

Comparison of Machine Learning Based Anomaly Detection for Energy Consumption Values in SDN-IoT Based Home Area Networks

Hilal Yıldız¹, Musa Balta^{2,*}

¹⁻²Sakarya University, Faculty of Computer and Information Sciences, Sakarya, Türkiye, ror.org/04ttnw109

Corresponding author:

Musa Balta, Sakarya University,
Faculty of Computer and Information Sciences,
Sakarya, Türkiye
mbalta@sakarya.edu.tr

ABSTRACT

The problems of traditional electricity grids have led to the emergence of smart grids. Unlike traditional energy systems, smart grids play an important role in the energy sector with their flexibility, programmability and reliability. However, the heterogeneous structure of smart grids consisting of different devices and protocols poses some problems in terms of complexity, service quality and security. In the literature, SDN (Software Defined Networks) paradigm is proposed as a solution to these problems. SDN and smart grid integration makes the energy sector more efficient, reliable and sustainable. On the other hand, smart meters used in the consumption area of smart grids provide instantaneous transmission of energy production and consumption data in homes to the center. With the support of IoT (Internet of Things) of these meters and components in the home area network (oven, IP camera, TV, etc.), the energy supply and demand balance can be managed more smoothly.

In this study, a software-defined and IoT-based smart home architecture is proposed to obtain real energy consumption data. The proposed architecture is developed and implemented on the Mininet simulator with python code. As a result of simulations run under different process and attack scenarios, energy consumption data sets were created. A comparison of the anomaly detection performances of machine learning algorithms on the data sets that are considered to contribute to the literature has been made. As a result of this comparison, it was observed that the success rate of the random forest algorithm was higher than the other algorithms with 90-95 percent.

Keywords: SDN, Smart grid, IoT network, Smart home, Energy consumption

Article History:

Received: 17.02.2025

Revised: 18.07.2025

Accepted: 17.09.2025

Published Online: 29.09.2025

1. Introduction

Today, existing electricity grids are facing increasing problems such as the use of outdated technologies, one-way communication systems, reliability issues and cost inefficiencies. The adoption and integration of smart grid technologies proposed as a solution to these problems is gaining momentum day by day. Smart grids enable efficient management of electricity grids with features based on real-time data collection, analysis and processing at every stage of energy generation, transmission, distribution and consumption. In addition, they enable more efficient use of energy resources and the creation of sustainable energy supply systems by providing the opportunity to manage energy supply through the coordination and monitoring of energy generation. Therefore, smart grids are recognized as an important part of the transformation process in the energy sector.

On the other hand, smart grids have a complex and heterogeneous network structure that requires many different devices and protocols to work together. This situation may create difficulties in managing integrated structures, security of the network and solving problems [1]. Problems such as flexibility, dynamic management, programmability, scalability and security encountered in traditional networks continue to be a problem for the existing network structures of today's smart grids. In the reviewed academic studies, it has been observed that the SDN paradigm is preferred for the network infrastructure of smart grids in order to avoid these problems [2, 3].

SDN is a technology that controls network devices and network resources with a centralized management system based on the principle of separation of data and control planes. With this technology that redefines network management, network administrators can manage network traffic more flexibly, quickly deploy new services and protect network security more effectively [5]. The integration of SDN technology with smart grids enables the creation of a more efficient, safer and more sustainable energy management system in the energy sector. This integration is considered as an important step for the widespread use of smart grids in the future.

In smart grids, energy production and consumption data within the consumption domain are obtained through smart meters. Smart meters in home area networks (HAN) have made it possible for consumers to instantly monitor energy consumption in their homes. On the other hand, when smart home systems, which emerged with the widespread use of IoT applications in home networks, work integrated with smart meters, energy consumption data in the home area network can be collected in a wider range [6]. This enables consumers to save energy by analyzing their energy consumption habits.

One of the applications of the Internet of Things is smart home systems, which are characterized by homes where all equipment is interconnected and communicates with one another. White goods like dishwashers, refrigerators, and washing machines, as well as air conditioning, lighting, and other sensor equipment, can all be part of the HAN's smart home systems. These IoT-enabled gadgets facilitate the transfer of energy production and consumption and improve the accuracy and dependability of data processing. Customers can then use this information to track and optimize their household energy usage [7].

Even while the spread of IoT technology makes life much more convenient, if safety measures are not followed, security flaws and other dangers will develop. Numerous studies in literature have looked at some of the concerns associated with the growing adoption of IoT-based smart home systems [8]. The nature of IoT is that everything is connected to the internet, which means that the legal and technical framework required to manage thousands or even millions of devices is quite challenging. Risks that require consideration include those related to privacy and security, cyberattack vulnerability, data integrity, and network compatibility [9]. Through internet connections, IoT devices create networks, and unauthorized access to these networks could be dangerous. For example, the security of the data can be threatened by targeting the protocols used during the transmission of data in the smart home to the meters or from the meters to the center. The vulnerabilities in the structure of some protocols are seen as an opportunity for attackers. The attacker can manipulate the data by performing an attack in accordance with the vulnerability of the protocol used or use the captured data for their own purposes [10]. While this situation threatens the overall security of the smart grid, it also means that the security of personal data is also violated.

With the ever-increasing data size of many end nodes within the smart grid structure, it is becoming increasingly difficult to detect these vulnerabilities. For this reason, machine learning algorithms are used in the processes of analyzing and processing the data both in the network and in the smart home system [11]. Thus, by analyzing the data produced by the system, possible anomalies can be easily detected, and network administrators can be informed and faster and more effective interventions become possible [12].

Recent research has demonstrated the significant potential of advanced machine learning techniques to enhance the security and operational reliability of critical infrastructures. Integrating machine learning algorithms for real-time anomaly detection in log management systems has proven effective in various critical infrastructures, including water systems [38]. Similarly, the application of optimized deep learning models has demonstrated high accuracy in predicting stability and uncovering dynamic patterns within smart grid environments [39]. These findings highlight the importance of leveraging state-of-the-art machine learning methods for timely detection of anomalies and cyber threats, as well as for gaining a deeper understanding of complex system behaviors in next-generation smart grids.

However, the literature lacks comprehensive studies that integrate SDN and IoT paradigms within the HAN domain of smart grids in a unified architecture, especially focusing on both normal and abnormal data, and the impact of this integration on both energy consumption and security has not been thoroughly investigated in previous studies.

In this context, in this study, SDN and IoT network paradigms are applied on scenario-based smart grids in an integrated manner and the performances of different machine learning algorithms in anomaly detection are measured. The main contributions of the study can be summarized as follows:

- Unlike the studies in literature, a HAN infrastructure has been designed by effectively integrating the characteristics of SDN and IoT network paradigms in the consumption area, which is the most important component of smart grids in energy supply/demand. It is thought that this designed infrastructure can be a reference for other studies in terms of technical content.
- The architecture, which includes various IoT components, was evaluated in normal and attack situations and different scenarios were created. As a result of the scenarios applied with the special codes developed, CSGEC-23 (CENTER Smart Grid Energy Consumption 2023), which is a comprehensive and special data set that takes into account the energy consumption values of the devices used in architecture as well as the physical and software features that affect these consumption values, has been created.
- In order to lead the process of deciding on the methods/algorithms to be used in the environmental security components of national (Republic of Turkey Ministry of Energy and Natural Resources, EMRA, etc.) and international standards (NIST, NERC, etc.) published on the security of smart grids, a performance comparison of the most commonly used machine learning algorithms in anomaly detection in the literature was made.

In summary, this study aims to fill the gaps identified in the literature by presenting an integrated architecture, generating a comprehensive dataset, and benchmarking anomaly detection algorithms specifically for SDN and IoT-enabled HANs in smart grids.

After the introduction, in Section 2, literature review and basic information will be given. In Section 3, the proposed architecture will be described, and the data sets obtained by simulating normal and attack situations on the architecture will be explained in detail. In Section 4, the performance comparison of the machine learning algorithms to be used in anomaly detection is made according to the determined metrics. Finally, in Section 5, the results obtained are evaluated and directions for future work are discussed.

2. Knowledge and Literature Review

2.1 General Information and Architecture of Smart Grids and Home Area Networks

In the literature, there are architectures that define the functions of smart grid components and how they interact with each other in different ways. The common goal of these architectures is to increase energy efficiency, integrate renewable energy sources, ensure energy security, balance energy supply/demand and facilitate energy consumption management.

The architecture developed by the National Institute of Standards and Technology (NIST) and designed to standardize smart grid applications in the USA is shown in Figure 1 [13].

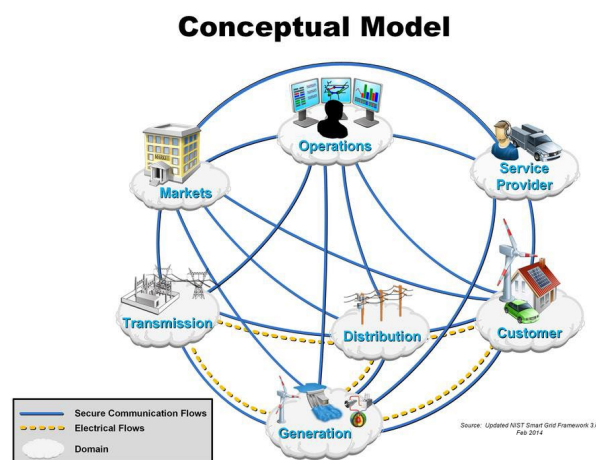


Figure 1. NIST smart grid architecture [13]

This architecture provides a framework that classifies grid components according to specific functions and standardizes the communication between each component. The architecture consists of 7 grid components.

- Generation is the area that provides the production of energy. In this area, facilities such as power plants generating electrical energy and renewable energy sources are located.
- Transmission is the transmission of the generated energy to distribution centers. In this area, there are structures such as high voltage lines and transmission centers.
- Distribution is the transmission and distribution of energy to end users. This area includes structures such as distribution networks that provide energy to homes and businesses.
- Customer represents individuals or organizations that consume energy. In this area, the energy supply-demand balance is ensured in line with the demands and needs of customers.
- Markets, where the commercial activities of energy take place. In this area, commercial transactions such as pricing, sale and distribution of energy are carried out.
- Service Provider is the provider of energy services. This area includes energy suppliers, distribution companies and companies offering energy management services.
- Operations are the management of energy generation, transmission and distribution systems. In this area, activities such as energy production facilities, operation and maintenance of networks are carried out.

Among these components, the consumption domain is the most exposed to cyber-attacks. There is no national or international standard or regulation determined for the architecture to be created in home area networks (HAN) in this domain. This domain, which is completely modelled according to the needs of the consumer, has been the subject of many studies [14, 15].

2.2 Types of Attacks on Smart Grids and IoT Networks

Smart grids are systems that have an important place in the energy sector and provide many advantages. With the use of these systems, energy efficiency is ensured, the use of energy obtained from renewable energy sources is increased and energy costs are reduced. Therefore, developing and increasing the use of smart grids is important for the future of the energy sector. On the other hand, since smart grids provide interaction between many different devices and systems, the risk of cyber-attack increases. Smart home systems and IoT networks connected to the smart grid system are among these attack interfaces.

Smart grids are systems that have an important place in the energy sector and provide many advantages. With the use of these systems, energy efficiency is ensured, the use of energy obtained from renewable energy sources is increased and energy costs are reduced. Therefore, developing and increasing the use of smart grids is important for the future of the energy sector. On the other hand, since smart grids provide interaction between many different devices and systems, the risk of cyber-attack increases. Smart home systems and IoT networks connected to the smart grid system are among these attack interfaces.

Many studies in the literature have examined the risks associated with the widespread use of smart home systems [8]. Risks that require consideration include those related to privacy and security, cyberattack vulnerability, data integrity, and network compatibility [9]. Especially in the context of IoT-based smart homes, these security concerns are particularly serious. For instance, cybercriminals can take control of the home's heating and cooling systems. By tampering with the heating system, they can raise energy costs or, worse, start a house fire that results in property damage and fatalities [7]. Additionally, homeowners' personal information is accessible to cybercriminals. Cybercriminals can intercept data that is continuously collected by household devices. Homeowners' personal information is compromised as a result [16]. This information might include the homeowner's location, interests, habits, actions, and even health. For example, a smart thermostat can collect information about when the homeowner is at home, what temperatures are preferred, and perhaps information about sleeping patterns. Similarly, a smart security camera can record when the homeowner is home, who arrives and who leaves. Cyber attackers can use this personal data for fraud or blackmail and/or share it with third parties.

Attackers may use different methods to carry out these cyber-attacks. The attack method to be chosen varies according to the competence of the attacker, the structure of the network and the purpose. Attack types are generally divided into two as active and passive. Passive attack is a type of attack made by the attacker monitoring the traffic on the network. In this type of attack, the attacker can stay passive and listen to the traffic on the network, intercept sensitive information between data, or learn the weak points in the network and then perform an active attack targeting these points. An active attack is a much more dangerous type of attack in which the attacker intervenes and modifies the network. In this type of attack, the attacker can directly intervene by targeting network components or systems [17]. For example, the attacker can inject false data into the network and manipulate network traffic by misrouting the data or perform attacks such as denial of service (DoS) attacks on the network.

When it comes to smart grids and IoT networks, a wide variety of attacks are carried out at different points of the system. This is due to the large attack surface caused by the heterogeneous structure of both smart grids and IoT networks. Figure 2 shows the different types of attacks that can be performed on smart grids and IoT networks. In this study, only attacks within the scope of manipulation of energy consumption values of smart devices are modelled. The realized attacks are underlined and shown as red in Figure 2.

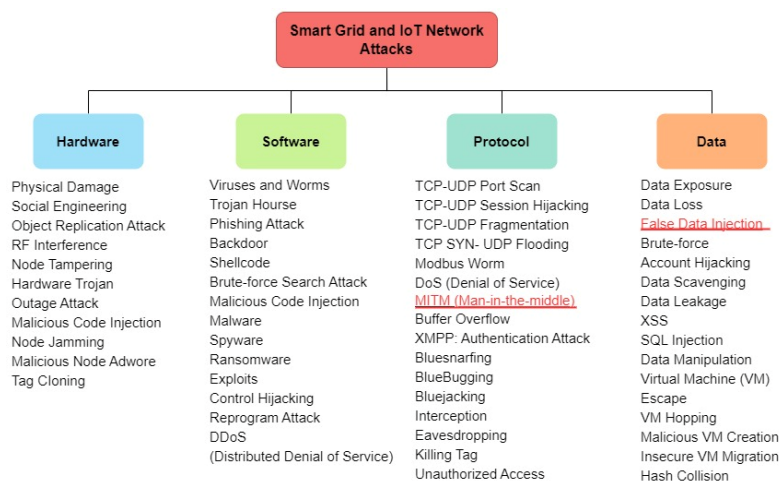


Figure 2. NIST smart grid cyber attack classification

3. The Proposed Architecture and Dataset Generation

As smart homes have become popular, IoT technology, which enables users to remotely monitor and manage smart devices in their homes and communicate with each other, also helps to optimize energy consumption. Today, smart meters, which

are an important tool in energy management, offer important opportunities in energy management and energy saving when used with IoT technology. Together with these technologies, SDN technology also makes network management in energy networks more effective, making it easier for energy companies to manage energy consumption and distribution.

As a result, managing smart homes with IoT technology and supporting them with SDN-based energy grids is an important step in energy management and saving. These technologies are important tools that can be used to optimize energy consumption and balance supply and demand in a world where energy supply is limited.

In this study, deficiencies are identified by analyzing the studies in literature, and then a software-defined and IoT-based smart home network architecture is proposed that can meet today's needs, where these systems work in an integrated manner. In this context, in this section of the study, information about the proposed software-defined and IoT-based smart home network architecture will be given.

3.1 Creation of Physical Infrastructure

Figure 3 shows the IoT-based smart grid architecture and software defined in this study. In this architecture, the home area network (HAN) in the consumption domain is designed as an Internet of Things (IoT) based smart home system, and the smart grid structure, which is composed of seven components (generation, transmission, distribution, consumption, market, service provider, and management), is planned as SDN based.

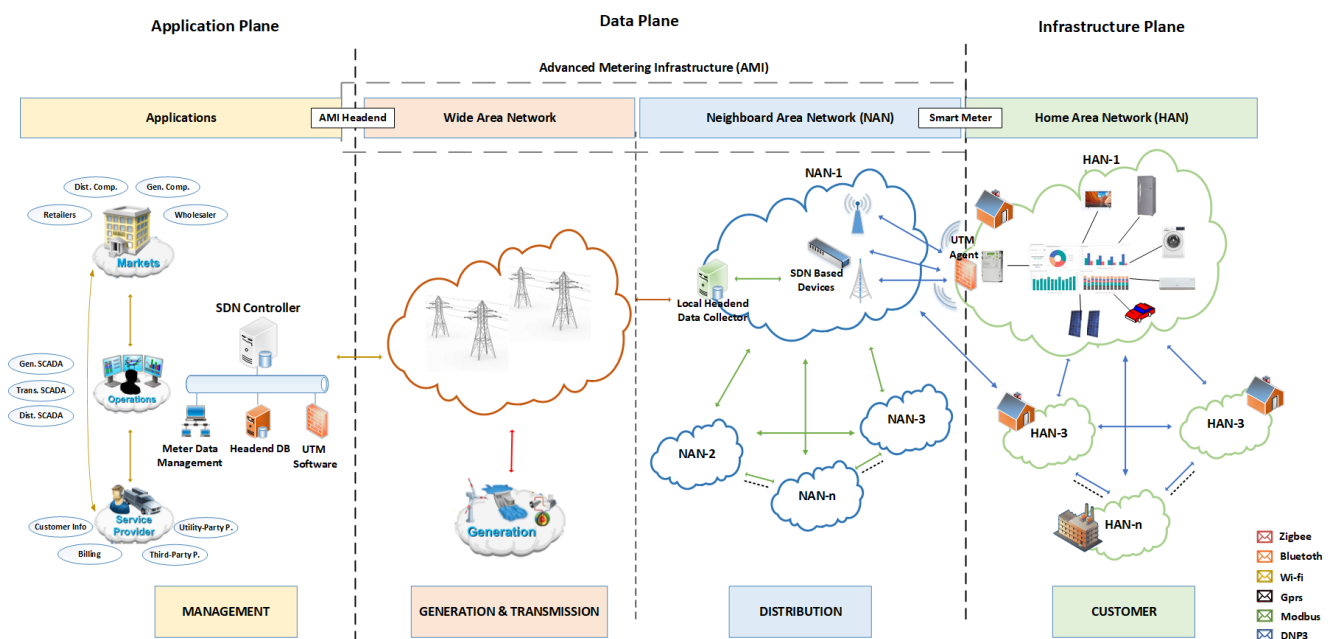


Figure 3. Proposed software defined and IoT based smart grid architecture

According to the designed architecture, smart homes in the consumption domain are connected to the energy management system via SDN network. Communication between the center and the smart home is carried out through the SDN controller that manages the wide area network (WAN). Smart homes participate in the SDN network with smart meters, which are the data output point. The amount of energy consumed and produced within the home area network is measured by smart meters and sent to the center via network devices in the SDN network. The energy management system in the management domain analyses this data and provides energy supply and demand balance. Thus, resources are utilized more efficiently. This proposed model is designed in accordance with the smart grid architecture proposed by NIST and smart home systems accepted in the literature.

The primary motivation for utilizing SDN technology in this architecture is its ability to provide centralized, flexible, and programmable network management. Through the SDN controller, network traffic can be dynamically monitored and controlled, allowing for rapid response to changing conditions and potential threats. Furthermore, the programmability of SDN enables the swift implementation of security policies and effective anomaly detection, thereby enhancing both the security and sustainability of the system. These features contribute to a more efficient and secure operation of the energy management system [40, 41].

In this study, a network virtualization tool called Mininet is used to implement the proposed architecture. Mininet is a Python application running on a Linux machine and is very useful for simulating software defined networks (SDN). This application creates a network topology with virtual hosts, switches, and routers. These virtual devices exhibit behavior similar to real-world devices, and Mininet simulates how they interact with each other. Miniedit is a graphical interface used to create Mininet simulations in a faster and more understandable way. Using this visual interface, virtual networks can be created,

edited, monitored, and managed. In this study, Mininet's default controller was used to manage the network. Figure 4 shows an example model of the architecture proposed in this study created with Miniedit.

In the architecture, a virtual host is created to represent IoT supported smart devices such as white goods, small appliances, air conditioning and lighting systems in smart homes. Each host actually works as an xterm, i.e. terminal. In each xterm, the codes belonging to the device it represents are executed and simulation is performed by imitating the energy consumption of the relevant device.

Unlike studies in literature, instead of selecting devices with fixed features, architecture is designed to have different features for each device in order to effectively reflect real-life use cases. Based on this approach, when xterm is run, for example, the features that the refrigerator will have been randomly selected according to the values that the predetermined fields may have. The host, which calculates the energy consumption of the refrigerator with all the properties determined, sends this consumption data to the network. Thus, energy consumption data is generated in the network as if a refrigerator is running.

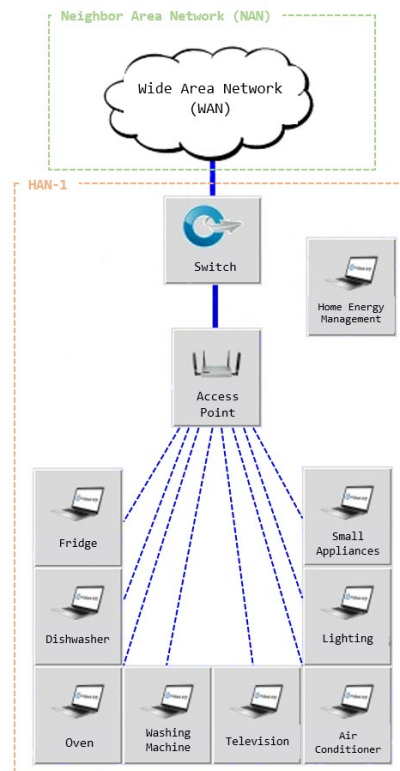


Figure 4. Smart home topology created in simulation

Each host imitates the operation of the device and sends the energy consumption value to the SDN switch device via the access point. The SDN supported switch device to which the access point is connected is responsible for transmitting the consumption data to the smart meter. The consumption data received by the smart meter is transmitted to the OSOS (automatic meter reading) software in the center via the SDN-based energy network. The consumption data sent to the smart meter is also shared with the HEM (Home Energy Management) software via the access point.

HEM software is a system that uses a control panel to read energy consumption data from devices and regulate it in accordance with user requests. This app gives customers comprehensive information about the energy usage and condition of smart gadgets in their homes. Energy consumption and smart home tracking are thus made manageable.

Anomalies in energy consumption can also be recognized thanks to HEM software. This software, which enables real-time monitoring of the consumed energy values, also saves this data in the database located locally. It also includes a system that detects abnormal situations by analyzing data with various machine learning algorithms. In this way, a cyber-attack on the smart home system or a situation where the devices are out of the usual working order can be easily detected.

The reason why anomaly detection is performed in the home area network is that there are many end nodes, i.e. subscribers, within the smart grid structure. Analyzing all subscriber data at the center and performing anomaly detection is not an appropriate and feasible approach in terms of performance in today's technologies. For this reason, the edge computing approach is adopted in the designed architecture. In this approach, data is captured and processed by network devices, control devices or other sources before being sent to cloud centers [18]. This approach, which is based on the logic of sending

processed data to the center instead of raw data, aims to improve network and service performance by reducing network traffic and delays.

The smart home architecture proposed in this study is shown in Figure 5 [14]. While designing this architecture, the needs of today's homes and the expectations of the users were taken into consideration. The devices in the designed smart home are as follows: Refrigerator, dishwasher, oven, washing machine, television, air conditioner, lighting/bulb, toaster, kettle, vacuum cleaner, iron.



Figure 5. Proposed smart home architecture [14]

In order to monitor the smart home system correctly, it is necessary to have detailed information about the devices used at home. For example, when two washing machines with the same features are compared, the electrical energy consumed by two machines with a capacity of 7 kg and 10 kg differs. For this reason, unlike the studies in literature, instead of selecting devices with fixed features, the architecture is designed to have different features for each device in order to effectively reflect the real-life energy consumption. For this purpose, the technical documentation and user manuals of the products of the 5 preferred brands in the market were examined in detail and the features that 8 different smart devices/systems in the designed house can have been determined. These features are presented in Table 1. Then, in order to approximately represent each house, the final consumption values of the relevant device were determined by averaging the energy consumption values of the devices with the same characteristics belonging to 5 different brands. The product information and energy consumption values obtained as a result of this intensive research were recorded to be used in the simulation environment.

Table 1. Characteristics of the devices used in smart home architecture

Device	Features					
Fridge	Energy Class	Volume	Number of people living at home	Energy consumed		
Dishwasher	Energy Class	Water consumption	Program	Number of uses per week	Energy consumed	
Oven	Energy Class	Type	Program	Number of uses per week	Energy consumed	
Washing Machine	Energy Class	Volume	Revolution	Program	Number of uses per week	Energy consumed
Television	Energy Class	Screen type	Screen size	Weekly usage time	Energy consumed	
Air Conditioning	Energy Class	Type	Capacity	Weekly usage time	Energy consumed	
Lighting	Type	Number of bulbs	Weekly usage time	Energy consumed		
Small Home Appliances	Type	Power	Weekly usage time	Energy consumed		

The communication of the devices whose characteristics and energy consumption structure are determined within the network is realized using IoT protocols. Protocols such as Bluetooth, MQTT, Wi-Fi, CoAP, Z-Wave and Zig-bee which are frequently used in today's smart home systems, are some of them [19]. The communication protocols used depend on the network technology with which the devices communicate, and the characteristics of the devices used. In the software-defined smart grid architecture proposed in this study, Wi-Fi protocol is preferred for communication within the HAN. This protocol was chosen because it is a preferred protocol in literature, it is used in today's smart home systems, and it can be integrated into the Mininet simulation program.

3.2 Scenarios

Communication within smart homes is established by the devices sending their data via Wi-Fi to the HEM software where the smart home system is monitored and managed. This data is generated in the Mininet simulation program according to the previously defined scenarios. One of the unique values of this study is that smart home architecture works in a similar way to real life. The key point to achieving this similarity is that, unlike the studies in the literature, instead of modelling a single smart home with specific features, an approach is presented in which all possible smart home scenarios that can be encountered in real life can be modelled within certain constraints. The basis of this approach is based on the fact that each of the devices in the smart home can be models with different characteristics. For example, if we assume that a smart home has a refrigerator of model A, a washing machine of model B and a television of model C, this constitutes a possible scenario in a smart home. With this logic, unique combinations of a total of 28 different features of the 8 smart devices in the architecture represent different smart home systems. The number of these combinations is calculated as 3,851,755,393,646,592. In this study, 2 million different scenarios were identified to be run in a simulation environment to increase usability. One of the scenarios used in the data set is given in Table 2. Additionally, Table 3 presents a sample from the dataset.

Table 2. Scenario Example

Device	Features					
Fridge	D	Large	4	8,281 kW		
Dishwasher	C	9,5	Eco	2	5,888 kW	
Oven	A	Built-In Oven	Chicken	3	10,131 kW	
Washing Machine	B	9 Kg	1200	Cotton	3	15,9 kW
Television	F	4K UHD	65"	8-12 Hours	1,102 kW	
Air Conditioning	A++	Split	12000 Btu/h	8-12 Hours	10,8585 kW	
Lighting	Fluorescent	13-12	20-30 Hours	8,3545 kW		
Small Home Appliances	Iron	2600	7-10 Hours	20,8 kW		

Table 3. Sample Record from the Dataset

F_EClass	3
F_Volume	1
F_PNumber	1
F_EConsum	7,441
D_EClass	2
D_WConsumption	1
D_Program	3
D_WUsage	3
D_EConsum	2,961
O_EClass	2
O_Type	2
O_Program	4
O_WUsage	3

O_EConsum	6,312
WM_EClass	1
WM_Volume	2
WM_Revolution	3
WM_Program	1
WM_WUsage	3
WM_EConsum	14,736
TV_EClass	3
TV_SType	1
TV_SSize	2
TV_WUsage	4
TV_EConsum	1,08
AC_EClass	2
AC_Type	3
AC_Capacity	4
AC_WUsage	1
AC_EConsum	12,3
L_Type	1
L_BNumber	1
L_WUsage	2
L_EConsum	0,157325
SHA_Type	2
SHA_Power	1
SHA_WUsage	3
SHA_EConsum	17,6
RESULT	0

When the model is run within the scope of the specified scenarios, the nodes in the simulation environment act as the device they represent and send their properties and the energy value they consume to the HEM software. These data are stored in the database of the HEM software in order to monitor the operating status of the smart home and the electrical energy consumed as a result. One of the objectives of this study is to process the data in the database and produce a data set.

While the smart home network performs its normal functioning, it may unexpectedly be subjected to cyber-attack and data manipulation by attackers. When cyber-attacks in literature and in real life are analyzed, attacks on IoT systems and smart homes have been identified. The CSGEC-23 dataset includes MITM (man-in-the-middle attack), false data injection, and masquerade attack. Based on the "MITM" and "false data injection" attacks in the Mitre ATT&CK matrix, data was manipulated using specially written codes during the transmission of process data. This manipulation was performed to add attack-induced anomalies to the dataset. These manipulated data were labeled as anomalous and included in the dataset.

The CSGEC-23 dataset generated in this study consists of a total of 2 million data points. The dataset was divided into four equal parts, each containing 500,000 data points. Each part has anomaly rates of 10%, 20%, 30%, and 40%, respectively. These rates were determined to simulate different levels of anomalies and to evaluate the performance of machine learning algorithms under varying anomaly conditions. This configuration allows for a comprehensive analysis of the anomaly detection capabilities of the proposed architecture and a comparative study of the algorithms' effectiveness against different anomaly levels.

3.3 Comparison of the Generated Dataset with Other Datasets in Literature

In this section, 5 of the most well-known datasets in the literature will be described, and then Table 4 will be presented with detailed characteristics for comparison with the CSGEC-23 dataset.

- UNSW-NB15: The dataset's raw network packets were produced at UNSW Canberra's Cyber Range Laboratory to combine synthetic modern attack behavior with actual, everyday operations [20].
- CIDDS-001: By emulating a small business environment, an approach is proposed to generate datasets of normal activities including harmless user behavior and malicious traffic [21].

- CIC-IDS-2017: A timestamp is a dataset created similar to real-world data, labelled on the basis of source and destination IPs, source and destination ports, protocols and attack [22].
- TON_IoT: IoT/IIoT datasets created to evaluate the accuracy and efficiency of different cyber security applications based on Machine/Deep Learning algorithms [23].
- NSL-KDD: KDD'99 is a proposed dataset to solve some problems in the structure of the dataset. It can be used as an effective benchmark dataset to help compare intrusion detection methods [24].

Table 4. Comparison of data sets

Dataset	Data Count	Anomaly (%)	Attack Types	Used Machine Learning Algorithms
UNSW-NB15	2.540.044	12,64	<ul style="list-style-type: none"> • Fuzzers • Exploits • Backdoors • Generic • Reconnaissance • Analysis • DoS • Shellcode • Worms 	<ul style="list-style-type: none"> • Naïve Bayes (NB) • Artificial Neural Network (ANN) • Decision Tree (DT) • Logistic Regression (LR) • Clustering • Expectation-Maximization (EM)
CIDDS-001	4.194.300	2,94	<ul style="list-style-type: none"> • Brute Force • DoS • Port Scanning 	-
CIC-IDS-2017	-	-	<ul style="list-style-type: none"> • DoS/ DDoS • XSS • Brute Force • SQL Injection • Port scanning • Infiltration • Botnet 	<ul style="list-style-type: none"> • Iterative Dichotomiser 3 (ID3) • K-Nearest Neighbors (KNN) • Random Forest • Adaboost • Multi-layer Perceptron (MLP) • Quadratic Discriminant Analysis (QDA) • Naive Bayes
TON_IoT	3. 270.022	16,12	<ul style="list-style-type: none"> • DoS/ DDoS • MITM • Injection • Password • Scanning • Ransomware • XSS 	<ul style="list-style-type: none"> • Support Vector Machines (SVM) • Classification and Regression Trees (CART) • k-Nearest Neighbour (kNN) • Random Forest (RF) • Naïve Bayes (NB) • Linear Discriminant Analysis (LDA) • Logistics Regression (LR)
NSL-KDD	-	-	-	<ul style="list-style-type: none"> • NB Tree • J48 (decision tree learning) • Naive Bayes • Random Tree • Random Forest • Support Vector Machine (SVM) • Multi-layer Perceptron
CSGEC-23	2.000.000	10, 20, 30, 40	<ul style="list-style-type: none"> • False Data Injection • MITM • Masquerade Attack 	<ul style="list-style-type: none"> • K-Nearest Neighbors (KNN) • Decision Tree • Support Vector Machine (SVM) • Naive Bayes • Random Forest • Artificial Neural Networks

4. Performance Comparisons of ML Algorithms for Energy Consumption

This section includes performance tests of the algorithms used to detect anomalous data in the generated data set. These tests aim to demonstrate the suitability of the machine learning algorithms for use in smart home systems.

For this purpose, firstly, the data preprocessing steps applied to the data set will be explained, then the machine learning algorithms used will be mentioned and finally the performance tests will be evaluated.

4.1 Data Preprocessing

The rapidly increasing data size worldwide has led to the development of solutions to facilitate processes such as analyzing and processing these data. Machine learning techniques developed for this purpose are highly sensitive to noise, deficiencies and incompatibility in the data, and this situation negatively affects performance. In order to obtain more accurate, effective and reliable results, the data should be subjected to a series of processes before the application of machine learning techniques.

Although data preprocessing is a concept that has found a wide place in literature, it consists of different steps depending on factors such as the current state of the data set and the expected system output. In this study, data cleaning, reduction, scaling and segmentation techniques were selected from the data preprocessing steps considering the current needs of the data set [25].

The presence of missing values in the data set directly affects the performance of the machine learning algorithm used. There are two methods that can be applied as a solution to this problem. The first one is the discarding of missing values, which can be applied when the ratio of missing values to the entire data set is insignificant. Since most machine learning algorithms cannot process data with missing values, this method can be applied in a small number of cases. The second method is to complete the missing values. This method, which is preferred when the data with missing values are too many to be discarded, involves completing the missing data with estimated values using various completion methods. Studies in the literature show that there is no absolute solution to replace missing data. The method to be chosen should be decided by considering parameters such as the size of the data set, the amount of missing values, and the computational cost [26 – 28]. In this study, the negative impact on performance is prevented by discarding missing data that may occur in the event that smart home appliances do not send data in case of possible malfunctions or in other cases.

If the size of the data set is too large in terms of rows or columns, this may adversely affect the machine learning performance. In such a case, data reduction techniques should be applied from data preprocessing steps. Data reduction can be performed in two ways: row-wise data sample reduction or column-wise feature variable reduction. There are different techniques applied for both reduction methods. Random selection, which can be used for row-wise data reduction, is a technique in which a sample is selected completely randomly from the data set. Although its use in very large data sets is favorable in terms of performance, the possibility of eliminating critical data may negatively affect the machine learning process and reduce the overall performance. Another technique, stratified sampling, is the process of determining the sample by maintaining a certain ratio in the categories. With this method, data reduction can be applied without loss of categories. There are three methods accepted in the literature for data reduction on column basis, which is another direction. The first method is to directly select the desired ones among the variables. The second method is to select variables using feature selection techniques. These techniques are divided into 3 groups. These are filtering techniques, wrapper techniques and embedded techniques. Finally, the third method is to use feature extraction techniques to remove the non-useful variables in order to determine the useful variables. The decision of which of these three techniques to choose should be made by considering the advantages and disadvantages they provide depending on the characteristics of the data set and the expected output [25 – 27]. In this study, in order to improve the performance in the processing and analysis processes of data obtained from smart home appliances, data reduction technique was applied on the generated data set by row-based random selection method.

Possible differences between the scales (values, dimensions) of the variables in the data set cause machine learning algorithms to experience various errors in analyzing these variables. Data scaling techniques are used to eliminate the measurement differences between variables by preventing this situation. With these techniques, it is made possible to evaluate the data on equal terms by the model without confusion in the perception of size/smallness between the data. Data scaling techniques can be divided into three groups. Max-min normalization to convert variable values into 0-1 range, z-score standardization which brings the variable to a normal distribution with mean value 0 and standard deviation 1, and finally decimal scale normalization technique in which decimal structures are carried to reduce the differences of data variables [25]. In this study, data scaling technique with max-min normalization method was applied to improve the performance of machine learning algorithms by reducing the difference between the dimensions of the variables in the generated data set.

Data partitioning aims to divide the data into several groups in order to perform various analyses with machine learning algorithms. This technique provides a test set to evaluate the performance of the model after it has been trained. The proportion and structure of the partitioning is not restricted by strict rules. The partitioning process is shaped according to different expectations such as the way the model works, the size and characteristics of the data set, and the desired output. However, the data can also be divided into three different groups for training, validation and testing. During the partitioning process, the data set should be mixed and randomly selected. Thus, the new groups created become independent from each other and contain the features of the data set. When the studies in literature are examined, it is seen that the data partitioning technique can help to improve model performance. In addition, this technique provides an objective evaluation of the performance of the model on real data and helps to avoid problems such as overlearning of the model [25]. The data in the data set created in this study is divided as 70% training and 30% test data.

After the data preprocessing steps, anomaly detection was performed with machine learning algorithms on the CSGEC-23 data set. In the literature, there are many machine learning algorithms used in intrusion detection [29]. In this section of the study, the preferred algorithms will be explained.

4.2 Machine Learning Algorithms Used in This Study

The selection of machine learning algorithms in this study is based on their complementary characteristics and proven effectiveness in anomaly detection tasks within smart grid and IoT environments, as highlighted in the literature. Specifically, k-Nearest Neighbors (k-NN) and Support Vector Machine (SVM) are widely used for their strong performance in classification problems involving high-dimensional and complex data. Decision Tree and Random Forest algorithms are chosen for their robustness, interpretability, and ability to handle both categorical and numerical features. Naive Bayes is included due to its simplicity, computational efficiency, and effectiveness in handling large datasets with probabilistic approaches. Artificial Neural Networks (ANN) are selected for their capability to model complex, non-linear relationships in data, which is particularly useful for capturing subtle patterns in energy consumption and anomaly detection.

Furthermore, these algorithms are commonly employed in similar studies in literature, which allows for a meaningful comparison between the results of this work and previous research. Prior to their implementation in the proposed architecture, preliminary tests were conducted to evaluate their suitability and effectiveness in addressing the specific requirements of the study. Thus, the use of these widely recognized algorithms not only ensures methodological robustness but also enhances the comparability of the findings with existing studies.

- KNN (K-Nearest Neighbors) is one of the supervised machine learning algorithms. Since its implementation is very simple, it is widely used in classification and regression problems. Due to this simplicity, it is used in many different fields from data mining to medicine, from pattern recognition to statistics. The basis of the algorithm is based on the principle that similar data are close to each other. This approach can be elaborated as calculating the similarity of the data to be classified with the data learnt from the training set and classifying according to the threshold obtained by averaging the K data closest to the sample [30].

Its use in this study can be explained as follows: A new data is to be classified in a system where the data is labelled as normal and abnormal. The algorithm examines K data closest to the new data and looks for similarity between the nearest neighbors. The new data belongs to the class with the majority of neighbors and is labelled as such. The choice of the value of K should be chosen carefully as it affects the precision. Because, if it is chosen too small, the stability of the prediction decreases and if it is chosen too large, the number of errors may increase.

- Support Vector Machines, or SVMs, are machine learning algorithms used for regression and classification. It is recommended to be used in cases where the connection of variables is not known in the classification of the data set. Its main purpose is to draw a hyperplane (decision boundary) by correctly identifying the classes and predicting to which class the new samples belong. The created hyperplane performs the classification process by dividing the data into two or more classes. In order to perform the classification correctly, the farthest point to each class in the plane should be determined while drawing the decision boundary [31].

Its use in this study can be explained as follows: The SVM algorithm is expected to categorize the data into two classes, normal and abnormal. For this purpose, SVM tries to find a hyperplane in a feature space by mapping data points in this space. This hyperplane can also be expressed as a decision boundary that allows discrimination between classes. The hyperplane is calculated by considering the position of each new data added to the space. According to the position of the new data in the plane, it is decided which class it is in.

- A popular supervised machine learning technique for classification and regression issues is the decision tree. This method, which is generally used in data mining, is based on the process of continuously dividing the data into two parts according to a number of parameters by creating attribute nodes. This process is repeated until all data is classified. The decision tree takes its name from the tree-like appearance of the model. The topmost node is called the "root node". The nodes at the bottom of the tree are called "leaves" and are connected to the root node by intermediate "branch" nodes. Testing the data for certain characteristics starts at the root node and branches according to the test result at each node. The test result is categorical, i.e. it contains results such as yes/no. This testing process continues downwards and forms a tree-like flow diagram. When the leaf node at the end of the diagram is reached, the final decision is reached [32].

The use of the algorithm in this study can be explained as follows: Since the decision trees are included in the supervised learning algorithms, they are realized in two stages. The first stage is the model building stage called learning. In this stage, the most important factor affecting the result is identified and determined as the root node. Starting from the root, the structures connected to each other with certain conditions/controls are called nodes. The leaves reached by moving through the nodes indicate the class (decision) to be assigned. Thus, the network, where each node contains a condition, is formed from the root to the leaves. The second stage is the decision stage. In this stage, each data in the test set is applied to the model and the class label of the data is determined according to the tree structure.

- A supervised learning technique for classification and regression issues is called random forest. To increase forecast accuracy and balance, it integrates several decision trees. Unlike decision trees, it is based on training multiple decision trees on randomly selected subsets of the dataset. In this way, it is aimed to improve classification performance. Each tree is trained on a subset of the dataset and then the outputs of all these trees are combined to make final predictions. In classification problems, the most frequently predicted class is determined using majority voting. In regression problems, the final prediction is made by averaging the prediction values of the trees. It gives much better results than other classification and regression algorithms in literature due to its advantages such as its ability to be divided into small and simple trees, to create as many trees as desired, and to make error estimation quickly and with high accuracy [33].

Its use in this study can be explained as follows: The training data set is divided into subsets and decision trees are created. Instead of splitting each tree according to the most important feature, each tree is split according to the best feature in its subset. This helps to increase the diversity in the learning process. Since it is used for the classification (normal/abnormal) problem in this study, the class is determined by majority vote. It is a flexible algorithm that can easily handle large datasets. It has excellent performance and fast training time, but it should be noted that the prediction process slows down as the number of trees increases.

- Naive bayes is a statistics-based classification algorithm based on Bayes theorem. It is calculated as given in Equation 1. A and B in the equation are events. The probabilities of these events occurring independently of each other are expressed as $P(A)$ and $P(B)$. $P(A|B)$ is a conditional statement, i.e. it represents the probability of A happening given B. Similarly, the expression $P(B|A)$ represents the probability of B given A. [34]. It is calculated as given in Equation (1).

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1)$$

It aims to categorize, i.e. classify, the data in the data set according to certain probability calculations. It does this by calculating the probability of the states that each data presented to the system may have. Whichever of the probability values is higher, it decides that the data belongs to that class.

In this algorithm, the learning data presented to the system must be labelled according to a certain category. The test data presented to the system is categorized according to the results of the probabilistic operations performed on the learning data. As the number of learning data increases, the accuracy of this classification increases. Naive bayes, which is one of the most frequently used algorithms in literature thanks to its fast computational capability and simple structure, has proven its success in areas such as text classification, sentiment analysis and spam filtering.

- An artificial neural network is a learning model consisting of a collection of artificial neurons that can detect complex features and patterns and detect anomalies. The model learns normal and attack patterns in the data set during the training process. When a new network traffic arrives in the system, it traverses the feature network and determines the output class. The class with the highest output value is the class label of data. It is a frequently preferred model in intrusion detection systems due to its high flexibility, ability to learn complex features and patterns, and ability to generalize.

Artificial neural networks are a model inspired by the biological structure of the human brain and created by imitating the working order of the neural network and neurons in the brain. Just like in the human brain, artificial neural networks are formed by connecting nerve cells (neurons) to each other. Each neuron receives the input given to it, processes it and transfers the result to the other neuron. This process is continued by interconnected neurons. Thus, an artificial neural network is formed. It usually consists of an input layer, hidden layer and output layer. It may consist of these three layers, or it may contain more than one hidden layer depending on the structure of the system and the expected output. The input layer is the layer where the data enters the neural network model. Here the data is transferred to the hidden layer, where it is prepared for the calculations in the network model. The hidden layer receives the data from the input layer, processes it and performs calculations to identify patterns in the data. This calculation is called the activation function and is a mathematical function applied to all neurons in the neural network. Its main purpose is to calculate the activation level of the neuron according to its inputs. An artificial neural network model may contain more than one hidden layer, and each layer may contain more than one neuron. The output layer produces the output of the neural network by performing calculations on the data coming from the hidden layer. The structure of the layer varies according to the designed model and needs. For example, models used for classification usually use more than one output neuron, each representing a different class, while regression problems may use a single neuron in the output layer. The data moves from input to output in these layers, allowing the data to be evaluated, processed and necessary calculations to be made. Since the data in this study were classified as normal and abnormal, two output neurons were used [35].

In addition to traditional machine learning methods, hybrid systems have gained popularity in recent years. In order to enhance the intrusion detection system's performance, Chuang et al. suggested a hybrid model that combined the Naive Bayes and C4.5 algorithms [36]. To preserve data confidentiality in intrusion detection, Shi et al. suggested a hybrid intrusion detection system that combines the Local Outlier Factor (LOF) and One Class Support Vector Machine (OCSVM) algorithms [37].

Some metrics are used to objectively evaluate and compare the performance of these algorithms.

The Complexity Matrix technique was used for the performance evaluation of the machine learning algorithms used. The FP (False Positive), TP (True Positive), FN (False Negative) and TN (True Negative) values obtained with this technique were then used to obtain the Accuracy, Recall, Precision and F1 Score parameters. The confusion matrix given in Table 5 is a table containing 4 different combinations of the algorithm's predicted value with the actual values.

Table 5. Confusion matrix

		Predicted Value	
		True	False
Actual Value	True	TP (True Positive)	FN (False Negative)
	False	FP (False Positive)	TN (True Negative)

The terms FP, FN, TP and TN, which are the parameters of this table, can be explained as follows in the context of normal-abnormal classification of data:

- FP (False Positive): the machine classifies the data as abnormal when the actual result of the data analysis is normal.
- FN (False Negative): the machine classifies the data as normal when the actual result of the data analysis is abnormal.
- TP (True Positive): the machine classifies that data as abnormal when the actual result of the data analysis is abnormal.
- TN (True Negative): the machine classifies the data as normal when the actual result of the data analysis is normal.

Using these values, performance evaluation metrics of machine learning algorithms such as accuracy, sensitivity, precision and F1 score are obtained.

Accuracy is the percentage of the algorithm's correct classification. It reveals how accurately the algorithm manages to give results. It is calculated as given in Equation (2).

$$\text{Accuracy} = \frac{TP + TN}{TN + TP + FN + FP} \quad (2)$$

Recall is the analysis of how many data that the algorithm should evaluate as positive are found to be positive. It can also be said that the answer to the question "How many of the true positives were evaluated as positive?" is sought. It is calculated as given in Equation (3).

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

Precision is the analysis of how accurate the algorithm's evaluation is. It reveals how many of the outputs labelled as positive are actually positive. It is calculated as given in Equation (4).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (4)$$

F1 score is the harmonic means of the sensitivity and precision values. It is a measure of the accuracy of an algorithm output and takes minimum values of 0 and maximum values of 1 (complete precision). It is calculated as given in Equation (5).

$$\text{F1 Score} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (5)$$

In this section of the study, the performance of machine learning algorithms used for anomaly detection on the CSGEEC-23 dataset is compared. Algorithm performances are evaluated in the light of accuracy, sensitivity, precision and f1 score parameters at different anomaly rates.

4.3 Performance Comparison of the Machine Learning Algorithms Used

Analyzing the device and energy consumption data in the smart home system is as important as storing it securely. For this reason, the data should be processed and correctly classified as normal or abnormal before being stored. Figure 6 shows the results of the test of how accurately the algorithms can perform this analysis.

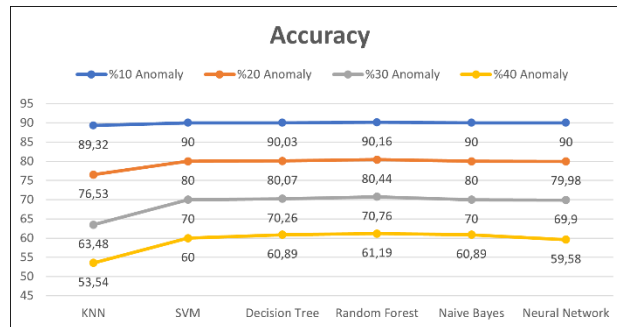


Figure 6. Comparison of the accuracy performance of algorithms

In order to ensure the correct and continuous operation of the smart home system and to eliminate security concerns, it is very important to detect possible cyber-attacks sensitively. Figure 7 shows the results of the recall test of the algorithms.

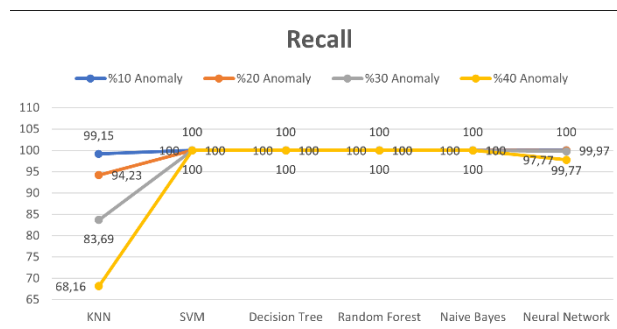


Figure 7. Comparison of the recall performance of algorithms

The accuracy of this detection is as important as the detection of cyber-attacks in the smart home system. Otherwise, undesirable situations such as normal situations being labelled as abnormal and generating alarms or abnormal situations being labelled as normal and not being noticed may occur. Figure 8 shows the results of the precision test of the algorithms.

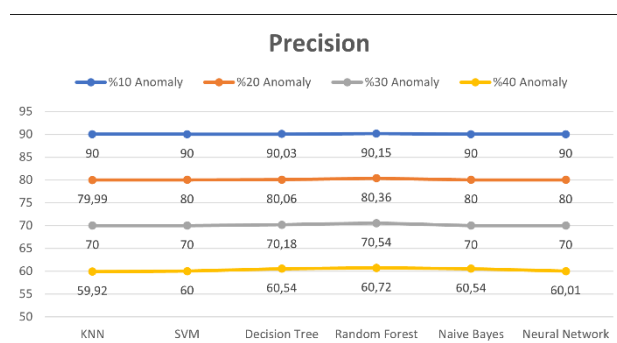


Figure 8. Comparison of the precision performance of algorithms

The performance results of the f1 score value, where sensitivity and precision values are considered together, are given in Figure 9.

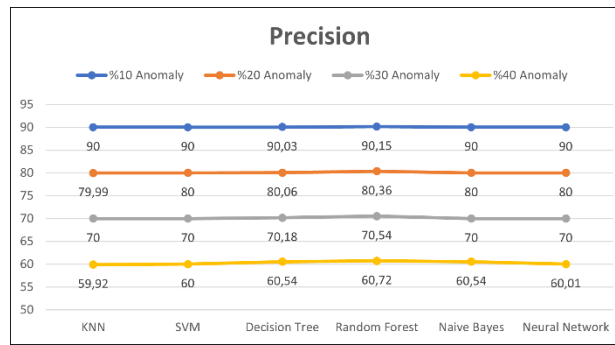


Figure 9. Comparison of the f1 score performance of algorithms

5. Conclusion and Future Work

In this study, IoT supported smart home network is modelled within the scope of the proposed software defined and IoT based smart grid architecture. The parameters influencing energy consumption and consumption values for each smart device have been identified through a thorough analysis of the literature and studies in the field in order to run the model in the simulation environment. A program that executes the model in the simulation environment realistically has been created based on the parameters that were determined. Within the parameters of the devices' primary characteristics (energy class, volume, program, etc.) that influence energy consumption, this software may model the devices' energy usage.

Comprehensive operational scenarios were established in the study's second phase by taking into account user profiles and the structures of modern smart homes. A simulator environment was used to run the scenarios, and system and consumption statistics were gathered. The CSGEC-23 data set, a study result, was produced after the generated data was arranged in the proper format.

The study's third phase involved comparing how well machine learning algorithms performed in identifying anomalous data in the produced data sets. In addition to their suitability for the data set, algorithms' popularity in other research in the literature was taken into consideration when choosing which ones to use.

A possible attack on smart home systems can be detected by analyzing abnormal data. This detection depends on how accurately the anomalous data is classified. When the performance of the algorithms at different anomaly rates is examined, it is seen that the most accurate anomaly detection belongs to the random forest algorithm at all rates.

In smart home systems, it is as important to detect anomalous data as it is to make this detection with precision. When the sensitivity of the algorithms in anomaly detection is compared, it is seen that SVM, decision tree, random forest, naive bayes algorithms show the highest performance. Considering the importance of the sensitivity of anomaly detection in the detection of cyber-attacks, it may be recommended to prefer these algorithms.

Another difficulty in intrusion detection is false positive (FP) classification. With this classification, the system may be misguided, and abnormal situations may not be recognized or normal situations may be evaluated as abnormal. When the precision performance of the algorithms in anomaly detection is compared, it is seen that the random forest algorithm shows the highest performance in all ratios.

The f1 score, which is a combined measure of sensitivity and precision scores, was used to examine the balance between the two. When the f1 scores of the algorithms in anomaly detection are compared, it is seen that the random forest algorithm shows the highest performance in all ratios.

As a result, it is understood that the random forest algorithm performs better than the other algorithms in the evaluations made for the smart home system energy consumption data set CSGEC-23.

In the future, it is aimed to realize the smart home system in hardware with IoT devices such as Raspberry Pi and NodeMCU. Furthermore, it is planned to develop the architecture by conducting detailed studies on the software-defined network structure of the proposed architecture. Specifically, future work will involve testing the architecture with different SDN controllers such as Floodlight, POX, and Ryu to evaluate their performance and impact on anomaly detection and energy management. The results obtained from these experiments will be compared and discussed to identify the most suitable controller for enhancing security and operational efficiency in SDN-IoT integrated smart grids. Additionally, to increase the traceability and cybersecurity of the smart home system, it is aimed to develop a hybrid artificial intelligence model suitable for SDN and IoT requirements that can detect anomalies in the network by working integrated with the proposed architecture.

References

- [1] Özçelik, İbrahim, et al. Center energy: A secure testbed infrastructure proposal for electricity power grid. In: 2021 International Conference on Information Security and Cryptology (ISCTURKEY). IEEE, 2021. p. 149-154.

- [2] Rehmani, Mubashir Husain, et al. Software defined networks-based smart grid communication: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 2019, 21.3: 2637-2670.
- [3] Demirci, Sedef; SAGIROGLU, Seref. Software-defined networking for improving security in smart grid systems. In: 2018 7th International Conference on Renewable Energy Research and Applications (ICRERA). IEEE, 2018. p. 1021-1026.
- [4] Soares, Arthur AZ, et al. 3AS: Authentication, authorization, and accountability for sdn-based smart grids. *IEEE Access*, 2021, 9: 88621-88640.
- [5] Jung, Oliver, et al. Anomaly Detection in Smart Grids based on Software Defined Networks. In: SMARTGREENS. 2019. p. 157-164.
- [6] Dileep, G. J. R. E. A survey on smart grid technologies and applications. *Renewable energy*, 2020, 146: 2589-2625.
- [7] Al-Fuqaha, Ala, et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE communications surveys & tutorials*, 2015, 17.4: 2347-2376.
- [8] Roman, Rodrigo; NAJERA, Pablo; LOPEZ, Javier. Securing the internet of things. *Computer*, 2011, 44.9: 51-58.
- [9] Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15), 2787-2805.
- [10] Wang, Minxiao; YANG, Ning; WENG, Ning. Securing a Smart Home with a Transformer-Based IoT Intrusion Detection System. *Electronics*, 2023, 12.9: 2100.
- [11] Alonazi, Wesam Abdulrhman, et al. SDN Architecture for Smart Homes Security with Machine Learning and Deep Learning. *International Journal of Advanced Computer Science and Applications*, 2022, 13.10.
- [12] Chen, Jian, et al. A multi-layer security scheme for mitigating smart grid vulnerability against faults and cyber-attacks. *Applied Sciences*, 2021, 11.21: 9972.
- [13] NIST (2018, 8 November). Update of the NIST Smart Grid Conceptual Model.
- [14] Marikyan, Davit; PAPAGIANNIDIS, Savvas; ALAMANOS, Eleftherios. A systematic review of the smart home literature: A user perspective. *Technological Forecasting and Social Change*, 2019, 138: 139-154.
- [15] Zaidan, A. A.; ZAIDAN, B. B. A review on intelligent process for smart home applications based on IoT: coherent taxonomy, motivation, open challenges, and recommendations. *Artificial Intelligence Review*, 2020, 53.1: 141-165.
- [16] Rondon, Luis Puche, et al. Survey on enterprise Internet-of-Things systems (E-IoT): A security perspective. *Ad Hoc Networks*, 2022, 125: 102728.
- [17] Ravinder, M.; KULKARNI, Vikram. Intrusion detection in smart meters data using machine learning algorithms: A research report. *Frontiers in Energy Research*, 2023, 11: 1147431.
- [18] Cao, Keyan, et al. An overview on edge computing research. *IEEE access*, 2020, 8: 85714-85728.
- [19] Danbatta, Salim Jibrin; VAROL, Asaf. Comparison of Zigbee, Z-Wave, Wi-Fi, and bluetooth wireless technologies used in home automation. In: 2019 7th International Symposium on Digital Forensics and Security (ISDFS). IEEE, 2019. p. 1-5.
- [20] Moustafa, Nour; SLAY, Jill. UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In: 2015 military communications and information systems conference (MilCIS). IEEE, 2015. p. 1-6.
- [21] Ring, Markus, et al. Flow-based benchmark data sets for intrusion detection. In: Proceedings of the 16th European conference on cyber warfare and security. ACPI. 2017. p. 361-369.
- [22] Sharafaldin, Iman; LASHKARI, Arash Habibi; GHORBANI, Ali A. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 2018, 1: 108-116.
- [23] MOUSTAFA, Nour. New generations of internet of things datasets for cybersecurity applications based machine learning: TON_IoT datasets. In: Proceedings of the eResearch Australasia Conference, Brisbane, Australia. 2019. p. 21-25.
- [24] NSL-KDD dataset. <https://www.unb.ca/cic/datasets/nsl.html>
- [25] FAN, Cheng, et al. A review on data preprocessing techniques toward efficient and reliable knowledge discovery from building operational data. *Frontiers in Energy Research*, 2021, 9: 652801.
- [26] YU, Xinran; ERGAN, Semiha; DEDEMEN, Gokmen. A data-driven approach to extract operational signatures of HVAC systems and analyze impact on electricity consumption. *Applied Energy*, 2019, 253: 113497.

- [27] FAN, Cheng; XIAO, Fu; YAN, Chengchu. A framework for knowledge discovery in massive building automation data and its application in building diagnostics. *Automation in Construction*, 2015, 50: 81-90.
- [28] FAN, Cheng, et al. Temporal knowledge discovery in big BAS data for building energy management. *Energy and Buildings*, 2015, 109: 75-89.
- [29] HASAN, Mahmudul, et al. Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. *Internet of Things*, 2019, 7: 100059.
- [30] GANGWAR, Amit Kumar; SHAIK, Abdul Gafoor. k-Nearest neighbour based approach for the protection of distribution network with renewable energy integration. *Electric Power Systems Research*, 2023, 220: 109301.
- [31] ROSE, Thomas, et al. A hybrid anomaly-based intrusion detection system to improve time complexity in the Internet of Energy environment. *Journal of Parallel and Distributed Computing*, 2020, 145: 124-139.
- [32] SHABAD, Prem Kumar Reddy; ALRASHIDE, Abdulmueen; MOHAMMED, Osama. Anomaly detection in smart grids using machine learning. In: *IECON 2021–47th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2021. p. 1-8.
- [33] LI, Qiang, et al. Simultaneous detection for multiple anomaly data in internet of energy based on random forest. *Applied Soft Computing*, 2023, 134: 109993.
- [34] VIGOYA, Laura, et al. IoT Dataset Validation Using Machine Learning Techniques for Traffic Anomaly Detection. *Electronics*, 2021, 10.22: 2857.
- [35] ARIBISALA, Adedayo; KHAN, Mohammad S.; HUSARI, Ghaith. Feed-Forward Intrusion Detection and Classification on a Smart Grid Network. In: *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE, 2022. p. 0099-0105.
- [36] CHUANG, Po-Jen; LI, Si-Han. Network intrusion detection using hybrid machine learning. In: *2019 International Conference on Fuzzy Theory and Its Applications (iFUZZY)*. IEEE, 2019. p. 1-5.
- [37] SHI, Jibo, et al. A hybrid intrusion detection system based on machine learning under differential privacy protection. In: *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*. IEEE, 2021. p. 1-6.
- [38] Balta, D. D., Kaç, S. B., Balta, M., Oğur, N. B., & Eken, S. (2025). Cybersecurity-aware log management system for critical water infrastructures. *Applied Soft Computing*, 169, 112613.
- [39] Breviglieri, P., Erdem, T., & Eken, S. (2021). Predicting smart grid stability with optimized deep models. *SN Computer Science*, 2, 1-12.
- [40] Singh, C., & Jain, A. K. (2024). A comprehensive survey on DDoS attacks detection & mitigation in SDN-IoT network. *e-Prime-Advances in Electrical Engineering, Electronics and Energy*, 100543.
- [41] Chaganti, R., Suliman, W., Ravi, V., & Dua, A. (2023). Deep learning approach for SDN-enabled intrusion detection system in IoT networks. *Information*, 14(1), 41.

Article Information Form

Authors Contributions

Hilal Yıldız: writing and editing, Musa Balta: conceptualization (lead), original draft (lead)

Acknowledgments

This work was supported by Sakarya University Scientific Research Projects Unit under Grants 2022-6-23-68.

Conflict of Interest Notice

The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical Approval and Informed Consent

It is declared that during the preparation process of this study, scientific and ethical principles were followed, and all the studies benefited from are stated in the bibliography.

Artificial Intelligence Statement

The authors confirm that no artificial intelligence tools were employed in the preparation or authorship of this article.

Availability of data and material

Not applicable / or link