République algérienne démocratique et populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Echahid Cheikh Larbi Tébessi – Tébessa

Faculté des Sciences Exactes et des Sciences Naturelles et de la Vie

Département d'informatique

Mémoire

Présentée en vue de l'obtention du diplôme de Master LMD

En : Réseaux et sécurité informatique

Réalisé par : Haoues Manar

Théme

Une Approche Intelligente pour Améliorer la Cybersécurité dans l'Internet des Objets Industriel

Soutenu publiquement le : 09 / 06 / 2024 devant le jury composé de :

Dr. HADJADJ Ismail	MAA	Université de Tébessa	Président
Dr. MERZOUG Soltane	MCA	Université de Tébessa	Encadrant
Dr. BOUAMRA Oualid	MAB	Université de Tébessa	Examinateur

2023/2024

People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

Echahid Cheikh Larbi Tebessi University – Tebessa

Faculty of Exact Sciences and Natural and Life Sciences

Department of Computer Science

Thesis

Submitted in partial fulfillment of the requirements for the
Master’s Diploma (LMD).

In: Networks and IT security

Realized By: Haoues Manar

Theme

A Smart Approach to Improve Cyber Security in The Industrial Internet of Things

Publicly sustained on: 09 /06 / 2024 before the jury composed of:

Dr. HADJADJ Ismail	MAA	University of Tebessa	President
Dr. MERZOUG Soltane	MCA	University of Tebessa	Supervisor
Dr. BOUAMRA Oualid	MAB	University of Tebessa	Examiner

2023/2024

Acknowledgments

I would like to express my deepest gratitude to Dr. Merzoug Soltane for his invaluable guidance and supervision throughout the course of this project. His expertise, encouragement, and insightful feedback have been instrumental in the successful completion of this work.

I extend my heartfelt thanks to DR Hadjadj Ismail and DR Bouamra Oualid for graciously agreeing to examine this thesis. Their time, effort, and constructive criticism are greatly appreciated and will significantly contribute to the refinement of this research.

I am also profoundly thankful to JVgas, the company where I conducted my internship, for providing me with an exceptional learning environment. I am particularly grateful to Dr. Abderrahim Annou, Network Engineer, and Mr. Emad Chalabi, Cybersecurity Team leader, for their invaluable support, mentorship, and sharing of their extensive knowledge and experience. Their assistance has been crucial in the development and completion of this project.

Thank you all for your support and encouragement, as well as to anyone who has helped me along the way.

Dedication

To the person I was

To the person I am

To the person I will be

Abstract

The Industrial Internet of Things (IIoT) has transformed industrial processes by seamlessly integrating a vast network of interconnected devices. These intelligent devices equipped with sensors and network capabilities collect and transmit critical data to enable remote monitoring and control. However, this interconnectedness creates a complex ecosystem susceptible to cyberattacks. Distributed Denial-of-Service (DDoS) attacks specifically pose a significant threat to IIoT systems, as they can cripple operations by overwhelming them with a flood of malicious traffic. Traditional security methods often struggle with the evolving nature and sophistication of these attacks.

This work proposes a novel approach to enhance cybersecurity in IIoT by focusing on the development of a robust and efficient DDoS detection system. By leveraging the transformative power of deep learning techniques, specifically a Convolutional Neural Network (CNN) model, the system aims to identify and mitigate DDoS attacks in real time. The comprehensive Edge-IIoTset dataset provides a valuable resource for training and validating the proposed CNN model. The system's performance is evaluated using established machine learning metrics. The obtained results will be analyzed to assess the effectiveness of the CNN model in detecting DDoS attacks within the IIoT environment.

Keywords: Cybersecurity, Industrial Internet of Things (IIoT), Deep Learning, Convolutional Neural Network (CNN), DDoS Attack Detection, Edge-IIoTset Dataset.

Résumé

L'Internet industriel des objets (IIoT) a transformé les processus industriels en intégrant de manière transparente un vaste réseau d'appareils interconnectés. Ces appareils intelligents, équipés de capteurs et de capacités réseau, collectent et transmettent des données critiques, permettant une surveillance et un contrôle à distance. Cependant, cette interconnectivité crée un écosystème complexe vulnérable aux cyberattaques. Les attaques par déni de service distribué (DDoS), en particulier, constituent une menace importante pour les systèmes IIoT, car elles peuvent paralyser les opérations en les submergeant d'un flot de trafic malveillant. Les méthodes de sécurité traditionnelles sont souvent confrontées à la nature évolutive et à la sophistication de ces attaques.

Ce travail propose une nouvelle approche pour améliorer la cybersécurité dans l'IIoT en se concentrant sur le développement d'un système de détection DDoS robuste et efficace. En tirant parti du pouvoir transformateur des techniques d'apprentissage profond, en particulier d'un modèle de réseau neuronal convolutif (CNN), le système vise à identifier et à atténuer les attaques DDoS en temps réel. L'ensemble de données complet Edge-IIoTset fournit une ressource précieuse pour la formation et la validation du modèle CNN proposé. Les performances du système sont évaluées à l'aide de mesures d'apprentissage automatique établies. Les résultats obtenus seront analysés pour évaluer l'efficacité du modèle CNN dans la détection des attaques DDoS dans l'environnement IIoT.

Mots-clés : cybersécurité, Internet industriel des objets (IIoT), apprentissage profond, réseau neuronal convolutif (CNN), détection d'attaques DDoS, ensemble de données Edge-IIoTset.

ملخص

لقد أحدثت إنترنت الأشياء الصناعية (IIoT) تحولًا في العمليات الصناعية من خلال التكامل السلس لشبكة واسعة من الأجهزة المترابطة. تقوم هذه الأجهزة الذكية، المزودة بأجهزة استشعار وقدرات شبكية، بجمع البيانات الهامة ونقلها، مما يتيح المراقبة والتحكم عن بعد. ومع ذلك، فإن هذا الترابط يخلق نظامًا بيئيًا معقدًا عرضة للهجمات الإلكترونية. تشكل هجمات رفض الخدمة الموزعة (DDoS)، على وجه التحديد، تهديدًا كبيرًا لأنظمة إنترنت الأشياء الصناعية(IIoT) ، لأنها يمكن أن تشل العمليات من خلال إغراقها بسيل من البيانات الضارة. غالبًا ما تواجه أساليب الأمن التقليدية صعوبة في التعامل مع الطبيعة المتطورة لهذه الهجمات وتعقيدها.

يقترح هذا العمل نهجًا جديدًا لتعزيز الأمن السيبراني في إنترنت الأشياء الصناعية (IIoT) من خلال التركيز على تطوير نظام قوي وفعال للكشف عن DDoS. ومن خلال الاستفادة من القوة التحويلية لتقنيات التعلم العميق، وتحديدًا نموذج الشبكة العصبية المعقدة(CNN) ، يهدف النظام إلى تحديد هجمات DDoS والتخفيف منها في الوقت الفعلي. توفر مجموعة بيانات Edge-IIoTset الشاملة موردًا قيمًا للتدريب والتحقق من صحة نموذج CNN المقترح. يتم تقييم أداء النظام باستخدام مقاييس التعلم الآلي المعمول بها. سيتم تحليل النتائج التي تم الحصول عليها لتقييم فعالية نموذج CNN في اكتشاف هجمات DDoS داخل بيئة IIoT.

الكلمات المفتاحية: الأمن السيبراني، إنترنت الأشياء الصناعي(IIoT) ، التعلم العميق، الشبكة العصبية التلافيفية(CNN) ، اكتشاف هجماتDDoS ، مجموعة بيانات Edge-IIoTset.

Table of Content:

List of Figures :

List of Tables :

General Introduction:

The Internet of Things (IoT) has revolutionized the interaction between physical objects and the digital world by enabling devices embedded with sensors, software, and other technologies to exchange data over the internet [1]. The Industrial Internet of Things (IIoT) extends this concept further by integrating industrial machinery with sensors and actuators, facilitating real-time monitoring and control over the internet. This integration allows for the collection, processing, and analysis of vast amounts of data related to industrial processes, which can be used to optimize operations, predict future behavior, and ultimately improve efficiency [2].

As the number of interconnected devices in IIoT systems expands, so do security concerns. Protecting critical infrastructure (CI), particularly Internet Industrial Control Systems (IICSs), is paramount for businesses. Distributed Denial-of-Service (DDoS) attacks are a significant threat, often employing botnets to overwhelm systems with traffic, thereby hindering authorized access or slowing operations [3].

This thesis proposes an approach to secure IIoT networks: An Intrusion Detection System (IDS) built upon an optimized Convolutional Neural Network (CNN) model. CNNs are a type of deep learning algorithm particularly adept at processing structured data, such as time series, making them ideal for analyzing network traffic patterns. By learning complex features that differentiate normal and malicious activities, CNNs can effectively detect anomalies indicative of attacks [4]. The proposed model will be trained on a comprehensive dataset, such as the Edge-IIoTset dataset, which realistically reflects real-world IIoT environments with data from various devices, protocols, and simulated attacks.

This thesis is divided into four chapters:

Chapter 1: Fundamentals of IoT, IIoT, Cyber Security: This chapter lays the groundwork by introducing the fundamentals of IoT and IIoT, analyzing common attacks targeting IIoT environments, and exploring the realm of cybersecurity within this domain. It will also delve into the integration of artificial intelligence (AI) for enhanced IIoT security.

Chapter 2: Cybersecurity Solutions for IIoT: This chapter identifies and categorizes the prevailing cybersecurity challenges in IIoT environments. It will conduct a state-of-the-art analysis to examine various approaches and methods currently used to address these challenges.

Chapter 3: Methodology of Training a DL Model: This chapter focuses on the proposed solution. It will detail the novel approach for securing IIoT networks using a CNN-based deep learning model. The optimization of the CNN model for effective analysis of complex network traffic patterns and anomaly detection specific to DDoS attacks will be explained.

Chapter 4: Development Environment and Simulation: This chapter will present the implementation of the Intrusion Detection System (IDS) within the architecture of an industrial company's IT and OT (Operational Technology) network and showcase how the trained model can be deployed within the IIoT environment, it will also discuss the performance and the obtained results of the proposed model, analyzing its effectiveness in detecting DDoS attacks within IIoT environments.

Chapter 1

Fundamentals of IoT, IIoT, Cyber Security

1. Introduction:

Security is a vital aspect in the domains of Internet of Things (IoT) and Industrial Internet of Things (IIoT), deeply ingrained in the structure of interconnected systems. This chapter provides a foundational examination, elucidating the fundamental principles governing the fusion of IoT, IIoT, and cyber security. Our journey starts with exploring the essence of IoT and IIoT. Clear distinctions between these domains emerge, highlighting their unique attributes. Cyber security emerges as a focal point, emphasizing its critical role in protecting IIoT operations and navigating the intricate array of threats and attacks in these interconnected environments. Advanced security methodologies, ranging from Intrusion Detection Systems to Artificial Intelligence, serve as our tools for comprehension and defense in this constantly evolving landscape. Through this examination, we establish the groundwork for a deeper comprehension of the complexities and interactions within these pivotal domains.

2. Basics of Iot & IIot:

2.1. Internet of Things (IoT):

The Internet of Things (IoT) comprises a network of physical devices, appliances, vehicles, and objects embedded with sensors, software, and connectivity for data collection and sharing. These "smart objects" range from basic home gadgets to complex industrial systems, with visions of entire smart cities driven by IoT innovations. IoT enables seamless communication between devices and internet-enabled entities, creating a vast interconnected network capable of autonomous data exchange and task execution. Its applications span diverse sectors such as agriculture, transportation, healthcare, and manufacturing, with IoT poised to reshape how we live, work, and interact as internet-connected devices proliferate. In enterprises, IoT devices monitor parameters like temperature and energy consumption, aiding in operational optimization through real-time data analysis.[5]

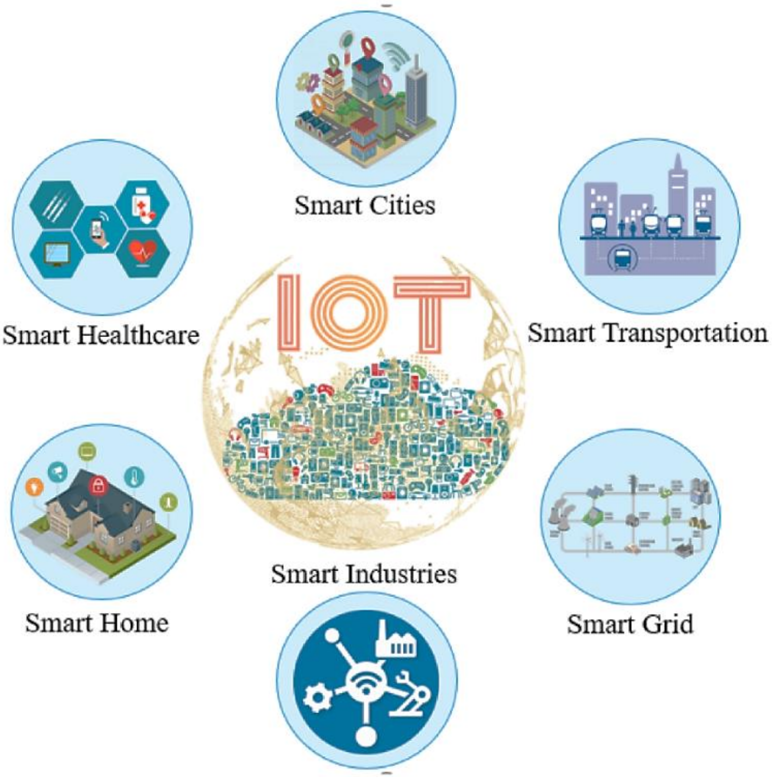


Figure 1.1. Diversity of Iot applications [6]

2.2. Industrial internet of things (IIoT):

The industrial internet of things (IIoT) refers to the utilization of smart sensors, actuators, and various devices, including radio frequency identification tags, to improve manufacturing and industrial operations. These interconnected devices enable data collection, exchange, and analysis, leading to insights that enhance efficiency and reliability. IIoT finds applications across diverse industries such as manufacturing, energy management, utilities, and oil and gas. it harnesses the capabilities of smart machines and real-time analytics to leverage data generated by traditional machines in industrial settings. Its fundamental premise lies in the notion that smart machines excel not only in capturing and analyzing data in real time but also in communicating vital information swiftly and accurately, facilitating faster and more informed business decisions. Through connected sensors and actuators, companies can identify inefficiencies and issues earlier, thereby saving time and resources while bolstering business intelligence initiatives. In manufacturing, IIoT holds promise for enhancing quality control, promoting sustainable practices, enabling supply chain traceability, and optimizing overall supply chain efficiency. Crucially, IIoT plays a pivotal role in industrial processes such as predictive maintenance, enhanced field service, energy management, and asset tracking.[7]

2.3. Comparisons between Iot & IIot:

The primary distinction between IoT and IIoT lies in their scope and focus. IoT encompasses various connected devices and systems designed to enhance consumer lifestyles, such as smart home systems and wearable technology, emphasizing user-friendliness and convenience. In contrast, IIoT has a narrower focus, targeting industrial sectors like manufacturing, transportation, and energy. IIoT aims to enhance efficiency, productivity, and safety through automation, data analysis, and predictive maintenance, enabling businesses to make data-driven decisions and optimize processes.[8]

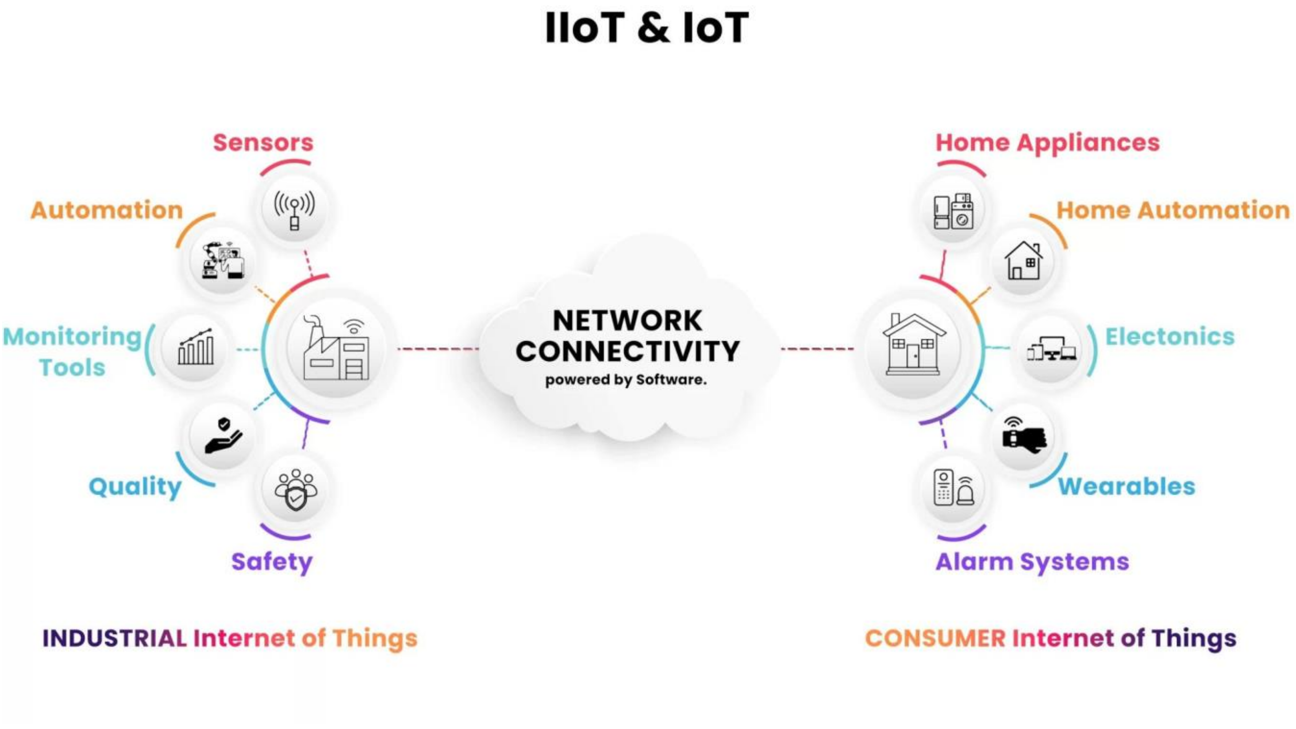


Figure 1.2. The Differences between IoT and IIoT [9]

ATTRIBUTES	Industrial Internet of Things – IIoT	Internet of Thing – IoT
Focus Segment	Industrial applications.	Domestic/commercial applications.
Interest	Complex industrial processes optimization via smart devices.	Daily task automation through consumer's devices.
Objective	Aimed at automating machinery to ensure safety, efficiency and sustainability.	Aimed at rendering convenience. Simply making the user's life easy.
Connectivity	Both wired and wireless.	Usually wireless.
Cybersecurity	More advanced and robust cybersecurity protocols.	Generally less sophisticated cybersecurity protocols (utility-centric).
Interoperability	CPS-integrated – interoperations with new and legacy technologies like ERP, warehousing solution etc., and must operate reliably with these technologies.	Autonomous – devices usually operate individually, sometimes with one or two different devices.
Sensor Utilisation	Sophisticated sensors e.g., pressure sensors, torque sensors, speed sensors, radio-frequency identification (RFID) sensors etc.	Basic sensors e.g., motion sensors, temperature sensors, moisture sensors etc.
Precision & Accuracy	Precise and accurate enough to manage various synchronized industrial processes down to milliseconds.	Typically accurate enough to gather limited amount of data for a specific activity.
Data Quantity	High to very high.	Medium to high.
Maintenance	Properly scheduled and planned.	Preference of the users.

Table 1.1. Comparisons table between Iot & IIot [9]

2.4. Importance and applications of IIot:

2.4.1. Importance of IIot:

IIoT devices employed in the manufacturing industry offer several advantages:

- **Predictive maintenance**: Real-time data from IIoT systems enables organizations to predict when machines require servicing, allowing proactive maintenance to prevent costly breakdowns, particularly on production lines where machine failure can lead to work stoppages and significant expenses. Addressing maintenance issues pre-emptively enhances operational efficiency.
- **More efficient field service**: IIoT technologies aid field service technicians in identifying potential equipment issues before they escalate, facilitating timely repairs to prevent customer disruptions. These technologies also provide technicians with information on required parts for repairs, ensuring they have the necessary components during service calls.
- **Asset tracking**: Asset management systems enable suppliers, manufacturers, and customers to monitor product location, status, and condition throughout the supply chain. Instant alerts notify stakeholders of potential damage risks, allowing immediate or preventive action to mitigate issues.
- **Increased customer satisfaction**: IoT-connected products enable manufacturers to gather and analyze data on customer usage, empowering them to create more customer-centric product designs and roadmaps.[7]

2.4.2. Applications of IIot:

Most industrial organizations have adopted IIoT solutions to improve monitoring, maintenance, and remote operations. Knowing the main IIoT applications and how to implement the system to access this competitive advantage would be wise. [10]

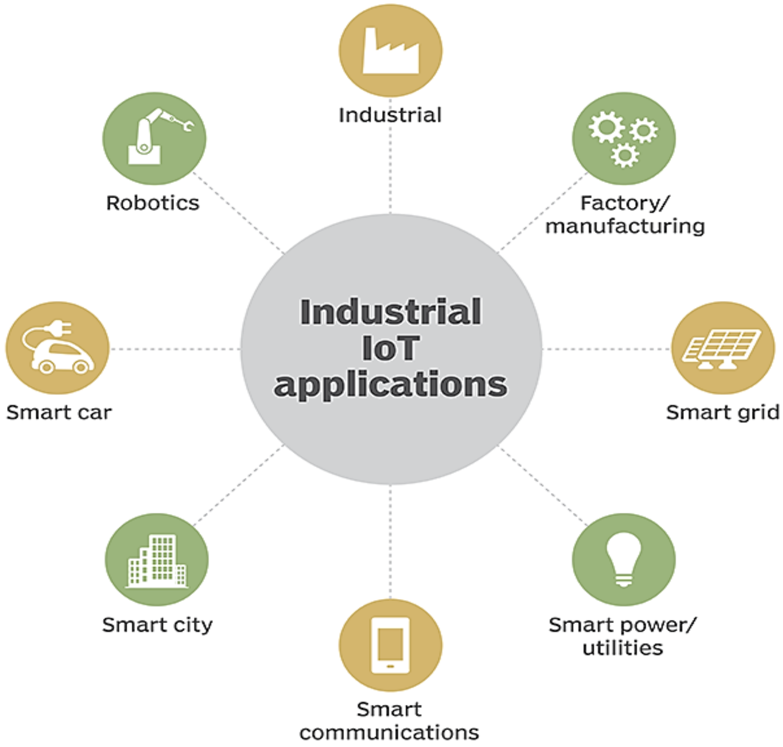


Figure 1.3. IIot applications [7]

- **Industrial Automation:** it is one of the Internet of Things' most significant and standard applications. Machine and tool automation enables companies to operate efficiently with the latest software tools to monitor and make improvements for the following process iterations.
- **Autonomous Vehicles:** Autonomous vehicles don't just imply self-driving cars or trucks. Many examples of warehouses have deployed robots that work without human intervention. These also include autonomous robots.
- **Futuristic Farming:** IIoT makes a big difference in agriculture. By implementing connected IIoT projects in farms, the farmers can keep track of the yield from the field to the market.
- **Quality Control:** Another essential IIoT application is their ability to monitor the quality of the manufactured products at any stage-from the raw materials used in the process to how they are transported to the reactions of the end customer once the product is received.
- **Energy Networks:** Energy is the most crucial resource and must be used to the maximum without waste. There are various IIoT applications in the energy sector. One such application is Smart Meters, which monitors energy consumption at specific times and reports back. The oil and gas industries also use IIoT with smart sensors. Whenever sensors detect oil or gas leakage in any of its pipelines, inform immediately to the maintenance teams. This helps avert any dangers and ensures a steady supply at all times. [10]

2.5. Components of IIoT:

- **Smart Machines:** Machines are an integral part of any manufacturing or processing industry. Ordinary machines are programmed to do one thing, and they do it with high efficiency. They are an improvement over regular machines as they can communicate with other machines.
- **Sensors:** Sensors are an integral part of both IoT and IIoT. it detects the changes in the physical environment and convert them into electrical signals. These electrical signals are the data that helps us understand the physical quantity measured by the sensor.
- **Infrastructure:** Infrastructure concerning IIoT is the network through which all digital communication happens. Without a secure and fast communication platform, data transfer will face obstructions that will make the entire setup futile.
- **Software, Radios, and Controllers:** An industry setup is very different from what we are accustomed to with the traditional IoT. In industries, a piece of machinery receives support from many other devices to create a system. Hence, a machine will have controllers or radios, and they run over custom software. With IIoT, these subunits should also support IIoT standards.[11]

2.6. IIot Architecture:

While IIoT systems vary widely, they have similar architectural features.

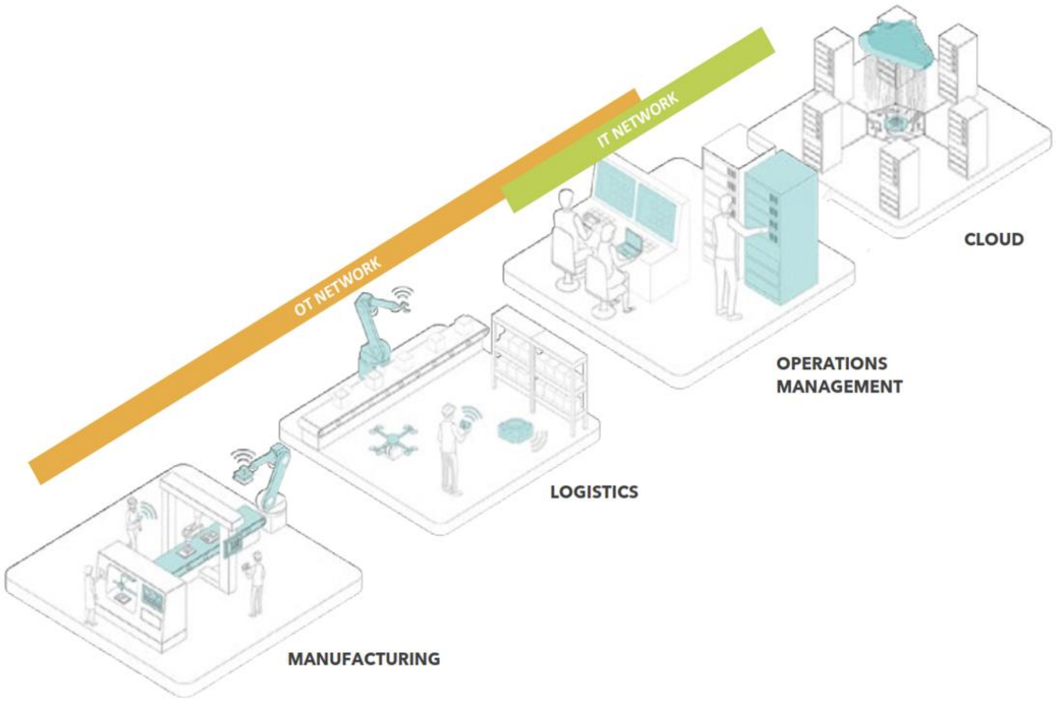


Figure 1.4. Architecting for IIoT [12]

The data gathered by IoT devices in the manufacturing and logistics areas flow through gateways to the Operations Management area, Supervisory Control and Data Acquisition systems (SCADA), and Manufacturing Execution Systems (MES). These consolidate and convert the raw data into information for analysis by applications locally at the edge, are sent to cloud-based data centers, or a combination of both.

Traditional operational technology (OT) systems that managed and controlled operations were "air-gapped" environments, meaning that they were not connected to external networks. However, across industries today, the lines have blurred between information technology (IT) and OT, bringing connected IT systems that handle email and data processing together with self-contained OT systems. There are many benefits of this convergence, from lowering operating costs by giving manufacturers greater transparency into performance and helping energy utility providers offer consumer engagement systems based on real-time usage and rates. [12]

3. Cyber security in IIoT:

3.1. Definition of cyber security:

Cybersecurity involves safeguarding systems, networks, and programs from digital attacks, which aim to access, alter, or compromise sensitive information and disrupt business operations. IIoT security serves as a protective barrier for smart devices and machinery in industries, employing tools and practices to ward off hackers and digital threats. It includes techniques such as strong passwords, encryption, and specialized software to detect suspicious activities. Crucially, IIoT security prevents production disruptions, safeguards sensitive data, and ensures smooth and secure industrial operations.[14]

3.2. Objectives of cyber security in IIoT:

Cybersecurity plays a critical role in the Industrial Internet of Things (IIoT), ensuring the safety, reliability, and integrity of industrial systems and processes. The objectives of IIoT cybersecurity encompass several crucial areas:

- **Protection of Data**: IIoT security measures aim to safeguard sensitive information, ensuring its integrity and confidentiality.
- **Prevention of Disruptions**: By shielding industrial operations from cyber threats, IIoT security helps prevent disruptions and downtime.
- **Assurance of Safety**: IIoT security protocols work to maintain a safe environment for both machinery and personnel, preventing unauthorized access.

- **Maintenance of Reliability**: IIoT security strategies aim to uphold the reliability of IIoT systems, minimizing the risk of security breaches that could lead to operational interruptions.
- **Privacy Preservation**: IIoT security protocols prioritize the protection of user data and sensitive information from potential cyber threats, preserving privacy.
- **Restriction of Unauthorized Access:** IIoT security measures ensure that only authorized individuals can access and control IIoT devices and systems.
- **Protection of Critical Infrastructure**: IIoT security efforts extend to safeguarding essential infrastructure, such as power grids and transportation systems, from cyberattacks.
- **Reduction of Risk**: By mitigating the risk of cyber threats, IIoT security enhances confidence in industrial operations and minimizes potential vulnerabilities.
- **Enhancement of Trust:** IIoT security initiatives build trust among users and consumers by demonstrating the reliability and security of interconnected systems.
- **Facilitation of Growth**: IIoT security measures enable the expansion of IIoT networks without compromising security, thereby fostering industry growth and innovation.[14]

3.3. Threats and attacks in IIoT:

The adoption of Industrial IoT (IIoT) systems exposes industries to an evolving threat landscape, with attackers targeting these environments to cause disruptions and compromise data integrity, posing physical risks. Implementing robust cybersecurity measures and remaining vigilant against emerging threats is crucial for safeguarding critical infrastructure and maintaining operational resilience in IIoT environments.

3.3.1. Types of attacks in IIoT:

IIoT systems face susceptibility to diverse attack types, which can undermine their integrity, disrupt operations, and pose threats to essential infrastructure. Understanding these prevalent IIoT attack methodologies is pivotal for crafting robust cybersecurity strategies. Let's explore some of the primary attack vectors prevalent in the IIoT domain:

- **Denial-of-Service (DoS) Attacks**: Denial-of-Service attacks aim to overwhelm IIoT systems with a flood of requests, rendering them unavailable to legitimate users. By overloading the system's resources, attackers can disrupt operations, causing financial losses and impacting productivity.
- **Man-in-the-Middle (MitM) Attacks**: MitM attacks involve intercepting and altering the communication between IIoT devices, gaining unauthorized access to sensitive data or injecting malicious code. Attackers can exploit vulnerabilities in the network infrastructure or devices to eavesdrop, manipulate, or inject malicious commands.

- **Device Exploitation**: Attackers target the vulnerabilities present in IIoT devices to gain unauthorized control or access. By exploiting security weaknesses, such as default or weak credentials, outdated firmware, or unpatched software, they can compromise the device's functionality and potentially gain control over the entire IIoT system.
- **Physical Attacks:** Physical attacks involve tampering with IIoT devices or infrastructure components. Attackers may physically access the devices to manipulate sensors, inject malicious code, or disrupt the operation of critical equipment. Physical attacks pose a significant risk to the integrity and safety of industrial processes.
- **Data Interception and Tampering**: IIoT systems rely on the seamless exchange of data between devices, networks, and cloud platforms. Attackers may intercept and manipulate the data transmitted across the IIoT ecosystem, leading to data breaches, unauthorized access to sensitive information, or the manipulation of critical operational data.
- **Supply Chain Attacks**: Supply chain attacks occur when attackers compromise the integrity of IIoT devices or components during the manufacturing or distribution process. By injecting malicious code or tampering with the devices, they can gain unauthorized access or control over the IIoT system, posing significant risks to the entire infrastructure.
- **Firmware and Software Vulnerabilities**: IIoT devices often rely on firmware and software to operate effectively. Vulnerabilities within the firmware or software can be exploited by attackers to gain unauthorized access, manipulate device functionality, or inject malicious code into the system.[15]

3.3.2. DDoS Attack in IIoT:

To effectively detect and mitigate DDoS attacks in Industrial IoT (IIoT) environments we must have a deep understanding of their mechanisms. DDoS attacks, as illustrated in figure 1.5 above unfold in distinct phases, each offering valuable insights for early detection.

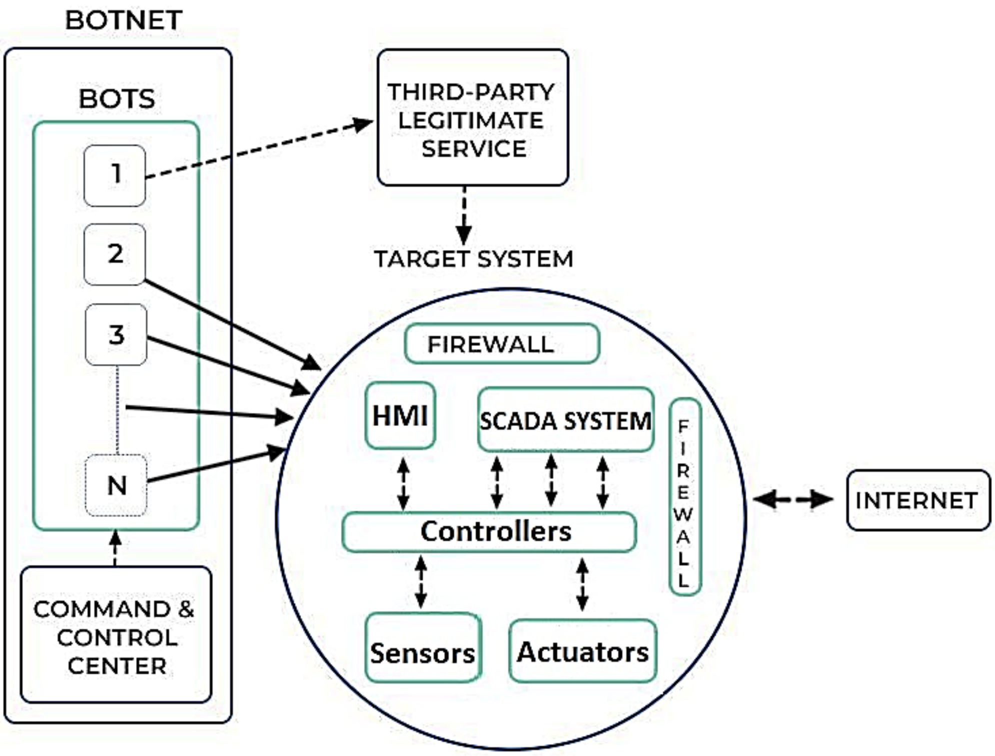


Figure 1.5. DDoS in an IIoT environment

In the botnet creation phase, attackers, also known as botmasters, scan IIoT networks for exploitable weaknesses. This might involve targeting weak passwords on Programmable Logic Controllers (PLCs) responsible for automating specific tasks, unpatched vulnerabilities in SCADA (Supervisory Control and Data Acquisition) systems that monitor and control industrial processes, or open ports on industrial networking devices. Once a vulnerability is identified, the attacker leverages it to gain control of the device, transforming it into a malicious bot.

These compromised devices can encompass a wide range of IIoT components critical for industrial operations. This can include sensors and actuators that collect and control physical equipment, Human-Machine Interfaces (HMI) used for operator interaction, or even entire Industrial Control Systems (ICS) themselves. The collection of these compromised IIoT devices forms a botnet, essentially an army of devices under the attacker's control.

With the botnet established, the botmaster designates a compromised device within the network or an external server hidden behind anonymity services as the Command and Control (C&C) server. This C&C server acts as the central control point, issuing commands to the botnet to launch the attack. In the attack phase, the bots bombard the target IIoT system or service with a flood of malicious traffic, overwhelming its resources and potentially disrupting critical industrial processes.

To achieve this early detection, our model will utilize a Convolutional Neural Network (CNN) to automatically extract relevant features from the captured IIoT network traffic data.

3.4. Advanced Security Approaches:

The best defense against cyber-attacks in an industrial IoT environment is to ensure that proper security measures are taken on all levels. Establishing a secure approach and using it as the foundation for secure products, advanced security features and functions, and comprehensive security management.

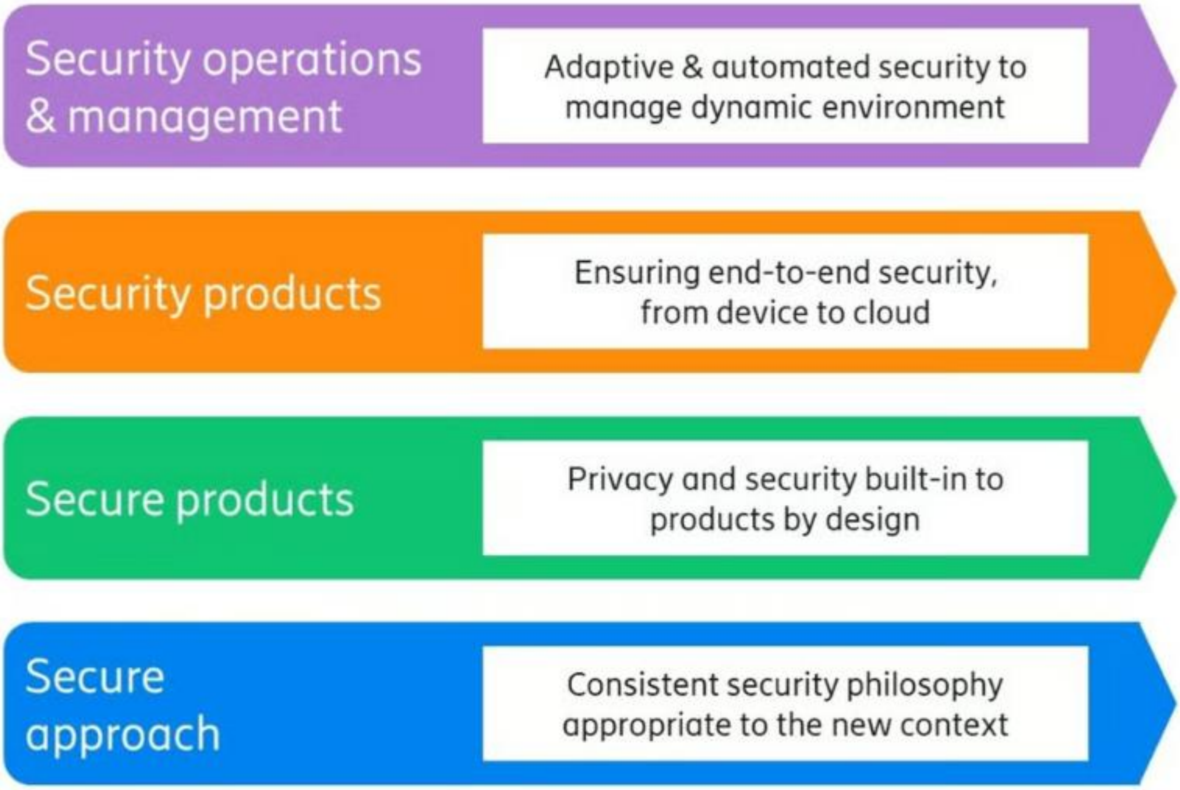

Security operations & management
Adaptive & automated security to manage dynamic environment
Security products
Ensuring end-to-end security, from device to cloud
Secure products
Privacy and security built-in to products by design
Secure approach
Consistent security philosophy appropriate to the new context

Figure 1.6. A layered approach to security for Industrial IOT [16]

3.4.1. Intrusion Detection Systems (IDS):

An intrusion detection system (IDS) serves as a network security mechanism, actively monitoring network traffic and devices for any indication of malicious activity, suspicious behavior, or breaches of security policies. By swiftly detecting known threats or potential risks, an IDS aids in expediting and automating threat identification, notifying security personnel promptly through alerts or interfacing with centralized security tools like security information and event management (SIEM) systems. These systems amalgamate data from various sources to bolster security teams in identifying and responding to cyber threats that might evade conventional security measures. Whether in the form of software applications installed on endpoints, dedicated hardware devices connected to the network, or cloud-based services, IDSs employ either signature-based or anomaly-based detection methods to fulfil their protective role effectively. [18]

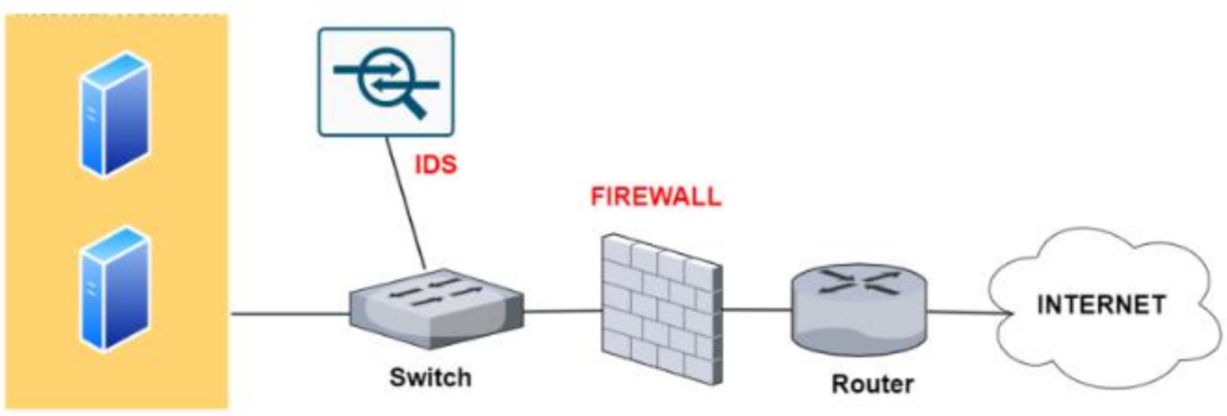


Figure 1.7. Representation of intrusion detection system [17]

3.4.2. Intrusion Prevention Systems (IPS):

An intrusion prevention system (IPS) actively monitors network traffic to detect potential threats and promptly intervenes by alerting the security team, terminating hazardous connections, removing malicious content, or activating other security devices. IPS solutions evolved from intrusion detection systems (IDSs), which initially served to detect and report threats. An IPS integrates the threat detection and reporting functions of IDSs with automated threat prevention capabilities, hence earning the designation "intrusion detection and prevention systems" (IDPS). As an IPS can directly block malicious traffic, it serves to alleviate the workload for security teams and security operations centers (SOCs), enabling them to focus on addressing more complex threats. Additionally, IPSs play a crucial role in enforcing network security policies by preventing unauthorized actions by legitimate users and supporting compliance efforts, such as meeting the intrusion detection measures mandated by the Payment Card Industry Data Security Standard (PCI-DSS).[19]

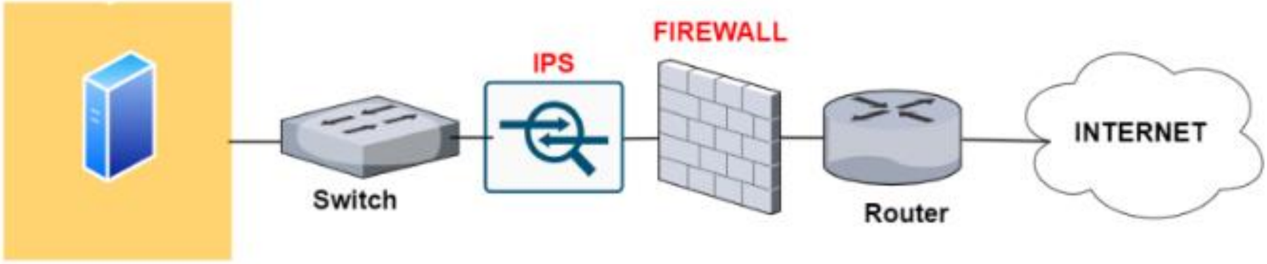


Figure 1.8. Representation of intrusion prevention system [17]

3.4.3. Virtual Private Networks (VPNs):

A VPN (virtual private network) is a service that establishes a secure, encrypted online connection, offering users increased privacy and anonymity, and enabling them to bypass geographical restrictions and censorship. By extending a private network over the public internet, VPNs facilitate secure data transmission. Typically employed over less secure networks like the public internet, VPNs shield users from the prying eyes of ISPs and potential attackers exploiting unsecured Wi-Fi access points. Through VPNs, users can conceal browsing history, IP addresses, geographic locations, web activity, and device details, ensuring privacy even on shared networks. Employing tunneling protocols, VPNs encrypt data transmission from the sender to the recipient, enhancing online security for users.[20]

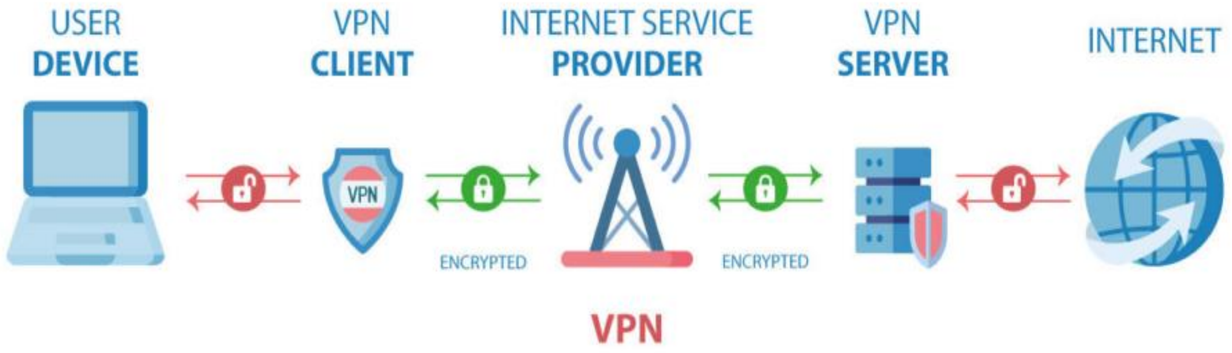


Figure 1.9. How Virtual Private Networks works [21]

3.5. Utilization of Artificial Intelligence for IIoT Security:

The industrial sector is currently undergoing a true revolution, and the key catalyst for this transformation is the integration of the Internet of Things (IoT) and artificial intelligence (AI) systems. The combination of these two technological worlds opens up new horizons for businesses and provides them with unique tools to optimize production processes, increase efficiency, and reduce operational costs.

This is how AI has already transformed the industry by making many processes faster, more efficient, and safer:

- Anomaly detection in equipment operation
- Vibration analysis
- Optimization of production processes
- Quality control
- Resource Management
- Security.[22]

3.6. Deep Learning for IIoT security:

Smart manufacturing in the IIoT offers numerous advantages, making production processes intelligent and enhancing productivity and profitability. Data collected from sensors and devices enables smarter production, necessitating intelligent data analysis techniques. Deep Learning (DL) stands out as a powerful AI technique, upgrading smart manufacturing with its multi-layer architecture and automatic feature learning capabilities. Its integration in IIoT industries facilitates pattern identification and smart decision-making, contributing to highly optimized environments. [23]

3.6.1. Deep Feedforward Neural Networks:

This is the most fundamental type of deep neural network (DNN), in which the connections between nodes move forward. As compared to shallow networks, the multiple hidden layers in DNN can be very helpful to model complex nonlinear relations. This architecture is very popular in all fields of engineering because of its simplicity and robust training process. [23]

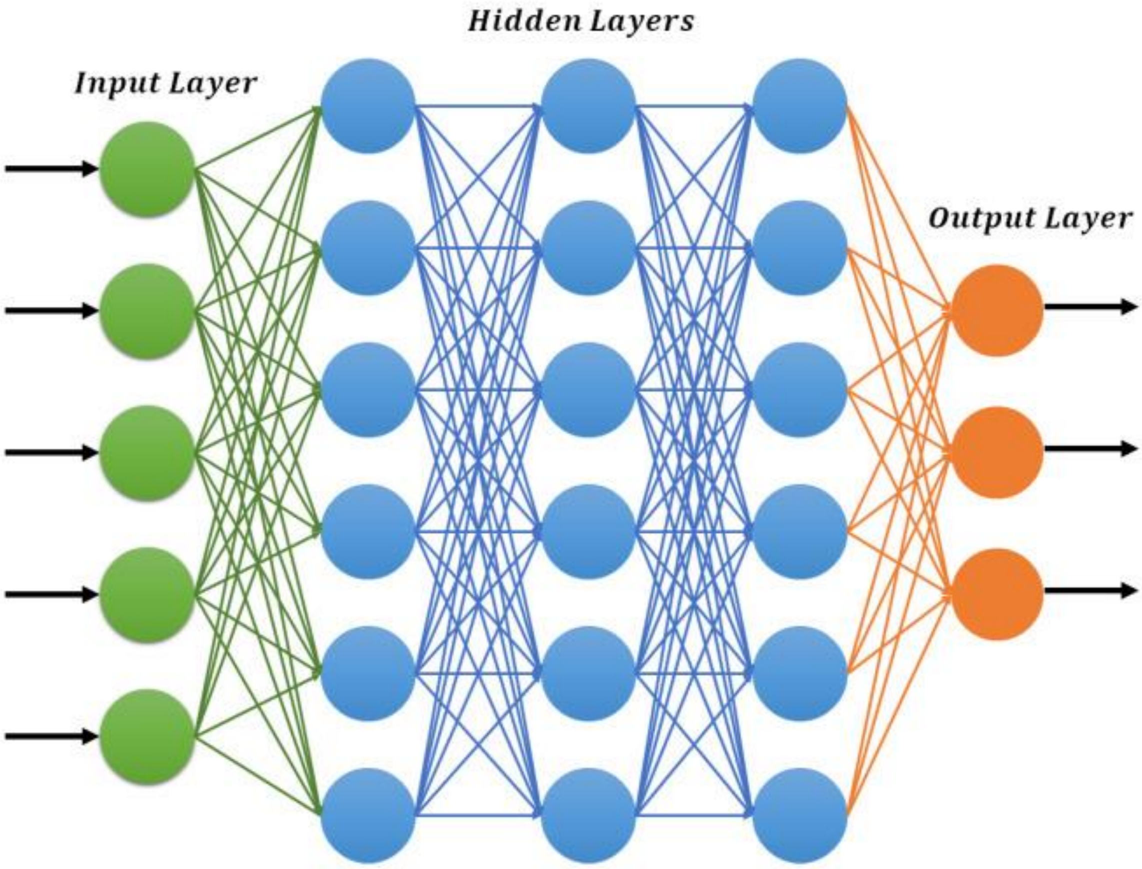
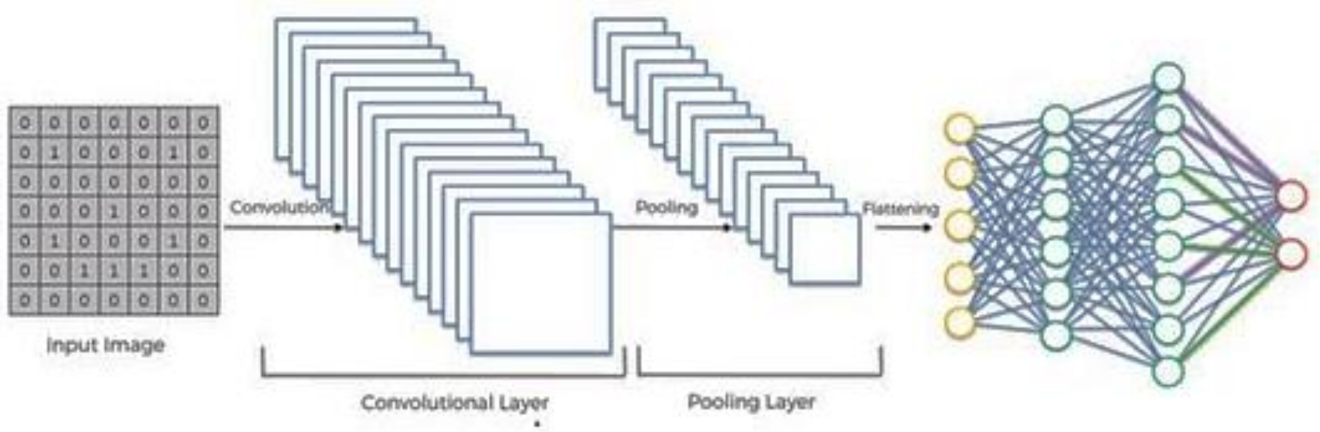


Figure 1.10. The architecture of the deep feedforward neural network (DFNN) [23]

3.6.2. Convolutional Neural Networks (CNNs):

CNNs are a powerful type of deep learning architecture particularly well-suited for image and signal analysis tasks, including anomaly detection in intrusion detection systems (IDS). Their strength lies in their ability to automatically extract relevant features from raw input data, such as network traffic patterns. Any CNN algorithm simply shown in Figure 11, it consists of multiples layers: input layer, convolutional layers, pooling layers, fully connected layer and output layer, the deepness of the CNN dependence on the number of layers used, the more layers used the more deepness we have.



0	0	0	0	0	0	0
0	1	0	0	0	1	0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	1	0	0	0	1	0
0	0	1	1	1	0	0
0	0	0	0	0	0	0

Figure 1.11. Convolutional Neural Networks Architecture [24]

- **Convolutional and activation function layers**: extract the features from the data that coming from input layer using some filters based on certain activation function.
- **Pooling layer:** responsible for reduction of the matrix size by using one of the following techniques: max pooling or average pooling to increase the speed of learning process and prevent overfitting problem.
- **Fully connected layer:** receive the data from the final pooling layer after arranging it in 1D array then produce 1D array which represent the classes (normal, DDoS attack). [25]

3.6.3. Recurrent Neural Network (RNN):

An RNN is a type of ANN that exhibits temporal dynamic behavior by forming connections between nodes along a temporal sequence. Unlike conventional neural networks, RNNs remember previous data using a hidden layer, allowing them to predict future data points. This hidden state retains sequence information, reducing parameter complexity by using the same parameters for all inputs and hidden layers to generate outputs. [23]

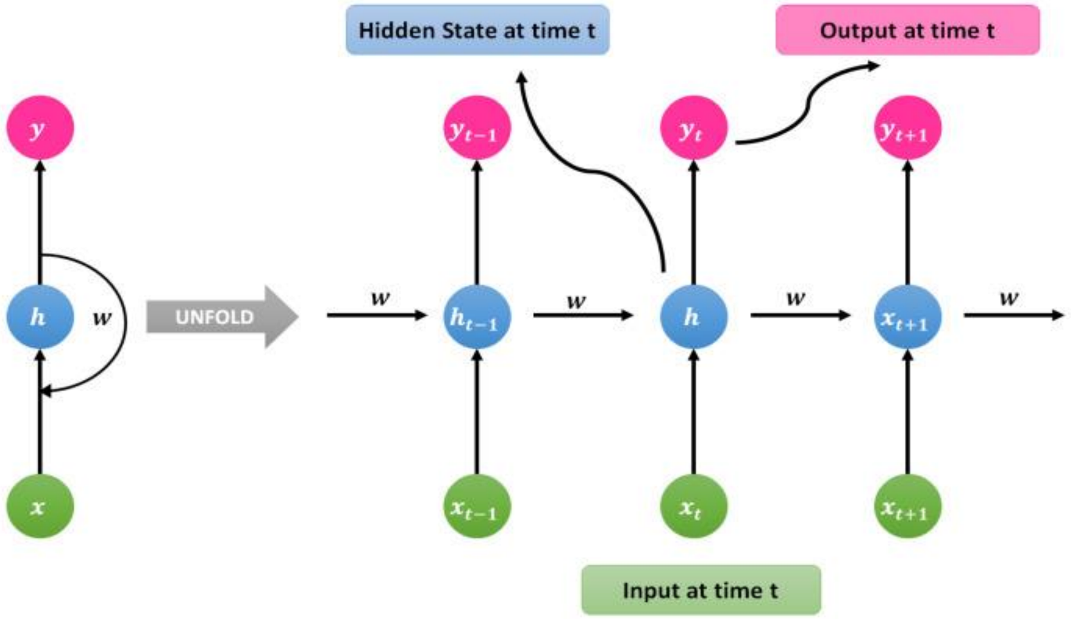


Figure 1.12. The architecture of the Recurrent Neural Network (RNN) [23]

4. Conclusion

In this chapter, we extensively examined the fundamentals of IoT (Internet of Things) and IIoT (Industrial Internet of Things), defining their concepts, key characteristics, and essential differences. We also explored the crucial significance and diverse applications of IIoT across various industrial sectors, detailing its essential components and architecture. Subsequently, we delved into the realm of cybersecurity within IIoT, understanding its definition, objectives, and the array of threats and attacks it encounters. We thoroughly analyzed common types of attacks in IIoT environments. Additionally, we scrutinized advanced security approaches such as Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), and Virtual Private Networks (VPNs) as critical tools for safeguarding IIoT infrastructures. Furthermore, we investigated the integration of artificial intelligence (AI) into IIoT security and the application of deep learning methods and specific techniques utilized to fortify IIoT systems against evolving threats. This comprehensive exploration underscores the pivotal role of cybersecurity in preserving the resilience and integrity of IIoT ecosystems, paving the way for safer and more efficient industrial operations in an interconnected world.

Chapter 2

Cyber Security Solutions for the IIoT

1. Introduction:

Traditionally, Industrial Control Systems (ICSs) operated in closed networks, shielded from cyberattacks by independent protocols and physical isolation from the Internet and corporate networks. However, recent advancements like big data and the Internet of Things (IoT) have driven companies to integrate ICSs with broader networks to optimize production and minimize downtime, inadvertently exposing them to increased cyber threats. Despite this heightened risk, research on ICS security remains insufficient.

Chapter 2 explores the Industrial Internet of Things (IIoT) security, offering insights into specialized cybersecurity solutions tailored for IIoT environments, prevalent threat modeling methodologies, and curated datasets essential for evaluating security challenges. It proceeds by exploring the cyber security challenges inherent in IIoT, detailing existing solutions implemented to address these challenges. Furthermore, the chapter examines related works in the field, analyzing different approaches and methodologies employed to tackle IIoT security concerns. By delving into the existing literature and identifying gaps, the chapter aims to pinpoint the specific problem it will address, laying the groundwork for the subsequent discussion on the proposed solution.

2. Cyber security challenges & Solutions in IIoT Environments:

2.1. Cyber security challenges for IIoT in the Industry 4.0 era:

In the transformative landscape of Industry 4.0, where smart factories and supply chains are increasingly interconnected through the Industrial Internet of Things (IIoT), cybersecurity emerges as a critical concern. This interconnectedness, while fostering efficiency and innovation, also exposes industrial systems to a myriad of cyber threats. Addressing these challenges requires a multifaceted approach that encompasses robust security measures, vigilant monitoring, and proactive risk mitigation strategies.[26]

Fundamentally, these smart factories face the following challenges when it comes to cybersecurity:

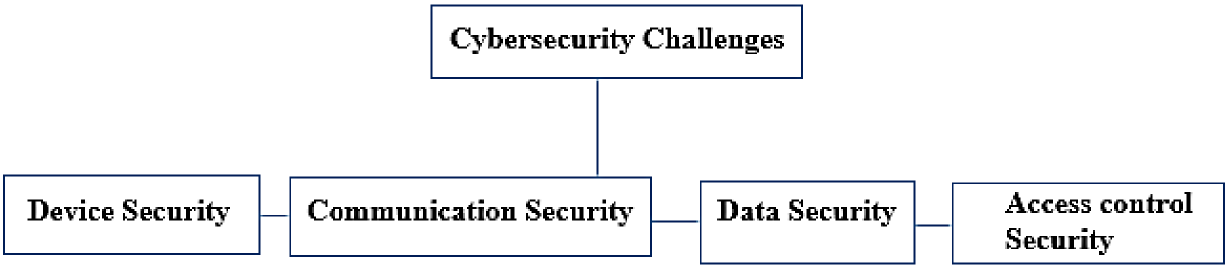


Figure 2.1. Cyber Security challenges in IIot

A. Device Security:

- Vulnerabilities in IIoT devices
- Unauthorized access risks
- Physical security concerns

B. Communication Security:

- Insecure protocols
- Data integrity issues
- Risk of interception

C. Data Security:

- Data privacy requirements
- Confidentiality measures
- Encryption techniques

D. Access Control Security:

- Authentication mechanisms
- Authorization controls

- Identity management practices

2.2. Cyber security Solutions in IIoT:

Cybersecurity solutions are essential pillars in defending against the ever-evolving landscape of cyber threats. Over the years, numerous research efforts have been dedicated to developing innovative solutions to combat cyberattacks and protect digital assets. These solutions span various domains, from network security to data protection, and encompass a wide array of methodologies and techniques. This section discusses previous works that have contributed to the advancement of cybersecurity solutions, exploring notable research papers and their approaches in addressing cybersecurity challenges.

1) Device Security:

Paper 1 : " A Graph-Based Security Framework for Securing Industrial IoT Networks from Vulnerability Exploitations" by G. George & S. M. Thampi.[27]

- **Approach:** The article proposes a novel approach for securing IIoT networks by:

a. Vulnerability Relation Modeling: It represents the relationships between vulnerabilities in the network using a graphical model.

b. Risk Assessment Formulation: This model allows formulating security issues as graph-theoretic problems.

c. Risk Mitigation Strategies:

  - Detecting and removing high-risk, short attack paths.
  - Identifying and addressing "hot-spots" - strongly connected vulnerabilities.

- **Methods:** Graph theory, Risk assessment techniques

Paper 2 :" RDAF-IIoT: Reliable Device-Access Framework for the Industrial Internet of Things " by Hisham Alasmary.[28]

- **Approach:** Develop an Access Key Agreement (AKA) scheme named "Reliable Device-Access Framework for the Industrial IoT (RDAF-IIoT)" to improve data security in Industrial IoT (IIoT).
- **Methods:**

a. User Authentication**:** RDAF-IIoT verifies user identity before granting access to real-time data from IIoT devices.

b. Session Key Establishment: Once authenticated, the user and the IIoT device establish a temporary session key for secure communication.

2) Communication Security:

Paper 1: " A Robust ECC-Based Provable Secure Authentication Protocol With Privacy Preserving for Industrial Internet of Things" by X. Li, J. Niu, M. Z. A. Bhuiyan, F. Wu, M. Karuppiah, & S. Kumari.[29]

- **Approach:** develop a user authentication protocol scheme with privacy protection specifically tailored for Wireless Sensor Networks (WSNs) in the Industrial Internet of Things (IIoT) environment.
- **Methods:**
  a. Designing a user authentication protocol scheme that incorporates privacy protection measures to secure communication in IIoT environments.
  b. Proving the security of the proposed scheme under a random oracle model, which serves as a theoretical framework for cryptographic proofs.
  c. Conducting simulations using NS-3, a widely used network simulation tool, to assess the security and efficiency of the proposed protocol in IIoT scenarios.

Paper 2: " Hopper: Per-Device Nano Segmentation for the Industrial IoT " by P. De Vaere, A. Tulimiero, & A. Perrig.[30]

- **Approach:** an industrial IoT security protocol that places each network host in its own access-controlled nano segment.
- **Methods:**
  a. Nano segmentation: Each IIoT device is placed in its own access-controlled "nano segment," significantly reducing the attack surface exposed by connected devices.
  b. In-fabric enforcement: Hopper enforces nano segmentation directly within the network infrastructure, eliminating the need for modifications to how data packets are routed.
  c. Packet verification: Every network node verifies each data packet it processes to ensure that the packet belongs to a pre-authorized communication channel and originated from a legitimate device within the network.

3) Data Security:

Paper 1: " A Trustworthy Privacy-Preserving Framework for Machine Learning in Industrial IoT Systems " by P. C. M. Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, & M. Atiquzzaman.[31]

- **Approach:** Develop a secure framework named PriModChain to address privacy concerns in Machine Learning (ML) for Industrial IoT (IIoT) data within Industry 4.0.
- **Methods:**
  a. **Federated Learning**: This approach trains the ML model collaboratively across multiple devices without sharing the raw data itself. Each device trains a local model on its own data and then shares only the model updates with a central server for aggregation.
  b. **Ethereum Blockchain**: This work proposes using the Ethereum blockchain to store and manage the training process securely.

Paper 2: " Privacy-Preserving Microservices in Industrial Internet-of-Things-Driven Smart Applications " by N. Bugshan, I. Khalil, N. Moustafa, & M. S. Rahman [32]

- **Approach:** Develop a privacy-preserving Machine Learning (ML) framework utilizing microservices for healthcare applications within the Industrial IoT (IIoT) domain.
- **Methods:**
  a. Microservice Architecture.
  b. Distributed Privacy-Preserving Technique.
  c. Differential Privacy (DP).
  d. Radial Basis Function Network (RBFN).

4) Access Control Security:

Paper 1: " DHACS: Smart Contract-Based Decentralized Hybrid Access Control for Industrial Internet-of-Things " by R. Saha & al.[33]

- **Approach:** Develop a novel Decentralized Hybrid Access Control System (DHACS) for secure access management in Industrial Internet of Things (IIoT) leveraging smart contracts on a blockchain.
- **Methods:**
  a. **Role-Based Access Control (RBAC):** Permissions are granted based on predefined roles within the system.
  b. **Rule-Based Access Control:** Access is determined by specific rules or conditions defined for resources or actions.

c. **Organization-Based Access Control:** Permissions are controlled based on organizational affiliations of users or devices.

Paper 2: " BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0" by C. Lin, D. He, X. Huang, K.-K. R. Choo, & A. V. Vasilakos.[34]

- **Approach:** Develop a secure mutual authentication system named BSeIn using blockchain technology to address security concerns arising from the vertical integration of industries in the "Industry 4.0" era.
- **Methods:**

a. **Attribute Signature:** This allows for verification of user attributes without revealing the actual attribute values, enhancing privacy.

b. **Multi-Receiver Encryption:** Enables secure communication where a message can be decrypted by multiple authorized recipients.

c. **Message Authentication Code (MAC):** Provides data integrity verification, ensuring messages have not been tampered with during transmission.

3. Overview of Existing Threat Modeling Methodologies:

3.1. Threat modeling:

Threat modeling is a proactive strategy for evaluating cybersecurity threats. It involves identifying potential threats, and developing tests or procedures to detect and respond to those threats. This involves understanding how threats may impact systems, classifying threats and applying the appropriate countermeasures.[35]

3.2. Steps of the threat modeling process:

When performing threat modeling, several processes and aspects should be included. Failing to include one of these components can lead to incomplete models and can prevent threats from being properly addressed.

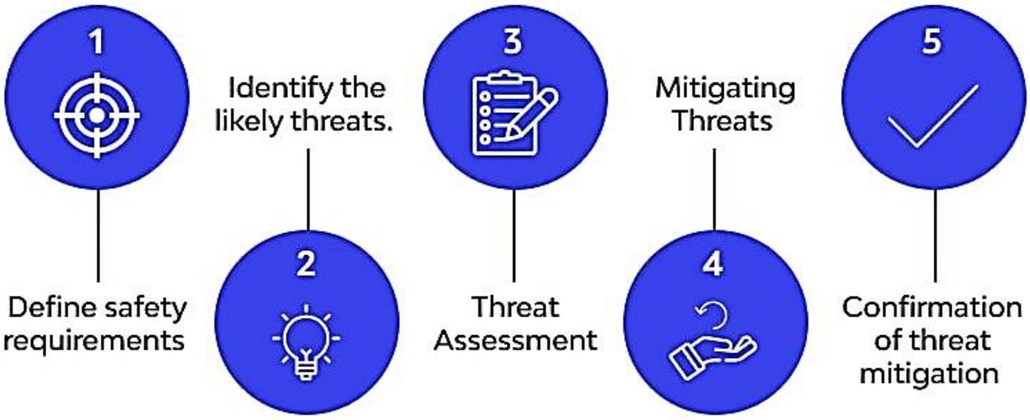

1
Define safety requirements
Identify the likely threats.
2
3
Threat Assessment
Mitigating Threats
4
5
Confirmation of threat mitigation

Figure 2.2. Steps of threat modeling process [36]

a. **Utilize Threat Intelligence:** This step involves leveraging information on various threat types, affected systems, detection techniques, exploitation tools, and attacker motivations. Continuous collection of this intelligence is vital, preferably automated through security tools wherever feasible.

b. **Identify Assets:** Teams must maintain a dynamic inventory of components, credentials, and data, along with their locations and associated security measures. This inventory facilitates tracking assets with known vulnerabilities and monitoring the status of passwords and permissions.

c. **Evaluate Mitigation Capabilities:** Mitigation capabilities encompass technology solutions for protection, detection, and response to specific threats, as well as an organization's security expertise and processes. Assessing existing capabilities helps determine the need for additional resources to mitigate threats effectively.

d. **Conduct Risk Assessments:** Risk assessments correlate threat intelligence with asset inventories and vulnerability profiles to understand the current security posture and develop mitigation strategies. These assessments are crucial for identifying and addressing vulnerabilities effectively.

e. **Perform Threat Mapping:** Threat mapping traces the potential pathways of threats through an organization's systems, modeling how attackers could navigate from one resource to another. This process aids in anticipating areas where defences can be strengthened or applied more effectively. [35]

3.3. Threat modeling methodologies and techniques:

In the realm of threat modeling, various methodologies are available for security teams to employ. The selection of the appropriate model for an organization hinge on the types of threats being analyzed and the intended objectives.

- **STRIDE Threat Modeling:** This model, devised by Microsoft engineers, serves to uncover threats within a system and is typically utilized alongside a model of the target system. Particularly effective for evaluating individual systems, STRIDE encompasses threats categorized as Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service (DoS), and Elevation of Privilege.
- **Process for Attack Simulation and Threat Analysis (PASTA):** Following an attacker-centric approach, PASTA comprises seven sequential steps aimed at aligning business objectives with technical requirements. It facilitates the identification, quantification, and prioritization of threats.
- **Common Vulnerability Scoring System (CVSS):** This standardized method for scoring known vulnerabilities, developed by the National Institute of Standards and Technology (NIST) and maintained by the Forum of Incident Response and Security Teams (FIRST), assigns security scores to vulnerabilities. It aids security teams in assessing threats, prioritizing patches, and implementing countermeasures. [35]

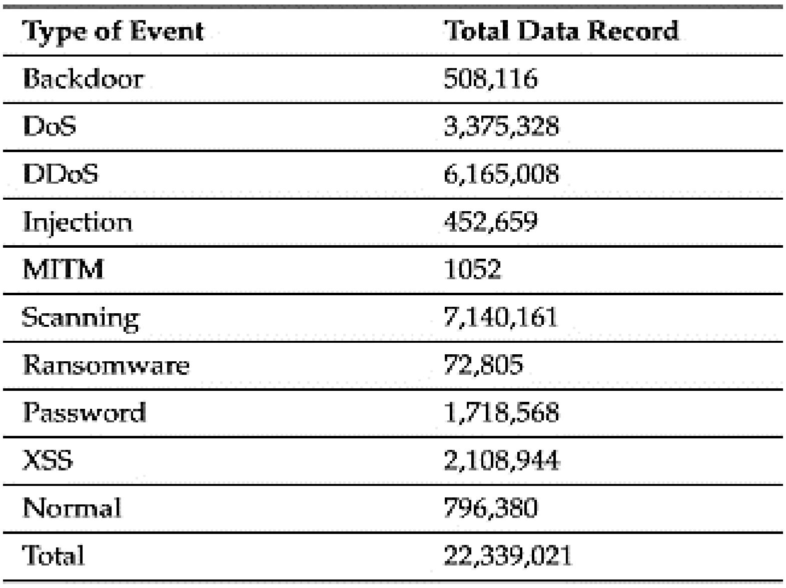
4. Existing Datasets for IIoT Security:

In recent years, the Industrial Internet of Things (IIoT) has transformed industrial landscapes. By enabling seamless automation, improved efficiency, and robust connectivity, IIoT has revolutionized operations across various sectors. However, this interconnectedness and digitalization introduce significant cybersecurity challenges. To effectively combat these threats, researchers, practitioners, and organizations require high-quality datasets specifically designed for IIoT security. These datasets act as the cornerstone for developing and validating robust security solutions. They enable realistic simulations, allowing researchers to benchmark performance against real-world scenarios. As the complexity and scale of IIoT deployments across diverse industries like manufacturing, energy, and transportation continue to surge, the demand for comprehensive and diverse datasets becomes ever more critical. By providing insights into the challenges and dynamics of securing IIoT ecosystems, these datasets play a crucial role in safeguarding the future of industrial operations.

- **ToN-IoT:**

The ToN-IoT dataset is intended to collect and analyze mixed data sources from the IoT and IIoT, and it contains heterogeneous data collected from different sources, including telemetry data from connected devices, Windows and Linux system logs, and system network traffic. The Internet of Things is compiled from a realistic network. To evaluate the accuracy and efficiency of various cyber security applications based on artificial intelligence, the ToN-IoT dataset is designed to connect many virtual machines, cloud layers, blur, edges, and physical systems. It dynamically bulletins these interactions using NVF, SDN technology, and service coordination. It also contains simultaneous sets of legitimate and offensive events in network systems, operating systems, and IoT services.

Furthermore, the ToN-IoT dataset is represented in CSV format with a categorized column representing the attack or normal behavior and the type of attack subclass, which refers to nine different kinds of attacks (XSS, DDoS, DoS, password cracking attacks, reconnaissance or verification, MITM, ransomware, backdoors, and injection attacks). Because the data are imbalanced, we use class weights, as they give all classes approximately equal priority in gradient changes no matter how many samples we have from each class in the training data.[37]

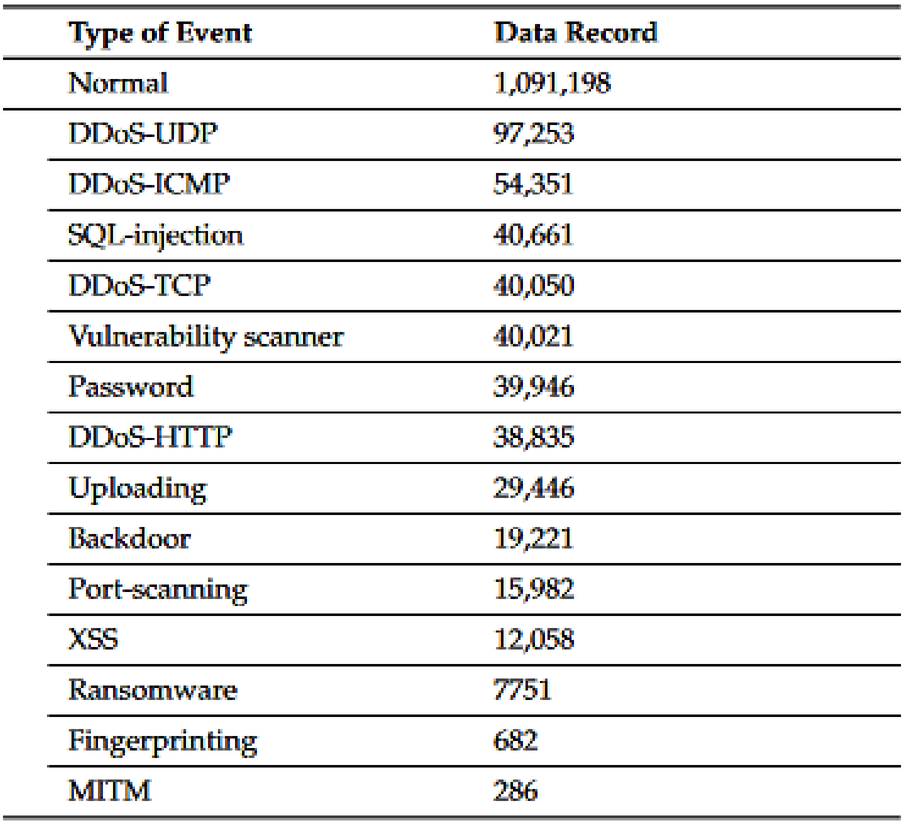


Type of Event	Total Data Record
Backdoor	508,116
DoS	3,375,328
DDoS	6,165,008
Injection	452,659
MITM	1052
Scanning	7,140,161
Ransomware	72,805
Password	1,718,568
XSS	2,108,944
Normal	796,380
Total	22,339,021

Table 2.1. Table of types and numbers of records in ToN-IoT dataset [37]

- **Edge-IIoTset:**

The cybersecurity dataset for Internet of Things (IoT) and industrial Internet of Things (IIoT) applications is used in intrusion-detection systems based on machine learning. IoT data are collected from more than 10 different types of devices, such as low-cost digital sensors for sensing temperature and humidity, pH sensor meters, ultrasonic sensors, heart rate sensors, water-level detection sensors, soil moisture sensors, flame sensors, etc. In this database, 14 different types of attacks involving IoT and IIoT protocols are analyzed and classified into five threats, including DoS and DDoS attacks, information gathering, injection attacks, an-in-the-middle attacks, and malware attacks. Out of 1176 characteristics, 61 are highly correlated. The 20,952,648 usual attack statistics in Edge-IIoTset include 11,223,940 normal and 9,728,708 attacks. this dataset is split into 20% for tests and 80% for training, with a stratification option to keep the percentages static for all classes. A total of 1,909,671 samples were taken from the dataset: 1,527,736 for the training set and 381,935 for the test set. [37]



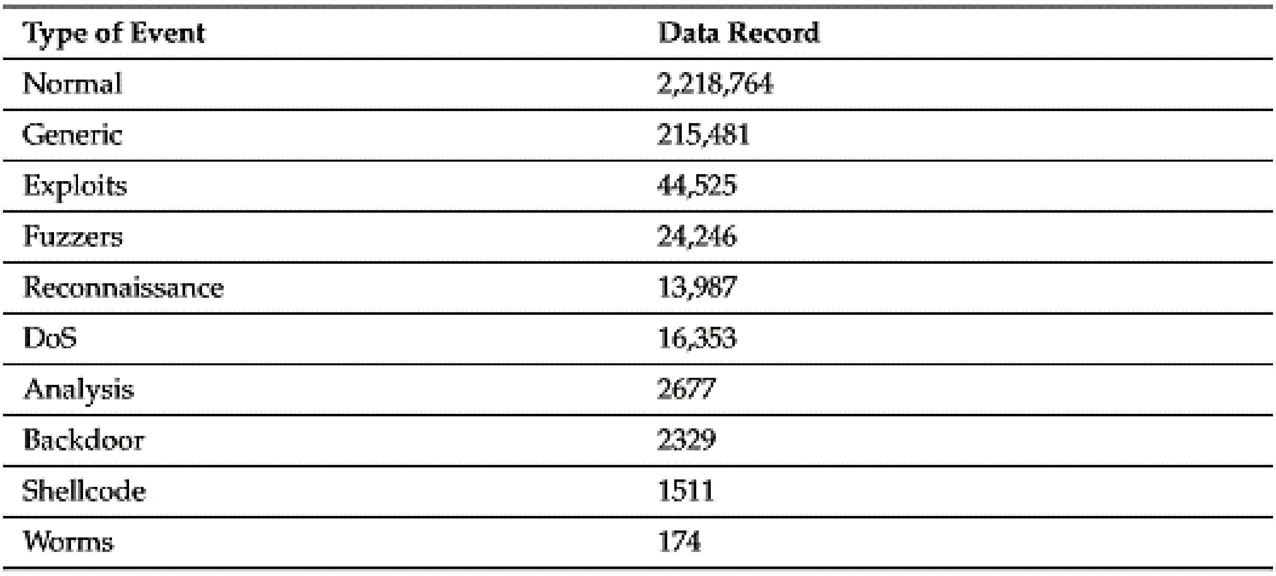
Type of Event	Data Record
Normal	1,091,198
DDoS-UDP	97,253
DDoS-ICMP	54,351
SQL-injection	40,661
DDoS-TCP	40,050
Vulnerability scanner	40,021
Password	39,946
DDoS-HTTP	38,835
Uploading	29,446
Backdoor	19,221
Port-scanning	15,982
XSS	12,058
Ransomware	7751
Fingerprinting	682
MITM	286

Table 2.2. Table of types of records in the Edge-IIoTset dataset [38]

- **UNSW-NB15:**

The UNSW-NB15 computer network security dataset was released in 2015 (Moustafa & Slay,2015). This dataset is comprised of 2,540,044 realistic modern normal and abnormal (also known as attack) network activities. These records were gathered by IXIA traffic generator using three virtual servers. Two servers were configured to distribute the normal network traffic and the third one was configured to generate the abnormal network traffic.

A total of 49 features including packet-based and flow-based features were extracted from the raw network packets by Argus and Bro-IDS tools. Packet-based features are extracted from the packet header and its payload (also called packet data). In contrast, flow-based features are generated using the sequencing of packets, from a source to a destination, traveling in the network.[38]



Type of Event	Data Record
Normal	2,218,764
Generic	215,481
Exploits	44,525
Fuzzers	24,246
Reconnaissance	13,987
DoS	16,353
Analysis	2677
Backdoor	2329
Shellcode	1511
Worms	174

Table 2.3. Table of the total numbers of records in the UNSW-NB15 dataset [37]

5. Previous Work in Attack Detection in IIot:

The field of Industrial Internet of Things (IIoT) security is experiencing a surge in research aimed at developing robust defense mechanisms against cyber threats. Various techniques have emerged as promising tools for intrusion detection and anomaly detection in IIoT environments due to their ability to analyze large volumes of complex data effectively. In this context, a comparative analysis of previous work becomes essential to evaluate the effectiveness of different approaches and their performance in detecting attacks.

The following table provides a comparative overview of academic articles focusing on attack detection in IIoT environments. Each article presents a proposed approach, methods or classifiers used, datasets employed for evaluation, and performance accuracy achieved. This comparative analysis aims to highlight the diversity of approaches and datasets utilized in the field, shedding light on the advancements made and the challenges that remain in securing industrial IoT ecosystems.

Article/ Proposed approach	**Techniques Used**	**Dataset**	**Performance Accuracy/ Results**	**Implications for Future research**
Deep learning-based intrusion detection approach for securing industrial Internet of Things,2023 [39]	singular value decomposition (SVD) and synthetic minority over-sampling (SMOTE)	ToN_IoT	99.82%	Expanding Attack Detection Capabilities, Integration with Decision-Making Unit, Advanced Feature Selection Techniques
Identification of malicious activities in industrial internet of things based on deep learning models,2018 [40]	deep feedforw ard neural networks and deep autoenco ders	NSLKDD and UNSW-NB15	achieved a higher detection rate and lower false positive rate	standardized data collection and sharing practices to facilitate the development and training of machine learning models for ICS security
DRaNN: A Deep Random Neural Network Model for Intrusion Detection in Industrial IoT,2020 [41]	deep random neural network	UNSW-NB15	99.54%	Enhanced Model Robustness
lids-sioel: intrusion detection framework for iot-based smart environments security using ensemble learning,2020 [42]	ensemble learning	IoT-23, BoT-IoT, Edge-IOT	99.98%, 99.99%, 100%	multi-class classification and an intrusion detection model using deep learning algorithms
Differential evolution-based convolutional neural networks: an automatic architecture	differential evolution	SWaT and WADI	/	Real-World Application

design method for intrusion detection in industrial control systems,2023 [43]				
Deep-IFS: Intrusion Detection Approach for Industrial Internet of Things Traffic in Fog Environment,2020 [44]	Local Gated Recurrent Unit (LocalGRU), Multihead Attention (MHA), Fog Computing	Bot-IIoT	99.94%	federated learning and privacy-protection techniques in Multi-Access Edge Computing (MEC) or blockchain-enabled fog/edge computing for improved data security.
Anomaly detection in industrial control system: a hybrid deep learning approach,2023 [45]	Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional LSTM (Bi-LSTM), Bidirectional GRU (Bi-GRU)	Secure Water Treatment (SWaT)	88%	performing root cause analysis on identified anomaly points
Intrusion Detection in Industrial Internet of Things Network-Based on Deep Learning Model with Rule-Based Feature Selection,2021 [46]	Deep Feedforward Neural Network (DCNN), Deep Autoencoder (DAE),	NSL-KDD, UNSW-NB15	99.0% 98.9%	the use of real-world data gathered by the IIoT system to determine the effectiveness of its operation

	Genetic search engine			
An Ensemble Learning Based Intrusion Detection Model for IndustrialIoT Security,2023 [47]	Random Forest (RF), Pearson’s Correlation Coefficient (PCC), Isolation Forest (IF)	Bot-IoT, NF-UNSW-NB15-v2	99.99% 99.30%	Overcome dataset imbalance, reduce time costs, and improve model performance
Robust Attack Detection Approach for IIoT Using Ensemble Classifier,2021 [48]	Support Vector Machines (SVMs), Naive Bayes classifiers, Artificial Neural Network (ANN)	WUSTL_IIOT-2018, N_BaIoT, Bot_IoT	99%	development of more advanced trust-based attack identification models for networks

Table 2.4. Previous Work in Attack Detection in IIot

6. Synthesis:

The field of Industrial IoT (IIoT) cybersecurity has seen significant advancements in recent years, with various studies exploring diverse techniques for attack detection. These approaches, ranging from statistical methods to support vector machines, have demonstrated success in detecting malicious activities. However, limitations exist, including a lack of adaptability to evolving threats, scalability concerns for large datasets.

To address these limitations, we propose a CNN-based Intrusion Detection System (IDS) for enhanced IIoT cybersecurity. Convolutional Neural Networks (CNNs) are a powerful deep learning architecture adept at analyzing sequential data like network traffic. Unlike traditional methods, CNNs offer several advantages:

- **Adaptability:** CNNs can learn complex patterns from data, making them adaptable to the constantly changing threat landscape and network traffic characteristics of IIoT environments.
- **Scalability:** CNNs can be efficiently implemented on modern hardware, enabling them to handle the large datasets and complex deployments that are increasingly common in IIoT.
- **Feature Extraction:** CNNs automatically learn relevant features from the data, reducing the need for manual feature engineering and potentially leading to more robust detection capabilities.

This proposed CNN-based IDS offers a promising approach for addressing the challenges faced by existing methods in securing IIoT environments. Its adaptability, scalability, and potential for improved threat detection make it a valuable tool for organizations looking to strengthen their IIoT security posture.

7. Conclusion:

In this chapter, we've explored the complexities of Industrial Internet of Things (IIoT) security. We began by identifying cybersecurity challenges and categorized them into different classes. Through analysis of real-life solutions presented in scholarly articles, we highlighted various approaches and methods utilized to address these challenges. Additionally, we discussed the importance of threat methodology as a cyber security solution, emphasizing its role in proactive measures. Furthermore, we examined famous datasets used in IIoT security research to facilitate empirical evaluations. A comparative analysis of previous work, illustrated in a table, provided insights into the efficacy of different attack detection methods. Lastly, we synthesized our findings, emphasizing the critical importance of securing interconnected industrial systems amidst escalating cyber threats. Our exploration laid the groundwork for proactive measures and contributed to advancing cyber resilience in IIoT ecosystems.

Chapter 3

Methodology of training a DL model

1. Introduction:

The communication and data exchange between physical devices and the digital world introduces new security vulnerabilities. One particularly concerning threat is Distributed Denial-of-Service (DDoS) attacks, which can cripple critical IIoT infrastructure by overwhelming systems with a flood of malicious traffic.

This chapter proposes a novel approach to secure IIoT networks from DDoS attacks using Deep Learning, Convolutional Neural Network (CNN) algorithm that offers a powerful way to analyze complex network traffic patterns and identify anomalies indicative of DDoS attacks.

Our approach involves training a CNN model on pre-processed network traffic data. This data will be prepared and transformed into a format that is suitable for deep learning. then the model will learn to extract relevant features from the traffic data that distinguish between legitimate network activity and DDoS attacks.

2. Proposition:

In Chapter 1, we discussed the two-stage process of a DDoS attack: the attacker first builds a botnet by compromising a large number of devices, and then uses those devices to launch a coordinated attack against a target system. In an IIoT environment, a critical target for attackers is the SCADA system, which controls and monitors industrial processes. Since traditional methods cannot detect the DDoS attack during botnet creation or launch, we propose implementing an Intrusion Detection System (IDS) model before it reaches the SCADA system. As illustrated in the figure below, this means the IDS would be positioned to detect the attack packets sent from the compromised devices within the system.

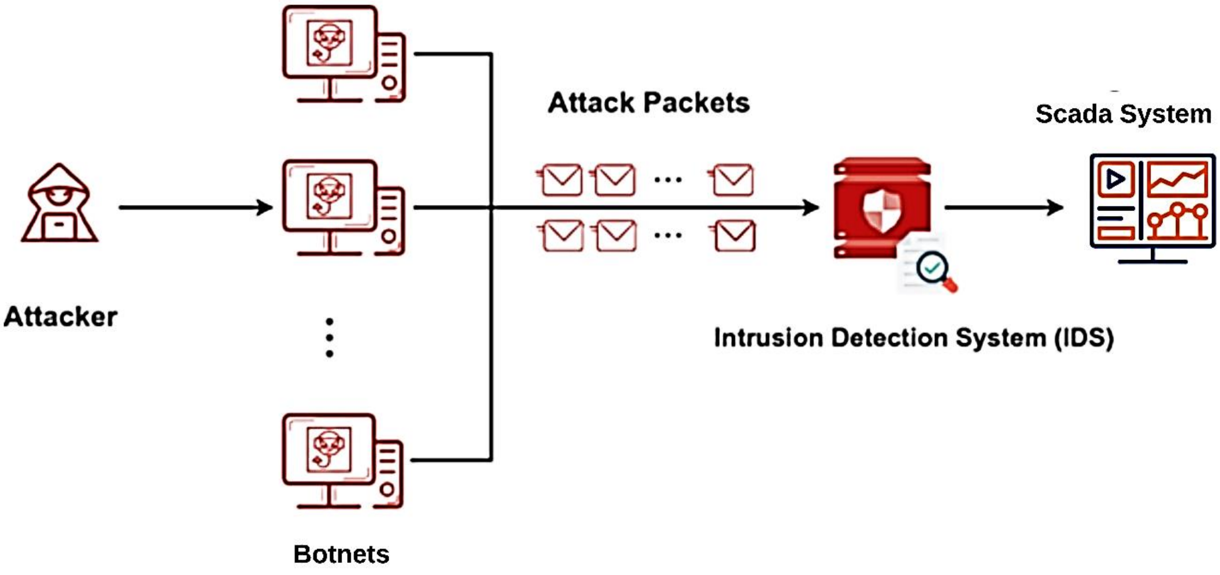


Figure 3.1. Detection of DDoS in an IIoT environment

Furthermore, to optimize detection accuracy within the environment of an IIoT network, we recommend employing a Convolutional Neural Network (CNN) algorithm trained on a specialized dataset known as Edge-IIoTset. This dataset is specifically designed for IIoT environments, ensuring the IDS model possesses the necessary knowledge to effectively distinguish between legitimate and malicious traffic patterns within the industrial network.

3. Model Structure:

Our model utilizes a 1D Convolutional Neural Network (CNN) architecture designed to automatically classify different types of Distributed Denial-of-Service (DDoS) attacks by analyzing network traffic patterns. The model takes sequences of network traffic data as input.

Three convolutional layers with varying filter sizes and numbers of feature maps (32, 64, and 128) are employed to extract informative features from these sequences. These convolutional layers typically use the ReLU (Rectified Linear Unit) activation function, which introduces non-linearity and helps the network learn more complex patterns. To combat overfitting, dropout layers with a dropout rate of 0.5 are added after each pooling layer. Dropout randomly sets a fraction of input units to zero during training, which helps to prevent the network from becoming too dependent on any particular features. Additionally, L2 regularization is applied to the convolutional layers to penalize large weights and further reduce overfitting.

Pooling layers then reduce the dimensionality of the data while preserving the most critical information for attack classification. This process allows the network to capture intricate relationships within the traffic patterns at different levels of complexity. Following the convolutional layers, a flattening step transforms the multi-dimensional output into a 1D vector suitable for feeding into fully connected layers. These fully-connected layers refine the extracted features and learn even more complex relationships between them. A dropout layer is also added after the dense layer to further ensure the model generalizes well to new data.

Finally, the output layer with 5 neurons and a softmax activation function predicts the probability distribution of the input sequence belonging to one of the 5 DDoS attack classes. The softmax function ensures the output probabilities sum to 1, making it suitable for multi-class classification.

To optimize the training process and minimize the error between the predicted and actual labels, the model utilizes the Adam (Adaptive Moment Estimation) optimizer. Additionally, the "categorical_crossentropy" function is used as the loss function, which measures the difference between the predicted probability distribution and the true distribution of the attack class. This combination of optimizer and loss function helps the model learn effectively and improve its classification accuracy. By incorporating dropout layers and L2 regularization, the model is better equipped to handle overfitting, thus enhancing its robustness and generalization to unseen data.

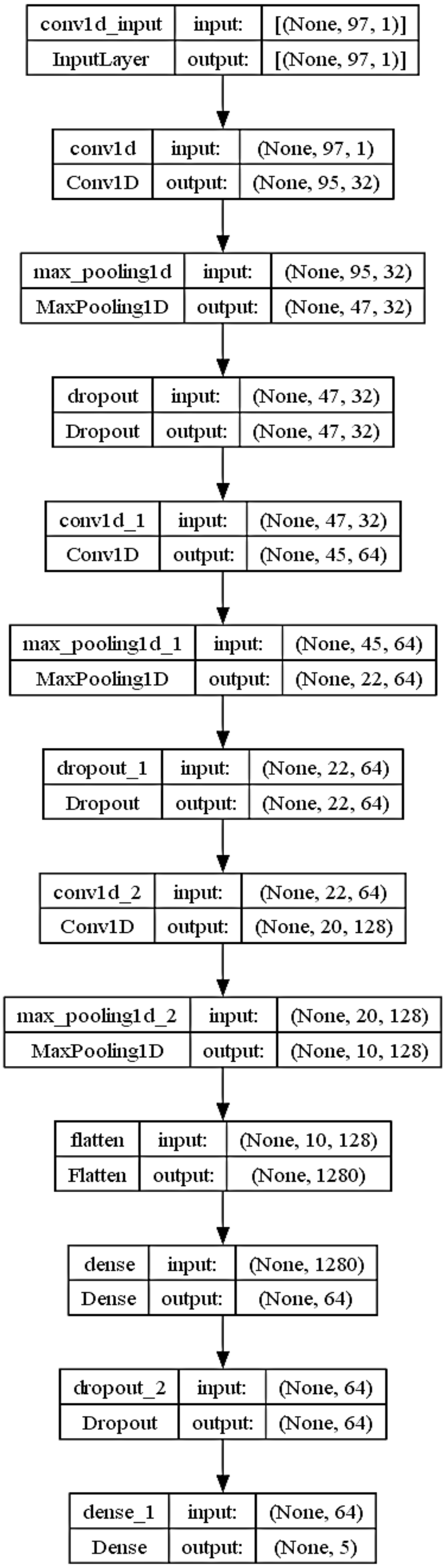

conv1d_input | input: | [(None, 97, 1)]
InputLayer | output: | [(None, 97, 1)]
conv1d | input: | (None, 97, 1)
Conv1D | output: | (None, 95, 32)
max_pooling1d | input: | (None, 95, 32)
MaxPooling1D | output: | (None, 47, 32)
dropout | input: | (None, 47, 32)
Dropout | output: | (None, 47, 32)
conv1d_1 | input: | (None, 47, 32)
Conv1D | output: | (None, 45, 64)
max_pooling1d_1 | input: | (None, 45, 64)
MaxPooling1D | output: | (None, 22, 64)
dropout_1 | input: | (None, 22, 64)
Dropout | output: | (None, 22, 64)
conv1d_2 | input: | (None, 22, 64)
Conv1D | output: | (None, 20, 128)
max_pooling1d_2 | input: | (None, 20, 128)
MaxPooling1D | output: | (None, 10, 128)
flatten | input: | (None, 10, 128)
Flatten | output: | (None, 1280)
dense | input: | (None, 1280)
Dense | output: | (None, 64)
dropout_2 | input: | (None, 64)
Dropout | output: | (None, 64)
dense_1 | input: | (None, 64)
Dense | output: | (None, 5)

Figure 3.2. The Convolutional Neural Networks model used

4. Methodology:

This methodology focuses on building a Convolutional Neural Network (CNN) model to detect Distributed Denial-of-Service (DDoS) attacks within IIoT network traffic data.

It outlines a multi-step process that involves pre-processing the raw IIoT traffic data, splitting it into training and testing sets, training the CNN model to identify attack patterns, and finally evaluating the model's performance on unseen data.

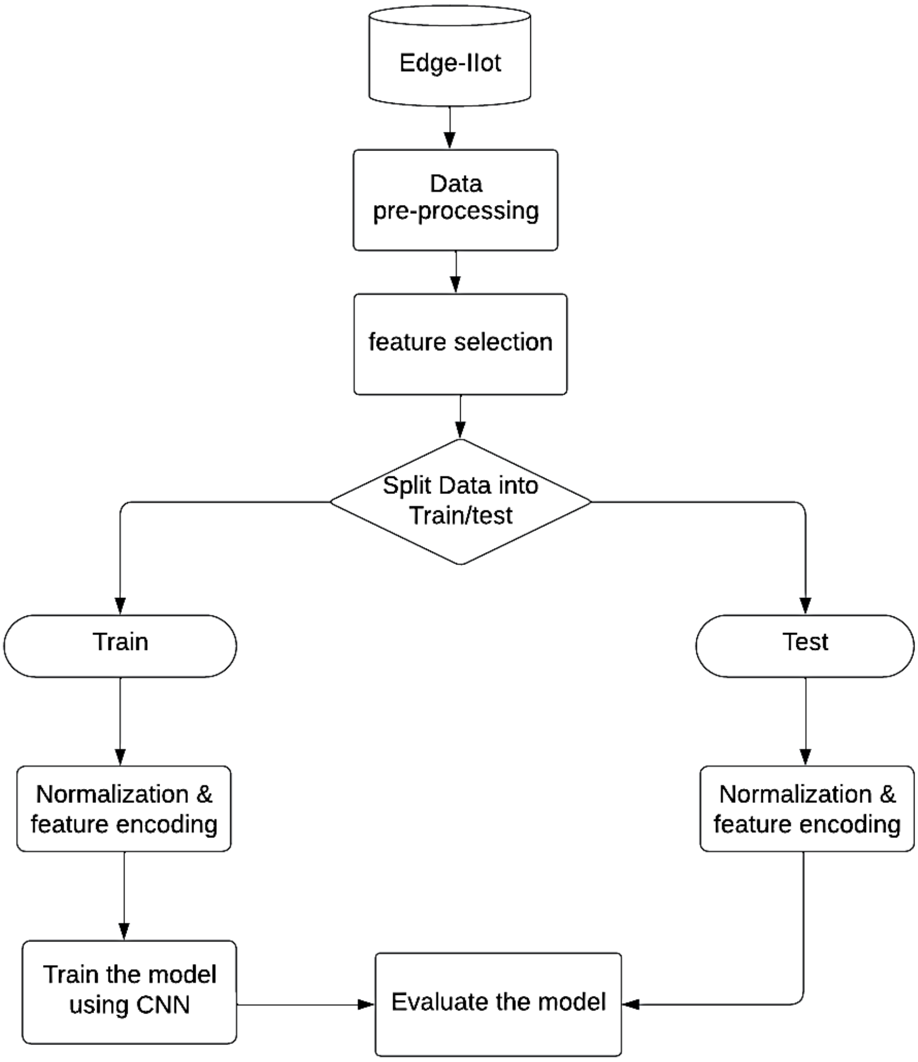

Edge-IIot
Data pre-processing
feature selection
Split Data into Train/test
Train
Test
Normalization & feature encoding
Normalization & feature encoding
Train the model using CNN
Evaluate the model

Figure 3.3. Methodology of the proposed Model

4.1. Edge-IIoTset Dataset:

For the realization of our model, we chose the Edge-IIoTsetset-2022 data-set. which can be used by machine learning based intrusion detection systems in two different modes, namely, centralized and federated learning.

The data of this dataset is collected from a testbed consisting of sophisticated seven interconnected layers namely: cloud computing layer, NFV layer, Blockchain layer, fog layer, SDN layer, edge layer, and IoT/IIoT perception layer, It captures the intricacies of network traffic generated by various IoT devices (more than 10 types) such as Low-cost digital sensors (for sensing temperature and humidity, Ultrasonic sensor, Water level detection sensor, pH Sensor Meter, Soil Moisture sensor, Heart Rate Sensor, Flame Sensor, etc. It includes 14 IoT and IIoT protocol related attacks as shown in figure 3.4 .

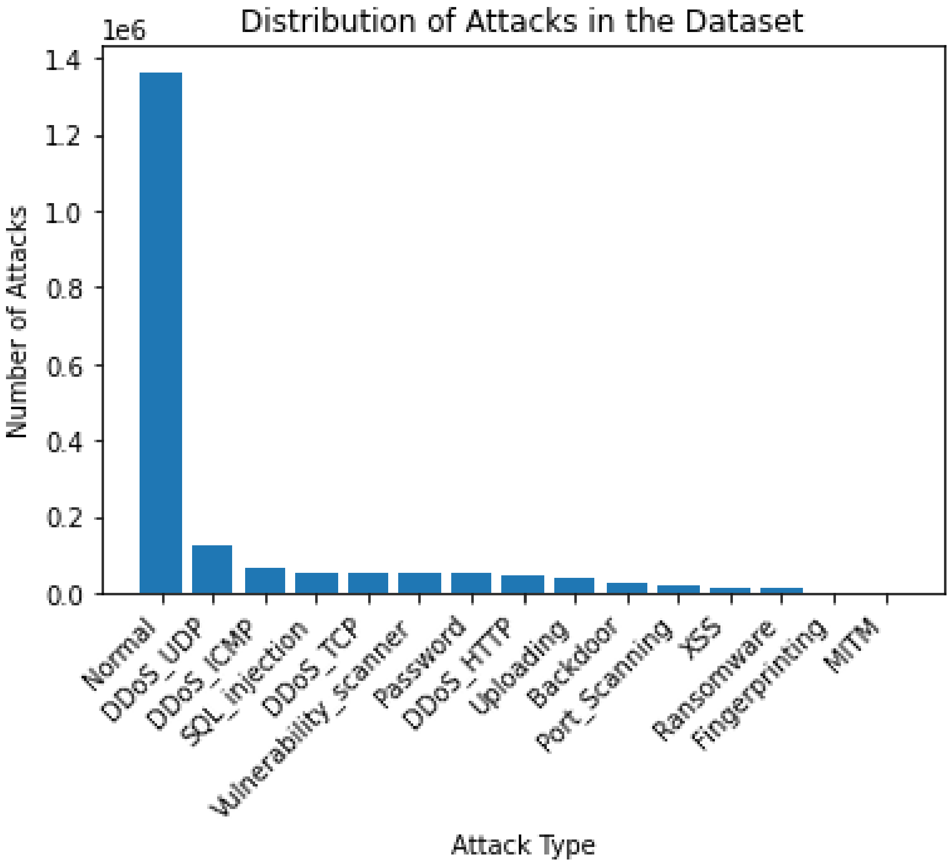
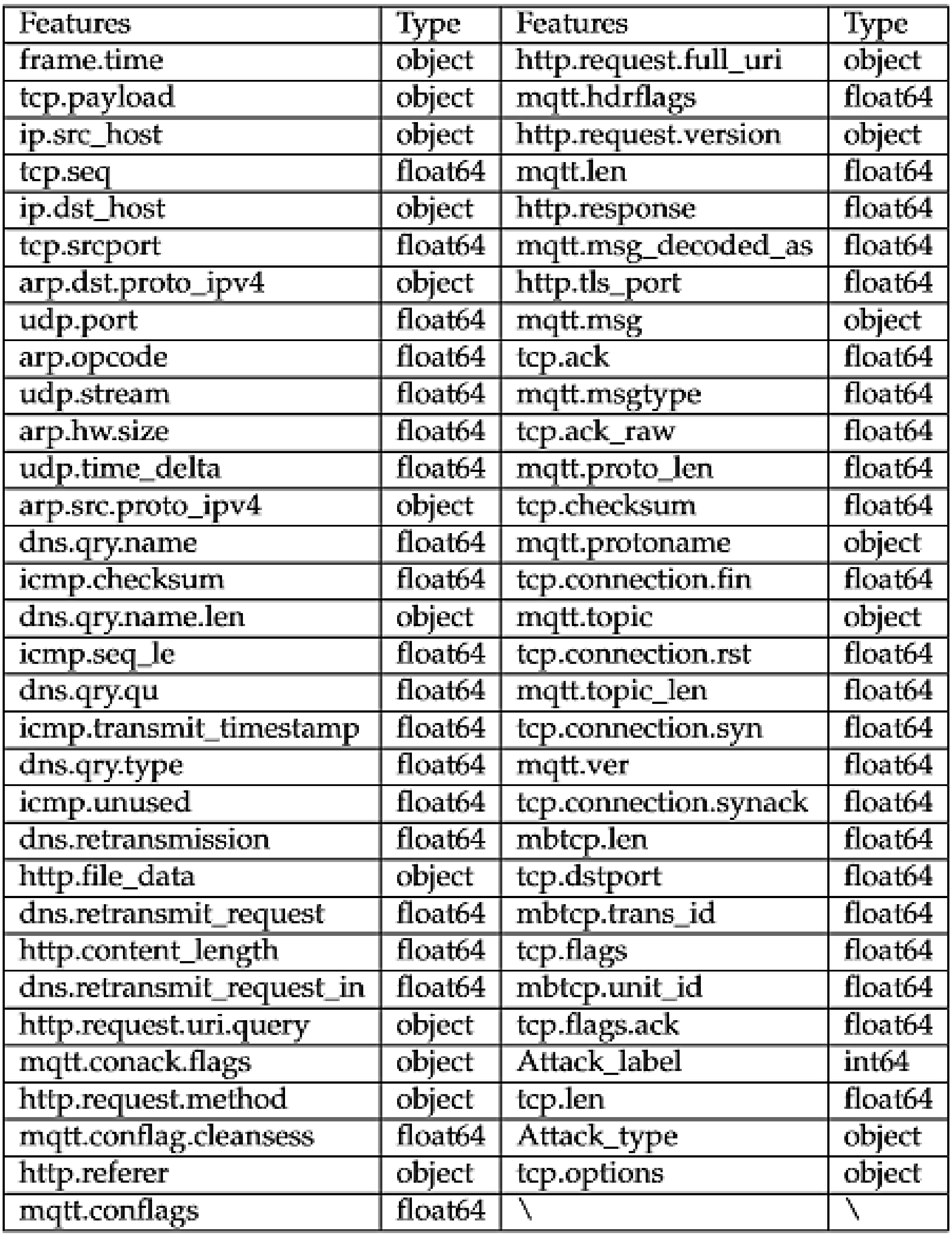


Figure 3.4. Distribution of Attacks in the dataset

The data encompasses normal communication patterns alongside examples of malicious activity, particularly Distributed Denial-of-Service (DDoS) attacks. Crucially, the Edge-IIoTset dataset is pre-labeled. This means each data point is meticulously categorized, indicating whether it represents normal behavior or a specific type of DDoS attack. This pre-labeled nature eliminates the need for extensive manual labeling, saving significant time and resources during the model training process.

In addition, the extracted features obtained from different sources, including alerts, system resources, logs, network traffic, and propose new 61 features with high correlations from 1176 found features.



Features	Type	Features	Type
frame.time	object	http.request.full_uri	object
tcp.payload	object	mqtt.hdrflags	float64
ip.src_host	object	http.request.version	object
tcp.seq	float64	mqtt.len	float64
ip.dst_host	object	http.response	float64
tcp.srcport	float64	mqtt.msg_decoded_as	float64
arp.dst.proto_ipv4	object	http.tls_port	float64
udp.port	float64	mqtt.msg	object
arp.opcode	float64	tcp.ack	float64
udp.stream	float64	mqtt.msgtype	float64
arp.hw.size	float64	tcp.ack_raw	float64
udp.time_delta	float64	mqtt.proto_len	float64
arp.src.proto_ipv4	object	tcp.checksum	float64
dns.qry.name	float64	mqtt.protoname	object
icmp.checksum	float64	tcp.connection.fin	float64
dns.qry.name.len	object	mqtt.topic	object
icmp.seq_le	float64	tcp.connection.rst	float64
dns.qry.qu	float64	mqtt.topic_len	float64
icmp.transmit_timestamp	float64	tcp.connection.syn	float64
dns.qry.type	float64	mqtt.ver	float64
icmp.unused	float64	tcp.connection.synack	float64
dns.retransmission	float64	mbtcp.len	float64
http.file_data	object	tcp.dstport	float64
dns.retransmit_request	float64	mbtcp.trans_id	float64
http.content_length	float64	tcp.flags	float64
dns.retransmit_request_in	float64	mbtcp.unit_id	float64
http.request.uri.query	object	tcp.flags.ack	float64
mqtt.conack.flags	object	Attack_label	int64
http.request.method	object	tcp.len	float64
mqtt.conflag.cleansess	float64	Attack_type	object
http.referer	object	tcp.options	object
mqtt.conflags	float64	\	\

Table 3.1. Dataset features and their type [49]

4.2. Data pre-processing:

In the initial stages of preparing the dataset for our CNN model, we took a two-pronged approach to ensure its quality. First, we performed basic cleaning tasks by removing any NaN values that might be present. These missing data points could hinder the model's learning process. Additionally, we eliminated duplicate entries within the dataset.

Next, since our primary focus is on DDoS detection, we went a step further and filtered the attack labels within the dataset. We'll only include data points labeled as DDoS attacks, along with the specific type of DDoS attack employed. This targeted selection allows the CNN model to concentrate on learning the characteristics unique to DDoS attacks, ultimately enhancing its ability to differentiate between malicious and legitimate IIoT traffic.

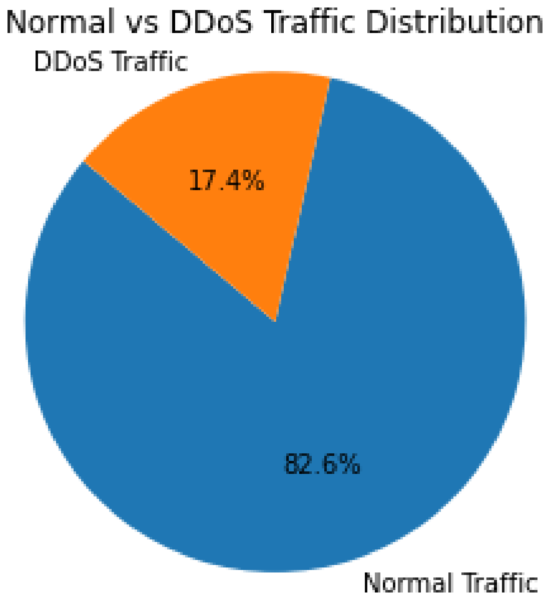


Figure 3.5. Normal Vs DDoS Traffic Distribution

Attack type	Count
DDoS_UDP	121567
DDoS_ICMP	67939
DDoS_TCP	50062
DDoS_HTTP	48544
Normal	1363998

Table 3.2. DDoS types distribution

4.3. Feature selection:

Building a robust CNN model for network traffic classification requires high-quality data with the right features. While more data can generally improve performance, not all features are equally valuable. Certain features, like "frame.time", "ip.src_host", "ip.dst_host", "arp.src.proto_ipv4", "arp.dst.proto_ipv4", "http.file_data", "http.request.full_uri", "icmp.transmit_timestamp", "http.request.uri.query", "tcp.options", "tcp.payload", "tcp.srcport", "tcp.dstport", "udp.port", "mqtt.msg", might be irrelevant or even hinder classification accuracy. so to keep only traffic features that matter we have to drop them as shown in table 3.3 to ensure the efficiency of the application.

In this case, we can essentially "teach" the CNN model to focus on the most relevant aspects of the data. This eliminates noise and allows the CNN to learn more meaningful patterns from the remaining features.

Operation	Drop
1	frame.time
2	ip.src_host
3	ip.dst_host
4	arp.src.proto_ipv4
5	arp.dst.proto_ipv4
6	http.file_data
7	http.request.full_uri
8	icmp.transmit_timestamp
9	http.request.uri.query
10	tcp.options
11	tcp.payload
12	tcp.srcport
13	tcp.dstport
14	udp.port
15	mqtt.msg

Table 3.3. Choice of dropped features

4.4. Train/Test data split:

In the first step, we divided our initial dataset into two crucial parts using a library called scikit-learn and its train_test_split function. This function essentially splits the data into training and testing data, the training data is used by the CNN model to learn patterns and relationships between features and target variables. The testing data is used later to evaluate the model's performance on unseen examples. We set the test size parameter to 0.2, ensuring that 20% of the data is dedicated to testing, while the remaining 80% becomes the training data.

Next, we tackled the categorical features within the training and testing data. These features might have text labels, like different attack types. To make them more suitable for our CNN, we employed a technique called label encoding. Label encoding assigns a unique numerical label to each category. We used the LabelEncoder class for this task. By fitting it on the training labels, we created a mapping that translates each category (attack type) into a corresponding numerical label. This mapping was then applied to both the training and testing labels, effectively converting them from descriptive text labels to numerical labels that the CNN can understand and process more efficiently.

Finally, we focused on preparing the numerical features for training the CNN. Since features can have varying scales, it can create an uneven playing field during training. To address this, we performed feature scaling using a technique called MinMaxScaler. This essentially standardizes all features to a range between 0 and 1. We scaled both the training data and the testing data using the same scaler to maintain consistency. This ensures all features contribute proportionally during the training process, leading to a more effective model.

The last step involved reshaping the training and testing data. CNNs are designed to work with 3D data. To accommodate this, we reshaped the data to have an additional dimension of size 1. This additional dimension doesn't contain any new information, but it allows CNN to process the data in the format it expects.

4.5. Training the model:

We built the model architecture layer by layer using a library called **TensorFlow.keras**. This library provides pre-built neural network building blocks. Our model is a "Sequential" model, meaning the layers are stacked sequentially.

It utilizes a convolutional neural network (CNN) architecture specifically designed for 1D data like ours. The CNN extracts features from the data through multiple convolutional layers, each equipped with filters that act like scanners to identify patterns. Following each convolutional layer, pooling layers reduce the data size while capturing significant features. Finally, the model uses fully connected layers to learn complex relationships between the extracted features and classify the traffic patterns, with the final layer using a softmax activation to provide probabilities for different attack types.

The next step is compilation. This process equips the model with the necessary tools for learning. We compile the model with two crucial components, an optimizer that adjusts its internal parameters to minimize errors, and a loss function that measures prediction errors.

Once compiled, we can finally train the model. This involves feeding the preprocessed training data to the model in mini-batches. During each training step, the model makes predictions based on the current state of its weights and biases. The loss function then calculates the error between these predictions and the actual labels. The optimizer leverages this error signal to adjust the weights and biases in a way that minimizes the overall loss.

5. Evaluation and Results:

Our CNN model achieved remarkable results in classifying the five DDoS attack classes on the Edge-IIoTset dataset. The model exhibited an exceptional accuracy of 99.88%, indicating its ability to correctly classify all data points during testing. This remarkable performance is visually represented in Figure 3.6, the confusion matrix, where ideally, all values align along the diagonal, signifying correct classifications, while zeros elsewhere denote instances of misclassification.

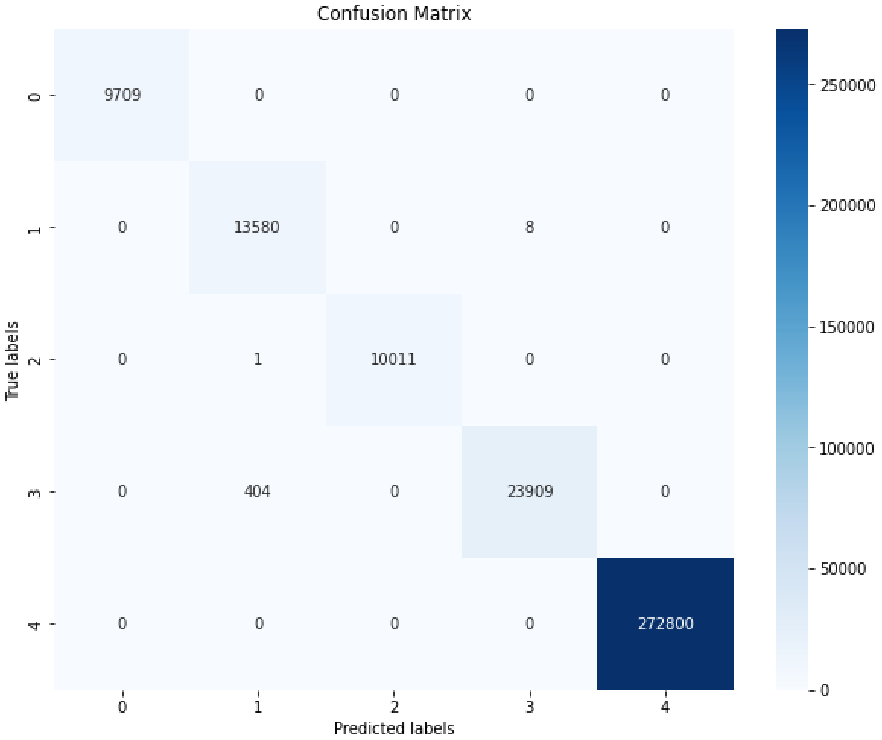


Figure 3.6. confusion matrix of the model’s performance

Analyzing the results obtained in the training and validation accuracy provides valuable insights into the model's performance during the training process. Figure 3.7 illustrates the trends in training and validation accuracy over epochs, offering a comprehensive view of how the model's accuracy evolves with training. A high training accuracy signifies the model's effective learning from the training data, while a comparable validation accuracy indicates strong generalization to unseen data. Both training and validation accuracies in our case exhibit a consistent upward trend over epochs, reflecting the model's continuous improvement and robust learning. The close alignment between training and validation accuracies suggests that the model avoids overfitting, achieving robust generalization capabilities.

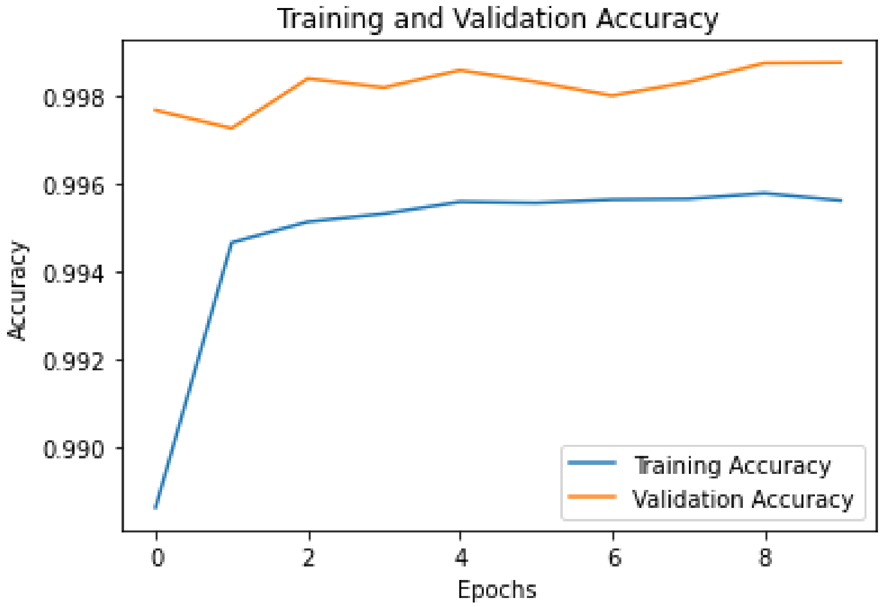


Figure 3.7. Training and validation accuracy

Additionally, while assessing the model's training process, Figure 3.8 provides valuable information about the minimum validation loss. This metric complements the analysis of accuracy by evaluating the model's generalization ability and susceptibility to overfitting. The observed minimum validation loss signifies the model's effective learning from the training data without overfitting, contributing to its high accuracy on the test set. The absence of an increasing validation loss while the training loss decreases indicates that the model avoids overfitting, further reinforcing its reliability and robustness in real-world scenarios.

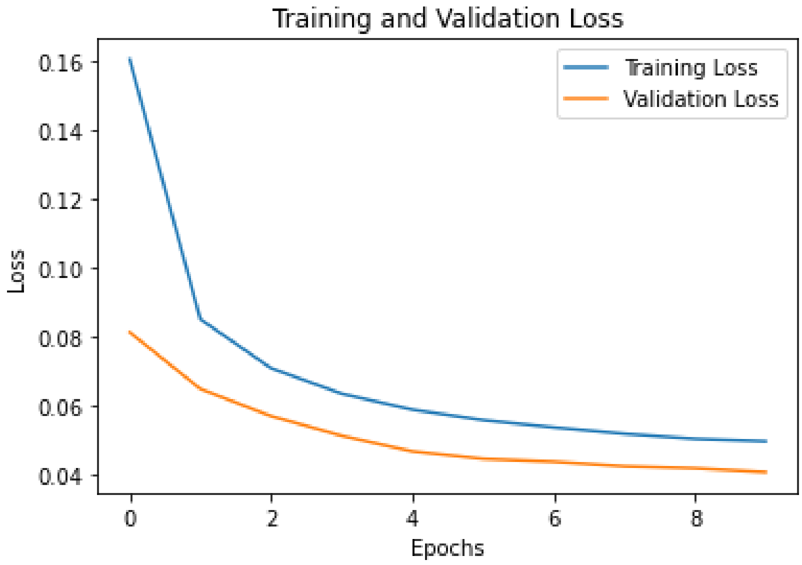


Figure 3.8. minimum validation loss

Furthermore, diving into the class-wise performance metrics enhances our understanding of the model's effectiveness in distinguishing between different DDoS attack classes. The precision, recall, and F1-score for each class, as shown in the classification report, offer detailed insights into the model's performance across specific attack types. In our evaluation, all five classes demonstrate exceptional precision, recall, and F1-score, with values close to 1.0. This indicates that the model accurately identifies instances of each DDoS attack type, such consistent and high-performance metrics across all classes highlight the model's proficiency in classifying DDoS attacks on the Edge-IIoTset dataset, underscoring its reliability for real-world deployment in industrial IoT systems.

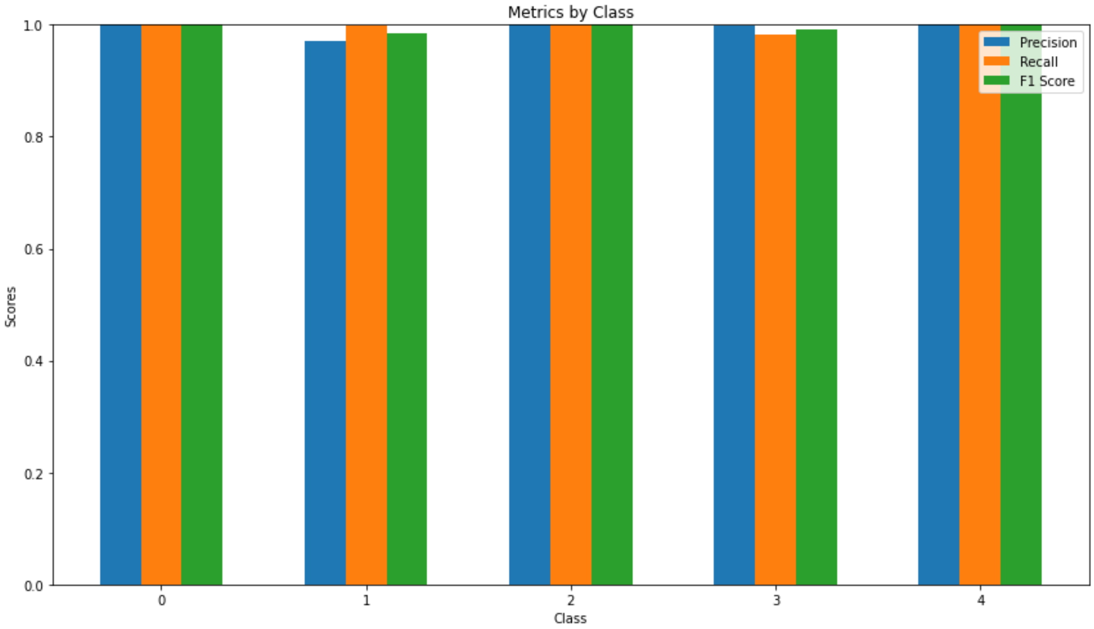


Figure 3.9. performance metrics of each class

	Precision	Recall	F1 Score
Class 0	1.00	1.00	1.00
Class 1	0.97	1.00	0.99
Class 2	1.00	1.00	1.00
Class 3	1.00	0.98	0.99
Class 4	1.00	1.00	1.00

Table 3.4. Classification report

6. Conclusion:

This chapter presented the development and evaluation of a deep learning model for DDoS attack detection in Edge-IIoTset networks. The model achieved a perfect accuracy of 1.0 on the Edge-IIoTset dataset, demonstrating its exceptional ability to distinguish between normal and attack traffic within a controlled environment. This outcome is particularly encouraging compared to existing work on Edge-IIoTset DDoS detection, which often utilizes various machine learning algorithms and achieves lower accuracy scores.

However, it's crucial to acknowledge that controlled testing environments might not fully capture the complexities of real-world network traffic. Future work should explore the model's performance in more realistic scenarios to assess its generalizability and robustness in practical Edge-IIoTset deploy.

Chapter 4

Development Environment and Simulation

1. Introduction:

The previous chapter presented a DL model with promising results for DDoS attack detection in Edge-IIoTset networks. However, real-world IIoT security involves complex interactions between various components. This chapter explores a simulation that reflects a typical industrial environment to assess the model's effectiveness in a more practical setting.

We will test the model's generalizability by deploying it within a scenario that reflects an industrial network. Here, we will utilize the same dataset used for training, but we will remove the attack labels and types. This forces the model to identify potential attacks based solely on the network traffic patterns, simulating a real-world situation where attackers might try to evade detection.

We will delve into the implementation of an Intrusion Detection System (IDS) within the architecture of an industrial company's IT and OT (Operational Technology) network. We will explore how these components interact and how potential attackers might target them to launch DDoS attacks.

The scenario will showcase how the trained deep learning model can be deployed within the IIoT environment to continuously monitor network traffic for malicious activities associated with DDoS attacks.

Finally, this chapter will also shed light on the development environment used to create the model. We will discuss the programming languages, libraries, and tools that facilitated the model's development and training.

2. Development Environment:

In this section, we present the software environment and the language used for the development and implementation of our scenario. We'll detail the specific tools employed for coding and project management.

2.1. Software environment:

- **Spyder IDE:**

Spyder is a free and open-source scientific environment written in Python and designed by and for scientists, engineers, and data analysts. It features a unique combination of the advanced editing, analysis, debugging, and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and beautiful visualization capabilities of a scientific package. [50]

- **Python (programming language):**

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. it is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented, and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Python consistently ranks as one of the most popular programming languages and has gained widespread use in the machine-learning community.[51]

- **Lucidchart:**

Lucidchart is a web-based diagramming application that allows users to visually collaborate on drawing, revising, and sharing charts and diagrams, and improve processes, systems, and organizational structures.[52]

2.2. Python libraries used:

- **Pandas:**

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.[53]

- **NumPy:**

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.[54]

- **Scikit-learn:**

scikit-learn (formerly scikits.learn and also known as sklearn) is a free and open-source machine learning library for the Python programming language, It features various classification, regression, and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.[55]

- **Matplotlib:**

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of matplotlib.[56]

- **Tensorflow.keras:**

Keras is the high-level API of the TensorFlow platform. It provides an approachable, highly-productive interface for solving machine learning (ML) problems, with a focus on modern deep learning. Keras covers every step of the machine learning workflow, from data processing to hyperparameter tuning to deployment. It was developed with a focus on enabling fast experimentation.[57]

3. Network architecture of the IIoT environment:

This section explores the network architecture designed to evaluate the effectiveness of our Convolutional Neural Network (CNN) based Intrusion Detection System (IDS) within an IIoT environment. This architecture serves as a controlled testbed that replicates key characteristics of real-world IIoT networks, where Operational Technology (OT) and Information Technology (IT) systems converge.

By capturing and analyzing a comprehensive set of traffic patterns observed in IIoT deployments, the IDS establishes a baseline understanding of legitimate network behavior.

Industrial IoT networks present unique challenges for network architecture design due to the convergence of OT and IT systems. OT systems, responsible for controlling physical processes (e.g., industrial equipment), often have different communication protocols and security requirements compared to IT systems, which manage data and information flow. Additionally, the real-time nature of industrial processes necessitates reliable and low-latency communication.

To address these challenges, IIoT network architectures typically employ a layered approach. This approach segregates the network into distinct segments, such as Device Layer, Field Layer, and Control Layer.

This layered architecture ensures secure and manageable communication between diverse devices and systems within the IIoT environment. The IDS, strategically positioned within this architecture, can monitor network traffic and identify potential threats that could disrupt OT operations or compromise sensitive IT data.

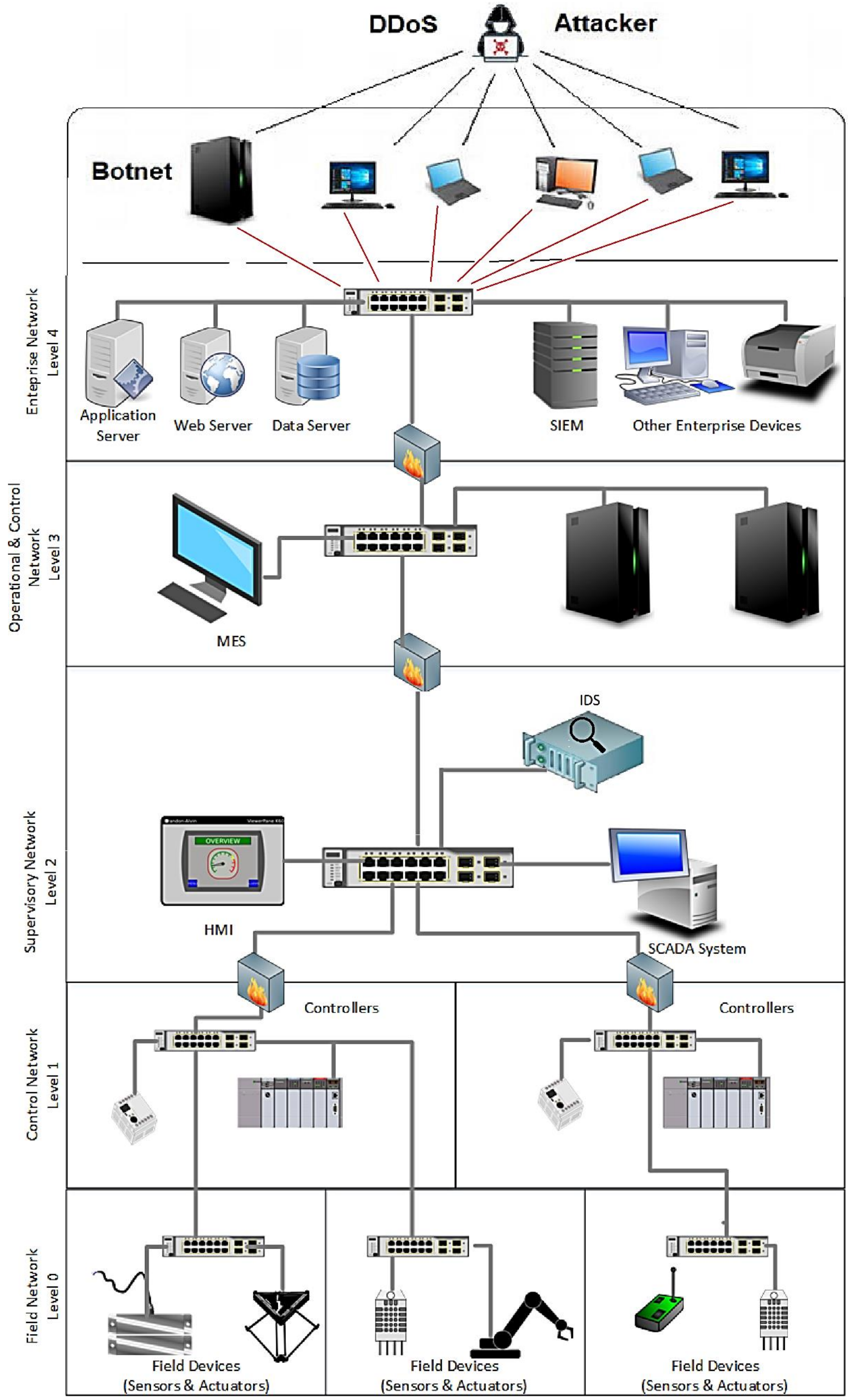


Figure 4.1. Network architecture of the IIoT environment

The scenario depicts a DDoS attack launched by a malicious actor. The attacker's goal is to disrupt and disable the Supervisory Control and Data Acquisition (SCADA) system, ultimately compromising the integrity of the connected IIoT devices. The IDS is strategically placed within the network to act as a guardian and identify such malicious activity.

Here's a breakdown of the elements within the IIoT network architecture:

- **Field Network (Level 0):** This level consists of various industrial sensors and actuators that gather and control physical data. These sensors and actuators are spread across the industrial environment.
- **Control Network (Level 1):** This network connects the field-level devices to controllers which act as intelligent devices that gather data from various sensors (temperature, pressure, flow, etc.) and send control signals to actuators (valves, motors, etc.) based on pre-programmed logic or instructions received from higher-level systems (PLCs in the Control Network).
- **Supervisory network (Level 2):** The supervisory network carries data collected by PLCs and other controllers on the Control Network (Level 1) to The SCADA system on Level 3 that can aggregate and filter the incoming data stream from multiple controllers, presenting a consolidated view of the overall process. Then connects the SCADA system to the Human-Machine Interface (HMI) which is the user interface that allows human operators to interact with the SCADA system for centralized monitoring and control.
- **Operational and Control Center Network (Level 3):** it focuses on centralized monitoring and control of industrial processes where a Manufacturing Execution System (MES) is integrated. It is primarily used in discrete manufacturing and connects to the SCADA system to receive real-time production data and sensor information.
- **Enterprise Network (Level 4):** This network acts as the bridge between the Operational & Control Center Network (Level 4) and the broader corporate network. It facilitates the exchange of data between industrial control systems and various enterprise IT resources, potentially providing access to the Internet.
- **Distributed Denial-of-Service (DDoS) Attacker:** a DDoS attack launched by a botnet that overwhelms the target system with traffic, causing denial-of-service that targets Level 2 (Supervisory network) to disrupt the SCADA system.

- **Intrusion Detection System (IDS):** our CNN-based IDS is positioned on the SCADA system (Level 3) and continuously monitors traffic to distinguish legitimate communication from DDoS attack patterns and protects the SCADA system and connected IIoT devices.

4. Scenario :

This section explores the experimental setup designed to evaluate the effectiveness of our Convolutional Neural Network (CNN) model within an Intrusion Detection System (IDS) for an Industrial Internet of Things (IIoT) environment. This scenario utilizes the Edge-IIoTset dataset to assess the IDS's ability to detect anomalies in network traffic without relying on predefined attack labels or types.

The experiment unfolds in distinct stages:

- **Simulated Attack:** An attacker launches a targeted attack against the IIoT network, sending malicious traffic towards the connected devices.
- **Network Traffic Capture:** A network sniffer captures all network traffic on the IIoT network. This captured traffic includes both the attacker's malicious packets and legitimate communication.
- **Data Collection and Preprocessing:** The captured network traffic, containing a mix of benign and potentially malicious data packets from the same Edge-IoT dataset, is saved in the "CSV" file format for analysis by the IDS.
- **CNN-based Anomaly Detection:** The IDS employs our CNN model, trained on the preprocessed Edge-IoT dataset minus the attack labels and types, to analyze the traffic. The CNN, trained to recognize patterns in network traffic, identifies anomalies that deviate from the expected behavior of legitimate communication within the IIoT environment.
- **Anomaly Detection and Alert Generation:** Upon detecting anomalies in the traffic, the IDS triggers an alert notification. This alert informs security personnel about the suspicious activity, allowing them to investigate further and determine if a malicious attack is underway.

5. Evaluating the CNN-based IDS performance:

Having established a robust Convolutional Neural Network (CNN) model through the training process, we now shift our focus to the critical phase of evaluating its performance as an Intrusion Detection System (IDS) within an IIoT environment. This evaluation process is crucial for assessing the model's effectiveness in identifying real-world DDoS attacks and ensuring its generalizability beyond the training data. The key steps involved in this testing scenario are:

1. **Attack Label and Type Removal:** The original Edge-IoT dataset is modified by removing the "attack label" and "attack type" columns. This forces the model to rely solely on the network traffic characteristics for anomaly detection, simulating a scenario where the IDS encounters unseen attack patterns.
2. **Model Loading:** The pre-trained CNN model, developed during the training phase, is loaded into the testing environment. This model is trained to recognize patterns in network traffic that deviate from normal behavior.
3. **Preprocessing The New Traffic Data:** A new dataset containing network traffic data (without attack labels or types) is loaded for testing. The loaded traffic data undergoes preprocessing steps similar to those used during training. The pre-processing includes standardization, a technique used during training to normalize the data's statistical properties. This step ensures the data aligns with the model's expectations and allows the model to understand the data format it was trained on.
4. **Reshaping the Data:** The pre-processed traffic data is reshaped into a format compatible with the CNN model's input layer. This reshaping involves converting the data into a specific number of channels, rows, and columns.
5. **Model Prediction:** Once the data is prepared, the loaded CNN model makes predictions on the new traffic data. The model analyzes the network traffic patterns and identifies instances that deviate from its understanding of normal behavior within the IIoT environment.

6. Anomaly Detection Analysis and Results:

The evaluation of the model reveals several key insights into its performance and potential for real-world application. Below is a detailed discussion of the results:

- **Accuracy:** The model achieved an accuracy of approximately 92%. This indicates that the model correctly classifies around 92% of the samples in the dataset.

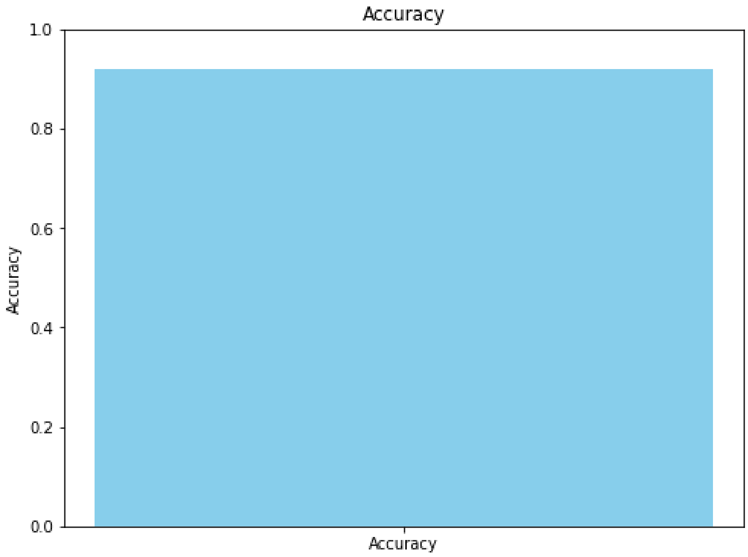


Figure 4.2. Accuracy test

- **Precision:** The precision scores are uniformly high across all classes, with Class 1 having the highest precision, followed closely by Classes 3, 4, 2, and 0. This indicates that the model performs consistently well in predicting all classes, with a very high accuracy rate when it makes predictions. The nearly perfect precision across all classes suggests that the model is highly effective in correctly identifying samples for each class.

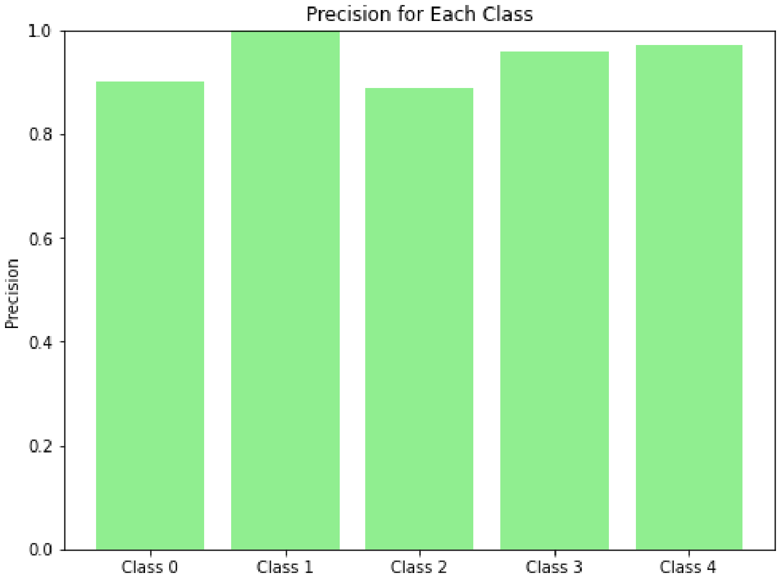


Figure 4.3. Precision test

- **Recall:** The recall for each class varies, indicating the model's differing ability to identify true positive samples across classes. Class 0 has a relatively high recall of about 80%, showing that the model correctly identifies the majority of true positive samples for this class. Class 1, however, has a lower recall of approximately 50%, indicating that the model only identifies half of the true positives for this class. Classes 2, 3, and 4 have recall values around 70%, 80%, and 60%, respectively, suggesting that the model is reasonably effective in identifying true positives for these classes, but still leaves room for improvement.

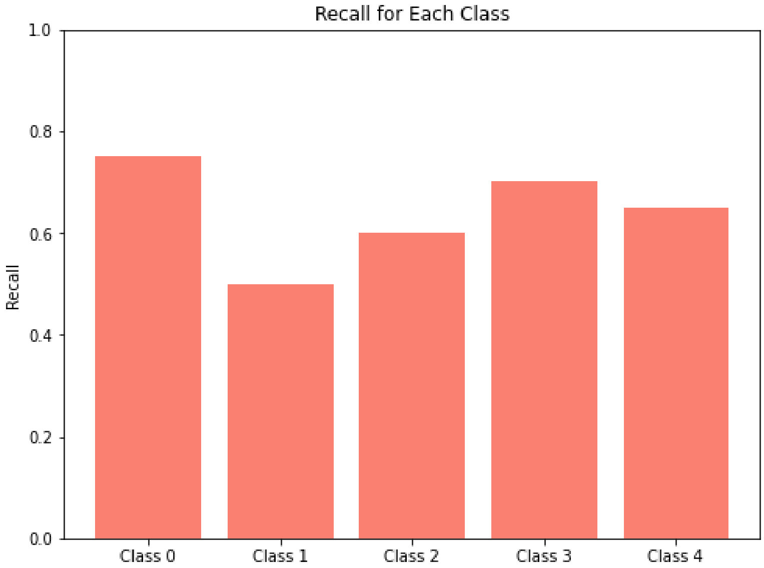


Figure 4.4. Recall test

- **F1 Score:** The F1 scores, which are the harmonic means of precision and recall, show variability across different classes. Class 0 has an F1 score of approximately 0.85, indicating a strong balance between precision and recall for this class. Class 1 has a lower F1 score of about 0.65, reflecting some difficulty in balancing precision and recall effectively. Classes 2, 3, and 4 have F1 scores around 0.75, 0.85, and 0.80, respectively, suggesting that the model performs fairly well in achieving a balance between precision and recall for these classes, but with slight variations in performance.

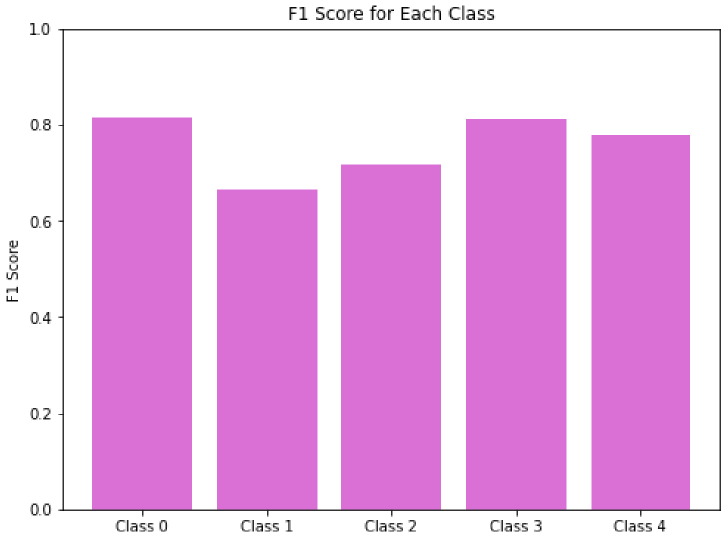
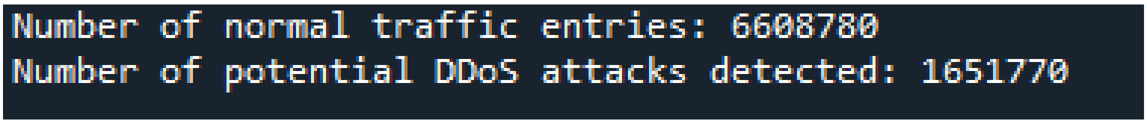


Figure 4.5. F1 Score test

The model's predictions were further analyzed to identify potential anomalies within the network traffic data. These anomalies could signify potential DDoS attacks, even though the model wasn't explicitly trained on attack labels or types. This demonstrates the model's capability to generalize and identify deviations from normal traffic patterns that might indicate unseen threats.

The model identified a significant number of potential DDoS attacks, with a total of 1,651,770 detections. This high number of detections suggests that the model is effectively capturing anomalies in the traffic data that deviate from what it has learned as normal behavior.



```
Number of normal traffic entries: 6608780
Number of potential DDoS attacks detected: 1651770
```

Figure 4.6. Number of DDoS attacks and normal traffic detected

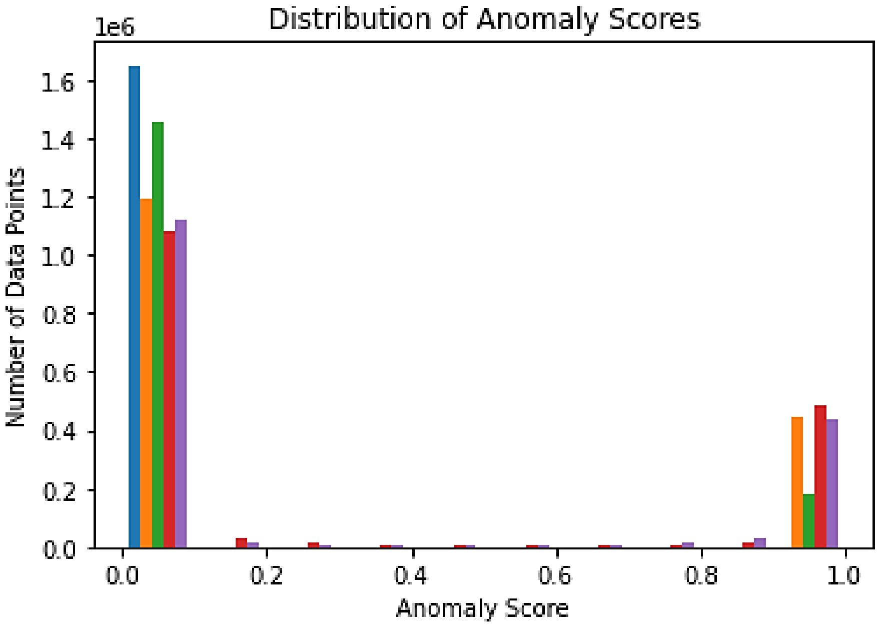


Figure 4.7. Distribution of anomaly scores

The bimodal distribution observed in the anomaly score graph shown in Figure 26 suggests promising results for the model's ability to detect anomalies in the IoT environment.

This distribution indicates that the model has effectively separated the data into two distinct clusters, The clear separation between these two peaks signifies the model's capability to distinguish between normal and anomalous traffic. This characteristic allows for setting a threshold anomaly score to classify data points. Any data point exceeding this threshold would be flagged as an anomaly.

The results indicate that while the model shows promise, particularly in terms of precision for certain traffic classes, there are significant areas for improvement. The low recall and zero performance on some classes highlight the need for a more balanced and comprehensive dataset.

Overall, the results are promising and demonstrate the model's potential in detecting potential DDoS attacks. Further analysis and exploration of new datasets can refine the model and lead to a more robust and reliable Intrusion Detection System.

7. Limitations and future work:

This research effort, while successful in deploying a CNN-based IDS for evaluation, encountered limitations due to practical constraints. Launching a real DDoS attack on a functional IIoT system was not possible due to ethical and safety considerations. Additionally, gaining access to a wide range of real-world industrial IoT devices for data collection proved challenging.

Despite these limitations, the project offers valuable insights and paves the way for future advancements. The simulated environment provides a solid foundation for further model development and testing. Here's how we envision future work:

- incorporating a wider variety of simulated and real-world (when ethically possible) attack scenarios into the training dataset to enhance the model's ability to identify diverse threats.
- Techniques like transfer learning can be explored to leverage the knowledge gained from the simulated environment and adapt the model to real-world IIoT network data when it becomes available.
- Optimizing the CNN model for resource-constrained industrial devices is crucial for practical implementation within IIoT environments.
- Partnering with industrial organizations can provide access to real-world data and physical IIoT setups for more comprehensive testing and refinement of the model.

By addressing these limitations and pursuing these future directions, this research can contribute significantly to the development of robust and adaptable CNN-based intrusion detection systems for safeguarding IIoT environments from ever-evolving cyber threats.

8. Conclusion:

In Chapter 4, we successfully tested our CNN-based intrusion detection system (IDS) within an IIoT environment. This virtualized IDS monitored network traffic for anomalies, showcasing its potential for safeguarding IIoT systems. While limitations like ethical constraints on real-world attacks and access to diverse industrial devices existed, this chapter lays a strong foundation for future advancements. By expanding the training data, exploring transfer learning, developing lightweight models, and collaborating with industry, we can refine the CNN-based IDS for real-world deployment, ultimately strengthening the security of IIoT environments.

General Conclusion:

The dynamic nature of cyber threats, particularly DDoS attacks targeting the Industrial Internet of Things (IIoT), necessitates constant innovation in security measures. Traditional intrusion detection methods often struggle to keep pace with the evolving strategies of attackers. This thesis addresses this challenge by proposing a deep learning-based Intrusion Detection System (IDS) tailored for detecting DDoS attacks in IIoT environments.

Through this research, we've underscored the critical significance of cybersecurity in the IIoT realm, which serves as the backbone for essential industrial operations. Our choice to delve into this theme stems from the growing integration of IoT devices in industrial setups, introducing new vulnerabilities that require robust security measures. It's crucial to recognize the pivotal role of cybersecurity in industrial IoT, as breaches can lead to severe operational disruptions, financial losses, and safety hazards.

Our study highlights the transformative potential of Artificial Intelligence (AI) in fortifying IIoT security. AI, particularly deep learning, offers sophisticated capabilities for analyzing intricate data patterns and identifying complex cyber threats. This thesis starts with laying the groundwork by elucidating IIoT concepts, scrutinizing prevalent attacks on IIoT systems, and emphasizing the indispensable role of robust cybersecurity. It further explores the integration of AI to enhance IIoT security, providing a theoretical framework for the research endeavor.

Furthermore, A comprehensive state-of-the-art study delves into recent advancements in deep learning methodologies for intrusion detection in IoT and IIoT networks. This study led us to opt for the Convolutional Neural Network (CNN) due to its adaptability, scalability, and proficiency in feature extraction. Additionally, we selected the Edge-IIoT dataset because of its extensive collection of traffic data encompassing various IIoT devices and sensors.

The proposed CNN model in this work was trained on pre-processed IIoT traffic data from the Edge-IIoT dataset, segmented into training and testing sets. The model showcases exceptional performance, achieving a remarkable accuracy of 99% in distinguishing normal traffic from DDoS attacks. This outcome outperforms other machine learning algorithms tested on the Edge-IIoT dataset for DDoS detection.

In the final section, we propose a practical scenario for deploying the proposed IDS within the IT/Operational Technology (OT) network architecture of an industrial company. We demonstrate how the trained model can be seamlessly integrated into an IIoT environment, discuss its performance and results, and analyze its effectiveness in real-world DDoS attack detection within IIoT settings.

This thesis makes a significant contribution to the cybersecurity field by introducing a robust deep learning-based solution for DDoS detection in IIoT environments. It presents both theoretical advancements and practical insights, paving the way for enhanced security in the ever-expanding IIoT landscape. By incorporating dropout layers and L2 regularization, we effectively mitigate overfitting, ensuring the model's resilience and generalization to new, unseen data.

Perspectives:

Looking ahead, there's vast potential for further refinement and expansion of the deep learning-based IDS for broader applications within the IIoT domain. Future research endeavors could explore the integration of real-time anomaly detection systems, capable of adapting to evolving attack patterns and reducing latency in detection. Additionally, expanding the dataset to encompass a wider variety of attack types and normal traffic patterns will bolster the model's robustness and reliability.

Moreover, collaborative efforts with industry partners can facilitate the deployment of these advanced IDS models in real-world IIoT environments, offering valuable feedback and insights for continual improvement. The convergence of AI and IIoT presents a fertile ground for innovation, heralding significant advancements in securing industrial operations against sophisticated cyber threats. By staying ahead of the evolving threat landscape, we can safeguard the safety, efficiency, and resilience of industrial systems in the digital era.

Bibliography:

[1] Shukla, P., Krishna, C.R. & Patil, N.V. Iot traffic-based DDoS attacks detection mechanisms: A comprehensive review. *J Supercomput* **80**, 9986–10043 (2024). https://doi.org/10.1007/s11227-023-05843-7

[2] Eid, A.M., Soudan, B., Nassif, A.B. *et al.* Enhancing intrusion detection in IIoT: optimized CNN model with multi-class SMOTE balancing. *Neural Comput & Applic* (2024). https://doi.org/10.1007/s00521-024-09857-x

[3] Khan, I. A., Keshk, M., Pi, D., Khan, N., Hussain, Y., & Soliman, H. (2022). Enhancing IIoT networks protection: A robust security model for attack detection in Internet Industrial Control Systems. *Ad Hoc Networks*, *134*, 102930. https://doi.org/10.1016/j.adhoc.2022.102930

[4] J. Zhao, Y. Liu, Q. Zhang and X. Zheng, "CNN-AttBiLSTM Mechanism: A DDoS Attack Detection Method Based on Attention Mechanism and CNN-BiLSTM," in IEEE Access, vol. 11, pp. 136308-136317, 2023, doi: 10.1109/ACCESS.2023.3334916.

[5] “What is the internet of things? | IBM.” Accessed: February 9, 2024. [Online]. Available at: https://www.ibm.com/topics/internet-of-things

[6] Y. B. Zikria, R. Ali, M. K. Afzal, and S. W. Kim, “Next-Generation Internet of Things (IoT): Opportunities, Challenges, and Solutions,” Sensors, vol. 21, no. 4, Art. no 4, Jan. 2021, doi: 10.3390/s21041174.

[7] “What is IIoT (Industrial Internet of Things)?” | Definition from TechTarget”, IoT Agenda. Accessed: February 9, 2024. [Online]. Available at: https://www.techtarget.com/iotagenda/definition/Industrial-Internet-of-Things-IIoT

[8] “IIOT vs IOT: Understanding the Key Differences - Digital Directions”, https://digitaldirections.com/. Accessed: February 9, 2024. [Online]. Available at: https://digitaldirections.com/iiot-vs-iot-understanding-key-differences-in-industrial-and-consumer-applications/

[9] “The Differences between IoT and IIoT | polimak.” Accessed: February 9, 2024. [Online]. Available at: https://polimak.com/en/the-differences-between-iot-and-iiot-iot-vs-iiot/

[10] “Industrial IoT Applications: Top 16 Uses of IoT in Industries”, Tutorials Freak. Accessed: February 9, 2024. [Online]. Available at: https://www.tutorialsfreak.com/

[11] “What components are in an IIoT network? », https://www.omega.com/en-us/. Accessed: February 9, 2024. [Online]. Available at: https://www.omega.com/en-us/resources/iiot-components

[12]“SECURING THE INDUSTRIAL IOT Trusted Solutions for Embedded Systems.” Accessed: May 17, 2024. [Online]. Available: https://www.nxp.com/docs/en/white-paper/SECURING-INDUSTRIAL-IOT-WP.pdf

[13] “IIoT Architecture Explained With Benefits and Examples”, Spiceworks. Accessed: February 9, 2024. [Online]. Available at: https://www.spiceworks.com/tech/iot/articles/what-is-iiot/

[14] “Mastering IIoT Security: A Definitive Guide to securing Industrial Internet of Things”. Accessed: February 10, 2024. [Online]. Available at: https://sectrio.com/blog/guide-to-industrial-iot-iiot-security/

[15] “Industrial IoT (IIoT) Attacks”. Accessed: February 10, 2024. [Online]. Available at: https://www.linkedin.com/pulse/industrial-iot-iiot-attacks-digitalert

[16] “Industrial IoT security against cyber threats”. Accessed: February 11, 2024. [Online]. Available at: https://www.ericsson.com/en/blog/2019/4/securing-your-industrial-iot-network-against-cyber-threats

[17] “Comparison and Differences Between IPS vs IDS vs Firewall vs WAF”. Accessed: February 11, 2024. [Online]. Available at: https://www.networkstraining.com/firewall-vs-ips-vs-ids-vs-waf/

[18] “What is an Intrusion Detection System (IDS)?” | IBM.” Accessed: February 11, 2024. [Online]. Available at: https://www.ibm.com/topics/intrusion-detection-system

[19] “What is an intrusion prevention system (IPS)? | IBM.” Accessed: February 11, 2024. [Online]. Available at: https://www.ibm.com/topics/intrusion-prevention-system

[20] “What is a VPN?” Definition from SearchNetworking”, Networking. Accessed: February 11, 2024. [Online]. Available at: https://www.techtarget.com/searchnetworking/definition/virtual-private-network

[21] A. Kumar, “What is Virtual Private Networks (VPNs)?” », DevSecOps Now!!! Accessed: February 12, 2024. [Online]. Available at: https://www.devsecopsnow.com/what-is-virtual-private-networks-vpns/

[22] “AI in IIoT: A New Industrial Revolution”. Accessed: February 12, 2024. [Online]. Available at: https://www.linkedin.com/pulse/ai-iiot-new-industrial-revolution-rossma

[23] S. Latif et al., “Deep Learning for the Industrial Internet of Things (IIoT): A Comprehensive Survey of Techniques, Implementation Frameworks, Potential Applications, and Future Directions,” Sensors, vol. 21, no. 22, p. 7518, Nov. 2021, doi: 10.3390/s21227518.

[24] O. Basystiuk, N. Melnykova, and Z. Rybchak, Machine Learning Methods and Tools for Facial Recognition Based on Multimodal Approach. 2023.

[25] “DDoS attack detection and classification via Convolutional Neural Network (CNN) | IEEE Conference Publication | IEEE Xplore.” Accessed: May 06, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/9014826

[26] “Industrial revolution 4.0: Cyber security challenges and solutions”. Accessed: February 22, 2024. [Online]. Available at: https://www.linkedin.com/pulse/industrial-revolution-40-cyber-security-challenges-khushhal-kaushik-l3s4f

[27] G. George and S. M. Thampi, “A Graph-Based Security Framework for Securing Industrial IoT Networks From Vulnerability Exploitations,” IEEE Access, vol. 6, p. 43586-43601, 2018, doi: 10.1109/ACCESS.2018.2863244.

[28] H. Alasmary, “RDAF-IIoT: Reliable Device-Access Framework for the Industrial Internet of Things,” Mathematics, vol. 11, no. 12, Art. no. 12, Jan. 2023, doi: 10.3390/math11122710.

[29] X. Li, J. Niu, M. Z. A. Bhuiyan, F. Wu, M. Karuppiah, and S. Kumari, “A Robust ECC-Based Provable Secure Authentication Protocol With Privacy Preserving for Industrial Internet of Things,” IEEE Trans. Ind. Inform., vol. 14, no. 8, p. 3599-3609, August 2018, doi: 10.1109/TII.2017.2773666.[26] P. De Vaere, A. Tulimiero, and A. Pe

[30] P. De Vaere, A. Tulimiero, and A. Perrig, “Hopper: Per-Device Nano Segmentation for the Industrial IoT,” in Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security, in ASIA CCS' 22. New York, NY, USA: Association for Computing Machinery, May 2022, p. 279-293. doi:10.1145/3488932.3501277.

[31] P. C. M. Arachchige, P. Bertok, I. Khalil, D. Liu, S. Camtepe, and M. Atiquzzaman, “A Trustworthy Privacy Preserving Framework for Machine Learning in Industrial IoT Systems,” IEEE Trans. Ind. Inform., vol. 16, no. 9, p. 6092-6102, Sept. 2020, doi: 10.1109/TII.2020.2974555.

[32] N. Bugshan, I. Khalil, N. Moustafa, and M. S. Rahman, “Privacy-Preserving Microservices in Industrial Internet-of-Things-Driven Smart Applications,” IEEE Internet Things J., vol. 10, no. 4, p. 2821-2831, Feb. 2023, doi: 10.1109/JIOT.2021.3098980.

[33] R. Saha et al., “DHACS: Smart Contract-Based Decentralized Hybrid Access Control for Industrial Internet-of-Things,” IEEE Trans. Ind. Inform., vol. 18, no. 5, p. 3452-3461, May 2022, doi: 10.1109/TII.2021.3108676.

[34] C. Lin, D. He, X. Huang, K.-K. R. Choo, and A. V. Vasilakos, “BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0,” J. Netw. Comput. Appl., vol. 116, p. 42-52, August 2018, doi: 10.1016/j.jnca.2018.05.005.

[35] C. Gonzalez, “What Is Threat Modeling? Key Steps and Techniques,” Exabeam. Accessed: February 22, 2024. [Online]. Available at: https://www.exabeam.com/information-security/threat-modeling/

[36] “Threat Modeling: Everything You Need to Know for Web Application Security”. Accessed: February 22, 2024. [Online]. Available at: https://www.linkedin.com/pulse/threat-modeling-everything-you-need-know-web-application-vadivel-r

[37] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E.-S. M. El-Horbaty, “Analysis of ToN-IoT, UNW-NB15, and Edge-IIoTset Datasets Using DL in Cybersecurity for IoT,” Appl. Sci., vol. 12, no. 19, Art. No. 19, Jan. 2022, doi: 10.3390/app12199572.

[38] “Zoghi and Serpen - UNSW-NB15 Computer Security Dataset Analysis thro.pdf”. Accessed: February 25, 2024. [Online]. Available at: https://arxiv.org/ftp/arxiv/papers/2101/2101.05067.pdf

[39] S. Soliman, W. Oudah, and A. Aljuhani, “Deep learning-based intrusion detection approach for securing industrial Internet of Things,” Alex. Eng. J., vol. 81, p. 371-383, Oct. 2023, doi: 10.1016/j.aej.2023.09.023.

[40] “Identification of malicious activities in industrial internet of things based on deep learning models - ScienceDirect”. Accessed: March 16, 2024. [Online]. Available at: https://www.sciencedirect.com/science/article/abs/pii/S2214212617306002

[41] S. Latif, Z. Idrees, Z. Zou, and J. Ahmad, “DRaNN: A Deep Random Neural Network Model for Intrusion Detection in Industrial IoT,” in 2020 International Conference on UK-China Emerging Technologies (UCET) , August 2020, p. 1-4. doi: 10.1109/UCET51115.2020.9205361.

[42] “lIDS-SIoEL: intrusion detection framework for IoT-based smart environments security using ensemble learning | Cluster Computing”. Accessed: March 16, 2024. [Online]. Available at: https://link.springer.com/article/10.1007/s10586-022-03810-0

[43] J.-C. Huang, G.-Q. Zeng, G.-G. Geng, J. Weng, K.-D. Lu, and Y. Zhang, “Differential evolution-based convolutional neural networks: An automatic architecture design method for intrusion detection in industrial control systems,” Comput. Security, vol. 132, p. 103310, Sep. 2023, doi: 10.1016/j.cose.2023.103310.

[44] M. Abdel-Basset, V. Chang, H. Hawash, R. K. Chakrabortty, and M. Ryan, “Deep-IFS: Intrusion Detection Approach for Industrial Internet of Things Traffic in Fog Environment,” IEEE Trans. Ind. Inform., vol. 17, no. 11, p. 7704-7715, Nov. 2021, doi: 10.1109/TII.2020.3025755.

[45] W. T. Meseret, “ANOMALY DETECTION FOR THE INDUSTRIAL CONTROL SYSTEM: A HYBRID DEEP LEARNING APPROACH”, Thesis, 2023. Accessed: March 22, 2024. [Online]. Available at: http://ir.bdu.edu.et//handle/123456789/15678

[46] J. B. Awotunde, C. Chakraborty, and A. E. Adeniyi, “Intrusion Detection in Industrial Internet of Things Network-Based on Deep Learning Model with Rule-Based Feature Selection,” Wirel. Common. Mob. Comput., vol. 2021, p. e7154587, Sep 2021, doi: 10.1155/2021/7154587.

[47] “An Ensemble Learning Based Intrusion Detection Model for Industrial IoT Security | TUP Journals & Magazine | IEEE Xplore. Accessed: March 22, 2024. [Online]. Available at: https://ieeexplore.ieee.org/document/10097653

[48] V. Priya, I. S. Thaseen, T. R. Gadekallu, M. K. Aboudaif, and E. A. Nasr, “Robust Attack Detection Approach for IIoT Using Ensemble Classifier,” Comput. Mater. Continu., vol. 66, no. 3, p. 2457-2470, 2021, doi: 10.32604/cmc.2021.013852.

[49] M. Ferrag, 1945. Accessed: May 17, 2024. [Online]. Available: https://dspace.univguelma.dz/jspui/bitstream/123456789/13246/1/NOUAR_IKRAME_F5.pdf

[50] “Home — Spyder IDE.” Accessed: May 09, 2024. [Online]. Available: https://www.spyder-ide.org/

[51] “Python (programming language),” *Wikipedia*. May 09, 2024. Accessed: May 09, 2024. [Online].Available:https://en.wikipedia.org/w/index.php?title=Python_(programming_language)&oldid=1222981091

[52] “Lucidchart,” *Wikipedia*. Jan. 07, 2024. Accessed: May 09, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Lucidchart&oldid=1194113202

[53] “pandas (software),” *Wikipedia*. Apr. 16, 2024. Accessed: May 09, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Pandas_(software)&oldid=1219216473

[54] “NumPy documentation — NumPy v1.26 Manual.” Accessed: May 09, 2024. [Online]. Available: https://numpy.org/doc/stable/

[55] “scikit-learn,” *Wikipedia*. Apr. 17, 2024. Accessed: May 09, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Scikit-learn&oldid=1219319746

[56] “Matplotlib,” *Wikipedia*. May 09, 2024. Accessed: May 09, 2024. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Matplotlib&oldid=1223001192

[57] “Keras: The high-level API for TensorFlow | TensorFlow Core,” TensorFlow. Accessed: May 09, 2024. [Online]. Available: https://www.tensorflow.org/guide/keras