WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

AI-BASED SMART BUILDING ENERGY MANAGEMENT SYSTEM USING DEEP LEARNING AND EXPLAINABLE AI

KAVIYA S[1], DR. JAI RUBY MCA.,M.PHIL.,PH.D[2]

Department of Computer Applications, Sarah Tucker College, Tirunelveli-7.

ABSTRACT :

This paper presents a smart building energy management system for next-day household energy consumption forecasting and occupancy-aware energy recommendations. Accurate forecasting can support the planning of building operations and help identify opportunities to reduce avoidable energy use. The study evaluates seven supervised learning models: Random Forest, XGBoost, Gradient Boosting, Long Short-Term Memory (LSTM), bidirectional LSTM (BiLSTM), CNN-LSTM, and an LSTM+CNN-LSTM-Attention hybrid. The models use historical consumption, calendar, weather, and appliance-activity features. Data preprocessing includes the removal of records with unusable values and the construction of lagged, rolling-window, and cyclical temporal features. The data are partitioned chronologically into training, validation, and test sets to reduce information leakage across time. Model performance is assessed using mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and the coefficient of determination (R^2). Based on the results reported in Table 1, the deployed LSTM achieves an RMSE of 8.017021 kWh (8.02 kWh rounded), the lowest among the evaluated models, and is retained as the project's final forecasting model. The system integrates this forecast with YOLOv8-based person detection to incorporate room occupancy into energy-saving recommendations. SHAP and permutation-importance methods are included to support interpretation of the prediction factors. The proposed application combines forecasting, occupancy detection, and recommendation generation within a unified workflow. Further evaluation using additional household data across different buildings and seasons is needed to assess the system's generalisability.

Keywords: Smart Building Energy Management, Deep Learning, Energy Forecasting, LSTM, Explainable AI, SHAP, YOLOv8, Occupancy Detection, Comparative Analysis, Regression.

1.INTRODUCTION

Energy management in residential and commercial buildings plays an important role in reducing avoidable electricity consumption, where timely identification of expected energy usage is necessary for scheduling appliances, planning cooling loads, and avoiding unnecessary consumption in unoccupied rooms. Historical consumption records alone do not indicate what a household will use tomorrow, as usage changes due to weather, the day of the week, occupant behaviour, and seasonal patterns. Therefore, energy management is not only a monitoring problem but also a prediction problem involving multiple time-dependent household characteristics.

Traditional building energy-monitoring systems mainly display fixed historical attributes such as last month's total usage or a simple running average. Although they summarise past consumption, they do not determine how much energy a household is likely to use next, nor whether the rooms consuming that energy are actually occupied. Manual inspection may be needed to identify unusual usage days, while the absence of any forward-looking estimate reduces the system's practical usefulness for day-to-day energy-saving decisions. These limitations indicate the need for an approach that analyses consumption information beyond simple historical display.

Deep learning provides an alternative by learning temporal patterns from recorded consumption data. The proposed paper uses historical consumption, calendar, weather and appliance-activity information to identify patterns associated with next-day energy usage. Analysing these attributes together over a rolling time window enables the model to capture relationships that may not be identified through a simple historical average alone.

Different machine learning and deep learning algorithms use different learning strategies. Random Forest and Gradient Boosting combine multiple decision trees for stable tabular prediction, XGBoost adds regularised gradient boosting, and LSTM, BiLSTM, CNN-LSTM and a LSTM+CNN-LSTM-Attention hybrid use recurrent and convolutional structures to learn sequential, day-by-day dependencies directly. Comparing these algorithms under the same conditions helps evaluate their suitability for next-day household energy forecasting.

The proposed system connects the selected forecasting model with explainability and live occupancy detection. When a room photograph is captured or uploaded, a YOLOv8 detector counts the people present, and this occupancy signal is combined with the forecasting model's predicted next-day energy value through a rule-based recommendation engine. The main contribution of this paper is integrating comparative deep-learning evaluation with a functional, explainable, occupancy-aware energy-management application. Seven supervised forecasting algorithms are evaluated using MAE, RMSE, MAPE and R^2, with the best-performing model integrated into the application. This combines accurate forecasting with explainable, occupancy-aware recommendations and provides a data-driven mechanism for prioritizing energy-saving action.

2. LITERATURE REVIEW

Deep learning and explainable AI have been applied in building energy management to improve consumption forecasting and occupancy-aware control. Different studies have explored load forecasting, model explainability, and occupancy sensing using machine learning techniques.

Hochreiter and Schmidhuber [1] proposed the Long Short-Term Memory (LSTM) network architecture. Their work demonstrated that a gated recurrent structure could learn long-range temporal dependencies without the vanishing-gradient problems of earlier recurrent networks, which is the reason LSTM remains a standard choice for sequential load-forecasting tasks such as the one addressed in this paper.

Breiman [2] introduced the Random Forest ensemble learning method. The study showed that averaging many decision trees, each trained on a bootstrap sample and a random feature subset, reduces prediction variance and produces a stable baseline for tabular prediction problems, including energy-consumption forecasting.

Chen and Guestrin [3] developed XGBoost, a scalable, regularised gradient-boosted tree system. Their work demonstrated that sequentially correcting the residual error of an ensemble, combined with explicit regularisation of tree complexity, produces strong, fast-to-train tabular models that remain competitive with deep learning approaches on structured data.

Lundberg and Lee [4] proposed SHAP (SHapley Additive exPlanations), grounding feature attribution in cooperative game theory. Their work demonstrated that a model's predictions could be explained consistently, both globally and per-prediction, which this paper applies to make the drivers of predicted energy consumption transparent to the end user.

Jocher et al. [5] released YOLOv8, a further refinement of the YOLO (You Only Look Once) single-stage object-detection family with an anchor-free detection head. Their work demonstrated a strong accuracy/speed trade-off suited to interactive applications, which this paper applies to real-time person counting for occupancy-aware energy recommendations.

Mathumitha et al. [6] reviewed intelligent deep learning techniques for energy consumption forecasting in smart buildings. Their study highlighted the importance of deep learning models in identifying energy usage patterns and predicting future consumption, which supports the use of LSTM for next-day household energy forecasting in this paper.

Runge and Zmeureanu [7] reviewed different deep learning techniques for building energy forecasting. Their study demonstrated how deep learning models can capture complex temporal patterns in historical energy data, supporting the selection of LSTM for sequential energy prediction in the proposed system.

Henkel et al. [8] investigated the application of SHAP in building energy management systems. Their work demonstrated how feature contributions can be explained to improve model transparency, which supports the use of Explainable AI for interpreting energy consumption predictions in this paper.

Forootani et al. [9] developed a feature-engineering-based approach for short-term residential load forecasting. Their study highlighted the importance of data preprocessing and relevant feature selection, which supports the use of lag features, rolling averages, and temporal attributes in the proposed energy forecasting system.

Deb et al. [10] reviewed various time-series forecasting techniques for building energy consumption. Their research highlighted the importance of historical consumption patterns and forecasting methods in estimating future energy requirements, supporting the time-series-based energy prediction approach used in this paper.

These works show that deep learning and explainable AI can be useful in building energy management and occupancy-aware forecasting. However, the present paper focuses on predicting next-day household energy consumption by comparing Random Forest, XGBoost, Gradient Boosting, LSTM, BiLSTM, CNN-LSTM and a LSTM+CNN-LSTM-Attention hybrid, using historical consumption, calendar, weather and appliance-activity attributes, and integrating the selected model with SHAP-based explainability and YOLOv8-based occupancy detection.

3.RESEARCH METHODOLOGY

The proposed system uses supervised machine learning and deep learning methods to predict next-day household energy consumption. The workflow includes dataset preparation, feature engineering, model training, evaluation, comparison, and selection. Seven algorithms—Random Forest, XGBoost, Gradient Boosting, LSTM, BiLSTM, CNN-LSTM, and an LSTM+CNN-LSTM-Attention hybrid—are trained on the prepared energy data and compared using regression metrics. LSTM remains the project's final forecasting model and is integrated with YOLOv8-based occupancy detection and SHAP-based explainability to support energy-saving recommendations.

3.1 Dataset Description

The proposed paper uses a five-year household energy dataset containing consumption, temporal, weather, and appliance-activity information covering 39 houses. The dataset includes both numerical and categorical attributes that can be used to identify patterns related to next-day energy consumption. The selected features include the previous day's kWh, the kWh from seven days earlier, the trailing seven-day rolling average, month, day of week, day of year, weekend flag, temperature, humidity, wind speed, solar irradiance, and appliance-activity indicators. Each feature provides different information about the household that can contribute to consumption prediction.

The target variable is target_next_day_kwh, which represents the household's measured energy consumption on the day immediately following the feature window. Lagged consumption features provide the household's own recent usage history, while calendar and weather features provide seasonal and

environmental context. Appliance-activity indicators represent the household's current equipment usage pattern. Personal and building-identification fields such as House ID that do not carry predictive information about future consumption are not used as prediction features.

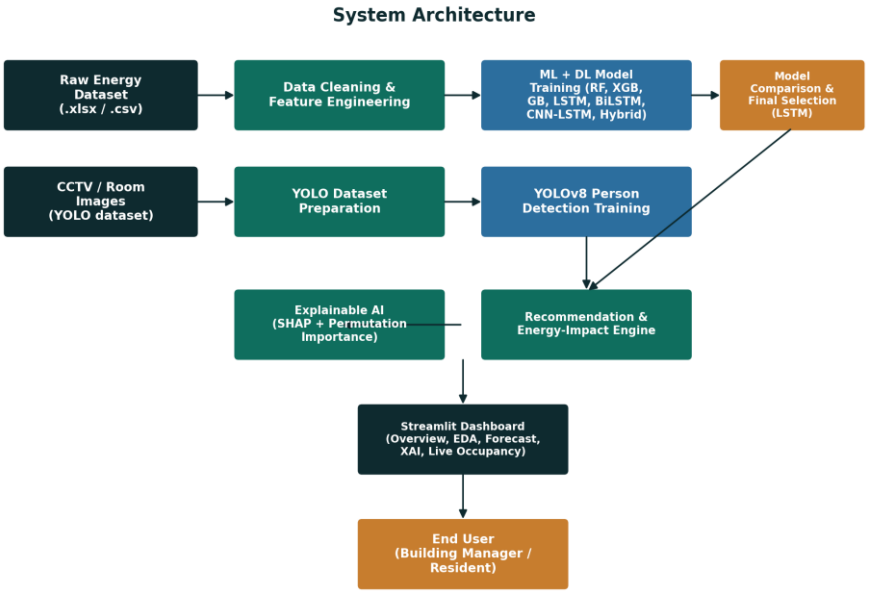


Figure 1. System Architecture of the Proposed Smart Building Energy Management System

3.2 Data Preprocessing

The energy dataset is preprocessed before training the forecasting models. Records with unusable timestamp, house-identifier or kWh values are removed, and duplicate house-timestamp records are dropped. Missing values are filled using a house-wise median for numeric columns, falling back to the global median, and the column mode for categorical columns. Lag features (previous day's kWh, seven-day-earlier kWh, seven-day rolling mean) and cyclical sine/cosine encodings of month, day-of-week and day-of-year are then engineered from the cleaned records.

The input features are separated from the target variable, target_next_day_kwh. The prepared dataset is then divided into training, validation and test data using a chronological 70:15:15 split by unique calendar date, rather than a random row-wise split, so that no date's records appear in more than one subset and the model is never evaluated on a date chronologically earlier than dates it was trained on. This chronological split is essential for time-series data, where a random split would allow information to leak across the training and test sets and produce an optimistically biased accuracy estimate.

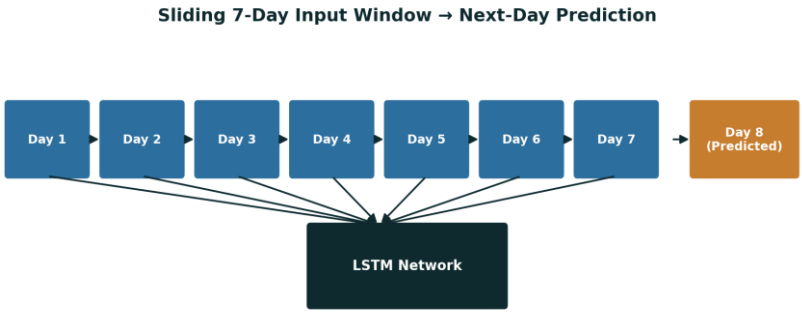


Figure 2. Sliding Seven-Day Input Window Used by the LSTM Forecasting Model

3.3 Forecasting Models

Four of the seven evaluated algorithms are described below: Random Forest, XGBoost, Gradient Boosting, and LSTM. They represent ensemble averaging, regularised boosting, robust boosting, and recurrent deep learning. All seven models—Random Forest, XGBoost, Gradient Boosting, LSTM, BiLSTM, CNN-LSTM, and the LSTM+CNN-LSTM-Attention hybrid—are trained on the same prepared dataset for comparison.

3.3.1 Random Forest Regressor

Random Forest is an ensemble learning method that combines multiple regression trees to produce a final prediction. Each tree is trained on a bootstrap sample of the training data and a random subset of features; the individual trees' predictions are averaged to produce the final output.

$\hat{y} = (1/B)\ \Sigma\ T_b(X),\ \ for\ b = 1\ to\ B$

where B represents the total number of trees and T_1, T_2, …, T_B represent the individual regression trees trained on the engineered feature table described in Section 3.1. Averaging many independently trained trees reduces the variance of the final prediction compared with a single tree, which is why Random Forest is included as a candidate model for next-day energy forecasting.

3.3.2 XGBoost

XGBoost is a regularised gradient-boosted tree algorithm that builds an ensemble of trees sequentially, with each new tree trained to correct the residual error of the trees already in the ensemble.

$F_m(x) = F_{m-1}(x) + \nu \cdot h_m(x)$

where $F_{m-1}(x)$ is the ensemble's prediction before adding the m-th tree, $h_m(x)$ is a regression tree fit to the negative gradient (residual) of the loss function with respect to F_{m-1}, and ν is the learning rate that controls each tree's contribution. XGBoost additionally regularises tree complexity directly in its training objective, which helps control overfitting on the engineered feature table.

3.3.3 Gradient Boosting

Gradient Boosting follows the same additive boosting principle as XGBoost (Section 3.3.2), but in this project is configured with a Huber loss function rather than squared error, so that training is less sensitive to the statistical outliers identified during data cleaning (Section 3.2).

$L_\delta(y,F(x)) = \tfrac{1}{2}(y-F(x))^2\ \ if\ |y-F(x)| \le \delta,\ \ else\ \ \delta|y-F(x)| - \tfrac{1}{2}\delta^2$

where y is the true next-day energy value, F(x) is the model's current prediction, and δ is a threshold controlling the transition between quadratic and linear penalty, so that large-residual (outlier) days contribute roughly linearly rather than quadratically to the training loss.

3.3.4 Long Short-Term Memory (LSTM)

LSTM is a recurrent neural network architecture designed to learn long-range temporal dependencies in sequential data, which is well suited to the day-by-day structure of household energy consumption. At each time step t, the network updates a forget gate, an input gate and an output gate that control what information is kept in, added to, or read from its internal cell state.

$f_t = \sigma(W_f \cdot [h_\{t-1\}, x_t] + b_f);\ \ i_t = \sigma(W_i \cdot [h_\{t-1\}, x_t] + b_i);\ \ C_t = f_t \odot C_\{t-1\} + i_t \odot \tanh(W_C \cdot [h_\{t-1\}, x_t] + b_C)$

where x_t is the feature vector for day t, h_(t−1) is the previous hidden state, C_t is the updated cell state, σ is the sigmoid function, and ⊙ denotes elementwise multiplication. The final hidden state after processing the seven-day input window (Fig. 2) is passed through a dense layer to produce the next-day kWh prediction. LSTM is retained as the project's final forecasting model, consistent with the implementation and the results in Table 1.

3.4 Performance Comparison of Forecasting Models

The seven forecasting models are evaluated using the same held-out test dataset. Their performance is compared using MAE, RMSE, MAPE and R^2. The experimental results are presented in Table 1.

Model	MAE (kWh)	RMSE (kWh)	MAPE (%)	R^2
Random Forest	6.16157	8.320019	86.61315	0.591174
XGBoost	6.338921	8.578975	95.58109	0.56533
Gradient Boosting	5.863321	8.024517	72.34165	0.619699
BiLSTM	5.844368	8.058797	60.34612	0.616443
CNN-LSTM	5.977814	8.23927	63.47515	0.599072
LSTM+CNN-LSTM(Hybrid)	5.851822	8.054205	63.76437	0.61688
LSTM (deployed)	5.825696	8.017021	64.38627	0.620409

Table 1. Performance Comparison of Forecasting Models

Table 1 shows the comparative test-set results for all seven models. The deployed LSTM records an RMSE of 8.017021 kWh, or 8.02 kWh when rounded to two decimal places. This is the lowest RMSE in the table, so LSTM remains the selected forecasting model for the proposed system. BiLSTM has the lowest listed MAPE (60.34612%), illustrating that model performance differs by metric; the selection of LSTM follows the project's deployed-model decision and its lowest RMSE.

3.5 Model Evaluation Metrics

MAE, RMSE, MAPE and R^2 are used to evaluate the seven forecasting models. These metrics measure different aspects of the model's ability to predict next-day energy consumption. Using multiple evaluation metrics provides a more complete comparison than relying on a single metric alone.

3.5.1 Mean Absolute Error (MAE)

$$MAE = (1/n)\ \Sigma^{n}_{i=1}\ |y_i - \hat{y}_i|$$

MAE represents the average absolute difference between the predicted and actual next-day energy values across all n test records, where y_i is the actual consumption and $\hat{y}_i$ is the predicted consumption. A lower MAE indicates that predictions are, on average, closer to the actual measured consumption.

3.5.2 Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{[(1/n)\ \Sigma^{n}_{i=1}\ (y_i - \hat{y}_i)^2]}$$

RMSE squares each error before averaging, so it penalises large individual errors more heavily than MAE. A lower RMSE indicates fewer large mispredictions, which is particularly relevant for energy forecasting since a single large miss can produce a misleading recommendation.

3.5.3 Mean Absolute Percentage Error (MAPE)

$$MAPE = (100/n)\ \Sigma^{n}_{i=1}\ |(y_i - \hat{y}_i)\ /\ y_i|$$

MAPE expresses the average error as a percentage of the actual value, which makes it easier to interpret across households with different baseline consumption levels. A lower MAPE indicates a smaller relative forecasting error.

3.5.4 Coefficient of Determination (R^2)

$$R^2 = 1 - [\Sigma^{n}_{i=1}(y_i - \hat{y}_i)^2\ /\ \Sigma^{n}_{i=1}(y_i - \grave{y})^2]$$

R^2 measures the proportion of variance in actual next-day energy consumption that is explained by the model, where $\grave{y}$ is the mean of the actual test-set values. A higher R^2, closer to 1, indicates that the model explains a larger share of the variation in household energy consumption. Therefore, R^2 is considered along with MAE, RMSE and MAPE when comparing the seven models.

4. RESULTS AND DISCUSSION

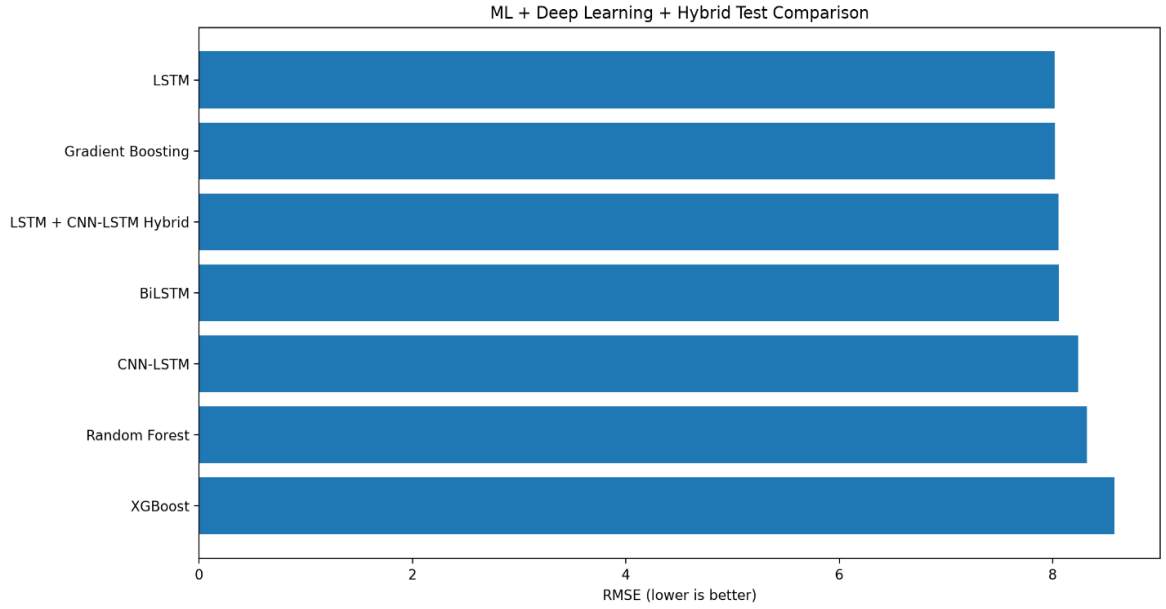


Figure 3. Comparative RMSE of Forecasting Models

The seven forecasting models were evaluated using MAE, RMSE, MAPE, and R^2 on the same chronologically split test data. Table 1 presents the numerical results, and Fig. 3 compares RMSE across the models.

The deployed LSTM achieves a test RMSE of 8.017021 kWh, reported as 8.02 kWh when rounded to two decimal places. This is the lowest RMSE among the seven models listed in Table 1.

Permutation-importance analysis for the tabular model identifies Kwh_rolling_7 (importance mean 6.285827108), Kwh_lag_1 (0.04165818), and day_of_week (0.027809426) as the three highest-ranked features. This suggests that recent consumption history and calendar information contribute to the model's predictions. Other features, including building-size ratios and appliance-activity aggregates, have lower reported importance.

The developed system uses Streamlit as the application layer and is deployed on Streamlit Community Cloud, connected directly to a GitHub source

repository. During the live occupancy workflow, the system captures or accepts an uploaded room photograph, runs YOLOv8 detection to count the people present, and combines the detected occupancy band with the LSTM's predicted next-day energy value to prioritize and justify the displayed energy-saving recommendation.

LSTM was retained as the project's final forecasting model and integrated with the YOLOv8-based occupancy detector and the explainability layer. The system forecasts next-day household energy consumption, detects room occupancy from an image, and uses these outputs to prioritize energy-saving recommendations.

Together, the forecasting and occupancy workflow demonstrates how the application can present next-day energy estimates alongside prioritized recommendations.

CONCLUSION

This study evaluated seven machine learning and deep learning algorithms—Random Forest, XGBoost, Gradient Boosting, LSTM, BiLSTM, CNN-LSTM, and an LSTM+CNN-LSTM-Attention hybrid—for next-day household energy forecasting. Their performance was compared using MAE, RMSE, MAPE, and R^2.

The deployed LSTM records a test RMSE of 8.017021 kWh (8.02 kWh rounded), the lowest RMSE reported in Table 1. LSTM is therefore retained as the project's final forecasting model.

The reported permutation-importance results rank Kwh_rolling_7, Kwh_lag_1, and day_of_week as the three most influential features. This highlights the contribution of recent consumption history and calendar information to the predictions. The results should be interpreted in light of the feature-importance method and model to which they apply.

The selected model was integrated with the occupancy detection and recommendation application. The system uses a captured or uploaded room photograph together with the forecasted next-day energy value to identify and prioritize energy-saving opportunities. The predicted occupancy band is used to scale the estimated potential saving shown to the user.

Future enhancement: evaluate the deployed LSTM and the complete occupancy-aware recommendation workflow on additional, independently documented household data over multiple years, and report the results across households and seasons.

REFERENCES

[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

[2] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001. DOI: 10.1023/A:1010933404324.

[3] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, 2016, pp. 785–794. DOI: 10.1145/2939672.2939785.

[4] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 4765–4774.

[5] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics.

[6] R. Mathumitha, P. Rathika, and K. Manimala, "Intelligent deep learning techniques for energy consumption forecasting in smart buildings: a review," Artificial Intelligence Review, vol. 57, article 35, 2024. DOI: 10.1007/s10462-023-10660-8.

[7] J. Runge and R. Zmeureanu, "A Review of Deep Learning Techniques for Forecasting Energy Use in Buildings," Energies, vol. 14, no. 3, 608, 2021. DOI: 10.3390/en14030608.

[8] P. Henkel, T. Kasperski, P. Stoffel, and D. Müller, "Interpretable data-driven model predictive control of building energy systems using SHAP," Proceedings of the 6th Annual Learning for Dynamics & Control Conference, PMLR 242, pp. 222–234, 2024.

[9] A. Forootani, M. Rastegar, and A. Sami, "Short-term individual residential load forecasting using an enhanced machine learning-based approach based on a feature engineering framework: A comparative study with deep learning methods," Electric Power Systems Research, vol. 210, article 108119, 2022. DOI: 10.1016/j.epsr.2022.108119.

[10] C. Deb, F. Zhang, J. Yang, S. E. Lee, and K. W. Shah, "A review on time series forecasting techniques for building energy consumption," Renewable and Sustainable Energy Reviews, 2017. DOI: 10.1016/j.rser.2017.02.085