applied sciences

MDPI

Article

A Multi-Model Proposal for Classification and Detection of DDoS Attacks on SCADA Systems

Esra Söğüt * and O. Ayhan Erdem

Department of Computer Engineering, Faculty of Technology, Gazi University, Ankara 06560, Turkey; ayerdem@gazi.edu.tr
* Correspondence: esrasogut@gazi.edu.tr

Abstract: Industrial automation and control systems have gained increasing attention in the literature recently. Their integration with various systems has triggered considerable developments in critical infrastructure systems. With different network structures, these systems need to communicate with each other, work in an integrated manner, be controlled, and intervene effectively when necessary. Supervision Control and Data Acquisition (SCADA) systems are mostly utilized to achieve these aims. SCADA systems, which control and monitor the connected systems, have been the target of cyber attackers. These systems are subject to cyberattacks due to the openness to external networks, remote controllability, and SCADA-architecture-specific cyber vulnerabilities. Protecting SCADA systems on critical infrastructure systems against cyberattacks is an important issue that concerns governments in many aspects such as economics, politics, transport, communication, health, security, and reliability. In this study, we physically demonstrated a scaled-down version of a real water plant via a Testbed environment created including a SCADA system. In order to disrupt the functioning of the SCADA system in this environment, five attack scenarios were designed by performing various DDoS attacks, i.e., TCP, UDP, SYN, spoofing IP, and ICMP Flooding. Additionally, we evaluated a scenario with the baseline behavior of the SCADA system that contains no attack. During the implementation of the scenarios, the SCADA system network was monitored, and network data flow was collected and recorded. CNN models, LSTM models, hybrid deep learning models that amalgamate CNN and LSTM, and traditional machine learning models were applied to the obtained data. The test results of various DDoS attacks demonstrated that the hybrid model and the decision tree model are the most suitable for such environments, reaching the highest test accuracy of 95% and 99%, respectively. Moreover, we tested the hybrid model on a dataset that is used commonly in the literature which resulted in 98% accuracy. Thus, it is suggested that the security of the SCADA system can be effectively improved, and we demonstrated that the proposed models have a potential to work in harmony on real field systems.

Keywords: critical infrastructure; SCADA; cybersecurity; DDoS; deep learning; testbed

Citation: Söğüt, E.; Erdem, O.A. A Multi-Model Proposal for Classification and Detection of DDoS Attacks on SCADA Systems. *Appl. Sci.* **2023**, *13*, 5993. https://doi.org/10.3390/app13105993

Academic Editor: Luis Javier Garcia Villalba

Received: 31 March 2023
Revised: 7 May 2023
Accepted: 11 May 2023
Published: 13 May 2023

1. Introduction

Facilities that produce, store, and transmit natural resources, such as water, oil, and natural gas, or energy sources, such as hydroelectric, solar, and nuclear, constitute critical infrastructures. Space, satellite, air, sea, or train transportation systems are also in these groups. These systems spread and work over small or large areas. Some systems monitor, control, and, when necessary, intervene in processes and events in critical infrastructures from a central point. One of them is the Supervisory Control and Data Acquisition (SCADA) system. For example, municipalities use SCADA systems to monitor water levels, pipe pressure, and the temperature in tanks located in utility water distribution facilities.

The reports and research published every year in the field of cybersecurity suggest one should always be ready for attacks that may occur from the inside or the outside [1]. Ensuring the cybersecurity of SCADA systems in the cyber world is a crucial issue and

has become mandatory. Since cyberattacks against SCADA systems are dangerous for critical infrastructure systems, these attacks should be investigated [2]. According to a special report by the National Institute of Standards and Technology, cyberattacks to control systems can disrupt the reliable operation of industrial processes. Therefore, providing cybersecurity is imperative [3]. Defence Research and Development Canada published a report aimed to increase the cyber resilience of Canada's critical infrastructure. According to the report, changes to the standard network configurations of SCADA networks can greatly improve the protection of control system fields [4]. In the study conducted by the U.S. Department of Energy Office of Electricity Delivery and Energy Reliability, security vulnerabilities found to be common in control systems such as SCADA were discussed and grouped by severity. In addition, security recommendations were provided for asset owners and system vendors [5]. Due to the architectural structure of SCADA systems, their integration with advanced technology has not been fully solved. On the other hand, internet usage, access to external networks, and remote control are increasing worldwide. These developments enhance the functionality of traditional SCADA systems, but they also bring many security vulnerabilities.

Critical infrastructures are designed to enable citizens to maintain their lives in better conditions. Problems experienced in the functioning of these structures may affect not only the relevant area but also the whole country. For example, the failure of electricity generation, storage, and transmission facilities can cause massive chaos in a country and directly affect other electrical systems. Countries experiencing power outages have realized how crucial such blackouts are. Attacks on critical infrastructures can destructively impact the economy, security, or health. The consequences of cyberattacks against SCADA systems may be far beyond estimates. As a result, necessary measures should be taken for the cybersecurity of SCADA systems. Security system developments, such as attack detection and prevention, will considerably contribute to the continuity of a country's critical infrastructures.

Models that include machine learning, deep learning, or artificial intelligence algorithms used in attack detection studies may also serve in SCADA systems. Studies to determine the "attacks" and "attack types" can contribute to the cybersecurity of SCADA systems. There are different types of cyberattacks, and distributed denial-of-service (DDoS) attacks are more common than other attacks. In particular, handling DDoS attacks for SCADA systems is essential in cybersecurity. Since the attack detection models in the algorithms have different structures, the analyses also give different results. For example, an attack detection model providing high performance on one dataset can deliver poor performance on another, or different models on a dataset may not yield the same highly successful results. For these reasons, developing an attack detection model that provides high performance for a particular dataset is essential.

The current study aimed to detect DDoS attacks that may occur against a SCADA system used in critical infrastructures and to determine the type of DDoS attack. For this purpose, a testbed was prepared that enables the processing of cyber and physical processes. Various DDoS attacks and tests were implemented on the testbed to damage the processes of the SCADA system and measure the system's reaction against attacks. Attack detection is essential to ensure the cybersecurity of the system. For this purpose, the network traffics in the baseline situation without any attack and the situations in which DDoS attacks were applied and recorded. Deep learning and machine learning algorithms were used to analyze the recorded network traffic packets and to determine whether there is an attack or not. In addition to intrusion detection, these algorithms have also been studied to determine the type of attack. The deep learning-based convolution neural network (CNN) model, long short-term memory (LSTM) model, and hybrid model using LSTM-CNN algorithms together were evaluated. Machine learning-based 13 algorithms such as K-Nearest Neighbors (KNN), LogitBoost, Naive Bayes, PART, decision tree, and random forest were used. High success rates were obtained with deep learning-based LSTM-CNN hybrid model and machine learning-based decision tree model. It is aimed to

provide different perspectives for ensuring the cybersecurity of SCADA systems and to prepare suitable models for determining the type of attack.

The main contributions of the present study are as follows:

- A testbed environment containing a SCADA system was prepared and different components, software and hardware were used from the studies in the literature.
- Various DDoS attacks (five different) and the baseline situation were evaluated together to add diversity to the literature.
- A new dataset was prepared to contribute to the literature by including various DDoS attacks and a baseline situation, enabling detection and identification of attack types.
- CNN and LSTM algorithms were used as separate models for attack detection and attack type determination. In addition, LSTM and CNN algorithms were evaluated together and used as a hybrid model. In the studies in the literature we examined, there are no such separate and hybrid uses in this way. By using a hybrid model, a higher success rate was obtained than using separate models. In addition to deep learning-based models, machine learning-based models were also prepared and evaluated in the study. Analyses were performed with 13 different machine learning algorithms and the highest success rate was obtained with the decision tree model.
- A commonly used dataset in the literature was selected and tested to evaluate the adequacy of the hybrid model. According to the results obtained, a high accuracy rate was achieved.

This study comprises six chapters. The first part provides an overview of SCADA systems, shows the security vulnerabilities, and explains the importance of ensuring the cybersecurity of SCADA systems. The second part examines the studies that detect attacks against SCADA systems using their own datasets, ready-made datasets, or their own testbeds. The third chapter discusses SCADA systems and cyberattacks against these systems. The fourth section covers the prepared testbed environment, DDoS attacks against this environment, the obtained dataset, the success metrics, and the proposed models. The fifth section presents the analyses for DDoS attack detection for SCADA system, experimental results of the proposed models, and the results of other studies in the literature. The study results and recommendations for future studies are summarized in the sixth section.

2. SCADA Systems and Cybersecurity

This section gave information about what SCADA systems are, what components they consist of, the cybersecurity of these systems, and possible attacks.

2.1. Scada System

SCADA systems perform control and monitoring tasks in critical infrastructure or facilities. Critical infrastructures, such as power generation plants, wind energy turbines, and natural gas distribution facilities are vital structures that produce and (or) transmit natural gas, oil, water, and similar resources to another place. To give more examples, many systems such as municipal water distribution facilities, airlines, and ship systems are also critical infrastructure systems and have a significant place nationally and internationally. SCADA systems are also used in production facilities, factories, or public institutions apart from these infrastructures.

SCADA systems consist of a master terminal unit (MTU), remote terminal units (RTUs), and a communication network. The MTU controls the processes in the system using a human–machine interface (HMI). There is data exchange and command transmission between RTUs and MTU. RTUs transmit the data collected from the field sensors to the MTU, and RTUs carry out the commands from MTU. Modbus, DNP3, and Profibus communication protocols—specific to SCADA systems—are used for communication between basic units. The sensors and actuators on the RTUs abide by the commands. Elements such as pumps and relays serve as actuators. The HMI also demonstrates the data obtained from the sensors [6,7].

2.2. Cybersecurity and Attacks in Scada Systems

Most structures where SCADA systems are used have not direct connections to the internet and work independently from external networks. Developing technologies expand the area of internet usage, and this situation also affects SCADA systems. Innovations such as the co-usage of different technologies and remote accessing the system via the internet create new cybersecurity problems for SCADA systems. SCADA systems, which cannot keep up with the developing technology, have many architecture-related security problems. For example, the frequently used Modbus protocol has many vulnerabilities that can be attacked, such as by a man-in-the-middle, command injection, and denial-of-service (DoS) [8–10]. SCADA systems in different sectors become attractive targets for malicious people who are aware of these situations and work in the national or international arena. Figure 1 shows the sectors where SCADA systems serve. Each can contain various threats that malicious applications can attack.

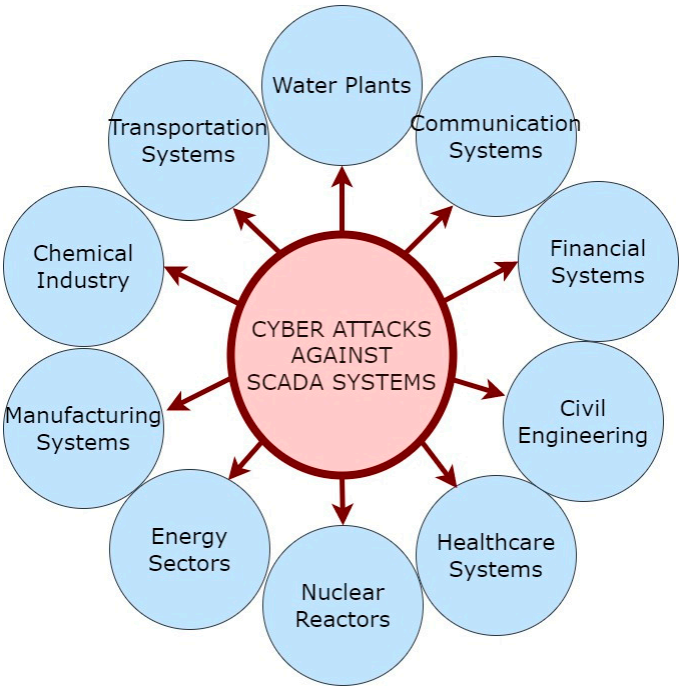

Figure 1. Sectors where SCADA systems are used.

Today, many attack scenarios may occur in SCADA systems, such as emerging new vulnerabilities, existing old security gaps, vulnerability exploitation, damaging systems, or rendering the system inoperable. Possible scenarios may cover malicious remote control of the system and power cut threats. For example, cyberattacks can occur by targeting electricity generation or distribution facilities. As a result of these attacks, there may be power cuts; cities may suddenly go dark. A nuclear power plant's centrifuges were remotely disrupted through the Stuxnet, which is one of the most dangerous attacks. In this attack, while the system was physically damaged, the field operators noticed the problem much later [11]. Another example of an attack is the remote poisoning of the Florida City Water Supply. The attackers seized the water facility and tried to increase the sodium hydroxide level in the city water. Once the authorities realized the situation, they quickly intervened and prevented the attack [12]. As can be understood from these examples, cyberattacks can also affect some or all of the SCADA systems. In addition, the experienced problems may adversely trigger other systems associated with SCADA systems. Today, actions to disrupt public peace, complicate their daily life, or harm their health have become possible using the vulnerabilities in SCADA systems. For these reasons, cybersecurity in SCADA systems is necessary today.

SCADA systems are vulnerable to numerous attacks due to their tasks, traditional architectural structure, and built-in communication technologies. Specially developed attack techniques make SCADA systems targets for aggressive attempts, and the security risk of these systems is increasing day by day. Various attacks are made against SCADA systems, such as man-in-the-middle, data injection, command injection, DoS, and DDoS [10,13,14]. Among these, DDoS attacks are common and dangerous attacks that can affect any SCADA system. These attacks aim to disrupt control and process operations and render the system out of use [15,16]. DDoS attacks against SCADA systems used in critical infrastructures may cause devastating harm to these infrastructures.

3. Literature Studies on SCADA Security

In the literature, studies carried out on the detection of DDoS attacks against SCADA systems are considerably popular. These studies have frequently used machine learning- and deep learning-based methods for attack detection. Some of these works are summarized below.

Marcio Andrey Teixeira et al. performed a study to detect cyberattacks on SCADA systems. The authors created a dataset using a test environment. They employed random forest, decision tree, logistic regression, Naive Bayes, and KNN algorithms in their study for attack detection [17].

Thomas Morris and colleagues worked on potential cyberattacks at Mississippi State University's SCADA Security Lab and investigated the security vulnerabilities of the most widely used communication protocols in SCADA systems. They aimed to detect attacks and minimize their effects with the security mechanisms developed with neural network methods [18].

Nader et al. carried out a study on the security of industrial control systems and critical infrastructures. They emphasized that traditional attack detection systems could not detect attacks newly developed and unregistered in databases. They used data from a water distribution system in France in the study and proposed machine learning algorithms for attack detection [19].

Focusing on the developments in information and communication technologies, Y. Yang et al. have emphasized that the complexity and security vulnerabilities in SCADA procedures are gradually increasing. They stated that new security measures were necessary for new-generation SCADA designs integrated into the internet and different systems. Therefore, they proposed an attack detection system with a behavior-based and multilayer framework [20].

Almalawi et al. proposed two approaches to detect attacks against SCADA systems. The first was to determine whether the data in the system were consistent or inconsistent. The second approach was to obtain proximity detection rules from specified situations. They stated that the KNN-based attack detection system showed significant accuracy [21].

Meir Kalech proposed techniques based on temporal pattern recognition for cyberattack detections in SCADA systems. The study proposed two algorithms based on Hidden Markov models (HMM) and artificial neural network-based self-organizing maps (ANN-based SOM). According to the results obtained, they stated that it was easier to detect cyberattacks [22].

Jun Gao et al. discussed temporally uncorrelated and correlated attacks against SCADA systems. They detected attacks using the feedforward neural network (FNN) and LSTM algorithms based on deep learning. The FNN-LSTM model, on the other hand, succeeded in detecting both types of cyberattacks, regardless of their temporal correlations [23].

While intrusions into SCADA systems will continue, defense mechanisms against different attack vectors remain insufficient. Therefore, Maglaras et al. conducted a study to ensure the cybersecurity of SCADA systems. Accordingly, they proposed an integrated attack detection mechanism against cyberattacks that captures network traffic, divides traffic by source, and creates a set of one-class support vector machine (OCSVM) models [24].

Gao et al. proposed two models aimed at detecting attacks against SCADA systems. These models have many-to-many (MTM) and many-to-one (MTO) architectures. The models used the LSTM algorithm. Both detection systems performed well in detecting temporally uncorrelated attacks [25].

There are many studies aimed at detecting unauthorized access to SCADA systems. Shitharth and Winston developed an intrusion detection system that classifies attacks based on optimization. They proposed intrusion weighted particle-based cuckoo search optimization (IWP-CSO) and hierarchical neuron architecture-based neural network (HNA-NN) techniques [26,27].

This study focused on DDoS attacks in SCADA architecture and presented models that work efficiently to detect attacks. In the literature, studies that detect attacks on SCADA systems have been examined and it has been seen that machine learning algorithms such as random forest, decision tree, logistic regression, Naive Bayes, KNN, and SVM are used more frequently than other algorithms for detection. In addition to these, there are models in which neural networks and deep learning algorithms such as LSTM are used. In this study, deep learning-based models (CNN, LSTM, LSTM-CNN hybrid) and machine learning-based models (13 models such as KNN, LogitBoost, Naive Bayes and decision tree) were proposed. The attack detection accuracy rates of the examined studies and this study are placed in the table in the Section 5.2. Thus, a general review and comparison is provided for the studies.

4. Materials and Method

This section elaborated on the prepared testbed, fictionalized the cyberattacks using scenarios, and gave information about the dataset's features obtained from the testbed. This section also covered the metrics to analyze the dataset as well as their explanations. Information was given about the proposed models and their architectural structures. The topics in this section are summarized in Figure 2.

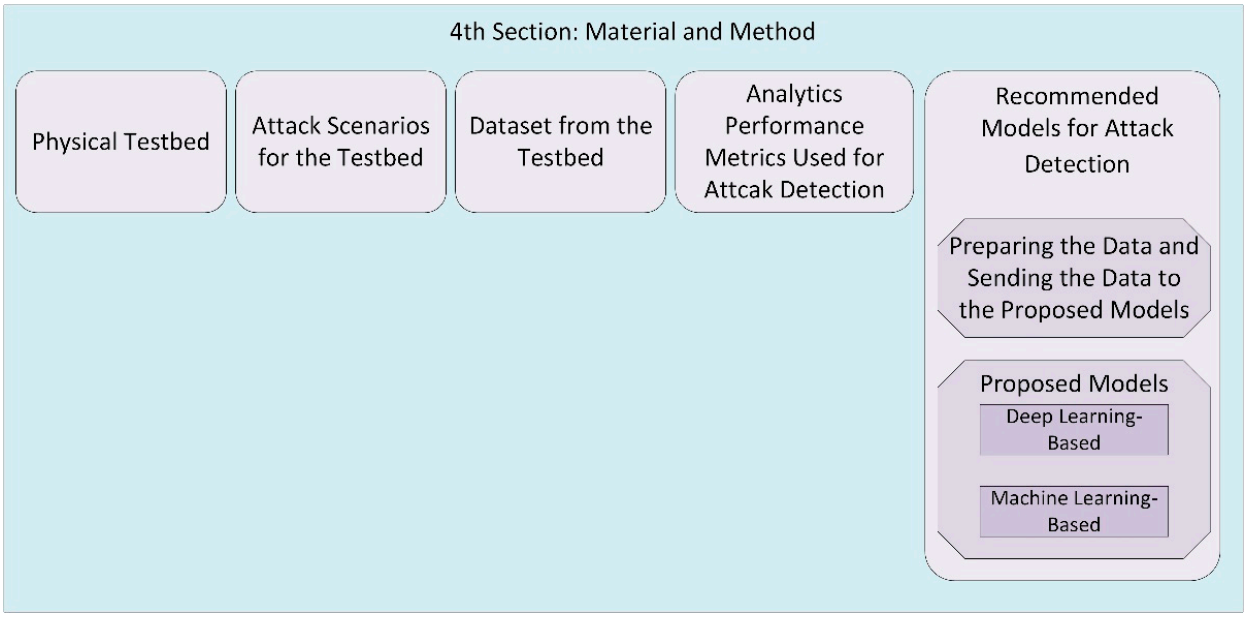

Figure 2. Organizational chart of the Materials and Method section.

4.1. Physical Testbed

The test environment aimed to simulate the industrial control systems of a plant as approximately as possible without completely copying them [28]. In addition, it aimed to contribute to the performance of national and international industrial control system stan-

dards and directives. The preparation and use of a testbed provides a suitable environment for performing real cyberattacks and even observing the results of the attack.

In order to contribute to cybersecurity research, a testbed environment including a SCADA system was prepared in the study. In this environment there are storage tanks, specific processes are operated, and Modbus TCP/IP communication is used. A SCADA system is usually realized by integrating Modbus communication protocol [29]. A simplified version of a real water plant was shown in this testbed. The SCADA system controls and monitors the water circulation processes and the status of the storage tanks. This section explained the configuration and architectural structure of the prepared SCADA system test environment. The equipment used in the test environment was selected from the components frequently used in real SCADA systems. The architectural structure of the test environment is shown in Figure 3.

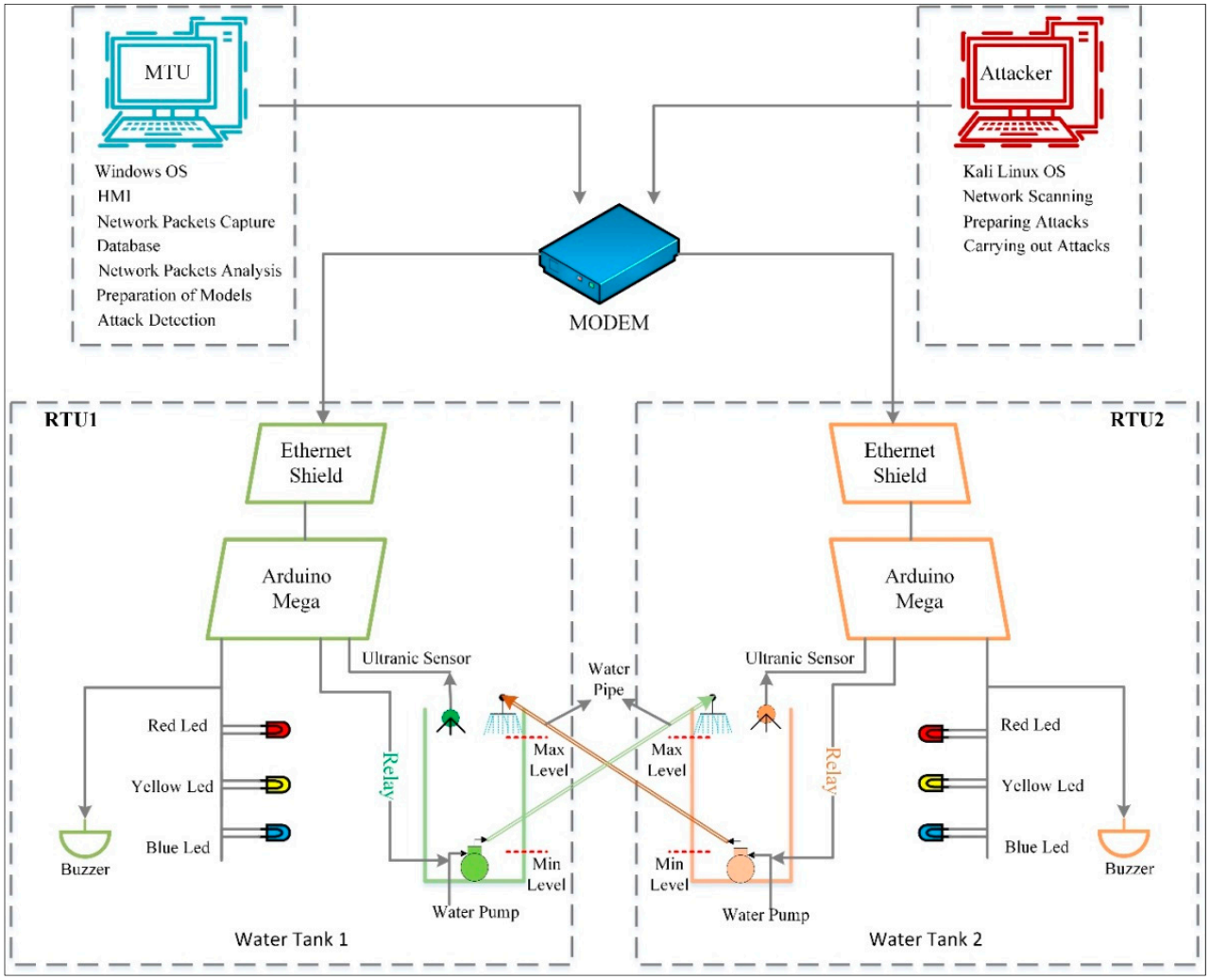

Figure 3. The architectural structure of the testbed.

As shown in Figure 3, there are two water circulation and storage tanks in RTUs. There were sensors and actuators connected to RTUs. The sensors monitor the water levels in the tanks, and the water pumps operate according to the levels. In order to prevent problems such as the overflowing of the tanks and running out of water in the tanks, the water level is continuously controlled. In addition, an alarm was generated according to the state of the water level and, thus, attracting the attention of the operator who is interested in the system. LEDs and buzzers are designed for alarm events. Modbus TCP/IP wired and wireless communication protocols were used for communication in the environment. The

data received from the RTUs were transmitted to the MTU and the processes were followed through the HMI simulators on the MTU. Incoming data were checked and stored, and new commands were sent to RTUs.

Attackers scanned the network and attacked the appropriate RTU. Whether there is an attack or not is checked on MTU. When 5 different DDoS attacks were applied to the RTU, network traffic packets were listened to and recorded separately for each attack. In addition, the same listening and recording operations were performed for baseline operation without attack. Google Colab, an environment offered by Google Research, enables Python coding for machine learning, data analysis, and training. The data were preprocessed in this environment to make the packets suitable for analysis. Pandas' libraries were added to this environment and different models were generated for attack detection using deep learning and machine learning algorithms.

4.2. Attack Scenarios for the Testbed

This section elaborates on the baseline situation of the testbed and the attacks against the testbed. DDoS attacks, one of the most common attacks on SCADA systems, were discussed and attacks against an RTU selected by the attacker were performed. Different types of DDoS attack scenarios were implemented and aimed to affect the operation of the system. These scenarios were:

1. Baseline (normal or no-attack) situation;
2. TCP flooding attack scenario;
3. UDP flooding attack scenario;
4. SYN flooding attack scenario;
5. Spoofing IP flooding attack scenario;
6. ICMP flooding attack scenario.

In the baseline situation scenario (when the SCADA system was not under attack), the obtained network traffic was listened to and recorded. In this scenario, water circulated continuously between the water tanks and the necessary operations were performed automatically according to the change in the water level. The pinging method was used to establish communication between RTU and MTU.

Specific coding was made by the attacker for each attack type and 5 different DDoS attacks were performed against the target RTU. Each of TCP, UDP, SYN, Spoofing IP and ICMP flooding attacks were carried out at different times and separately. Each of these attack scenarios were executed for approximately 2 min. During the attacks, the target RTU system processes were interrupted for a short period of time. Processes such as water recirculation and alarm generation were disrupted. These adverse conditions also affected the other RTU system and the operation of the entire testbed system was interrupted for short periods of time. Abnormal situations such as incorrect measurement of the tank water level or buzzer alarming at the wrong time were observed. When the execution of the attack scenarios ended, the system operation slowly recovered and, after a while, the system returned to its former state. If the time taken to restore the system operation is too long to be tolerated, irreversible major problems may occur for SCADA systems. For this reason, it is important to attack SCADA systems and monitor and analyze the attack responses. In this study, this issue is emphasized.

4.3. Dataset from the Testbed

This section provides information about the total dataset obtained as a result of the scenarios performed separately on the testbed. Network traffic packets of each scenario were collected with Wireshark network listening and analysis tool. Then, the packets of these 6 scenarios were collected in a single file and the total dataset was created. The features frequently used in the literature and specific to the Modbus TCP/IP protocol were determined for this dataset [23,25]. Table 1 shows the features used in this research.

Table 1. Features used in the dataset and their descriptions.

No	Features	Descriptions
1	No	Data number
2	Time	Time
3	SourceIP	Source Internet Protocol
4	DestinationIP	Destination Internet Protocol
5	SourcePort	Source port
6	DestinationPort	Destination port
7	Protocol	Protocol
8	Length	Data packet length
9	Info	Information about packet
10	Modbus_ByteCount	Modbus protocol data area (in bytes) size
11	Modbus_ResponseTime	Modbus protocol response time
12	Modbus_ReqFrame	Modbus protocol message format
13	DeltaTime	Duration between the start and end of an operation
14	ModbusEventCount	Number of Modbus device transactions
15	TimeSince_FirstFrameInThisTCPStream	Time elapsed since the first frame in this TCP stream
16	TimeSince_PreviousFrameInThisTCPStream	Time elapsed since the previous frame in this TCP stream
17	TimeDeltaFromPrevious_CapturedFrame	Time difference from the previous captured frame
18	TimeDeltaFromPrevious_DisplayedFrame	Time difference from the previous displayed frame
19	TimeSince_ReferenceOrFirstFrame	Time elapsed since the reference or first frame
20	FrameLength_OnTheWire	Frame length on the wire
21	FrameLength_StoredIntoTheCaptureFile	Frame length stored into the capture file
22	TimeToLive	Time to live
23	TotalLength	Total length
24	FrameLengthStoredIntoTheCaptureFile	Frame length stored into the capture file
25	ModbusTCPLength	Modbus TCP packet length
26	ModbusByteCount	Modbus packet byte count
27	ModbusTimeFromRequest	Modbus packet time from request
28	TCPHeaderLength	TCP header length
29	ModbusRegNum	Modbus register number
30	Register Value (UINT16)	Modbus register value
31	Class	Classification column

A new and comprehensive dataset consisting of 30 attributes, 1 deterministic class, and a total of 22.768 samples was obtained. While preparing the dataset, the attacks were observed on the SCADA system and abnormal situations were noticed clearly by the operator. It is detected whether there is a DDoS attack and if there is an attack, which of the 5 different types is determined. This dataset is suitable for training and testing deep learning and machine learning models. Due to these properties, a new perspective and contribution to the literature is presented.

4.4. The Performance Analysis Metrics in Attack Detection

Performance metrics serve for the evaluation and comparison of the deep learning, and the machine learning algorithms for the model. While working on a problem, using

these metrics makes it easier to propose more solutions and apply the proposed methods. To determine the most effective method in problem solving, the performance information of each method is obtained one by one. Then the method producing the highest success rate is selected. Table 2 shows the confusion matrix containing the values for performance metrics.

Table 2. Confusion Matrix.

		Actual Values	
		Positive	**Negative**
Predictive Values	**Positive**	TP	FP
	Negative	FN	TN

The values in the confusion matrix show the actual values and the estimation values [30]. The fact that the value with a positive label in reality also has a positive label in the prediction part makes it a true positive (*TP*). The fact that the value with a negative label is positively labeled in the prediction part makes it a false positive (*FP*). The fact that the value with a positive label is negatively labeled in the prediction portion makes it a false negative (*FN*). The fact that the value with a negative label also has a negative label in the prediction part makes it a true negative (*TN*). The success metrics calculated with the values on the confusion matrix are below.

Accuracy is the ratio of the correctly predicted values to the total values. Equation (1) shows this situation:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

Precision is the ratio of the correctly predicted positive values to the predicted values with a positive label. Equation (2) shows this ratio:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

The recall is the ratio of correctly predicted positive values to the values with a positive label. Equation (3) shows this ratio [31]:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

F-1 Score—ranging from 0 to 1—is the harmonic mean of precision and recall values. Equation (4) calculates this average:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

The current study used accuracy, precision, recall, and f1-score among traditional performance metrics. While comparing the literature studies, this research preferred the frequently used "accuracy success metric".

4.5. Recommended Models for Attack Detection

In this section, attacks against the testbed containing the SCADA system were detected. For this purpose, a deep learning-based model and a machine learning-based model were applied to the previously prepared dataset. The analysis results obtained were compared with each other according to particular metrics.

In order to achieve successful results in proposed models, data were preprocessed, and experiments were performed. Innovative and different approaches were proposed for the cybersecurity of a physical testbed containing a SCADA system.

4.5.1. Preparing the Data and Transmitting Them to the Proposed Models

This section concerns turning the dataset into an analyzable state and designing the appropriate models, which Figure 4 summarizes. Several data pre-processes were determined to make the dataset analyzable. The primary operations were deleting attributes and instances deemed unnecessary or containing too many null values. The next step was completing the attributes containing missing data using the mean method. Another process is to convert data types to the same type using categorization processes. In the end, a dataset with 25 features was obtained.

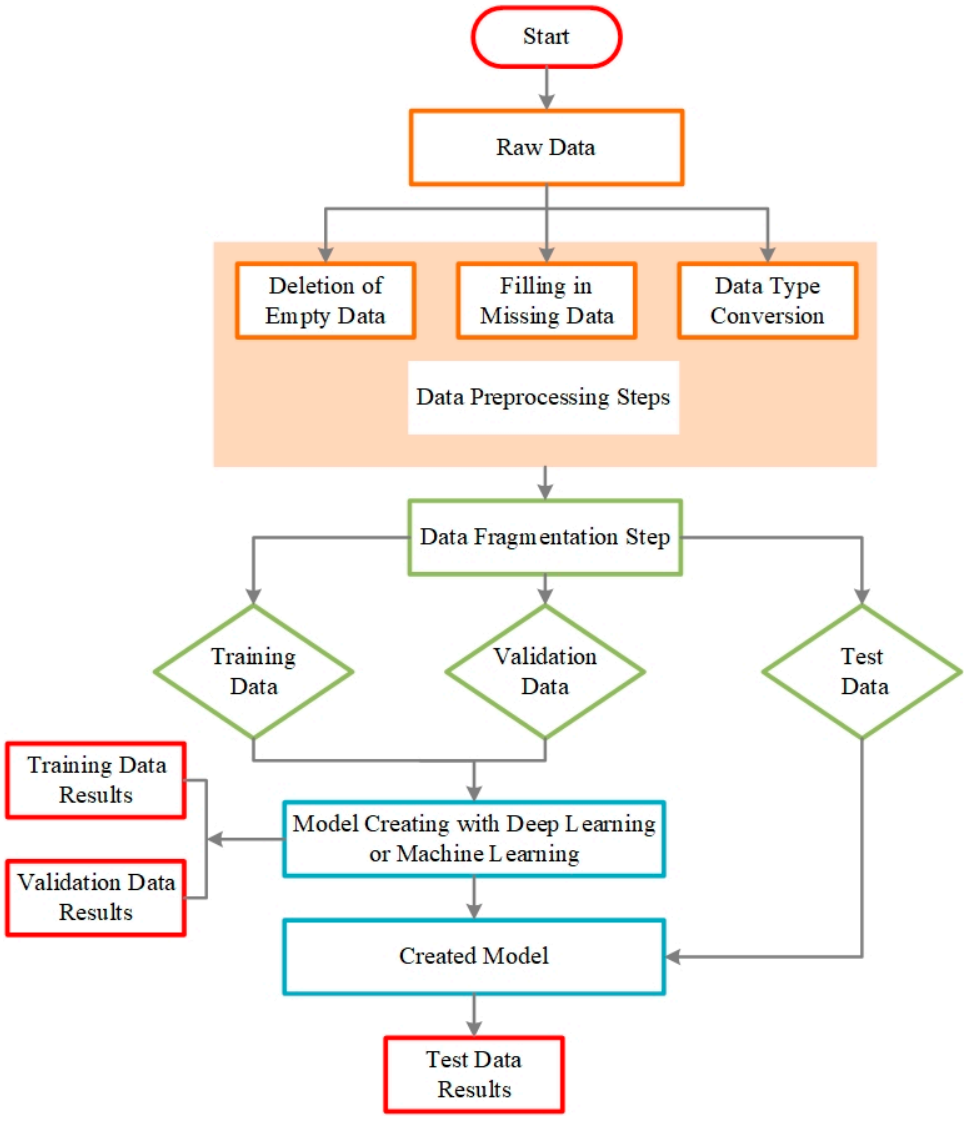

Figure 4. Processing the data and delivering them to the proposed models.

After preprocessing, the dataset was divided into parts for training, validation, and testing in the data fragmentation stage. The split ratios here were kept constant in the models used. The obtained training and validation data were combined and sent to the proposed models. Tests were carried out on the model using the test data, and analysis results were obtained for training, validation, and test data. According to the results, the attack-detection success of the proposed model was evaluated. These stages were essential for obtaining the most suitable model which achieved the highest success rate in detecting the attack.

4.5.2. Recommended Models

LSTM and CNN, which are important deep learning algorithms, were used alone and in combination with different algorithms in the literature. In this study, CNN and LSTM algorithms were evaluated and tested separately in order to contribute to the literature. Then, a hybrid model was created by considering these two algorithms together and tests were performed. Tests with different properties were applied to the LSTM model and CNN model. The parameter values in the LSTM-CNN hybrid model architecture were changed and different test procedures were performed. These models were analyzed separately and their attack detection success rates were discussed.

In addition to these, machine learning algorithms that are frequently used in the literature were determined. By using these algorithms, suitable models for attack detection were obtained. All prepared models were compared according to the determined success metrics and the results are presented in the Section 5.1. Information about the models used, their architectural structures, and parameter values were given in this section.

A 70% randomly selected dataset—that is, 15,937 rows of data—was used in the models' training. The remaining 30% was split into two to evaluate the testing and validation of the proposed model. Accordingly, 3415 rows were used for data validation and 3416 rows (including the class column) for testing. There were 22,768 rows of data (samples) in total.

Deep Learning-Based Models

The deep learning-based models used in the study are explained in this section. Analyses were made on the LSTM models, the CNN models, and the hybrid models in which LSTM-CNN were used together.

Since categorical data were included for all three proposed models, categorical_crossentropy was chosen as the loss function. Adaptive moment estimation (ADAM) was used as the optimization algorithm because it works efficiently on datasets containing many parameters [32]. In order to ensure stability, the rectified linear units' (ReLU) activation function was preferred. ReLU has a simple computational form and determines the output by evaluating the input [33]. The batch size was left by default. The softmax function was used to finish the classification.

LSTM-Based Models

In this model, analyses were performed on the LSTM algorithm. The LSTM algorithm is an iterative neural network and has been used frequently recently. Due to its structure, it is very effective in catching long-term addictions. It can store information for a long time with its special memory cell architecture. LSTM consists of repetitive sequential blocks known as memory blocks.

In this algorithm, there are input, output, and forget gates that enter and exit between cells and regulate the flow of information. For the iteration process, the input is generated, the predicted output value is obtained according to the current situation, and the next output vector is generated. Figure 5 shows the architecture of the LSTM-based deep learning models.

The first proposed model was based on deep learning using the LSTM algorithm. LSTM networks contain a sequential input layer. In the proposed LSTM network architecture, the LSTM layer was placed after the input layer. Next came a smoothing layer and, finally, the fully connected classification and output layers. In the study, two LSTM models with 200 epoch and 300 epoch parameters were prepared (LSTM1a and LSTM1b). Other parameters selected for the models were mentioned at the beginning of the chapter.

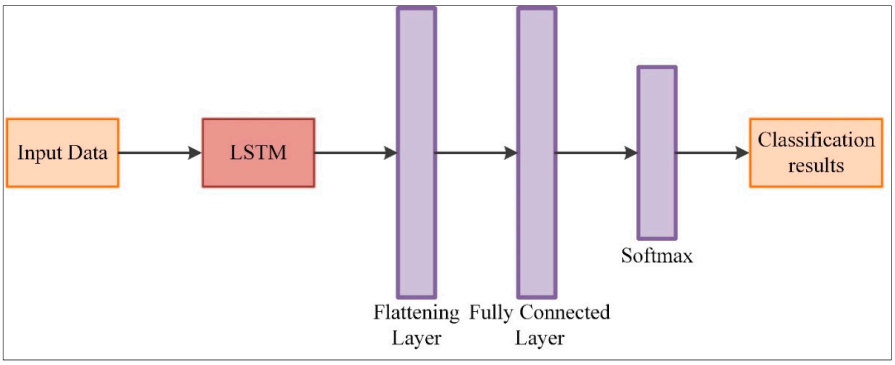

Figure 5. The architecture of the LSTM-based deep learning models.

CNN-Based Models

The CNN algorithm was studied in this model. The CNN algorithm is a variant of feed forward neural network. The architecture of the CNN algorithm is similar to the multilayer perceptron and consists of three layers. These are the convolution layer, pooling layer, and fully connected layer [34]. Multiple filters are included in this algorithm to extract or retrieve hidden features from the dataset. Figure 6 shows the architecture of the CNN-based deep learning models.

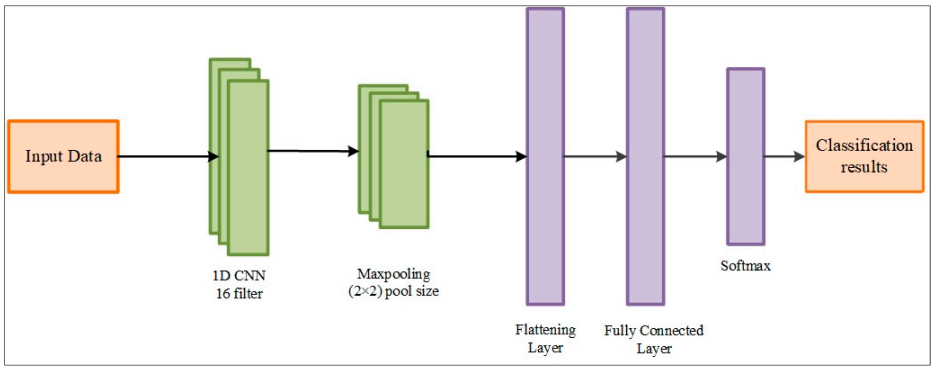

Figure 6. The architecture of the CNN-based deep learning models.

Deep learning was performed with the CNN1a model using 200 epochs and the CNN1b model using 300 epochs. Both models employed a 1-D CNN layer and pooling layer followed by normalization and flattening. Finally, connected, classification, and output layers were used. Other parameters selected for these models were explained at the beginning of the chapter.

Hybrid-Based Models

In the other model proposed in the study, LSTM and CNN algorithms were used as a hybrid and analyses were carried out. Three different models (HYBRID1, HYBRID2, and HYBRID3) were prepared using hybrid deep learning. The architecture of the first model (HYBRID1) is shown in Figure 7 as the others were prepared with reference to the first model.

In this hybrid model (HYBRID1), deep learning was performed using LSTM and CNN algorithms. Pooling layers and 1-D CNN layers were used. Normalization processes were done and, after the last pooling layer, the LSTM layer was placed in the model. Smoothing, fully connected, classification, and output layers were used. The HYBRID1a model with 200 epochs and the HYBRID1b model with 300 epochs were obtained.

In the second hybrid model (the HYBRID2), unlike the HYBRID1, normalization and activation processes were applied twice. Then, the HYBRID2a model was obtained by applying 200 epochs to the model and the HYBRID2b model was obtained by applying 300 epochs to the model.

The kernel size in the 1-dimensional CNN layers in the HYBRID1 model was increased and the number of filtering operations was reduced. In this way, the HYBRID3 model was obtained. The HYBRID3a model for 200 epochs and the HYBRID3b model for 300 epochs were prepared.

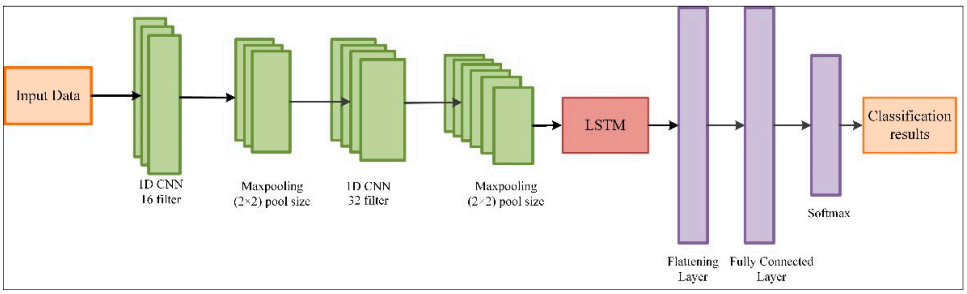

Figure 7. The architecture of the hybrid deep learning model (HYBRID1).

Machine Learning Based Models

When the literature was examined, it was seen that machine learning methods are also used in the detection of attacks on SCADA systems. Algorithms such as random forest, decision tree, logistic regression, Naive Bayes, and KNN were frequently used in the literature. In this study, in addition to deep learning algorithms, machine learning algorithms were also evaluated. Machine learning models were prepared for the detection of DDoS attacks and DDoS attack types for the testbed environment using the SCADA system. The results obtained were given in Table 3.

Table 3. Performance values of the proposed models.

Models		Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Deep Learning Based Models					
LSTM	LSTM1a	84.60	86.03	84.60	83.73
	LSTM1b	84.28	84.63	84.28	83.63
CNN	CNN1a	93.53	94.01	93.53	93.57
	CNN1b	94.26	94.79	94.26	94.35
LSTM-CNN HYBRID	HYBRID1a	94.09	94.23	94.09	94.12
	HYBRID1b	93.97	93.99	93.97	93.97
	HYBRID2a	93.91	94.05	93.91	93.93
	HYBRID2b	91.92	92.33	91.92	91.93
	HYBRID3a	92.77	92.99	92.77	92.82
	HYBRID3b	**94.73**	**94.90**	**94.73**	**94.74**
Machine Learning Based Models					
Lazy	KStar	79.93	81.93	79.95	79.03
	LWL	66.00	59.62	66.02	58.53
	KNN	86.15	86.08	86.15	86.11
Meta	LogitBoost	83.91	88.33	83.93	83.13
	AdaBoost	42.96	-	43.01	-
Bayes	NaiveBayes	84.03	85.43	84.00	83.54
	BayesNet	85.24	86.44	85.20	84.82
Rules	ZeroR	22.55	-	22.51	-
	PART	79.24	91.32	79.23	77.14
	DecisionTable	59.39	-	59.40	-
Trees	DecisionTree	**98.77**	**98.77**	**98.77**	**98.77**
	RandomForest	95.84	97.21	95.84	96.51
	RandomTree	83.07	85.71	83.14	82.44

KStar, locally weighted learning (LWL) and KNN algorithms from lazy learning methods were preferred. LogitBoost and AdaBoost algorithms from Meta Learning Methods and Naive Bayes and Bayes Net algorithms from Bayesian methods were used. ZeroR, PART, and decision Table algorithms based on rules and the decision tree, random forest, and random tree algorithms based on trees were analyzed.

5. Experimental Results

This section discussed the analysis results of the proposed deep learning-based, and machine learning-based models. In addition, the discussion section covered the comparison between previous studies and the current study for attack detection success. The analysis results of the proposed hybrid model on a different dataset were also placed in the table in the Discussion section.

5.1. Results

In the study, the steps mentioned in Title 4 were carried out on the dataset. Table 4 shows statistical information about network traffic captured while applying attack scenarios and the baseline situation. Captured packets in network traffic represent samples in datasets.

Table 4. Statistical information about network packets of attack scenarios.

	Attack Scenarios Values					
Measurement	**Normal**	**TCP Flooding**	**UDP Flooding**	**SYN Flooding**	**Spoofing IP Flooding**	**ICMP Flooding**
Total number of packets	3391	5253	3118	3238	3217	4551
Average packet size (bytes)	109	60	89	60	143	60
Total size of packet (bytes)	370,724	315,180	277,679	194,280	461,615	273,084
Duration of capture (ms)	530	294	253	163	286	271

As shown in Table 4, while there were 3391 packages in the normal situation scenario, there were 19,377 packages in the attack scenarios. The distribution of the number of packages was in a balanced state in all scenarios. The average sizes of packets (in bytes) were the same for TCP, SYN, and ICMP flooding attack scenarios. The spoofing IP flooding attack scenario had the maximum value. When the attack scenarios were analyzed separately, the total packet sizes (in bytes) took different values. In attack scenarios, when the packet capture times were examined, the most listening was done for the baseline situation. The least time was spent on the SYN Flooding attack scenario.

Analyses were made to reveal the attacks and DDoS attack types on the system. Suggestions were made for the attack detection system. Table 3 presents the analysis of the proposed models results.

When the performance results were examined, it was seen that the HYBRID3b model was more successful in analyzing and classifying DDoS attack data among deep learning algorithms. Among the machine learning algorithms, the highest success rate was obtained with the decision tree model. Considering the accuracy, precision, recall, and f1-score success metrics, these two models were found to be the most suitable models for attack detection. LSTM models from deep learning algorithms and the ZeroR model from machine learning algorithms performed the attack detection with the lowest success rate. The confusion matrix values obtained with the HYBRID3b model were placed in Table 5 and are shown below.

Table 5. Confusion matrix values of the proposed HYBRID3b model.

		Predicted Class							
		Normal (%)	**TCP Flooding (%)**	**UDP Flooding (%)**	**SYN Flooding (%)**	**Spoofing IP Flooding (%)**	**ICMP Flooding (%)**	**TP Rate (%)**	**FN Rate (%)**
Actual Class	**Baseline Situation**	88.27	0.00	8.65	0.00	3.08	0.00	88.30	11.70
	TCP Flooding	0.00	100	0.00	0.00	0.00	0.00	100	0.00
	UDP Flooding	3.67	0.00	92.01	0.00	4.32	0.00	92.10	7.90
	SYN Flooding	0.00	0.00	0.00	100	0.00	0.00	100	0.00
	Spoofing IP Flooding	6.34	0.00	9.90	0.00	83.76	0.00	83.80	16.20
	ICMP Flooding	0.00	0.00	0.00	0.00	0.00	100	100	0.00

The confusion matrix values in Table 5 were evaluated according to the accuracy metric frequently used in the literature [35]. Accordingly, among the attack types, TCP, SYN, and ICMP Flooding attacks were correctly detected with 100%. The worst detection performance was achieved in the spoofing IP flooding attack with a rate of 84%. For this attack, 423 of 505 samples were correctly detected. In the baseline situation, 459 of 520 samples were determined as not attacked and a high detection rate of 88% was obtained. The UDP flooding attack detection also had a rate close to the non-attack detection rate. The confusion matrix values obtained with the decision tree model were placed in Table 6 and shown below.

Table 6. Confusion matrix values of the proposed the decision tree model.

		Predicted Class							
		Normal (%)	**TCP Flooding (%)**	**UDP Flooding (%)**	**SYN Flooding (%)**	**Spoofing IP Flooding (%)**	**ICMP Flooding (%)**	**TP Rate (%)**	**FN Rate (%)**
Actual Class	**Baseline Situation**	98.08	0.00	0.58	0.00	1.34	0.00	98.10	1.90
	TCP Flooding	0.00	100	0.00	0.00	0.00	0.00	100	0.00
	UDP Flooding	0.86	0.00	95.25	0.00	3.89	0.00	95.30	4.70
	SYN Flooding	0.00	0.00	0.00	100	0.00	0.00	100	0.00
	Spoofing IP Flooding	0.59	0.00	2.18	0.00	97.23	0.00	97.30	2.70
	ICMP Flooding	0.00	0.00	0.00	0.00	0.00	100	100	0.00

The values in Table 6 were evaluated according to the accuracy metric. As in the HYBRID3b model, all TCP, SYN, and ICMP flooding attacks were correctly detected with 100% in the decision tree model. The UDP flooding attack was the worst-detected attack with 95%. For this attack, 441 of 463 samples were correctly detected. In the baseline situation, 510 of 520 samples were determined as non-attack and a high detection rate of 98% was obtained. Spoofing IP flooding attack detection also had a rate close to the non-attack detection rate.

5.2. Discussion

The analysis results of the two models with the highest success rates among the models proposed in the study (deep learning-based and machine learning-based) were compared with the analysis results of the studies in the literature. The results obtained are given in Table 7.

Table 7. Comparison of studies in the literature.

References	Datasets	Algorithms	Detection Rate (%)
[17]	Their own dataset	Random Forest	99.89
		Decision Tree	99.89
		Logistic Regression	99.59
		Naive Bayes	99.60
		KNN	72.29
[18]	Mississippi State University SCADA Laboratory	Neural Network	Average 83.00
[19]	Water distribution system real dataset	SVDD	Average 84.00
		Robust SVM	Average 76.00
		Slab SVM	Average 82.00
		Proposed Method	Average 91.00
[20]	Their own dataset	Hybrid SCADA-IDS	100
[21]	DUWWTP Dataset	KNN	92.86
[22]	CyberGym SCADA Lab dataset Ben-Gurion University of the Negev SCADA Lab Dataset	ANN-based SOM	Average 85.00
		HMM	Average 88
[23]	Their own dataset	FNN	Average 99.00
		LSTM	Average 99.00
		FNN-LSTM	Average 99.00
[24]	Their own dataset	OCSVM	96.30
[25]	Their own dataset	MTO-based LSTM	Average 99.00
		MTM-based LSTM	Average 98.00
[26]	ADFA-LD Dataset	IWP-CSO + SVM	91.50
		HNA-NN	83.20
		IWP-CSO + HNA-NN	93.10
		SVM	74.90
Our Study	Our Dataset	HYBRID3b Model	94.73
		Decision Tree Model	98.77
	Mississippi State University SCADA Laboratory	HYBRID3b Model	98.09

The study addresses the detection of DDoS attacks against the physical testbed using SCADA systems. For this, approaches based on deep learning and machine learning were used. The number of previous studies that detected attacks using ready-made datasets was very high. Fewer studies created a testbed for attack detection, prepared their own dataset, and performed analyses using the dataset. Both types of studies were equally included and reviewed.

Machine learning-based classifier methods such as KNN, Naive Bayes, and random forest were generally used in attack detection. There were also studies based on deep learning approaches such as LSTM and neural networks. As can be seen in Table 7, different algorithms were used for various datasets in the detection of attacks on SCADA systems. Each dataset had different characteristics and should be evaluated on its own.

In the studies examined in the literature, machine learning and deep learning approaches had achieved an average of over 90% success in attack detection on SCADA systems. As a result of the analyses performed in this study, two models based on deep learning and machine learning were proposed. With the hybrid model using LSTM and

CNN algorithms together, 95% success was achieved. A higher success rate of 99% was achieved with the decision tree-based model.

When we consider the existing research on the subject, we obtained promising models by creating an appropriate testbed, utilizing relevant technologies, and preparing dataset features. It is difficult to directly compare the performances of different models with the results obtained from studies using different datasets. Therefore, a different technique was used to demonstrate the performance of our proposed deep learning-based hybrid model. A dataset [5], which is frequently used in the literature and included in the benchmark table, was selected and evaluated for analysis. This dataset prepared by Morris et al. was analyzed with our proposed HYBRID3b model and a high success rate was obtained for attack detection. This result was shown in the last row of Table 7.

It has been observed that our proposed models have higher or very close performances compared to other models in the literature. Due to the diversification and development of attacks, it is important to carry out new analyses on different environments, and this has been achieved in this study. As a result, it is important that attack detection studies for SCADA systems are frequently updated and diversified.

6. Conclusions

The continuous functionality of a SCADA system enables smooth operation of cri-tical infrastructure systems. DDoS attacks against SCADA systems may interrupt the whole system causing functionality lost. Interruption in the operation of the SCADA system can be costly from both financial and time aspects. The methods proposed in the study will reinforce SCADA systems against cyberattacks. Thus, early DDoS attack detection on the system will be possible, and it will be easier to prevent disaster scenarios.

In this study, DDoS attacks were performed against the prepared testbed using the SCADA system. The obtained data both under the attacks and without attacks were recorded. LSMT, CNN, LSTM-CNN hybrid, and machine learning-based models were tested on the preprocessed dataset. After modifying the parameters of the models, various versions were obtained and used. The deep learning-based LSTM-CNN hybrid model achieved a classification accuracy of 95.00%, and the machine learning-based decision tree model achieved a classification accuracy of 99%. For a further evaluation of the success of the hybrid model, tests were conducted on a commonly used dataset in the literature which resulted in a high success rate of 98%. A higher success rate was achieved compared to the study in the literature using this dataset.

In addition to DDoS attack detection, DDoS attack type detection was also performed. With deep learning-based and machine learning-based models, all TCP, SYN, and ICMP flooding attacks were correctly detected. These models will provide high success and efficiency in the detection of such attacks. In this respect, it is aimed to contribute to the literature and provide guidance for future studies.

More detection studies should be carried out to reduce the effects of DDoS attacks on SCADA systems. Since SCADA systems are used in many different sectors, studies should be diversified by using different and new technologies and environments. In future studies, it should be an aim to prepare the SCADA system testbed environment more comprehensively and effectively. It should be an aim to apply different type of attacks other than DDoS attacks to this environment and to diversify the models used for their detection.

Author Contributions: The first author (E.S.) conducted experiments, wrote the manuscript, and executed software process. The second author (O.A.E.) was responsible for supervision and correcting the direction of the work. All authors have read and agreed to the published version of the manuscript.

Funding: The authors received no financial support for the research, authorship, or publication of this paper.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fanuscu, M.C.; Kocak, A.; Alkan, M. Detection of Counter-Forensic Incidents Using Security Information and Incident Management (SIEM) Systems. In Proceedings of the 2022 15th International Conference on Information Security and Cryptography (ISCTURKEY), Ankara, Turkey, 19–20 October 2022; pp. 74–79. [CrossRef]
2. Domínguez, M.; Prada, M.A.; Reguera, P.; Fuertes, J.J.; Alonso, S.; Morán, A. Cybersecurity training in control systems using real equipment. *IFAC-PapersOnLine* **2017**, *50*, 12179–12184. [CrossRef]
3. Stouffer, K.; Pillitteri, V.; Lightman, S.; Abrams, M.; Hahn, A. Guide to Industrial Control Systems (ICS) Security. *NIST Spec. Publ.* **2015**, *800*, 16. [CrossRef]
4. Fabro, M. *Study on Cyber Security and Threat Evaluation in SCADA Systems*; Lofty Perch Inc Markham, Defence Research and Development Canada: Markham, ON, Canada, 2012; pp. 13–16.
5. Fink, K.R.; Spencer, D.F.; Wells, R.A. *Lessons Learned from Cyber Security Assessments of Scada and Energy Management Systems*; United States Department of Energy Office of Electricity Delivery and Energy Reliability: SW Washington, DC, USA, 2006.
6. Dominguez, M.; Fuertes, J.J.; Prada, M.A.; Alonso, S.; Morán, A.; Perez, D. Design of Platforms for Experimentation in Industrial Cybersecurity. *Appl. Sci.* **2022**, *12*, 6520. [CrossRef]
7. Söğüt, E.; Erdem, O.A. Endüstriyel Kontrol Sistemlerine (SCADA) Yönelik Siber Terör Saldırı Analizi. *J. Polytech.* **2019**, *23*, 557–566. [CrossRef]
8. Zhang, L. An Implementation of SCADA Network Security Testbed. Master's Thesis, University of Victoria, Victoria, BC, Canada, 2015.
9. Gao, W.; Morris, T.H. On Cyber Attacks and Signature Based Intrusion Detection for Modbus Based Industrial Control Systems. *J. Digit. Forensics Secur. Law* **2014**, *9*, 3. [CrossRef]
10. Queiroz, C.; Mahmood, A.; Tari, Z. SCADASim—A Framework for Building SCADA Simulations. *IEEE Trans. Smart Grid* **2011**, *2*, 589–597. [CrossRef]
11. Farwell, J.P.; Rohozinski, R. Stuxnet and the Future of Cyber War. *Survival* **2011**, *53*, 23–40. [CrossRef]
12. Available online: https://www.securityweek.com/remote-hacker-caught-poisoning-florida-city-water-supply/ (accessed on 5 March 2023).
13. Tesfahun, A.; Bhaskari, D.L. A SCADA testbed for investigating cyber security vulnerabilities in critical infrastructures. *Autom. Control. Comput. Sci.* **2016**, *50*, 54–62. [CrossRef]
14. de Brito, I.B.; de Sousa, R.T., Jr. Development of an open-source testbed based on the modbus protocol for cyber-security analysis of nuclear power plants. *Appl. Sci.* **2022**, *12*, 7942. [CrossRef]
15. Khan, A.A.Z. Misuse intrusion detection using machine learning for gas pipeline SCADA networks. In Proceedings of the International Conference on Security and Management (SAM), Las Vegas, NV, USA, 29 July–1 August 2019; pp. 84–90.
16. Polat, H.; Türkoğlu, M.; Polat, O.; Şengür, A. A novel approach for accurate detection of the DDoS attacks in SDN-based SCADA systems based on deep recurrent neural networks. *Expert Syst. Appl.* **2022**, *197*, 116748. [CrossRef]
17. Teixeira, M.A.; Salman, T.; Zolanvari, M.; Jain, R.; Meskin, N.; Samaka, M. SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach. *Futur. Internet* **2018**, *10*, 76. [CrossRef]
18. Morris, T.; Srivastava, A.; Reaves, B.; Gao, W.; Pavurapu, K.; Reddi, R. A control system testbed to validate critical infrastructure protection concepts. *Int. J. Crit. Infrastruct. Prot.* **2011**, *4*, 88–103. [CrossRef]
19. Nader, P.; Honeine, P.; Beauseroy, P. Detection of cyberattacks in a water distribution system using machine learning techniques. In Proceedings of the 2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC), Beirut, Lebanon, 21–23 April 2016; pp. 25–30. [CrossRef]
20. Yang, Y.; McLaughlin, K.; Sezer, S.; Littler, T.; Im, E.G.; Pranggono, B.; Wang, H.F. Multiattribute SCADA-Specific Intrusion Detection System for Power Networks. *IEEE Trans. Power Deliv.* **2014**, *29*, 1092–1102. [CrossRef]
21. Almalawi, A.; Yu, X.; Tari, Z.; Fahad, A.; Khalil, I. An unsupervised anomaly-based detection approach for integrity attacks on SCADA systems. *Comput. Secur.* **2014**, *46*, 94–110. [CrossRef]
22. Kalech, M. Cyber-attack detection in SCADA systems using temporal pattern recognition techniques. *Comput. Secur.* **2019**, *84*, 225–238. [CrossRef]
23. Gao, J.; Gan, L.; Buschendorf, F.; Zhang, L.; Liu, H.; Li, P.; Dong, X.; Lu, T. Omni SCADA Intrusion Detection Using Deep Learning Algorithms. *IEEE Internet Things J.* **2020**, *8*, 951–961. [CrossRef]
24. Maglaras, L.A.; Jiang, J.; Cruz, T. Integrated OCSVM mechanism for intrusion detection in SCADA systems. *Electron. Lett.* **2014**, *50*, 1935–1936. [CrossRef]
25. Gao, J.; Gan, L.; Buschendorf, F.; Zhang, L.; Liu, H.; Li, P.; Dong, X.; Lu, T. LSTM for SCADA Intrusion Detection. In Proceedings of the 2019 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM), Victoria, BC, Canada, 21–23 August 2019; pp. 1–5. [CrossRef]
26. Shitharth, S.; Prince Winston, D. An enhanced optimization based algorithm for intrusion detection in SCADA network. *Comput. Secur.* **2017**, *70*, 16–26. [CrossRef]

27. ADFA. Intrusion Detection Datasets. 2013. Available online: https://research.unsw.edu.au/projects/adfa-ids-datasets (accessed on 1 January 2023).
28. An Industrial Control System Cybersecurity Performance Testbed. 2015. Available online: http://nvlpubs.nist.gov/nistpubs/ir/2015/NIST.IR.8089.pdf (accessed on 25 December 2022).
29. Yang, Y.-S.; Lee, S.-H.; Chen, W.-C.; Yang, C.-S.; Huang, Y.-M.; Hou, T.-W. Securing SCADA Energy Management System under DDos Attacks Using Token Verification Approach. *Appl. Sci.* **2022**, *12*, 530. [CrossRef]
30. Güllü, M.; Akcayol, M.A.; Barışçı, N. Machine Learning-Based Comparative Study for Heart Disease Prediction. *Adv. Artif. Intell. Res.* **2022**, *2*, 51–58. [CrossRef]
31. Duman, E. Implementation of XGBoost Method for Healthcare Fraud Detection. *Sci. J. Mehmet Akif Ersoy Univ.* **2022**, *5*, 69–75.
32. Hernandez-Suarez, A.; Sanchez-Perez, G.; Toscano-Medina, L.K.; Olivares-Mercado, J.; Portillo-Portilo, J.; Avalos, J.-G.; Villalba, L.J.G. Detecting Cryptojacking Web Threats: An Approach with Autoencoders and Deep Dense Neural Networks. *Appl. Sci.* **2022**, *12*, 3234. [CrossRef]
33. Oyucu, S. A Novel End-to-End Turkish Text-to-Speech (TTS) System via Deep Learning. *Electronics.* **2023**, *12*, 1900. [CrossRef]
34. Krithivasan, K.; Pravinraj, S.; Shankar Sriram, V.S. Detection of Cyberattacks in Industrial Control Systems Using Enhanced Principal Component Analysis and Hypergraph-Based Convolution Neural Network (EPCA-HG-CNN). *IEEE Trans. Ind. Appl.* **2020**, *56*, 4394–4404. [CrossRef]
35. Demirtas, M.; Koc, K. Parameter Extraction of Photovoltaic Cells and Modules by INFO Algorithm. *IEEE Access* **2022**, *10*, 87022–87052. [CrossRef]