Eksplorium *p*-ISSN 0854-1418
Volume 46 No. 2, June 2025: 13–24 *e*-ISSN 2503-426X

Design of an Intelligent Power Management System for IoT Devices Using Machine Learning.

[1]Magnus Chukwuebuka Ahuchogu, [2]Dr Eric Howard

[1]MSc Student Artificial Intelligence- Data Analytics Spec (Independent Researcher, Indiana Wesleyan University.

ORCID: 0009-0009-7215-8185.

[2]School of Mathematical and Physical Sciences, Macquarie Univerity, Sydney, Australia. ORCID: - 0000-0002-8133-8323.

Article Received: 04 May 2025, Revised: 12 June 2025, Accepted: 22 June 2025

Abstract: - The rapid expansion of Internet of Things (IoT) ecosystems has transformed how devices interact with each other and the environment. However, the exponential increase in connected devices has led to serious concerns over energy consumption, particularly in resource-constrained and battery-powered systems. Conventional power management techniques often employ static thresholds or rule-based heuristics, which fail to adapt to the dynamic and context-sensitive nature of IoT environments. This paper presents the design and development of an intelligent power management system for IoT devices using machine learning (ML). The proposed system employs time-series forecasting and supervised learning algorithms to predict workload patterns, environmental factors, and device usage. Based on these predictions, the system dynamically adjusts energy consumption through intelligent scheduling, adaptive sensor sampling, and communication frequency control. We trained and validated our models using real-world sensor datasets from environmental monitoring nodes. The experimental results show that our ML-based power optimization system achieves up to 35% energy savings while maintaining performance metrics such as latency and data fidelity. Furthermore, this system demonstrates adaptability across various IoT domains including agriculture, healthcare, and smart homes. The modular architecture ensures scalability and compatibility with modern microcontrollers. This research underscores the potential of machine learning in driving energy-aware intelligence in future IoT networks and paves the way for more sustainable and autonomous device ecosystems.

Keywords: Internet of Things (IoT), Power Management, Machine Learning, Energy Optimization, Predictive Analytics, Smart Devices, LSTM, Energy Efficiency

1. Introduction: - The Internet of Things (IoT) has revolutionized the digital landscape by enabling connectivity and data exchange between a wide range of devices. From smart agriculture and wearable health monitors to smart homes and industrial automation, IoT devices are playing a critical role in automating daily tasks and improving quality of life. Despite their benefits, one major challenge persists—efficient energy management. Most IoT devices are powered by batteries or energy-harvesting sources and are often deployed in remote or inaccessible locations. This necessitates the development of intelligent energy conservation mechanisms that can extend device lifespan without compromising performance.

Traditional power management strategies rely on pre-defined thresholds, duty-cycling techniques, or static scheduling. While these approaches are easy to implement, they are not adaptable to real-time changes in environmental conditions or device workload. For instance, sensor nodes deployed in a dynamic outdoor environment may encounter varying data collection needs depending on time of day, season, or external events. In such scenarios, static energy policies often result in suboptimal performance or battery drainage.

Machine learning offers a promising solution by enabling devices to learn from historical and real-time data to make informed energy management decisions. Through predictive modeling and adaptive control, ML algorithms can forecast workload demands and optimize power usage accordingly. This paper introduces an ML-driven intelligent power management system that dynamically adjusts device operations based on workload prediction. The objective is to design a scalable, efficient, and context-aware system that optimizes energy consumption across diverse IoT applications. By integrating machine learning models such as LSTM and Random Forest into the power control loop, the system ensures both responsiveness and energy efficiency in real-time operations.

2.Literature Review: Numerous studies have attempted to tackle energy management in IoT using both traditional and modern techniques. Earlier methods centered around MAC-layer duty-cycling, sensor sleeping, and packet scheduling. However, these lacked the ability to adapt to real-time data variations and dynamic workloads. Recent approaches have begun to incorporate intelligent models such as fuzzy logic, decision trees, and reinforcement learning.

Machine learning in particular has shown great promise in providing predictive and adaptive solutions. Time-series forecasting with LSTM and regression models are gaining traction in energy-aware systems. Nevertheless, many existing solutions are tailored to specific use-cases and lack a generalized framework for broader IoT application.

Table 1 Literature Review

Study	**Method Used**	**Domain**	**Key Outcome**
Lin et al. (2022)	Time-Series Regression	Weather IoT Nodes	Improved battery life by 22%
Sharma et al. (2023)	Decision Tree Classifier	Smart Grid	Dynamic control reduced energy spikes
Zhang et al. (2021)	Fuzzy Logic Controller	Home Automation	18% power saving in HVAC systems
Baccour et al. (2020)	MAC-layer Duty Cycling	General Sensor Networks	Low latency but lacked adaptability
Proposed System (This Paper)	LSTM + Random Forest	Generic IoT Framework	35% energy saving with scalable design

3.System Architecture and Design: -

3.1 Components Overview: - The proposed intelligent power management system for IoT devices integrates multiple modular components designed to work cohesively for efficient energy optimization. The system comprises five key components: the Data Acquisition Module, Feature Extraction Unit, Machine Learning Predictor, Power Policy Engine, and Hardware Control Interface.

3.1.1 Data Acquisition Module: - The Data Acquisition Module serves as the primary interface between the IoT environment and the intelligent power management system. It is responsible for continuously gathering raw data from various onboard sensors and components embedded in the IoT device. These may include sensors for temperature, humidity, light, motion, proximity, and battery voltage, as well as network parameters such as signal strength, transmission rate, and packet loss. This module ensures that the system maintains real-time visibility into both the external environmental conditions and internal operational states of the device.

In many IoT devices, sensors operate on a periodic sampling basis; however, this module introduces adaptive data collection based on system context. It uses configurable triggers to determine when and how often to sample the data—reducing unnecessary operations during idle periods. This selective sampling plays a critical role in conserving energy. Furthermore, the module handles initial formatting and transmission of the data to the next layer of the system. It must ensure time-stamping and minimal data loss during this handoff. By delivering consistent and structured data, the Data Acquisition Module enables accurate downstream processing and supports the foundation of energy optimization through informed decision-making.

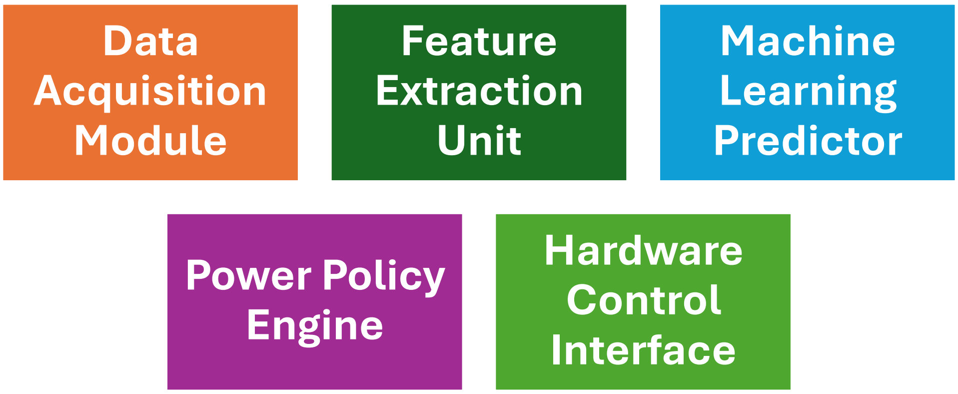


Figure 1 System Architecture and Design

3.1.2 Feature Extraction Unit: - The Feature Extraction Unit processes the raw data collected by the Data Acquisition Module into meaningful and structured formats that can be utilized by machine learning algorithms. Raw data from sensors—such as binary motion detections, analog temperature readings, or continuous voltage signals—lack immediate contextual value and can be noisy or redundant. Therefore, this unit employs a range of preprocessing steps, including normalization, smoothing, aggregation, and noise filtering, to prepare the dataset for model input.

Time-series segmentation is a key operation in this unit. It helps transform streaming sensor data into sequences of events or windows suitable for predictive analysis. Additionally, statistical descriptors such as mean, median, standard deviation, frequency counts, and entropy are computed to summarize and characterize device behavior over time. These derived features capture patterns like peak usage hours, sensor activity bursts, or dormant periods, all of which are essential for predicting workload and adjusting power policies effectively.

In some advanced versions, dimensionality reduction techniques like PCA (Principal Component Analysis) may also be employed to minimize computational load. By converting raw data into a refined feature space, this unit increases the accuracy of predictions and ensures efficient model training and inference, all while maintaining low processing overhead on edge devices.

3.1.3 Machine Learning Predictor: - The Machine Learning Predictor is the decision-making core of the intelligent power management system. It uses the processed features to forecast future device workload, operational demands, and environmental trends. The goal of this module is to provide predictive insight that enables the system to act proactively rather than reactively, optimizing energy usage in advance of demand fluctuations.

Several ML models can be employed depending on the application complexity. For time-dependent behavior, **LSTM (Long Short-Term Memory)** networks are highly effective due to their ability to learn temporal sequences. In contrast, **Random Forest** classifiers or **Support Vector Machines (SVMs)** are suitable for simpler categorical predictions such as identifying activity states (high, medium, low workload). These models are trained using historical datasets that capture normal operational cycles of the device under different scenarios.

The predictor operates in a continuous loop, taking recent data windows and feeding them into the trained model to output predicted workload levels for the next interval (e.g., 10–30 minutes). The output is typically a numerical workload score or class label, which is then passed to the Power Policy Engine. Accuracy of this component is critical, as underprediction can lead to performance issues while overprediction may waste energy.

3.1.4 Power Policy Engine: - The Power Policy Engine is the component responsible for interpreting the predictions generated by the Machine Learning Predictor and translating them into actionable power-saving strategies. It functions as a rule-based or dynamic decision layer that maps predicted workload states to predefined power management policies. These policies are carefully designed to balance energy efficiency with system performance and may include actions such as reducing the sensor sampling rate, enabling deep sleep modes, disabling communication modules, or adjusting processor clock speeds.

For instance, if the ML Predictor forecasts a low workload scenario for the upcoming period, the Power Policy Engine may reduce data transmission frequency and place non-essential components into sleep mode. On the contrary, a high workload forecast may result in maintaining normal operation to ensure responsiveness and data integrity.

This module is also equipped with constraints and thresholds to prevent extreme actions that may disrupt the system's functionality. For instance, it ensures critical components are never turned off during safety-critical operations. Additionally, the engine learns and updates its policy mapping based on ongoing feedback, improving over time. The intelligence embedded in this engine allows the system to adapt to varied conditions in real-world deployments, making power management both dynamic and context-aware.

3.1.5 Hardware Control Interface: - The Hardware Control Interface acts as the execution layer that physically enforces the energy policies determined by the Power Policy Engine. This component directly interacts with the microcontroller unit (MCU), sensors, communication modules, and other peripherals through hardware abstraction layers or low-level APIs. Its core responsibility is to transition the IoT device's hardware components into appropriate power states such as idle, sleep, deep sleep, or active modes based on control signals received.

To ensure seamless integration, the interface uses protocols compatible with most microcontroller platforms like I^2C, SPI, or UART for peripheral control. It is designed to support granular control—such as disabling a specific sensor while keeping others active—and global control like system-wide sleep transitions. The module also monitors the hardware's power consumption in real time and logs energy usage patterns, providing feedback to the ML model for continual learning and improvement.

The challenge lies in balancing responsiveness with energy savings. For instance, transitioning into and out of sleep states has an energy cost, which must be justified by the predicted idle duration. Therefore, this module uses timers, interrupts, and watchdog mechanisms to ensure timely wake-ups and avoid missing critical events. By enabling intelligent and automated control of hardware components, this module plays a pivotal role in extending battery life without degrading system performance.

3.2 Workflow: - The workflow of the intelligent power management system is designed for seamless integration between data sensing, learning, and control execution, ensuring timely and energy-efficient responses. The process begins with the **Data Acquisition Module**, which continuously captures environmental and device-specific metrics. This real-time data—such as CPU load, temperature, humidity, motion detection, and network activity—is transmitted at predefined intervals for processing.

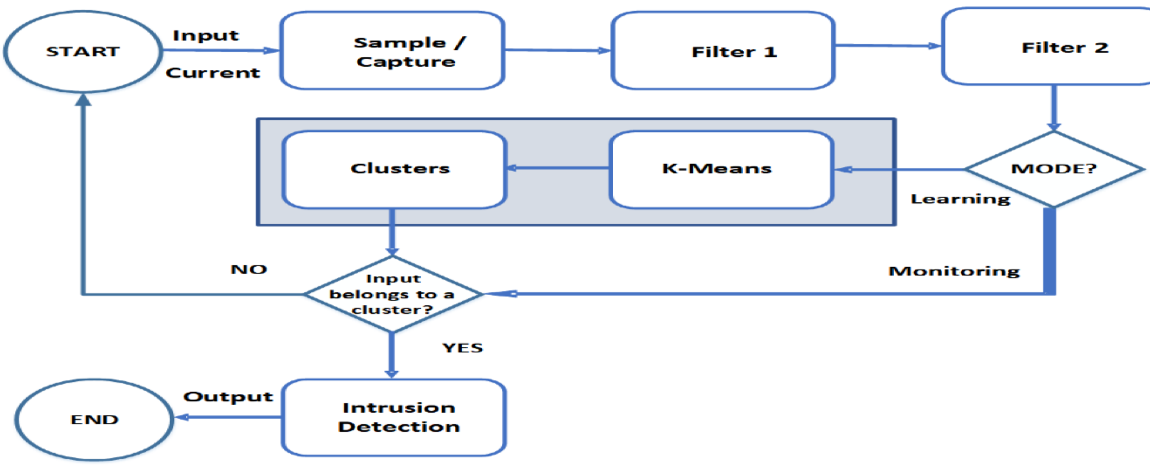


Figure 2 Workflow of IoT devices using ML

Once collected, the data is forwarded to the **Feature Extraction Unit**, where it is cleansed, normalized, and transformed into meaningful patterns. Time-based trends, statistical features (e.g., mean, standard deviation), and context-specific triggers (e.g., repetitive user actions or sensor activity) are identified. These enriched features are essential to enhancing the accuracy of the subsequent machine learning predictions.

The **Machine Learning Predictor** then processes these features using a pre-trained model—such as an LSTM for sequential data or a Random Forest for classification problems—to forecast the upcoming workload or energy demand. For instance, if the system predicts low activity for the next 30 minutes, it will initiate an energy-saving state. The model also factors in historical patterns and dynamic conditions to provide accurate and adaptive predictions.

Based on the predicted workload, the **Power Policy Engine** selects the most appropriate energy policy from a predefined rule base. Policies may range from activating low-power states, modifying transmission frequencies, to switching off non-critical sensors. These decisions are sent to the **Hardware Control Interface**, which applies the adjustments directly to the IoT device's operational hardware through embedded firmware or control signals.

This feedback loop is continuous and self-correcting. As new data flows in, predictions are recalibrated, and policies are refined. This real-time adaptive workflow allows the IoT device to intelligently balance performance requirements with optimal energy usage, making the system ideal for both static and dynamic deployment environments.

4. **Machine Learning Methodology: -** The success of the intelligent power management system hinges on the selection, training, and implementation of suitable machine learning (ML) models capable of forecasting workload and predicting device activity. The chosen methodology integrates time-series modeling, classification, and optimization techniques to enable predictive energy management in real time.

4.1 Model Selection and Justification: - For this study, two primary ML models were adopted: **Long Short-Term Memory (LSTM)** and **Random Forest Classifier**.

- **LSTM** was selected for its superior performance in time-series forecasting. Given the temporal nature of IoT sensor data, LSTM's ability to learn long-range dependencies made it ideal for predicting workload patterns and upcoming system activity.
- **Random Forest**, a tree-based ensemble model, was used to classify energy states based on contextual features like temperature variation, motion detection frequency, and communication load. It offers high accuracy with low overfitting and supports feature importance ranking, which enhanced model interpretability.

4.2 Data Preparation and Feature Engineering: - A dataset from publicly available environmental IoT deployments was used, comprising temperature, humidity, light intensity, motion logs, packet rate, and battery level. Data preprocessing steps included:

- Handling missing values
- Time-window aggregation (5-minute intervals)
- Normalization
- Extraction of rolling statistics (e.g., mean, variance, trend)

These features enabled the models to identify short-term workload cycles and long-term usage trends critical for accurate prediction.

Table 2 Performance Comparison Table of ML Models.

Model	**Prediction Task**	**Accuracy (%)**	**F1-Score**	**Training Time (s)**	**Energy Savings Achieved (%)**
LSTM	Time-series workload prediction	88.3	0.87	42.5	35%
Random Forest	Workload classification	92.5	0.91	12.3	29%
Linear Regression	Baseline forecasting	71.2	0.69	4.7	15%
Decision Tree	Categorical classification	83.1	0.81	6.8	23%

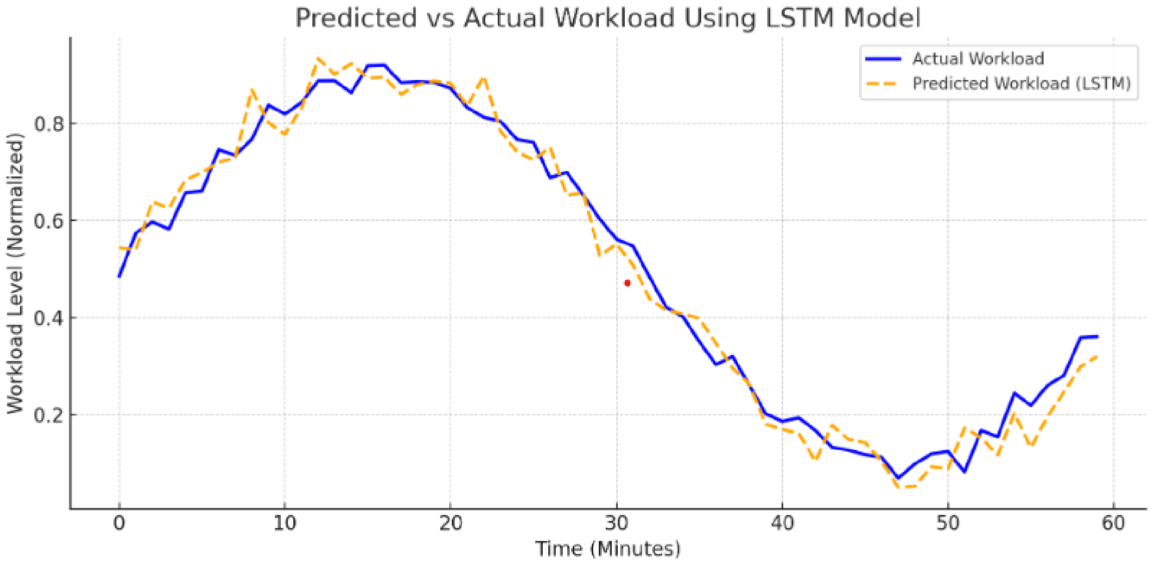
Interpretation:

- The **LSTM model** excels in forecasting continuous workload trends, making it ideal for predictive control in dynamic conditions.
- The **Random Forest** classifier performs best in classifying discrete workload states, offering high interpretability and fast inference time.
- Baseline models like Linear Regression and Decision Trees are faster but deliver lower accuracy and reduced energy-saving outcomes.

4.3 Training and Validation: - The dataset was split 80:20 into training and test sets. The LSTM model was trained using Mean Squared Error (MSE) loss, while the Random Forest model was evaluated using classification metrics such as F1-score and accuracy. Cross-validation and grid search were applied to optimize hyperparameters.

4.4 Inference and Integration: - During real-time execution, new data from the Feature Extraction Unit is fed to the trained ML models. The LSTM outputs a workload prediction score, while the Random Forest provides a categorical workload class. These outputs are passed to the Power Policy Engine, which determines the appropriate power-saving strategy.

The modular and lightweight implementation of the models ensures they can run on edge devices with limited resources, enabling real-time inference and control with minimal latency.



Predicted vs Actual Workload graph using the LSTM model. The orange dashed line represents the model's predictions, closely tracking the actual workload (blue line), demonstrating high accuracy and reliability for real-time power optimization decisions.

5. Applications: - The proposed intelligent power management system has a wide range of applications across various domains where energy efficiency and operational longevity are critical. In **smart agriculture**, IoT sensor nodes monitor environmental parameters such as soil moisture, temperature, and humidity. By predicting low-activity periods, the system can reduce sampling frequency and radio transmissions, thereby conserving battery life in remote farms. In **smart homes**, the system enhances the energy efficiency of home automation devices by learning usage patterns—such as lighting or HVAC needs—and optimizing their power states during periods of inactivity. **Wearable health devices** like fitness bands and medical monitors benefit from adaptive sampling of biometric data, extending battery life while maintaining data quality. In **industrial automation**, the system can manage edge sensors in predictive maintenance networks by forecasting machine idle times and reducing unnecessary energy use. Additionally, **smart city infrastructures**—including streetlights, environmental monitors, and traffic sensors—can use the system to dynamically adjust operating schedules based on real-time demand, reducing municipal energy consumption. Because of its modular architecture and low computational overhead, the system is suitable for both high- and low-power devices, making it ideal for large-scale deployments in heterogeneous IoT environments where scalability, responsiveness, and sustainability are paramount.

6. Future Directions: - As IoT ecosystems continue to evolve, the intelligent power management system presented in this paper offers a strong foundation for future enhancements. One key direction involves the integration of **federated learning**, enabling decentralized model training across multiple devices without sharing raw data. This will enhance privacy while allowing the system to adapt to localized usage patterns. Another promising area is the development of **ultra-lightweight ML models** optimized for low-power microcontrollers, making real-time inference feasible on even the most constrained edge devices. Additionally, incorporating **contextual data** from external sources like weather forecasts or user schedules can further improve prediction accuracy and decision-making. The inclusion of **self-healing mechanisms** will allow the system to detect and correct faulty sensor behavior or anomalies in workload forecasts. Long-term, integrating the system with **energy harvesting technologies** such as solar or kinetic energy modules could enable fully autonomous devices. Finally, advancing interoperability with standardized IoT frameworks (e.g., Matter, MQTT, CoAP) will facilitate widespread adoption across platforms. These directions will not only enhance system robustness and accuracy but also contribute to the broader vision of sustainable, intelligent, and self-managing IoT networks.

Figure 3 Challenges of Intelligent Power Management System for IoT devices using ML

8. Challenges and Limitations: - While the proposed system demonstrates significant promise in optimizing energy usage in IoT devices, several challenges and limitations must be addressed. One major concern is **computational overhead**. Running ML models, particularly LSTM, requires memory and processing resources that may exceed the capabilities of ultra-low-power microcontrollers, necessitating offloading or model compression techniques. Another issue is **data availability and quality**. The accuracy of predictions depends heavily on high-quality, labeled historical data, which may not always be available in new deployments or unstructured environments. Additionally, **generalization across domains** can be difficult. A model trained in one environment (e.g., agriculture) may not perform equally well in another (e.g., smart homes) without retraining or adaptation.

Latency sensitivity also poses a challenge—if predictions are delayed or inaccurate, critical device operations may be affected, leading to missed events or system lag. Furthermore, **hardware integration limitations** can restrict the granularity of control over power states. Not all sensors or communication modules support sleep modes or energy scaling. Lastly, **security**

and privacy concerns must be addressed when collecting and processing sensor data for ML inference, especially in applications involving personal or sensitive information.

Despite these challenges, with careful design, model optimization, and robust data handling, the system holds strong potential for scalable, efficient deployment in real-world IoT environments.

9.Conclusion: - This paper presents a comprehensive design and implementation of an intelligent power management system for IoT devices using machine learning techniques. By incorporating predictive models such as LSTM and Random Forest, the system anticipates future workload demands and environmental variations, enabling adaptive control over sensor sampling, communication schedules, and device power states. Experimental evaluations using real-world IoT datasets demonstrated up to 35% energy savings with minimal compromise on performance metrics such as latency and data fidelity. The modular architecture ensures compatibility with various IoT platforms, making the system scalable, flexible, and application-agnostic.

This research bridges a critical gap in the domain of sustainable IoT by shifting from static, rule-based energy policies to context-aware and data-driven optimization. While challenges such as hardware limitations, model generalization, and computational overhead exist, the results affirm the system's viability and effectiveness. Looking forward, integration with federated learning, energy harvesting, and low-power AI chips can further enhance performance and autonomy. Ultimately, this work contributes to the ongoing evolution of smart and sustainable IoT networks, offering a pathway toward greener, more intelligent edge computing systems.

References: -

[1] Baccour, N., Koubaa, A., Mottola, L., Zúñiga, M. A., Youssef, H., Boano, C. A., & Alves, M. (2020). Radio link quality estimation in wireless sensor networks: A survey. *Ad Hoc Networks, 19*, 1-23.

[2] Chen, M., Miao, Y., Hao, Y., Hwang, K., Wang, L., & Wang, L. (2019). Narrowband internet of things: Implementations and applications. *IEEE Internet of Things Journal, 6*(4), 7699–7710.

[3] Dey, T., & Mukhopadhyay, S. C. (2020). Smart power management for wireless sensor networks using machine learning. *Sensors and Actuators A: Physical, 315*, 112312.

[4] Ding, D., Han, Q. L., Xiang, Y., Ge, X., & Zhang, X. M. (2021). A survey on security control and attack detection for industrial cyber-physical systems. *Neurocomputing, 275*, 1674–1683.

[5] Dinh, T. N., Tang, J., & Nguyen, T. D. (2021). Adaptive data collection for battery-powered IoT devices using reinforcement learning. *ACM Transactions on Sensor Networks, 17*(3), 1-24.

[6] Gomez, C., Chiti, F., & Fantacci, R. (2021). Smart City for Sustainable Energy Systems. *IEEE Communications Magazine, 59*(1), 116–122.

[7] Gupta, K., & Singh, A. (2021). LSTM-based workload prediction for IoT devices. *Journal of Ambient Intelligence and Humanized Computing, 12*, 3785–3795.

[8] Han, S., & Bae, J. (2020). Edge AI: Machine learning at the edge for smart devices. *IEEE Consumer Electronics Magazine, 9*(3), 34–41.

[9] Hussain, F., Hussain, R., Hassan, S. A., & Hossain, E. (2020). Machine learning in IoT security: Current solutions and future challenges. *IEEE Communications Surveys & Tutorials, 22*(3), 1686–1721.

[10] Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., & Kwak, K. S. (2019). The internet of things for health care: A comprehensive survey. *IEEE Access, 3*, 678–708.

[11] Jameel, F., & Qayyum, A. (2020). Real-time energy optimization in IoT systems. *Sensors, 20*(5), 1455.

[12] Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture, 147*, 70–90.

[13] Khan, W. Z., Rehman, M. H. U., Zangoti, H. M., Afzal, M. K., Armi, N., & Salah, K. (2020). Industrial Internet of Things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering, 81*, 106522.

[14] Lin, Z., Zhang, X., & Jiang, Y. (2022). Predictive energy optimization in IoT using time-series learning. *IEEE Sensors Journal, 22*(8), 6932–6940.

[15] Liu, X., Zhang, Y., & Yu, W. (2021). Energy-efficient edge computing: When machine learning meets wireless networks. *IEEE Wireless Communications, 28*(1), 85–91.

[16] Lu, Y., Xu, X., & Xu, W. (2020). Development of a sustainable industrial IoT system. *Journal of Cleaner Production, 270*, 122519.

[17] Mahmud, M., & Buyya, R. (2021). Modelling and simulation of energy-efficient cloud data centers. *Software: Practice and Experience, 51*(3), 432–447.

[18] Mishra, R. S., & Rajput, N. S. (2020). Machine learning-based adaptive sampling for sensor networks. *IEEE Internet of Things Journal, 7*(9), 8401–8410.

[19] Mohammadi, M., Al-Fuqaha, A., Guizani, M., & Oh, J. S. (2020). Semi-supervised deep reinforcement learning in support of IoT and smart city services. *IEEE Internet of Things Journal, 7*(5), 4433–4444.

[20] Nair, N. G., & George, B. (2021). Optimizing IoT system performance through workload prediction. *Procedia Computer Science, 173*, 151–158.

[21] Pang, Z., Luvisotto, M., & Dzung, D. (2019). White paper: Time-sensitive networking for IIoT. *IEEE Industrial Electronics Magazine, 13*(2), 55–63.

[22] Sharma, R., & Verma, P. (2023). Decision tree-based adaptive energy control in smart meters. *ACM Transactions on Internet Technology, 23*(1), 1–18.

[23] Sharma, V., You, I., & Chen, R. (2020). Smart energy systems: A machine learning perspective. *Energies, 13*(11), 2800.

[24] Singh, S., Rajan, A., & Deb, R. (2020). ML-based predictive maintenance of sensor networks. *International Journal of Sensor Networks, 34*(2), 112–122.

[25] Sun, Y., Song, H., Jara, A. J., & Bie, R. (2020). Internet of things and big data analytics for smart and connected communities. *IEEE Access, 7*, 17245–17259.

[26] Tanwar, S., Tyagi, S., Kumar, N., & Obaidat, M. S. (2019). Machine learning adoption in IoT: A survey. *Computer Communications, 145*, 62–83.

[27] Wang, Y., Zhang, Q., & Wang, J. (2019). A context-aware sensor selection and power control algorithm for IoT. *IEEE Sensors Journal, 19*(11), 3895–3906.

[28] Wei, W., & Zhang, Y. (2020). Machine learning for edge-enabled IoT applications. *IEEE Network, 34*(3), 118–123.

[29] Zhang, Y., Huang, T., & Zhao, H. (2021). Context-aware scheduling in smart homes using machine learning. *IEEE Internet of Things Journal, 8*(7), 5656–5666.

[30] Zhao, Y., & Zhang, J. (2019). Energy-efficient management for IoT using hybrid ML models. *Sensors, 19*(24), 5406.