Scientific Research Publishing

Journal of Power and Energy Engineering, 2025, 13(8), 293-306
https://www.scirp.org/journal/jpee
ISSN Online: 2327-5901
ISSN Print: 2327-588X

Comparative Study of Machine Learning Models for Load Prediction and Energy Management

Ainain Nur Hanafi[1]*, Noor Ashiqin Nor Azli[1], Aida Fazliana Abdul Kadir[1], Hussain Shareef[2]

[1]Faculty of Electrical Technology and Engineering, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia
[2]Electrical and Communication Engineering Department, United Arab Emirates University, Al Ain, United Arab Emirates
Email: *ainain@utem.edu.my

How to cite this paper: Hanafi, A.N., Nor Azli, N.A., Abdul Kadir, A.F. and Shareef, H. (2025) Comparative Study of Machine Learning Models for Load Prediction and Energy Management. *Journal of Power and Energy Engineering*, **13**, 293-306.
https://doi.org/10.4236/jpee.2025.138016

Received: July 27, 2025
Accepted: August 25, 2025
Published: August 28, 2025

Abstract

Accurate energy load prediction is crucial for optimizing energy management in smart grid systems. This study evaluates the performance of four machine learning models, which are random forest, gradient boosting, support vector regression (SVR), and linear regression, for load prediction using a dataset from Universiti Teknikal Malaysia Melaka (UTeM). The dataset consists of 21 months of hourly energy consumption data, including photovoltaic (PV) generation, battery storage, and grid meter readings. Among the models tested, gradient boosting model achieved the highest accuracy with an R^2 of 0.72, demonstrating its effectiveness in forecasting energy demand. Random forest model exhibited strong training performance but suffered from overfitting, while SVR and linear regression models showed lower predictive accuracy. The predicted load values were integrated into an if-then rule-based control strategy for managing energy distribution among PV, battery, and grid sources. The findings highlight the potential of machine learning in enhancing energy efficiency by improving demand forecasting and optimizing resource allocation.

Keywords

Machine Learning, Load Prediction, Energy Management, Gradient Boosting

1. Introduction

The increasing global demand for energy has driven the need for more efficient and sustainable energy management solutions. Renewable energy sources, such as photovoltaic (PV) systems, are widely integrated into power grids to reduce reliance on conventional electricity supply and minimize costs [1]. However, the intermittent nature of renewable energy introduces challenges in predicting energy generation and consumption, making efficient energy management crucial [2]. Accurate load prediction is essential for optimizing energy distribution, ensuring grid stability [3] and maximizing the utilization of renewable sources [4]. This study addresses these challenges by conducting a comparative analysis of several machine learning models for load prediction and integrating the most effective model into an intelligent energy management framework. As highlighted in [5], accurate load forecasting is vital for energy conservation, enabling effective strategies for environmental protection and reduced carbon emissions.

This research investigates four prominent machine learning models, which are linear regression, support vector regression (SVR), random forest and gradient boosting. The linear regression model assumes a linear relationship between the input features (e.g. time of day, weather conditions) and the energy load. While simple to implement and interpret, it serves as a valuable baseline for comparison and has been widely used in long-term forecasting [6]. However, its assumption of a linear relationship can limit its effectiveness in capturing complex energy consumption patterns. It often struggles to capture nonlinear dependencies and seasonal variations in energy consumption data, reducing their predictive accuracy [7]. On the other hand, other machine learning approaches have gained significant attention due to their ability to model complex energy patterns. For instance, random forest, an ensemble learning method, constructs multiple decision trees during training and outputs the average prediction of the individual trees. Random forest is robust against overfitting and capable of handling high dimensionality, making it suitable for complex datasets [8]. However, it can be less interpretable than simpler models like linear regression. Next, SVR is known for its ability to handle non-linear relationships between features and load by using kernel functions to map the data into a higher-dimensional space. Moreover, it is known for its ability to generalize well even with limited data but can be computationally intensive, making it suitable for long-term load forecasting [9]. While gradient boosting, an ensemble learning method, iteratively improves predictions by minimizing errors, leading to high accuracy in time-series analysis [10]. Despite these advancements, several challenges remain. Many studies focus on a single machine learning model without conducting a comparative analysis under consistent conditions. Additionally, research on integrating machine learning-based forecasting with real-time energy management strategies, such as optimizing battery charging and minimizing grid dependency, remains limited [11].

To address these limitations, this study conducts a comparative evaluation of four machine learning models: Linear Regression, Random Forest, Support Vector Regression, and Gradient Boosting. The goal is to compare how well different machine learning models can predict the future and determine the best way to use machine learning-based predictions in an intelligent energy management framework. By optimizing load prediction, this study aims to enhance decision-making in renewable energy utilization, ensuring efficient allocation of photovoltaic power, battery, and grid power. The findings will contribute to the development of smart energy management systems, improving both sustainability and cost-effectiveness in line with SDG-7.

2. Methodology

2.1. Overview

This section outlines the methodology used for energy load prediction, covering data collection, preprocessing, algorithm training, performance evaluation, and integration into the energy management system (EMS). A structured approach is followed to ensure a reliable comparative analysis of machine learning models. The data set consists of 21 months of hourly energy consumption data from the buildings at Faculty of Technology and Electrical Engineering (FTKE), UTeM, incorporating photovoltaic generation, battery storage, and grid meter readings. Four machine learning models, namely Random Forest, Gradient Boosting, Support Vector Regression, and Linear Regression, are evaluated. These models are trained using different techniques to enhance robustness and accuracy. The best-performing model is subsequently integrated into an if-then rule-based energy management system to optimize energy distribution. The implementation is carried out using Python in the JupyterLab environment. **Figure 1** illustrates the overall methodology, detailing the steps from data collection to the final integration of machine learning predictions into the energy management system.

2.2. Data Preprocessing

The dataset comprised 21 months of hourly energy consumption records, where each entry corresponded to a specific hourly timestamp. The input features included: Time (timestamp), Temp (ambient temperature in degrees Celsius), PV (photovoltaic power generation in watts), SOC (state of charge of the battery in percentage), Battery (power stored or drawn from the battery in watts), and Meter (power consumed from the grid in watts). An additional target feature, Load, was computed using the relationship: Load = PV + Battery − Meter.

Initial preprocessing involved converting the Time column from an object data type to datetime format. From this, temporal features such as Hour, Day, and Month were extracted to capture daily and seasonal patterns. Furthermore, a binary categorical feature, Weekday/Weekend, was derived using the day-of-week indicator, where weekdays (Monday to Friday) were encoded as 0 and weekends (Saturday and Sunday) as 1, to facilitate learning by machine learning models.

The dataset was examined for data types, completeness, and consistency using the *info*() and *isnull*().*sum*() functions. No missing values were identified, and thus, imputation was not required. The numerical features (Temp, PV, SOC, Battery, and Meter) were retained in their native formats but scaled using either standard normalization or min-max scaling, depending on the requirements of the respective learning algorithm.

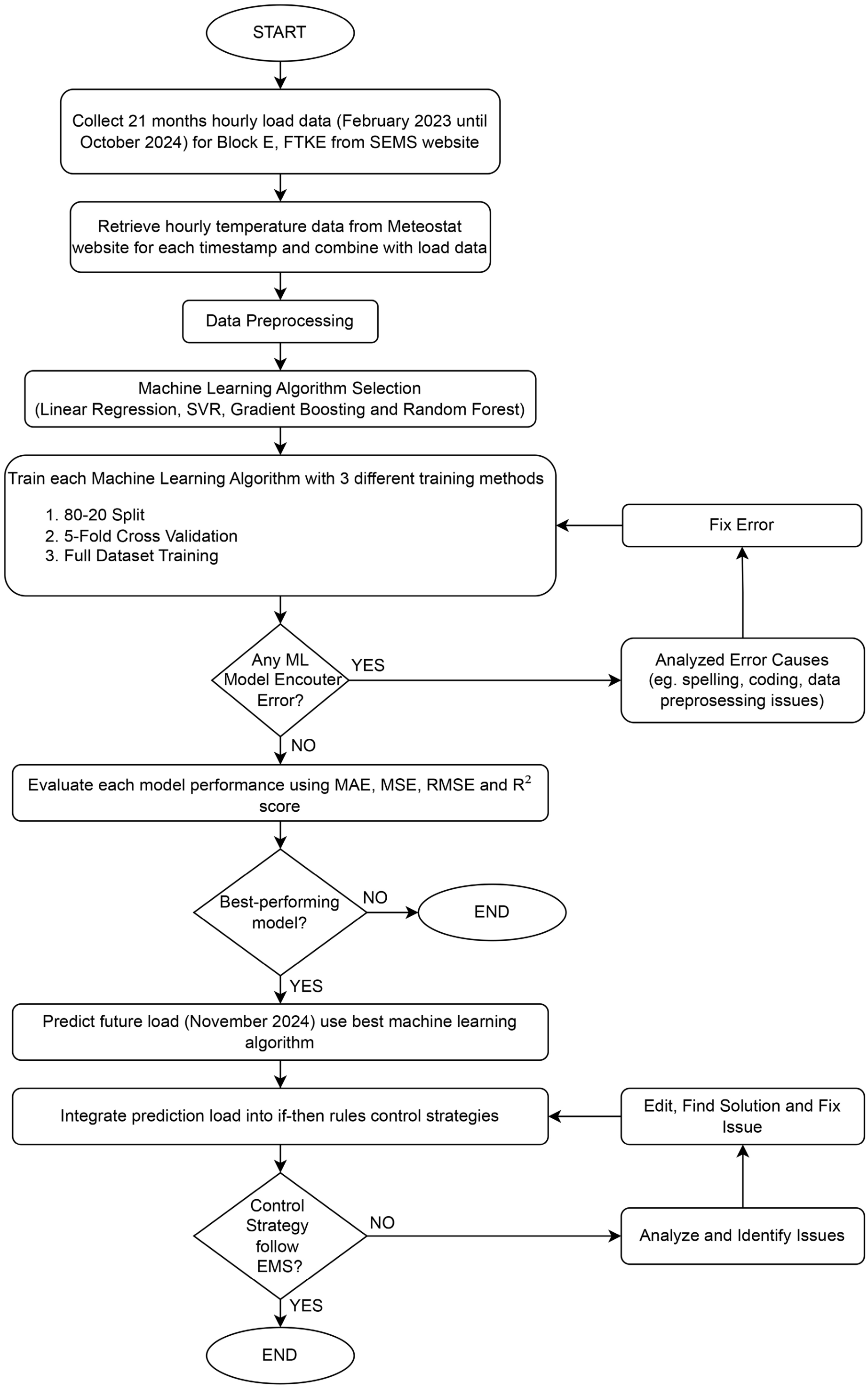

Figure 1. Load prediction flowchart.

2.3. Algorithm Training Techniques

Three different training techniques were applied separately to evaluate the predictive performance of the machine learning models, ensuring a comprehensive assessment before making comparisons.

2.3.1. 80 - 20 Train-Test Split

In this approach, the dataset is divided into two sets, with eighty percent (80%) allocated for training, allowing the model to learn patterns from historical data, and twenty percent (20%) reserved for testing, where the model is evaluated on unseen future data. Since the data is split chronologically, the model is tested on a dataset that follows the training period, ensuring that the evaluation mimics real-world forecasting conditions. This technique helps assess how well the model generalizes to future unseen data.

2.3.2. 5-Fold Cross-Validation

In this technique, the dataset is divided into five equal subsets, also known as folds. In each iteration, one-fold serves as the test set, while the remaining four folds are used for training. This process is repeated five times, ensuring that each fold is used as a test set once. The final model performance is averaged across all five runs, reducing variability in the results. Unlike the eighty-twenty split, which evaluates performance on a fixed unseen portion, cross-validation allows the model to be tested on multiple subsets, providing a more robust evaluation. This technique is particularly useful for detecting overfitting and improving model stability.

2.3.3. Full Dataset Training

This approach used the entire dataset for training, eliminating the need for a separate test set. This technique means the model learns from all available data, capturing as many patterns as possible. However, since no independent test set is available, this method does not provide an unbiased assessment of the model's performance on unseen data. Instead, it ensures the model has the maximum possible training data before making final predictions.

All models were trained using default hyperparameter settings provided by their respective libraries. No additional hyperparameter tuning was performed, as the primary aim of this study was to compare the baseline performance of each algorithm under standard configurations. While tuned models may yield improved accuracy, this approach ensures a fair, out-of-the-box comparison and avoids the risk of overfitting during the tuning process. All three training techniques were applied to each machine learning algorithm, and the results were compared to determine the best-performing model based on key performance metrics.

2.4. Performance Metrics

The trained models were evaluated using four standard regression metrics [7], computed using Python-based programming. The following metrics quantify the predictive accuracy of each model.

Mean absolute error measures the average absolute difference between predicted and actual values, providing a straightforward evaluation of prediction accuracy.

$$\text{MAE} = \frac{\sum_{i=1}^{n}(y_i - x_i)}{n} \tag{1}$$

Root means squared error, which is the square root of mean squared error, represents the error magnitude in the same unit as the target variable, making it easier to interpret.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{x}_i)^2}{n}} \tag{2}$$

The R-squared score (coefficient of determination) quantifies how well the model explains variance in the dataset. A value close to one point zero indicates high predictive accuracy, while a value near zero suggests poor model performance.

2.5. Energy Management System via IF-THEN Rules

The energy management system integrates an if-then rule-based strategy to optimize energy distribution based on real-time and predicted load data. The primary objective is to maximize renewable energy utilization from photovoltaic sources and battery storage while minimizing reliance on grid power. This approach reduces operational costs and enhances energy efficiency.

The energy management system decides how to distribute energy based on four main factors: actual load, which shows how much energy is being used in real time; predicted load, which shows how much energy is expected to be needed one hour from now; state of charge, which shows how much energy is stored in batteries; and available photovoltaic power, which shows how much energy is being made by photovoltaic panels in real time. To improve energy storage performance, the battery capacity was increased from 10 kW to 13.5 kW, ensuring better system reliability during periods of low photovoltaic availability or high demand.

The energy management system applies to a multi-case decision-making approach based on state of charge levels, predicted loads, and actual loads. Three cases are considered as follows. In the first case, when the battery needs charging, the system checks if the state of charge is below 40% and whether the predicted load is higher than the actual load. If these conditions are met, priority is given to charging the battery using available photovoltaic power. However, if photovoltaic power is insufficient, the grid supplements the charging process. For the second case, where the battery does not need charging, the system evaluates whether the state of charge is below 40% and if the predicted load is less than or equal to the actual load. Under these conditions, photovoltaic power is first used to meet the actual load demand. The battery is charged only if excess photovoltaic power is available. If a power deficit occurs, the battery is discharged. The system turns to grid power if photovoltaic and battery power are both insufficient. Lastly, the third case, which represents normal operation when the state of charge is 40% or higher, the system follows a hierarchical approach. If photovoltaic power is sufficient, it meets the actual load demand and charges the battery with any excess energy. If photovoltaic power proves insufficient, the system discharges the battery to meet the load. If both photovoltaic and battery power prove insufficient, the system resorts to using grid power as a final measure.

These if-then rules are embedded into the energy management system to ensure efficient energy allocation, following a priority order where photovoltaic power is utilized first, followed by battery storage, and finally, grid power. The system dynamically adjusts battery usage and charging strategies based on real-time and predicted energy requirements, thereby minimizing grid consumption and optimizing the battery state of charge. This approach enhances sustainability while reducing overall energy costs.

3. Results and Discussion

This section presents the findings of the study, focusing on the selection of the most suitable machine learning model for energy load prediction and its application in the energy management system. The performance of different models is analyzed based on key evaluation metrics, highlighting their strengths and limitations. Additionally, the integration of predictive insights into the energy management framework is discussed, demonstrating how the selected model aids in optimizing energy distribution and minimizing reliance on the grid.

3.1. Performance of Machine Learning Models

The Linear Regression model demonstrated moderate effectiveness in predicting energy load but showed notable limitations due to its assumption of linearity. The model captured general trends but lacked the flexibility to handle the nonlinear nature of energy consumption patterns. Table 1 summarizes the Linear Regression model's performance under different training methods.

Table 1. Performance metrics of the linear regression model.

Training Method	MAE (W)	MSE (W)	RMSE (W)	R^2 Score
80 - 20 Split	3119.34	17952437.57	4237.03	0.43
5-Fold CV	3054.57	14601086.59	3821.14	0.39
Full Dataset	2812.86	14416606.75	3796.92	0.43

With an R^2 score ranging from 0.39 to 0.43, the model explained only 39% - 43% of the variance in energy load, indicating limited predictive power and inability to capture complex relationships between temperature, time, and other dynamic factors. Next, the Support Vector Regression (SVR) model, utilizing a radial basis function (RBF) kernel, effectively captured non-linear patterns in energy load. However, its ability to generalize varied across different training methods, as detailed in Table 2.

Table 2. Performance metrics of the SVR model.

Training Method	MAE (W)	MSE (W)	RMSE (W)	R^2 Score
80 - 20 Split	2088.78	12756897.50	3571.68	0.60
5-Fold CV	2030.01	11468889.46	3386.57	0.55
Full Dataset	1901.20	10442661.16	3231.51	0.59

While SVR improved predictive accuracy compared to Linear Regression, it struggled with extreme load fluctuations due to kernel constraints. The full dataset training resulted in the best performance, but independent validation was limited, highlighting the need for further optimization. On the other hand, the Gradient Boosting model exhibited stronger predictive capability across all training methods, as shown in **Table 3**. With an R^2 score of up to 0.70, Gradient Boosting outperformed both SVR and Linear Regression. The full dataset training yielded the lowest error metrics, suggesting its superior ability to capture complex dependencies in the data. However, independent testing is necessary to confirm generalizability.

Table 3. Performance metrics of the Gradient Boosting model.

Training Method	MAE (W)	MSE (W)	RMSE (W)	R^2 Score
80 - 20 Split	2088.78	12756897.50	3571.68	0.60
5-Fold CV	2030.01	11468889.46	3386.57	0.55
Full Dataset	1901.20	10442661.16	3231.51	0.59

Random Forest demonstrated the highest accuracy when trained on the full dataset, achieving an R^2 score of 0.96. However, its generalisability was lower under the 80 - 20 split and 5-fold cross-validation, as detailed in **Table 4**. While the full dataset approach provided outstanding results, its high R^2 score suggests potential overfitting. The 80 - 20 split and cross-validation methods, though less accurate, offer more reliable insights into real-world performance.

Table 4. Performance metrics of the Random Forest model.

Training Method	MAE (W)	MSE (W)	RMSE (W)	R^2 Score
80 - 20 Split	2043.53	12017665.71	3466.65	0.62
5-Fold CV	2054.20	10916864.52	3304.07	0.57
Full Dataset	576.30	994998.70	997.50	0.96

The Linear Regression model demonstrated the weakest performance, with the highest error values and the lowest R^2 scores across all training methods. It struggled to capture variations in energy consumption effectively, leading to limited predictive accuracy. For instance, using the 80 - 20 split, it achieved an MAE of

3119.34 W with an R^2 score of 0.43. Although the performance improved slightly with full dataset training (MAE = 2812.86 W), this approach lacks independence between training and testing data, reducing its reliability for future predictions. Support Vector Regression (SVR) performed significantly better than Linear Regression, achieving lower error values and higher R^2 scores. With the 80 - 20 split, SVR recorded an MAE of 2088.78 W and an RMSE of 3571.68 W, with an R^2 score of 0.60. Its performance further improved with 5-fold cross-validation approach, demonstrating better consistency. However, despite its improved accuracy, SVR was slightly outperformed by Gradient Boosting, particularly in handling complex energy consumption patterns. Gradient Boosting emerged as the best-performing model, with the lowest error values and the highest R^2 scores across different training methods. Using the 80 - 20 split, it achieved an MAE of 2013.80 W and an RMSE of 3187.72 W, with an R^2 score of 0.68. Performance further improved with 5-fold cross-validation (MAE = 1909.57 W, RMSE = 2941.17 W, R^2 = 0.66). While full dataset training produced even lower errors (MAE = 1786.96 W, R^2 = 0.70), this approach risks overfitting, limiting its ability to generalize to unseen data. Random Forest also demonstrated strong predictive capability, particularly in independent training methods. With the 80 - 20 split, it achieved an MAE of 2043.53 W and an RMSE of 3466.65 W, with an R^2 score of 0.62, performing better than SVR but slightly below Gradient Boosting. However, when trained on the full dataset, Random Forest achieved an exceptionally low MAE of 576.30 W and an R^2 score of 0.96, indicating potential overfitting.

3.2. Model Performance Comparison Using Paired T-Test

Pairwise t-tests were conducted to evaluate whether the performance differences between Gradient Boosting (GB) and other models were statistically significant. The paired t-test is appropriate because all models were evaluated on the same data splits, allowing for a fair comparison. A standard significance level of 0.05 was used. A p-value below 0.05 indicates a statistically significant difference in performance. To ensure a robust and fair evaluation, all t-tests were conducted using results from 5-fold cross-validation rather than a single 80 - 20 train-test split. This approach yields five MAE scores per model, enhancing statistical reliability. Importantly, the folds preserved the temporal order of the data which is essential for time-dependent forecasting tasks. By evaluating model performance across multiple, chronologically ordered segments, this method reduces the risk of bias from a single split and provides a more consistent basis for statistical comparison.

The comparison between Gradient Boosting and SVR yielded $t = 2.307$ with $p = 0.0823$, indicating that the difference was not statistically significant at the 0.05 level. Thus, there is no strong evidence that Gradient Boosting outperforms SVR. In contrast, the Gradient Boosting vs. Random Forest comparison resulted in $t = 3.185$ and $p = 0.0334$, suggesting a statistically significant improvement in performance by Gradient Boosting over Random Forest. The most pronounced difference was observed between Gradient Boosting and Linear Regression, with t = 24.198 and p = 0.00002, providing strong statistical evidence that Gradient Boosting significantly outperforms Linear Regression. The results are presented in Table 5.

Table 5. Paired T-Test on MAE in 5-fold cross-validation.

Fold	Gradient Boosting	SVR	Random Forest	Linear Regression
1	1855.28	1817.72	2160.54	2866.41
2	1782.63	1876.43	1892.07	3010.41
3	1775.29	2065.46	1888.60	2885.84
4	2128.74	2323.25	2292.61	3229.42
5	2005.93	2099.39	2037.19	3280.75

From the comparative analysis, the Gradient Boosting model was identified as the most suitable model for predicting energy consumption. It consistently outperformed the Linear Regression, Support Vector Regression (SVR), and Random Forest, achieving the lowest error values and the highest R^2 scores across independent evaluation methods. Its ability to generalize well with unseen data makes it the most reliable choice for forecasting future energy load. While Random Forest achieved an exceptionally high R^2 score of 0.96 during full dataset training, this was likely due to overfitting, as its performance significantly dropped when evaluated using independent methods, with R^2 scores of 0.62 and 0.57 for the 80 - 20 split and 5-fold cross-validation, respectively. This overfitting issue became more evident when Random Forest was used for November 2024 predictions.

In contrast, Gradient Boosting demonstrated strong predictive capability for November 2024, producing consistent and accurate results. The actual vs. predicted load values using Gradient Boosting are visualised in Figure 2, illustrating how well the model captures energy consumption patterns. The detailed performance metrics for Gradient Boosting predictions are shown in Table 6, further confirming its effectiveness in minimising prediction errors and explaining 73% of the variance in energy load. Based on these findings, Gradient Boosting is the best model for this study and will be used in the next phase for integrating energy management strategies and decision-making.

Table 6. Performance metrics of the random forest and gradient boosting models on November 2024 dataset.

Machine Learning Model	MAE (W)	MSE (W)	RMSE (W)	R^2 Score
Random Forest	1942.00	9629520.70	3103.15	0.65
Gradient Boosting	1839.07	7558345.78	2749.24	0.73

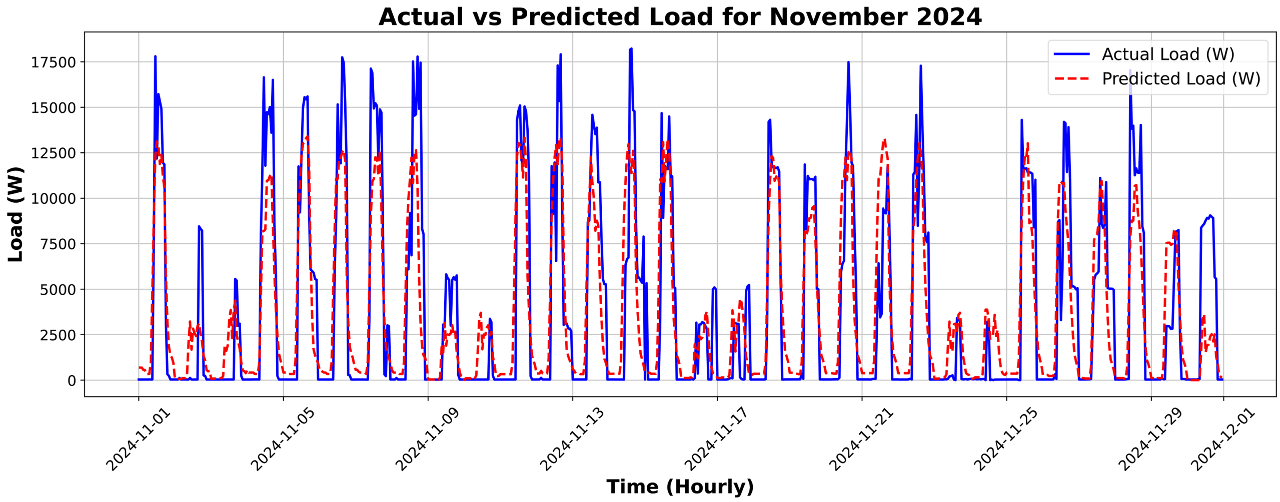

Figure 2. Actual vs predicted load for November 2024 using Gradient Boosting model.

3.3. If-Then EMS Rules

The If-Then rules approach is used to manage energy distribution among PV (solar power), battery, and the meter (grid) based on predefined conditions. These rules help prioritize energy sources to minimize grid dependency and optimize energy usage. The system follows the conditions stated in the methodology and uses predicted load values generated by the Gradient Boosting model for November 2024.

In Case 1, where the state of charge (SOC) is initially below 40% and the predicted load is equal to or lower than the actual load, the system begins with an SOC of 30%, a predicted load of 3477.25 W, and an actual load of 10,882 W, while the photovoltaic (PV) generation is limited to 33 W. The system utilizes 33 W from the PV and discharges 4050 W from the battery. However, the total available energy of 4083 W is insufficient to meet the actual demand, necessitating the grid to supply the remaining 6799 W. As a result, the battery is fully depleted, reducing the SOC to 0%, and the control strategy does not prioritize battery charging under this condition.

In Case 2, where the SOC remains below 40% but the predicted load is higher than the actual load, the system operates with an initial SOC of 25%, a predicted load of 348.30 W, and an actual load of only 34 W, while the PV generation is 10 W. The system initially supplies 10 W from the PV to meet part of the load demand, with the remaining 24 W drawn from the grid. Due to the significant discrepancy between the low actual load and the high predicted load, the system takes the opportunity to recharge the battery. Consequently, the battery is charged at a high rate of 10,125 W, achieving full charge (100% SOC), which results in a substantial net energy draw from the grid, totaling −10,149 W.

In Case 3, where the SOC is above 40% and the predicted load is lower than the actual load, the system starts with an SOC of 50%, a predicted load of 2245.38 W, and an actual load of 3053.29 W, with PV generation reaching 4000 W. The entire load is supplied directly from the PV, utilizing 3053.29 W. The surplus PV energy, amounting to 946.71 W, is used to charge the battery, resulting in an increase in SOC to 57.01%. Since the PV generation fully satisfies both the load and battery charging requirements, the meter records zero energy exchange with the grid.

In Case 4, where the SOC is exactly 40% and the predicted load is higher than the actual load, the system is presented with a predicted load of 995.80 W, while the actual load is only 31.86 W, and no PV generation is available. The load is supplied entirely by the battery, which slightly discharges to support the system, causing the SOC to decrease marginally to 39.7%. As the battery alone is sufficient to meet the demand, there is no energy drawn from or exported to the grid, and the meter reading remains at 0 W. The If-Then rules ensure efficient energy allocation based on real-time conditions. The system prioritizes PV power, discharges the battery when needed, and only uses the meter as a last resort. This method provides precise and predictable control, making it more reliable for energy management, as shown in Figure 3.

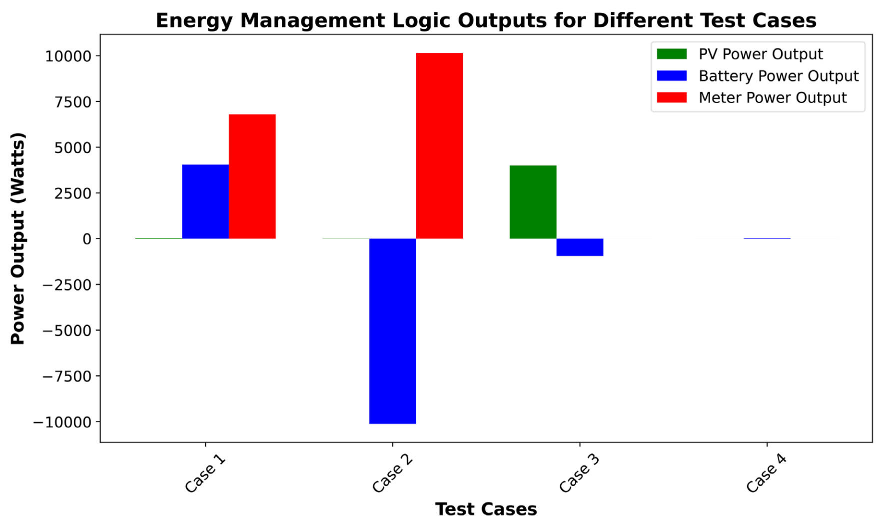

Figure 3. Energy management outputs for different test cases.

4. Conclusion

This study explored energy management optimization using machine learning models and rule-based decision-making. Various machine learning techniques were employed to predict energy load, including Linear Regression, Random Forest, Support Vector Regression, and Gradient Boosting. Among these, Random Forest initially showed the best performance, achieving the highest accuracy. However, further independent testing revealed signs of overfitting, indicating that the model performed exceptionally well on training data but struggled to generalize to unseen data. After further evaluation, Gradient Boosting emerged as the best-performing model, offering a more balanced performance with improved generalization and lower error rates. The predicted energy load from this model was then integrated into an if-then rule-based system to optimize energy distribution. This system prioritized photovoltaic (PV) energy usage, managed battery charging efficiently, and minimized reliance on the grid, ultimately reducing energy costs. Future work may focus on refining the predictive model with additional real-time data, improving the rule-based logic with adaptive optimization techniques, and exploring hybrid AI-driven energy management systems. The findings from this study contribute to the advancement of intelligent energy management systems, paving the way for more sustainable and cost-effective solutions.

Acknowledgements

The authors thank the Universiti Teknikal Malaysia Melaka for laboratory facilities and financial support. This work is funded by the Short Term Grant PJP/2024/FTKE/PERINTIS/SA0009.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Mystakidis, A., Koukaras, P., Tsalikidis, N., Ioannidis, D. and Tjortjis, C. (2024) Energy Forecasting: A Comprehensive Review of Techniques and Technologies. *Energies*, **17**, 1662. https://doi.org/10.3390/en17071662

[2] Akdemir, H., Durusu, A., Erduman, A. and Nakir, I. (2018) Effect of Energy Management of a Grid Connected Photovoltaic/Battery/Load System on the Optimal Photovoltaic Placement on a National Scale: The Case of Türkiye. *Journal of Solar Energy Engineering*, **140**, Article ID: 021009. https://doi.org/10.1115/1.4039077

[3] MacMackin, N., Miller, L. and Carriveau, R. (2019) Modeling and Disaggregating Hourly Effects of Weather on Sectoral Electricity Demand. *Energy*, **188**, Article ID: 115956. https://doi.org/10.1016/j.energy.2019.115956

[4] Kobylinski, P., Wierzbowski, M. and Piotrowski, K. (2020) High-Resolution Net Load Forecasting for Micro-Neighbourhoods with High Penetration of Renewable Energy Sources. *International Journal of Electrical Power & Energy Systems*, **117**, Article ID: 105635. https://doi.org/10.1016/j.ijepes.2019.105635

[5] Guo, N., Chen, W., Wang, M., Tian, Z. and Jin, H. (2021) Appling an Improved Method Based on ARIMA Model to Predict the Short-Term Electricity Consumption Transmitted by the Internet of Things (IoT). *Wireless Communications and Mobile Computing*, **2021**, 1–11. https://doi.org/10.1155/2021/6610273

[6] Hammad, M.A., Jereb, B., Rosi, B. and Dragan, D. (2020) Methods and Models for Electric Load Forecasting: A Comprehensive Review. *Logistics & Sustainable Transport*, **11**, 51-76. https://doi.org/10.2478/jlst-2020-0004

[7] Fumo, N. and Rafe Biswas, M.A. (2015) Regression Analysis for Prediction of Residential Energy Consumption. *Renewable and Sustainable Energy Reviews*, **47**, 332-343. https://doi.org/10.1016/j.rser.2015.03.035

[8] Lin, W., Wu, Z., Lin, L., Wen, A. and Li, J. (2017) An Ensemble Random Forest Algorithm for Insurance Big Data Analysis. *IEEE Access*, **5**, 16568-16575. https://doi.org/10.1109/access.2017.2738069

[9] Lin, K., Lin, Q., Zhou, C. and Yao, J. (2007) Time Series Prediction Based on Linear

Regression and SVR. 3*rd International Conference on Natural Computation* (*ICNC* 2007), Haikou, 24-27 August 2007, 688-691. https://doi.org/10.1109/icnc.2007.780

[10] Aziz, N., Akhir, E.A.P., Aziz, I.A., Jaafar, J., Hasan, M.H. and Abas, A.N.C. (2020) A Study on Gradient Boosting Algorithms for Development of AI Monitoring and Prediction Systems. 2020 *International Conference on Computational Intelligence* (*ICCI*), Bandar Seri Iskandar, 8-9 October 2020, 11-16. https://doi.org/10.1109/icci51257.2020.9247843

[11] Lee, D. and Cheng, C. (2016) Energy Savings by Energy Management Systems: A Review. *Renewable and Sustainable Energy Reviews*, **56**, 760-777. https://doi.org/10.1016/j.rser.2015.11.067