AALBORG UNIVERSITY

Lígia Soster Ramos

Real-Time Cyberattack Detection, Diagnosis and Behaviour Forecast on an Offshore Refrigeration System with LSTM

ESBJERG

2023

LÍGIA SOSTER RAMOS

Real-Time Detection and Diagnosis of Cyberattacks on the Modbus TCP/IP communication between a Refrigeration System and SCADA and Forecast of the System Behaviour in an Offshore Platform using LSTM

Offshore Energy Systems Semester 4

AAU Supervisor: Zhenyu Yang

ESBJERG

2023

Abstract

This study explores the critical area of cyberattack detection and diagnosis in the scope of offshore control systems, specifically focusing on Modbus TCP/IP communication between refrigeration systems and SCADA. The project emphasizes the importance of swiftly and accurately identifying cyberattacks, followed by predicting system responses using neural networks.

The research incorporates the design and implementation of five distinct Long Short-Term Memory (LSTM) neural networks (NNs), each independently tailored for the classification or regression of various operational and cyberattack scenarios based on the respective system reaction. The first three LSTM NNs focus on the classification of 4 cyberattacks, while the other two forecast the room temperature following 5 hours into an attack detection. The uniqueness of each NN lies in their use of different solvers, architecture and hyperparameters.

The use of classification and forecasting models offers a comprehensive approach to handle cyber threats. While the classification networks effectively classified 100 datasets into their respective healthy or cyberattack scenario, the regression networks predicted the room temperature after a cyberattack detection 5 hours in advance with a RMSE of 0.17 and 0.19.

While providing promising results, it also opens doors to further enhancements, potentially contributing to the development of a more robust and effective cybersecurity framework for offshore industrial control systems.

List Of Abbreviations

Abbreviation	Description
ADU	Application Data Unit
APDU	Application Protocol Data Unit
APCI	Application Protocol Control Information
ASDU	pplication Service Data Unit
CCR	Central control room (located at South Brooklyn Marine Terminal)
CCTV	Central Controlled TV, Closed Circuit TV
DDOS	Distributed Denial of Service
DOS	Denial of Service
DS	Detection System
HMI	Human Machine Interface
HVAC	Heating, Ventilation and Air Conditioning
IDS	Intruder Detection System
IEC	International Electrotechnical Commission
IED	Intelligent Electronic Device
IP	Internet Protocol
LSTM	Long-Short Term Memory
LV	Low Voltage
MBAP	Modbus Application Protocol
NERC CIP	North American Electric Reliability Corporation Critical Infrastructure Protection
NN	Neural Network
ONS	Onshore Substation
OSS	Offshore Substation
PDU	Protocol Data Unit
PLC	Programmable Logic Controller
SCADA	Supervisory Control and Data Acquisition
SGDM	Stochastic Gradient Descent with Momentum
TCP	Transmission Control Protocol

Figures

Tables

Table Of Contents

Preface

I wish to express my deepest appreciation to my family and my supervisors Zhenyu Yang, Stig Bisgaard, and Thomas Frederiksen. Their guidance, feedback and support have immensely enriched this project.

This research has truly been a pleasure to develop, and I hope you enjoy reading it.

Aalborg University, 28th May 2023

Lígia Soster Ramos
Control and Automation Engineer

1. Introduction

Over the recent years, the emission of greenhouse gases has rapidly increased its negative influence on the climate. In order to prevent further growth of the average global temperature, necessary actions need to be taken, such as transforming current energy systems through the implementation of emission-free power sources. According to current projections, despite record breaking amounts of offshore wind energy capacity, global wind power growth needs to triple in the remaining years of the decade in order to limit global warming to a 2 degree C level with regard to the pre-industrial age (Zhao & Lee, 2021).

According to (Zhao & Lee, 2021) offshore wind energy is at the core of how Europe can go carbon-neutral. Moreover, the International Energy Agency states that it could become the number one source of power generation in Europe by 2042. As illustrated by Figure 1-1, by 2050, it is expected that Europe will need between 230 and 450 GW of offshore wind, meeting 30% of the overall continent's energy demands. While today Europe installs 3GW per year on average, 20GW per year will need to be installed after 2030. Lastly, from 2050 onwards, the installation rate for offshore wind will settle at 15GW per year to repower projects assuming a 30-year lifetime.

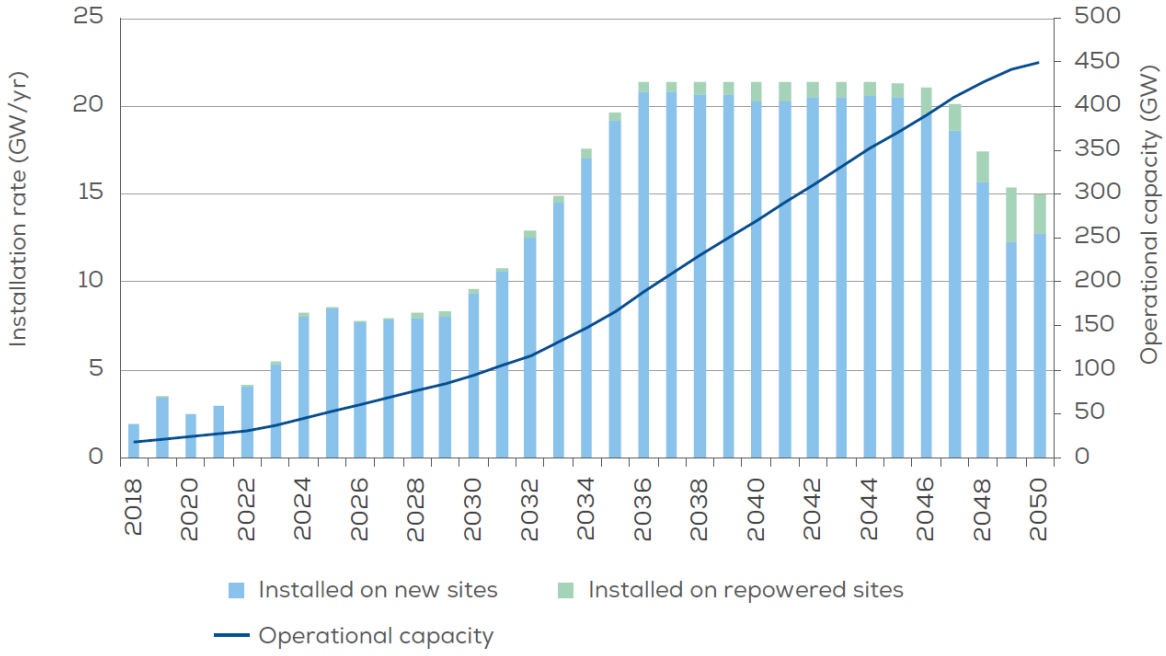

Figure 1-1: Rate required to Achieve 450 GW by 2050 (Zhao & Lee, 2021).

1.1 Offshore Platforms LV Communication

On offshore wind power plants, energy harvested by groups of turbines at sea are transported at 66k VAC to the platform through radials. This passes through the electrical substation, which includes gas insulated switchgears to collect said radials into fewer cables, a step up transformer to minimize current, and therefore losses, and a shunt reactor to absorb and compensate the reactive power in the long high voltage export cables.

One of the focuses of the SCADA system is to interface the low voltage control systems. SCADA, or Supervisory Control and Data Acquisition system, is an industrial control system, which monitors and controls assets. These are time critical systems, which support complex interactions between their logical and physical infrastructure, and, therefore, require real-time response, high availability and reliability.

Due to these platforms mostly being unmanned, a control room must have remote access to each of these systems. Further, these same systems must be able to be remotely connected to by their own supplier for system updates and troubleshooting. Therefore, not only does a secure and effective connection from the systems to a platform-owned control room need to take place, but also remotely to their own vendors, which increases the need for cybersecurity measures.

1.2 Cybersecurity at Offshore Platforms

Should a system be breached in an offshore station, the consequences of a system malfunction may vary from minor financial loss to complete shutdown of the plant, rending cybersecurity a critical section of the overall plant plan. Many SCADA and control systems have been on the market for more than 20 years and originally manufactures like Siemens, ABB and General Electric have put less emphasis on cybersecurity than features. This situation is rapidly changing, however, and today SCADA and control systems manufactures strive compliance to IEC 62443 and NERC CIP.

Offshore facilities are typically established in harsh conditions and require Heating, Ventilation and Air Conditioning (HVAC) systems to ensure operational conditions for equipment such as servers, control systems, Intelligent Electronic Devices (IEDs) and network equipment in terms of temperature, humidity and pressure. Refrigeration systems are part of the HVAC utility that use refrigerant to transfer out heat through evaporation and condensation. In the event of a cyberattack on the HVAC system, the amount of time an offshore facility could continue operating without HVAC would depend on a variety of factors such as weather conditions, location, and size, but personal experience from the author's offshore engineering colleagues in an offshore energy company in Texas, USA suggest around 4 hours. A cyberattack may alter the settings of the cooling utility to overburden the system or cause it to malfunction, resulting in damage to the devices or creating hazardous circumstances.

The significance of cybersecurity extends beyond HVAC to include all systems in a plant, as such systems can potentially allow attackers to access the plant's network. By gaining entry into a refrigeration system, for example, intruders may compromise other systems, and vice versa, causing unauthorized access to sensitive data and potential damage to the plant.

According to a report by IBM Security in 2020, there was a 2000% rise in attacks on operational technology systems, including building automation systems like HVAC, between 2018 and 2019. Further, it states that these attacks are becoming more sophisticated, with attackers employing techniques such as phishing emails, social engineering, and the exploitation of unpatched vulnerabilities (Alvarez, Michelle; IBM Security, 2020).

It is worth noting that the actual frequency of cyberattacks on HVAC systems is difficult to determine since many attacks may remain unnoticed or unreported. However, some public cases exist where cyberattacks have breached HVAC systems. In 2013, a significant data breach affected Target, a US-based retail chain, exposing personal data of millions of customers. The HVAC system vulnerability allowed hackers to access Target's payment network (Timberg et al., 2013). Another instance was the 2019 Norsk Hydro Attack, where a phishing email aimed at an employee led to a ransomware attack that disrupted Norsk Hydro's global operations. The malware spread throughout the company's network, including its HVAC systems (Ferguson, 2019). Lastly, in 2016, a massive botnet attack, known as the Mirai Botnet Attack, targeted IoT devices, including HVAC systems, causing widespread internet service disruption through DDoS attacks (Greenber, 2015).

These examples highlight the importance of securing utlity systems from cyber threats and highlight why it is imperative to implement robust cybersecurity measures and prevent unauthorized access.

Finally, the mostly used protocol for low voltage systems is the Modbus protocol, which will be explained in Chapter 4.

Bhatia et Al. (2020) explain that the Modbus protocol has been lacking in security since its introduction in 1979, lacking confidentiality and data integrity. The report goes on to outline some of the protocol's vulnerabilities and describes the simulation of two cyberattacks by infecting the master with malware and a man-in-the-middle attack. Bhatia notes that while these control systems have been reformed from serial communication to TCP/IP more recently, these systems and associated protocols have not been designed to withstand cyberattacks. In both simulated cyberattacks, the master accepted incorrect information on the slave, illustrating that the Modbus protocol is highly insecure, and caution is advised when using it in networks.

1.3 Real-Time Cyberattack Detection

One way to bolster cybersecurity is through real-time cyberattack detection, which involves identifying cyber threats as they happen. This approach enables organizations to minimize the damage caused by breaches and prevent sensitive data from being compromised. Real-time cyberattack detection also assists in detecting and mitigating system vulnerabilities

before hackers can exploit them. By continuously monitoring their systems for unusual activity, organizations can proactively identify and remediate potential security risks.

"Real-time" in the context of computing systems often refers to the ability of a system to process data and provide responses within an expected time frame that's considered immediate or nearly immediate. The concept of "near real-time" describes systems that process and respond quickly enough for the output to still be very useful, even if it's not instantaneous. The acceptable delay in near real-time systems can vary widely depending on the specific use case, and can range from a couple of seconds to several minutes. In the application at hand, since the results will be dependent on the system reaction to the cyberattack, it is necessary to wait for said reaction.

Some examples of real-time cybersecurity include: Intrusion Detection Systems (IDS), a technology that monitors networks and systems for suspicious activity or behavior by analyzing network traffic in real-time to identify patterns that could indicate an attack, such as unusual login attempts, unauthorized access, or data exfiltration (Classification of Intrusion Detection Systems, u.d.); Security Information and Event Management (SIEM), a system that collects and analyzes security data from various sources, such as network devices, servers, and applications by correlating events across different sources and providing alerts to security teams (IBM, u.d.); Endpoint Detection and Response (EDR) monitors endpoints, such as laptops, desktops, and mobile devices, for signs of compromise, responding to threats in real-time by identifying anomalous behavior, isolating infected endpoints, and blocking malicious processes (Endpoint Security Market, u.d.).

The accuracy of Intrusion Detection Systems (IDS) may vary based on different factors, including the IDS type, operating environment, and the kinds of threats it's designed to detect. IDS can generally identify known threats and attacks matching predetermined patterns or signatures with a high degrees of accuracy. For instance, if an IDS is programmed to recognize a particular malware or network attack, it can effectively identify instances of that attack. Nevertheless, a 2021 survey by Waterfall Security Solutions revealed that just 62% of SCADA security professionals believe that IDS is efficient in detecting cyber threats in SCADA systems. The most commonly employed IDS types were network-based IDS and signature-based IDS, according to the survey, and false positives were a significant problem for IDS in SCADA systems (Ginter & Hale, 2021).

There are several disadvantages to IDS - false positives can happen when normal network activity is mistakenly identified as malicious, resulting in wasted resources. False negatives can also occur when an actual attack goes undetected, allowing attackers to gain access to the system unnoticed. These devices will also not recognize when the plant is under maintenance and the SCADA behaves differently, or after a SCADA and/or its network devices have been updated, generating more false results.

Also, IDS will fail in detecting new or unknown threats that do not have predefined signatures or patterns, known as zero-day attacks, or in detecting hacks occurring outside of its coverage area, such as those happening on the physical layer of the network. Further, in SCADA systems that require real-time data processing, IDS can impact system performance, causing potential latency.

Artificial Intelligence, however, has shown promise as a tool for Intrusion Detection Systems, as it makes use of machine learning algorithms to identify patterns and anomalies in network traffic, and to detect potential cyber threats in real-time. Table 1-1 shows the comparison between advantages and disadvantages of the use of AI IDS retrieved from Wolf, Coleman (2021).

Table 1-1: Comparison of Advantages and Disadvantages of Using AI IDS.

Advantages of AI IDS	**Disadvantages of AI IDS**
Improved Detection Accuracy: AI-based IDS can detect previously unknown threats that may not be detectable with traditional signature-based IDS. Machine learning algorithms can learn from large amounts of data to identify patterns and anomalies, allowing them to detect more sophisticated attacks.	Training Data: AI-based IDS require a large amount of training data to accurately detect cyber threats. This can be challenging in SCADA systems where the data is highly specialized and sensitive.
Real-Time Detection: AI-based IDS can detect cyber threats in real-time, allowing security personnel to respond quickly to potential attacks.	Adversarial Attacks: Attackers can attempt to evade detection by intentionally manipulating the network traffic to fool AI-based IDS.
Reduced False Positives: AI-based IDS can reduce false positives by learning normal behavior patterns and identifying deviations from them, reducing the number of false alarms.	Additional computing power may be required.
Scalability: AI-based IDS can be easily scaled to handle large amounts of data, allowing them to be effective in high-traffic environments.	

Neural networks are a form of AI that are widely used in machine learning, particularly in deep learning, to identify patterns in data, based on how the human brain functions. Multiple layers of neural networks, known as deep learning, allow for more complex pattern recognition and improved performance. One example of how neural networks have been used in cybersecurity to examine network traffic and recognize anomalous activity that could indicate a cyberattack, are recurrent neural networks (RNNs) used for intrusion detection. RNNs are trained to recognize normal behavior patterns and detect anomalies that differ from them, aiding in the identification of potential cyberattacks in real-time.

Although the subject of cybersecurity and neural networks have been studied in the past, their use in offshore platforms, which consist of increased complexity, has not been analyzed. Different factors constitute said complexity – amongst them are:

(1) Control System Complexity: The complexity of offshore platform control systems, with numerous interconnected components, makes securing them a difficult task. Detecting and responding to cyber threats may also be challenging due to this added complexity.
(2) Human Error: In offshore platform cybersecurity, human error is a significant risk factor. Workers and contractors may not receive adequate cybersecurity training, leading to unintentional vulnerabilities.

(3) Insider Risk: Authorized insiders who have access to sensitive data on offshore platforms may pose a cybersecurity threat.
(4) Inadequate Security Measures: Insufficient cybersecurity measures due to equipment that has been offshore for long periods of time.
(5) Security Standards Disparity: The lack of standardization in cybersecurity requirements for offshore platforms may result in confusion and inconsistency in security practices, posing challenges to ensuring adequate protection for all platforms.
(6) Maintenance Challenges: Maintaining and monitoring remote connections to offshore platforms regularly to ensure proper functionality can be difficult given the harsh environments and distance involved.

The field of cybersecurity at offshore platforms is complex and rapidly evolving, requiring a multidisciplinary approach to adequately address the numerous challenges that exist. In the scope of emerging technologies lies the study of the latest technologies, such as AI and machine learning, which can help identify new approaches to mitigating cybersecurity risks at offshore platforms.

Therefore, this report will study the use of neural networks to detect and diagnose cyberattacks in real-time between the communication of a SCADA system and a refrigeration system, as well as to forecast post-cyberattack conditions to aid in the repair process, since these are mostly unmanned platforms.

2. Problem Analysis

In this analysis, it was found that a limited amount has been explored and reported on the use of Neural Networks for cybersecurity at offshore platforms. This chapter will explain SCADA and its limitations in terms of cybersecurity, present studies on Modbus cybersecurity, Neural Network (NN) cybersecurity and finally explore the state-of-the-art studies on the topics of artificial intelligence for cyberattack detection, so that a task statement is reached and these topics are introduced. These will be further explored in later chapters.

2.1 SCADA And Its Limitations

Firstly an examination of the cybersecurity in SCADA systems should be made. As mentioned, due to these offshore platforms mostly being unmanned, there is a need for remote connections from said vendors to their systems' control,

When a control system is hacked, the hacker can gain access to the logic of the controller and find vulnerabilities in the field equipment. In a utility system with both a PLC and a local HMI, where both are made accessible via the network, the HMI will be set as the first target, as this often is windows or Linux based. There is no definitive guide to writing secure PLC codes, which means that the responsibility for cybersecurity lies mainly in the upper layers of the system (Jr, 2013).A survey conducted by Fortinet in 2019 revealed that 56% of multinational companies had experienced a SCADA breach in the previous year, and another 32% had been breached, while only 12% of SCADA users reported never having been breached earlier (Fortinet, 2019).In another study by Trend Micro in 2022, 89% of the 310 energy companies surveyed reported disruptions in their cybersecurity impacting their supply chain in the past year. Of these, 56% experienced a disruption lasting 4 days or more, resulting in an average financial loss of over 3 million US dollars (Trend Micro, 2022).

Additionally, recent research conducted by Jones Walker in the United States revealed that SCADA systems and Field Device Management Systems are responsible for 68% of the vulnerabilities perceived in data breaches for port and terminal platforms. The study involved directors, security and compliance officers, and general counsel from the industry (Lee et al., 2022).

2.2 Artificial Intelligence used for Cyberattack Detection

There has been some research on the use of neural networks for cybersecurity at offshore platforms, although the amount of research is relatively limited compared to other areas of cybersecurity.

One study that specifically investigated the use of neural networks for cybersecurity at offshore platforms was conducted by Hassan et al. (2020). The study focused on the detection

of anomalies in sensor data from offshore platforms using a deep neural network architecture. The results showed that the neural network approach was effective in detecting anomalies in the data and could potentially be used for real-time cybersecurity monitoring at offshore platforms. Hassan et al. (2020) found that one of the main limitations of their neural network-based anomaly detection system for offshore platforms was the limited availability of real-world datasets. They noted that the system's performance could potentially be improved with more data.

Another study by Zhao et al. (2021) proposed a neural network-based intrusion detection system for offshore oil and gas facilities. The proposed system used a deep neural network to learn from network traffic data and identify anomalous behavior. The system was evaluated on a dataset of network traffic from an offshore oil platform, and the results showed that the neural network approach was effective in detecting various types of attacks. Zhao et al. (2021) identified the complexity of offshore platform systems and the large volume of data generated by sensors as a challenge for developing and training neural network models that are accurate and efficient in detecting cyber threats.

In A Deep Learning-based Intrusion Detection System for Offshore Platforms, a novel intrusion detection system for offshore platforms was suggested utilizing Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks that employ deep learning techniques. The system was trained on network traffic data collected from an offshore platform and performed with a high level of accuracy in identifying various forms of cyberattacks. The authors highlighted that one of the key difficulties faced in developing this neural network-based intrusion detection system was the need to tackle the uneven distribution of normal and anomalous traffic in the training data (Khan et al., 2021).

In the review article Network intrusion detection system using machine learning: a comprehensive review, Alrawais et Al. provide a comprehensive overview of machine learning techniques for network intrusion detection, including the use of neural networks. The article discusses the potential benefits and limitations of using neural networks for intrusion detection and provides examples of their use in various contexts, including offshore platforms. Alrawais et al. (2019) identified several challenges associated with the use of neural networks for network intrusion detection, including the need for large amounts of training data and the difficulty of interpreting the results of neural network models (Alrawais et al., 2019).

In Anomaly detection for industrial IoT: a neural network approach. Journal of Industrial Information Integration, Rahman investigated the use of neural networks for anomaly detection in Industrial Internet of Things (IIoT) systems, including offshore platforms. The study proposed a neural network-based approach for anomaly detection in sensor data from IIoT systems and demonstrated its effectiveness in detecting anomalous behavior in a dataset from an offshore platform. Rahman et al. (2018) found that one of the challenges of developing a neural network-based anomaly detection system for industrial IoT, including offshore platforms, was the need to address the trade-off between model complexity and interpretability (Rahman et al.).

In Network Intrusion Detection System using Deep Learning Technique, Edeh proposed a Feedforward Deep Neural Network (FFDNN) for an intrusion detection. Three

variants of FFDNNs were trained showed accuracies of 89%, 84%, and 87%. The experiment was also conducted on the same training dataset (NSL-KDD) using the conventional machine learning algorithms (Random Forest; K-nearest neighbor; Logistic regression; Decision tree; and Naïve Bayes) and predictions of each algorithm on the test data gave different performance accuracies of 81%, 76%, 77%, 77%, 77%, respectively. Overall, when compared to the conventional Machine Learning algorithms, the outcome shows that the deep neural networks performed best due to their dense architecture that made it scalable with the large size of the dataset and also offered a faster run time during training in contrast to the slow run time of the Conventional Machine Learning. This implies that when the dataset is large and a faster computation is required, then neural networks are a better choice for best performance accuracy.

Some of the challenges found by the author included limited availability of datasets: This can make it difficult to develop and test accurate models that can generalize well to new data. Further, model complexity was an issue - deep learning models are typically more complex than traditional machine learning models, which can make them more difficult to train, interpret, and optimize. This can lead to issues with overfitting, where the model performs well on the training data but poorly on new, unseen data (Edeh, 2021).

In Evaluation of Efficient Classification Algorithm for Intrusion Detection System Devi and Priyalakshmi apply different classification algorithms to detect different types of attacks in order to develop the performance of IDS. The results showed that the random forest algorithm achieved sufficient results and the highest accuracy to classify different types of attacks. The effectiveness of dimension reduction to reduce big data sets' complexity leads to select most favorable features to obtain better performance in classification in terms of accuracy and speed.

This was yet another report, which stated the challenge of high-quality data availability limitation. Another challenge was the imbalanced data distribution in regards to the ones with and without intrusions, making the model biased towards the majority class. There was also high false positive rate, where normal traffic is incorrectly classified as an intrusion. Finally, the authors emphasized the importance of developing efficient classification algorithms that can accurately classify network traffic in real-time. This is especially important for large-scale networks, where the volume of data can be extremely high (Priyalakshmi & Devi, 2022).

In A Novel Hybrid Deep Learning Framework for Intrusion Detection Systems in WSN-IoT Networks, Maheswari et Al describe the limitations of current IDS and use Multi-tiered Intrusion Detection with hybrid deep learning models for improved detection accuracy in wireless networks. Long short-term memory was studied to design IDS effectively. The results show that the proposed algorithm has shown consistent performance, such as accuracy of 99.89%, precision of 98%, 97.5% recall, and 99% f-score in predicting the various categories of attacks in the WSN-IoT environment using scalable datasets. The other hybrid deep learning models have exhibited high performance using benchmarks but showed a marked dip in performance while using the real-time datasets. Once again, the author emphasizes the limited

availability of high-quality datasets, the need for efficient algorithms in scalability (Maheswari & Karthika, 2021).

Overall, these reports highlight several common limitations and challenges associated with the use of neural networks for cybersecurity at offshore platforms, including the limited availability of real-world datasets, the complexity of offshore platform systems and the large volume of data generated, and the difficulty of interpreting the results of neural network models. Due to these, the project at hand will aim to develop a model specific to offshore refrigeration systems to produce equal amounts of data for healthy and unhealthy behaviors. Also, LSTM will be used due to its accuracy and efficiency, as proven by this chapter, when compared to other NN and other machine learning algorithms. Further, the author will strive for an LSTM network with the least complexity in architecture while still aligned with the project goal, so as to aid in the efficiency of scalability. Lastly, this neural network will learn from the reaction of the system and its communicating variables to the specific events, and therefore, it will be designed to be set between the Switch and the Refrigeration Cycle. In this manner, the author aims to minimize the number of false positives and negatives that could be read from network traffic, and therefore augment the NN accuracy.

3. Task Statement And Delimitation

3.1 Task Statement

Based on discussions from the past bibliography described in chapter 2, the following final task statement question was formulated for the ongoing project:

Can a NN detect and diagnose a cyberattack in the Modbus TCP/IP communication between SCADA and a refrigeration system from an offshore platform in real-time? If so, can a NN be trained to forecast the reaction of said system to the cyberattack so as to aid in the post-attack procedure?

The motivation to study neural networks as intrusion detection systems for Modbus TCP/IP communication to a refrigeration system in an offshore platform is to enhance the security of the control systems that operate in such environments. Offshore platforms are complex and critical infrastructures that are vulnerable to cyberattacks, and any breach of their security can have serious consequences from environmental damage, platform shutdown, financial losses and even access to further cyberattacks on connecting power grids. Refrigeration systems are essential components of the HVAC system and Modbus TCP/IP is a widely used communication protocol that allows different devices in the HVAC system to exchange data and commands. However, Modbus communication is vulnerable to a range of cyberattacks, including intrusion and modification, which can compromise the integrity and availability of the HVAC system. In turn, due to the importance of HVAC in offshore platforms, this lack of integrity or availability can damage and shut down the existing plant. Finally, should a NN be able to forecast the behavior of the refrigeration system after the cyberattack, there can be better risk mitigation, real-time response and damage control.

3.2 Delimitation

In order to solve the above-stated problem, firstly a model of a refrigeration system will be constructed to offer the large amounts of data needed by neural networks to learn from. Next, its communication through Modbus TCP/IP to a SCADA system will be analyzed with the finality of choosing where to emulate each cyberattack. For this finality, Denial of Service and Man in the Middle are chosen as attacks known to target Modbus TCP/IP. Next, neural networks will be trained to understand the healthy behavior of a refrigeration model, as well as its behavior with the given cyberattacks. Finally, the neural network will be tested for its capability of detecting and diagnosing different cyberattacks to the Modbus TCP/IP communication, based on the behavior of the refrigeration system. Different solvers will be used along with different parameter tuning and different reaction times will be tested to achieve the closest to a real-time detection system. Should the neural network respond accurately to the classification training, the regression training will start to learn to predict the behavior of the system after a cyberattack.

4. Modbus Protocol

This chapter will explain the Modbus TCP/IP protocol and its cybersecurity.

A communication protocol defines the regulations for transmitting data over a communication channel, allowing devices to interact with each other. Open protocols permit external vendors to offer their systems as part of a decentralized solution, where they are responsible for maintaining and providing technical support for their products.

Many protocol standards used in critical systems, such as the power industry, were designed with little or no attention to security, and some of them are decades old, making them vulnerable and ineffective in certain applications. The industrial marketplace is currently dominated by around 10 protocols, including MODBUS, OPC UA, PROFINET, and IEC 60870-5-104 (Byres, 2004).

MODBUS, an open protocol developed in 1979 by Modicon, was intended as an internal point-to-point communication protocol between the company's PLCs and programming panels using a master-slave method. Initially, it was implemented over a serial line, but it has evolved to operate over a TCP/IP network, as illustrated in Figure 4-1 Illustration of Modbus TCP/IP Communication .

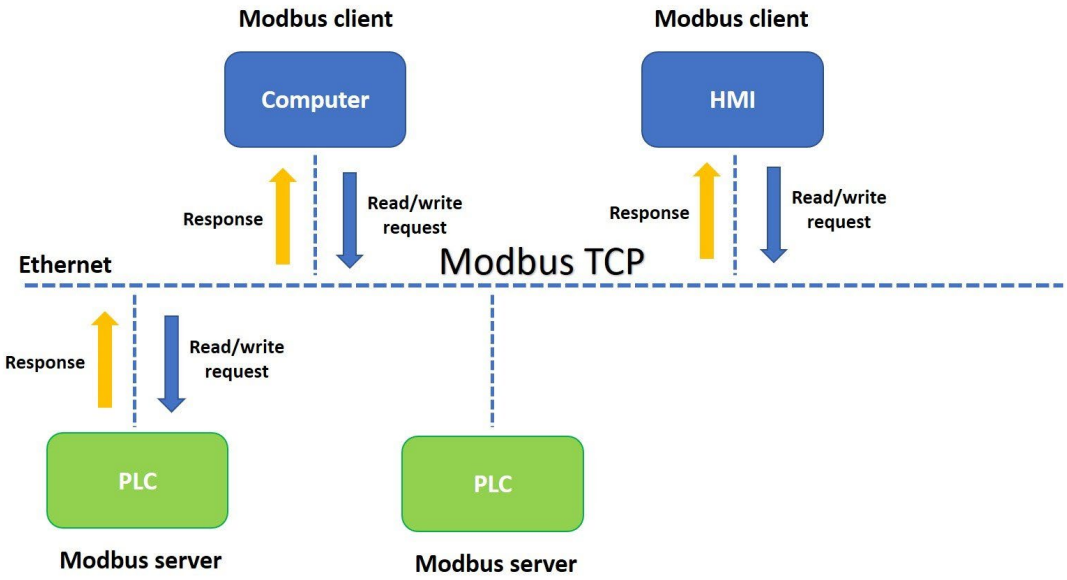

Figure 4-1 Illustration of Modbus TCP/IP Communication (What is modbus TCP Protocol?, 2021).

The Modbus TCP protocol allows a master device to perform multiple transactions while a slave device can communicate with multiple masters. In general, a slave device listens for incoming TCP connections on port 502, processes the received data, and responds back to the master through Modbus. The master device can either send a broadcast message to all slave devices or send individual requests to each of them (Acromag, 2005).

When a message is transmitted through a network, each layer encapsulates the received data with its own protocol header and footer, which is then sent down the OSI layers. At the physical layer, the data is electronically transmitted to the destination, and then travels upward through each layer, with each one decoding its portion of the message and removing its header and footer. Finally, the message reaches the application layer of the recipient device (Acromag, 2005).

Although it is the mostly used protocol for low voltage utilities at offshore platforms, there are several downsides to using Modbus TCP/IP networks in these stations. One main concern is that it does not incorporate any built-in security mechanisms, such as authentication or encryption, leaving it vulnerable to various forms of cyberattacks.

On one hand, it allows for easy setup and integration of devices across various platforms. On the other hand, it exposes the system to potential security breaches as the communication between devices is sent unencrypted, making it easy for attackers to read, intercept, and manipulate data. This lack of encryption and authentication in the Modbus TCP/IP protocol means that a malicious entity can easily inject false data, disrupt communication, or even take control of the connected devices.

Moreover, Modbus TCP/IP lacks provisions for user roles and access levels. This means all devices on the network can potentially issue commands to any other device. If an attacker gains access to the network, there are no inherent mechanisms in place to limit what they can do, thereby posing a significant risk to the operational integrity of the entire system.

This vulnerability is further exacerbated when these systems are connected to the internet for remote access or monitoring, which is increasingly common in today's interconnected world. An inadequately secured device can provide an entry point to the network, enabling attackers to exploit the protocol's vulnerabilities and compromise the entire system.

5. Cyberattacks

This chapter will address four possible cybersecurity threats regarding DOS and MitM within the system and later simulate them into the model.

Firstly, to properly function, the refrigeration cycle relies on a signal from its controller to turn the compressor on or off. This is based on the room temperature signal that it sends back. This controller is connected through a switch to the HVAC PLC, which controls not only this refrigeration system, but also other utilities, such as heat pumps and fans. Figure 5-1 illustrates the different stages of communication from the refrigeration system to a third-party computer, and while the labelled Original System is what is found in platforms today, the labelled Additional Sub-System is what is envisioned for this implementation. While the HVAC PLC informs the controller of any emergency shutdown, the controller informs the PLC about the room temperature and the compressor status. Between the HVAC PLC and the SCADA the threshold for temperature rise alarm and the target temperature are given, while the room temperature and the compressor status are received. Finally, for remote access to the unmanned platform, this information is available through the internet.

A Man in the Middle attack between the switch and the refrigeration cycle would give access to the communication bus, permitting the controller to obtain any status of the room temperature and the compressor to freely change its status. Should a hacker choose to turn the compressor off, the refrigerated room would no longer condense its heat and thus warm up. With no room temperature status update, the temperature rise alarm would not go off and the hack would go unnoticed as far as communication goes.

A Man in the Middle attack between SCADA and the HVAC PLC would give the hacker ability to change the target temperature of the room and the threshold for the temperature rise alarm. As a hacker increases the target temperature of the system and raises the set point for the alarm, the refrigerated room warms up and no alarm is given.

A Man in the Middle attack at the Switch would give someone the ability to activate the emergency shutdown option, change the target temperature and the threshold for the temperature rise alarm. The compressor status and room temperature could be made to look like the standard values to the operator.

A Denial of Service attack between the controller and the refrigeration cycle would overwhelm the communication bus, deactivating the change of status of the compressor and halting any status briefing of the room temperature. In this case, the room could either heat up if the hack halted the compressor in off or cool down if on. In both cases the SCADA would not have any information on the room temperature.

As HVAC components are significantly heavy consumers of power, any alteration in this could be observed by operators. Even if these results are being faked through the Modbus connection, an anomaly in the power consumption would signal the SCADA of unusual activity. The source of this, however, would not be perceived and the proper procedure would not be followed.

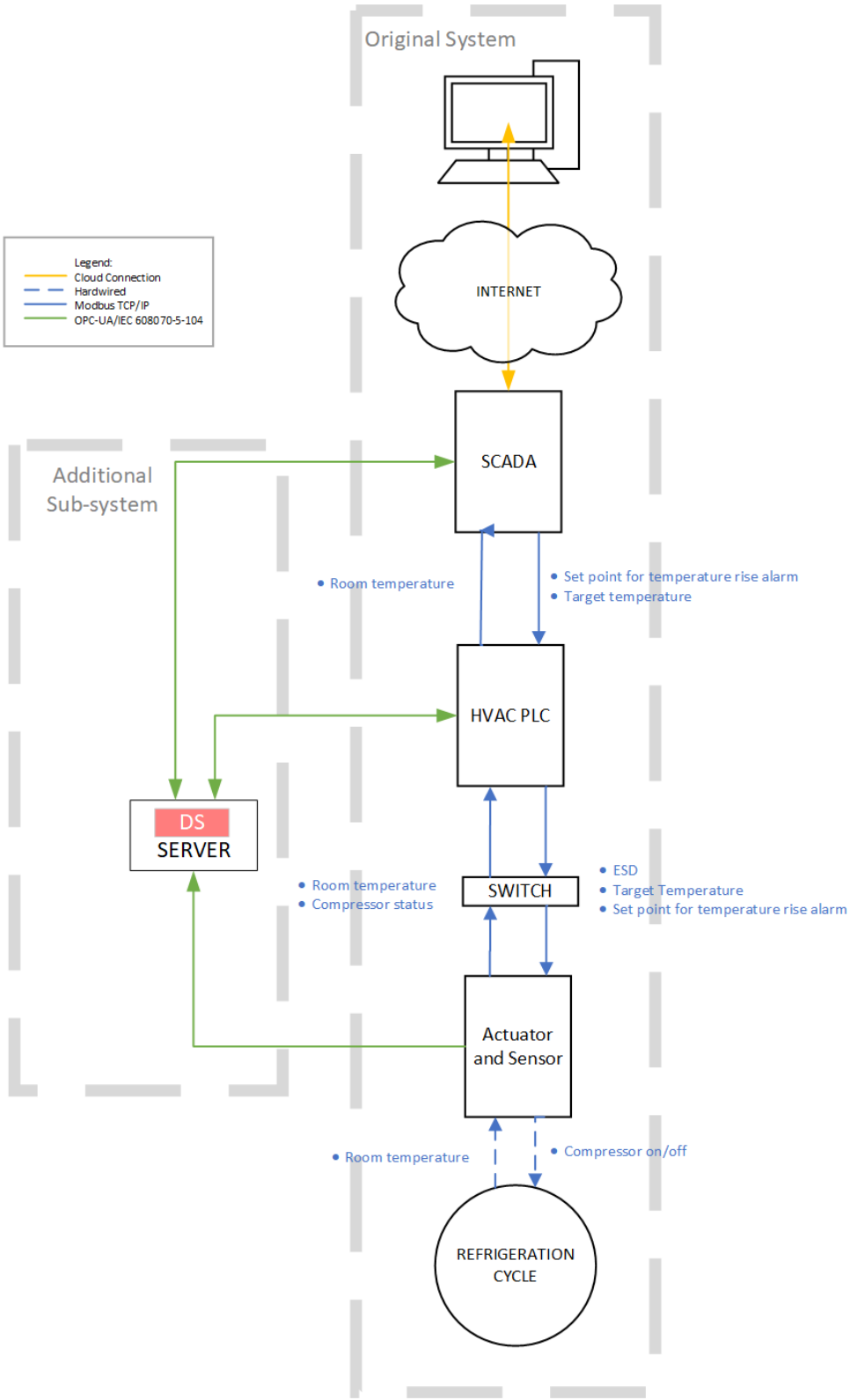

Figure 5-1 Control Interface Architecture of the Cooling System Utility.

5.1 Product Implementation

Due to the commonly limited processing power and memory of PLCs and to the difference in programming languages when compared to conventional servers or computers, the implementation of this system is envisioned on a separate server equipped with more computational power, which would communicate with the Switch. The server would receive information from the actuator, sensor and the parameters received by the system from the HVAC PLC and SCADA, run the LSTM network and communicate back the results (normal operation or detected anomaly) to the PLC, as shown in Figure 5-1.

When the server detects an anomaly in the system behavior, it alerts the PLC, which can then take appropriate actions. While this alert could be a pop-up notification in the SCADA system, it could also take shape as a more complex automated response, such as a switchover to a redundant system or system shutdown.

As studied in chapter 1.2, in a utility system as critical as the HVAC, reliability and availability are paramount. For this reason, it is envisioned that this IDS contains redundant and fail-safe measures. This can include a backup server to run the LSTM if the primary one fails, and a system for handling eventual false positives and negatives. This system could be a manual override, which operators can use to keep running the system under a false positive. This system should also not be swapped for usual cybersecurity measures, such as firewalls and routers, so that the plant would remain safe during a false negative.

As cyberattacks are constantly changing and developing, it is also envisioned that this IDS should be regularly updated with new data from the utility system to remain efficient. This could be a regular process, where the LSTM is automated to periodically retrain itself, or an event-driven process, where it is programmed to retrain itself when a new significant event, such as a new type of cyberattack, is observed.

Finally, protecting the IDS is also important, since its communication could also be attacked. For this, encryption is envisioned to prevent eavesdropping or data manipulation. Also, the server itself should be designed with best practices in cybersecurity such as authentication and regular patching. For this, a protocol like Modbus TCP/IP would not be ideal, and one such as OPC-UA or IEC 60870-5-104 would be preferred.

6. Refrigeration Model

In this chapter, we will discuss the development of a refrigeration system model in Simulink, utilizing the principles of Carnot's cycle. The model incorporates a controller to manage the compressor operation for refrigerating a specific room. The primary goal is to maintain the room temperature within a range of 15 to 20 degrees Celsius. The compressor is designed to turn on when the temperature deviates from the target range by a tolerance of 2 degrees.

Table 6-1 Variables used in the Model

$\dot{m}$	Mass flow rate of the refrigerant
$\dot{Q}_{in}$	Refrigeration capacity or heat transfer rate
h	Enthalpy
T_C	Temperature Outside
T_H	Temperature Inside
$\dot{W}_c$	Power input rate

6.1 Vapor-Compression System

Carnot's refrigeration cycle is an idealized model of a refrigeration system, operating on a closed loop and consisting of four processes: isothermal compression, adiabatic compression, isothermal expansion, and adiabatic expansion. The fluid used for this system was the R-134a, one of the suitable options for offshore refrigeration systems and is a more environmentally acceptable alternative to chlorine-containing refrigerants. Its properties are inserted into matlab through a table.

Consider the steady-state operation of a vapor-compression system, as depicted in Figure 6-1, where the major heat and work transfers are indicated by the arrows. Any changes in kinetic or potential energy are neglected in the analysis of the system components.

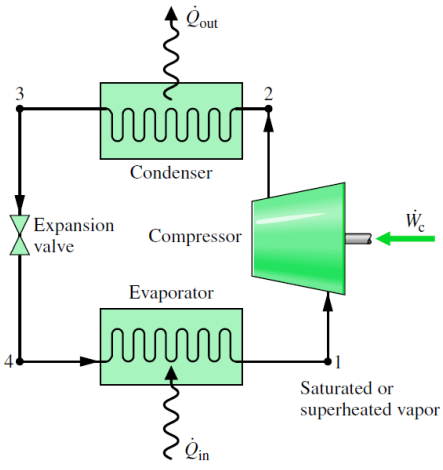

Figure 6-1 Vapor-Compression Cycle (Shapiro, 2014)

The process starts with the activation of the compressor. The signal to start the compressor undergoes a delay, which is introduced to account for the mechanical inertia of the system, the time for the electric motor to achieve the required speed, and other potential control systems.

The compressor is modeled as an ideal energy source. It compresses the refrigerant, raising its pressure and temperature. The compressor is designed to maintain a controlled mass flow rate, even when there is a variation in pressure. There is no flow resistance or heat exchange with the environment at this stage. The mass and energy rate balances for a control volume around the compressor can be expressed by the following equation.

$$\frac{\dot{W}_c}{\dot{m}} = h_2 - h_1$$

The refrigerant, now hot and under high pressure, proceeds into the condenser pipe. Here, the pipe flow dynamics are modeled, including the viscous friction losses, which occur due to resistance to the flow in the pipe, and convective heat transfer through the pipe wall. The condenser serves as a heat exchanger, releasing the absorbed heat from the refrigerated space and the heat produced by the compressor into the surrounding environment. This heat transfer process causes the refrigerant to condense. A control volume enclosing the refrigerant side of the condenser can be used to determine the rate of heat transfer from the refrigerant per unit mass of refrigerant flowing, which can be expressed using the next equation.

$$\frac{\dot{Q}_{out}}{\dot{m}} = h_2 - h_3$$

Next, the refrigerant, still under high pressure but now cooler, moves into the expansion valve. Here, $h_4 = h_3$. A block in the simulation mimics the drop in pressure that occurs due to the choking process. The refrigerant pressure decreases in an irreversible adiabatic expansion and exits the valve at state 4 as a 2-phase liquid-vapor mixture. During this stage there is no heat exchange with the environment.

The refrigerant, now a cool, low-pressure mixture, enters the evaporator pipe. Inside the evaporator, the refrigerant absorbs heat from a 5 m^2 room, leading to its evaporation. The evaporator, like the condenser, acts as a heat exchanger, but in the opposite sense - it absorbs heat rather than releasing it.

The heat absorbed from the room changes the phase of the refrigerant from a liquid to a gas, significantly cooling the room in the process. For a control volume enclosing the refrigerant side of the evaporator, the mass and energy rate balances reduce to give the rate of heat transfer per unit mass of refrigerant flowing as:

$$\frac{\dot{Q}_{in}}{\dot{m}} = h_1 - h_4$$

Finally, the gaseous refrigerant is drawn back into the compressor, marking the end of one refrigeration cycle.

6.2 Controller and Relay Operation

The compressor in the system does not have variable speed, and its operation is controlled by a relay. The controller monitors the difference between the room temperature and the target temperature. When this difference exceeds the predefined tolerance of 2 degrees, the controller sends a signal to the relay, which in turn activates the compressor.

6.3 Simulating the Outside Temperature

To simulate the outside temperature, the average summer temperature for an offshore location in the Baltic Sea was considered. This average value was added to a sine curve that represents the temperature fluctuations throughout the day, with higher temperatures at midday and lower temperatures at midnight, ranging at 8 degrees of amplitude. Additionally, short-term variations in the form of random spikes and falls of 1 to 2 degrees were added to account for unpredictable temperature changes.

It is important to note here, that in addition to the outside temperature, which impacts the cooling system, there are heavy consumers in such refrigerated rooms that radiate heat, such as transformers, shunt reactors and servers. Their quantity, placement and properties, however, vary significantly and thus have not been included in this model.

6.4 Temperature Rise Alarm Logic and Emergency Shutdown System (ESD)

To ensure the safety and functionality of the system, a temperature rise alarm logic is incorporated. If the room temperature rises beyond the acceptable limits, an alarm is triggered for the operator in the onshore control room. Furthermore, the system includes an emergency shutdown mechanism. If the HVAC PLC is sent this signal, all systems in this utility are immediately shut down.

In the simulation, this is achieved through multiplying the compressor operation with the converted Boolean stating whether it is true that the system is not under ESD (1) or false (0). Should the system be under ESD, the compressor will stop.

6.5 Man In the Middle Attack between the Switch and the Refrigeration Cycle

The simulation of a man-in-the-middle attack on the communication after the switch can be effectively studied by using a comparison method that helps determine the system's vulnerability and reaction to such cybersecurity risks. In this scenario, the hacker chooses to shut down the compressor permanently to heat up the room. This simulation method involves generating random attack times and comparing them to the simulation time to assess the start of the attack.

This works very similarly to the ESD logic, where if the random attack time is shorter than the current simulation time, the system sends a logical '1', which means that the system is healthy. On the other hand, if the random attack time is larger than the current simulation time, the system sends a logical '0', showing that the attack has started. This logical value is then converted into a double format to work with the next steps.

The double value is combined with the compressor's operational status by multiplying the two values. This ensures that the compressor operates normally until a simulated MITM attack is detected, at which point the compressor is shut down.

This attack will be referred to as MitM(2) for the remainder of the report.

6.6 Denial of Service Attack between the Switch and the Refrigeration Cycle

In the context of simulating a DoS attack, the focus is primarily on replicating the effects of a system freeze, which can adversely impact the functionality of the compressor. Under this scenario, if the compressor is on when the simulated attack occurs, it remains in a permanent state of activation, unable to stop its operation. Conversely, if the compressor is inactive during the attack, it will be unable to start, effectively rendering the system non-functional. To simulate this effect, a latch logic system utilizing NAND gates is employed.

A NAND latch consists of two cross-coupled NAND gates, where the output of one gate serves as an input to the other. This configuration allows the latch to store and maintain a binary value until an external signal alters the state. In the context of the DOS attack simulation, the NAND latch fixes the state of the compressor either permanently on or off, depending on the conditions at the time of the attack.

The same comparison tactic employed in the man-in-the-middle attack simulation is utilized to initiate the DoS attack. The logical value is then processed by the latch logic system, simulating the system freeze effect on the compressor.

6.7 Man in the Middle Attack at the SCADA

An examination of 10 prevalent server models reveals that their optimal operating temperatures typically span between 10 and 25 degrees Celsius, with the exception of some more costly servers, such as the Oracle Server X8-2, the Lenovo ThinkSystem SR630 and the IBM Power System S914, which range from 5 to 35 degrees Celsius. This analysis encompasses servers such as the Dell PowerEdge R740, Cisco UCS C220 and IBM Power System S914 (Environmental Requirements, 2021) (Dell EMC PowerEdge R740 Technical Specifications, u.d.) (Environmental Specifications, u.d.) (Family 9009+03 IBM Power System S914 (9009-41A), u.d.) (Lenovo Specifications, u.d.).

Consequently, to emulate this attack, the author elected to modify the target temperature to a value below the servers' functional threshold. In this manner, when the designated attack time is reached, as with the previous attacks, the target temperature is changed, and the room temperature is lowered well below the 15 to 20 degrees Celsius platform

standard and the 5 to 35 degrees Celsius servers' optimal operating range. As a result, the compressor works more constantly than before, striving to maintain the colder temperature.

This is achieved through automatically selecting a random value for the target temperature between -5 and 0 degrees in the code for each simulation.

This attack will be referred to as MitM(1) for the remainder of the report.

6.8 Man in the Middle Attack at the Switch

To simulate this attack on the refrigeration system, the ESD is activated and the threshold for the rise temperature alarm is increased. In this manner, as the system undergoes ESD and the compressor shuts down, the rise temperature alarm does not sound, and the room heats up. This is done through a switch logic added to the healthy scenario. The switch's threshold is the attack time. When the attack is not in place, the rise temperature alarm set point is its usual value of 293K. When the system undergoes this cyberattack, the switch turns to a step pulse to the previously selected random value.

This attack will be referred to as MitM(3) for the remainder of the report.

6.9 Data Generation

To generate data usable by the classification LSTM network, firstly data was extracted from the 5 simulation scenarios for the following parameters: time, room temperature, outside temperature, compressor status, temperature rise alarm, emergency shutdown signal and target temperature. The simulation scenarios, as explained were Healthy, DOS, MitM(1), MitM(2) and MitM(3), where MitM(1) is in regards to the attack to the communication between SCADA and HVAC PLC, MitM(2) is in regards to the attack between the Switch and the Refrigeration Cycle and MitM(3) is an attack directed at the switch. These were sampled every 10 seconds, resulting in 17280 samples per 24 hours. This sample rate allows for faster training of the network while not missing significant data to the dynamics of the system, of the outside temperature and of the cyberattacks. These were normalized to be used by the LSTM. The normalization process will be explained in chapter 6.9.6.

To generate data usable by the regression LSTM network, firstly data was extracted from the 5 simulation scenarios for the room temperature and compressor status parameters, with the addition of room temperature after 5 hours. Further, the model was made cyclic, so that data could be gathered throughout the entire 24 hours for all parameters.

For some of the simulations in each scenario, plots of the room temperature, compressor status and emergency shutdown signal by time were also saved for the finality of explaining each setup to the reader. Although these show a 24h period for the purpose of illustrating the reaction of the system to the attack, the used data varied in length, stopping 2 minutes after the attack. This is due to the aim of this research of obtaining a trained network

that functions as close as possible to the real-time scenario, while still allowing time for the system to react.

6.9.1 Healthy Scenario

In the healthy scenario, the room temperature is maintained within a range of 2 degrees of 290K and the compressor starts and stops according to the room temperature needs. Additionally, the spikes to the temperature are due to larger spikes in the outside temperature. The compressor also works unevenly throughout the day due to the natural changes of the outside temperature.

Figure 6-3 illustrates this healthy behavior, while Figure 6-4 illustrates a healthy behavior with the extra function of the emergency shutdown. For this purpose, the emergency shutdown signal was also graphed, although it is important to note that all parameters were fed into the LSTM network for all scenarios, not only the ones depicted in the following images. The healthy range is also shown in green, going from 288.15K to 293.15K.

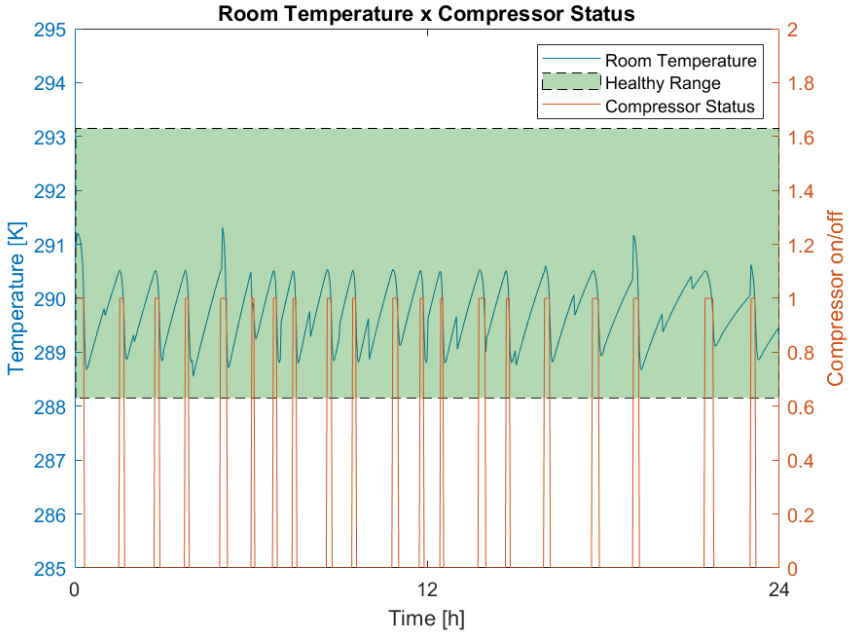

Figure 6-2 Room Temperature x Compressor Status of a Healthy Scenario.

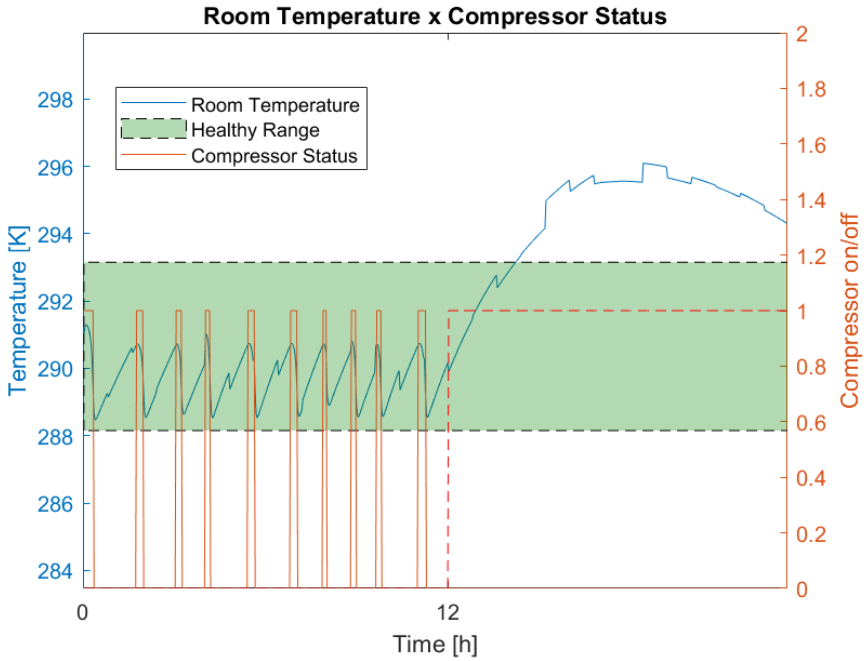

Figure 6-3 Room Temperature x Compressor Status and ESD of a Healthy Scenario.

6.9.2 Man in the Middle (1) Scenario

In this scenario, as the target temperature is lowered, one can see that the room temperature lowers and the compressor works with more frequency. In the case of Figure 6-4 the target temperature is lowered so significantly that the compressor does not turn off.

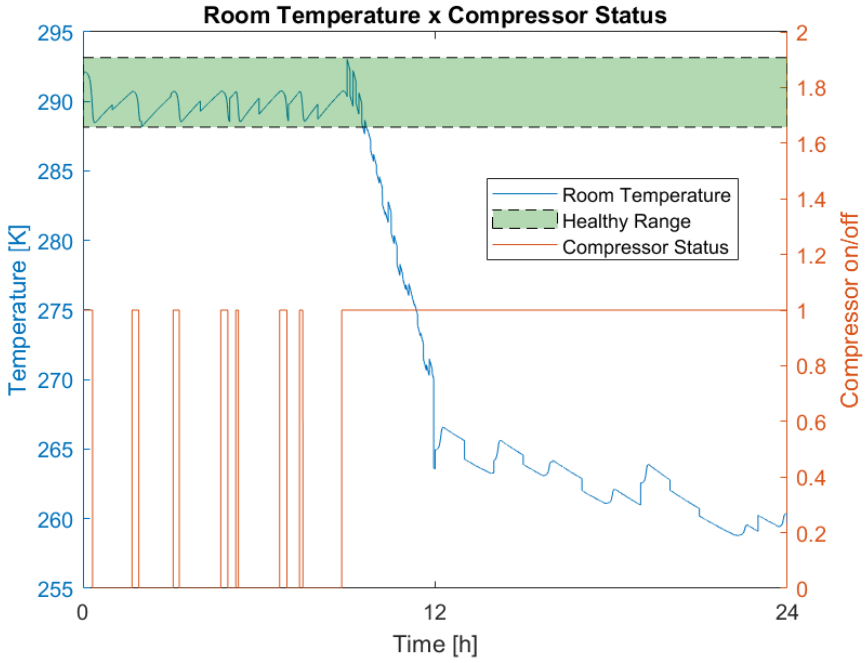

Figure 6-4 Room Temperature x Compressor Status of a Man in the Middle (1) Scenario.

6.9.3 Man in the Middle (2) Scenario

In this scenario, the compressor is shut off at will. Figure 6-5 illustrates a compressor being turned off and the temperature increasing closer to that of the outside temperature.

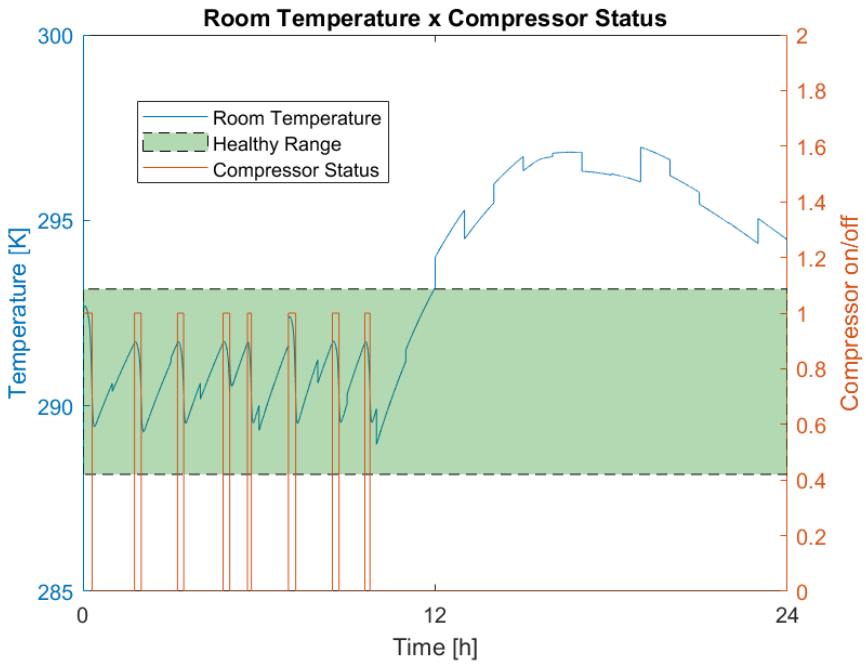

Figure 6-5 Room Temperature x Compressor Status of a Man in the Middle (2) Scenario.

6.9.4 Man in the Middle (3) Scenario

Here, the ESD function is switched on and the compressor stops. Further, the rise temperature alarm does not go off because its set point has been increased.

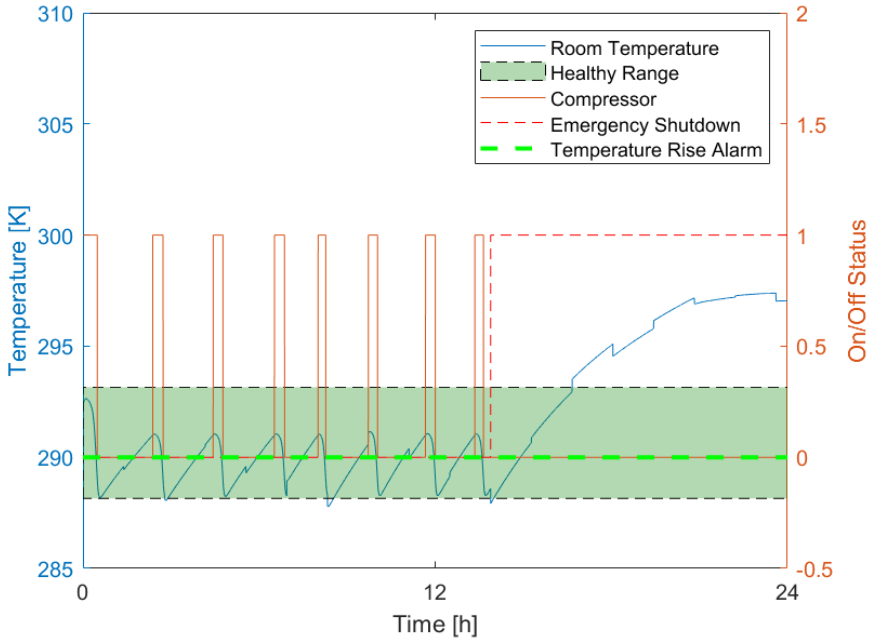

Figure 6-6 Room Temperature x Compressor Status, ESD and Temperature Rise Alarm of a Man in the Middle Scenario.

6.9.5 DOS Scenario

This scenario is represented by Figure 6-7 and Figure 6-8, where the first depicts a compressor that was already turned on by the time the attack struck, and the second a compressor that was already turned off.

Although Figure 6-7 looks similar to Figure 6-4, it is important to once more note that these are not the only parameters being fed into the neural network. While the Man in the Middle (2) attack would also show a change in target temperature, this one would not.

Figure 6-8 shows that the reaction to the system to the DOS attack is the same as the reaction to the Man in the Middle (2) attack. At this point in the communication there are no extra parameters to tell the two apart. For this reason, the testing accuracy for the neural network will only be used as a guiding parameter, and a confusion matrix will be used for the real analysis of the neural network's accuracy.

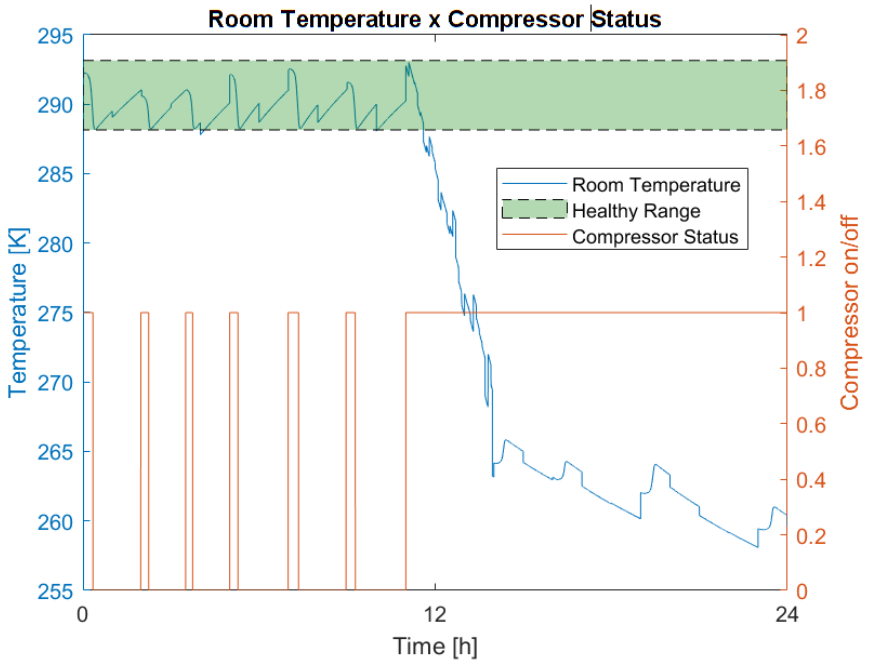

Figure 6-7 Room Temperature x Compressor Status of a DOS Scenario.

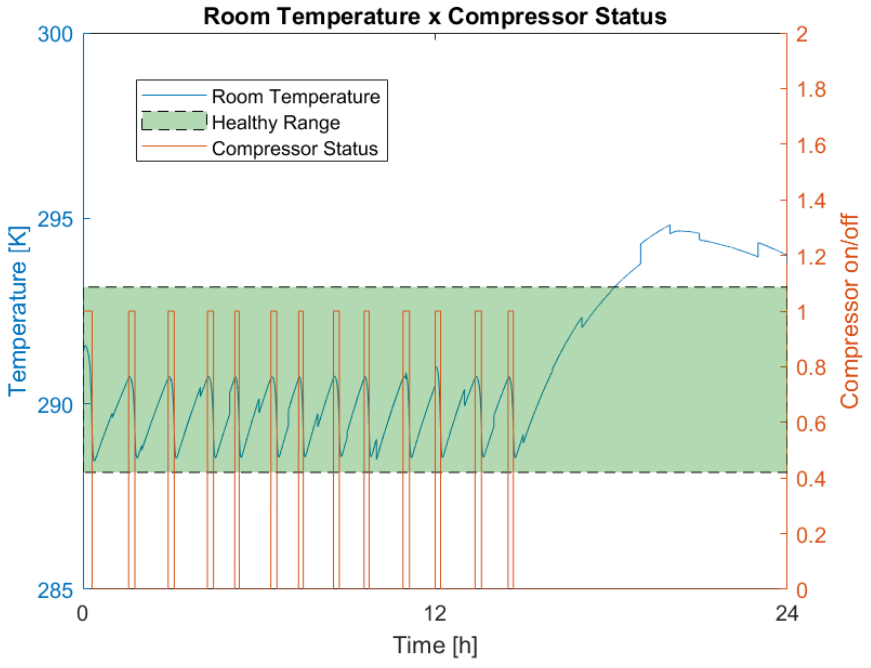

Figure 6-8 Room Temperature x Compressor Status of a DOS Scenario.

6.9.6 Data Generation for Classification

As mentioned, for the NN to have an attack detection time of 2 minutes, the actual datasets fed to the LSTM networks look differently than the previous figures. While Figure 6-9 shows an example in the structure previously seen in Figure 6-8, Figure 6-10 shows this data cut to detection time and Figure 6-11 illustrates one of the 1000 normalized datasets actually used by the LSTM in the classification training. Min-max normalization is a data preprocessing technique commonly used in machine learning, including Long Short-Term Memory (LSTM) networks. The process scales numerical data to a fixed range, usually 0 to 1, by subtracting the minimum value of the dataset from each data point, and then dividing by the range of the dataset (the difference between the maximum and minimum values). This scaling ensures all inputs to the LSTM network have a similar data distribution, reducing the chance of certain features dominating others due to their scale, and can help improve the accuracy and efficiency of the training process. Once more, these graphs are visual representations of two of the overall parameters used for classification.

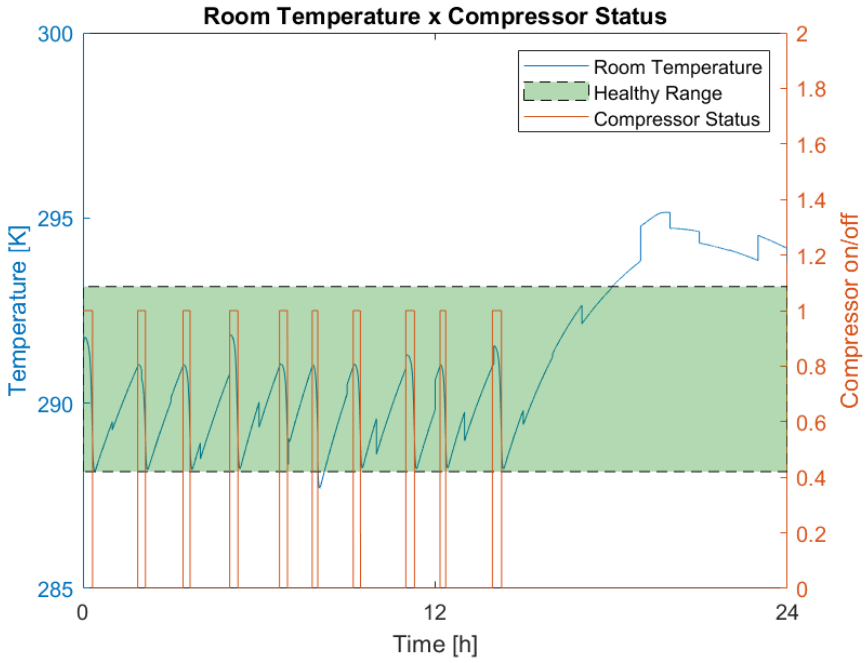

Figure 6-9 Example of 2 variables from a dataset obtained from DOS attack model.

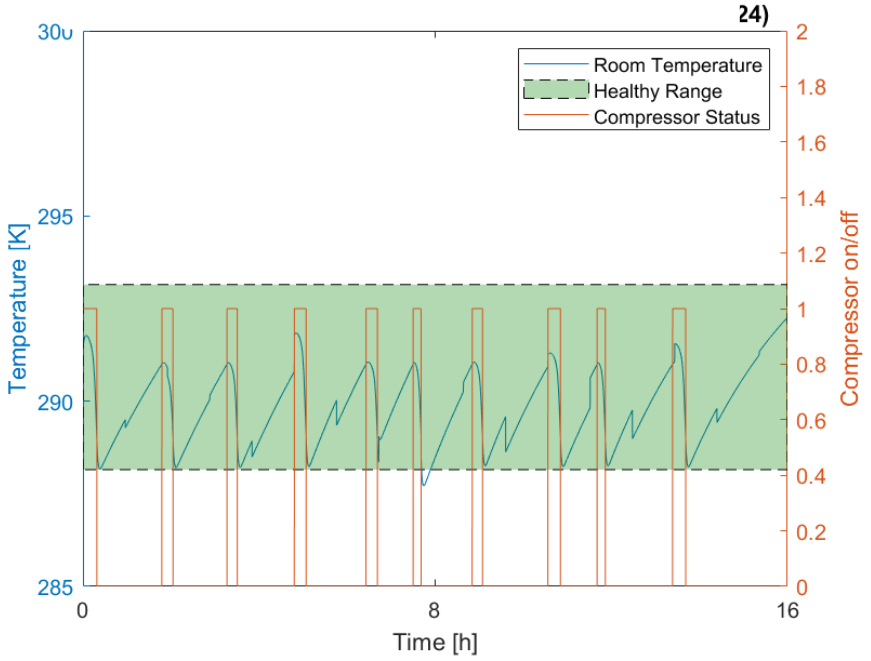

Figure 6-10 Example of 2 variables cut to 2 minutes after attack time.

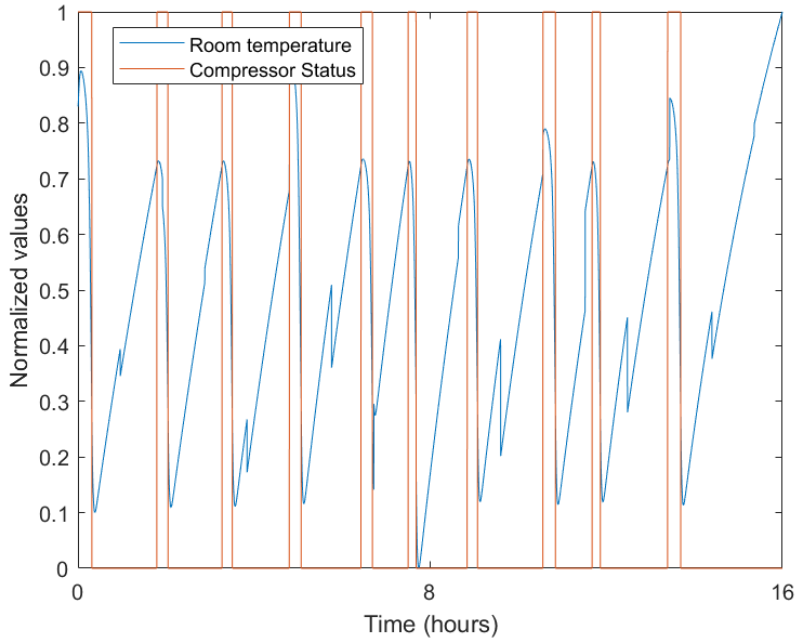

Figure 6-11 Example of two variables from a dataset used for DOS attack in the classification LSTM.

6.9.7 Data Generation for Regression

When all scenarios studied in this research are split to only show the parameters room temperature and compressor status, all can be explained in two main reactions: the compressor

stops and the temperature rises or the compressor remains on and the temperature falls. In order for the neural network to be able to forecast the room temperature based solely on the system behavior in these two parameters, all information concerning these variables from all scenarios were prepared and the data was generated for the NN to forecast the effect of a cyberattack on the system at hand after 5 hours within detection time.

For comparison purposes, the next figure contrasts the information given to the DOS attack detection regression network, as opposed to the classification network. In Figure 6-12, the attack detection time is at 4 o'clock. The graph on top reveals the original data retrieved from the simulation – the room temperature, the outside temperature and the room temperature 5 hours ahead. The graph below shows these three variables after being normalized, so that they range from 0 to 1. The graph illustrates one of the 200 datasets that was given to the neural network for the Man in the Middle (2) attack.

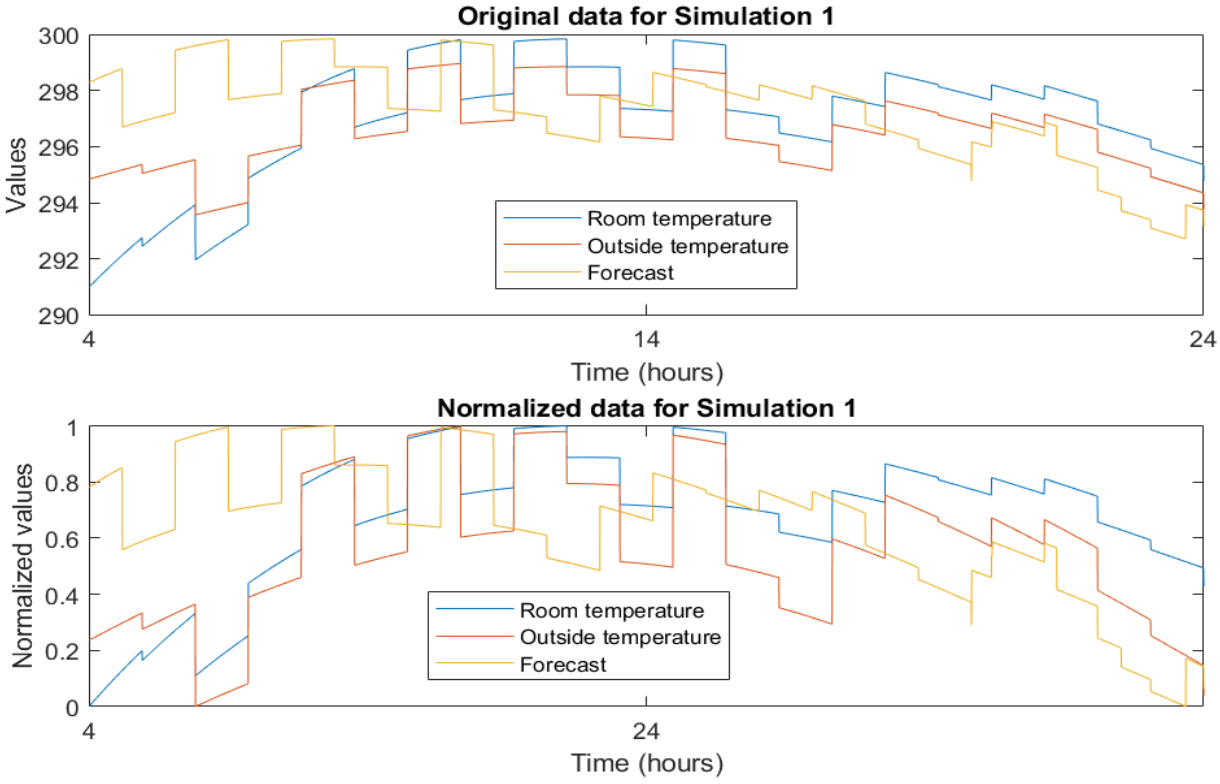

Figure 6-12 Example of dataset used for MitM (2) attack in the regression LSTM.

7. Neural Network

Deep Learning is a type of Machine Learning that imitates the human brain's computational abilities to make decisions. Through mathematical functions, it maps inputs to outputs and extract patterns from data, allowing for a relationship between variables. This is known as learning, with the process of learning referred to as training.

Artificial neural networks are used to achieve this, which recognize numerical patterns in transformed real-world data, such as images, sounds, text, or time series, using simple mathematical functions stacked in layers. Each layer contains neurons that receive information as inputs and combine them with a bias and a set of weights, which produce an output through a combination function. The activation function then determines the network's output based on its role.

7.1 Network Training

During the training of neural networks, various parameters are modified to achieve optimal performance, including the global learning rate, validation frequency, epochs, batch size, and solver, among others. In the next section, we will introduce these concepts and explain how they impact the network's ability to learn.

To train a neural network, the first step involves obtaining features from a database, such as sequences and their corresponding labels. These sequences are then fed into the network to generate predictions, which are compared to the true labels by computing the loss. The gradients of these calculations are then used to adjust the network's weights, and the process is repeated iteratively until the network's performance is satisfactory. This is observed in Figure 7-1

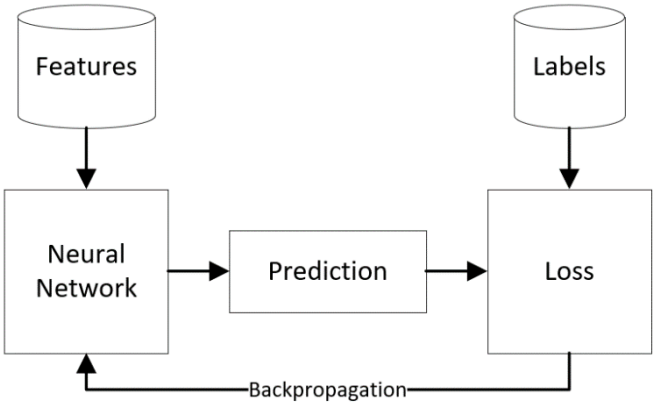

Figure 7-1 Illustration of the Training Cycle of a neural network.

The next paragraphs will explain terms that will be used throughout the analysis of the obtained networks.

A single execution of a for-loop or iterative update is referred to as an iteration, which could include updating network weights or making predictions. The validation frequency is the number of iterations between the assessments of the validation metrics (Options for Training Deep Learning neural network - MATLAB, u.d.).

The Learning Rate is a network parameter that determines how much the weights and biases are updated during training. This parameter communicates the magnitude of adjustment needed for the model to respond to the estimated error, every time the model's weights and biases are updated. If the learning rate is set too small, training may take a long time or require more computing power than is available. On the other hand, if the learning rate is too large, it may result in an unstable training process. The step size refers to the amount that the weights are updated, which ranges between 0 and 1. Hence, the learning rate controls how rapidly the model adjusts to the given problem.

This process involves multiplying a certain factor with the global learning rate, which is set in the configuration to determine the learning rate of the weights in the layer. The factor is also used in computing the step size by multiplying it with the gradient (Understand the Impact of Learning Rate on Neural Network Performance, 2019).

A complete iteration through the entire dataset is called an epoch in neural network training. However, because the gradient descent algorithm updates the weights in small increments, multiple epochs are necessary to train a network effectively.

The Batch Size determines how many samples will be processed at once in the network. There are two ways to present the training data to the network.

Hidden Units in a neural network serve as intermediaries, taking input data, transforming it via learned parameters (weights and biases), and communicating the transformed information to subsequent layers. These units extract and encapsulate complex data patterns, thus forming the network's internal representation of the data. Increasing the network with more hidden units enhances its capacity to recognize intricate patterns, while increasing the risk of overfitting. Conversely, fewer units may limit the network's learning capacity, potentially impairing performance.

A solver is responsible for optimizing the model by coordinating the forward inference and backward gradients to update parameters that minimize loss in the network. Throughout this project, various solvers were examined.

7.2 Network Validation and Testing

To avoid overfitting during the model training process, a validation dataset is used to assess the accuracy of the model while adjusting its hyperparameters. The validation dataset is separate from the training dataset and is not used during training. The model is evaluated on this validation dataset to avoid any bias towards the training dataset. 30% of the data for each scenario is separated from each scenario for this purpose.

Once the model is fully trained and the weights and biases are optimized, a separate test dataset is used to provide an unbiased evaluation of the final model's performance consisting of 10% of the overall amount of data for this experiment. This test dataset is independent of the training and validation datasets and is only used once the model is fully trained. By using a test dataset, an unbiased estimate of the overall accuracy of the model can be obtained.

7.3 LSTM Network

The Long Short-Term Memory network (LSTM) is a type of deep learning algorithm that is suitable for sequence prediction tasks since it is a recurrent neural network that can learn the order dependence of data, particularly time-series data. Unlike feed-forward neural networks, LSTMs are recurrent neural networks that utilize context in prediction tasks. In these networks, nodes are connected sequentially, which enables temporal dynamics.

7.3.1 Network Architecture

RNNs utilize a feedback loop for processing sequential data and making predictions, which may lead to two issues: vanishing gradient and exploding gradient. Vanishing gradient happens when the backpropagation process uses small values, resulting in a shrinking update that can reach zero, rendering the network unable to learn. On the other hand, exploding gradient occurs when the gradient is too great, requiring increasing amounts of computational power until the network crashes.

LSTM has a forget rate, a feature that distinguishes signals that have already gone through the network. By allowing signals to pass through the forget rate with a value between 0 and 1, LSTM can maintain a sequence of signals and solve the update gradient problem of RNNs (So, 2019).

There are two network architectures used in this project and it was planned for the LSTM architecture to remain as simple as possible while achieving the project goals due to the speed of the training in accordance with the weight of the data files and complexity in scalability.

Figure 7-2 illustrates the first network. Its architecture is a design that effectively balances simplicity and speed with accuracy. Beginning with a Sequence Input layer, it receives and processes sequence data while preserving the critical order of individual elements. This data is then passed onto an LSTM layer, whose unique capability to recall past contextual information is ideal for processing sequences of data. The LSTM layer's output is then densified through a Fully Connected layer, which, by linking every node of the LSTM output to every node of the following layer, fosters the learning of intricate relationships within the data. A Softmax layer then takes these complex relationships and maps them onto probabilities that sum to one, enabling a probabilistic understanding of the class membership of each input. Finally, a Classification Output layer is used, providing discrete predictions for the class of each input.

The second network, illustrated by Figure 7-3, specifically designed to handle the more complex task of forecasting. This starts with the Sequence Input layer accepting sequence inputs, as with the previous network, which is then processed by the first LSTM layer. This data then undergoes a normalization process in the Batch Normalization layer to ensure stability and efficiency during training, mitigating the problem of internal covariate shift. This stable output enters another LSTM layer, which, layered atop the previous one, allows the network to learn more intricate temporal abstractions, thus enhancing its ability to decode complex sequences. The output of the LSTM layer is then fed into a Fully Connected layer, establishing an extensive network of nodes that promotes the learning of complex relationships. To avoid overfitting and ensure a uniform distribution of learned features, a Dropout layer is incorporated, which randomly nullifies a fraction of input neurons during each training update. Post dropout, another Fully Connected layer further fine-tunes the learning from the abstract representations, mapping them onto the final output. Lastly, the Regression Output layer, as opposed to predicting discrete labels, predicts a continuous value. To achieve this final architecture, various architectures were trained and tested, and while this is a slightly more complex alternative than the first, it is the simplest network achieved while maintaining acceptable results to the author.

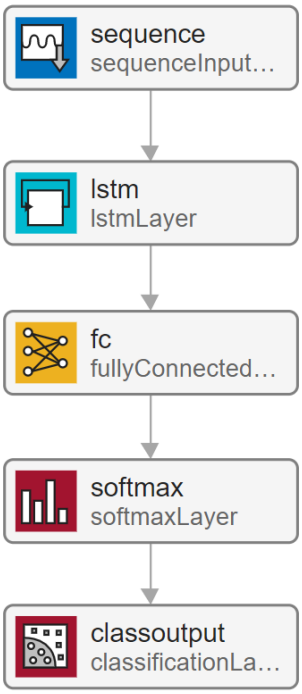

Figure 7-2 Illustration of the used classification LSTM architecture.

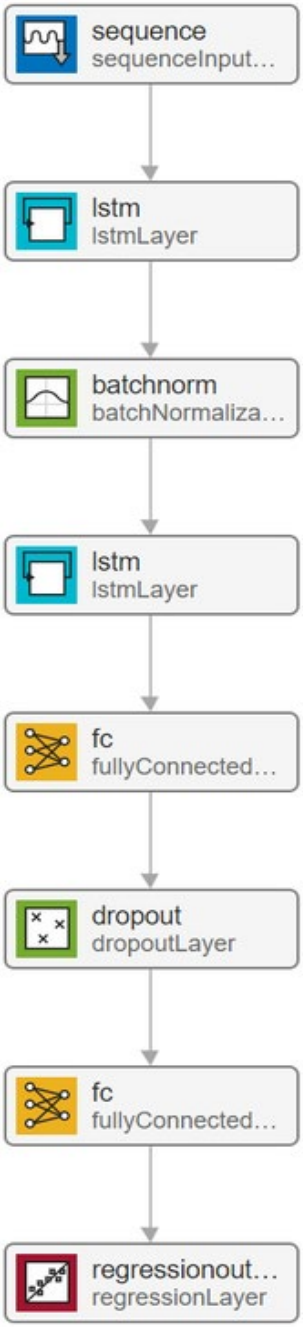

Figure 7-3 Illustration of the used regression LSTM architecture.

8. Results

The simulations were used to get 200 datasets from each scenario. Their normalized results were split into 60% training data, 30% validation data and 10% testing data equally between each scenario. Therefore, 120 datasets were used for training, 60 datasets were used for validation and 20 datasets were used for testing for each scenario.

Multiple versions of the networks were trained with intensive search over the following hyper-parameters: number of hidden units, number of max epochs, mini batch size, initial learning rate and validation frequency. Different solvers were also tested against each other.

8.1 Classification

The networks chosen as the final products for each solver were tuned to the following parameters and trained to the following accuracies and time:

Table 8-1 Tuning Parameters and Network Training Time and Accuracies.

Solver	Hidden Layers	Max Epochs	Mini Batch Size	Learning Rate	Validation Frequency	Training Time [mm:ss]	Validation Accuracy	Testing Accuracy
Adam	30	15	8	0.0004	30	63:23	93.33%	97%
sgdm	20	15	12	0.0004	30	43:41	92%	96%
rmsprop	20	13	8	0.0004	30	52:26	88.33%	92%

The confusion matrix for the testing set can be seen in Figure 8-1. This network classifies with 78% accuracy for the test set.

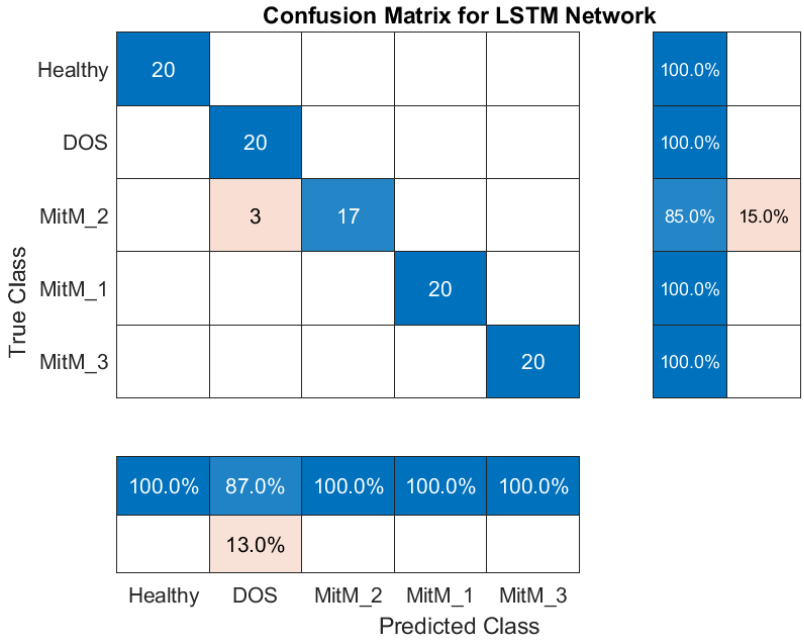

Figure 8-1 Confusion Matrix for the final Adam network.

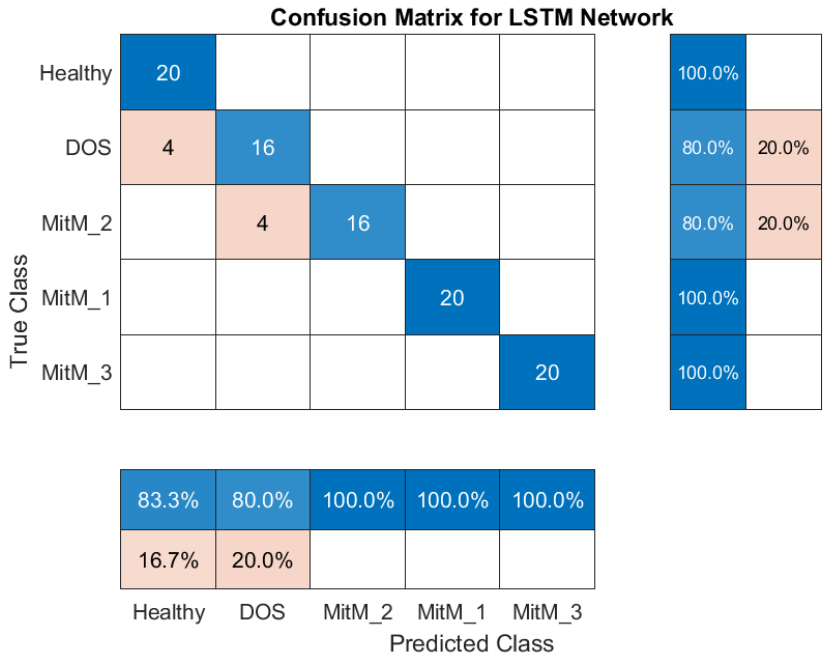

Figure 8-2 Confusion Matrix for the final rmsprop network.

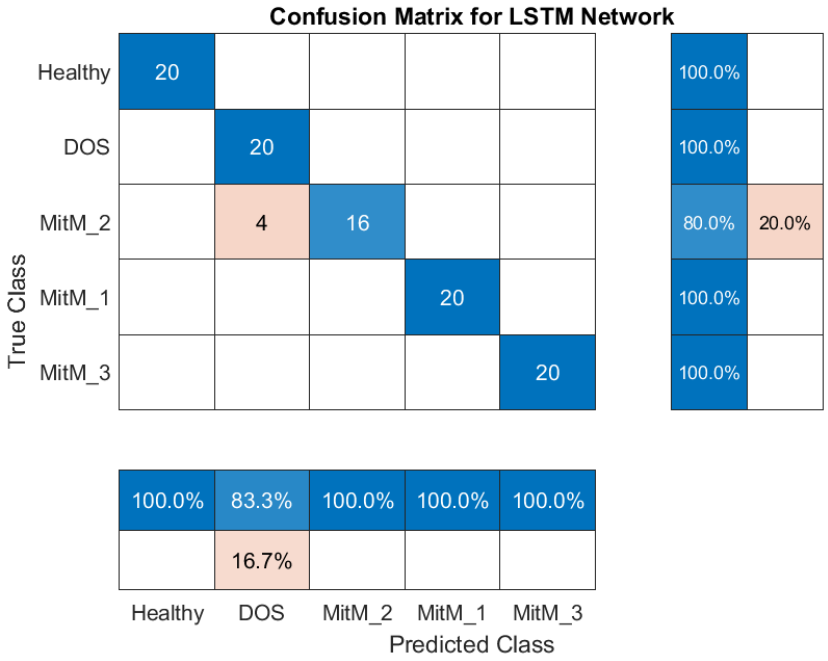

Figure 8-3 Confusion Matrix for the final sgdm network.

8.2 Regression

While three networks were worked on for classification, two networks were achieved for the regression based on the system behavior, and thus its two parameters room temperature and compressor status. When all scenarios studied in this research are split to only show these two parameters, all can be explained in two behaviors: the compressor stops and the temperature rises (learned by network 1) or the compressor remains on and the temperature falls (learned by network 2). In this manner, all information concerning these parameters from all scenarios were fed into these two networks.

For both were used: Adam solver, 199 hidden layers, learning rate of 0.0064 and validation frequency of 5. Table 8-2 depicts the different parameters and outputs for both networks, including the RMSE's in Kelvin for the predicted room temperature.

Table 8-2 Difference in Parameters and Outputs from Regression NN 1 and 2.

NN	Mini Batch Size	Training Time [mm:ss]	Validation RMSE	Testing RMSE	Validation RMSE in K	Testing RMSE in K
1	32	64:43	0.18977	0.19076	1.3521	1.3596
2	12	77:49	0.17569	0.17326	1.2366	1.2195

Figure 8-2 and Figure 8-3 illustrate the original data of one test dataset versus the predicted data based on the current room temperature and the compressor status. These graphs start 5 hours and 2 minutes after the attack, which for the first network is at 23 o'clock of the day before, and for the second network at 19 o'clock of the day before.

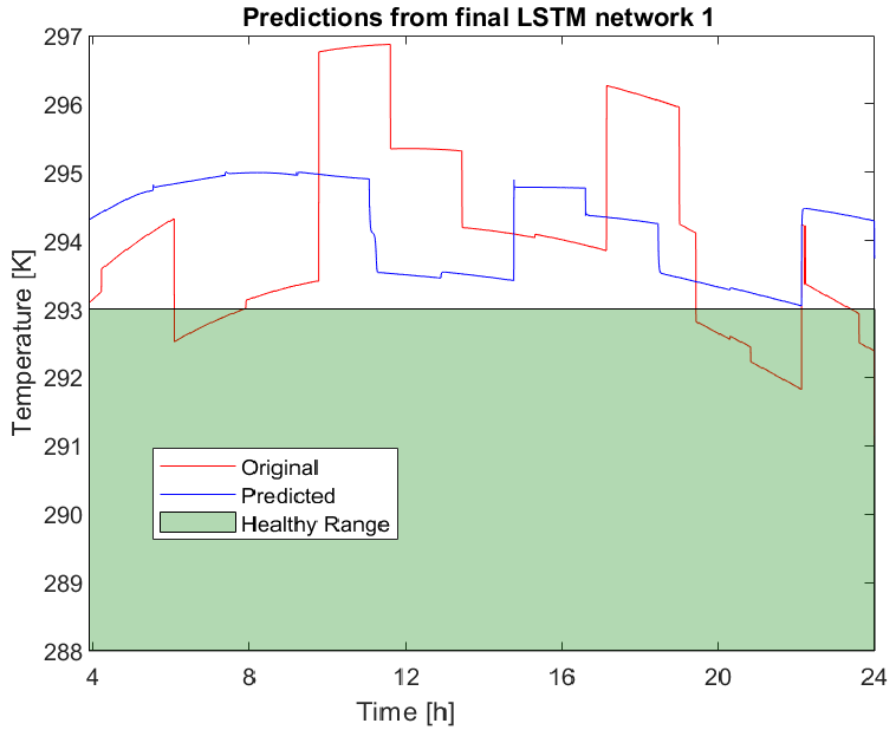

Figure 8-2 Original data x Predicted data of one test dataset for Network 1.

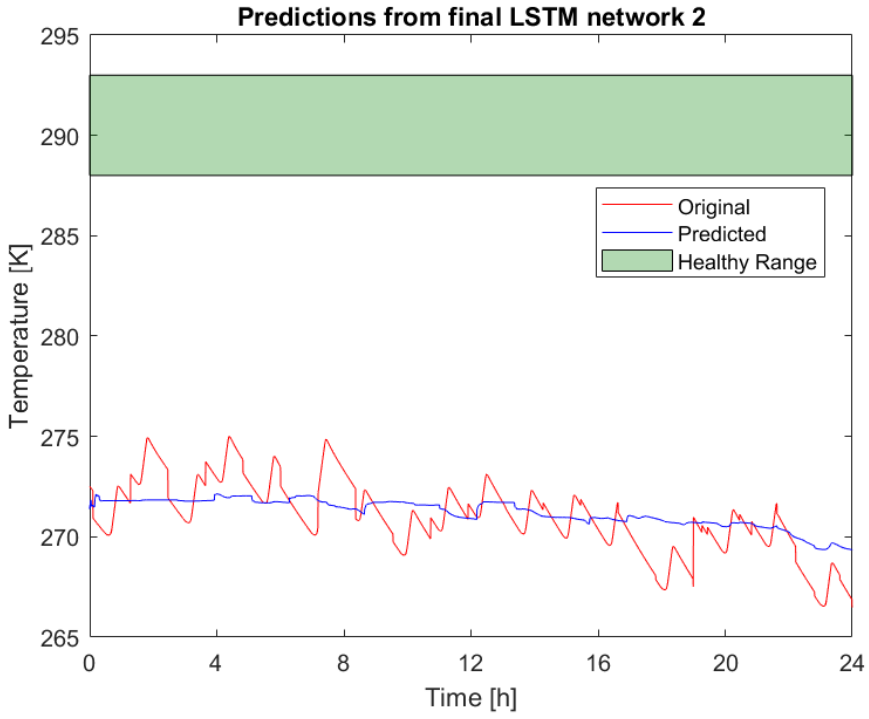

Figure 8-3 Original Data x Predicted Data of one test dataset for network 2.

9. Analysis

9.1 Detection and Diagnosis of Classification

For the detection and diagnosis of the 4 cyberattacks, classification networks were trained, validated and tested with 200 datasets of each scenario, including behavior reactions from cyberattacks and healthy operation. These were sampled at 10-second intervals from simulation runs. Importantly, each dataset was initiated two minutes into the system's reaction to the cyberattack, ensuring the system had time to respond. For optimal neural network performance, all datasets underwent min-max scaling to ensure features resided within a 0 to 1 range.

Building upon the foundational principles of neural network architecture, the selection of the activation function in this context was critical to ensuring the network's ability to learn effectively. The ReLU (Rectified Linear Unit) activation function was chosen for its computational simplicity and performance efficiency. Its non-saturating nature made it a suitable choice for the initial layers of the network. Simultaneously, the softmax activation function was employed in the output layer, providing a probabilistic distribution across the classes and hence facilitating the classification task.

The chosen network architecture, though elementary, yielded satisfactory outcomes, nullifying the need for exploring alternative architectures. However, it did require fine-parameter tuning. Adam, the chosen solver, outperformed others, delivering 97% accuracy on the test dataset and misclassifying merely three cyberattacks.

As previously mentioned, as a man in the middle (2) attack and a DOS attack may look the same depending on the status of the compressor when the DOS strikes, this number may be given to chance, as the network would not have a way to know the difference, and thus it is speculated that although the same number of datasets for each scenario was organized into the test database, more DOS attacks which struck when the compressor was on were selected. The Stochastic Gradient Descent with Momentum (SGDM) network's four mistakes and Rmsprop's 8 mistakes between the DOS and MitM(2) can also be explained by this.

Finally, the Rmsprop solver offered the lowest testing accuracy, givinge 4 false healthy scenarios, deeming it only 80% reliable in this case, although it was trained, validated and tested on the same database as the other networks.

Despite the minor misclassifications, Adam and SGDM exhibited high accuracy in all scenarios except for DOS and MitM, where the two attacks can mirror each other in their manifestations. Therefore, the misclassifications in these cases were deemed acceptable. Upon evaluating the performance metrics, SGDM stood out due to its fewer hidden layers, leading to faster training time, an essential aspect considering scalability and maintenance in real-world offshore applications. Therefore, SGDM emerged as the most efficient classification network in this study.

9.2 Forecasting of Regression

In forecasting room temperature five hours post-detection of a cyberattack, this study worked with two distinctive networks aligned with the binary nature of compressor operation scenarios: one representing a shutdown condition and the other a continuously operational state. The data for these divergent scenarios were categorized, normalized, and fed into the respective networks.

Several network architectures were trained, but the final choice is more complex than the classification networks discussed previously. The additional complexity is given from a few extra layers, contributing only slightly to the extended training duration.

The performance of both networks was impressive, with minimal deviations in their temperature forecasts. The first network, addressing the shutdown scenario, reported a testing Root Mean Square Error (RMSE) of 1.4 Kelvin, indicating minimal predictive discrepancies. The second network, tailored for the continuous compressor operation scenario, fared even better, with an RMSE of 1.2 Kelvin. This remarkably small error defends the robust predictive capabilities of the chosen regression networks.

10. Conclusion

In bringing this thesis to a conclusion, the advances made in the domain of cyberattack detection and countermeasures are clear. This investigation has not only elevated comprehension of cybersecurity as applied to offshore refrigeration systems but also initiated an approach by focusing on system behavior rather than the conventional method of network traffic analysis.

Revisiting the task statement:

Can a NN detect and diagnose a cyberattack in the Modbus TCP/IP communication between SCADA and a refrigeration system from an offshore platform in real-time? If so, can a NN be trained to forecast the reaction of said system to the cyberattack so as to aid in the post-attack procedure?

In regards to the classification, the SGDM solver emerged as a robust tool in this study, achieving not only some of the highest accuracy in all classes on the test dataset, but also in the shorter training time and most efficient fine-parameter tunings.

The two networks for forecasting room temperature following a detected cyberattack presented unique insights into the different compressor operational states. The use of system responses to forecast parameters presented a more context-specific and nuanced understanding of post-cyberattack conditions.

In summarizing this thesis, it is evident that this approach has taken a different perspective than the ones mentioned in chapter 2.2, using a model that can give as much data as needed, using LSTM with the simplest architectures possible while maintaining satisfactory results and learning from system reactions rather than network traffic. The findings derived from this offer an alternate viewpoint to how cybersecurity can be designed, delivering accurate and useful information towards the considered application.

11. Discussion

This chapter reflects on several aspects of the research, acknowledging potential limitations and areas for improvement, as well as recognizing the substantial strides taken in understanding cyberattacks within the context of refrigeration system operations.

One significant point to consider is the lack of server and control systems' heat dissipation from the model, which would reside in these refrigerated rooms in real platforms. This causes a variation in the data since server equipment is expected to generate heat throughout the day, irrespective of the external temperature. While the lack of this factor might not drastically affect the results of the networks, incorporating it could undoubtedly enhance the model's realism and provide a more comprehensive understanding of the system. However, it's worth mentioning the challenge this integration presents, given the uniqueness of every platform in terms of equipment number and brand. While it would be possible to base the model on a specific design, it would still need updating of the networks for each unique installation.

Furthermore, the absence of real data could be perceived as a shortfall in bringing realism to the model. Without real data to validate the model, one can only rely on theoretical calculations and cannot guarantee the system's realistic representation. It further emphasizes the need for onsite network updating. Although a more extensive dataset might not negate the necessity of updating, it could potentially enhance the network's efficiency.

A considerable challenge in the research process was the time-intensive aspect of the experiments, primarily due to the vast amounts of data. Each experiment took up to two hours, making the process of obtaining reliable hyper-parameters quite exhaustive. Nonetheless, by investing ample time in teaching the network in various ways and exploring different solvers for the fastest convergence, reliable and accurate hyper-parameters were successfully identified for the final networks.

Concerning the regression networks, a crucial issue was determining an optimal order for the extra layers. While the possibilities are infinite, simplicity was the goal. After several experiments, the chosen architecture demonstrated impressive performance with both regression networks.

Despite the discussed challenges and limitations, the networks and techniques developed and utilized throughout this work have proven effective, demonstrating their potential for practical application in real-world scenarios.

On the implementation and scalability of these, as mentioned, the simplest of architectures have been used to aid in this process, as well as real-life disturbances such as temperature oscillations to approximate the model to a real-life scenario. However, as all platforms are unique in structure, equipment and location, the NNs need to be given new data and updated in each case of application.

12. Future Work

The development of this project provided an opportunity to delve into an important domain: real-time detection and diagnosis of cyberattacks on the Modbus TCP/IP communication between a refrigeration system and SCADA, and the subsequent forecasting of system behavior on an offshore platform. While the utilization of Long Short-Term Memory (LSTM) neural networks served as a powerful tool to predict and classify various cyberattack scenarios, it's also vital to appreciate the scope for enhancement and continuous improvement. Several opportunities for further development have been identified, each potentially contributing towards an even more robust and accurate predictive model.

First, and most importantly, future work could involve the inclusion of heat dissipation from actual equipment in the LSTM model. By considering this, the model can more accurately simulate real-world operations of the refrigeration system in offshore platforms. Incorporating such an extra layer of complexity could significantly increase the precision of the model, and therefore the system's responses and forecasts. This implementation would require the collection of reliable data regarding equipment heat dissipation patterns, necessitating further collaboration with field engineers or thermal physicists. The difficulty of this lies in the uniqueness of each platform, thus signaling a primary need for a study such as equipment surveys.

Second, the use of actual data rather than simulated data can significantly improve the efficacy and realism of the LSTM model. Real data inherently includes a wide range of fluctuations and anomalies that, although attempted, may not be accurate to real environments. With real data, the model can better learn to identify and respond to subtle patterns and discrepancies, thus improving the detection and diagnosis of these cyberattacks. This could be achieved by coordinating with vendors of these systems to collect such data.

Third, there is potential to combine both the classification and regression LSTM models into a single, unified regression LSTM. This unification could lead to more efficient data processing, potentially improving both the speed and accuracy of cyberattack detection. The practicalities of this integration would involve developing a hybrid LSTM architecture that can handle both classification and forecasting tasks concurrently. This would require advanced algorithmic design and intensive testing to ensure the unified model preserves the efficacy of its distinct shares.

Moreover, the predictive model's capabilities could be significantly enhanced by incorporating parameters of the Intrusion Detection System (IDS) itself into the LSTM training. By including data about the behavior and performance of the IDS, the LSTM can learn patterns associated with system failures or vulnerabilities, further improving the detection of potential cyber threats.

Finally, the use of year-round temperature data in LSTM training rather than only summer data could significantly improve the model's forecasting applicability. This allows for a more comprehensive understanding of the refrigeration system's behavior under various temperature conditions, further enhancing the model's predictive capabilities.

In conclusion, while the current project has demonstrated promising results, there is also much potential for future work. By refining the model design, updating the network's architecture, and collaborating more closely with vendors, the application capabilities of the LSTM model could be significantly improved. This work represents just one stepping-stone in the journey towards a safer, more secure digital future in offshore industrial operations.

Bibliography

Acromag. (2005). *Introduction to modbus TCP/IP.* Acromag.

Alrawais, A., Alshammari, R., & Al-Dweik, A. (2019). Network intrusion detection system using machine learning: a comprehensive review. *Journal of Ambient Intelligence and Humanized Computing, 10*, 457-478.

Alvarez, Michelle; IBM Security. (2020). *X-Force Threat Intelligence Index 2020.* IBM Security.

Bhatia, S., Parian, C., & Guldiman, T. (2020). *Fooling the Master: Exploiting Weaknesses in the Modbus Protocol.*

Byres, E. (2004). *The Use of Attack Trees in Assessing Vulnerabilities.*

Classification of Intrusion Detection Systems. (n.d.). Check point: https://www.checkpoint.com/cyber-hub/network-security/what-is-an-intrusion-detection-system-ids/

Dell EMC PowerEdge R740 Technical Specifications. (n.d.). Dell Technologies: https://www.dell.com/support/manuals/da-dk/poweredge-r740/per740_techspecs_pub/standard-operating-temperature?guid=guid-7c06327e-0776-43bd-8347-0604f24b1598&lang=en-us

Edeh, D. I. (2021). *Network Intrusion Detection System Using Deep Learning Technique.* University of Turku.

Endpoint Security Market. (n.d.). Markets and Markets: https://www.marketsandmarkets.com/Market-Reports/endpoint-security-market-29081235.html

Environmental Requirements. (2021). Oracle: https://docs.oracle.com/en/servers/x86/x8-2l/installation-guide/gqqor.html

Environmental Specifications. (n.d.). Cisco: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/c/hw/C220M5/install/C220M5/C220M5_appendix_011.html

Family 9009+03 IBM Power System S914 (9009-41A). (n.d.). IBM: https://www.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_sm/3/897/ENUS9009-_h03/index.html&lang=en&request_locale=en

Ferguson, S. (2019, March 27). Ransomware Attack Costs Norsk Hydro $40 Million - So Far. *Bank Info Security*. https://www.bankinfosecurity.com/ransomware-attack-costs-norsk-hydro-40-million-so-far-a-12269

Fortinet. (2019). *Significant SCADA/ICS Security Risks.* Fortinet.

Ginter, A., & Hale, G. (2021). *OT Secuirty Incidents - 2021 Trends and Analyses.* Waterfall.

Greenber, A. (2015, December 28). How the Internet of Things Got Hacked. *Wired*. https://www.wired.com/2015/12/2015-the-year-the-internet-of-things-got-hacked/

IBM. (n.d.). *IBM*. What is SIEM?: https://www.ibm.com/topics/siem

Jr, S. E. (2013). *PLC Code Vulnerabilities Through SCADA Systems.* University of South Carolina. https://scholarcommons.sc.edu/cgi/viewcontent.cgi?referer=&httpsredir=1&article=1804&context=etd

Khan, M., Khan, M., Ullah, S., Ahmad, j., Jamal, S., Shah, A., Pitropakis, N., & Buchanan, W. (2021). *A Deep Learning-Based Intrusion Detection System for MQTT Enabled IoT.* Internet of Things for Industrial Applications.

Lee, A., Wogan, H., Kearns, J., & Luther, I. (2022). *Ports and Terminals Cybersecurity Survey.* Jones Walker. https://safety4sea.com/wp-content/uploads/2022/10/Jones-Walker-2022-Cyber-Security-Survey-2022_10.pdf

Lenovo Specifications. (n.d.). Lenovo: https://pubs.lenovo.com/sr630/server_specifications

Maheswari, M., & Karthika, R. (2021). *A Novel Hybrid Deep Learning Framework for Intrusion Detection Systems in.* Tech Science Press.

Options for Training Deep Learning neural network - MATLAB. (n.d.). Mathworks: https://se.mathworks.com/help/deeplearning/ref/trainingoptions.html

Priyalakshmi, V., & Devi, R. (2022). *Evaluation of Efficient Classification Algorithm.* IJarsct.

Rahman, M., Hassan, R., & Ibrahim, Z. (n.d.). Anomaly detection for industrial IoT: a neural network approach. . *Journal of Industrial Information Integration, 12*, 1-8.

Shapiro, H. N. (2014). *Fundamentals of Engineering Thermodynamics.* Wiley.

So, G. (2019, March 29). *Should We Abandon LSTM for CNN?* Medium: https://medium.com/ai-ml-at-symantec/should-we-abandon-lstm-for-cnn-83accaeb93d6

Timberg, C., Yang, J. L., & Tsukayama, H. (2013, December 19). Target says 40 million credit, debit cards may have been compromised in security breach. *The Washington Post*.

https://www.washingtonpost.com/business/technology/target-data-breach-affects-40-million-accounts-payment-info-compromised/2013/12/19/5cc71f22-68b1-11e3-ae56-22de072140a2_story.html

Trend Micro. (2022). *The State of Industrial Cybersecurity.* Trend Micro.

Understand the Impact of Learning Rate on Neural Network Performance. (2019, January 25). Machine Learning Mastery: https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/

What is modbus TCP Protocol? (2021, January 23). PLC Ynergy: https://plcynergy.com/modbus-tcp-protocol/

Wolf, C. (2021, April 1). *Breaking Down the Pros and Cons of AI in Cybersecurity*. ASIS International: https://www.asisonline.org/security-management-magazine/monthly-issues/security-technology/archive/2021/april/breaking-down-the-pros-and-cons-of-ai-in-cybersecurity/

Zhao, F., & Lee, J. (2021). *GWEC Global Wind Report.* GWEC.