INTERNATIONAL JOURNAL OF INTEGRATED ENGINEERING
ISSN: 2229-838X e-ISSN: 2600-7916

IJIE

Vol. 16 No. 9 (2024) 410-422
https://publisher.uthm.edu.my/ojs/index.php/ijie

AHEACS: A Streaming Dataset for Automatic Home Electrical Appliance Control Systems

Mochamad Susantok[1,2], Farhana Ahmad Po'ad[1]*, Endah Chilfani[3]

[1] *Faculty of Electrical and Electronic Engineering*
University Tun Hussein Onn Malaysia Parit Raja Johor, 86400, MALAYSIA
[2] *Electronics Telecommunication Engineering*
Politeknik Caltex Riau, Pekanbaru, 28265, INDONESIA
[3] *Applied Master of Computer Engineering*
Politeknik Caltex Riau, Pekanbaru, 28265, INDONESIA

*Corresponding Author: farhana@uthm.edu.my
https://doi.org/10.30880/ijie.2024.16.09.033

Article Info

Received: 19 July 2024
Accepted: 24 October 2024
Available online: 30 December 2024

Keywords

Streaming dataset, sensor hub, MQTT, home automation, IoT, Raspberry Pi, machine learning

Abstract

Introducing a real-time dataset designed for Automatic Home Electrical Appliance Control Systems (AHEACS). Unlike existing datasets with fixed time periods and static nature, AHEACS continuously collects data from an IoT smart home application. This paper presents a method namely Sensor Hub for creating real-time datasets with uses leveraging Message Queuing Telemetry Transport (MQTT) as the communication protocol. AHEACS is constructed and subsequently evaluated by using various machine learning algorithms. The dataset encompasses data from diverse sensors, including smart sockets linked to home appliances, a smart power meter, temperature and humidity sensors, as weel as motion detectors. AHEACS captures appliance ON/OFF status, total household power consumption, and environmental parameters like temperature, humidity, and occupancy. This research explores correlations between variables, particularly the air conditioner's ON/OFF state and other factors. A home automation model for air conditioners is developed based on household power consumption patterns and environmental conditions. The dataset is recorded minutely over a week, accumulating 20,403 data points or equivalent to approximately 2 weeks and 4 hours, stored in .csv format on a Raspberry Pi 4 edge device. Python in Google Colab is employed for data analysis, including Exploratory Data Analysis (EDA) and six machine learning algorithms: Artifical Neural Network (ANN), Decission Tree (DT), Random Forest (RF), Super Vector Machine (SVM), Linier Regression (LR), and Extreme Gradient Boosting (XGBoost). A case study on the air conditioner's ON/OFF status reveals strong correlations with Total Power (TP) and Presence Bedroom (Pb) variables. ANN emerges as the top-performing algorithm, achieving 86.8% accuracy, 0.097 Mean Square Error (MSE), 0.312 Root Mean Square Error (RMSE), and 0.607 R^2.

1. Introduction

The burgeoning Internet of Things (IoT) has transformed homes into intelligent environments capable of monitoring [1], controlling [2], and optimizing energy consumption [3]. While previous research has focused on static datasets for tasks like energy prediction, conservation, and occupancy detection [4-8], the dynamic nature of human behavior necessitates a real-time approach.

The control function in smart home applications can be classified into two categories: deterministic and natural. The advancement of Artificial Intelligence (AI) technology has made natural control an intriguing area of development, involving control based on datasets, also known as machine learning technology. Datasets from sensors installed in homes serve as knowledge for machine learning, a subset of AI, to create natural control functions. The research strongly focuses on behavior, examining the habits of home occupants in using electrical appliances influenced by various factors such as time, total power consumption in the home, temperature and humidity levels, and the presence of someone in the room, forming the characteristics of the proposed dataset in this study.

A behavior-based dataset can be accessed publicly for use in other research, datasets like KDD98 for IDS systems [9], the London Smart Meter dataset for power consumption over 47 months [10], and several electrical loads reviewed by Luca Tari [11]. However, these datasets are static and do not reflect the dynamic nature of behavior that adapts to the latest environmental conditions. For data scientists, acquiring consistently fresh data for further analysis, commonly known as data-driven techniques [12], is fascinating as it enables gaining new insights and improving model experiences [13]. In the context of IoT, Big Data technology plays a crucial role in processing and analyzing large volumes of data, such as the data generated by IoT devices. Its key features are represented by the 4Vs: volume, variety, velocity, and veracity, which align with the characteristics of data sourced from IoT devices. The challenge lies in collecting, processing, analyzing, and visualizing datasets [14].

Creating a real-time dataset, a component of Big Data, poses a challenge in both IoT and AI research. Addressing issues related to resources and AI's capacity to process real-time data, especially considering its large volume, presents ongoing challenges in handling sensor data. Real-time datasets for profiling household electrical usage can aid in machine learning processes for automatic control, aimed at optimizing electricity consumption in households and buildings at large [3], [15]. Profiling household electrical usage by monitoring the ON/OFF status of electrical appliances during their operation proves highly accurate and is widely employed by researchers as a means of constructing learning datasets for machine learning [5], [16], [17]. However, few studies account for additional factors like the rationale behind occupants turning appliances ON and OFF, environmental variables such as room temperature, humidity, and light intensity [16], as well as energy-related factors like water tank availability [6]. Incorporating these additional variables into the dataset can enhance control accuracy beyond habitual aspects and offer fresh insights for decision making.

This research introduces AHEACS, a novel real-time dataset designed for automatic home electrical appliance control systems. Unlike existing datasets, AHEACS captures the intricate interplay between human behavior, appliance usage, and environmental conditions. By incorporating factors such as temperature, humidity, and occupancy, AHEACS offers a more comprehensive understanding of household energy consumption patterns. This implies that different occupants will generate distinct AHEACS datasets. In this proposed dataset, each electrical appliance is connected to a smart plug socket that measures power usage and detects the ON/OFF status of the appliance. Additionally, the AHEACS dataset collects data from other sensors besides timestamp (TS) and total power consumption (TP). These include variables for automatic control determination, such as room temperature (T), room humidity (H), and presence of individuals in the room (P)

To facilitate the creation of AHEACS, we propose a simplified data collection method called Sensor Hub. This method efficiently aggregates data from various sensors and stores it in a .csv format for subsequent analysis. By streamlining the data collection process, Sensor Hub enables researchers to focus on higher-level data analysis and modeling.

The primary contribution of this research is the development of AHEACS, a novel real-time dataset capturing the intricate interplay between human behavior, appliance usage, and environmental conditions. AHEACS significantly advances machine learning-based control systems by providing a rich data foundation for developing more accurate and adaptive models for energy optimization, comfort control, and other smart home applications.

The remainder of this paper is structured as follows. Section 2 details the IoT network infrastructure, including sensor communication and the Sensor Hub algorithm, culminating in the creation of the AHEACS streaming dataset. Section 3 evaluates the dataset's utility for predicting appliance usage through the application of various machine learning algorithms. Finally, Section 4 presents the conclusions drawn from this work.

Nomenclature	
TS	Time series
TP	total power consumption

Int. Journal of Integrated Engineering Vol. 16 No. 9 (2024) p. 410-422

S-X	state on/off of electrical appliance X, S-AC for Air Conditioner, S-WP for Water Pump, S-TV for Television, S-WM for Washer Machine
T	temperature of room
H	humidity of room
P	presence of someone in the room Pb for bedroom, Pg for guestroom
MQTT	Message Queuing Telemetry Transport
ANN	Artificial Neural Network algorithm
DT	Decision Tree algorithm
RF	Random Forest algorithm
LR	Linier Regression algorithm
SVM	Super Vector Machine algorithm
XGBoost	Extreme Gradient Boosting
EDA	Exploratory Data Analysis

2. System Framework

To construct a real-time dataset, as proposed in this research, namely the AHEACS streaming dataset, it is imperative to first develop IoT devices equipped with sensors capable of direct communication with edge devices within the IoT network [18]. Furthermore, it is essential for the edge devices to be located as close as possible to the IoT devices, preferably within the same LAN/WLAN network segment, to minimize delay issues. Figure 1 illustrates the topography of the system framework designed to create the AHEACS streaming dataset. Here, all sensors and the edge server, acting as an edge device responsible for computational processes, are situated within the WLAN network segment.

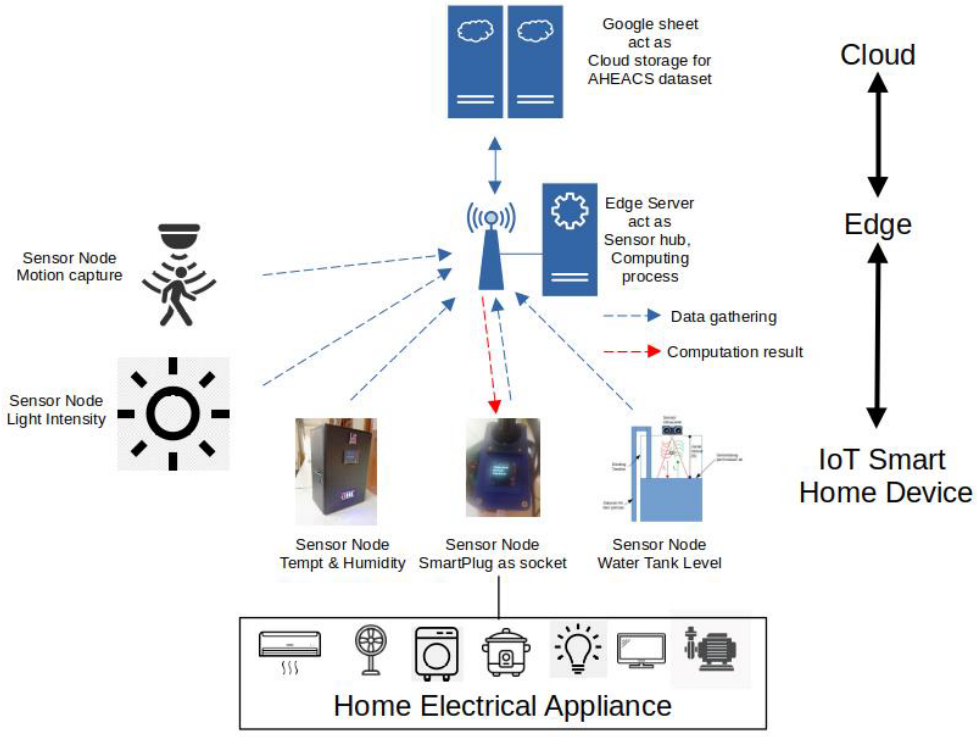

Fig. 1 *Smart home automation system framework*

In the context of the proposed smart home automation system framework, it can seamlessly connect as a plug-and-play device to the existing smart home network, serving as a wireless client from the existing WLAN. Both the edge server and IoT devices act as wireless clients connected to a wireless router to facilitate communication. Internet access, such as cloud connectivity, serves solely as an option for storing data in spreadsheets for monitoring purposes. In other words, the AHEACS streaming dataset can be constructed locally and stored on the

edge server without requiring Internet connectivity. As seen in Figure 1, there are two types of communication among edge devices: data gathering, indicated by blue dashed arrows, and computation result, indicated by red dashed arrows. In data gathering communication, it involves the real-time collection of data from sensors or IoT devices to the edge server as a sensor hub, storing it in the form of a dataset. On the other hand, computation result communication is the process of transmitting signal information from the edge server to the smart plug, where the information contains either the value 1 or 0 as the output of the machine learning computation process based on the formed dataset.

2.1 Wi-Fi Sensor Node Design

The Wi-Fi sensor node, functioning as a data generator, is meticulously designed to produce clean data, eliminating the need for subsequent data pre-processing stages. Therefore, the reliability of data acquisition becomes pivotal in the hardware design phase of the sensor. All sensors within the IoT device are crafted using the NodeMCU ESP8266, equipped with a built-in Wi-Fi and is called Wi-Fi sensor node, facilitating seamless connection to the existing WLAN. As depicted in Figure 2, the schematic diagram outlines each sensor component, including the smart plug, temperature and humidity sensor, and motion detection sensor, each of which will be elaborated on in the following details.

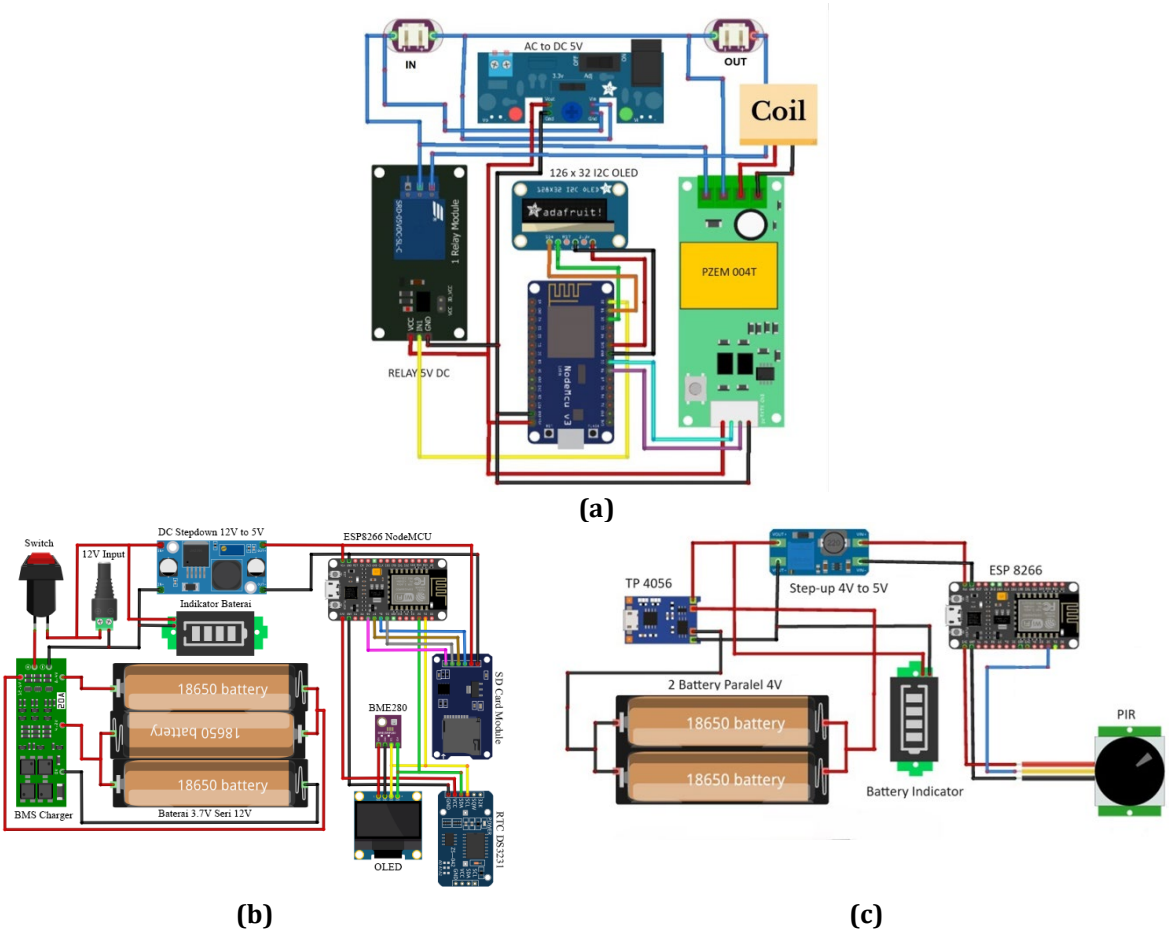

Fig. 2 *Wi-Fi sensor node schematic circuit: (a) Smart plug; (b) Temp & humidity; (c) Motion detector*

The smart plug in Figure 2(a) is an IoT device capable of measuring the power consumption of connected electrical appliances. It utilizes the PZEM-004T module, which can measure and display current, voltage, power, and power factor [19], along with a relay module for ON/OFF control functionality. In the AHEACS dataset system, the smart plug plays a crucial role in generating status data for connected electrical equipment, indicating whether they are in the ON or OFF state, normalized to values of 1 for "on" and 0 for "off." Additionally, real-time data on power consumption is recorded, referred to in the above nomenclature as TP and S-X.

The temperature and humidity sensor depicted in Figure 2.b utilizes the BME280 environmental sensor to observe indoor air conditions, encompassing temperature, humidity, and pressure [20]. The data generated by

Int. Journal of Integrated Engineering Vol. 16 No. 9 (2024) p. 410-422

this IoT device serves as decision-support information for residents to operate electrical appliances like air conditioners. In addition to being transmitted to the edge server, the data is also displayed on an OLED screen for immediate reading, functioning as a portable air quality monitor. Equipped with a rechargeable battery system, a battery indicator, and overcharging protection, ensuring reliable continuous operation while being mobile.

Figure 2.c features a motion sensor designed to detect the presence of individuals within a room. This IoT device incorporates a PIR sensor and a battery system, enabling portability and placement anywhere in a room. Consequently, activity data of residents within a room can be reliably transmitted to the edge server at any time. This activity information can serve as behavioural data for residents in operating electrical appliances.

2.2 Sensor Hub Algorithm

In the proposed sensor Hub system architecture Figure 3, each Wi-Fi sensor node acts as an MQTT client, creating unique topics with message content representing sensor values or specific conditions. For instance, the topic "aheacs/sp1/status" owned by a smart plug connected to an Air Conditioner conveys a "ON" message when turned on. The MQTT broker makes this topic and its value available to other subscribing MQTT clients. In this case, Node Red subscribing to the topic records "ON" or "OFF" in the dataset as 1 or 0, respectively. This value signifies the status of the Air Conditioner connected to the smart plug in the AHEACS dataset. The identical concept applies to the subsequent smart plug connected to various electrical appliances, including TV, water pump, refrigerator, and washing machine.

Similarly, calls iHTLog, another WiFi sensor node, broadcasts temperature and humidity information with topics "aheacs/ihtlog/temperature" and "aheacs/ihtlog/humidity". Node Red subscribing to these topics displays the data on a dashboard and stores it in the dataset. Another dataset variable in AHEACS is the total power consumption obtained from a smart wattmeter with the topic "aheacs/wattmeter/daya" and stored in the AHEACS dataset.

Overall, this system architecture can be easily implemented, where is Wi-Fi sensor node in the same WLAN with Raspberry Pi as edge server. Without needing an internet connection, an IoT network can be easily deployed in a home environment. Using the Node-RED dashboard, which displays all sensor data, it can be accessed locally on a smartphone through a browser at http://noderedipaddress:1880/ui, as illustrated in Figure 4. If Internet monitoring is required, ensure Node-RED is connected to the Internet via Wi-Fi. It will then send data to Google Sheets.

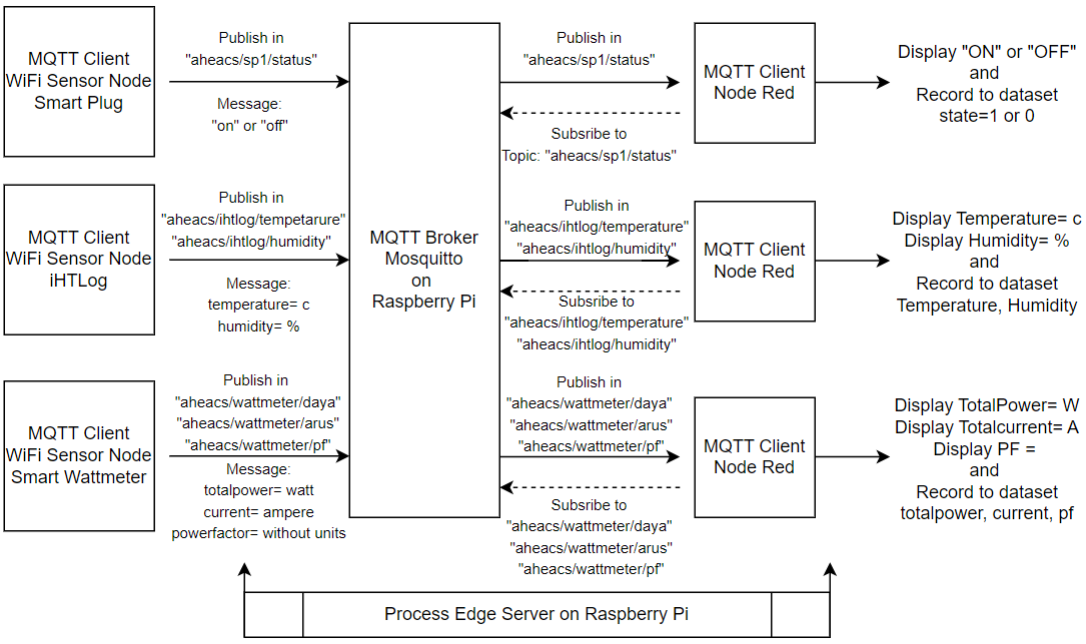

Fig. 3 *System architecture sensor hub*

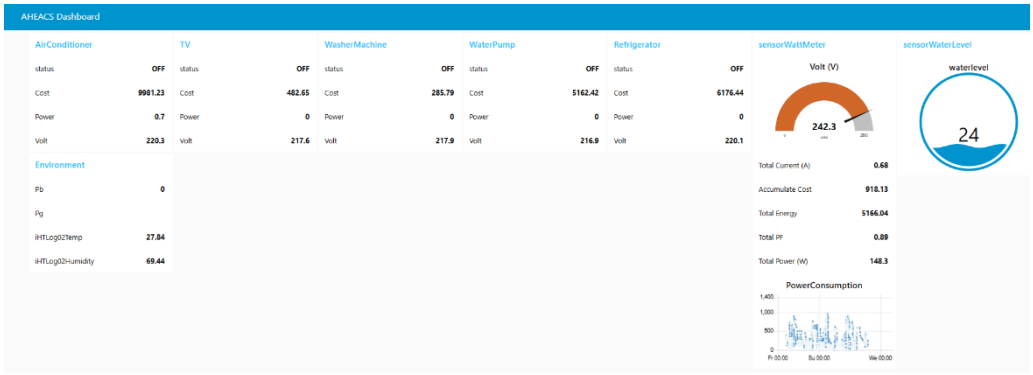

Fig. 4 *Node red dashboard for sensor hub*

The term "Hub" is inspired by the star topology in computer networks where the hub device serves as the central data communication point. In general, in IoT networks, each sensor independently sends its data directly to cloud storage, resulting in raw data in the context of Big Data that needs further processing before becoming a dataset. In the proposed Sensor Hub Algorithm, edge devices can gather all sensor data, then organize, and compile it into a unified real-time dataset. Node-RED, as a flow-based programming tool, takes on the role of Sensor Hub with functions such as aggregation, classification, and writing to a csv file, as illustrated in the Sensor Hub algorithm Figure 5 below. Node-RED operates on a 'flow' concept, connecting various node components. These nodes include input nodes for receiving data from sensors or messages via MQTT. Additionally, function nodes process data, while storage nodes store data temporarily or permanently.

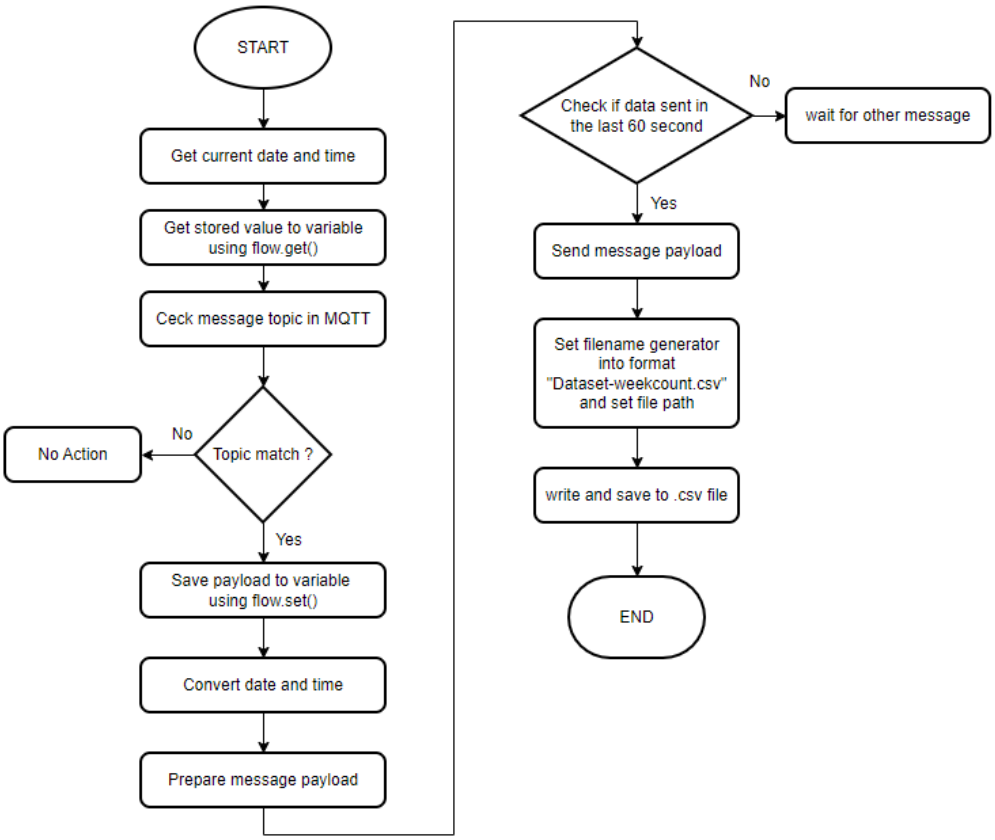

Fig. 5 *Sensor hub algorithm*

Int. Journal of Integrated Engineering Vol. 16 No. 9 (2024) p. 410-422

Figure 5 illustrates a sensor hub algorithm flowchart. It begins by capturing a timestamp using the Date() function within a function node, creating a time series aligned with household behaviors. Subsequently, the same function node retrieves sensor data from input nodes acting as MQTT clients for each sensor. The acquired data is stored in respective sensor data variables using flow.get().

Next, Node-RED monitors MQTT topics for incoming messages and updates variables accordingly. If the topic matches, the payload is extracted and stored using flow.set(). The timestamp from the Date() function is then converted into more granular components like hours, minutes, seconds, month, day, and year for detailed time-based analysis.

To accommodate potential delays in sensor data transmission, each payload is held for 60 seconds before aggregation. This serves as a buffer for varying sensor response times and defines a one-minute recording duration for each dataset. Each payload consists of multiple rows, each representing a minute's worth of data from all sensors, including the aggregated timestamp. Finally, the algorithm stores the data in a storage node using a CSV format named 'Dataset-weekcount.csv'. This weekly dataset allows for long-term data analysis and comparison.

2.3 AHEACS Streaming Dataset

The data acquisition for the AHEACS dataset is sourced from sensor nodes including power meters, smart plugs, and home environment sensors, with labels as described in Table 1. The dataset within AHEACS comprises variables such as timestamps provided by all Wi-Fi sensor nodes. The subsequent variables are 8 annotations, where TP represents the total power consumption at that moment. S-X indicates the on/off status of electrical appliances used by household occupants, with each electrical appliance being assigned an additional label X means the total number of electrical appliances controlled. Meanwhile, T, H, W, L, and P encompass supporting data for machine learning decisions, which also contribute to the automatic control model of electrical appliances based on household occupants' habits.

The labelling and annotation process in the AHEACS dataset revolves around S-X, specifically the “ON” or “OFF” label for electrical appliances. This label serves as the output for the machine learning algorithm, acting as a reference during testing to determine whether to turn the smart plug’s electrical appliance on or off. Data transmission from sensor nodes to the edge server occurs every minute, 24 hours a day and for a week. Learning data for machine learning is collected over a span of 7 days or per week. With these offered concepts, the AHEACS streaming dataset is highly dynamic, depending on household behaviour characteristics since each household has different behaviours in using electrical appliances. Therefore, for data scientists, gaining new insights from this AHEACS dataset would be straightforward. The core parameters include TS, TP, and S-X. Additionally, its nature is tailored to the needs regarding which factors influence residents' behaviour in operating their electrical appliances.

Table 1 *Data sensor label*

Variable	Description	Data type	Type of Sensor Node
TS	Timestamp Date and time	Date and time	All type
TP	Total Power (watt) consumption	Number	Power Meter / Smart Plug
S-X	ON/OFF electrical appliance state	Boolean	Smart plug
T	Temperature	Number	Air Quality
H	Humidity	Number	Air Quality
Pb	Bedroom activity	Boolean	PIR
Pg	Guestroom activity	Boolean	PIR

2.4 Machine Learning Algorithm

This study employs machine learning to evaluate the quality of the AHEACS dataset. We focus on classification tasks to predict the ON/OFF status of electrical appliances. Six widely used algorithms, including ANN, SVM, LR, DT, RF, and XGBoost, are implemented to develop classification models.

The dataset was evaluated using the machine learning algorithms mentioned above, following the block diagram depicted in Figure 6. The experiments were conducted on Google Colab utilizing TensorFlow, scikit-learn, pandas, NumPy, and Keras libraries for model building, training, and evaluation. The first block in the diagram (Figure 6) involves loading the AHEACS dataset after undergoing exploratory data analysis, which will be detailed in the results section. Subsequently, the dataset is split into an 80% training set and a 20% testing set.

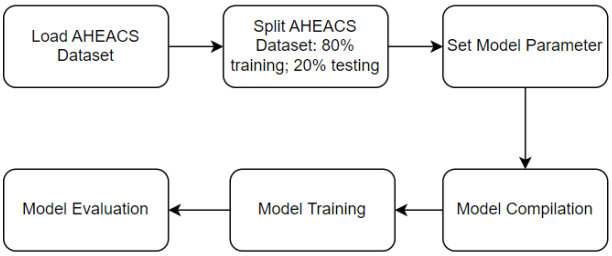

Fig. 6 *Block diagram dataset evaluation using machine learning*

The 'set model parameters' block configures the models as follows: The ANN employs a fully connected architecture with three layers: an input layer with 64 neurons, a hidden layer with 32 neurons, and an output layer with 1 neuron. The activation functions used are ReLU for the hidden layers and sigmoid for the output layer, which is typical for binary classification. SVM normalizes data using 'StandardScaler', calculating the mean and standard deviation of the training data and transforming each feature to have a mean of 0 and a standard deviation of 1, enhancing SVM performance. LR and DT share the same initialization with random_state=42 for reproducibility. RF and XGBoost use the same random_state but additionally have 100 estimators to improve model performance.

In the 'model compilation' block, the ANN is compiled using the Adam optimizer, binary crossentropy loss, and accuracy as the evaluation metric. This configuration optimizes the model for binary classification tasks. SVM utilizes the 'SVC' class from scikit-learn with a linear kernel while other models do not require explicit compilation. During the 'model training' block, ANN is trained for 10 epochs with a batch size of 32 using the 'fit' method. SVM, LR, DT, RF, and XGBoost are also trained using the 'fit' method to classify data into different classes. Finally, the 'model evaluation' block employs MSE, RMSE, and R-squared to assess the performance of all models. While MSE and RMSE are commonly used in regression to evaluate prediction errors, R-squared measures the proportion of variance explained by the model."

3. Result and Discussion

3.1 Exploratory Data Analysis

In this section, the process of Exploratory Data Analysis (EDA) is explained to understand the characteristics, patterns, and relationships within the AHEACS dataset. Table 2 displays the descriptive statistical analysis of the AHEACS dataset, consisting of 20,403 data points collected every minute over 7 days, with numerical data types except for the timestamp variable. This data presents the characteristics of residents operating electrical appliances such as Water Pump, AC, Washing Machine, and TV, with numerical values of 1 for turning ON and 0 for turning OFF. From this binary data, it is evident that AC and TV are more frequently in the ON state than OFF, with a mean of 0.54 for AC and 0.3 for TV, while other electrical appliances like water pump and washing machine are more often in the OFF state. The status data of these electrical appliances will be used by machine learning as the output label or target in the learning phase.

The total power consumption data of all electrical appliances used is indicated by the variable TP, with an average total power consumption of around 367.82 Watts, and significant variation with a standard deviation value of 241.35 Watts. The data ranges from a minimum of 38 Watts to a maximum of 1191.9 Watts, with the majority of data falling between 182.8 Watts and 555.03 Watts.

Furthermore, there are supporting data considered in decision-making for the machine learning process. First is the room temperature (T) with an average value of 31.3°C and small variation of 1.31. The room temperature mostly ranges from 30.8°C to 31.8°C. Another supporting factor is humidity (H) as indoor air quality data besides temperature, with an average value of around 76.7% and moderate variation with a standard deviation of 7.72. In general, room humidity is around 72% to 82%.

Additionally, there is supporting data such as Present bedroom (Pb) and Present guest room (Pg), which are data from motion sensors indicating the presence of residents in the bedroom and living room. These supporting data are further analyzed for their patterns in relation to the decision to turn ON or OFF AC and TV. From the data presented in Table 2, it is observed that residents are more frequently in the bedroom than in the living room, with an average value of 0.65 for Pb and 0.27 for Pg.

While the total power consumption (TP) data provides valuable insights into overall energy usage, the primary focus of this study is not on analyzing the specific power consumption of each electrical appliance. Instead, this research aims to explore the relationships between the operation of various appliances. For instance, we examine how turning the air conditioner (AC) ON or OFF may be related to the operation of other appliances, such as the water pump, TV, or washing machine. The total power consumption serves as an important variable

Int. Journal of Integrated Engineering Vol. 16 No. 9 (2024) p. 410-422

in machine learning models, helping to assess the impact of AC usage in situations where power availability is limited, rather than providing a detailed breakdown of power consumption per device.

Table 2 *Descriptive analysis dataset AHEACS*

	count	mean	std	min	25%	50%	75%	max
TP	20403	367.82	241.35	38	182.8	272.7	555.03	1191.9
T	20403	31.3	1.31	25.14	30.8	31.3	31.8	35.2
H	20403	76.7	7.72	51	72	79	82	90
Pb	20403	0.65	0.48	0	0	1	1	1
Pg	20403	0.27	0.44	0	0	0	1	1
S-WP	20403	0.12	0.33	0	0	0	0	1
S-AC	20403	0.54	0.5	0	0	0	1	1
S-WM	20403	0.02	0.13	0	0	0	0	1
S-TV	20403	0.3	0.46	0	0	0	1	1

Later, in the EDA process, the correlation between features or variables within the dataset is also examined to identify relationships among variables. Variables that have a strong relationship with the target variable can be considered highly relevant and worthy of inclusion in modeling as input variables. This is crucial to enhance computational performance, especially in terms of data dimensionality, including avoiding overfitting and expediting machine learning processes.

Figure 7 displays the correlation coefficient between variables in the dataset in the form of a heatmap. Each cell in the heatmap represents the correlation between two variables, with the color indicating the strength and direction of the correlation. Darker shades indicate a strong negative correlation, while brighter shades indicate a strong positive correlation. The correlation coefficient used is Pearson, calculated using this formula [21]:

$$r_{xy} = \frac{\sum(x-\bar{x})(y-\bar{y})}{N.S_xS_y} \quad (1)$$

r_{xy} is the Pearson correlation coefficient for two variables x and y, Meanwhile N represent the total number of data samples and S_xS_y represent the standard deviation from two variable. The value of r_{xy} ranges from -1 (dark color) to 1 (bright color), while 0 signifies no correlation between the two variables. All diagonal cells have a value of 1, representing the correlation of each feature with itself (perfect positive correlation).

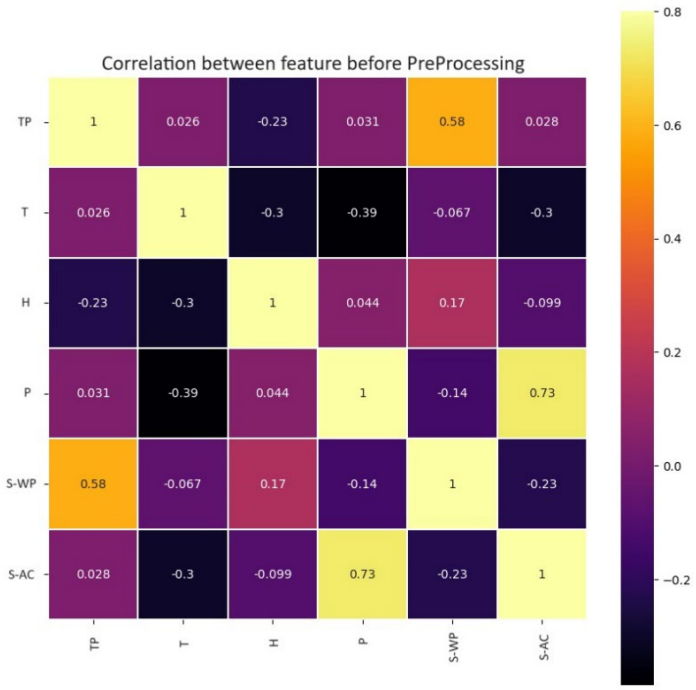

Fig. 7 *Correlation between variables*

For example, S-AC has a very strong positive correlation with P, which is 0.73. This means that when the Air Conditioner is ON, there is detected movement in the bedroom, in other words, there is someone in the bedroom. On the other hand, the variable TP has a very small impact on the operation of the Air Conditioner, which is 0.028. This means that the use of other electrical appliances has minimal impact on the air conditioner. Other factors such as T and H, as well as S-WP, also influence S-AC negatively, with values of -0.3, -0.099, and -0.23, respectively. Identifying variables that have a strong correlation with the target variable is the main goal of the EDA process, and they will be designated as input parameters in the machine learning process.

In this work, we present a case study where S-AC is used as the target variable. This selection is based on the data in Table 2, which shows that the AC is used more frequently than the TV compared to other electrical appliances. Additionally, the AC generally consumes more power than the TV. The summary of correlation values for all variables, including TS (Timestamp) that has been transformed per unit of time, against the target variable S-AC is shown in Table 3 below. Next, prepare the AHEACS dataset with nine input variables: TP, T, H, Pb, S-WP Day, Hour, Minute, and Second and one target variable, which is S-AC.

Table 3 *Summary of correlation coefficient to S-AC*

Variable	TP	T	H	Pb	S-WP	Day	Hour	Minute	Second
S-AC (Target)	0.028	-0.3	-0.099	0.73	-0.23	-0.064	-0.035	0.0051	0.014
Category	Positive	Negative	Negative	Positive	Negative	Negative	Negative	Positive	Positive

3.2 Evaluation Using Machine Learning

After the AHEACS dataset has been prepared with input and target variables, the process of evaluating the dataset using machine learning can be carried out as described in the previous subsection and illustrated in Figure 6. The selection of this algorithm is justified by the time-series characteristics of the AHEACS dataset, which captures variations in household behavior, total power consumption, and environmental conditions over time. This approach has also been used in previous studies [22][23]. Third, model performance is evaluated using performance evaluation metrics, including MSE, RMSE, and R^2, with the following equations.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_t - \hat{y}_t)^2 \quad (2)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_t - \hat{y}_t)^2} \quad (3)$$

$$R^2 = \frac{\sum_{i=1}^{N}(y_t - \hat{y}_t)^2}{\sum_{i=1}^{N}(y_t - \bar{y}_t)^2} \quad (4)$$

The test scenario for the AHEACS dataset in this study involves setting a classification objective. As detailed in the correlation explanation, S-AC was chosen as the target variable for evaluating the accuracy of the machine learning model. The objective is to predict whether the S-AC variable falls into class 1 or class 0, where class 1 indicates that the AC switch is turned on, and class 0 indicates that it is turned off. The input variables are selected based on their correlation with S-AC in the dataset.

Through calculations using Equation 1, as shown in Figure 7 and Table 3, the input variables are determined to have values between -1 > r_{xy} < 1. These variables include Total Power (TP), Temperature (T), Humidity (H), Present in bedroom (Pb), State ON/OFF of Water Pump (S-WP) and Time Series (TS) which encompasses Hour, Minutes, and Second. Testing with the Artificial Neural Network (ANN) model yielded a loss of 0.33, accuracy of 86.8%, and evaluation metrics: MSE=0.097, RMSE=0.312, and R^2=0.607. On the other hand, Support Vector Machine (SVM), using NumPy and pandas for data manipulation, SVM resulted in a very low accuracy of 5.2%, with MSE=786.55, RMSE=28.05, and R^2=-1.305. Logistic Regression (LR) also performed poorly, achieving an accuracy of 4.8%, with MSE=796.7, RMSE=28.23, and R^2=-1.335.

Further testing with The Random Forest (RF) algorithm resulted in an accuracy of 41.6% with evaluation metrics of MSE=130.9, RMSE=11.44, and R^2=0.62. Extreme Gradient Boosting (XGBoost) achieve accuracy of 32.24%, with MSE=253.5, RMSE=15.92, and R^2=0.257. Lastly, the Decision Tree (DT) model produces an accuracy of 40.24%, with MSE=174.01, RMSE=13.19, and R^2=0.49. The result indicates varying levels of performance across different algorithms.

The summary of accuracy from the six models on the streaming dataset AHEACS is presented in Figure 8, where ANN shows the highest accuracy at 86.8%. Although some algorithms, such as DT and RF, which are generally suitable for binary classification problems, achieve less satisfactory accuracy, i.e., below 50%. Evaluation

Int. Journal of Integrated Engineering Vol. 16 No. 9 (2024) p. 410-422

metrics, summarized using formulas (2), (3), and (4), are compared in Table 4. The table shows that RF has the highest R^2 value of 0.62, followed closely by ANN with a value of 0.607. The smallest MSE and RMSE values indicate higher accuracy in predicting S-AC data. Among the models, ANN stands out as the most accurate, with an accuracy of 86.8%, whereas other models fall below 50%. Therefore, the ANN model is highly suitable for testing the streaming dataset AHEACS compared to the other five models.

Table 4 *Evaluation metric comparation between models for AHEACS dataset*

Models and Evaluation Metric	ANN	SVM	LR	RF	XGBoost	DT
MSE	0.097	786.55	796.7	130.9	253.5	174.01
RMSE	0.312	28.05	28.23	11.44	15.92	13.19
R^2	0.607	-1.305	-1.335	0.62	0.257	0.49

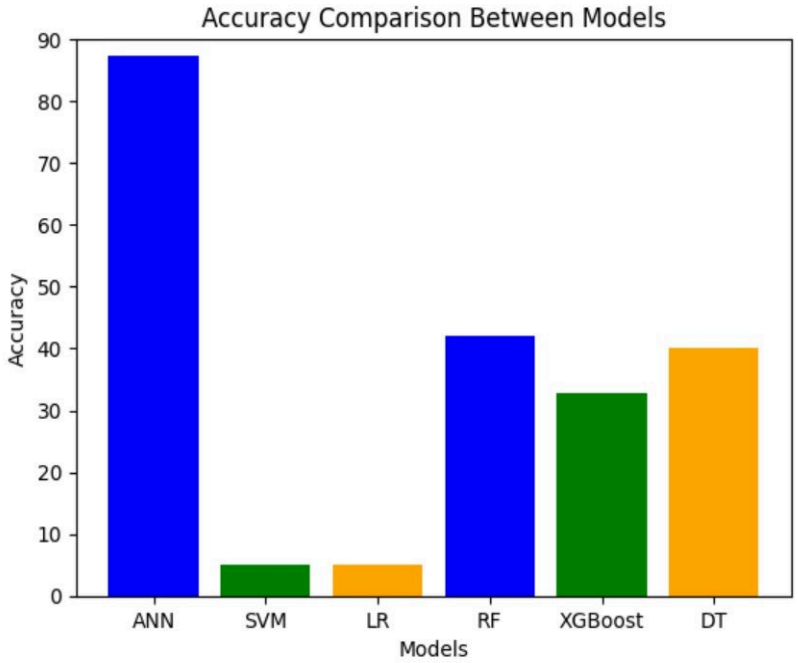

Fig. 8 *Accuracy comparation between models on the AHEACS dataset*

4. Conclusion

The availability of real-time datasets is vital for data analysis requiring swift adaptation to dynamic conditions, particularly in smart home environments where rapid data processing enables automated appliance control based on household behaviors. This work contributes to the design and development of the AHEACS streaming dataset, a novel real-time data stream leveraging MQTT based streaming capabilities. This dynamic dataset captures comprehensive information on total power consumption, appliance status, indoor activities, and environmental factors such as temperature and humidity. By overcoming the limitations of static datasets, AHEACS based on household behavior provides a comprehensive and high-quality resource for machine learning applications in smart home automation. Six machine learning algorithms such as ANN, RF, LR, SVM, DT, and XGBoost were employed to evaluate the AHEACS dataset, using metrics such as accuracy, MSE, RMSE, and R^2. Among them, the ANN model demonstrated superior performance, achieving an accuracy of 86.8%, MSE of 0.097, RMSE of 0.312, and R^2 of 0.607, indicating its potential for further in-depth analysis and practical applications with the AHEACS dataset. Future research can leverage the AHEACS dataset to develop more sophisticated predictive models by integrating factors like seasonal patterns, weather conditions, and interactions between appliances. Advanced techniques of ANN, such as deep learning, could further enhance model performance. Additionally, optimizing energy consumption in smart homes by integrating predictive models with control algorithms presents a promising avenue for creating intelligent and efficient smart home systems. The AHEACS dataset thus lays a strong foundation for advancing the field of smart home technology.

Acknowledgement

This research is partially funded by the Research and Community Service Department of Politeknik Caltex Riau and is fully supported by the Faculty of Electrical Engineering and Electronics (FKEE) at UTHM.

Conflict of Interest

The are no conflicts of interest with respect to the publication of this manuscript.

Author Contribution

The authors confirm contribution to the paper as follows: ***study conception and design:*** *Author 1, Author 2;* ***data collection:*** *Author 1;* ***analysis and interpretation of results:*** *Author 1, Author 3;* ***draft manuscript preparation:*** *Author 2. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] K. Luechaphonthara and A. Vijayalakshmi, "IOT based application for monitoring electricity power consumption in home appliances," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 6, pp. 4988–4992, 2019, https://doi.org/10.11591/ijece.v9i6.pp4988-4992.

[2] O. Munoz, A. Ruelas, P. Rosales, A. Acuña, A. Suastegui, and F. Lara, "Design and Development of an IoT Smart Meter with Load Control for Home Energy Management Systems," *Sensors*, vol. 22, no. 19, p. 7536, Oct. 2022, https://doi.org/10.3390/s22197536.

[3] R. Olu-Ajayi, H. Alaka, I. Sulaimon, F. Sunmola, and S. Ajayi, "Building energy consumption prediction for residential buildings using deep learning and other machine learning techniques," *Journal of Building Engineering*, vol. 45, Jan. 2022, https://doi.org/10.1016/j.jobe.2021.103406.

[4] M. A. Paredes-Valverde, G. Alor-Hernández, J. L. García-Alcaráz, M. del P. Salas-Zárate, L. O. Colombo-Mendoza, and J. L. Sánchez-Cervantes, "IntelliHome: An Internet of things-based system for electrical energy saving in smart home environment," *Comput Intell*, vol. 36, no. 1, pp. 203–224, Feb. 2020, https://doi.org/10.1111/coin.12252.

[5] A. Lentzas and D. Vrakas, "Machine learning approaches for non-intrusive home absence detection based on appliance electrical use," *Expert Syst Appl*, vol. 210, p. 118454, Dec. 2022, https://doi.org/10.1016/j.eswa.2022.118454.

[6] M. Susantok and T. Ramadhan, "Manajemen Ketersediaan dan Penggunaan Air pada Rumah Tangga Berbasis IoT," *Jurnal Elektro dan Mesin Terapan*, vol. 7, no. 1, pp. 1–10, May 2021, https://doi.org/10.35143/ELEMENTER.V7I1.3743.

[7] O. Taiwo, A. E. Ezugwu, O. N. Oyelade, and M. S. Almutairi, "Enhanced Intelligent Smart Home Control and Security System Based on Deep Learning Model," *Wirel Commun Mob Comput*, vol. 2022, 2022, https://doi.org/10.1155/2022/9307961.

[8] T. K. Ghazali and N. H. Zakaria, "Security, comfort, healthcare, and energy saving: A review on biometric factors for smart home environment," *Journal of Computers (Taiwan)*, vol. 29, no. 1. Computer Society of the Republic of China, pp. 189–208, Feb. 01, 2018. https://doi.org/10.3966/199115992018012901017.

[9] Y. Al-Hadhrami and F. K. Hussain, "Real time dataset generation framework for intrusion detection systems in IoT," *Future Generation Computer Systems*, vol. 108, pp. 414–423, Jul. 2020, https://doi.org/10.1016/j.future.2020.02.051.

[10] M. M. Sachin, M. P. Baby, and A. S. Ponraj, "Analysis of energy consumption using RNN-LSTM and ARIMA Model," *J Phys Conf Ser*, vol. 1716, no. 1, p. 012048, Dec. 2020, https://doi.org/10.1088/1742-6596/1716/1/012048.

[11] T. Luca, G. Berrettoni, C. Bourelly, G. Cerro, D. Capriglione, and L. Ferrigno, "eLAMI-An Innovative Simulated Dataset of Electrical Loads for Advanced Smart Energy Applications," *IEEE Access*, vol. 10, pp. 91177–91191, 2022, https://doi.org/10.1109/ACCESS.2022.3201960.

[12] S. Imai, "Development of IoT technology using data-driven control," in *2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, Sep. 2021, pp. 1–4. https://doi.org/10.1109/ETFA45728.2021.9613557.

[13] H. Jahani, R. Jain, and D. Ivanov, "Data science and Big Data analytics: a systematic review of methodologies used in the supply chain and logistics research," *Ann Oper Res*, Jul. 2023, https://doi.org/10.1007/s10479-023-05390-7.

[14] S. Khare and M. Totaro, "Big Data in IoT," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, Jul. 2019, pp. 1–7. https://doi.org/10.1109/ICCCNT45670.2019.8944495.

[15] I. Machorro-Cano, G. Alor-Hernández, M. A. Paredes-Valverde, L. Rodríguez-Mazahua, J. L. Sánchez-Cervantes, and J. O. Olmedo-Aguirre, "HEMS-IoT: A Big Data and Machine Learning-Based Smart Home System for Energy Saving," *Energies (Basel)*, vol. 13, no. 5, p. 1097, Mar. 2020, https://doi.org/10.3390/en13051097.
[16] S. Zulaikha, A. Adria, and A. Rahman, "SISTEM OTOMASI LAMPU RUMAH ADAPTIF BERBASIS ARTIFICIAL NEURAL NETWORK," vol. 3, no. 2, p. 68, 2018, Accessed: Nov. 06, 2023. [Online]. Available: https://jurnal.usk.ac.id/kitektro/article/view/11394
[17] M. F. Ramdani, C. Setianingsih, and F. C. Hasibuan, "Sistem Kendali Alat Elektronik Berdasarkan Kebiasaan Pengguna Menggunakan Algoritma Backpropagation," Bandung, Oct. 2021. Accessed: Nov. 06, 2023. [Online]. Available: https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/16493
[18] H. Yar, A. S. Imran, Z. A. Khan, M. Sajjad, and Z. Kastrati, "Towards Smart Home Automation Using IoT-Enabled Edge-Computing Paradigm," *Sensors 2021, Vol. 21, Page 4932*, vol. 21, no. 14, p. 4932, Jul. 2021, https://doi.org/10.3390/S21144932.
[19] A. I. Satriananda, L. Kamelia, M. R. Efendi, and A. Kusnawan, "The Prototype of Smart Power Meter at Home Based on Internet of Things," in *2021 7th International Conference on Wireless and Telematics (ICWT)*, IEEE, Aug. 2021, pp. 1–3. https://doi.org/10.1109/ICWT52862.2021.9678480.
[20] G. V. Shevchenko, N. A. Glubokov, A. V. Yupashevsky, and A. S. Kazmina, "Air Flow Sensor Based on Environmental Sensor BME280," in *2020 21st International Conference of Young Specialists on Micro/Nanotechnologies and Electron Devices (EDM)*, IEEE, Jun. 2020, pp. 432–435. https://doi.org/10.1109/EDM49804.2020.9153474.
[21] A. Mukasine, L. Sibomana, K. Jayavel, K. Nkurikiyeyezu, and E. Hitimana, "Correlation Analysis Model of Environment Parameters Using IoT Framework in a Biogas Energy Generation Context," *Future Internet*, vol. 15, no. 8, p. 265, Aug. 2023, https://doi.org/10.3390/fi15080265.
[22] Y.-H. Lin, H.-S. Tang, T.-Y. Shen, and C.-H. Hsia, "A Smart Home Energy Management System Utilizing Neurocomputing-Based Time-Series Load Modeling and Forecasting Facilitated by Energy Decomposition for Smart Home Automation," *IEEE Access*, vol. 10, pp. 116747–116765, 2022, https://doi.org/10.1109/ACCESS.2022.3219068.
[23] Z. A. Jaaz, M. E. Rusli, N. A. Rahmat, I. Y. Khudhair, I. Al Barazanchi, and H. S. Mehdy, "A Review on Energy-Efficient Smart Home Load Forecasting Techniques," in *2021 8th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, IEEE, Oct. 2021, pp. 233–240. https://doi.org/10.23919/EECSI53397.2021.9624274.