

AI-Enabled IoT prototype for priority-aware electrical load management

Sadiq Ur REHMAN^{1*}, Halar MUSTAFA², Muhammad UZAIR³

¹ Department of Computer Science, Faculty of Engineering, Sciences and Technology,
Iqra University, Pakistan
sadiq.rehman@iqra.edu.pk (*Corresponding author)

² Department of Electrical Engineering, Faculty of Engineering, Sciences and Technology,
Hamdard University, Pakistan
halar.mustafa@hamdard.edu.pk

³ Department of Electrical Engineering, Faculty of Engineering,
Islamic University of Madinah, Saudi Arabia
muzair@iu.edu.sa

Abstract: Electricity shortages and load-shedding create challenges for reliable electricity management when available capacity is insufficient to supply all connected loads. This paper presents a low-cost, proof-of-concept Internet of Things (IoT) system for priority-aware electrical load management that integrates real-time sensing, lightweight Artificial Intelligence (AI)-based demand prediction, and deterministic priority-based load control. The system estimates sector-level electricity demand using linear regression based on real-time consumption, time of day, sector type, and historical load patterns, while predefined priority rules allocate available capacity between industrial and residential loads. During a four-hour test, the system achieved an average network response time of approximately 48 ms, 99.6% data-transmission success, and 99.8% reported system uptime. Under emulated capacity constraints ranging from 100 W to 400 W, sensor readings remained within $\pm 2\%$ of expected values. The retained AI results comprised more than 1,000 data points, with a training time of approximately 2.5 s, an MSE of 0.021, and an R^2 of 0.94. The prototype reduced connected load by 25% during peak operation and 50% under an emergency condition.

Keywords: Priority, Load management, IoT, Artificial Intelligence, Demand prediction, Smart energy management.

1. Introduction

Electricity is an essential component of economic and social activity, and interruptions in electricity supply can affect industrial, residential, and other sectors. Pakistan has experienced electricity-supply challenges associated with increasing demand, power shortages, and load-shedding. Previous studies have discussed the consequences of power outages and the challenges associated with improving electricity distribution and smart-grid implementation in Pakistan (Raza et al., 2022; Ali et al., 2024). Load-shedding has also been examined in terms of its consequences and potential policy and management approaches (Kazmi et al., 2019).

During periods of constrained electricity availability, effective prioritization of electrical loads remains an important challenge, particularly when essential and non-essential loads must be managed selectively. Existing approaches include scheduled load-shedding, priority-based control, Internet of Things (IoT)-enabled monitoring, and Artificial Intelligence (AI)-based demand prediction; however, these approaches differ considerably in their computational requirements, control strategies, hardware architectures, and implementation environments. A practical low-cost system that combines demand estimation with real-time sensing and priority-based physical load control can therefore be relevant for resource-constrained applications. IoT technologies have enabled real-time monitoring and control of electrical equipment, allowing information from sensors and connected devices to be transmitted to monitoring interfaces and control systems (Rehman et al., 2016; Rehman et al., 2021; Tudora et al., 2021; Rehman & Khan, 2023; Shakir et al., 2024). In parallel, artificial intelligence and machine-learning methods have been investigated for electricity-demand forecasting and energy-management applications (Rafique & Jianhua, 2018; Ahmad & Chen, 2019; Mai et al., 2022; Liu et al., 2024).

For this purpose, this study presents an AI-enabled IoT prototype that combines lightweight demand estimation with deterministic priority-based load control. The prototype uses low-cost embedded hardware to monitor electrical loads, communicate with a web-based interface, estimate expected demand using linear regression, and apply predefined priority rules when the available capacity is constrained.

The main contribution of this work is therefore more practical and architectural in nature than developing a new algorithm for machine learning. In this system, the four components that have been integrated into the system are: (i) electrical load sensing, (ii) IoT communication and monitoring, (iii) lightweight AI-based demand estimation, and (iv) deterministic priority-based physical load control.

This section gives an overview of the paper structure. Section 2 summarizes the related work on load management, IoT-based energy consumption monitoring and AI-based electricity demand prediction. Section 3 details the hardware configuration, power supply, load configuration, AI model, decision-making process, experimentation procedure and fault handling procedures. Section 4 presents the experiment results and its limitations.

2. Literature review

There have been attempts to use load management to address issues such as the reduction of wastage, stabilization of the power grid, and electricity demand. While traditional systems used manual or scheduled load shedding systems, modern systems incorporate automation in monitoring, communication, and controls. Energy management and building energy management systems research has focused on monitoring, demand management, and control issues (Abdullah et al., 2024; Digitemie & Ekemezie, 2024).

IoT technologies have provided practical mechanisms for monitoring and controlling electrical equipment. IoT-based energy-management systems have been developed for load sharing, source management, smart metering, and home energy management (Gupta et al., 2017; Saleem et al., 2021; Condon et al., 2022; Munoz et al., 2022). Such systems demonstrate the feasibility of combining sensors, communication modules, and controllers for real-time energy-related applications. AI has also been applied to electricity-demand prediction and energy management. Regression, autoregressive, random-forest, and other machine-learning approaches have been investigated for estimating electricity consumption and supporting energy-management decisions (Rafique & Jianhua, 2018; Ahmad & Chen, 2019; Mai et al., 2022; Liu et al., 2024). These studies demonstrate that data-driven models can provide useful estimates of changing electricity demand.

Priority-based load management has similarly been investigated in microgrids and demand-responsive energy-management systems. (Korkua & Thinsurat, 2014) proposed a load-prioritization model for residential demand-responsive energy management. (Rajbhandari et al., 2022) investigated load prioritization for maintaining continuous supply to essential loads in rural microgrids, while (Alahmed & Al-Muhaini, 2020) examined an intelligent load-priority-list-based integrated energy-management system. These studies demonstrate that individual elements of the architecture used in the present work have already been investigated. The research opportunity addressed here is therefore not the introduction of a new AI algorithm or a new priority-control theory. Instead, the work focuses on the practical integration of lightweight demand prediction, IoT-based monitoring, and deterministic priority-based physical load control using low-cost embedded hardware.

The proposed system is demonstrated through a small-scale prototype in which industrial and residential loads are managed under emulated capacity constraints. The resulting system is intended to serve as a proof-of-concept for resource-constrained applications rather than as a replacement for utility-scale load-management infrastructure.

2.1. Representative energy-management technologies

Several commercial and research-oriented systems provide energy monitoring, control, analytics, or automation capabilities. Table 1 presents a qualitative overview of selected systems and technologies discussed in the literature.

Table 1. Qualitative overview of representative energy-management and connected-device technologies

Reference	System/Device	Features	Advantages	Limitations
Flotats Boix (2023)	EcoStruxure by Schneider Electric	Cloud-based energy-management system, data analytics, energy monitoring, optimization, predictive maintenance	Real-time energy-usage monitoring, demand-response optimization, remote monitoring and control	High cost, complex setup and configuration
Lascu (2019)	EnergyIP by Siemens	Grid-management platform, distributed-energy-resource integration, real-time grid monitoring, analytics	Real-time grid management, renewable-energy integration, predictive maintenance	Complex setup and configuration, limited end-user functionality
McHugh (2017)	Nest Learning Thermostat	Smart thermostat, temperature control, energy-saving features, remote monitoring and control	Learning-based temperature control, energy-saving features, remote monitoring	Compatibility limitations with some HVAC systems and relatively high cost
Liu et al. (2019)	Belkin WeMo	Smart plug, energy monitoring, remote control and scheduling	Real-time energy monitoring, remote control, scheduling, easy setup	Limited energy-optimization capabilities and appliance compatibility
Hilbolling et al. (2021)	Philips Hue	Smart lighting, remote control, scheduling, energy-efficient LED lighting	Energy-efficient lighting, remote control, scheduling, customizable lighting	Cost and limited energy-optimization functionality

Table 1 provides a qualitative overview of representative technologies relevant to monitoring, automation, energy management, and connected-device control. The table is intended to provide contextual background rather than a direct performance comparison.

3. Components and methodology

The prototype was developed to demonstrate priority-aware electrical-load control using an AI-enabled IoT architecture. The system contains hardware and software components for electrical sensing, communication, demand estimation, local display, web-based monitoring, and physical load switching.

3.1. System overview

The prototype uses separate low-voltage control circuitry and a 220 V AC supply for the experimental load circuit. The Arduino Mega 2560 serves as the main controller. It is connected to an ESP8266 Wi-Fi module for network communication, a 16×2 LCD for local monitoring, mechanical relays for ON/OFF load control, and an ACS712 Hall-effect current sensor for measuring load current. The overall system architecture is shown in Figure 1. The architecture

consists of the electrical-load section, sensing section, Arduino-based control section, Wi-Fi communication section, local LCD interface, and web-based monitoring interface.

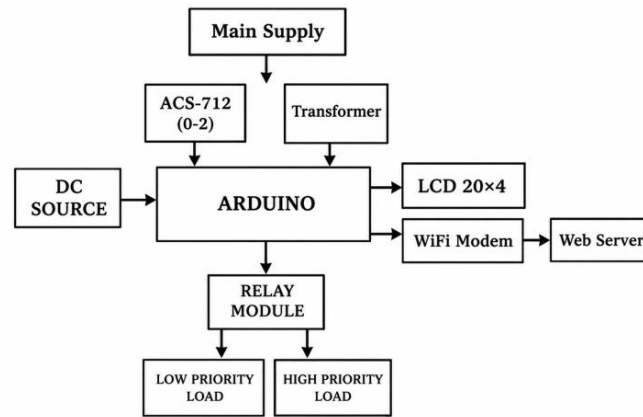


Figure 1. System block diagram (According to our own elaboration)

The physical-components specifications are summarized in Table 2.

Table 2. System components and specifications

Component	Specification/Model
Arduino Mega 2560	ATmega2560, 54 digital pins, 16 analog pins, 16 MHz
ESP8266	Wi-Fi module, 802.11 b/g/n, 80 MHz processor
Mechanical relay	10 A, 5 V coil, SPDT type
LCD display	16 × 2 character module
Control power supply	12 V regulated DC output
Current sensor	ACS712 Hall-effect current sensor

Figure 2 presents the overall methodological workflow of the proposed AI-enabled IoT load-management prototype, including electrical load sensing, data acquisition, AI-based demand prediction, priority-based decision-making, physical load control, and IoT-based monitoring.

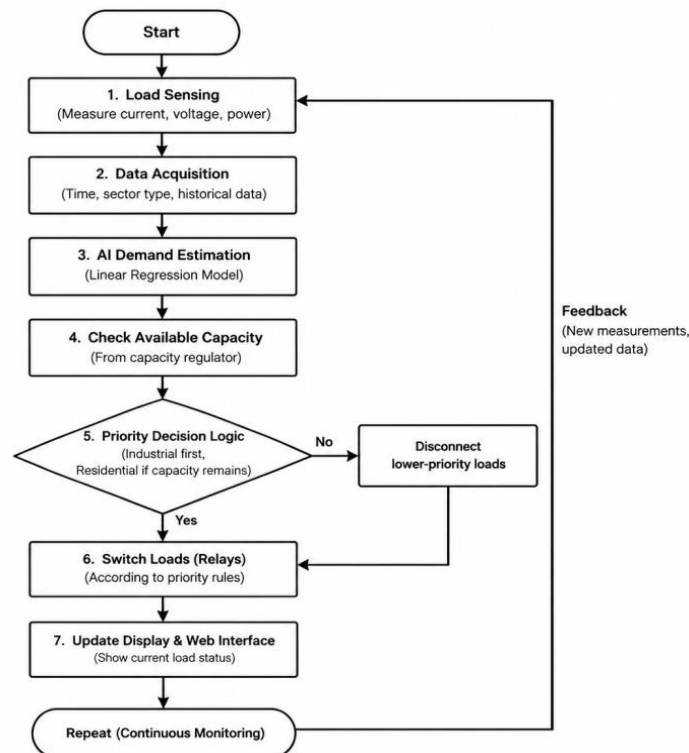


Figure 2. Workflow of the system (According to our own elaboration)

3.2. Power supply design

The control-circuit power supply converts the available 220 V AC source to a regulated 12 V DC supply for the low-voltage control components. A step-down transformer reduces the AC voltage from 220 V AC to approximately 25 V AC. The output is then converted to DC through a bridge rectifier and regulated to 12 V using a 7812 voltage regulator. The 220 V AC supply is used separately for the prototype load circuit. This arrangement separates the low-voltage control circuitry from the experimental AC load circuit.

3.3. Load configuration and priority control

The prototype was configured to emulate a constrained-capacity operating scenario using incandescent bulbs divided into two sectors: industrial and residential. Each sector was represented by a maximum connected load of 200 W, implemented in 100 W increments.

The system makes use of a four-button based capacity regulator interface to represent different available capacity conditions in 100 W increments with the highest being 400 W. A button click sends a signal to the Arduino controller, which uses predefined priority rules to decide which loads can be connected. The priority of the prototype is given to the industrial side. The priority sequence is as follows:

- 100 W: 100 W Industrial Load is connected.
- 200 W: Two 100 W industrial loads are connected.
- 300 W: 200 W Industrial load is still connected and 100 W residential load is connected.
- 400 W: 200 W Industrial load and 200 W Residential load are connected.

Figure 3 (a) shows the physical configuration of the industrial and residential loads, and Figure 3 (b) shows the entire configuration of the system.



(a) Industrial and residential load setup



(b) Prototype hardware configuration

(According to our own elaboration)

3.4. Data acquisition and AI model

The part of the system related to artificial intelligence was developed for the purpose of predicting expected load demand for both residential and industrial sectors. The input variables included:

- The current load as measured by sensors,
- The time of day, which is a categorical variable,
- The sector – either residential or industrial; and
- The historical load patterns as observed at each hour.

The output variable was expected load demand for the respective sector. A linear regression model was chosen due to its simplicity and relatively low computational cost. It could be written in the following way:

$$\hat{P} = \beta_0 + \sum_{i=1}^n \beta_i X_i \quad (1)$$

where $\hat{P}$ represents the estimated electrical load, β_0 is the intercept, β_i represents the coefficient associated with feature X_i and n is the number of input features. The retained training results are summarized in Table 3. The experimental results show that the model was trained using over 1,000 data points. The training time of the model is reported to be around 2.5 seconds, with the mean square error (MSE) equal to 0.021 and R^2 equals 0.94.

Table 3. Linear regression training performance

Metric	Value
DataSet Size	>1,000 observations
Training Time	~2.5 seconds
Mean Squared Error (MSE)	0.021
R^2 Score	0.94
Documented update interval	Approximately every 2 hours

There is a two-hour update period for the model according to new data points. However, the original dataset and the details of the training files are no longer available to replicate the entire process. Hence, the training time, MSE, and R^2 values mentioned are preserved as per the original results.

The AI model estimates demand, while load allocation is controlled by deterministic priority rules implemented through the Arduino. Since reproducible experiments with alternative models are unavailable, quantitative comparisons with other machine-learning methods cannot be performed.

3.5. Decision-making procedure

The prediction and priority-control mechanisms operate together, with the controller continuously monitoring load demand and available capacity. The system initializes the Arduino, ESP8266, LCD, sensor, and relays, establishes IoT communication, measures and transmits load data, and uses real-time and historical inputs to estimate demand through linear regression. The estimated demand is compared with available capacity, higher-priority loads are maintained, and lower-priority loads are disconnected when necessary. The resulting status is updated on the LCD and IoT interface, and the process is continuously repeated. Thus, AI provides demand estimation, while deterministic priority rules ensure appropriate allocation of limited capacity.

3.6. Experimental procedure

The prototype was evaluated through communication/system-operation and constrained load-control tests. Load-control experiments used the four-button regulator to emulate 100–400 W capacity conditions, with industrial loads prioritized up to 200 W before residential loads. During the four-hour communication experiment, the measured response times achieved were an average of 48 ms, 99.6% communication success rate, 99.8% uptime, and web command response times of less than 0.3 seconds. Sensor outputs varied within $\pm 2\%$ of their expected values. The capacity settings mentioned above are emulations of the utility grid capabilities. The retained AI results included over 1,000 data points, 2.5 s training time, MSE of 0.021, and R^2 of 0.94; however, the unavailable raw dataset prevents retrospective reproduction of the complete statistical analysis.

3.7. Fault handling

The prototype included predefined fault-handling behavior for communication loss, sensor abnormalities, overload conditions, and erroneous load readings.

For communication loss, the prototype behavior included automatic network reconnection. For sensor abnormalities, the intended behavior was to detect and flag abnormal conditions and notify the operator. For overload conditions, lower-priority connections were intended to be disconnected rather than disconnecting all loads. For erroneous readings, the system was designed

to identify and ignore abnormal measurements. The fault-handling scenarios were evaluated at the prototype level through simulated or injected conditions. They therefore demonstrate the intended fault-handling behavior of the prototype rather than validation under actual distribution-grid fault conditions.

4. Results and discussion

In this section, the performance evaluation of the proposed AI-enabled IoT load management system is presented for different performance metrics.

4.1. Connectivity and network performance

Communication between the Arduino controller and the IoT web interface is important for monitoring and controlling the prototype. During the continuous four-hour test, the system achieved an average network response of approximately 48 ms.

The reported data-transmission success rate was 99.6%, corresponding to 0.4% reported data loss. The reported hardware/Wi-Fi uptime was 99.8%. The web interface also demonstrated an average command delay of less than 0.3 s. These measurements indicate stable communication performance during the four-hour prototype test. The main connectivity and network-performance measurements are summarized in Table 4.

Table 4. Connectivity and network performance

Metric	Reported value
Continuous test duration	4 h
Average network response	≈ 48 ms
Data-transmission success rate	99.60%
Reported data loss	0.40%
Reported system uptime	99.80%
Average web-command response	<0.3 s

The network reliability metrics obtained during the four-hour test are also illustrated in Figure 4. The network measurements should be interpreted within the experimental environment of the prototype. They do not constitute a characterization of communication performance across utility-scale networks or geographically distributed electrical infrastructure.

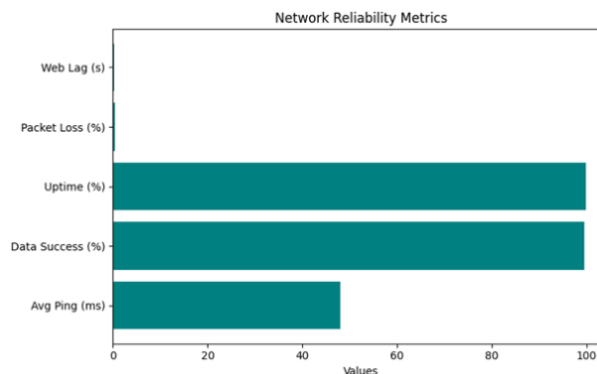


Figure 4. Network reliability metrics (According to our own elaboration)

4.2. Priority-based load control

After the power supply is turned on, the system enters a “connecting” status and then establishes a connection to the IoT web server via Wi-Fi. As shown in Figure 5, initial system values are displayed both on the local LCD and the web interface.



Figure 5. System startup and initial values on the IoT web server (According to our own elaboration)

The load-control logic was tested incrementally using 100 W capacity steps up to a total of 400 W. The resulting sequence is summarized in Figure 6.

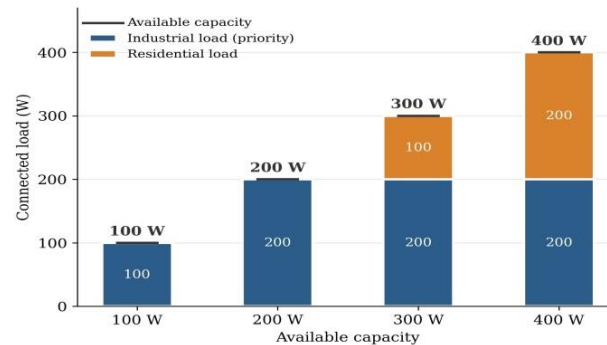


Figure 6. Priority-based load-control states (According to our own elaboration)

The measurements indicate that the prototype implemented the intended load-control sequence and that sensor readings remained within the reported $\pm 2\%$ range under the tested conditions.

4.2.1. 100 W capacity: Industrial load

The system starts with the ability to switch on a 100 W load using the capacity regulator, which sends a signal to the Arduino to allow this load. The Arduino processes this signal according to the priority of the connected loads and switches on the corresponding load. As shown in Figure 7, pressing the button to allow 100 W is able to switch on a 100 W industrial load, which is considered the highest priority load.

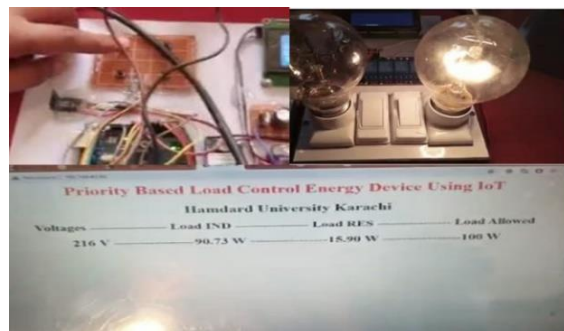


Figure 7. 100W Industrial load activated (According to our own elaboration)

4.2.2. 200 W capacity: Industrial load

To increase the industrial load up to 200 W, the same steps are repeated by pressing another button on the capacity regulator. This action triggered the system to activate an additional 100 W bulb, raising the total industrial load to 200 W. The changes are reflected both in the hardware setup and on the web interface, as shown in Figure 8. Notably, only the industrial load was affected, while the residential load remained unchanged.

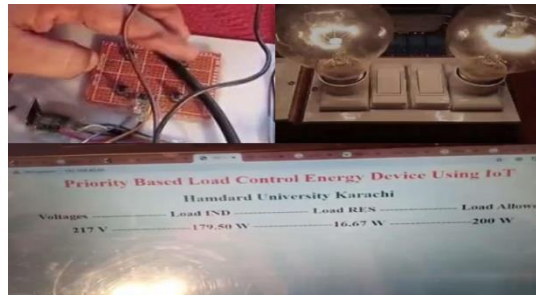


Figure 8. 200W Industrial load activated (According to our own elaboration)

4.2.3. 300 W capacity: Industrial and residential loads

However, now the total system load was increased to 300 W by pressing another button on the capacity regulator. This command enabled the Arduino system to switch on another load of 100 W. Since the industrial load was already switched on at 200 W, the Arduino system switched on another load of 100 W in the residential sector. The status of the load is shown in Figure 9, which is available both in hardware and on the web.

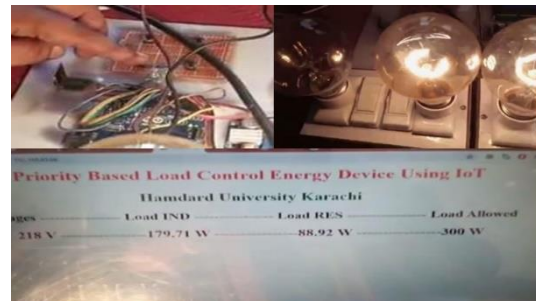


Figure 9. 200W Industrial load and 100w residential load (According to our own elaboration)

4.2.4. 400 W capacity: Industrial and residential loads

To attain this total load of 400 W, another button on the capacity regulator was pressed, which switched on another 100 W bulb. This ensured that the industrial sector had a load of 200 W, while the residential sector had 200 W as well. The load status of both sectors is shown in Figure 10.

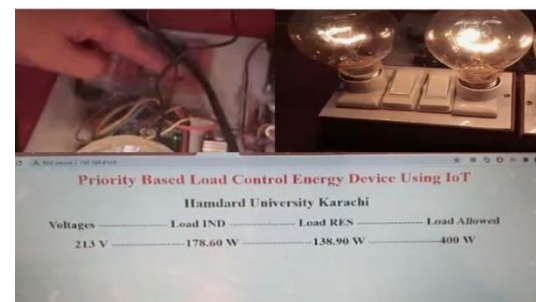


Figure 10. 200W Industrial load and 200w residential load (According to our own elaboration)

4.3. AI-Based load prediction results

The AI results provide several representative examples of the relationship between predicted and actual sector-level load.

For the industrial sector, the model predicted 175 W during the morning condition compared with an actual load of 180 W. The corresponding reported percentage agreement was 97.2%. During the evening condition, the predicted value was 195 W compared with an actual load of 200 W, corresponding to a reported percentage agreement of 97.5%.

For the residential sector, the model predicted 105 W during the morning condition compared with an actual load of 100 W, corresponding to a reported percentage agreement of 95.0%. During the evening condition, the model predicted 190 W compared with an actual load of 200 W, also corresponding to a reported percentage agreement of 95.0%. The percentage-agreement values are retained from the original experimental record and are presented only as representative prediction-error measures. They should not be interpreted as classification accuracy

The primary regression metrics are an MSE of 0.021 and an R^2 value of 0.94, based on more than 1,000 data points. The four examples in Table 5 should therefore be interpreted as representative retained prediction examples rather than as a substitute for a complete test-set evaluation. Because the original raw dataset and detailed training records are no longer available, the complete distribution of prediction errors, confidence intervals, train/test performance, and temporal generalization cannot be independently reproduced from the present documentation.

Table 5. Reported AI-Based load prediction results

Sector	Period	Predicted load	Actual load	Reported Prediction Agreement
Industrial	Morning	175 W	180 W	97.20%
Industrial	Evening	195 W	200 W	97.50%
Residential	Morning	105 W	100 W	95.00%
Residential	Evening	190 W	200 W	95.00%

4.4. Load reduction under constrained operating conditions

The prototype results demonstrate different connected-load levels under the tested constrained operating conditions. The connected-load states and corresponding reductions under the tested operating conditions are presented in Figure 11.

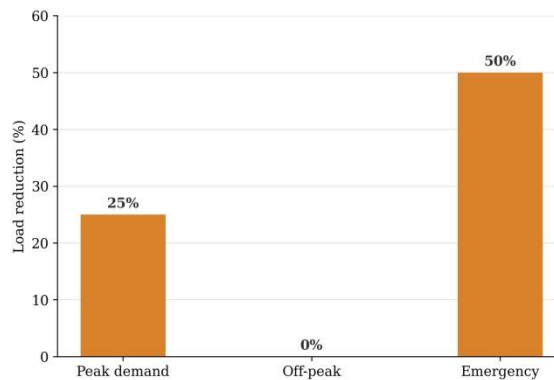


Figure 11. Load reduction under operating conditions (According to our own elaboration)

During the peak condition, the connected load was reduced from 400 W to 300 W, corresponding to a 25% reduction in connected load. During the off-peak condition, the connected load remained at 400 W. Under the emergency condition, the connected load was reduced from 400 W to 200 W, corresponding to a 50% reduction.

4.5. Fault handling and recovery behavior

The prototype had predefined behavior for the loss of the network, abnormal sensors, overload conditions, and load reading errors. For network disconnection, according to the experimental documentation provided, reconnection occurs automatically within 5 s. For an abnormal sensor, the system is supposed to recognize the abnormal situation and notify the operator. In case of an overload situation, lower-priority loads were supposed to be disconnected. For load reading errors, the system would recognize the abnormality and ignore it. The reported fault-handling behavior for these scenarios is summarized in Table 6.

Table 6. Reported Fault-Handling Behavior

Test Scenario	Prototype Behavior	Result/Status
Network loss	Automatic reconnection	Approximately 5 s reported reconnection
Sensor failure	Detect/flag abnormal condition	Operator alert
Overload condition	Disconnect lower-priority loads	Simulated condition
False load reading	Detect/reject abnormal reading	Injected error

The fault-handling behavior provides automated handling within the prototype architecture. The overload and load reading error cases were simulated at the prototype level and cannot be considered validation for an electrical distribution grid fault.

4.6. Qualitative comparison with existing systems

The comparison in Table 7 has been conducted qualitatively because of differences in the datasets, objectives, prediction targets, hardware setups, and procedures among the referenced studies.

Table 7. Qualitative comparison with representative energy management studies

Study	Technology Used	Real-Time Control	AI-Based Prediction	Hardware Implementation	Reported Prediction Metric
Rehman et al. (2021)	IoT-based monitoring	Yes	No	Yes	N/A
Nazir & Li (2025)	AI-based electricity-demand forecasting	No	Yes	No	93%
Liu et al. (2024)	Machine-learning-based electricity-metering analysis	Limited	Yes	No	85%
Kazmi et al. (2019)	Load-shedding analysis/scheduling	No	No	Yes	N/A
Present prototype	AI + IoT integrated prototype	Yes	Yes	Yes	MSE = 0.021; R² = 0.94

The comparison shows that previous studies have addressed different combinations of AI prediction, IoT sensing and communication, priority-based control, and hardware implementation. The present prototype integrates these functions within a single experimental architecture; however, the differences in evaluation conditions prevent direct quantitative performance comparison. The current prototype incorporates demand prediction, IoT communication, sensing, and physical load prioritization in one hardware setup.

4.7. Limitations and future experimental validation

There are a few limitations to the study. First of all, the 400 W prototype is limited to a small-scale proof-of-concept model and is not representative of utility-scale operation. In addition, the power range of 100–400 W was simulated with the four-button regulator.

In terms of the evaluation of the AI algorithm, only one type of analysis, namely linear regression, was used without any comparison with other lightweight models. In addition, the original dataset used and the training record is unavailable, and thus the preprocessing, feature encoding, train/test split, hyperparameters, and error distribution cannot be recovered. Thus, the results presented, including the dataset with more than 1,000 points, 2.5 s training time, MSE equal to 0.021 and R² of 0.94, are taken from the original study record.

Future experimental validation should therefore employ a newly documented dataset with a clearly defined preprocessing procedure, training/test partition, and reproducible model configuration. Multiple lightweight regression or machine-learning models should be evaluated under the same experimental protocol, together with longer-duration testing, more diverse load profiles, and a larger number of controllable loads.

5. Conclusions

In this study, a cheap IoT solution is introduced for managing electrical loads on a priority basis through real-time sensing, lightweight AI-based demand estimation, and load control. The prototype includes an Arduino Mega 2560, an ESP8266, an ACS712 sensor, relays, an LCD screen, and a web interface.

Linear regression estimates demand based on real-time consumption, time, sector-type, and historical consumption. The algorithm was able to estimate demand with an MSE of 0.021 and an R^2 of 0.94, with 95–97.5% agreement. For load control testing, it was possible to manage industrial loads with higher priority than residential loads for up to 100–400 W loads. Sensor readings were within $\pm 2\%$ of the estimated value.

For a four-hour communication test, the prototype showed an average response time of 48 ms, a 99.6% successful transmission rate, 99.8% availability, and a web command-response delay of less than 0.3 s. Peak and emergency situations resulted in reducing connected load from 400 W to 300 W and 200 W, respectively.

Overall, the experiments demonstrate the potential of integrating lightweight AI, IoT and load control. Further research will be carried out based on larger data sets, different AI methods and loads, additional testing, and larger-scale implementation.

Author contributions

Conceptualization: S.U.R., H.M., and M.U.; Data Curation: S.U.R. and M.U.; Formal Analysis and Methodology: S.U.R. and H.M.; Supervision: H.M.; Validation: H.M. and M.U.; Visualization: S.U.R. and M.U.; Writing—original draft: S.U.R. and M.U.; Writing—review and editing: All authors under the lead of S.U.R. All authors have read and agreed to the published version of the manuscript.

Submission received: 13 March 2026; Revised: 17 September 2026; Accepted: 20 September 2026; Published: 30 September 2026.

REFERENCES

- Abdullah, F. B., Aqeeq, M. A., Iqbal, R. et al. (2024) Enhancing electricity distribution efficiency in Pakistan: A framework for progress and action. *Utilities Policy*. 88, 101746. <https://doi.org/10.1016/j.jup.2024.101746>.
- Ahmad, T. & Chen, H. (2019) Nonlinear autoregressive and random forest approaches to forecasting electricity load for utility energy management systems. *Sustainable Cities and Society*. 45, 460-473. <https://doi.org/10.1016/j.scs.2018.12.013>.
- Alahmed, A. S. & Al-Muhaini, M. M. (2020) An intelligent load priority list-based integrated energy management system in microgrids. *Electric Power Systems Research*. 185, 106404. <https://doi.org/10.1016/j.epsr.2020.106404>.
- Ali, Z., Hussain, A. & Hussain, S. (2024) Investigating the impact of power outages on sustainable community development: exploring the local community's perception in Pakistan. *Kybernetes*. 53(10), 3462-3483. <https://doi.org/10.1108/K-03-2023-0383>.
- Condon, F., Martínez, J. M., Eltamaly, A. M. et al. (2022) Design and implementation of a cloud-IoT-based home energy management system. *Sensors*. 23(1), 176. <https://doi.org/10.3390/s23010176>.

- Digitemie, W. N. & Ekemezie, I. O. (2024) A comprehensive review of Building Energy Management Systems (BEMS) for improved efficiency. *World Journal of Advanced Research and Reviews*. 21(3), 829-841. <https://doi.org/10.30574/wjarr.2024.21.3.0746>.
- Flotats Boix, G. (2023) *Design and simulation of a solution based on Ecostruxure for a Data Center*. B.Sc. thesis, Universitat Politècnica de Catalunya.
- Gupta, Y., Bajoria, S. P., Singh, R. K. & Swathika, O. V. G. (2017) IoT based energy management system with load sharing and source management features. In: *Proceedings of the 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)*. pp. 169-177. <https://doi.org/10.1109/UPCON.2017.8251042>.
- Hilbolling, S., Berends, H., Deken, F. & Tuertscher, P. (2021) Sustaining complement quality for digital product platforms: A case study of the Philips Hue ecosystem. *Journal of Product Innovation Management*. 38(1), 21-48. <https://doi.org/10.1111/jpim.12555>.
- Kazmi, H., Mehmood, F., Tao, Z. et al. (2019) Electricity load-shedding in Pakistan: Unintended consequences, opportunities, and policy recommendations. *Energy Policy*. 128, 411-417. <https://doi.org/10.1016/j.enpol.2019.01.017>.
- Korkua, S. K. & Thinsurat, K. (2014) A load prioritization model for a smart demand responsive energy management system in the residential sector. *Walailak Journal of Science and Technology*. 11(1), 7-18.
- Lascu, C. E. (2019) Smart Metering, a first step towards smart grids. In: *International Conference on Management and Industrial Engineering*. 9, pp. 490-502. Niculescu Publishing House.
- Liu, H., Spink, T. & Patras, P. (2019) Uncovering security vulnerabilities in the Belkin WeMo home automation ecosystem. In: *Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. pp. 894-899. <https://doi.org/10.1109/PERCOMW.2019.8730685>.
- Liu, K., Liu, A., Ma, X. & Jia, X. (2024) Artificial Intelligence-based Real-Time Electricity Metering Data Analysis and its Application to Anti-Theft Actions. *International Journal of Advanced Computer Science and Applications*. 15(6). doi: <https://doi.org/10.14569/IJACSA.2024.0150676>.
- Mai, T. T., Nguyen, P. H., Haque, N. A. N. M. & Pemen, G. A. J. M. (2022) Exploring regression models to enable monitoring capability of local energy communities for self-management in low-voltage distribution networks. *IET Smart Grid*. 5(1), 25-41. <https://doi.org/10.1049/stg2.12049>.
- McHugh, C. J. (2017) *Evaluation of the Nest Learning thermostat in a multifamily apartment setting*. PhD thesis, University of Georgia.
- Munoz, O., Ruelas, A., Rosales, P. et al. (2022) Design and development of an IoT smart meter with load control for home energy management systems. *Sensors*. 22(19), 7536. <https://doi.org/10.3390/s22197536>.
- Nazir, M. U. & Li, J. (2025) Forecasting of electricity consumption in Pakistan based on integrating machine learning algorithms and Monte Carlo simulation. *Electrical Engineering*. 1-21. <https://doi.org/10.1007/s00202-024-02923-6>.
- Rafique, S. F. & Jianhua, Z. (2018) Energy management system, generation and demand predictors: a review. *IET Generation, Transmission & Distribution*. 12(3), 519-530. <https://doi.org/10.1049/iet-gtd.2017.0354>.
- Rajbhandari, Y., Marahatta, A., Shrestha, A. et al. (2022) Load prioritization technique to guarantee the continuous electric supply for essential loads in rural microgrids. *International Journal of Electrical Power & Energy Systems*. 134, 107398. <https://doi.org/10.1016/j.ijepes.2021.107398>.
- Raza, M. A., Aman, M. M., Abro, A. G. et al. (2022) Challenges and potentials of implementing a smart grid for Pakistan's electric network. *Energy Strategy Reviews*. 43, 100941. <https://doi.org/10.1016/j.esr.2022.100941>.
- Rehman, S. U. & Khan, A. (2023) Integrating IoT Technology for Improved Distribution Transformer Monitoring and Protection. *Electrical, Control and Communication Engineering*. 19(1), 22-28. <https://doi.org/10.2478/ecce-2023-0004>.

Rehman, S. U., Khan, I. U., Moiz, M. & Hasan, S. (2016) Security and privacy issues in IoT. *International Journal of Communication Networks and Information Security*. 8(3), 147.

Rehman, S. U., Mustafa, H. & Larik, A. R. (2021) IoT-based substation monitoring & control system using Arduino with data logging. In: *Proceedings of the 4th International Conference on Computing & Information Sciences (ICCIS)*. pp. 1-6. <https://doi.org/10.1109/ICCIS54243.2021.9676384>.

Saleem, M. U., Usman, M. R. & Shakir, M. (2021) Design, implementation, and deployment of an IoT based smart energy management system. *IEEE Access*. 9, 59649-59664. <https://doi.org/10.1109/ACCESS.2021.3070960>.

Shakir, M., Karim, S., Memon, S., Rehman, S. U. & Mustafa, H. (2024) An improvement in IoT-based smart trash management system using Raspberry Pi. *International Journal of Computer Vision and Robotics*. 14(2), 191-201. <https://doi.org/10.1504/IJCVR.2024.136997>.

Tudora, E., Tîrziu, E. & Gheorghe-Moisii, M. (2021) Internet of Things in energy: technologies, applications, challenges and future trends. *Romanian Journal of Information Technology and Automatic Control [Revista Romana de Informatica si Automatica]*. 31(2), 45–58. <https://doi.org/10.33436/v31i2y202104>.



Sadiq Ur REHMAN is an Associate Professor at Iqra University, Pakistan, with a Ph.D. in Electrical Engineering from Hamdard University and an M.Sc. from the University of Duisburg-Essen, Germany. His research focuses on 5G, WBAN, IoT, and smart energy systems. He has published extensively and obtained research grants in sustainable energy and smart technologies.



Halar MUSTAFA received B.E. and M.E. degrees from Hamdard University, Pakistan, where he is currently pursuing a Ph.D. in Electrical Engineering and serving as a Lecturer. His research focuses on energy management systems for microgrids, including data-driven control, optimization, renewable energy, battery storage, demand response, and machine learning to enable efficient and sustainable smart grids.



Muhammad UZAIR is an Associate Professor at the Islamic University of Madinah, Saudi Arabia. He holds B.Sc., M.Sc., and Ph.D. degrees in Electrical Engineering from UET Lahore, Concordia University, and the University of Guelph, respectively. His research interests include image and video processing, signal processing, engineering and medical applications, IoT, wireless protocols, and wireless video optimization.



This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.