Journal of Multidisciplinary Engineering Science Studies (JMESS)
ISSN: 2458-925X
Vol. 10 Issue 5, May - 2024

Development Of An Iot-Based Anomaly Detection In Smart Energy Meter

Emmanuel Samson Itodo
Electrical and Electronics Engineering Department, Federal University of Technology Akure, Ondo State Nigeria
esitodo@futa.edu.ng

***Abstract*—The deployment of smart energy meters in modern utility grids has revolutionized the monitoring and management of energy consumption. However, the vast amount of data generated by these meters presents challenges in identifying abnormal patterns indicative of energy theft, faulty equipment, or system inefficiencies. This paper proposes the development of an Internet of Things (IoT)--based anomaly detection system for smart energy meters to address these challenges. The proposed system leverages IoT technology to collect real-time energy consumption data from smart meters deployed across the utility grid. Advanced data analytics techniques, are employed to detect anomalies in energy consumption patterns. The system utilizes historical consumption data to establish baseline profiles for individual meters and identifies deviations from these profiles as potential anomalies. The effectiveness of the through real-world deployment scenarios. The developed IoT-based anomaly detection system for smart energy meters holds significant promise in enhancing the efficiency, reliability, and security of utility grid operations. By proactively identifying and addressing anomalies, utilities can improve energy management, reduce revenue losses, and enhance customer satisfaction in the era of smart grid technology.**

Keywords—Smart meter, Meter, Internet of things and energy

I. INTRODUC TION

Energy is one of the problems facing the Nigerian economy. Many industries and companies have relocated due to the insufficient supply of electrical energy in the country [1, 2, 3]. The stakeholders in the electricity industry in Nigeria are facing a lot of challenges in providing clean, efficient, and reliable power supply to the citizens [4]. The distribution companies that interface with the end user face many challenges, such as illegal connection, tampering of metering facilities, and non-payment of electricity bills, which reduces the revenue generated by the distribution companies. In recent years, the proliferation of smart technologies has revolutionized various sectors, including the energy industry [5, 6, 7]. Among these advancements, integrating Internet of Things (IoT) devices in energy metering systems has emerged as a promising avenue for enhancing efficiency, monitoring, and management of energy consumption [8]. However, in Nigeria for instance, this smart energy meter has not yet been explored by the distribution companies. Most of the prepaid meters adopted by the distribution companies in Nigeria are not smart since the end users' load cannot be monitored by the distribution company. Due to this, the tampering of the metering facilities cannot be detected by the distribution companies. Smart energy meters, equipped with IoT capabilities, provide real-time data collection and analysis, enabling utilities and consumers to make informed decisions regarding energy usage. However, with the increasing complexity and interconnectedness of IoT-enabled energy metering systems, the susceptibility to anomalies and cybersecurity threats has become a significant concern [9]. Anomalies such as tampering, meter malfunction, or unauthorized access can lead to inaccurate readings and billing discrepancies and even compromise the integrity of the entire energy infrastructure [10, 11]. Furthermore, the integration of Internet of Things (IoT) technologies in smart energy metering systems has ushered in a new era of efficiency and connectivity in the energy sector. However, with this advancement comes the challenge of ensuring the security and reliability of these systems. Anomalies such as tampering, meter malfunction, and unauthorized access pose significant threats to the integrity of energy data, billing accuracy, and overall system performance [11, 12]. Existing anomaly detection methods often lack the specificity and adaptability required to effectively identify and mitigate these anomalies in real-time [13]. Therefore, the primary objective of this study is to develop a robust IoT-based anomaly detection system tailored for smart energy meters. This system aims to address the following key challenges: Anomaly Identification: Designing algorithms capable of accurately detecting deviations from expected energy consumption patterns, including both intentional tampering and unintentional errors, implementing mechanisms for real-time monitoring and detection of anomalies to enable prompt intervention and mitigation measures, developing a scalable solution capable of accommodating the diverse and evolving nature of energy metering environments, including different meter types, communication protocols, and usage patterns. By addressing these challenges, this research aims to contribute to the enhancement of security, reliability, and efficiency in IoT-enabled smart energy metering systems, ultimately facilitating the transition towards a more sustainable and resilient energy infrastructure. This research seek to contribute to the ongoing efforts aimed at enhancing the security, resilience, and performance of IoT-enabled smart energy systems, ultimately fostering a more sustainable and resilient energy ecosystem for the benefit of society. By leveraging machine learning algorithms, data analytics, and IoT connectivity, these solutions aim to detect deviations from expected patterns in energy consumption and metering data, thereby enabling timely intervention and mitigation of potential risks.

II. METHODOLOGY

The circuit involves in the development of the IOT based anomalies detection for smart metering is presented in Fig. 1.

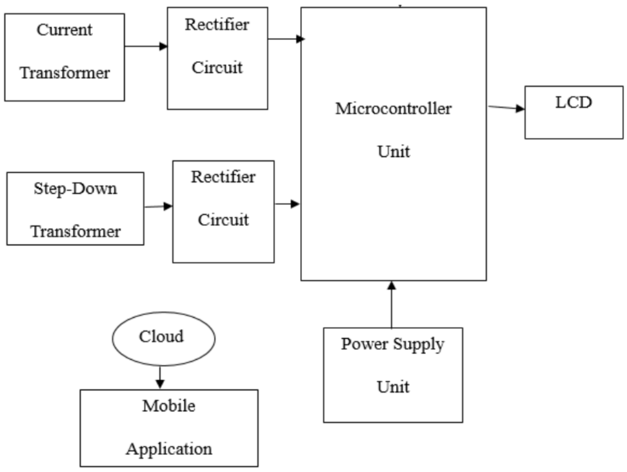

Fig. 1. Circuit Sketch

A. Hardware Simulation Process on Proteus

i) Setting up the ESP8266 Microcontroller: This involved searching and dropping the microcontroller into the workspace via the “Pick from Libraries”. The VCC and GND pins of the ESP8266 are then connected to the voltage regulator's output and ground, respectively. Additionally, the required GPIO is connected to the LCD

ii) Configuring the LCD Display: After adding the LCD module from the Proteus Library, the VCC and GND pins of the LCD are connected to the voltage regulator's output and ground. Then, the data pins (D4 to D7) and control pins (RS, RW, E) of the LCD are connected to the ESP8266's GPIO pins.

iii) The voltage regulator: After adding the voltage regulator (LM7805), the input pin is connected to a DC source representing the power supply and the output pin to the VCC and GND pins of the ESP8266 and other components.

iv) Rectification: A bridge rectifier is added and connected to the AC power source, simulating the conversion from AC to DC. Then resistors and capacitors are included for appropriate voltage division and filtering. The selection of the required resistors and capacitors also considered the ESP8266 and other component requirements.

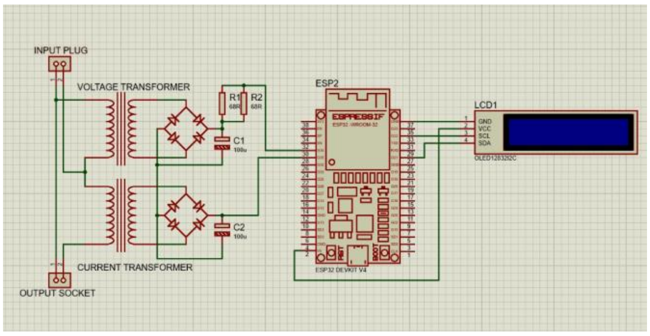

Fig. 2: Proteus Simulation of the Energy Monitoring System

B. Cloud Integration

i) **Firebase Configuration:** Although Proteus does not allow for the simulation of the actual cloud connection, the communication can be simulated by creating a representation of Firebase. A generic "Cloud" symbol is used to represent the cloud service. It is then connected to the ESP8266 to symbolize data transfer.

ii) **ESP8266 Programming:** This involved a number of things. These are the Arduino codes for the ESP8266 in the Arduino IDE, configuration of the ESP8266 to connect to Wi-Fi and Firebase, inclusion of the necessary libraries for ESP8266, Firebase, and any other peripherals. It also includes functions for data collection, processing, and communication with Firebase.

iii) **Code Upload:** The code upload stage is a crucial step in the development process, where the compiled and validated code for the ESP8266 is transferred to the microcontroller. This was done using the Arduino IDE.

iv) **Final Validation:** This involved the verification that ESP8266 successfully communicates with the simulated cloud representation (Firebase). It also includes confirming that the LCD displays the expected data and the overall system functions correctly.

C. Hardware Design Processes

The hardware design involved creating detailed specifications for the physical system's components. It involved defining the architecture, specifying how the components were interconnected, and creating detailed diagrams. Fig. 3 shows the flowchart that indicates how the system is expected to interact for operation. In the hardware design of our voltage and current monitoring system, we employed a combination of current and voltage transformers to accurately measure voltage and current. For actual cloud testing, we deployed the code at the Appendix to physical hardware, connected to the internet.

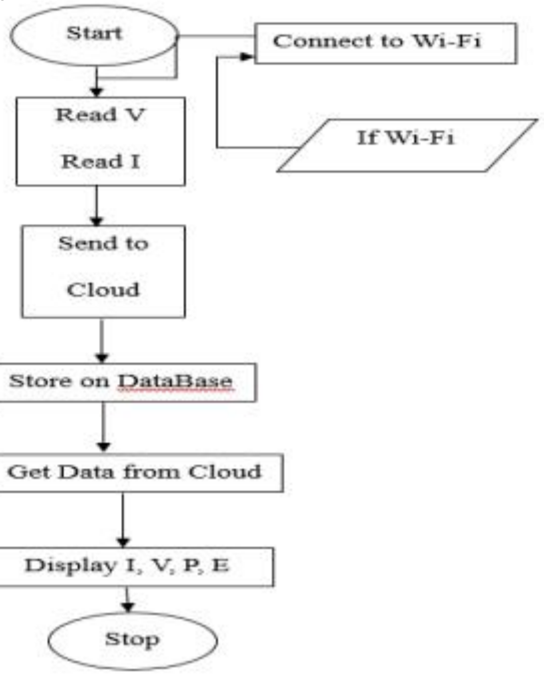

Fig. 3: Flowchart for System Processes

D. Voltage Reading Process

The voltage reading process involved a series of steps to ensure accurate and reliable measurements. It was initiated with bridge rectification to convert alternating current (AC) to direct current (DC). Subsequently, a filtering stage was implemented to smoothen the rectified signal. Following this, voltage division was employed to achieve a 5V output, which is then fed into the microcontroller unit (MCU) for processing.

E. Current Reading Process

The current reading process capitalized on voltage induction in an inductor, converting current from the current transformer into an electromotive force (EMF). This EMF undergoes rectification and filtering stages. Unlike the voltage signal, the current signal, with its typically lower value, is directly fed into the MCU without undergoing voltage division. Calibration is then applied to align the output voltage with the current from the transformer.

F. Alternating Current and Voltage Readings

Given the limitation of the ESP8266 with only one analog input (A0), a mechanism was established to alternate between reading voltage and current. A relay system was implemented, with the current reading passing through a normally open configuration and the voltage through a normally closed configuration. This strategic arrangement allows for continuous and comprehensive monitoring of both voltage and current.

General Signal Processing Operations

G. Bridge Rectification

The bridge rectification step was necessary for converting alternating current (AC) to direct current (DC) allowing for a unidirectional flow of current. It ensures the alternating voltage signal becomes suitable for further microcontroller-based electronic applications that require a consistent and polarized power supply. The rectifier comprised of four 1N4007 diodes arranged in a typical bridge configuration. The AC input was applied to the bridge, and the diodes selectively allowed the current to flow in one direction during different halves of the AC cycle. As a result, the negative and positive portions of the AC waveform are effectively separated, producing a pulsating DC signal. This rectified output serves as a foundation for the subsequent signal processing.

H. Filtering

Following the bridge rectification, two capacitors rated 100µf, 25v and 220µf, 35v were used to smoothen out the fluctuations of the pulsating DC signal that results from the bridge rectifier. This process helped stabilize the DC voltage, reduced unwanted noise and high-frequency components, ensuring a cleaner and more improved signal.

I. Voltage Division

Voltage division was then subsequently employed to reduce the magnitude of the voltage signal from 12v to 5v, making it suitable for processing by the microcontroller unit (MCU). In the context of the voltage reading process, involving the use of a resistor network with R1 and R2 as 2k Ohms and 1k Ohms, respectively, the voltage division process obtains a fraction of the original voltage that is prone to fluctuation. The resistor values are chosen to achieve a scaled-down voltage, typically the 5 volts that is readable by the MCU. This voltage becomes the output signal to be used for reliable voltage readings in the system.

J. Calibration Process for Accurate Voltage and Current Correlation

The calibration was necessary to establish a precise correlation between the raw readings of the voltage by the MCU and the actual current from the input of the primary side of the series-connected current transformer they represent. The current was noted to induce a low EMF of 2V on the transformer's secondary end, and the calibration between both quantities was done using a 60W soldering iron.

Mathematically, on the primary side the design current was obtained using (1)

$$P = IV \quad (1)$$

Since the signals are in the AC form at P = 60W and V = 220V, the current was obtained as follows: $I = \frac{P}{V} = \frac{60W}{220V} = 0.273A$

Using a linear model, this value is then calibrated against the 2V induced at the secondary side for efficient current monitoring.

K. Mobile Application and User Interface Development

The Flutter framework was employed in the development of the mobile application for energy monitoring, as shown in Appendix A. The application's interface consists of several components, including frequency filters, cost input, current readings display, and charts for energy consumption. The methodology for designing the user interface and integrating functionalities is structured as follows:

A. Frequency Filters: A horizontal list of frequency filters, such as 'Min,' 'Hour,' 'Day,' 'Week,' 'Month,' and 'Year,' was implemented to enable users to select the desired time range for energy consumption analysis. Tapping on a filter updates the selected frequency for data visualization.

B. Cost Input: To facilitate user input for the cost of electricity per unit, a text field with proper formatting and validation was included. Users can input the cost, and the value is captured for further calculations in the application.

C. Current Readings Display: The application displays the current readings, including current (in Amperes), voltage (in Volts), and power (in Watts). This information is presented in a structured card format, offering a clear overview of the real-time electrical parameters.

L. Anomaly Detection Algorithm

This involves codes that implement an outlier detection algorithm using z-scores for identifying anomalies in a time series of power readings obtained from a monitoring system. The methodology of the algorithm involves the calculation of the mean and standard deviation of the power readings to establish a baseline. Z-scores are then computed for each data point, representing the number of standard deviations a particular reading is from the mean. This provides a normalized measure of how far each data point deviates from the average. There is an 'OutlierAnalyzer' that class encapsulates the functionality of the algorithm. Upon its instantiation, the algorithm is initialized with a set of power readings. The analyze Data method calculates the mean, standard deviation, and a threshold for identifying outliers. The threshold is determined by multiplying the standard deviation of the z-scores by a factor, which is set to 2.0 by default. This factor influences the sensitivity of the algorithm, allowing users to adjust the threshold for anomaly detection based on their requirements.

Finally, the 'addDataPoint' method enables the algorithm to adapt to incoming data by adding new readings and re-calculating the statistical parameters. This ensures that the algorithm remains dynamic and responsive to changes in the monitored system.

III. RESULTS AND DISCUSSION

A. Energy Consumption Data Visualization

For the developed mobile application to successfully provide users with real-time insights into their energy consumption, it has the following features:

i) Energy Elements Readings: The application displays the real-time readings from the smart meter. These readings are the Current, Voltage and Power readings, sectioned into three instances, the minimum, maximum and current Power readings over a period.

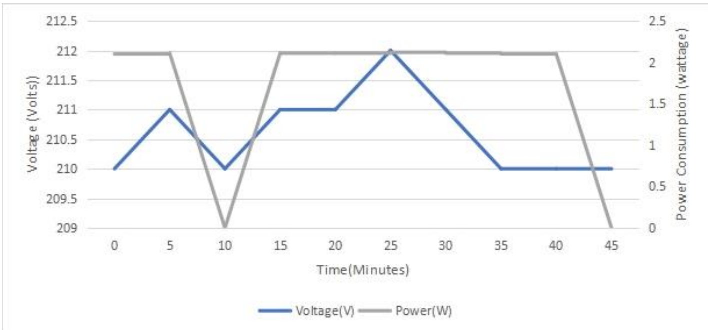

Fig. 4: Power, and Voltage readings on no-load

ii) Energy Consumption Chart: The application incorporates an energy consumption chart displayed in kilowatt-hours (kWh). This interactive chart, aimed at enhancing understanding of energy usage trends, allows users to visualize their energy consumption patterns over different time intervals, ranging from minutes to years.

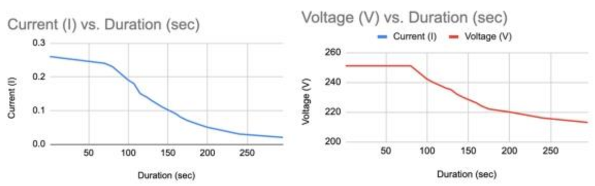

Fig. 5: Voltage and Current discharge rate after load removal

The user design interface of the mobile app is presented in Fig. 4. The user-provided cost of electricity per unit is utilized to calculate the current cost of electricity. The calculation is straightforward, where the cost per unit is multiplied by the total energy consumption, resulting in the current cost incurred by the user for the monitored period. For instance, if a user specifies that the cost of 1 unit of electricity is 15 NGN, and the monitored energy consumption is 20 kWh, the application computes the current cost as 15 NGN * 20 kWh = 300 NGN. This dynamic cost calculation feature enables users to have a real-time understanding of their financial expenditure based on their energy usage, fostering awareness and informed decision-making regarding electricity consumption patterns. The calculated cost is presented in both numeric and currency format, providing users with insights into their current energy expenses.

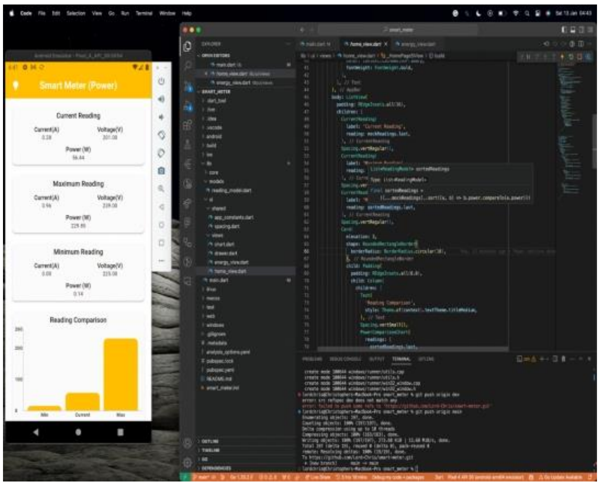

Fig. 5: The User's Mobile Interface of the Energy Monitoring System

B. Performance of the Anomaly Detection System

For the anomaly detection algorithm, testing involved simulating scenarios with known anomalies and verifying that the algorithm correctly flags these instances. Additionally, the algorithm's responsiveness to changes in the system, such as sudden spikes or drops in power consumption, was thoroughly tested. The algorithm, together with the Outlier method, applied to a time series of power readings, are employed to evaluate whether a given data point is an outlier based on its z-score in relation to the established threshold. If a data point is flagged as an outlier, it indicates a significant deviation from the expected power consumption pattern. This anomaly detection mechanism is performed decently in identifying irregularities in the monitored electrical system, potentially signifying malfunctions, faults, or unusual energy consumption patterns. This efficient implementation provides a robust foundation for anomaly detection in power consumption data, contributing to the system's ability to identify and address irregularities promptly.

C. Comparison with Existing Systems

IoT integration and Connectivity: unlike some traditional systems that may lack IoT features, limiting accessibility and remote monitoring capabilities, this project utilizes IoT technology, enabling remote access to real-time data through a web server and a mobile application. Data is also stored in a cloud database for accessibility from anywhere. Data is also stored in a cloud database for accessibility from anywhere. In addition, the system employs cost-effective hardware, such as the ESP8266 module and voltage and current sensors, making it an affordable solution. Comparatively, commercial solutions might involve more expensive hardware, potentially limiting their accessibility for certain applications. Additionally, the main program is developed in an open-source environment using the Arduino Software (IDE), fostering community collaboration and customization. Moreso, the system incorporates a feature for the user to input the cost of electricity, allowing the system to calculate the current cost of electricity based on real-time readings. On the other hand, some systems may lack this dynamic cost calculation feature, providing static or pre-configured cost values that is unable to detect anomalies

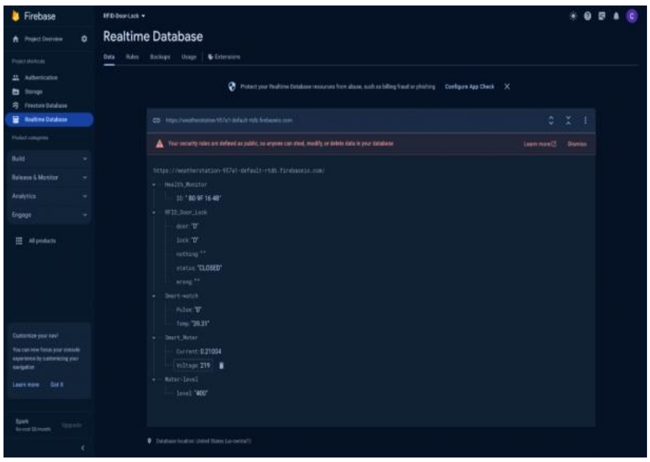

Fig. 6. The Real-time Database of the System

CONCLUSION

The research, IoT-Based Energy Monitoring System with Mobile App for Usage Tracking, presents a robust and cost-effective solution for real-time tracking and analysis of electrical parameters in buildings. By integrating IoT technology and utilizing affordable yet efficient hardware components such as the Voltage and Current sensors and ESP8266 module, the system ensures flexibility and accessibility. The comprehensive monitoring of voltage, current, power, and other parameters, coupled with the incorporation of an anomaly detection algorithm, enhances the system's reliability and performance.

The incorporation of both a web server and a mobile application provides users with convenient options for remote monitoring and control, aligning with the evolving trends in smart home technologies. Furthermore, the system's ability to calculate the current cost of electricity based on user-inputted rates adds a practical and user-centric dimension to the project, contributing to effective energy management. To further enhance and extend the capabilities of the energy monitoring system, there is a need to expand compatibility and integration by exploring connections with other popular IoT platforms and smart home ecosystems, broadening the scope of potential applications. Moreover, implementing a continuous validation and calibration routine against commercial devices will maintain confidence in the accuracy of the system's readings, ensuring its long-term success.

REFERENCES

[1] Emovon, I., Samuel, O. D., Mgbemena, C. O., and Adeyeri, M. K. (2018). Electric Power generation crisis in Nigeria: A Review of causes and solutions. *International Journal of Integrated Engineering*, *10*(1).

[2] Babatunde, O., Buraimoh, E., Tinuoye, O., Ayegbusi, C., Davidson, I., and Ighravwe, D. E. (2023). Electricity sector assessment in Nigeria: the post-liberation era. *Cogent Engineering*, *10*(1), 2157536.

[3] Ale, T., Akinboboye, O. E., Makanju, T., and Oyeleye, M. O. (2023). Evaluation Of Omotosho Gas Power Station Based on Best Industrial Practice. *International Journal of Engineering and Applied Physics*, *3*(2), 756-764.

[4] Adu, M. R., Oshiobugie, I. O., and Makanju, T. D. (2023). Electrical load flow analysis of Auchi distribution network without load shedding.

[5] Ahmad, T., and Zhang, D. (2021). Using the internet of things in smart energy systems and networks. *Sustainable Cities and Society*, *68*, 102783.

[6] Yang, F., and Gu, S. (2021). Industry 4.0, a revolution that requires technology and national strategies. *Complex and Intelligent Systems*, *7*, 1311-1325.

[7] Butt, O. M., Zulqarnain, M., and Butt, T. M. (2021). Recent advancement in smart grid technology: Future prospects in the electrical power network. *Ain Shams Engineering Journal*, *12*(1), 687-695.

[8] Benhamaid, S., Bouabdallah, A., and Lakhlef, H. (2022). Recent advances in energy management for Green-IoT: An up-to-date and comprehensive survey. *Journal of Network and Computer Applications*, *198*, 103257.

[9] Abir, S. A. A., Anwar, A., Choi, J., and Kayes, A. S. M. (2021). Iot-enabled smart energy grid: Applications and challenges. *IEEE access*, *9*, 50961-50981.

[10] Maruvala, P. K., Nayak, R., and Suvarna, M. (2024). Cyberattack issues on smart metering infrastructure. In *Smart Metering* (pp. 105-118). Elsevier.

[11] Althobaiti, A., Jindal, A., Marnerides, A. K., and Roedig, U. (2021). Energy theft in smart grids: a survey on data-driven attack strategies and detection methods. *IEEE access*, *9*, 159291-159312.

[12] Hock, D. (2020). *Detecting Energy Theft and Anomalous Power Usage in Smart Meter Data* (Doctoral dissertation, University of Plymouth).

[13] Mitra, S., Chakraborty, B., and Mitra, P. (2024). Smart meter data analytics applications for secure, reliable and robust grid system: Survey and future directions. *Energy*, *289*, 129920.