J. Electrical Systems 22-01 (2026): 74-87

[1]Rawan Sanyour
[2]Sara Alhindi
[3]Sara Abdulrahman
[4]Samaher Zolaly
[5]Mayar Albouq
[6]Fai Alturki

Context-Aware Smart-Home Energy Forecasting Using an Energy-Derived Presence Proxy: A Hybrid Deep Learning and Machine Learning Approach

Journal of Electrical Systems

Abstract: - The demand for household electricity continues to rise, increasing both energy costs and greenhouse-gas emissions. An accurate forecast of short-horizon consumption will contribute significantly to sustain such efforts through facilitating demand response, load shifting, and general smart home energy management. This paper presents a hybrid model that integrates (i) appliance-level usage and weather features with (ii) an energy-derived occupancy proxy (presence rate), aimed at improved next-step household electricity consumption prediction. The proposed framework first estimates a presence-proxy signal from the appliance activity patterns and then utilizes this signal as an added contextual feature of consumption forecasting. We evaluate tree-based ensemble learners (Random Forest, XGBoost, LightGBM) and deep learning models (CNN and LSTM) using a time-aware train/test split. Results show that incorporating the presence-proxy enhances forecasting accuracy for the best-performing tree-based model, while LSTM maintains robustness due to its temporal modeling capacity. Beyond predictive accuracy, we discuss how the improved forecasts can be operationalized for peak reduction and energy-efficiency interventions, and we highlight validity considerations when inferring occupancy-related signals from smart meter data.

Keywords: Energy forecasting; Smart homes; Demand response; Sustainable energy management; radient boosting; LSTM; Occupancy proxy; Data-driven sustainability.

1. INTRODUCTION

Global energy demand and electricity consumption continue to increase, thereby reinforcing the urgency of energy efficiency and demand-side management. Recent reports indicate an approximately 2% year-on-year increase in global energy demand accompanied by increasing electricity demand due to more electrification, digitalization, and extreme weather (IEA, Global Energy Review; Electricity Market Report [1]). High-frequency usage patterns and weather variability in the residential sector create strong short-term fluctuations that make forecasting difficult and thus reduce the effectiveness of manual energy-saving behaviors.

Short-horizon energy prediction is not just a problem of machine learning. It has also emerged as an enabler of sustainability: reliable forecasts can inform home energy management systems, support time-of-use pricing strategies, facilitate demand response and help households to shift discretionary loads away from the peak. However, behavioral context such as occupancy or presence related activity that can be the primary driver of appliance usage is neglected in most data-driven household forecasting studies that use only historical consumption and weather features.

This paper proposes a hybrid framework that augments conventional forecasting inputs by introducing an energy-derived presence proxy, constructed from appliance activity patterns. The resulting model aims to provide (1) improved prediction accuracy and (2) a practical pathway for translating prediction gains into sustainability-relevant actions (e.g., peak shaving and reduced energy waste). Following is the list of paper contributions:

- A hybrid forecasting framework that integrates appliance-level consumption, weather signals, and an energy-derived presence proxy to improve household electricity prediction.
- A transparent definition of the presence proxy (presence rate) and an evaluation protocol that mitigates leakage by computing test-time features without access to future information.
- A comparative study of ensemble ML and deep learning models under the same preprocessing and evaluation pipeline, including an ablation analysis (with vs. without presence proxy).

2. RELATED WORKS

Tostado-Velez et al. [2] addressed key challenges in household energy management by developing new demand response strategies. These strategies aimed to improve energy efficiency while reducing the financial burden on consumers and addressed the gaps in balancing grid performance with user affordability. The study introduced three decision-based demand response strategies: peak clipping, load allocation, and demand leveling. These

1 *,2,3,4,5,6 *Information System Department, Faculty of Computer Science and Engineering, Taibah University, Al Madinah Al Munawwara, Saudi Arabia, Email: {*rsanyour, tu4252230, tu4152969, tu4257396, tu4258416, tu4255710}@taibahu.edu.sa*

strategies were further analyzed and compared to conventional Price-Based Demand Response (PBDR) approaches, showing superior performance with up to 70% improvement in metrics such as Peak-to-Average Ratio (PAR) and peak reduction. This was followed by Bhol et al. [3] who focused on reactive power demand forecasting based on real power consumption. The study highlighted the need for more robust predictive models and used the Holt-Winters exponential model optimized with the Global Flower Pollination Algorithm (GFPA) to predict reactive power, outperforming conventional techniques such as Autoregressive integrated moving average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA). These techniques showed superior performance in short-term reactive power demand forecasting. In a different context, Almoallem [4] studied electricity consumption across different residential buildings, examining the impact of factors such as population and thermal insulation systems. This study relied on statistical data analysis to examine the relationship between different factors such as building type, age, the number of air conditioning systems, and electricity consumption. Fahmy et al. [5] employed two predictive models, the Polynomial Model and the ARIMA model, which were integrated into a composite model to enhance the accuracy of electricity consumption predictions in Saudi Arabia. This integration resulted in a significant improvement in prediction accuracy by 33.5%, while reducing the Root Mean Square Error (RMSE) by 18.5% compared to using the Polynomial model alone. The research highlights the effectiveness of combining statistical and time series models for improving forecast performance in large-scale energy management systems.

Sundaram et al. [6] focused on the development of a model for predicting the annual energy consumption of residential buildings using the LSTM model. This study compared the Long Short-Term Memory LSTM model with other models such as Deep Neural Networks (DNN) and Artificial Neural Networks (ANN), showing that the LSTM model achieved an accuracy of 97%, compared to 95% for DNN and 92% for ANN. Elhabyb et al. [7] utilized Random Forest, LSTM, and Gradient Boosting Regressor (GBR) to predict energy usage in three teaching buildings by incorporating various data attributes; aimed to address gaps in previous data-centric approaches using actual historical data.

Utilizing LSTM networks and an enhanced sine-cosine optimization model, Somu et al. [8] created eDemand, a new model for predicting building energy consumption, which outperformed previous state-of-the-art models in predicting energy load in real time. Suranata et al. [9] predict the energy consumption of the kitchen. They used LSTM and feature engineering techniques. Two tables were subjected to the LSTM model and Principal component analysis (PCA) was used to extract significant features. Faiq et al. [10]created an energy forecasting system based on LSTM for business buildings, achieving the best RMSE values with data from the Malaysian Meteorological Service.

Ndife et al. [11] introduced a sophisticated model to predict power consumption trends using Convolutional Long Short-Term Memory (ConvLSTM) encoder-decoder algorithms, showing higher accuracy and computational efficiency compared to traditional methods. Ferdoush et al. [12]developed a hybrid prediction model that combined bidirectional long-distance storage techniques with random forests, which outperformed standard models when tested with electricity consumption data from Bangladesh.

Shapi et al. [13] build an energy demand prediction model. The study employed k-nearest neighbors, and vector machines to prove that the model could efficiently predict daily energy usage based on meteorological data, showing a specific distribution pattern for each tenant's energy consumption. This was followed by Ajayi et al. [14] who used various machine learning methods to predict annual energy consumption in residential buildings, with the Deep Neural Network (DNN) proving to be the most effective model. He and Tsang.[15] developed a hybrid model combining LSTM with Improved Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (iCEEMDAN) to individually forecast energy consumption patterns, employing a closed-loop recurrent unit neural network and a temporal attention layer for enhanced prediction accuracy.

Musleh et al. [16] introduced the Kstar (K*) model, an instance-based learning algorithm that effectively predicted monthly electricity consumption with superior metrics (Correlation Coefficient. (CC): 0.9373, Mean Absolution Percentage Error (MAPE): 0.1569, RMSE: 0.0488) compared to traditional ANN models; it demonstrated improved performance when used in bagging ensemble technique. Xie et al. [17]and Javaid et al. [18] addressed energy usage in smart homes, utilizing the Factorial Hidden Markov Model (FHMM) algorithm in Non-Intrusive Load Monitoring (NILM) to break down aggregated electrical data into consumption statistics for individual devices, helping users manage energy usage more effectively. Zhang et al. [19] proposed the RCNN-SVR model, which combined convolutional layers of a neural network with SVR to predict power consumption more effectively than using SVR or RCNN alone. The RCNN-SVR model solved key issues like low prediction accuracy, high computing expense, and inadequate training data, demonstrating superior results compared to traditional models.

Wu's [20] introduced customer segmentation based on electricity consumption characteristics utilizing k-means algorithm and its optimized versions, such as Particle Swarm Optimization k-means (PSO-k-means) and Grey Wolf Optimization k-means (GWO-k-means) This segmentation classified customers into four groups, providing valuable insights into how different types of consumers use energy. Prasetiyo et al. [21]employed the Naïve Bayesian Classifier to classify energy efficiency datasets, such as heating load and cooling load, achieving an accuracy value of 82.81%. Raza et al. [22] focused on optimizing load forecasting and scheduling techniques in Home Energy Management Systems (HEMS), introducing methods such as ARIMA, Genetic Algorithm (GA), and Support Vector Machines (SVM), while also discussing the future technical advancements in HEMS such as blockchain, federated learning, and AI.

Rashid et al. [23] emphasized the implementation of a Cognitive IoT (CIoT) model, coupled with machine learning for monitoring energy usage in smart homes. This system used Raspberry Pi as a gateway to collect data from appliances and trained the dataset on LSTM models, achieving high prediction accuracy. Rambabu et al. [24] analyzed household energy consumption with weather and surroundings using machine learning algorithms such as Linear Regression, Random Forest, and XG Boost, showing that tree-based models provided the best results. Bonetto et al. [25] explored machine learning techniques for energy prediction, comparing ARMA, SVM, Nonlinear Autoregressive (NAR), and LSTM models, and recommended using NAR for short-term predictions and SVM for long-term predictions.

It is worth noting that these studies have neglected the user’s occupancy when predicting energy consumption. This feature can be utilized to translate prediction gains into sustainability-relevant actions through reducing peak demand and overall energy waste through home energy management systems. For example, forecast-informed scheduling can shift discretionary loads (e.g., dishwasher/washer/dryer) to off-peak periods, reducing peak-to-average ratio and enabling higher penetration of renewables on the grid.

3. Developmental Methodology

Fig. 1 depicts the layered architectural design and explains how its components interact to achieve the functions of the proposed tool. It consists of three different layers: data collection, data preprocessing and cleaning, and prediction model development. Each layer consists of several phases that will be illustrated in the subsequent sections.

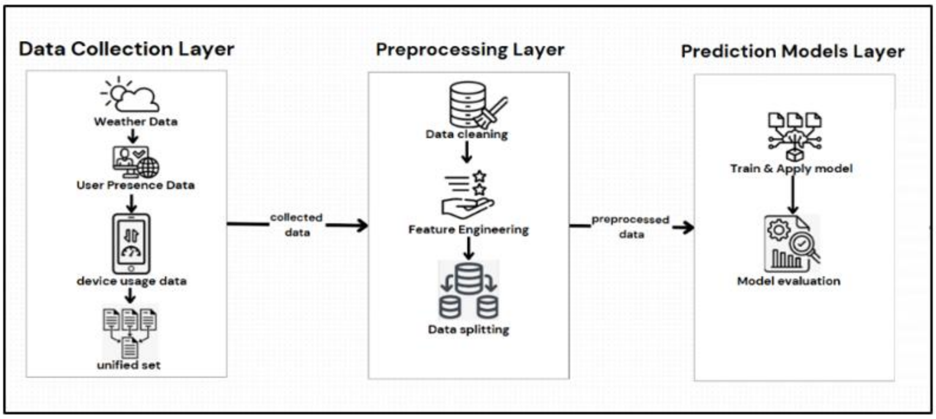

Figure 1: IoT-based Smart Home Energy Management System - Layered Architectural Design.

3.1. Data collection layer

The Smart Home Dataset with weather Information is publicly available at [26]. It contains one-minute readings of household appliances in kilowatts from a smart meter and the weather conditions for a specific area. Organized by Category, Table 1 details the different features within the dataset. Each entry in the table includes the variable's Name and a corresponding Description.

Table 1: Summary of Dataset Utilized

Category	Name	Description
Time	time	Time of the readings, with a time span of 1 minute
Energy Usage	use [kW]	Total energy consumption
	gen [kW]	Total energy generated by solar or other power generation resources
	House overall [kW]	Overall house energy consumption
	Dishwasher [kW]	Energy consumed by a specific appliance (Dishwasher)
	Furnace 1 [kW]	Energy consumed by a specific appliance (Furnace 1)
	Furnace 2 [kW]	Energy consumed by a specific appliance (Furnace 2)
	Home office [kW]	Energy consumed by a specific appliance (Home office)
	Fridge [kW]	Energy consumed by a specific appliance (Fridge)
	Wine cellar [kW]	Energy consumed by a specific appliance (Wine cellar)
	Garage door [kW]	Energy consumed by a specific appliance (Garage door)
	Kitchen 12 [kW]	Energy consumption in kitchen 1
	Kitchen 14 [kW]	Energy consumption in kitchen 2
	Kitchen 38 [kW]	Energy consumption in kitchen 3
	Barn [kW]	Energy consumed by a specific appliance (Barn)
	Well [kW]	Energy consumed by a specific appliance (Well)
	Microwave [kW]	Energy consumed by a specific appliance (Microwave)
	Living room [kW]	Energy consumption in Living room
	Solar [kW]	Solar power generation
Weather	temperature	Temperature is a physical quantity expressing hot and cold.
	humidity	Humidity is the concentration of water present in air.
	visibility	Visibility sensors measure the meteorological optical range which is defined as the length of atmosphere over which light travels.
	apparentTemperature	Apparent temperature is the equivalent temperature perceived by humans, caused by the combined effects of air temperature, relative humidity and wind speed.
	pressure	Falling air pressure indicates that bad weather is coming, while rising air pressure indicates good weather.
	windSpeed	Wind speed, caused by air moving from high to low pressure, usually due to changes in temperature.
	cloudCover	Cloud cover refers to the fraction of the sky obscured by clouds.
	windBearing	In meteorology, an azimuth of 000° is used when no wind is blowing; 360° means the wind is from the North.
	dewPoint	The atmospheric temperature below which water droplets begin to condense.
	precipProbability	A probability of precipitation is a measure of the probability that at least some minimum quantity of precipitation will occur.
	precipIntensity	The intensity of rainfall measures the amount of rain that falls over time.
	windSpeed	Wind speed, caused by air moving from high to low pressure, usually due to changes in temperature.
	cloudCover	Cloud cover refers to the fraction of the sky obscured by clouds.
	windBearing	In meteorology, an azimuth of 000° is used when no wind is blowing; 360° means the wind is from the North.
	precipProbability	A probability of precipitation is a measure of the probability that at least some minimum quantity of precipitation will occur.

3.2. Preprocessing Layer

Data preprocessing and cleaning are essential steps to ensure the quality and consistency of the dataset. This step is crucial as machine learning models are highly sensitive to noise, missing values, and improper formatting.

3.2.1. Handling missing values

The dataset was examined for missing values. Most columns had one missing value. Therefore, these rows have been deleted from the dataset.

3.2.2. Redundant element removal

It has been observed that two columns have identical values across all entries as shown in Fig. 2. To streamline the data and avoid redundancy which can negatively impact the performance of certain machine learning algorithms, one of the duplicate columns was removed with ensuring no loss of data integrity or predictive value.

3.2.3. Normalization and Scaling

Min-Max scaling is utilized to normalize all continuous numeric variables, standardizing the input space and ensuring that all features contributed uniformly during training. This step is crucial due to the significant differences in units and scales among features like temperature, humidity, and power usage. This step guaranteed that the input data is normalized, thus, appropriately tailored for the developed hybrid ML model.

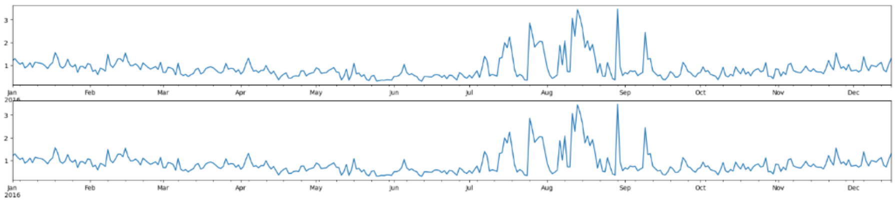

Figure 2: Dataset After Removing Duplicate Column

3.2.4. *Feature engineering*

This stage is essential for converting raw data into insights by utilizing significant visual representations and statistical investigation. Fig.3 depicts the correlation between each device's energy consumption and the total energy usage of the household.

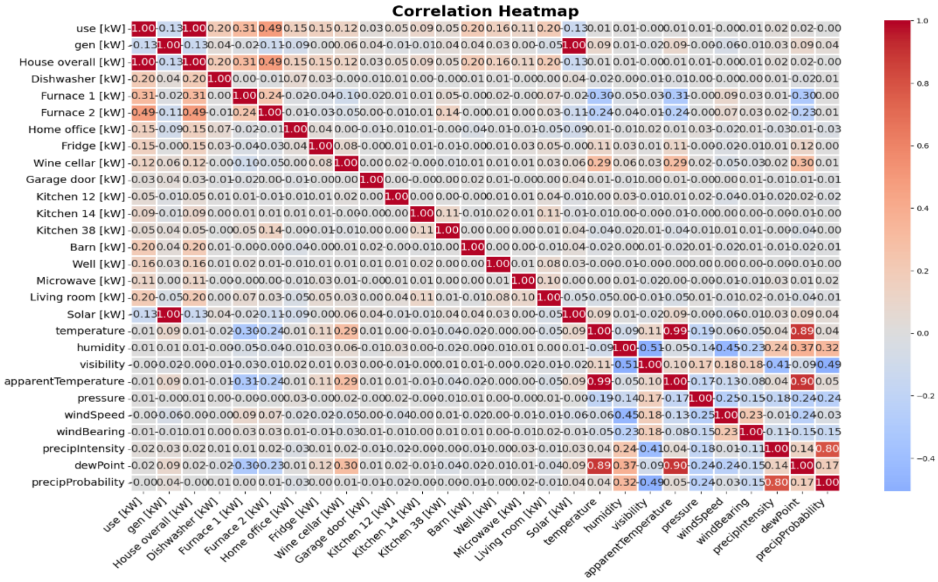

Figure 3: The Correlation Between Device's Energy Consumption and The Total Energy Usage

To capture behavioral context that is associated with household electricity demand, we engineer an energy-derived presence proxy from appliance-level consumption signals. Unlike sensor-based occupancy, this proxy is computed solely from energy measurements and is therefore considered as a contextual indicator of likely household activity rather than a ground-truth occupancy label.

Let E_{it} denote the electricity consumption of appliance *i* at time t and Total Energy Consumptiont denote the total household consumption at time *t*. Appliance importance weights (*wi*) are computed on the training period using the absolute correlation between each appliance series and *Total Energy Consumption*. The weights are then normalized to sum to 1. Using these weights, a time-indexed Presence Rate (PR_t) is computed via Eq. (1) as a weighted fraction of appliance contributions to Total Energy Consumption. The proxy label pr_t is obtained by thresholding PR_t using a threshold τ estimated on the training period (τ = median(PR_t)).

This engineered proxy is used to test whether behavioral context provides incremental value for short-horizon load forecasting and for sustainability-oriented applications such as demand response, peak reduction, and personalized energy-saving recommendations. Accordingly, energy consumption is associated with an energy-derived Presence Rate (PR_t) computed from appliance-level energy participation as follows:

$$\text{Presence Rate}(PR_\text{t}) = \frac{W_1{*}E_{1t}+W_2{*}E_{2t}+W_3{*}E_{3t}+\cdots+W_n{*}E_{nt}}{Total\ Energy\ Consumption_t} \qquad (1)$$

In Eq. (1), E_{it} represents the energy consumed by appliance *i* at time *t*, w_i represents the normalized importance weight for appliance *i* (estimated from training data), and *Total Energy Consumption*$_t$ represents the total household energy consumption at time *t*. To avoid information leakage, both w_i and τ are computed using the training period only and then held fixed when generating PR_t and pr_t for validation and test periods.

If $PR_t > \tau$: the household is classified as "active" (presence proxy $pr_t = 1$).

If $PR_t \leq \tau$: the household is classified as "inactive" (presence proxy $pr_t = 0$).

It is worth noting that, in order to prevent information leakage, all train/validation/test splits are chronological. In addition, feature scaling parameters (e.g., Min-Max normalization), appliance weights w_i, and the PR_t threshold (τ) are fitted on the training set only. These learned parameters (thresholds/weights) are then applied to compute PR_t in the test set without re-estimating parameters from test data.

3.3. Prediction model layer

At this stage, a number of algorithms are selected to build a hybrid framework for both predicting household energy consumption and quantifying the contribution of contextual behavioral signals. The framework consists of two phases: (i) estimating an energy-derived presence proxy (*PR*) from appliance-level energy participation, and (ii) forecasting household energy consumption using the original dataset features augmented with *PR* as an additional contextual feature. As shown in Fig 4, the first model generates output *PR* (active/inactive proxy). This output is appended to the original feature set and passed on to the second model which then provides a final forecast on consumption levels. This design allows for a comparison with and without *PR* to quantify its incremental predictive value to the overall prediction accuracy.

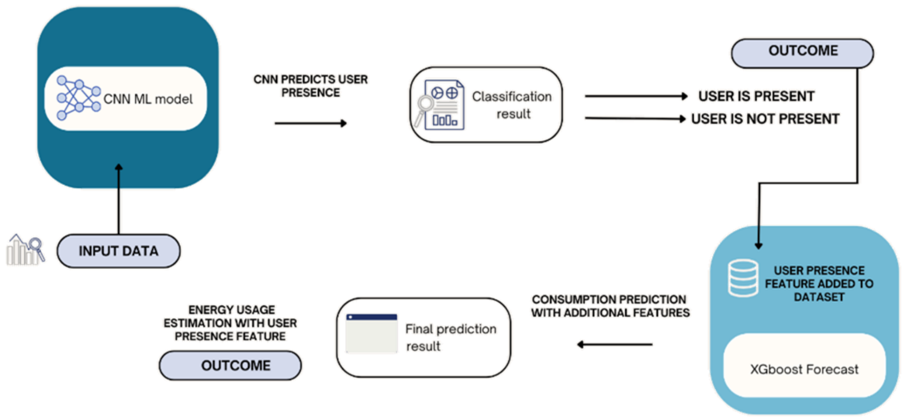

Figure 4: The Hybrid Framework for Energy Consumption Prediction.

3.3.1. Presence proxy estimation

A. Convolutional Neural Network (CNN)

The CNN algorithm is selected due to its capabilities in exhibiting complex temporal patterns in datasets. CNNs are particularly effective in handling large and intricate datasets, enabling the extraction of hidden features that help to discover the user presence based on device activity. The mathematical background of this model is presented through the following equations [27]. The convolution operation: responsible for feature extraction.

$$G[m,n] = (f * h)[m,n] = \sum_j \sum_k h[j,k] f[m-j, n-k] \quad (2)$$

- f = the input
- h= our kernel.
- N and M= The indexes of rows and columns of the result matrix.

The Rectified Linear Unit ReLU activation function introduces non-linearity.

$$f(x) = max(0, x) \quad (3)$$

- x is the input to the neuron.
- The function returns x if x is greater than 0.
- If x is less than or equal to 0, the function returns 0.

The Binary Cross-Entropy loss function measures the performance of the classification model.

$$\text{BCE} = -\frac{1}{N}\sum_{i=1}^{N} [y_i \log(p_i) + (1-y_i)\log(1-p_i)] \quad (4)$$

- N is the number of observations.

- y_i is the actual binary label (0 or 1) of the i^{th} observation.
- p_i is the predicted probability of the i^{th} observation being in class 1.

The purpose of the Binary Cross-Entropy loss function is to compare the predictions made by the model with the actual labels. If the predictions are close to the actual labels (i.e., the predicted probability is close to the real label), the resulting value from the function will be small, indicating that the model is performing well. However, if the predictions are far from the actual labels, the loss value will be higher, indicating that the model needs improvement [28].

B. Artificial Neural Network (ANN)

ANN is selected due to its strong capability in modeling complex, non-linear relationships between input features and target outputs. The primary objective is to detect the user presence within a residential setting based on electrical energy consumption data collected from various household appliances. This task was framed as a binary classification problem, wherein the goal was to determine the presence (labelled as 1) or absence (labelled as 0) of a user, using a derived indicator termed the Presence Rate (PR). The network architecture comprised an input layer responsible for receiving normalized input features (such as appliance-level energy usage), followed by one or more hidden layers employing (ReLU) activation function. The output layer utilized a Sigmoid activation function, defined in equation 5 to produce a probabilistic prediction in the range [0,1] [29]. The ANN model was trained using the Binary Cross-Entropy loss function demonstrated in Eq. 5.

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{5}$$

- x is the input value,
- e is Euler's number (≈2.718)

C. Logistic Regression

Logistic Regression is employed as a baseline model for binary classification, aiming to predict user presence (1) or absence (0) using appliance usage patterns and environmental indicators. Logistic Regression is selected due to its interpretability, computational efficiency, and proven efficacy in binary classification problems. The algorithm uses the logistic (sigmoid) function to map the weighted sum of input features to a probability between 0 and 1[30]. The predictive equation is given by Eq. 6.

$$y = \frac{e^{(b_0+b_1x)}}{1+e^{(b_0+b_1x)}} \tag{6}$$

- x = input value
- y = predicted output
- b0 = bias or intercepting term
- b1 = coefficient for input (x)

D. Decision Tree

Decision tree is a classification model that uses a tree structure to make decisions based on simple conditions applied to data features. Each node represents a question, and each branch represents the outcome of that question. It is simple, easy, and helpful in analyzing numerical data. It starts with the root and gradually splits the data. The final nodes represent predictions. Eq. 7 calculates Gini Impurity, and the parameters values are: *random_state* = 42,criterion = *'gini',max_depth* = 5.

$$Gini = 1 - i = \sum_{i=1}^{c} p_i^2 \tag{7}$$

- C represents the number of classes in the dataset.
- pi is the probability of a sample being classified into class i.

The Gini Impurity measures the likelihood of an element being incorrectly classified in a dataset. It plays an essential role in decision trees in determining the best attribute to split data on at each node[31].

E. Support Vector Machine (SVM)

Support Vector Machine (SVM) algorithm is utilized for binary classification to identify user presence based on appliance-level energy consumption. It is selected for its effectiveness in handling high-dimensional data and its capacity to create optimal decision boundaries in linearly separable classification tasks. Prior to model training, the feature space was normalized to transform the input features to the [0, 1] range ensuring uniformity in feature scale. A linear kernel was selected, with the regularization parameter C set to 1. The random state was fixed at 42 to maintain consistency across experimental runs.

3.3.2. Energy consumption prediction

The second phase includes the development of the predictive models for energy consumption using the following algorithms.

A. XGBoost

XGBoost, short for Xtreme Gradient Boosting, is a scalable ensemble learning algorithm based on gradient boosting decision trees. It is widely utilized for its effectiveness in structured data prediction tasks. XGBoost sequentially builds an ensemble of weak learners (i.e., decision trees), where each new tree attempts to correct its predecessor's prediction errors. The algorithm offers built-in handling of missing data, efficient memory usage, and robust performance in classification tasks, making it an appropriate choice for energy consumption prediction scenarios [32]. The objective function was set to *reg:squarederror* to indicate a regression task, while *n_estimators*=100 specified the construction of 100 trees. A fixed *random_state*=42 ensured result reproducibility.

B.Random Forest

It is a set of randomly generated decision trees whose results are combined to obtain a more accurate and stable prediction. It uses the principle of group voting to arrive at the final result, helping with complex data analysis and consumption prediction. The algorithm creates many decision trees; each trained on a random sample of data. In each tree, a random subset of features is selected to determine each split. For classification, the score of each tree is calculated, and the final decision is made based on the ensemble vote (for classification) or the mean (for regression) [33]as shown in Eq. 8.

$$y\hat{} = mode(h1(x), h2(x), \ldots, hn(x)) \quad (8)$$

- hat{y}: This is the final predicted output from the Random Forest model.
- hn(x): These are the predictions from each individual decision tree in the forest (there are n trees).
- {mode}(…): This function takes the most frequent (majority) value among the predictions.

C. LightGBM

LightGBM is a powerful gradient boosting framework based on decision trees, designed for enhanced speed and memory efficiency. Similar to other boosting algorithms, it is applicable for both classification and regression tasks. LightGBM operates by sequentially training decision trees, with each new tree aiming to minimize the residual errors of its predecessor. In contrast to conventional techniques, it employs a leaf-wise tree development approach with histogram-based feature binning, resulting in greater speed and precision on extensive datasets. The LightGBM is selected due to its effectiveness with large-scale tabular datasets where rapid iteration and robust predictive ability matter [34] The prediction is computed using Eq. 9.

$$\hat{y}_i = \sum_{k=1}^{R} f_k(x_i) \quad (9)$$

- Where f_k is the prediction of the k-th decision tree.

D. Convolutional Neural Network (CNN)

CNNs, though traditionally used for image data, are adopted for one-dimensional time-series related to energy consumption. The input data was reshaped using np.expand_dims(). The architecture featured two convolutional blocks: the first Conv1D layer had 128 filters and a kernel size of 3, followed by batch normalization and a dropout layer (rate = 0.2). The second block included a Conv1D layer with 64 filters, also followed by normalization and dropout. The output was flattened and passed through two dense layers (128 and 64 units with ReLU activation), followed by a single-unit output layer for regression. CNN is trained using the Adam optimizer [35] and MSE as the loss function. This architecture was effective in extracting features and learning patterns from sequential energy data.

E. Recurrent Neural Network (RNN)

RNNs are a category of neural networks tailored for managing sequential and time-series data, which makes them ideal for forecasting energy consumption patterns using past data. In contrast to conventional neural networks, RNNs preserve a memory of past inputs via a structure known as a hidden state, enabling them to recognize patterns over time. This is particularly helpful for simulating how energy consumption varies over the course of a day or week. The RNN computes a hidden state at every time step by utilizing the present input and the preceding hidden state, subsequently generating an output prediction [36]. In mathematical terms, this procedure is represented as Eq. 10.

$$h_t = \tanh\,(W_h x_t + U_h h_{t-1} + b_h), \hat{y}_t = W_y h_t + b_y \tag{10}$$

- Where h_t is the hidden state and $\hat{y}_t$is the predicted output.

RNN is used to predict future energy consumption utilizing previous readings as input. We used a chronological split with 80% of the earliest records for training and the remaining 20% (later records) for testing. Min-Max scaling parameters were fitted on the training set only and then applied to the test set. The model was trained with the Adam optimizer and mean squared error (MSE) loss, with early stopping based on a validation subset taken from the training set to reduce overfitting and training time.

F.Long Short- Term Memory (LSTM)

LSTM networks, a specialized type of recurrent neural network (RNN), are selected due to their strong ability to model and retain long-term dependencies in sequential data [37]. We formulate the problem as one-step-ahead forecasting using sliding-window sequences: a window of 10-time steps (10 minutes of past readings) is used to predict the next minute's consumption. We used a chronological split with 80% of the earliest records for training and the remaining 20% (later records) for testing. All preprocessing steps (including Min-Max scaling and any engineered context features such as the presence proxy) were fitted on the training set only and applied unchanged to the test set. The architecture consists of two stacked LSTM layers (50 units each, with return sequences enabled on the first), followed by a dense output layer. The model was trained with the Adam optimizer and MSE loss for up to 100 epochs with batch size 200, using early stopping on a validation subset from the training period. Fig. 5 illustrates the training and prediction processes.

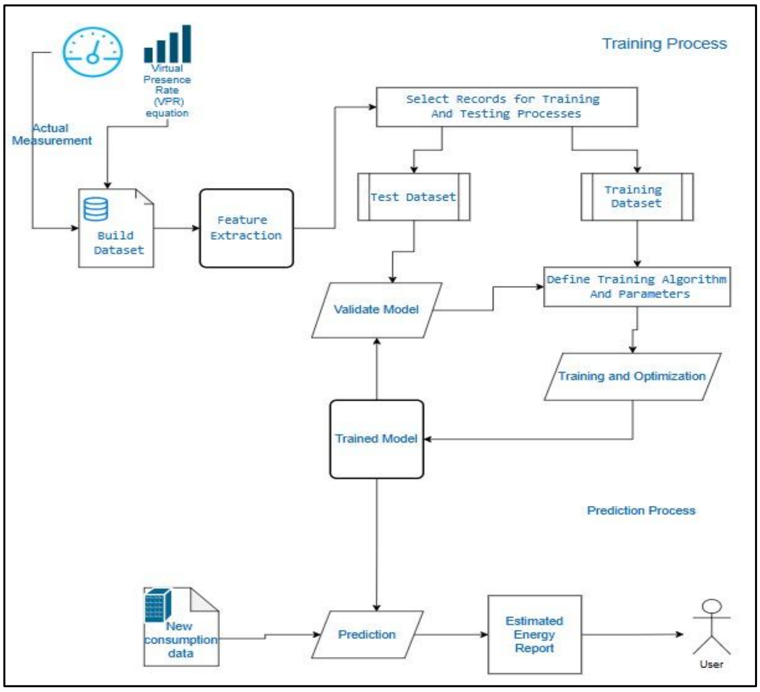

Figure 5: IoT-based Smart Home Energy Management System - Training and Prediction Processes

3.3.3. Model Evaluation and Validation

Multiple evaluation metrics are employed to assess the performance of the developed models. This multi-metric approach ensures that different aspects of model accuracy and error behavior are captured effectively.

- Root Mean Square Error (RMSE): This metric penalizes larger errors more heavily than smaller ones and is especially useful when large deviations are undesirable. It is computed using Eq. 11.

$$RMSE = \sqrt{\frac{1}{n}\sum_{n}^{I}(y^i - \tilde{y}^i)^2} \quad (11)$$

- Mean Absolute Error (MAE): This metric provides the average absolute difference between predicted and actual values and is often used due to its simplicity and direct interpretability. It is computed using Eq. 12.

$$MAE = \frac{1}{n}\sum|y, -\bar{y}| \quad (12)$$

Using these two metrics, RMSE and MAE, allows for a more robust and multi-dimensional evaluation of prediction accuracy. We applied two metrics across different models to ensure objective performance comparison.

In addition to these regression-based evaluation metrics, accuracy is considered as a key metric to evaluate the performance of classification models, particularly those used to detect user presence at home. Accuracy is defined as the ratio of correctly predicted instances to the total number of predictions made as shown in Eq. 13.

$$\text{Accuracy} = \frac{\text{Correct predictions}}{\text{All predictions}} \quad (13)$$

This metric is selected because it provides a straightforward and interpretable measure of overall model correctness in binary classification tasks such as user presence detection (i.e., present vs. not present). The results will be presented with the selected metrics in detail in the subsequent Section.

4. Results and Discussion

This section presents a comprehensive analysis of the performance of all trained models. The outcomes are assessed using standard metrics to ensure a consistent comparison. A thorough discussion is provided to justify the choice of the most appropriate model, considering its predictive accuracy, ability to generalize, and computational efficiency in the context of occupancy detection.

4.1. Comparative Analysis of Presence Proxy Classification Models

This section presents a comparative evaluation of six selected models applied to the classification of the "user presence" feature, a critical component in improving the effectiveness of electricity consumption prediction. The models are evaluated using classification accuracy as the primary metric. Neural Network, XGBoost, and Logistic Regression achieved the highest classification accuracy of 0.99, making them as the most effective models for this task.

The Neural Network demonstrated excellent capability in learning complex patterns, benefiting from multiple layers of abstraction that enabled it to generalize well across diverse data points. XGBoost, provided robust performance through iterative refinement and resistance to overfitting, particularly with tabular data.

Interestingly, Logistic Regression, a linear model, matched the performance of the more complex models, indicating that the feature space for this classification task may exhibit linear separability. The CNN model followed closely with an accuracy of 0.98. Although CNNs are primarily used in image and spatial data processing, their ability to capture local patterns through convolutional layers can be beneficial for structured or sequential datasets, such as time-aligned user activity records.

The Decision Tree model achieved a slightly lower accuracy of 0.96. Despite its interpretability and low computational cost, it tends to overfit in some cases, especially when the dataset contains noise or lacks sufficient regularization.

The SVM recorded the lowest performance, with an accuracy of 89.03%. SVMs are highly sensitive to hyperparameters and kernel selection; inadequate tuning can significantly affect their effectiveness, especially in cases where the data contains non-linear or overlapping classes. Table 2 presents the classification accuracy achieved by various machine learning models.

Model	Accuracy
Convolutional Neural Network (CNN)	0.98
Artificial Neural Network (ANN)	0.99
XGBoost	0.99
Logistic Regression	0.99
Decision Tree	0.96
Support Vector Machine (SVM)	0.89

Table 2: Overview of Algorithms Utilized for User Presence Feature Evaluation.

These results emphasize that although advanced ensemble and deep learning models deliver superior performance, simpler models like Logistic Regression can be highly effective when the feature structure is well-aligned with the classification goal. Selecting a model that balances accuracy, interpretability, and computational efficiency is essential.

4.2. Analysis and Interpretation of Electricity Consumption Prediction Models

XGBoost emerged as the best performing traditional machine learning model, achieving an RMSE of 0.01 and MAE of 0.13 when the user presence feature was included. Its ability to model complex non-linear interactions using gradient boosting contributed to its high accuracy. LightGBM followed closely, providing an excellent balance between speed and accuracy with an RMSE as low as 0.04, while being highly efficient in terms of memory usage. Random Forest also performed well, particularly when the presence feature was available, although its reliance on independently trained trees made it slightly less accurate than boosting methods. XGBoost and LightGBM provide consistently high performance, particularly when contextual features such as "User presence" are included.

Among the deep learning models, LSTM demonstrated exceptional performance, achieving RMSE values as low as 0.002 regardless of the presence feature. demonstrating its robustness in capturing temporal patterns directly from the data, even without contextual input, however, it remains susceptible to overfitting. In contrast, RNN performed significantly worse (RMSE up to 0.261), likely due to its known limitations with long-term dependencies. CNN, while theoretically applicable to time-series data through convolutional pattern recognition, underperformed in this context with RMSE around 0.19. indicating that CNNs are less suited for sequential electricity consumption modeling unless specifically customized. Table 3 compares the final evaluation results of all the implemented algorithms across the selected performance metrics. It emphasizes that by incorporating user presence as a contextual variable, the framework improves prediction performance and reflects more realistic energy usage patterns based on occupant behavior.

ML/DL Model	Without User Presence Feature		With User Presence Feature	
	MAE	RMSE	MAE	RMSE
XGBoost	0.29	0.087	0.13	0.01
Random Forest	0.05	0.172	0.009	0.05
LightGBM	0.07	0.169	0.013	0.04

Convolutional Neural Network (CNN)	0.09	0.197	0.08	0.19
Recurrent neural network (RNN)	0.11	0.261	0.05	0.12
Long short-term memory (LSTM)	0.02	0.002	0.019	0.002

Table 3: Overview of Algorithms Applied in Prediction Modeling

5. Limitations

Although Improved short-horizon forecasts can be utilized to reduce peak demand and overall energy waste through home energy management systems. For example, forecast-informed scheduling can shift discretionary loads (e.g., dishwasher/washer/dryer) to off-peak periods, reducing peak-to-average ratio and enabling higher penetration of renewables on the grid. However, the proposed model has some limitations:

- Occupancy/presence proxy validity: the presence proxy is inferred from energy signals and may not match true occupancy; results should be interpreted as behavior/activity context rather than ground-truth presence.
- Generalizability: performance may vary across households with different appliance portfolios, climates, and behavioral routines; external validation on additional homes is recommended.

6. Conclusion

This research is proposed to address the critical issue of rising energy consumption in households, where the excessive use of electrical appliances often leads to higher costs and environmental impacts. Many individuals remain unaware of their actual consumption patterns, making it difficult to adopt more efficient and sustainable energy practices.

In response to this problem, the primary objective was to develop a prediction model capable of analyzing electricity usage data, monitoring consumption patterns, and offering actionable recommendations for energy optimization. These objectives were fully achieved through the design and implementation of a robust hybrid machine learning model, which not only predicted future consumption patterns but also identified key controllable factors influencing energy usage.

The methodology adopted to achieve these outcomes involved a systematic approach: starting from problem definition, followed by data preprocessing and cleaning, model selection and training, and model evaluation. Each phase contributed critically to building a reliable and practical system that aligns closely with the project goals.

References

[1] "International Energy Agency (IEA). Global Energy Review 2025." Accessed: Jan. 24, 2026. [Online]. Available: https://www.iiea.com/energy/?gad_source=1&gad_campaignid=1065244934&gbraid=0AAAAAD30q8CE5Sb9wEb6MH2PCrb-7OkJ2&gclid=Cj0KCQiA-NHLBhDSARIsAIhe9X2HBM-V1ZDkpiYgD4mUK2CBJEqgMmvEjXF70t-BjKRFOZczi-I-iYEaAqA_EALw_wcB

[2] M. Tostado-Véliz, P. Arévalo, S. Kamel, H. M. Zawbaa, and F. Jurado, "Home energy management system considering effective demand response strategies and uncertainties," Energy Reports, vol. 8, pp. 5256–5271, Nov. 2022, doi: 10.1016/J.EGYR.2022.04.006.

[3] R. Bhol, S. C. Swain, R. Dash, K. J. Reddy, C. Dhanamjayulu, and B. Khan, "Short-Term Reactive Power Forecasting Based on Real Power Demand Using Holt-Winters' Model Ensemble by Global Flower Pollination Algorithm for Microgrid," Int. J. Energy Res., vol. 2023, no. 1, p. 9733723, Jan. 2023, doi: 10.1155/2023/9733723.

[4] A. D. Almoallem, "Electricity consumption analysis and management for different residential buildings in Jeddah, Saudi Arabia," International Journal of Energy Production and Management, vol. 6, no. 3, pp. 245–262, Sep. 2021, doi: 10.2495/EQ-V6-N3-245-262.

[5] M. S. Eid. Fahmy, F. Ahmed, F. Durani, Š. Bojnec, and M. M. Ghareeb, "Predicting Electricity Consumption in the Kingdom of Saudi Arabia," Energies 2023, Vol. 16, Page 506, vol. 16, no. 1, p. 506, Jan. 2023, doi: 10.3390/EN16010506.

[6] K. Sundaram, K. R. Sri Preethaa, Y. Natarajan, A. Muthuramalingam, and A. A. Y. Ali, "Advancing building energy efficiency: A deep learning approach to early-stage prediction of residential electric consumption," Energy Reports, vol. 12, pp. 1281–1292, Dec. 2024, doi: 10.1016/J.EGYR.2024.07.034.

[7] K. Elhabyb, A. Baina, M. Bellafkih, and A. F. Deifalla, “Machine Learning Algorithms for Predicting Energy Consumption in Educational Buildings,” Int. J. Energy Res., vol. 2024, no. 1, p. 6812425, Jan. 2024, doi: 10.1155/2024/6812425.

[8] N. Somu, G. R. M R, and K. Ramamritham, “A hybrid model for building energy consumption forecasting using long short term memory networks,” Appl. Energy, vol. 261, p. 114131, Mar. 2020, doi: 10.1016/J.APENERGY.2019.114131.

[9] I. W. A. Suranata, I. N. K. Wardana, N. Jawas, and I. K. A. A. Aryanto, “Feature engineering and long short-term memory for energy use of appliances prediction,” Telkomnika (Telecommunication Computing Electronics and Control), vol. 19, no. 3, pp. 920–930, Jun. 2021, doi: 10.12928/TELKOMNIKA.V19I3.17882.

[10] M. Faiq et al., “Prediction of energy consumption in campus buildings using long short-term memory,” Alexandria Engineering Journal, vol. 67, pp. 65–76, Mar. 2023, doi: 10.1016/J.AEJ.2022.12.015.

[11] A. N. Ndife, W. Rakwichian, P. Muneesawang, and Y. Mensin, “Smart power consumption forecast model with optimized weighted average ensemble,” IAES International Journal of Artificial Intelligence, vol. 11, no. 3, pp. 1004–1018, Sep. 2022, doi: 10.11591/IJAI.V11.I3.PP1004-1018.

[12] Z. Ferdoush, B. N. Mahmud, A. Chakrabarty, and J. Uddin, “A short-term hybrid forecasting model for time series electrical-load data using random forest and bidirectional long short-term memory,” International Journal of Electrical and Computer Engineering, vol. 11, no. 1, pp. 763–771, Feb. 2021, doi: 10.11591/IJECE.V11I1.PP763-771.

[13] M. K. M. Shapi, N. A. Ramli, and L. J. Awalin, “Energy consumption prediction by using machine learning for smart building: Case study in Malaysia,” Developments in the Built Environment, vol. 5, Mar. 2021, doi: 10.1016/J.DIBE.2020.100037.

[14] R. Olu-Ajayi, H. Alaka, I. Sulaimon, F. Sunmola, and S. Ajayi, “Building energy consumption prediction for residential buildings using deep learning and other machine learning techniques,” Journal of Building Engineering, vol. 45, p. 103406, Jan. 2022, doi: 10.1016/J.JOBE.2021.103406.

[15] Y. He and K. F. Tsang, “Universities power energy management: A novel hybrid model based on iCEEMDAN and Bayesian optimized LSTM,” Energy Reports, vol. 7, pp. 6473–6488, Nov. 2021, doi: 10.1016/J.EGYR.2021.09.115.

[16] E. Karapidakis, A. Paspatis, J. A. Gómez-Pulido, D. A. Musleh, and M. A. Al Metrik, “Machine Learning and Bagging to Predict Midterm Electricity Consumption in Saudi Arabia,” Applied System Innovation 2023, Vol. 6, Page 65, vol. 6, no. 4, p. 65, Jul. 2023, doi: 10.3390/ASI6040065.

[17] S. Liu, Z. Xie, and Z. Hu, “Optimizing smart home appliance energy monitoring using Factorial Hidden Markov Models for Internet of Behavior,” Journal of Building Engineering, vol. 97, p. 110732, Nov. 2024, doi: 10.1016/J.JOBE.2024.110732.

[18] M. Javaid, A. Haleem, R. P. Singh, S. Rab, and R. Suman, “Internet of Behaviours (IoB) and its role in customer services,” Sensors International, vol. 2, p. 100122, Jan. 2021, doi: 10.1016/J.SINTL.2021.100122.

[19] Y. Zhang and Q. Li, “A Regressive Convolution Neural network and Support Vector Regression Model for Electricity Consumption Forecasting,” Apr. 2019, Accessed: Dec. 22, 2025. [Online]. Available: http://arxiv.org/abs/1810.08878

[20] R. Wu, “Behavioral analysis of electricity consumption characteristics for customer groups using the k-means algorithm,” Systems and Soft Computing, vol. 6, p. 200143, Dec. 2024, doi: 10.1016/J.SASC.2024.200143.

[21] B. Prasetiyo, Alamsyah, and M. A. Muslim, “Analysis of building energy efficiency dataset using naive bayes classification classifier,” J. Phys. Conf. Ser., vol. 1321, no. 3, Nov. 2019, doi: 10.1088/1742-6596/1321/3/032016.

[22] A. Raza, L. Jingzhao, Y. Ghadi, M. Adnan, and M. Ali, “Smart home energy management systems: Research challenges and survey,” Alexandria Engineering Journal, vol. 92, pp. 117–170, Apr. 2024, doi: 10.1016/J.AEJ.2024.02.033.

[23] R. A. Rashid, L. Chin, M. A. Sarijari, R. Sudirman, and T. Ide, “Machine Learning for Smart Energy Monitoring of Home Appliances Using IoT,” International Conference on Ubiquitous and Future Networks, ICUFN, vol. 2019-July, pp. 66–71, Jul. 2019, doi: 10.1109/ICUFN.2019.8806026.

[24] M. Rambabu, N. S. S. Ramakrishna, and P. Kumar Polamarasetty, “Prediction and Analysis of Household Energy Consumption by Machine Learning Algorithms in Energy Management,” E3S Web of Conferences, vol. 350, May 2022, doi: 10.1051/E3SCONF/202235002002.

[25] R. Bonetto and M. Rossi, “Machine Learning Approaches to Energy Consumption Forecasting in Households,” Jun. 2017, Accessed: Dec. 22, 2025. [Online]. Available: http://arxiv.org/abs/1706.09648

[26] “Smart Home Dataset with weather Information.” Accessed: Oct. 08, 2025. [Online]. Available: https://www.kaggle.com/datasets/taranvee/smart-home-dataset-with-weather-information?select=HomeC.csv

[27] “Gentle Dive into Math Behind Convolutional Neural Networks | by Piotr Skalski | TDS Archive | Medium.” Accessed: Oct. 13, 2025. [Online]. Available: https://medium.com/data-science/gentle-dive-into-math-behind-convolutional-neural-networks-79a07dd44cf9

[28] “ReLU Activation Function in Deep Learning - GeeksforGeeks.” Accessed: Oct. 13, 2025. [Online]. Available: https://www.geeksforgeeks.org/deep-learning/relu-activation-function-in-deep-learning/

[29] “Activation functions in Neural Networks - GeeksforGeeks.” Accessed: Oct. 13, 2025. [Online]. Available: https://www.geeksforgeeks.org/machine-learning/activation-functions-neural-networks/

[30] “Everything You Need to Know About Logistic Regression - Spiceworks.” Accessed: Oct. 14, 2025. [Online]. Available: https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-logistic-regression/

[31] “Gini Index Formula | How to calculate Gini Index in Decision Tree?” Accessed: Oct. 14, 2025. [Online]. Available: https://www.upgrad.com/blog/gini-index-for-decision-trees/

[32] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 13-17-August-2016, pp. 785–794, Aug. 2016, doi: 10.1145/2939672.2939785;CSUBTYPE:STRING:CONFERENCE.

[33] L. Breiman, “Random Forests,” vol. 45, pp. 5–32, 2001.

[34] G. Ke et al., “LightGBM: A Highly Efficient Gradient Boosting Decision Tree”, Accessed: Jan. 26, 2026. [Online]. Available: https://github.com/Microsoft/LightGBM.

[35] D. P. Kingma and J. Lei Ba, “ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION”.

[36] K. Cho et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, pp. 1724–1734, 2014, doi: 10.3115/v1/d14-1179.

[37] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/NECO.1997.9.8.1735.