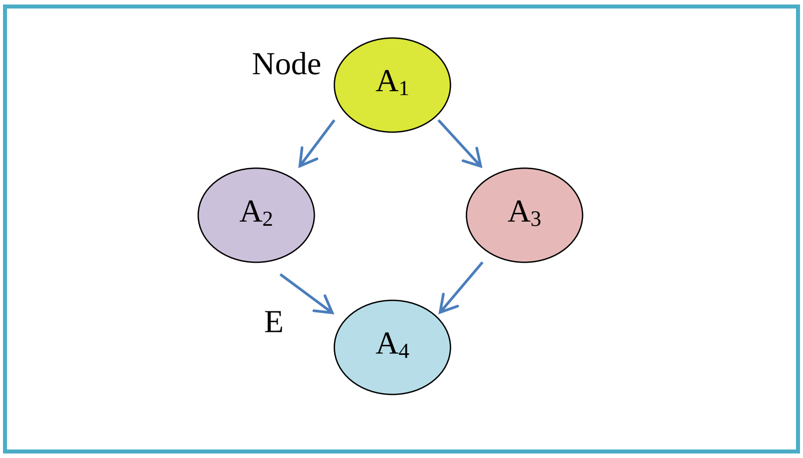
$$P\ (A\ |B) = \frac{\mathrm{P(A)*P(B\ |A)}}{P(B)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.1)}$$

Where

- P(A) and P(B) are the prior probabilities of A and B without regard for each other.
- P(A|B) is the conditional probability(the posterior probability), the probability of A given that B is true.
- P(B|A) is the prior probability of B, given that A is true.

1) Bayes Net Classifier

Bayes networks are a graphical representation of probability relationships between a set of random variables. It uses in classification has received much attention. Bayes' algorithms predict the class depending on the probability of belonging to that class. Figure (2.9) shows Bayes network consists of a set of discrete random variables where each variable may take values from a finite set represented by = {A1, A2,. . ., AN} and a set of directed edge, E, between the variables that make up a directed acyclic graph (DAG) G that encodes a common probability distribution across A. where the product of conditional distributions of each variable given its parents represents a joint distribution of variables.



Figure(2.9): The structure of bayes net classifier

If there is direct relationship from the A1 variable to the A2 variable, A1will be the origin of variable A2 representing the conditional dependency between A1 and A2. Each variable is independent of non-descendants, given its parental value in G. This coding independence in G reduces the number of parameters required to characterize the co-distribution.

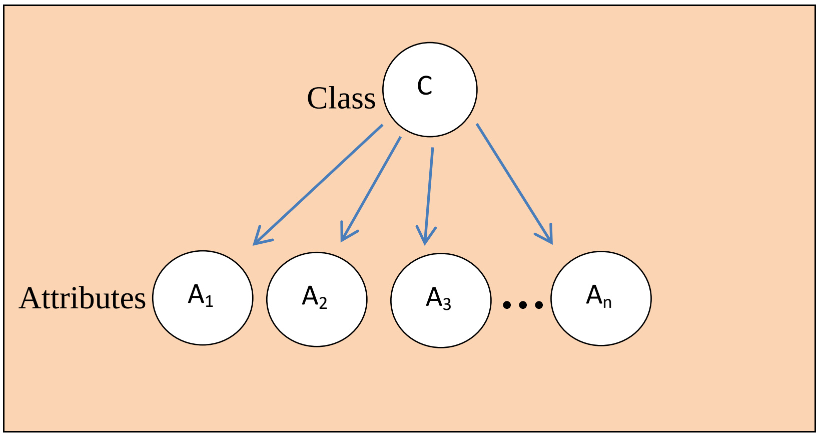
The joint distribution P(V) in a Bayesian network is the product of all conditional distributions defined in the Bayesian network, such as

$$P(A_1, A_2, \ldots, A_N) = \prod_{i=1}^{N} P(Ai/Pai) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2.2)}$$

Where Pa (Ai) refers to the parents of Ai in G and P(Ai/Pai) represents a conditional probability distribution. The conditional distribution of each variable has a parameterized shape that can be learned by estimating the maximum likelihood [55][56].

2) Naïve Bayes Classifier

A simple application of Bayes' theorem is to the case of classification. Naïve Bayes classifier uses conditional probability to determine the likelihood of an event based on the presumption that the predictive variables are conditionally independent given the class. Figure (2.10) shows the structure of Naïve Bayes.



Figure(2.10): The structure of naïve bayes[57].

It simplifies learning by assuming that features are independent of a given class. Let's say [56][58]:

- (A1, A2,...A n) is a vector of features of X that are conditionally independent.

- B= (B1,B2) its two specific classes.
- (P1,P2) probabilities of X, which belong to class B1 and B2,
- The class of instance X is corresponding to max(p1,p2)
- Therefore The Naive Bayes classifier predicts the class with the maximum posterior probability as can be identified as the following equation:

$$B_X = argmax \frac{P\ (A1,A\ 2,.......A\ n\ \ |Bj\ )\prod_{i=1}^{N}\ \ P\ (Bj)}{P(\ X)} \qquad .... (2.3)$$

P(X) does not affect the solving of the maximum value because it is usually considered constant. So Equation (2.3) is equal to Equation (2.4):

$$B_X = argmax\ P\ (A1, A\ 2\ , \dots\dots . A\ n\ \ |Bj\ )\prod_{i=1}^{N}\quad P\ (Bj) \dots\dots\dots\dots\dots\dots\dots (2.4)$$

The vector X is features that are independent of each other so can simplify the calculation of equation (2.4) and can be defined as follows:

B$_X$ = argmax $\Sigma P\ (xi\ |Bj\ )P\ (Bj)$ .... (2.5)

Parameters of P(B$_j$) and P(X$_i$ |B$_j$) are defined as in Equation (2.6) and Equation (2.7)

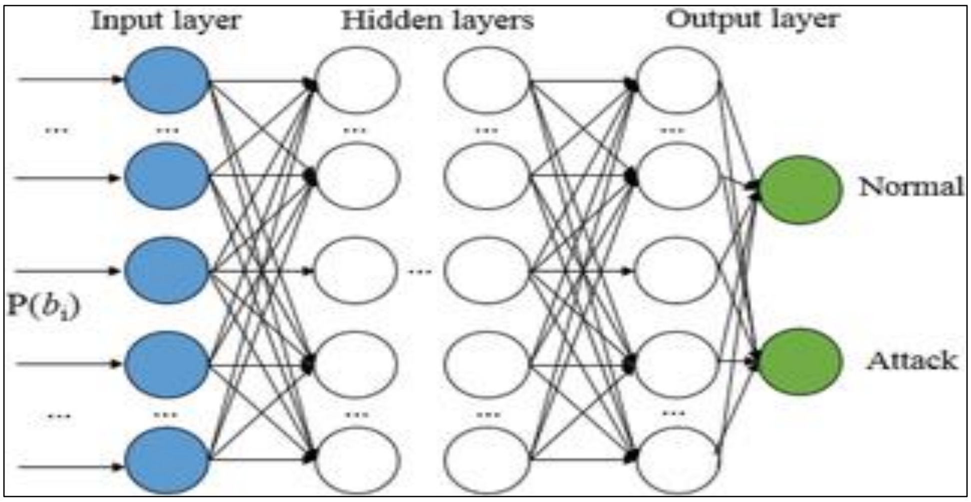
$$P\ (\mathrm{B}_j) = \frac{the\ number\ of\ traffic\ in\ class\ Bj}{the\ number\ of\ all\ traffic\ in\ the\ training\ set} \qquad \dots\dots\dots\dots\dots\dots\dots\dots\dots (2.6)$$

Where B$_j$ = type of class which is either normal or attack.

$$P\ (x_i\ |\mathrm{B}_j\ ) = \frac{the\ number\ of\ feature\ xi\ appearing\ in\ Bj}{t\ he\ total\ number\ of\ all\ features\ appearing\ in\ Bj} \qquad .... (2.7)$$

2.12 Deep Neural Networks

Deep learning is one of the most important and recent achievements in the era of machine learning approaches, as it can be said that it has become the most important attraction for research. DL offers solutions that could only be imagined in science fiction movies only a decade ago. It's a particular branch of machine learning that can handle patterns and complex things in huge data sets that began in 2006. Figure (2.11) shows DNN networks consist of an input layer, multiple processing layers (hidden layers), and an output layer that classify a normal and an attack packet. Hidden layers generates a non-linear output from the input layer data. It can have a strong learning ability and be improved further with increasing depth - or evenly with the number of hidden layers. DL function is inspired by the signal-processing ability of human brain neurons [59]. There are various types of DL supervised learning models, including Recurrent Neural Networks (RNNs), Feed-forward (or Artificial) Neural Networks (ANN), and CNN. The next part focuses on DL's primary tool Deep (conventional) CNN's, briefly explaining their basic features, blocks, and architectures [54].



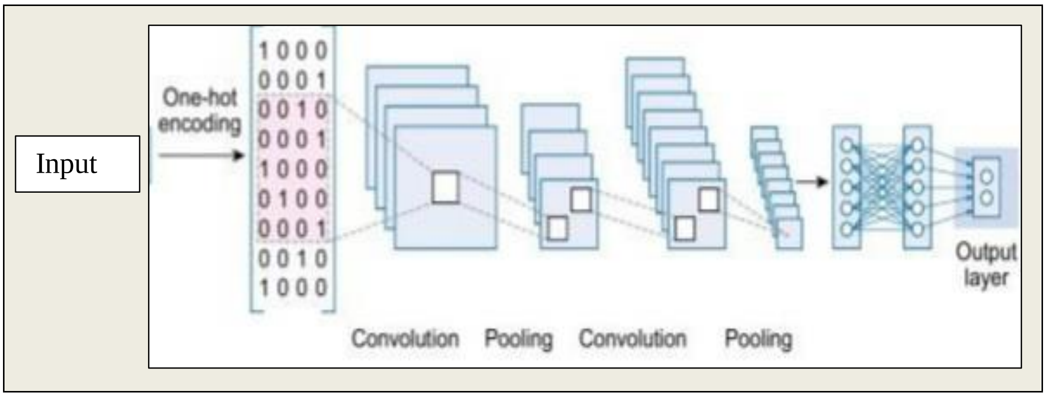
Figure(2.11): Deep neural network structure in network security[60].

2.12.1 Convolutional Neural Networks

Convolutional Neural Networks have recently brought about an important development in deep learning. It greatly improves classification performance and has high accuracy rates because it extracts "optimized" features directly from the raw data of the problem at hand. Deep CNNs are designed and built specifically for 2D signals such as images and video clips. It has been developed as a modified version to deals with 1D signals and is useful for specific applications, therefore, preferred over its 2D counterparts. The compact 1D CNNs are well suited for real-time, low-cost applications, especially on mobile devices, and attractive for complex engineering applications. CNN's use of 1-D signal processing application naturally requires adequate 1D to 2D conversion, and its application was not straightforward for 1D signal signals, especially when data is scarce [59].

2.12.2 Convolutional Neural Networks Architecture

The CNN architecture has three levels where learning accuracy is measured in terms of the number of filters/cells used in a network as the smaller amount of detail increases. Generally, CNN architecture mainly consists of two processes, convolution is the most important building block of a CNN, and it converts input data into output through a set of filters or a core. Another process called clustering (pooling) is to take a sub-sample to cut out irrelevant data and remove data glitches that lead to reduced computational overhead, memory usage, and the number of parameters (thus reducing the risk of over-processing) and improving learning for the next layers. The torsion product is further processed by the activation function, and then samples are taken down by aggregation [61].

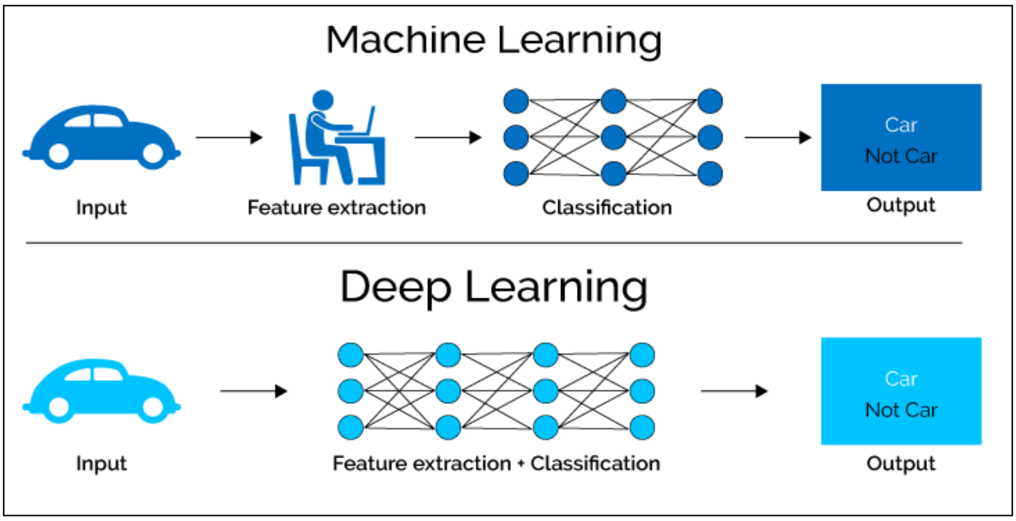


Figure(2.12): Convolutional neural networks architecture[62].

Figure (2.12) shows the architecture of CNN, the data feed to the model as input, and then the preprocessing phase start followed by convolutional and pooling layer and then fully connected layers.

2.13 Machine Learning versus Deep Learning

ML methodology is widely used in many fields especially identifying several types of attacks where ML can assist the network administrator in implementing the required procedures to avoid interferences [63]. Most of the familiar techniques in ML usually focus on trait engineering, selection and lie in shallow learning that cannot efficiently solve a classification problem with massive data. The DL approach has the ability to extract improved representations from a dataset. It can generate more efficient prototypes with less human intervention, which makes DNN more efficient and robust compared to other methods. In comparison, traditional ML methods require high computational complexity, some pre-processing steps, and then use fixed, handcrafted features that are suboptimal. Figure 2.9 shows the basic variance between ML and DL, where conventional ML involves manual feature chosen and, on the contrary, DL uses automatic feature selection. Besides the highest performance levels, another significant benefit of DLs is that, unlike traditional ANNs, they can integrate feature extraction and classification tasks in a single object process that can be completed in less time.

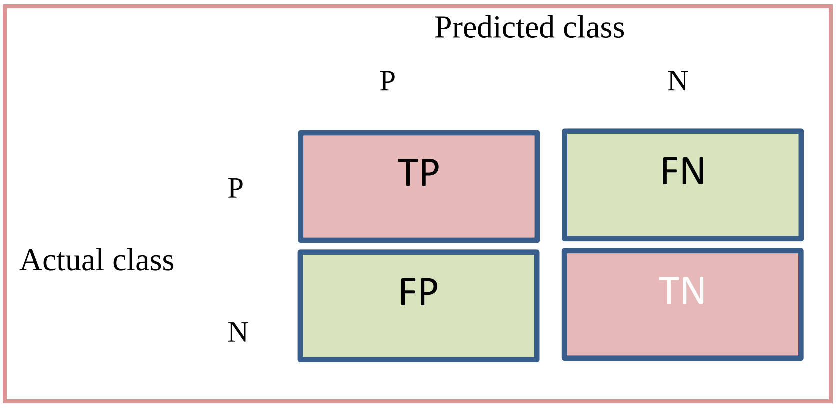


Figure(2.13): Basic variance between ML and DL [64]

DL can be considered as the foundation of both representation learning and ML together. In ML, “representational learning” or “feature learning” is a series of techniques that make a system able to automatically learning representations desired to find out features from a training data set. Deep Learning can also be used to analyze IoT data that is heterogeneous and multi-modal. For IoT devices, which are typically linked for longer periods of time, traditional machine learning algorithms struggle to produce long-term performance [54][59].

2.14 Evaluation Methods

There are various measures that can be used to evaluate the effectiveness of classification algorithms, such as accuracy, recall, precision, and F1-measure. The calculation of these measures depends on the computing confusion matrix. The Figure (2.10) shows is a matrix used to describe the performance of a classification approach or "classifier" by test data such as UNSW-NB15 datasets [65][66].



		Predicted class	
		P	N
Actual class	P	TP	FN
	N	FP	TN

Figure(2.14): Confusion matrix.

- **True Positive (TP):** positive instances that are correctly classified.
- **False Negative (FN):** positive instances incorrectly classified as negative.
- **False Positive (FP):** negative instances incorrectly classified as positive.
- **True Negative (TN):** negative instances that are correctly classified.

The evaluation metrics were defined based on the confusion matrix, as shown in equations (2.8) to (2.11).

❖ **Precision (Positive predictive value):** can be defined as the number of right positive predictions divided by the entire number of positive predictions. It is also called the positive predictive value (PPV).

Precision = TP/TP+FP …………………………………………………….. (2.8)

❖ **Recall (Sensitivity or True positive rate)** is the number of TP divided by the number of TP and the number of FN.

Recall = TP/TP+FN ……………………………………………………. (2.9)

- **F1- measure** conveys the balance between the precision and the recall.

F1- measure = 2*((precision*recall) / (precision + recall)) ………………. (2.10)

- **False alarm rate** is the proportional fraction of false positives to the actual normal class size [65].

FAR= FP / FP+TN .... (2.11)

2.15 Summary

This chapter mentioned the theoretical foundations of the specified security system. It also explains some techniques of machine learning, including BN and NB, used in this research. The chapter then illustrates the architecture of CNN and clarifies deep learning benefit over machine learning. The system's implementation and design will be discussed in the next chapter.

CHAPTER THREE

Design and Implementation

Chapter Three

Design and Implementation

3.1 Introduction

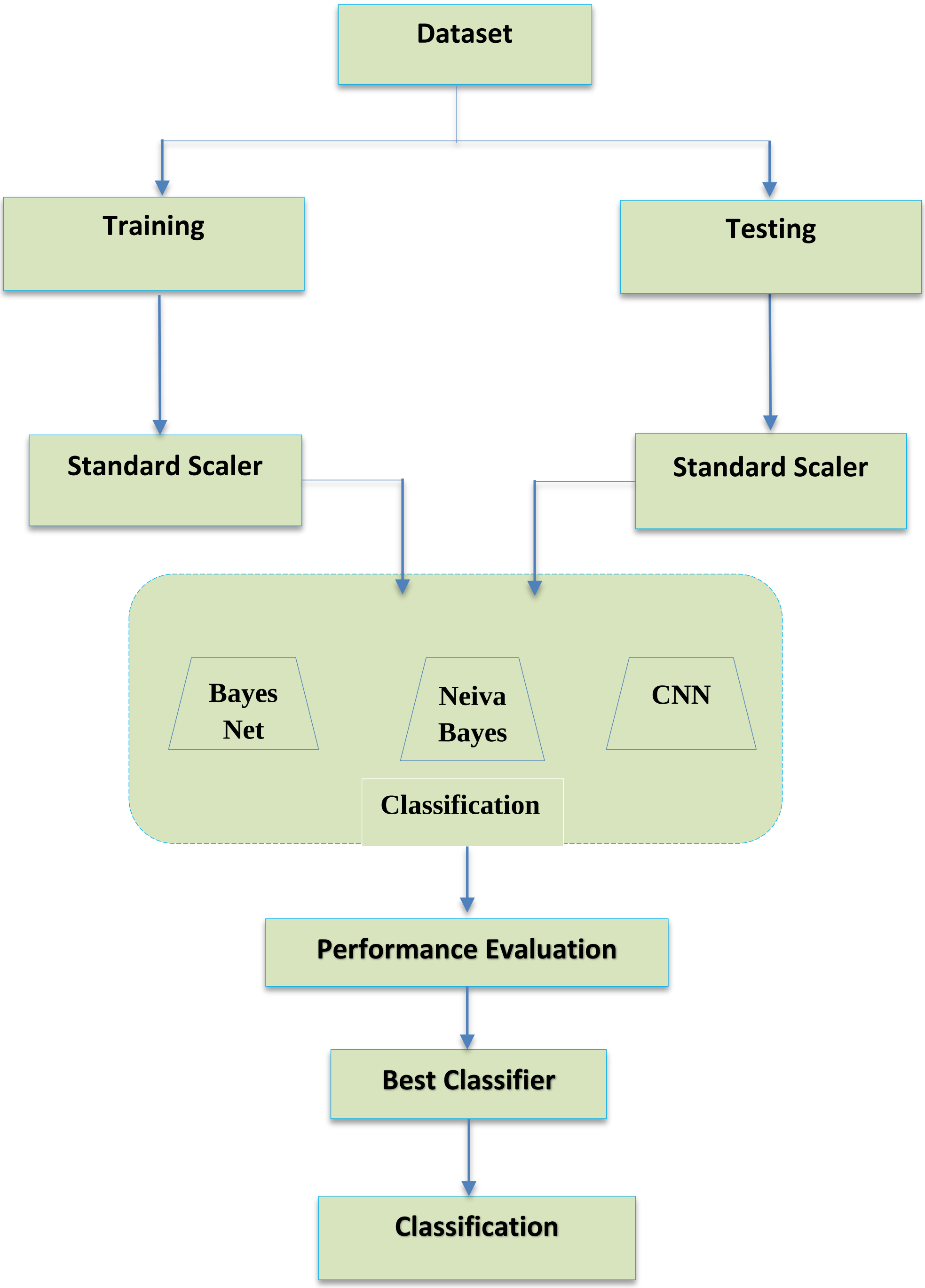
This chapter discusses the research procedure for the project and the main phases of designing an intelligent approach to detect network layer attacks in the IoT architecture. The essential steps of the practical aspect are described as well as the methodology for data processing. The use of supervised ML algorithms for classification such as Naive Bayes and Bayes Net for comparison purposes are presented. It also explains the methods will took to develop the proposed CNN classification model.

3.2 The Main Structure of The Proposed Approach

This section explores the architecture characteristics and learning setup to design the approach and the dataset used in building an intelligent model for detecting attacks in IoT networks. Adaptive, multi-layer, lightweight, distributive, and learning from the experience are all important characteristics for IoT solutions.

Three main experiments were conducted, as this allows us to perform a comprehensive analysis. The results of a study was implemented on a standard dataset UNSW-NB15 which contains information about “normal” and “attacks” connection types logged in an IoT network. Figure 3.1 depicts a general flowchart of intrusion detection system.

1. First, the UNSW-NB15 was divided into group training and testing.
2. Then, pre-process the dataset. Rigorous data analysis was performed and prepared in the required format before applying these data as an input to the model. The training and testing set data samples are standardized and label_banrize to fit into a compatible data structure.



Figure(3.1): General diagram of intrusion detection system .

3. Train the two ML classification algorithms, NB and BN, and the proposed CNN model in DL on the same training data set.
4. The fourth stage of this detection approach contains a test of algorithms used with normal and attack behaviors. All experiments were evaluated based on prediction accuracy.

3.3 UNSW-NB15 Dataset

The evaluation of the IoT intrusion detection system is related to some extent to the dataset used and is one of the main research difficulties in designing the model. It is important to study the dataset carefully as part of a machine learning approach because many of the datasets collected may contain a large amount of redundant data resulting in unreliable evaluation results that differ between different datasets. Different benchmark datasets are used for IDS by many researchers, including KDD99, NSL-KDD, and UNSW-NB15.

The proposed approach uses the UNSW-NB15 dataset as a streak dataset, which is very suitable for training repetitive neural networks. According to the literature, UNSW-NB15 is the most recently released and powerful dataset for intrusion detection and IoT research since, unlike previous structured datasets, it shows current threat trends as well as recent usual traffic patterns. The UNSW-NB15 dataset is the most recent set created by the Australian Center for Cyber security (ACCS) in 2015. It includes many features that have been built by IXIA perfecStorm in the laboratory that generates normal activities and offensive behaviors. The experiments used the dataset UNSW-NB15_train file, which contains 82,332 records split into 57632 on train and 24700 on a test. Each row in the dataset is categorized as normal and attack logs. UNSW-NB15 contains nine different attacks, which means several classifications (or classes), among which five types are often found in IoT attacks (denial of service, worms, backdoor, reconnaissance, and analysis) as well as Exploits, Fuzzers, General, and Shellcode. It has 42 features; classification is based on the following three categories: basic, content, and time.

3.4 Preprocessing Phase

Before feeding the model with data, both the training and testing sets must be preprocessed. This stage deals with converting the input data into a data structure that the Algorithms’ framework accepts. An important step in preprocessing in ML algorithms is feature standardization. The Preprocessing library standardizes data to broaden the range of inputs because input data may have varied distributions by different means and standard derivatives, affecting learning efficiency. Remove outliers (feature scaling) to improve trained predictive performance and to prevent slow training time. Standardization involves re-measuring features and removing outliers by normalizing all feature values in the dataset so that they have the characteristics of a standard normal distribution with a mean of zero and a standard deviation of one as it is often classifiers. The Standard Scaler function choice in our model.

Label_Binarizer facilitates by using the inverse transform method at prediction time one assigns the class to which the corresponding model has given the most confidence. An appropriate number of samples was used in the UNSW-NB15 dataset, where the UNSW_NB15_training-set file was split with the help of the train_test_split function. Before building the CNN model, reconstruct the train and tested input into one dimension (1D) using the reshape function to bring the data structure into line with our input format. The reshape method is used to execute the train reconfiguration, and it is passed as an argument to this method along with the other corresponding parameter values. When this method receives the test as an argument, it reconfigures the test into a format that can be used to test the model.

3.5 Classification with Machine Learning Algorithms

This part of the thesis explains and shows the implementation of two of the most famous classification algorithms of ML. The BN algorithm, and NB, are implemented on the same dataset.

3.5.1 Bayes Net Classifier

Bayes Net is working on Bayes' basic theory. By calculating the conditional probability of each node, a Bayesian network is build. The training set used the as an input to the classifier. Bayes Net learns in several stages: structural learning, parameter learning, and building probability distribution tables for each node in the network. The task of structural learning of Bayesian networks is to determine which DAG topology represents the best underlying co-distribution of the data. A Bayes net algorithm is a powerful tool for data visualization and inference under conditions of uncertainty. Algorithm (3.1) illustrates the Bayes Net classifier.

Algorithm (3.1) BN Classifier

Input: Dataset: //70% from the dataset for training and 30% from the dataset for testing

Output: A class of testing dataset based on Bayes Net which is two sets normal or attack.

Steps:

```
1. Read the training dataset T;
2. Calculate the probability of fi in each class; // to compute the probabilities for a variable, the values of its parents must already be known.
Until the probability of all predictor variables (fi, f2, f3,.., fn) has been calculated.
3. Calculate the likelihood for each class;
4. Get the greatest likelihood;
5. learned A conditional distribution for each variable given it parent
6. Join distribution
7. Posterior distribution  inferred.
End
```

3.5.2 Naïve Bayes Classifier

This section describes the implementation of NB with the train and testing phases where it promises as very simple and shows a good probability model. This model is implemented by taking a list of features for each category as an input to the classifier, and each document in this chapter is represented as a vector for one dimension that is fed to the classifier to train the features. The testing phase starts from entering the unlabeled documents and performing preprocessing algorithm to obtain the symbol set. Then, this set of symbols is fed to the prediction pane to classify unlabeled documents with the correct category prediction. Algorithm (3.2) illustrates the Naïve Bayes classifier.

Algorithm (3.2) NB Classifier

```
Input: Dataset:    //70% from the dataset for training and 30% from the dataset
for testing
Output:  Classification based on naïve Bayes which is two sets normal or attack
Begin:
Training phase
Step1: for i=1to the number of the classes
Start:
Calculate the prior possibilities of every one of the classes with the use
End
Step2: Compute predictor's prior possibility.

/* Testing */
  Step3: Calculate the posterior probabilities P(C|d) of the class (i.e., the target).
for i=1 to the number of classes.
Step4: Obtain the class that has the maximal posterior probabilities.
End
```

3.6 Classification with Deep Learning Algorithms

Existing network intrusion detection systems based on ML are ineffective in detecting novel assaults in IoT. This part describes the proposed DL algorithm and the steps to implement it to build a more robust DL model that detects attacks in IoT.

3.6.1 CNN Classifier Architecture

In these experiments, CNN model choses because the convolutional layer is one of the main classes of attack detection and has the advantage of significantly reducing training time as the CNN architecture was used on the UNSW-NB15 datasets. The dataset has passed through multi-steps in preprocessing phase in order to enter the CNN classification algorithm layers and make it gives desired and accurate results. Figure (3.2) describes the detection approach layers architecture by CNN model . Algorithm (3.3) illustrates the CNN classifier.

Algorithm 3: CNN Algorithm

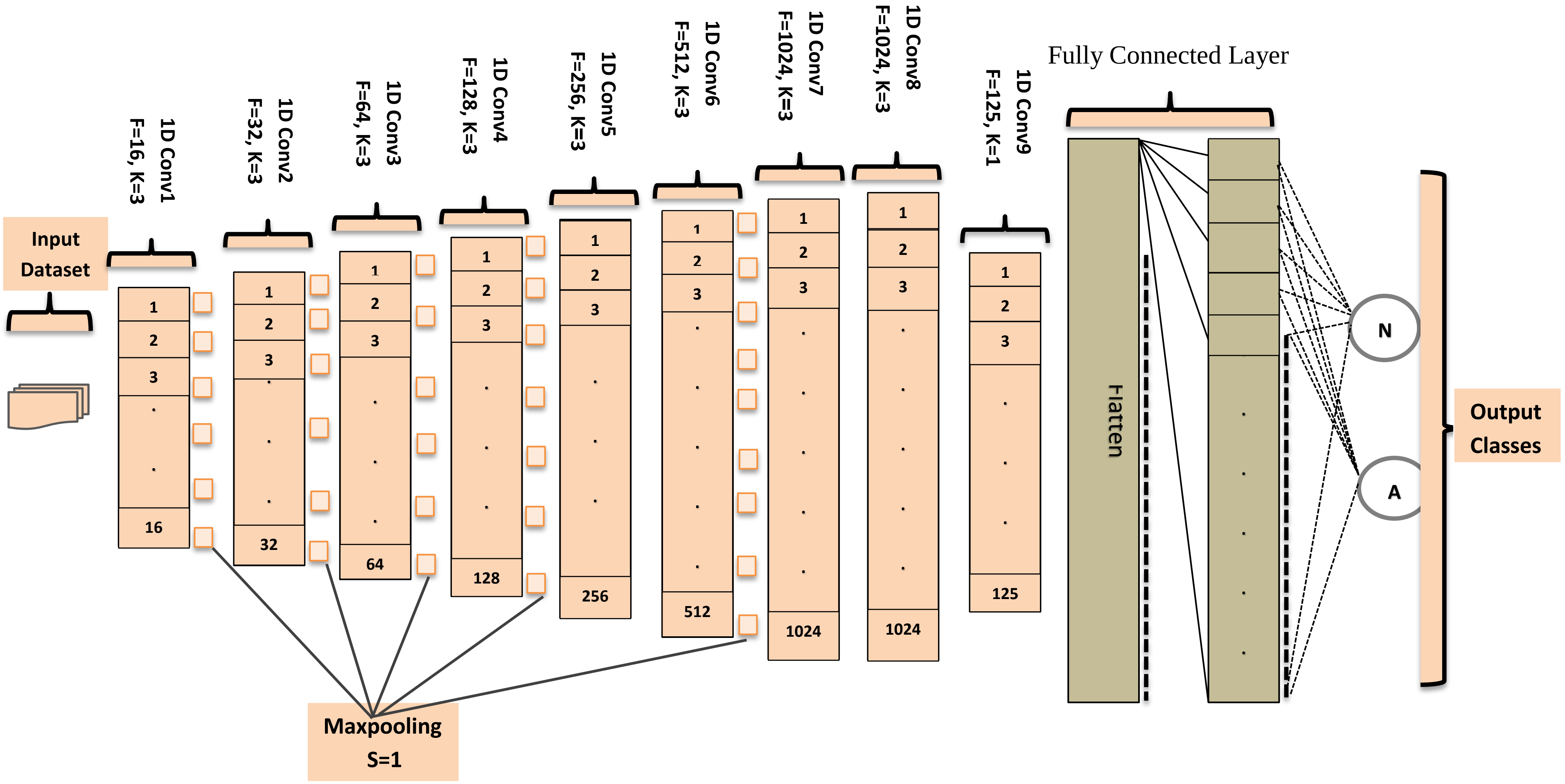
Input: Dataset // features and class label

output: Classification results of CNN model. It is either attack or normal.

Steps:

```
1.  Read the dataset
2: Splitting dataset for learning    // define X train, Y train and X test, Y test
3: Preprocessing the data using StandardScaler
4: Constriction CNN sequential model:  // set the CNN layer
    –  Convolution 1D_1 with 16 filter, activation function = ReLU
           Max_pooling layer with 16 filter
    –  Convolution 1D_2 with 32 filter, activation function = ReLU
           Max_pooling layer with 32 filter
    –  Convolution 1D_3 with 64 filter, activation function = ReLU
           Max_pooling layer with 64 filter
    –  Convolution 1D_4 with 128 filter, activation function = ReLU
           Max_pooling layer with 128 filter
    –  Convolution 1D_5 with 256 filter, activation function = ReLU
    –  Convolution 1D_6 with 512 filter, activation function = ReLU
           Max_pooling layer with 512 filter
    –  Convolution 1D_7 with 1024 filter, activation function = ReLU
    –  Convolution 1D_8 with 1042 filter, activation function = ReLU
    –  Convolution 1D_9 with 125 filter, activation function = ReLU
    –  Flatten output
    –  Fully connected layer, activation function = softmax
5: Classification results
6: Classification Report exports
```

End



Figure(3.2): The Proposed 1D CNN model

Depending on the applications, warping is performed on one-dimensional or multi-dimensional data to break into the network. The 1D wrap is easy to use in design because the data elements in a dataset often come from network packets and can be easily displayed in vectors. The CNN architecture includes a combination of four types of multiple computation layers through which the input data passes, including ConvNet with Filters (Kernals), Clustering, and Fully Connected Layers (FC) for traffic classification with probabilistic of attack or normal. The attained CNN layers and their parameters will be explained in this thesis as follows:

1D convolution layer: the CNN architecture contains nine Conv1Ds that extract features from the package. 1D CNNs have advanced performance levels and are relatively easier to train with low computational requirements, so they are suitable for real-time hardware implementation.

MaxPooling1D: CNN architecture also contains five MaxPooling1D modules that implement dimension reduction to reduce entry size and reduce computation time. This new layer is added after the convolutional layer precisely after applying nonlinearity (e.g., ReLU). By adding a maximum clustering layer between the convolutional layers, the Kernel extracts the maximum value for the region it contains. The area is changed in wide steps. When using the "valid" padding option, it has the form:

output_shape = (input_shape - pool_size + 1) / strides) ……………………..…..(3.1)

Dense layer: is a single completely connected (dense) and it where every neuron in a layer receives input from all the neurons in the previous layer, and thus it is densely connected by neurons in the network layer. It is the most popular and most used layer. The dense layer performs the process below on the input and returns the output

Output = activation (point (input, kernel) + bias) …………………………….(3.2)

Rectified linear activation function (ReLU): The most commonly used activation functions are mathematical equations that define the output of a neural network. This function can be represented as follows:

$$f(x) = \begin{cases} 0 \; if \; x < 0 \\ x \; if \; x \geq 0 \end{cases} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3.3)$$

It chooses as a function that is easy to compute and gives quick learning because it draws directly to one if it is positive or zeroes if it is negative. In addition to it is more effective for the binary classification problem, it use as the activation function for the convolution layer in the proposed approach.

Softmax function: is the activation function of the final output layer in a neural network. The penultimate layer in the multi-layered neural network produces results of real value that are not easily scaled and can be difficult to work with. The Softmax function is applied to dilute the output and pass it to the output layer. The input values can be positive or negative or zero or greater than one. But softmax converts its vector of real K values into a vector of real K values to output values between 0 and 1 which sum to one so that they can be interpreted on They are possibilities.

Stride: is a parameter of a neural network filter that indicates how many pixels will move over the input matrix at each convolutional step. The value is set to 1, which is the default as the filter will move one unit at a time.

Flattening: involves converting the output matrix of the previous layers into a single column that is then fed to the neural network (fully connected layer) for processing.

Table(3.1) shows the CNN network layer parameters in the proposed model, which is 16 layers.

Table(3.1): The CNN layers parameters.

Parameter	**Value**
Convolution layer	9 layers
Max Pooling layer	5 layers
Dense layer	1

Flatten layer	1
Optimizer	ADAM
Activation function	ReLU and Softmax
Learning rate	0.001
Epoch	100
Batch Size	64

The Table showing learning rate value reach to 0.001 which is the best from through experiments and using different values such as 0.01 , 0.0001, 0.00001 all of them give 100% precision values but with a high percentage of errors. The batch size value is either 32 , 64 or 100 depending on the computer specifications.

Padding: The process of adding zeros to the input matrix symmetrically to preserve the dimensions of the output as in the input when processed by the CNN core. VALID Padding used that means no padding and assumes all dimensions are valid so that the input matrix is completely covered by a filter and the step you specify. Dimensions are reduced from 40 to 26 before applying to flatten. When using valid, the filter window stays at a valid position inside the input map, so output size shrinks by filter_size - 1. No padding occurs.

3.7 Summary

This chapter explores the architecture characteristics & learning setup to design approach and the dataset used in building an intelligent model for detecting attacks in IoT networks. A set of machine learning algorithms and the proposed CNN model are used to detecting attacks. It also uses a set of standards to test the performance of algorithms and the proposed model to measure recognition rate and the approach in general. The experimental results will be presented in the next chapter with the system analysis and comparison between ML classifier, BN, and NB and the proposed CNN.

CHAPTER FOUR

Results and Discussion

Chapter Four

Results and Discussion

4.1 Introduction

This chapter provides an evaluation for the performance of a proposed approach to building a high-fidelity deep learning model for solving the problem of detecting attacks on IoT devices and subsequently protecting them. The detection process is used to distinguish between malicious and normal behaviors, where the classifier classifies each sample as “normal” or “attack.” The results of the system's process, such as the pre-processing stage where its methods are mentioned in Chapter three, and measuring the accuracy, security, and performance, will be described. The efficiency of deep learning-based IDS and NB, BN algorithm used for comparative purposes, is estimated using evaluation metrics for classification criteria such as precision, recall, and F-measures

Finally, this chapter presents the results of all experiments conduct within the framework of proposed approach to compare the performance of IDS systems for each ML and DNN architecture in the adversarial environment.

4.2 Software and Tools

In our experiments, java8 under the NetBeans (IDE 8.2) integrated development environment was used, and Python version 3.6.5 under IDL IDE. The experiments are conducted in the below-mentioned environment:

- CPU: Intel ® Core ™ i5.
- OS: Windows 10.
- RAM: 8GB.
- Programming Language: Python, java.

4.3 Evaluation Measurement of Classification Performance

There are various measure that can be used to evaluate the classification approaches. To describe efficiency and detection, such as Precision, Recall, and f1- measure for the system, a confusion matrix would be utilized. The confusion matrix is a two-dimensional matrix that depicts the relationship between detected and real values.

Precision: can be defined as the proportion of the predicted positive cases that were correct. In other words, it measures the degree to which the attacks were correctly predicted, as calculated using Equation (2.8).

Recall: can be defined as the proportion of positive cases that were correctly identified. In other words, it is the ratio of accurately expected attack cases to the attack class's real scale. It is calculated using Equation (2.9).

F- measure: a measure of the accuracy of a test that can be considered as the harmonic mean of recall and precision, which is giving in Equation (2.10).

4.4 Experimental Results of Bayes Net Classifier

In the IDS approach, Bayes Net was used for training the features and testing them to compare the obtained results to the results attained using the CNN DL algorithm. The implementation of the network system with 42 input features of the dataset. BN successfully identifies the samples into normal and attack activities and generates a satisfactory precision value of 89.60%. The approach can be evaluated using performance metrics, which are precision, recall, and F1-measure. Table and Figure (4.1) shows the performance measures of the Bayes Net classifier.

Table (4.1): The performance metric for the bayes net classifier

Metric	Bayes Net
Precision (%)	89.60
Recall (%)	89.51
F- measure(%)	89.52

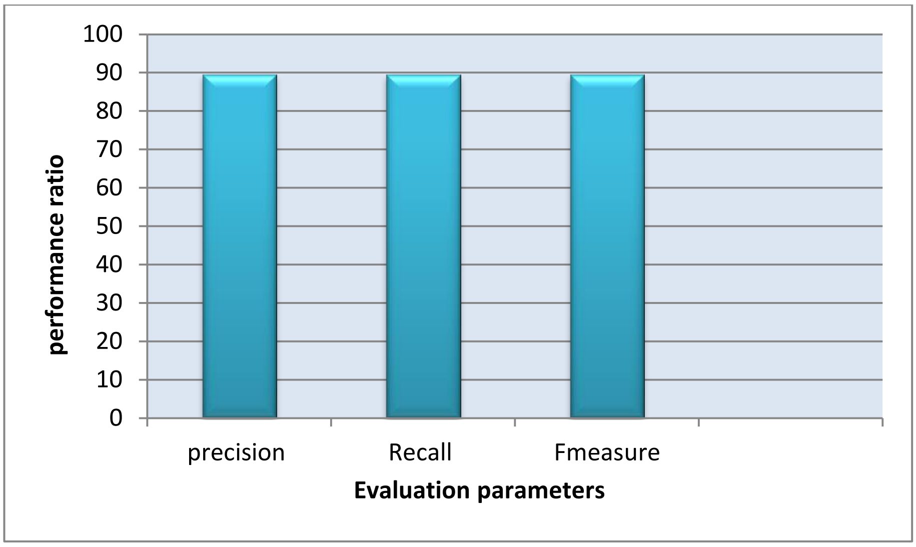


Figure (4.1) :Performance measures for bayes net.

From the results showing BN classifier reach a good precision value is 89.60 with convergent values to recall and F-measure 89.51 , 89.52 respectively due to using cross validation that take the data as groups.

4.5 Experimental Results of Naïve Bayes Classifier

Like the BN algorithm mentioned above, Naïve Bayes used the same features and the same training and testing dataset. The generated results are used to compare with the performance of the CNN model. A classifier is capable of classifying benign and attack traffic on UNSW-NB15 data sets. Table and Figure(4.2) shows the performance of the NB-based IDS model.

Table (4.2): The performance metric for the Naïve Bayes classifier

Metric	Naïve Bayes
Precision (%)	94.30
Recall (%)	94.29
F- measure(%)	94.29

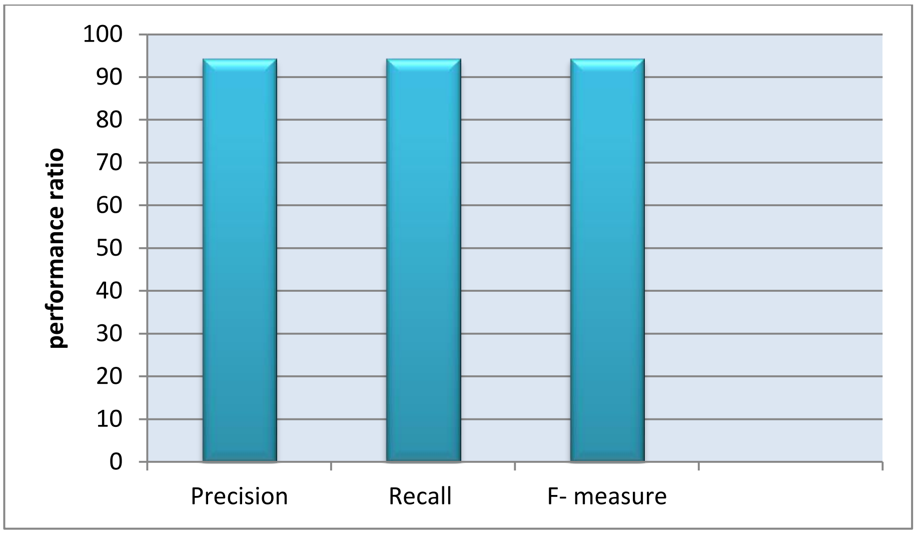


Figure (4.2): Performance measures for naïve bayes.

Figure (4.2) shows the result of implementing an NB classifier that has given a very good level of precision which was 94.30%. Also, a very good recall and f-measure rate reached 94.29%. The convergent values to recall and F-measure due to using cross validation that take the data as groups.

4.6 Classification Phase Results Based on The Proposed CNN

In this section, reports the performance of the proposed CNN model. Like the other algorithms mentioned above, CNN used the same features and the same training and testing dataset. The proposed CNN classification algorithm for identifying normal and attack traffic patterns have a set of layers shown in Table (4.3) which presents detailed information on each layer in the proposed model.

Table (4.3): CNN layers detailed information

Type	Filter	Parameter
Convolution 1D	16	64
Max pooling 1D	16	-
Convolution 1D	32	1568
Max pooling 1D	32	-
Convolution 1D	64	6208
Max pooling 1D	64	-
Convolution 1D	128	24704
Max pooling 1D	128	-
Convolution 1D	256	98560
Convolution 1D	512	393728
Max pooling 1D	512	-

Convolution 1D	1024	1573888
Convolution 1D	1024	3146752
Convolution 1D	125	128125
Flatten 1D	3250	-
Dense 1D	2	6502

In addition to the preceding table that describes the specific information about the CNN layers, the total number of parameters is 5,380,099. The model shows performance and capability to learn normal and attack traffic on the UNSW-NB15 dataset with high Precision. Table(4.4) and figure(4.3) show the performance of CNN -based IDS model.

Table (4.4): The Performance metric for the CNN model

Metric	CNN
Precision (%)	100%
Recall (%)	45%
F- measure	62%
False alarm rate	0%

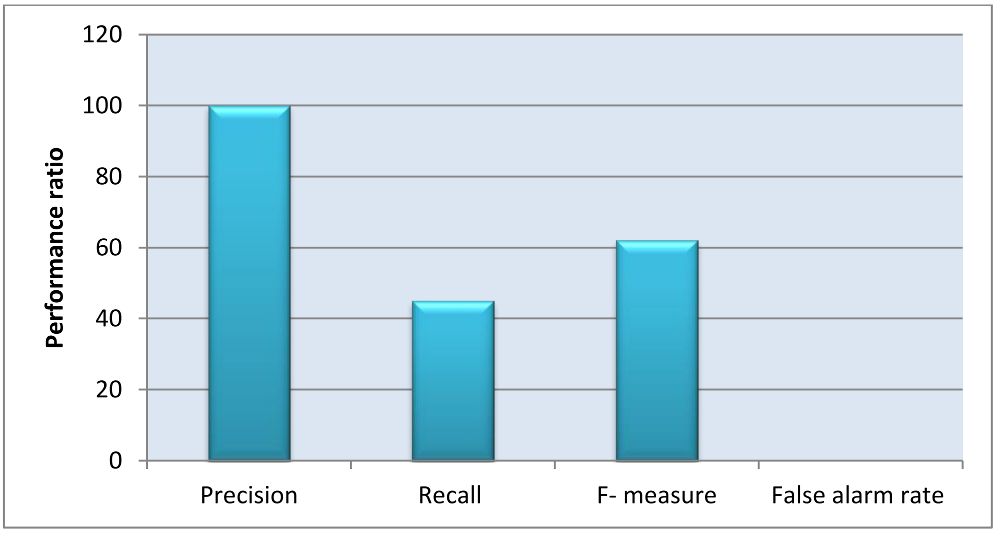


Figure (4.3): Performance measures for CNN model.

Table (4.8) illustrates the results which show that the proposed CNN has given a very excellent level of precision which was 100% successfully identifies the samples into normal and attack activities. Also, a satisfactory recall rate reached 45%, and a medium f_measure 63% because the imbalance between normal and attack traffic in dataset.

4.7 Comparison Between The Performance of IDS Systems For BN, NB, and CNN Algorithms

The intrusion detection process is a binary classification problem where the system implementing the process classifies individual samples as either “attack” or “normal.” To analyze the model performance of the IDS that use BN, NB, and CNN in the adversarial environments on UNSW-NB15 datasets. Table and Figure 4.4 below illustrates a practical comparison between ML Algorithms (BN and NB) in term of precision, recall and F-measure that concluded from experiments results. It should be noted that the protection of the approach relied on precision.

Table (4.5): Comparison between the ML classifiers.

Metric	Bayes Net	Naïve Bayes
Precision (%)	89.60	94.30
False alarm rate (%)	10.30	5.7

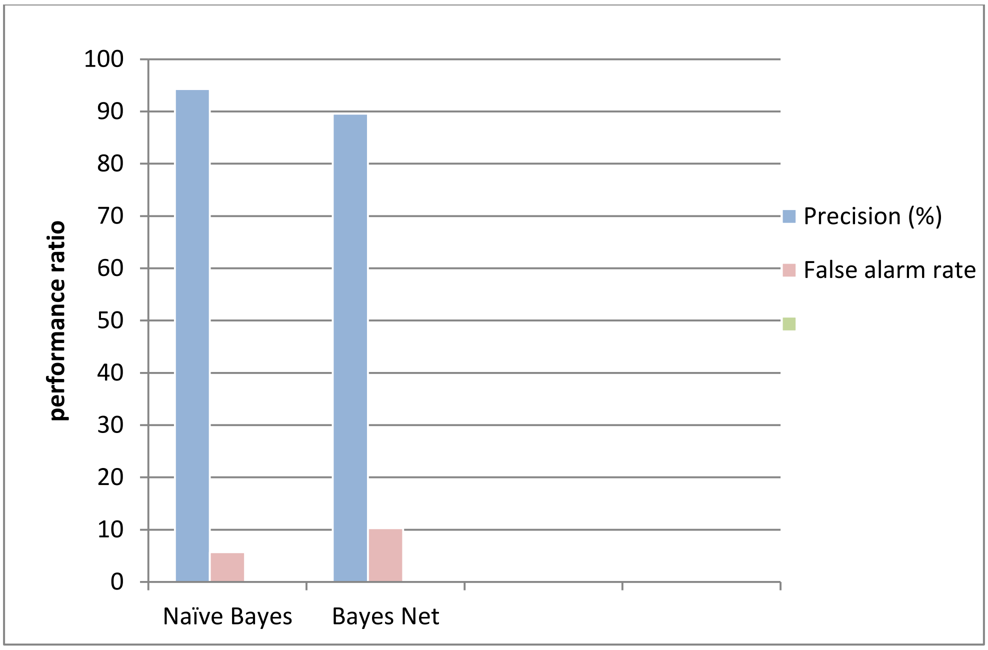


Figure (4.4): Comparison between the ML classifiers

Table (4.6): Comparison between the ML and DL classifiers models

Metric	Bayes Net	Naïve Bayes	CNN
Precision (%)	89.60	94.30	100
Recall (%)	89.51	94.29	45
F measure(%)	89.52	94.29	62
False alarm rate	10.30	5.7	0

Table 4.5 shows the BN classifier recorded the lowest percentage is 89%.On the other hand, the results of the NB classifier reach 94% on most samples for the UNSW-NB15 dataset. The NB performance indicates that the classifier is able to classify most attacks but with less reliability in identifying the attacking traffic.

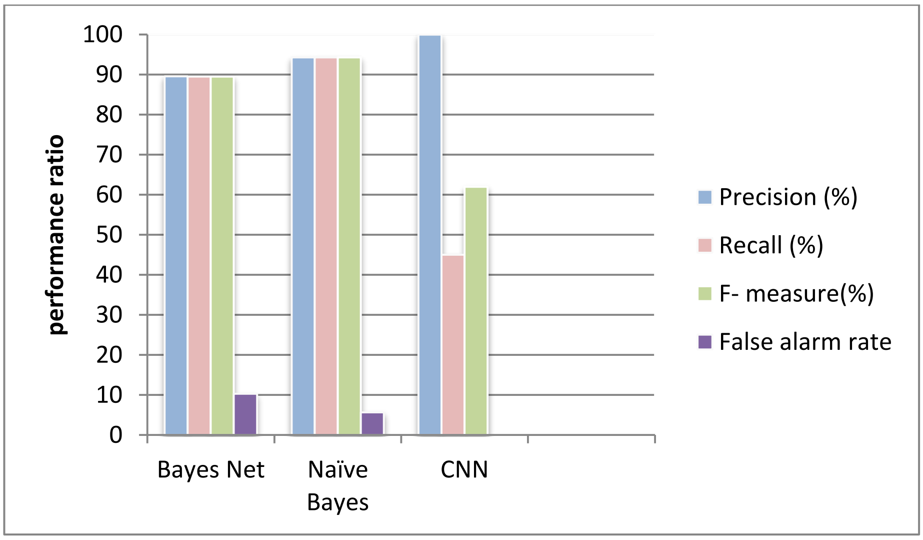


Figure (4.5): Comparison between the ML and DL classifiers models

This is an acceptable result, but CNN was proposed to have a more efficient and protected system. Comparing them in terms of precision, recall, and F-measure is shown in Figure(4.5) and Table (4.6). As noted above, the precision of the Bayes net and naïve Bayes classifier was convergent results. Though there remain no major fluctuations in the precision, recall, and f1 value, but the relatively weak performance reliability in identifying attack.

The result obtained by applying CNN recorded the highest precision percentage in the approach reach 100% on a dataset. The system performance was improved and became more secure by having much higher reliability in identifying attack traffic, reducing the FAR in which threatens the security of the users. It is worth noting that because of the cross-validation used in classical machine learning algorithms, the recall value was high by the imbalanced UNSW-NB15 dataset, as is expected. While in the CNN algorithm, the recall was a low value because the data are imbalanced and validation Cross-validation cannot be used because it greatly multiplies the training time.

The proposed approach proves that the adversarial training-based CNN is considered a reliable defense technique against different adversarial attacks on UNSW-NB15 datasets.

Here is a comparison among the experimental results of the proposed IDS and the experimental results of other works as shown in table (4.7).

Table (4.7): comparison between the proposed approach and past works

Authors	**Technique**	**Dataset**	**Precision %**
Kaiyuan Jiang et al.[67]	CNN	UNSW-NB15	81.01
Kaiyuan Jiang et al.[67]	CNN -BiLSTM	UNSW-NB15	82.63
Liu Zhiqiang et al.[15]	FNN	UNSW-NB15	99.5
Hongpo Zhang et al.[68]	SGM-CNN	UNSW-NB15	99.74
Vinayakumar et al.[69]	DNN	UNSW-NB15	97.9
The proposed approach	CNN	UNSW-NB15	100

The researchers mentioned in the previous table have achieved good precision rates by implementing the mentioned techniques in their security systems. However, notice that the performance of the CNN algorithm implemented in the proposed approach is better than the others, It also has lower computation overhead than the others.

4.8 Summary

In this chapter, evaluating the performance of classification algorithms in the attack recognition approach in the IoT is presented. Three performance measures were calculated, which are precision, recall and f-measure. The user interface system that categorizes the traffic into two types, either attack or normal, is explained. The proposed approach results were presented and compared with the performance of the NB and BN classifier. in next chapter will explain the conclusion and future works.

CHAPTER FIVE

Conclusion and Future Works

Chapter Five

Conclusion and Future Works

5.1 Conclusion

In this thesis, an intrusion detection approach is proposed to provide authentication and security for the IoT. The main thrust behind the system design in this research work is discovering intrusions in the IoT against various attacks. A deep learning approach was specifically adopted for the CNN network for approach development. In light of the results obtained, the study concluded the following :

1. The use of ML algorithms in the experiment for the purposes of comparison, where a Naïve Bayes Classifier obtained a precision rate of 94%, and then Bayes Net Classifier of 89%.
2. To improve the accuracy of the results, an IDS approach is proposed using the CNN algorithm in DL. The approach has the ability to detect many different types of attacks with 100% precision in effective attack recognition and normal sampling with a low FAR ratio.
3. The proposed CNN approach guarantees robustness, as demonstrated in the experimental results in identifying attacks against the IoT network.
4. The DL networks generally have better performance than the classical ML algorithms; the DL networks have a higher precision rate and a lower FAR than classical ML methods.

4.2 Future Works

The primary purpose of this thesis work was to discover intrusions into the IoT. To achieve the goal, the CNN deep learning network has been adopted. This research shows that DL can effectively deal with attacks in the IoT. There are several ideas for the future expansions of this work as mentioned below:

- Test the proposed intrusion detection approach by other methodologies employing DL algorithms.
- Use another dataset for evaluating the proposed approach. Expands the IoT attack dataset in the future by including more IoT attack types and real IoT network traffic.

Reference

[1] Chopra, K., K. Gupta, and A. Lambora. Future Internet: The Internet of Things-A Literature Review. *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Prespectives Prospect. Com. 2019*:135–139. (2019).

[2] Zhou, J., Z. Cao, X. Dong, and V. V. Security and Privacy for Cloud-Based IoT: Challenges, Countermeasures, and Future Directions Jun Zhou, Zhenfu Cao, Xiaolei Dong, and Athanasios V. Vasilakos The. *IEEE Commun. Mag.*(January):26–33. (2017).

[3] Number of IoT devices 2015-2025 | Statista. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/

[4] Weng, Z., R. Liu, and X. Li. A New Secure Model for IoT Devices. (Itoec):518–522. (2020).

[5] Köse, U. An Artificial Intelligence Perspective on Ensuring Cyber-Assurance for the Internet of Things. *Cyber Assur. Internet Things*:249–256. (2016).

[6] Elrawy, M.F., A.I. Awad, and H.F.A. Hamed. Intrusion detection systems for IoT-based smart environments: a survey. *J. Cloud Comput.* **7**(1):1–20. (2018).

[7] Venkatraman, S., and B. Surendiran. Adaptive hybrid intrusion detection system for crowd sourced multimedia internet of things systems. 79:3993–4010. (2019).

[8] Shukla, P. ML-IDS: A Machine Learning Approach to Detect Wormhole Attacks in Internet of Things Prachi. *2017 Intell. Syst. Conf. IntelliSys 2017* 2018-Janua(September):(2017).

[9] Liu, Y., Y. Kuang, Y. Xiao, and G. Xu. SDN-Based Data Transfer Security for Internet of Things. *IEEE Internet Things J.* 5(1):257–268. (2018).

[10] Wazid, M., A.K. Das, V. Odelu, N. Kumar, M. Conti, and M. Jo. Design of Secure User Authenticated Key Management Protocol for Generic IoT Networks. *IEEE Internet Things J.* 5(1):269–282. (2018).

[11] Mehmood, A., M. Mukherjee, S.H. Ahmed, H. Song, and K.M. Malik. NBC-MAIDS: Naïve Bayesian classification technique in multi-agent system-enriched IDS for securing IoT against DDoS attacks. *J. Supercomput.* 74(10):5156–5170. (2018).

[12] Elhoseny, M., K. Shankar, S.K. Lakshmanaprabu, A. Maseleno, and N. Arunkumar. Hybrid optimization with cryptography encryption for medical image security in Internet of Things. *Neural Comput. Appl.* 32(15):10979–10993. (2020).

[13] Alsamiri, J., and K. Alsubhi. Internet of things cyber attacks detection using machine learning. *Int. J. Adv. Comput. Sci. Appl.* 10(12):627–634. (2019).

[14] Jan, S.U., S. Ahmed, V. Shakhov, and I. Koo. Toward a Lightweight Intrusion Detection System for the Internet of Things. *IEEE Access* **7**(c):42450–42471. (2019).

[15] Zhiqiang, L., G. Mohi-Ud-Din, L. Bing, L. Jianchao, Z. Ye, and L. Zhijun. Modeling Network Intrusion Detection System Using Feed-Forward Neural Network Using UNSW-NB15 Dataset. *Proc. 2019 7th Int. Conf. Smart Energy Grid Eng. SEGE 2019*:299–303. (2019).

[16] Song, H., H. Liang, L. Liu, J. Ma, and X. Huang. A Hybrid Data Security System of Internet of Things. 269–273. (2019).

[17] Fenanir, S., F. Semchedine, and A. Baadache. A machine learning-based lightweight intrusion detection system for the internet of things. *Rev. d'Intelligence Artif.* 33(3):203–211. (2019).

[18] Thamilarasu, G., and S. Chawla. Towards deep-learning-driven intrusion detection for the internet of things. 19(9):(2019).

[19] Ferrag, M.A., L. Maglaras, A. Ahmim, M. Derdour, and H. Janicke. RDTIDS: Rules and decision tree-based intrusion detection system for internet-of-things networks. *Futur. Internet* 12(3):1–14. (2020).

[20] Arul Anitha, A., and L. Arockiam. ANNIDS: Artificial neural network based intrusion detection system for internet of things. *Int. J. Innov. Technol. Explor. Eng.* 8(11):2583–2588. (2019).

[21] Choudhary, S., and N. Kesswani. Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets using Deep Learning in IoT. *Procedia Comput. Sci.* 167(2019):1561–1573. (2020).

[22] Malliga, S., S. Darsniya, and P.S. Nandhini. A network intrusion detection system for IoT using machine learning and deep learning approaches. *Int. J. Adv. Sci. Technol.* 29(3 Special Issue):1017–1023. (2020).

[23] Aldhaheri, S., D. Alghazzawi, L. Cheng, B. Alzahrani, and A. Al-Barakati. DeepDCA: Novel network-based detection of iot attacks using artificial immune system. *Appl. Sci.* 10(6):(2020).

[24] Mao, B., Y. Kawamoto, and N. Kato. AI-based Joint Optimization of QoS and Security for 6G Energy Harvesting Internet of Things. 4662(c):1–11. (2020).

[25] Abdullah, Z., G. Chen, M.A.M. Abdullah, and J.A. Chambers. Enhanced Secrecy Performance of Multihop IoT Networks with Cooperative Hybrid-Duplex Jamming. *IEEE Trans. Inf. Forensics Secur.* 16(c):161–172. (2021).

[26] Ahmad, M., 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.* 2021(1):(2021).

[27] Rahman, M.A., A.T. Asyhari, O.W. Wen, H. Ajra, Y. Ahmed, and F. Anwar. Effective combining of feature selection techniques for machine learning-enabled IoT intrusion detection. *Multimed. Tools Appl.*:1–19. (2021).

[28] Serpanos, D., and M. Wolf. *Internet-of-things (IoT) systems: Architectures, algorithms, methodologies*. 2018.

[29] Tripathy, B.K., and J. Anuradha. *Internet of Things (IoT) Technologies, Applications, Challenges, and Solutions*. 2018.

[30] Yang, Y., L. Wu, G. Yin, L. Li, and H. Zhao. A Survey on Security and Privacy Issues in Internet-of-Things. *IEEE Internet Things J.* 4(5):1250–1258. (2017).

[31] Fei Hu. *Security and Privacy in Internet of Things (IoTs) Models, Algorithms, and Implementations*. CRC Press, 2016.

[32] Datta, P., and B. Sharma. A survey on IoT architectures, protocols, security and smart city based applications. *8th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2017*:(2017).

[33] Ahmad, M., T. Younis, M.A. Habib, R. Ashraf, and S.H. Ahmed. A review of current security issues in internet of things. *EAI/Springer Innov. Commun. Comput.*:11–23. (2019).

[34] Hassija, V., V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access* 7:82721–82743. (2019).

[35] Liu, J., Y. Xiao, S. Li, W. Liang, and C.L.P. Chen. Cyber security and privacy issues in smart grids. *IEEE Commun. Surv. Tutorials* 14(4):981–997. (2012).

[36] Zanella, A., N. Bui, A. Castellani, L. Vangelista, and M. Zorzi. Internet of things for smart cities. *IEEE Internet Things J.* 1(1):22–32. (2014).

[37] Kouicem, D.E., A. Bouabdallah, and H. Lakhlef. Internet of things security: A top-down survey. *Comput. Networks* 141:199–221. (2018).

[38] Mario Gerla, Eun-Kyu Lee, Giovanni Pau, and U.L. Internet of Vehicles : From Intelligent Grid to Autonomous Cars and Vehicular Clouds The Genesis of IOT. 241–246. (2014).

[39] Sadeghi, A.-R., C. Wachsmann, and M. Waidner. Security and Privacy Challenges in Industrial Internet of Things. *Proc. 2nd Int. Conf. Trends Electron. Informatics, ICOEI 2018*:454–460. (2015).

[40] Alam, M., K. Shakil, and S. Khan. *Internet of Things (IoT) Concepts and Applications*. 2020.

[41] Naveen, S., and M.R. Kounte. Key Technologies and challenges in IoT Edge Computing. *2019 Third Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud)*:61–65. (2019).

[42] Prasath, J.S., U. Ramachandraiah, S. Prabhuraj, and G. Muthukumaran. Internet of things based hybrid cryptography for process data security. *J. Math. Comput. Sci.* 10(6):2208–2232. (2020).

[43] Manda, S., and N. Nalini. Denial-of-Service or Flooding Attack in IoT Routing. 118(19):29–42. (2018).

[44] Lakshmi, S., E.A. Mary Anita, and J. Jenefa. Detection and prevention of black hole attacks in vehicular ad hoc networks. *Int. J. Innov. Technol. Explor. Eng.* 8(7):1253–1257. (2019).

[45] Bysani, L.K., and A.K. Turuk. A survey on selective forwarding attack in wireless sensor networks. *2011 Int. Conf. Devices Commun. ICDeCom 2011 - Proc.*:(2011).

[46] Meghana, S., and R. Srinath. A Novel Mechanism for Clone Attack Detection in Hybrid IoT Devices. (May):2194–2198. (2019).

[47] Iqbal, W., H. Abbas, M. Daneshmand, B. Rauf, and Y.A. Bangash. An In-Depth Analysis of IoT Security Requirements, Challenges, and Their Countermeasures via Software-Defined Security. *IEEE Internet Things J.* **7**(10):10250–10276. (2020).

[48] Da¸s, R., and M.Z. Gündüz. Analysis of Cyber-Attacks in IoT-based Critical Infrastructures. *Int. J. Inf. Secur. Sci.* 8(4):122–133. (2019).

[49] Čekerevac, Z., Z. Dvorak, L. Prigoda, and P. Čekerevac. Internet of Things and the Man-in-the-Middle Attacks – Security and Economic Risks. *MEST J.* 5(2):15–5. (2017).

[50] Zarpelão, B.B., R.S. Miani, C.T. Kawakani, and S.C. de Alvarenga. A survey of intrusion detection in Internet of Things. *J. Netw. Comput. Appl.* 84:25–37. (2017).

[51] Khraisat, A., I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab. A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks. *Electron.* 8(11):(2019).

[52] Chaudhari, R.R., and S.P. Patil. Intrusion Detection System : Classification , Techniques and Datasets To Implement. *Int. Res. J. Eng. Technol.* 4(2):1860–1866. (2017).

[53] Rachel. How does AI and IoT impact the future of work? - SMBHD. 2019. https://www.smbhd.com/ai-and-iot/

[54] Beysolow II, T. *Introduction to Deep Learning Using R: A step-by-Step Guide to learning and Implementing Deep Learning Models Using R*. 2017.

[55] Muralidharan, V., and V. Sugumaran. A comparative study of Naïve Bayes classifier and Bayes net classifier for fault diagnosis of monoblock centrifugal pump using wavelet analysis. *Appl. Soft Comput. J.* 12(8):2023–2029. (2012).

[56] Vijayarani, M.S., M.M. Muthulakshmi, and A. Professor. Comparative Analysis of Bayes and Lazy Classification Algorithms. *Int. J. Adv. Res. Comput. Commun. Eng.* 2(8):(2013).

[57] Kaviani, P., and S. Dhotre. Short Survey on Naive Bayes Algorithm. *Int. J. Adv. Eng. Res. Dev.* 4(11):607–611. (2017).

[58] Khurana, S. Naive Bayes Classifiers - GeeksforGeeks. https://www.geeksforgeeks.org/naive-bayes-classifiers/

[59] Kiranyaz, S., O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D.J. Inman. 1D convolutional neural networks and applications - A survey. 1–20. (2019).

[60] Kang, M.J., and J.W. Kang. Intrusion detection system using deep neural network for in-vehicle network security. *PLoS One* 11(6):1–17. (2016).

[61] Vinayakumar, R., K.P. Soman, and P. Poornachandrany. Applying convolutional neural network for network intrusion detection. *2017 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2017* 2017-Janua:1222–1228. (2017).

[62] Al-Ajlan, A., and A. El Allali. CNN-MGP: Convolutional Neural Networks for Metagenomics Gene Prediction. *Interdiscip. Sci. Comput. Life Sci.* **11**(4):628–635. (2019).

[63] Yin, C., Y. Zhu, J. Fei, and X. He. A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks. *IEEE Access* **5**:21954–21961. (2017).

[64] Bhatt, D. Machine Learning versus Deep Learning - Studytonight. https://www.studytonight.com/post/machine-learning-versus-deep-learning

[65] Elhamahmy, M.E., H.N. Elmahdy, and I.A. Saroit. A New Approach for Evaluating Intrusion Detection System. *CiiT Int.* 2(11):290–298. (2010).

[66] Moustafa, N., 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).

[67] Jiang, K., W. Wang, A. Wang, and H. Wu. Network Intrusion Detection Combined Hybrid Sampling with Deep Hierarchical Network. *IEEE Access* 8(3):32464–32476. (2020).

[68] Zhang, H., L. Huang, C.Q. Wu, and Z. Li. An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Comput. Networks* 177(January):(2020).

[69] Vinayakumar, R., M. Alazab, K.P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman. Deep Learning Approach for Intelligent Intrusion Detection System. *IEEE Access* 7:41525–41550. (2019).

Appendix A

System Interface

This thesis demonstrated the final results of DL and ML algorithms. Where , the efficiency of the proposed CNN DL is showed with a very high precision rate. Accordingly, created a user interface that classifies each sample data at the IoT network layer and generates a two-fold classification: “attack” or “normal.” Figure (1) appears as the main interface of the tool program.

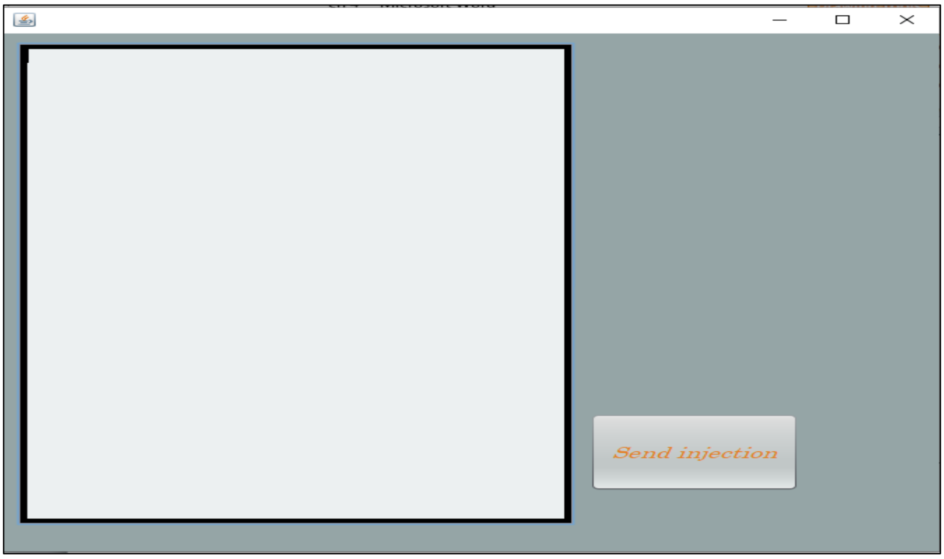


Figure (1): Main Interface of Tool Windows.

Which is load data contain normal and attack traffic that sent it to host by TCP/IP protocols. Figure (2) appears the main interface of the host program. The tool window performs a one-way communication with the host, where the tool sends a message to the host, and the host reads the message and prints it.

Appendix A

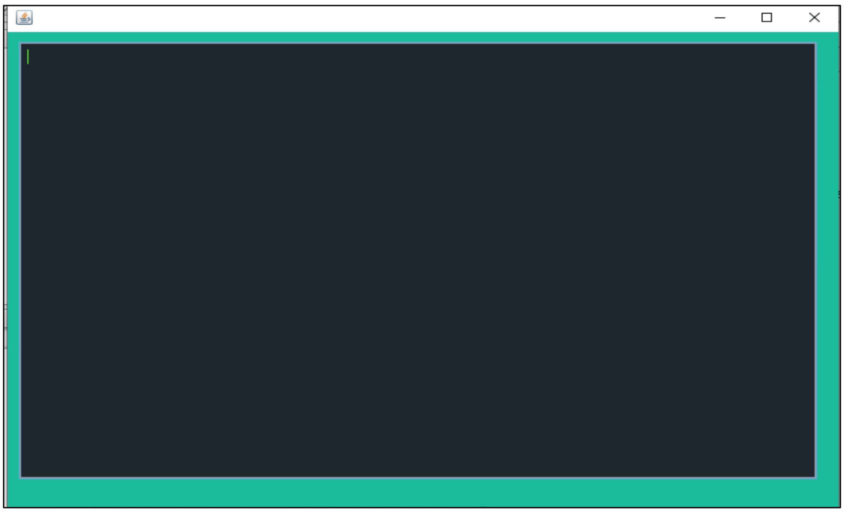
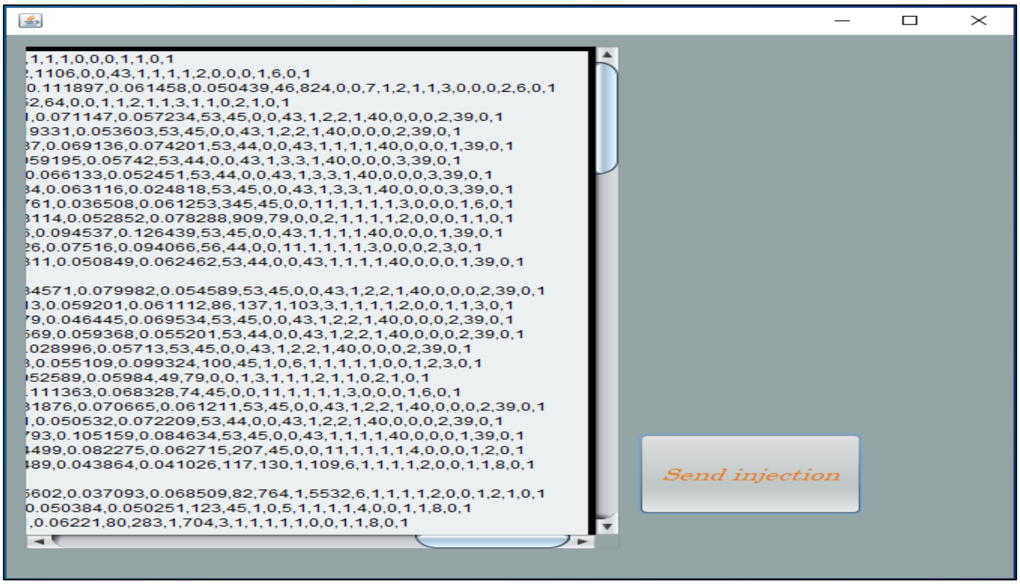


Figure (2): Main Interface of Host Windows.

The tool must know the server's IP address and port number. After clicking the button Sent Injection in the tool, a parsing process will begin in the host window as shown in Figure(3).



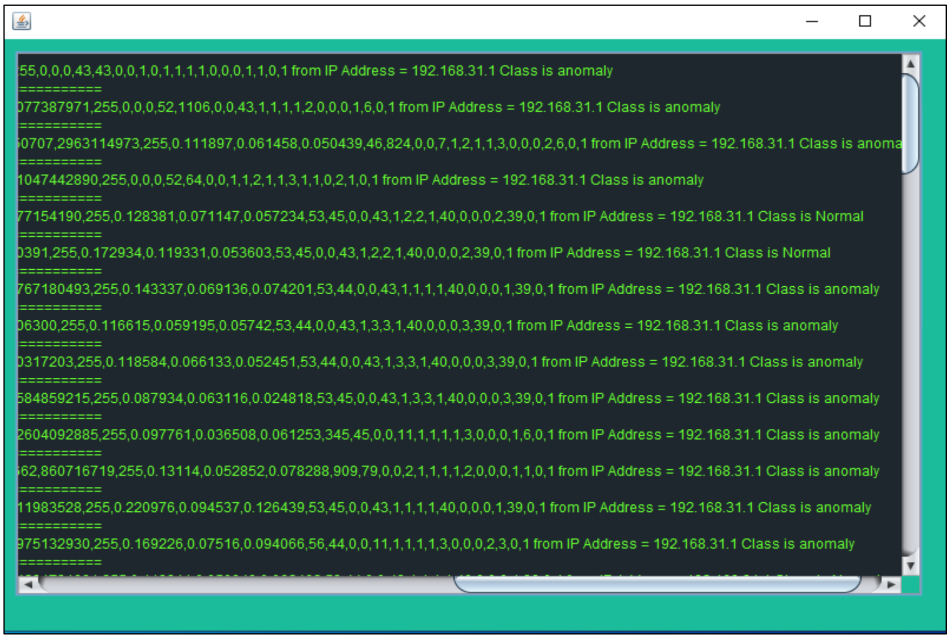
```
,1,1,1,0,0,0,1,1,0,1
2,1106,0,0,43,1,1,1,1,2,0,0,0,1,6,0,1
0.111897,0.061458,0.050439,46,824,0,0,7,1,2,1,1,3,0,0,0,2,6,0,1
52,64,0,0,1,1,2,1,1,3,1,1,0,2,1,0,1
1,0.071147,0.057234,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
19331,0.053603,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
37,0.069136,0.074201,53,44,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1
059195,0.05742,53,44,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1
0.066133,0.052451,53,44,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1
34,0.063116,0.024818,53,45,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1
761,0.036508,0.061253,345,45,0,0,11,1,1,1,1,3,0,0,0,1,6,0,1
3114,0.052852,0.078288,909,79,0,0,2,1,1,1,1,2,0,0,0,1,1,0,1
5,0.094537,0.126439,53,45,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1
26,0.07516,0.094066,56,44,0,0,11,1,1,1,1,3,0,0,0,2,3,0,1
311,0.050849,0.062462,53,44,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1

34571,0.079982,0.054589,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
13,0.059201,0.061112,86,137,1,103,3,1,1,1,1,2,0,0,1,1,3,0,1
79,0.046445,0.069534,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
569,0.059368,0.055201,53,44,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
.028996,0.05713,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
3,0.055109,0.099324,100,45,1,0,6,1,1,1,1,1,0,0,1,2,3,0,1
052589,0.05984,49,79,0,0,1,3,1,1,1,2,1,1,0,2,1,0,1
.111363,0.068328,74,45,0,0,11,1,1,1,1,3,0,0,0,1,6,0,1
31876,0.070665,0.061211,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
1,0.050532,0.072209,53,44,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1
793,0.105159,0.084634,53,45,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1
4499,0.082275,0.062715,207,45,0,0,11,1,1,1,1,4,0,0,0,1,2,0,1
489,0.043864,0.041026,117,130,1,109,6,1,1,1,1,2,0,0,1,1,8,0,1

5602,0.037093,0.068509,82,764,1,5532,6,1,1,1,1,2,0,0,1,2,1,0,1
0.050384,0.050251,123,45,1,0,5,1,1,1,1,4,0,0,1,1,8,0,1
,0.06221,80,283,1,704,3,1,1,1,1,1,0,0,1,1,8,0,1
```

Figure (3): Tool Window Start System Test

Appendix A



```
55,0,0,0,43,43,0,0,1,0,1,1,1,1,0,0,0,1,1,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
077387971,255,0,0,0,52,1106,0,0,43,1,1,1,1,2,0,0,0,1,6,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
0707,2963114973,255,0.111897,0.061458,0.050439,46,824,0,0,7,1,2,1,1,3,0,0,0,2,6,0,1 from IP Address = 192.168.31.1 Class is anoma
==========
1047442890,255,0,0,0,52,64,0,0,1,1,2,1,1,3,1,1,0,2,1,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
77154190,255,0.128381,0.071147,0.057234,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1 from IP Address = 192.168.31.1 Class is Normal
==========
0391,255,0.172934,0.119331,0.053603,53,45,0,0,43,1,2,2,1,40,0,0,0,2,39,0,1 from IP Address = 192.168.31.1 Class is Normal
==========
767180493,255,0.143337,0.069136,0.074201,53,44,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
06300,255,0.116615,0.059195,0.05742,53,44,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
0317203,255,0.118584,0.066133,0.052451,53,44,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
584859215,255,0.087934,0.063116,0.024818,53,45,0,0,43,1,3,3,1,40,0,0,0,3,39,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
2604092885,255,0.097761,0.036508,0.061253,345,45,0,0,11,1,1,1,1,3,0,0,0,1,6,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
62,860716719,255,0.13114,0.052852,0.078288,909,79,0,0,2,1,1,1,1,2,0,0,0,1,1,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
11983528,255,0.220976,0.094537,0.126439,53,45,0,0,43,1,1,1,1,40,0,0,0,1,39,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
975132930,255,0.169226,0.07516,0.094066,56,44,0,0,11,1,1,1,1,3,0,0,0,2,3,0,1 from IP Address = 192.168.31.1 Class is anomaly
==========
```

Figure (4): System Test Traffic Pass Through CNN Classifier

At hit on the injection button, the data goes through a CNN filter in the host window to reads the message that has arrived and IP to classify it and analyze them according to whether the normal or attack in addition to the time of arrival the message to know the time of the attack behavior is shown in Figure(4) .

الخلاصة

يربط إنترنت الأشياء كل شيء بالإنترنت ويمكّن الأشياء من التواصل مع بعضها البعض عبر شبكات سلكية أو لاسلكية. زاد عدد تطبيقات إنترنت الأشياء بشكل كبير ، مثل المدن الذكية والمنازل الذكية والأجهزة القابلة للارتداء والرعاية الصحية. يصبح الأمان أكثر أهمية مع زيادة عدد الأجهزة المتصلة بإنترنت الأشياء بسبب أنواع الأجهزة وحجم البيانات التي يتم إرسالها عبر الشبكة وطبيعة الهيكل وطرق الاتصال المختلفة (لاسلكيًا بشكل أساسي). تعمل هذه الطبيعة الأساسية لهندسة إنترنت الأشياء على تكثيف عدد أهداف الهجوم التي قد تؤثر على النمو المستدام لإنترنت الأشياء. ومن ثم ، تصبح القضايا الأمنية عاملاً حاسماً يجب معالجته. لذلك ، أصبح من الضروري تطوير نظام للكشف عن الهجمات لمواكبة التطور الحالي لإنترنت الأشياء ، حيث إنه يتعامل مع المعلومات الحساسة التي يجب حمايتها وتأمينها. في هذه الأطروحة ، تم اقتراح نهج للتعلم العميق يعتمد على الشبكات العصبية التلافيفية لإجراء الكشف في الوقت الفعلي عن سلوكيات الهجوم في أنظمة إنترنت الأشياء. تم استخدام مجموعة بيانات (UNSW-NB15) لتدريب واختبار النهج المقترح. يستخدم الأسلوب التصنيف الثنائي للتمييز بين أنماط الهجوم والأنماط العادية. يمر النهج المقترح بمرحلتين. الاولى : بعد تحميل مجموعة البيانات ، يتم إجراء المعالجة المسبقة للحصول على نتائج أكثر دقة. المرحلة الثانية هي التصنيف بواسطة مصنف CNN تظهر النتيجة التجريبية كفاءة النموذج المقدم فيما يتعلق بالدقة والاستدعاء والقياس حيث بلغت الدقة ١٠٠٪. تعتبر نتيجة التجارب ذات كفاءة عالية في اكتشاف التسلل للتمييز بين السلوكيات العادية وسلوك الهجوم وتوفر منهجية بحثية.

جمهوريـــــــــة العـــــــــراق
وزارة التعليم العالي و البحث العلمي
جامــعــة الأنبـار
كليـة علوم الحاسوب وتكنلوجيا المعلومات
قســـم علوم الحاسبات

مناهج كشف التسلل لإنترنت الأشياء

رسالة مقدمة إلى

كلية علوم الحاسوب وتكنلوجيا المعلومات – جامعة الانبار – قسم علوم الحاسبات

وهي جزء من متطلبات نيل درجة الماجستير في علوم الحاسبات

قدمت من قبل

مها ماجد محمد

بإشراف

أ.م.د. خطاب معجل الهيتـي

٢٠٢١م

1443هـ