

Feature-Driven Electrical Load Prediction Using Interpretable Machine Learning Models and Meta-Ensemble Learning: Evidence from Mosul's 132 kV Grid Energy

Anas Thamer Mustafa 
anas.enp146@student.uomosul.edu.iq

Omar Sharaf Al-Deen Al-Yozbaky 
o.yehya@uomosul.edu.iq

Electrical Engineering Department, College of Engineering, University of Mosul, Mosul, Iraq

Received: October 4th, 2025 Received in revised form: January 11th, 2026 Accepted: January 19th, 2026

ABSTRACT

Accurate forecasting of electrical load plays a critical role in enhancing the efficiency, reliability, and operational planning of power distribution systems—particularly in regions facing infrastructure limitations and volatile consumption patterns. This study proposes a robust forecasting framework based on real daily load data obtained from the 132 kV Al-Intisar substation in Mosul, Iraq, covering the years 2022 to 2024. The dataset, sourced in collaboration with the Nineveh Electricity Distribution Directorate, provides a practical foundation for modeling realistic demand behavior in emerging grid environments. A domain-informed and carefully engineered set of 21 predictive features was designed by the authors, integrating temporal dynamics, statistical indicators, and thermo-sensitive variables such as load-to-temperature ratios, multi-scale lag features, and moving averages. Nine machine learning models were selected to represent diverse algorithmic paradigms—ranging from tree-based methods and classical regressors to ensemble learners. Among them, the Gradient Boosting model achieved the highest accuracy Coefficient of Determination = 0.986, Mean Absolute Error = 0.260 MW, followed closely by the Meta-Learner (Coefficient of Determination = 0.983), strategically introduced in this study as a novel stacked ensemble, which demonstrated strong generalization by integrating multiple high-performing base learners. Explainability was addressed through Shapley Additive Explanations analysis and feature importance visualization, confirming the predictive value of the engineered features and enabling interpretable insights into model behavior. The findings affirm that hybrid and tree-based models, when coupled with purposeful feature design and interpretability tools, offer scalable and high-performing solutions for long-term load forecasting in real-world power systems.

Keywords: Electric Load Forecasting, Machine Learning Models, Energy Demand Prediction, Stacking Ensemble, Time Series Analysis.

This is an open access article under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).
<https://jamh.uomosul.edu.iq/index.php/rengj>
Email: alrafidain_engjournal3@uomosul.edu.iq

1. INTRODUCTION

Accurate prediction of long-term electricity demand has become imperative for today's power grid operations. It can aid in planning decisions associated with resourcing, generation dispatch, and long-term grid planning. In contrast to short-term predictions over hours or days, long-term forecasts of a few years are critical for seasonal planning and infrastructure preparedness [1–3]. Having daily-resolution data for this purpose increases its temporal sensitivity as well as its variability and noise, which makes the design of the model more difficult [4,5]. Conventional statistical methods for this problem, such as (exponential smoothing methods) are

often inadequate to capture these nonlinear dynamics, especially in the presence of volatile and transitioning loads [6,7]. As a result, machine learning models have gained popularity in energy research for their ability to learn complex demand patterns and improve forecasting accuracy under real-world conditions [8,9].

Nonetheless, while machine learning (ML) algorithms such as XGBoost, LightGBM, and Long Short-Term Memory (LSTM) have demonstrated impressive performance in load forecasting, such models are frequently argued to be non-transparent. In practice, these models are typically treated as “black box” systems, revealing very little about how inputs affect their predictions

[10,11]. This black-box nature may stifle trust and utilization in operational settings that demand transparency and accountability [12]. To address it, interpretable AI methods have been proposed, such as SHAP (Shapley Additive Explanations), which uncover feature-level contributions, bringing model performance closer to interpretability [13]. But despite their increasing use, such tools are infrequently tested on actual granular data samples from the developing world. For example, Mosul, Iraq—undergoing reconstruction following the 2014–2017 ISIS occupation and the city’s 2017 liberation—lacks empirical forecasting studies using substation-level operational data, despite evolving demand and fragile infrastructure[14].

Artificial Intelligence (AI) and advanced algorithmic methods have played a key role in modern power-system operation and planning. They enable real-time prediction, dynamic demand response, and enhanced system resilience [15]. Artificial Intelligence (AI) and advanced algorithmic methods have played a key role in modern power-system analytics. They enable real-time prediction, enhance operational efficiency and load balancing, and handle high-frequency operational data to support decision-making in complex and data-constrained environments [16]. In addition, the ability to handle high-frequency energy data enables decision-makers to act more flexibly in complex situations and data-poor environments [17]. The application of interpretable AI techniques to such AI systems would also help underscore their utility in sensitive and/or mission-critical applications.

Considering these requirements, the current paper presents a reliable and interpretable machine learning model for long-term load forecasting in electricity using real operational data from the Al-Intisar 132kV substation in Mosul, Iraq. The suggested approach is tested on nine machine learning models, including tree-based learners, boosting algorithms, the stacked ensemble method, and Ridge Regression with Cross-Validation (RidgeCV) as the meta-learner of the ensemble. A unique space with 21 engineered variables (based on temporal, thermal, and statistical features) was used as input across all models. SHAP analysis was performed on the best models to improve transparency and enhance interpretability. Through the combination of high-fidelity modelling and interpretability, the framework targets a scalable approach to forecasting in emerging/post-crisis energy systems.

The structure of this paper is as follows: Section 2 presents the significance. Section 3 details the methodology, feature extraction, and

model architecture. Section 4 results and interpretability. Section 5 presents conclusions.

2. SIGNIFICANCE OF THE STUDY

The current study is a timely and very realistic issue that uses machine learning on real-life operational load information in Mosul, a city that has suffered years of infrastructure weakness and a high rate of demand increase. Unlike most previous investigations, which use simulated, benchmark, or highly aggregated publicly available data, this study uses raw daily substation measurements from the Al-Intisar 132 kV substation in the local distribution network to develop forecasting models that reflect real operating conditions. The proposed framework, combining interpretable ensemble learning methods with physically meaningful domain-specific features, not only provides accurate long-term and daily load predictions but also offers practical insights into the causes of demand variability. This is especially useful in small- and medium-sized utilities that are either under data scarcity or under monitoring, and where system reliability is delicate and vital.

The principal contributions of this research may be summarized as follows.

1. It builds a predictable, explainable machine-learning model to predict long-term daily loads using real operational data from an Iraqi 132 kV substation, thus contributing to empirical evidence on post-conflict, under-studied power systems.
2. It presents a feature-driven predictive architecture that relies on a 21-dimensional, domain-informed input space, combining temporal, thermal, and statistical predictors, and uses the same engineered variables with nine different machine learners, including tree-based, boosting, and a stacked ensemble with Ridge Regression Cross-Validation (RidgeCV) as the meta-learner.
3. It provides a single, stringent standard across these nine models using the same data, preprocessing, and feature engineering, isolating the impact of the learning algorithms themselves and enabling a fair comparison of performance.
4. It improves transparency by using SHAP analysis on the most effective models, which produces a comprehensive explanation of the role of calendar, temperature, and load-dynamics characteristics in the forecasts and

addresses the black-box characteristics of ensemble procedures.

5. The proposed framework provides a scalable and robust high-fidelity modelling approach to long-term load forecasting based on interpretability and low-data and noisy conditions, which is highly applicable to post-crisis and emerging energy systems.

3. METHODOLOGY

This section outlines the methodology for the current study to predict long-term daily loads at the Al-Intisar 132 kV substation. It initially describes the data acquisition, preprocessing, and feature engineering procedures in some detail, then elaborates on the range of machine learning models and the stacked meta-ensemble used. This is followed by a description of the training process, the adopted interpretability plan, and the evaluation metrics used.

3.1 Data description and forecasting setting

This paper uses operational data from the Mosul electrical network in Iraq, collected between 1 January 2022 and 31 December 2024. The data set consists of time-series data (daily) that has substation-level electrical load (in megawatts), ambient temperature, and calendar data (day, month, year), and population estimates of the served area. The target variable is the daily electrical load, expressed in megawatts (MW), which varies between 118.9 MW and 238.5 MW during the study period, reflecting common variations in demand at the substation within the local grid environment. A total of 1,096 daily observations are available. The data are split chronologically into a training set comprising 730 days (years 2022–2023) and a test set comprising 366 days corresponding to the leap year 2024, without shuffling, in order to preserve the temporal structure of the series and mimic a realistic forecasting setting.

In this work, long-term forecasting refers to the forecast of the entire daily load curve for the full upcoming calendar year (including all days of 2024) based solely on past measurements over the previous two years. As a result, two complete yearly cycles of the Al-Intisar 132 kV substation are sufficient to cover the 730 training days, which are sufficient to expose the predictive models to current seasonal variations, weekly consumption trends, and customary peak-to-peak variations. This structure is consistent with the planning-based forecasting needs of distribution networks, where operators require reliable annual demand patterns rather than just short-term (day-ahead or hour-ahead) forecasts.

3.2 Data preprocessing (missing values, noise, and outliers)

Before the models were built, the raw daily load and temperature series were visually analyzed to identify structural weaknesses, implausible values, and sudden spikes. The original operational records provided by the Nineveh Electricity Distribution Directorate did not have any structurally missing values. The sole observations that were added in later analysis were due to the creation of lagged and rolling-window characteristics (e.g. statistics calculated on 7-day, 14-day, 30-day, and 60-day rolling windows), which necessarily impact the first entries of the time series. Once all 21 engineered features had been created, rows with NaN values were removed via listwise deletion, thereby mostly removing the first days while maintaining the integrity and continuity of the rest of the series. This dataset was trained on all nine machine-learning models that were examined in this study (Decision Tree, Random Forest, Gradient Boosting, Bagging, XGBoost, LightGBM, AdaBoost, SVR, and the stacked ensemble).

In the present study, data noise and outliers were not explicitly filtered using denoising filters or outlier-removal algorithms, since high peaks and sudden changes in the daily load were considered legitimate high-demand events associated with local climatic and operational conditions rather than measurement errors. As a result, their effect was indirectly reduced in two complementary ways. To begin with, rolling-window statistics (7-day, 14-day, 30-day, and 60-day moving averages) were used as smoothing operators to minimize short-term variation in both load and temperature-related characteristics. Second, strong ensemble models (Random Forest, Gradient Boosting, Bagging, XGBoost, and LightGBM) were used, as tree-based ensembles are less sensitive to local noise and isolated extreme cases. The common preprocessing across all models ensured physically meaningful variability and numerical stability and robustness throughout the portfolio.

3.3 Feature Engineering

To enhance the predictive performance of machine learning models, a comprehensive feature engineering strategy was adopted, grounded in domain knowledge and data-driven insights. The original dataset, obtained from the Al-Intisar 132kV substation in Nineveh Electricity Distribution Directorate – Iraq, included daily electrical load, temperature, date, and population data. From this raw data, a total of 21 features were systematically generated and grouped into seven semantic categories, as illustrated in Fig. 1.

These categories include:

- **Calendar Features:** capturing temporal seasonality such as month, day, season, day_of_week, and is_weekend.
- **Demographic Feature:** represented by the population data, reflecting load variability due to socio-economic changes.
- **Lag Features:** including lag_1 and lag_7, to capture autocorrelation patterns.
- **Rolling Statistics,** such as avg_last_7, avg_last_14, avg_last_30, avg_last_60, std_last_7, and std_last_30, capture recent consumption trends and volatility.
- **Load Dynamics:** measuring first-order load changes, including delta_load and relative_change_load.
- **Temperature Metrics:** represented by temp_diff and avg_temp_7, to quantify short-term and smoothed temperature effects.
- **Composite Ratios:** such as load_temp_ratio and monthly_load_ratio, which normalize load by temperature or monthly averages to reflect context-aware behavior.

The monthly averages used in the monthly_load_ratio feature were calculated solely from the 2022-2023 training period and applied consistently to both the training and test samples, thereby avoiding unintended information leakage from the 2024 evaluation year into the feature construction process.

The authors did not extract all features from commercially available libraries; instead, they designed them specifically, thereby making it a major methodological contribution to the current study. The feature-design process focused on reproducibility, physical interpretability, and compatibility with other model families, which are considered vital for deployment in operational load-forecasting systems.

Moreover, to maintain evaluation fairness and scientific rigor, the same 21-dimensional feature set was provided to all nine machine-learning models- tree-based, boosting, bagging, kernel-based, and stacked ensemble architecture. This consistency in the application of engineered features guarantees that any observed performance differences can be explained solely by the nature of the algorithm, not by variation in the input representation.

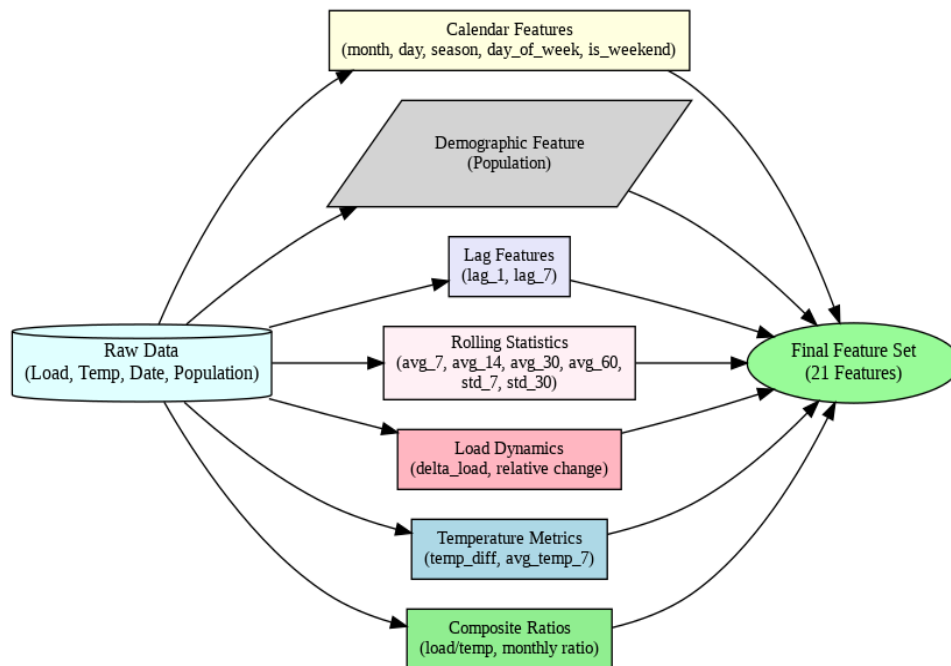


Fig.1 Feature engineering pipeline showing the structured transformation of raw variables into the final.

3.4 Model selection and taxonomy

The nine machine learning models used in this research were carefully selected in order to cover a balanced range of algorithmic paradigms, including single-tree (e.g., Decision Tree) and ensemble-based (Random Forest, Gradient

Boosting, Bagging, XGBoost, LightGBM, and AdaBoost), and even a kernel-based model (SVR) and a stacked meta-learning model. This choice of strategy provides methodological diversity, allowing the framework to capture both linear and extremely nonlinear dependencies and to reflect

best practices based on the current literature on medium- and long-term load forecasting. All models were tested under the same conditions and with the same set of engineered features, ensuring a fair and reproducible comparison. The research is therefore framed as a single standard, comprising nine machine learning models for long-term daily load prediction in a real-life Iraqi power system environment, rather than a set of independent contributions. In this benchmark, the stacked ensemble is considered a methodological contribution applied to combine the strengths of several high-performing tree-based learners.

In our current study, we use a strict and methodical taxonomy to categorize the nine machine learning algorithms in question, based on their theoretical basis and mathematical models. This taxonomy enables systematic comparison of model performance and can be used to explain the conceptual interrelationships among different families of algorithms. The algorithms are grouped into five major categories as shown in Fig. 2.

Despite the diversity of modelling philosophies represented by the chosen algorithms, each family has its own unique set of trade-offs that should be reasonably considered in the context of long-term load forecasting. Although decision-tree models are very interpretable, they are prone to over-fitting and produce volatile forecasts. Random Forest and

Conventional Bagging ensembles, which are used to reduce variance, are mostly opaque and do not explicitly model temporal dependencies. Gradient-boosting methods such as Gradient Boosting, XGBoost, LightGBM, and AdaBoost are often more predictive, but are prone to over-fitting

on small data sets, are black-box predictors, and are especially vulnerable to noise and outliers. Kernel-based Support Vector Regression provides non-linear modelling flexibility, but is computationally expensive, very sensitive to feature scaling, and has poor interpretability. Instead of ruling out such methodological families, the current study places their limitations within its scope by combining several high-performing tree-based learners into an open linear meta-model. As a consequence, the resulting stacked ensemble will inherit the overall strengths of its members, thereby increasing robustness and interpretability.

In comparison with the existing literature on long-term load forecasting, which often focuses on one of the so-called champion models, including a single neural network, an XGBoost instance, or a black-box hybrid, this study is different in several material aspects. To begin with, it uses real-life operational data collected at a 132 kV substation in Mosul, Iraq, and, therefore, does not rely on synthetic or highly aggregated system-level data. Second, the systematically engineered work provides a 21-dimensional, domain-based feature set that is explicitly adapted to local climatic and calendrical circumstances. Third, it provides a rigorous, detailed benchmark of nine machine-learning models in a single taxonomy, rather than evaluating a single or two algorithms separately. Lastly, it presents an interpretable stacking architecture, which combines highly predictive tree-based learners with a linear RidgeCV meta-model and SHAP-based explanations, explicitly balancing predictive accuracy and model transparency, which is typically overlooked in previous studies.

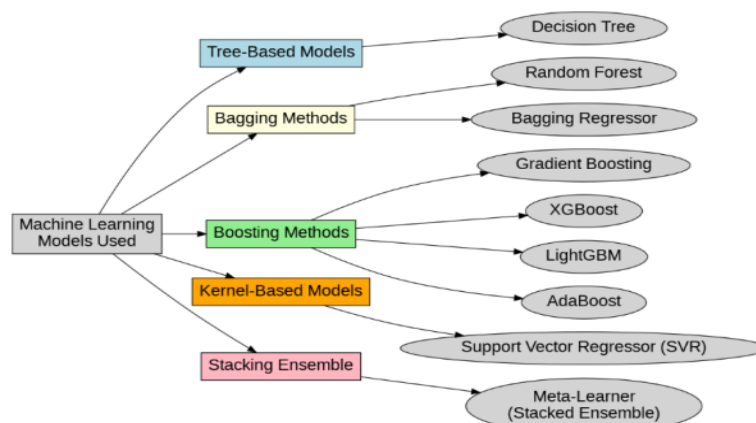


Fig. 2 The machine learning models Taxonomy in this study were categorized by the theoretical model structure and the category.

3.5 Individual forecasting models

The subsection outlines the machine-learning algorithms to be used in the current investigation. These models represent a broad

range of learning paradigms and can be used as independent predictors, as well as as base learners to the proposed stacked meta-ensemble.

3.5.1 Decision Tree Regressor

A Decision Tree Regressor is a non-linear statistical model with high interpretability and a hierarchical structure. It divides the input space into separate sub-regions through a series of decision rule applications and, hence, enables the extraction of complex dependencies embedded in multivariate time-series data. The model is widely used for task prediction due to its ability to handle heterogeneous inputs and the clarity of its decision logic [18–20].

The model is built based on a recursive partitioning plan, according to which the input domain is divided into sub-regions R_1, R_2 , and so on, R_M . The estimated response at every terminal node is then estimated as the local average of the target values in that region.

$$\hat{y}_m = \frac{1}{|R_m|} \sum_{x_i \in R_m} y_i \quad (1)$$

The best split at each inner node is found by minimizing the sum of squared errors over the branch:

$$\min_{(j,s)} [\sum_{x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2] \quad (2)$$

This form is based on the method CART (Classification and Regression Tree) described by Breiman et al. [21–23]. However, it tends to overfit and cannot model temporal dependencies, unlike ensemble or sequence-based models.

3.5.2 Random Forest Regressor

One of the most widely used ensemble learning methods is the Random Forest. It builds a large number of independent decision trees by using bootstrap sampling. The results from these trees are then pooled by taking the arithmetic mean of their predictions, which reduces variance and improves the model's generalization.

The terminal predicted value of a given observation can be given as follows:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T h_t(x) \quad (3)$$

where T represents the number of trees $h_t(x)$ is the final output, and t - is the output of the tree for input x . This architecture enables the model to capture structural variants in temporal and climatic data[24–26].

However, Random Forests may suffer from interpretability limitations and cannot explicitly model sequential dependencies, unlike temporal models such as LSTMs.

3.5.3 Bagging Regressor

Bootstrap aggregating (also known as bagging) is an ensemble learning methodology developed to reduce errors arising from the high variance of individual predictive models. The underlying idea of the method is to train several base models on different resampled subsets of the original data, each resampled by a resampling-with-replacement scheme. The forecasts of these constituent models are then combined by taking the arithmetic mean of these constituent models and thus a more stable and more accurate overall forecast is obtained. This method was proposed by Breiman and it works especially well with high-variance learners like decision trees. The ultimate predictive result is mathematically stated in Equation (4).

$$\hat{y} = \frac{1}{M} \sum_{m=1}^M f_m(x) \quad (4)$$

Where M denotes the number of base learners and $f_m(x)$ is the prediction of the m th learner for input x . [27,28]. However, while bagging effectively reduces variance, it offers limited interpretability and, like other ensemble methods, does not inherently model temporal dependencies across sequences.

3.5.4 GradientBoosting Regressor

Gradient boosting is a sequential ensemble method that builds additive models by minimising a loss function using the gradient descent algorithm. At every iteration, a new model is trained to reduce the residual errors of the existing ensemble until that point. This approach was initially described by Friedman, in which weak learners (mostly decision trees) are trained using the negative gradient of the loss function with respect to the current ensemble prediction. The overall model is described as follows:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \quad (5)$$

Where $F_m(x)$ is the ensemble prediction at iteration, $h_m(x)$ is the new weak learner trained on the residuals, and γ_m is the learning rate or shrinkage factor that controls the contribution of each learner [29–31]. However, gradient boosting models are prone to overfitting without careful tuning and, like other boosting methods, lack transparency and do not inherently model temporal dependencies.

3.5.5 XGBoost Regressor

XGBoost, also known as Extreme Gradient Boosting, is widely considered one of the most powerful ensemble learning methods for high-dimensional predictive modeling. Its strengths lie in its computational efficiency, scalability, and resistance to sparse and missing data. XGBoost, first proposed by Chen and

XGBoost, is an extension of the traditional gradient boosting framework that incorporates second-order derivatives and uses both L1 and L2 regularization to reduce overfitting and improve overall generalization.

The loss function of XGBoost can be formulated as:

$$\mathcal{L}(t) = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)}) + \sum_{k=1}^t \Omega(f_k) \quad (6)$$

Where l is the loss function, and $\Omega(f_k)$ represents the regularization term, defined as:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \quad (7)$$

This regularization penalizes model complexity through the number of tree leaves T and their corresponding weights w_j , improving the model's ability to generalize [32–34]. Despite its high accuracy and robustness, XGBoost is often treated as a black-box model, and like other tree-based ensembles, it lacks the capacity to learn temporal dependencies unless such features are manually engineered.

3.5.6 LightGBM Regressor

Light Gradient Boosting Machine (LightGBM) is a machine-learning algorithm developed by Microsoft with the peculiarity of high computational speed, computational efficiency, and the ability to work with large and complex data volumes. Unlike traditional algorithms like XGBoost, which use a level-wise tree expansion scheme (Level-wise), LightGBM uses a leaf-wise growth scheme (Leaf-wise) with a maximum depth parameter. In this context, the leaf that has the highest loss decrement is designated to be split, thus accelerating the training process and increasing the error-reducing effectiveness. LightGBM can therefore build stronger models that can better generalize, especially when faced with large and complex data.

The goal function, which LightGBM optimizes is based on a generalized version of the regularized gradient boosting algorithm.

$$\mathcal{L}(\theta) = \sum_{i=1}^n l(y_i, \hat{y}_i) + \lambda R(\theta) \quad (8)$$

Where $l(y_i, \hat{y}_i)$ is the differentiable loss function (e.g., squared error), and $R(\theta)$ is a regularization term penalizing model complexity. This structure helps prevent overfitting and ensures model sparsity and interpretability [35–37]. Despite its efficiency and accuracy, LightGBM can overfit on small datasets and is less interpretable due to its complex leaf-wise tree

structure, which also lacks inherent temporal modeling capabilities.

3.5.7 AdaBoost Regressor

One of the earliest and most significant ensemble learning algorithms, which was proposed by Freund and Schapire in 1997, is AdaBoost (Adaptive Boosting). The main objective of the algorithm is to reduce a weighted loss function among a number of weak learners. AdaBoost was initially designed to perform classification, and later was expanded to regression through AdaBoost.R2. AdaBoost, when used in regression, trains weak models, e.g. shallow decision trees, sequentially, with training data weights adjusted according to the size of prediction errors at each step. Therefore, every new model gives more weight to the samples that were hard to predict in the previous models. The overall shape of the ensemble output after

M iterations is given by:

$$\hat{y}(\mathbf{x}) = \sum_{m=1}^M \alpha_m h_m(\mathbf{x}) \quad (9)$$

Where $h_m(\mathbf{x})$ is the prediction from the m -th weak learner, and α_m is its associated weight, typically derived based on the learner's performance. The weights of instances that were predicted poorly are increased so that the subsequent learner focuses more on difficult examples[38–40]. However, AdaBoost is sensitive to noisy data and outliers because it emphasizes hard-to-predict samples, and it does not inherently model temporal dependencies as sequence-based models do.

3.5.8 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a variation of Support Vector Machine (SVM) that is targeted at regression tasks. It is particularly effective when the underlying data are nonlinear in nature, and the traditional ensemble tree models may not reflect this. The main idea of SVR is to project the original input data into a higher-dimensional feature space using a kernel function, most often the radial basis function (RBF), and then to construct a linear regression model in this higher-dimensional space. The model aims at estimating the actual output values and still within a given margin of tolerance, which is represented by ε .

The mathematical expression of SVR is as follows:

$$\min_{w, b, \xi_i, \xi_i^*} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (10)$$

Subject to the constraints:

$$\begin{aligned} y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle - b &\leq \varepsilon + \xi_i \\ \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle + b - y_i &\leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* &\geq 0 \end{aligned} \quad (11)$$

Here, $\phi(\mathbf{x})$ represents the mapping to the higher-dimensional space, $\mathbf{w}$ is the weight vector, b is the bias term, and ξ_i, ξ_i^* are slack variables allow deviations beyond the margin. The regularization parameter C balances the trade-off between model complexity and tolerance to deviation from the ε -margin. This formulation ensures that the model remains both accurate and sufficiently generalized [41,42]. However, SVR can be computationally intensive for large datasets and offers limited interpretability, while also lacking built-in mechanisms to capture temporal dependencies like recurrent models do.

3.6 Stacked model: meta-learner ensemble

In this paper, an integrated stacked ensemble framework is proposed specifically for long-term electric load prediction to compensate for the limitations of single-model forecasting methods. The proposed model takes full advantage of Gradient Boosting (GB), XGBoost (XGB), and Random Forest (RF) trees. Each of these models is independently trained on an identical set of 21 engineered features, including calendar-based indicators, statistical lag variables, temperature-related dynamics, and rolling-window metrics. This unified feature strategy ensures methodological consistency and allows for unbiased evaluation across learners.

The level-1 predictions from the three base models are subsequently passed to a level-2 meta-learner implemented using Ridge CV—an L2-regularized linear regression algorithm that performs internal cross-validation to optimize the

regularization parameter λ . The final prediction is generated by computing a linear weighted combination of the base model outputs as follows:

$$\hat{y} = \mathcal{M}_{\text{meta}}(\mathcal{M}_1(\mathbf{x}), \mathcal{M}_2(\mathbf{x}), \dots, \mathcal{M}_k(\mathbf{x})) \quad (12)$$

where $\mathcal{M}_i(\mathbf{x})$ denotes the prediction from the i -th base learner, and $\mathcal{M}_{\text{meta}}$ corresponds to the RidgeCV meta-learner. The objective function optimized by the meta-model is given by:

$$\mathcal{L}(\beta) = \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \|\beta\|^2 \quad (13)$$

In this formulation, $\mathbf{y}$ is the actual target vector representing daily load values, $\mathbf{X}$ is the matrix of predictions from the base models, β is the vector of regression coefficients, and λ is the regularization term that penalizes model complexity [43,44].

By integrating this regularized linear meta-learner, the ensemble not only reduces variance and mitigates multicollinearity among the base predictions but also ensures full interpretability through transparent coefficient attribution. The simplicity of Ridge CV enhances the traceability of the model's logic, distinguishing it from deep learning or black-box ensemble strategies. Moreover, the entire training pipeline is structured using a 3-fold cross-validation to eliminate data leakage and maintain model robustness.

Fig. 3 illustrates the full structure of the stacked ensemble, showing how the 21 engineered features are fed into the three base models, whose outputs are then linearly fused by RidgeCV to produce the final prediction. Each base learner is configured with 50 estimators to balance training efficiency with predictive performance.

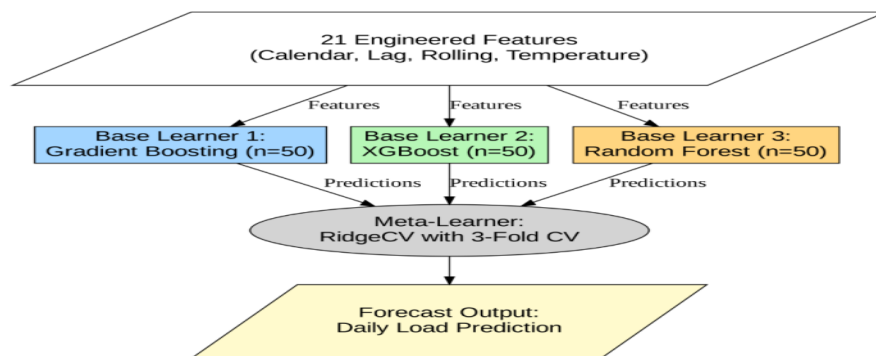


Fig. 3 The proposed stacked ensemble framework integrating GB, XGB, RF, and RidgeCV.

This architecture represents a clear methodological contribution in the field of energy forecasting. It integrates highly expressive tree-based models within a transparent linear meta-learning system, tailored to real-world electric load dynamics in

Iraq. Unlike previous stacking models that relied on complex deep networks or nonlinear combiners [45,46], this work achieves both high predictive performance and mathematical clarity.

The effectiveness of combining heterogeneous tree models with a linear meta-learner has been documented in recent literature. Mohan and Pachauri [47], for instance, reported that stacking XGBoost, Decision Trees, and RF outperformed standalone algorithms, including neural networks, in predicting building energy consumption. Similarly, Villalonga et al. [48] demonstrated that using Ridge regression as a meta-learner improved both transparency and accuracy in electric vehicle load prediction tasks. Other studies, such as those by Moon et al. [45] and Dostmohammadi et al. [49], emphasized that stacking enhances generalization and reduces variance in complex time-series environments.

Importantly, none of the reviewed works implemented the exact ensemble structure proposed here, specifically, a three-tree stacking configuration (GB, XGB, RF) integrated with Ridge CV. This distinction underscores the originality of this study, particularly given its deployment on actual operational load data from Iraq, making it both a novel and practical contribution to the energy forecasting literature.

3.7 Training procedure and hyperparameter settings

All machine learning models were implemented in Python using scikit-learn, XGBoost, and LightGBM. The time-varying data were split into two time periods, and models were trained only on the 2022-2023 data and tested on the isolated 2024 data, thereby preserving the chronological order and simulating a real-world forecasting scenario. To ensure reproducibility, the Decision Tree regressor and all tree-based ensemble algorithms were seeded with a constant `random_state`. In the case of the ensembles, Random Forest, Gradient Boosting, XGBoost, LightGBM, Bagging, and AdaBoost, the size of the estimators was limited to between fifty and one hundred, thus balancing the cost of computation with predictive accuracy. In particular, each standalone ensemble was used with one hundred estimators in the benchmarking stage, but the Gradient Boosting, XGBoost, and Random Forest sub-elements of the stacked ensemble were restricted to fifty estimators to reduce the computational cost. Other hyperparameters, such as maximum tree depth, learning rate, and subsampling ratios, were chosen based on library defaults, supplemented by an initial sensitivity analysis on the training data to avoid grid search exhaustion and maintain computational tractability and reproducibility. Support Vector Regression was trained with an RBF kernel with default regularization parameters, thereby offering a non-linear kernel baseline.

A three-fold cross-validation process was utilized in the stacked meta-ensemble to produce out-of-fold predictions of the three base learners of Gradient Boosting, XGBoost, and Random Forest. These out-of-fold predictions are the input matrix X for the RidgeCV meta-learner, which then selects the regularization parameter λ via internal cross-validation. This design clearly excludes the possibility of test information leaking into the training phase and provides a strong approximation of the meta-learner coefficients.

3.8 Full forecasting pipeline

Fig. 4 outlines the workflow used to compile the daily load forecasting model in the current study. The first stage of the pipeline involves purchasing raw input data, including daily electricity load, temperature measurements, calendar dates, and a population indicator. These variables were selected based on their relevance to demand patterns at the Al-Intisar 132 kV substation in the Nineveh Governorate. After the data collection, a series of preprocessing functions, as outlined in Section 3.2, are applied to clean and standardise the dataset and to create lagged and rolling-window features. During this step, structural gaps and implausible values are analyzed in the time series; missing values due to the construction of features are removed, and the effects of short-term noise and extreme values are reduced with the help of rolling-window statistics and the application of robust ensemble models. The resulting processed data is then used as a shared input to all nine machine-learning models. Section 3.3 defines feature engineering, which is used to derive more variables that capture temporal dynamics, statistical trends, and climatic effects. The result of this process is a filtered list of twenty-one domain-informed features that form the input space to model training. The development of models is then carried out using the wide range of algorithms presented in Sections 3.4 and 3.6, along with the training protocol and hyperparameter settings discussed in Section 3.7. The stacked ensemble combines the predictions of three powerful tree-based learners using a linear meta-model, which is the final phase of the forecasting process. Performance forecasting is tested on the held-out 2024 test year using four common error measures: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), coefficient of determination (R^2), and Symmetric Mean Absolute Percentage Error (SMAPE), whose formal definitions are provided in Section 3.9. This assessment plan provides detailed descriptions of absolute error size, larger-deviation penalty, explained variance, and scale-free percentage errors.

Overall, the proposed pipeline provides a rigorous and repeatable data-to-feature engineering-to-model training-to-performance evaluation pipeline. Its modular design makes it

scalable and adaptable to a wide variety of forecasting contexts, enabling its application in other case studies and distribution networks.

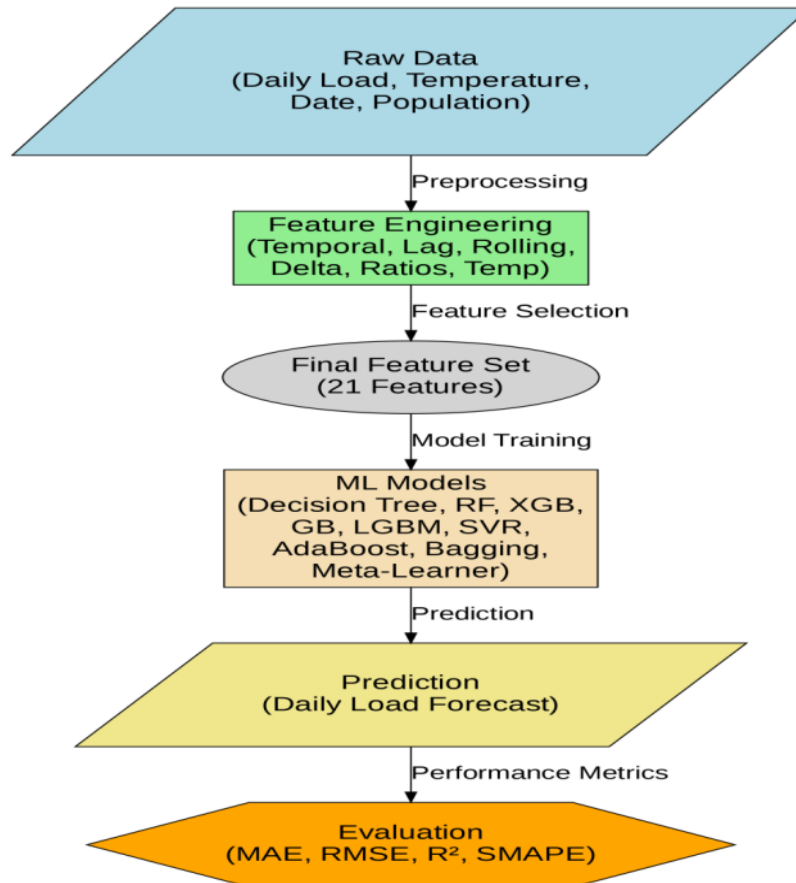


Fig. 4 Complete forecasting pipeline from raw data to evaluation.

3.9 Evaluation Metrics

All models were tested on the held-out test year of 2024 using the training-test partition described in Sections 3.1 and 3.7. Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), coefficient of determination (R^2), and Symmetric Mean Absolute Percentage Error (SMAPE) were four complementary measures that were used to collectively evaluate the magnitude of absolute error, the punishment of large deviations, the explained variance, and the scale-independent percentage errors. Together, these criteria represent both relative and absolute features of error, thus providing a complete picture of model performance in the context of energy-forecasting applications.

3.9.1 Mean Absolute Error (MAE)

MAE measures the average absolute size of the prediction errors, regardless of direction. It is easy to interpret and less sensitive to the

influence of outliers, which are the reasons for its popularity in energy load prediction [50–52].

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (14)$$

Where y_t is the actual load, $\hat{y}_t$ is the predicted load, and n is the number of data points

3.9.2 Root Mean Squared Error (RMSE)

RMSE is the root mean square error. It punishes larger errors more than MAE, which makes it more relevant to applications where you don't want large errors [53].

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \quad (15)$$

While RMSE shares the same units as the target variable, it is more sensitive to outliers than MAE.

3.9.3 Coefficient of Determination (R^2)

R^2 quantifies the proportion of variance in the actual values that the model explains. It is commonly used to evaluate model fit and generalization capacity [54–56].

$$R^2 = 1 - \frac{\sum_{t=1}^n (y_t - \hat{y}_t)^2}{\sum_{t=1}^n (y_t - \bar{y})^2} \quad (16)$$

Where $\bar{y}$ denotes the mean of actual values. Values closer to 1 imply better explanatory power.

3.9.4 Symmetric Mean Absolute Percentage Error (SMAPE)

SMAPE is a normalized percentage-based metric that avoids the issue of division by zero, unlike traditional MAPE. It is especially useful when comparing performance across varying load scales.

$$\text{SMAPE} = \frac{100\%}{n} \sum_{t=1}^n \frac{|y_t - \hat{y}_t|}{(|y_t| + |\hat{y}_t|)/2} \quad (17)$$

SMAPE ensures symmetry in penalizing under- and overestimation, making it well-suited for applications in volatile energy-demand contexts [57–60].

3.10 Interpretability strategy

Besides the fact that the proposed framework ensures numerical accuracy, it has an explicit interpretability layer. To examine linear dependencies, identify potential redundancy, and verify the coherence of the 21-dimensional feature space, a Pearson correlation analysis of the original and engineered features was conducted first. Second, Shapley Additive Explanations (SHAP) were calculated on a subset of models, which are Gradient Boosting, Random Forest, and the stacked meta-ensemble, the most performant tree-based predictors, and the main building blocks of the final forecasting system. In these models, SHAP values were computed with the TreeExplainer algorithm and used to decompose each prediction into additive contributions from all input features, enabling a data-driven analysis of both local (per-instance) and global (overall) feature influence. The strategy balances redundancy across the vast number of highly similar ensemble structures by centering the SHAP analysis on this representative subsample, providing a fine-grained understanding of the behaviour of the most powerful and useful models. This interpretability approach reduces the black-box nature of ensemble-based approaches and provides physically meaningful insights into how the models convert input data into daily load predictions, thereby increasing system operators'

and planners' capacity to understand, verify, and trust the resulting predictions.

3.11 Practical applicability to other Iraqi and regional systems

Although the empirical analysis in this paper is based on the Al-Intisar 132 kV substation in Mosul, the proposed forecasting framework has been specifically designed to apply to other Iraqi and regional distribution systems. The pipeline uses only quantities that are regularly measured or readily available in the majority of utilities, namely daily load, ambient temperature, calendar data (including weekends and public holidays), and crude demographic proxies. This would mean that to transfer the approach to a new substation or control area would only require: (i) gathering the historical daily load and temperature data at that location; (ii) re-training the models on the new historical data without changing the model structures; and (iii) re-training the models on the new historical data without changing the model structures.

In this respect, the framework can be successfully applied to other Iraqi or Middle Eastern systems of power with minimal retraining, assuming that sufficient historical data are available and that the main similar demand drivers are broadly available. However, the quantitative performance in every new setting will be based on the quality of the local data, climate, and consumption patterns.

4. RESULTS AND DISCUSSION

In this section, the results for the nine algorithms used with different measurement methods will be discussed.

4.1 Comparative Visualization of Forecasted Load

Fig. 5 presents a comprehensive visual comparison of the actual electrical load and predictions generated by nine machine learning models for 2024. Each subplot illustrates the temporal behavior of a specific model, while the bottom panel provides a unified overlay of all models against the actual load values. This layout enables simultaneous assessment of both individual and collective model performance. Notably, ensemble-based regressors such as Gradient Boosting, Random Forest, and the Meta-Learner (Stacked) exhibit strong alignment with the actual values across all seasonal periods, indicating superior predictive fidelity and robustness to fluctuations. In contrast, models like SVR demonstrate evident underfitting and

instability in capturing both baseline demand and peak variations. The consolidated subplot shows that the top-performing models tend to closely match the actual series, particularly during high-demand periods and transition periods between

seasons. This visualization supports the quantitative findings reported in subsequent sections and confirms the advantage of integrated ensemble architectures in achieving temporal generalization under real-world conditions.

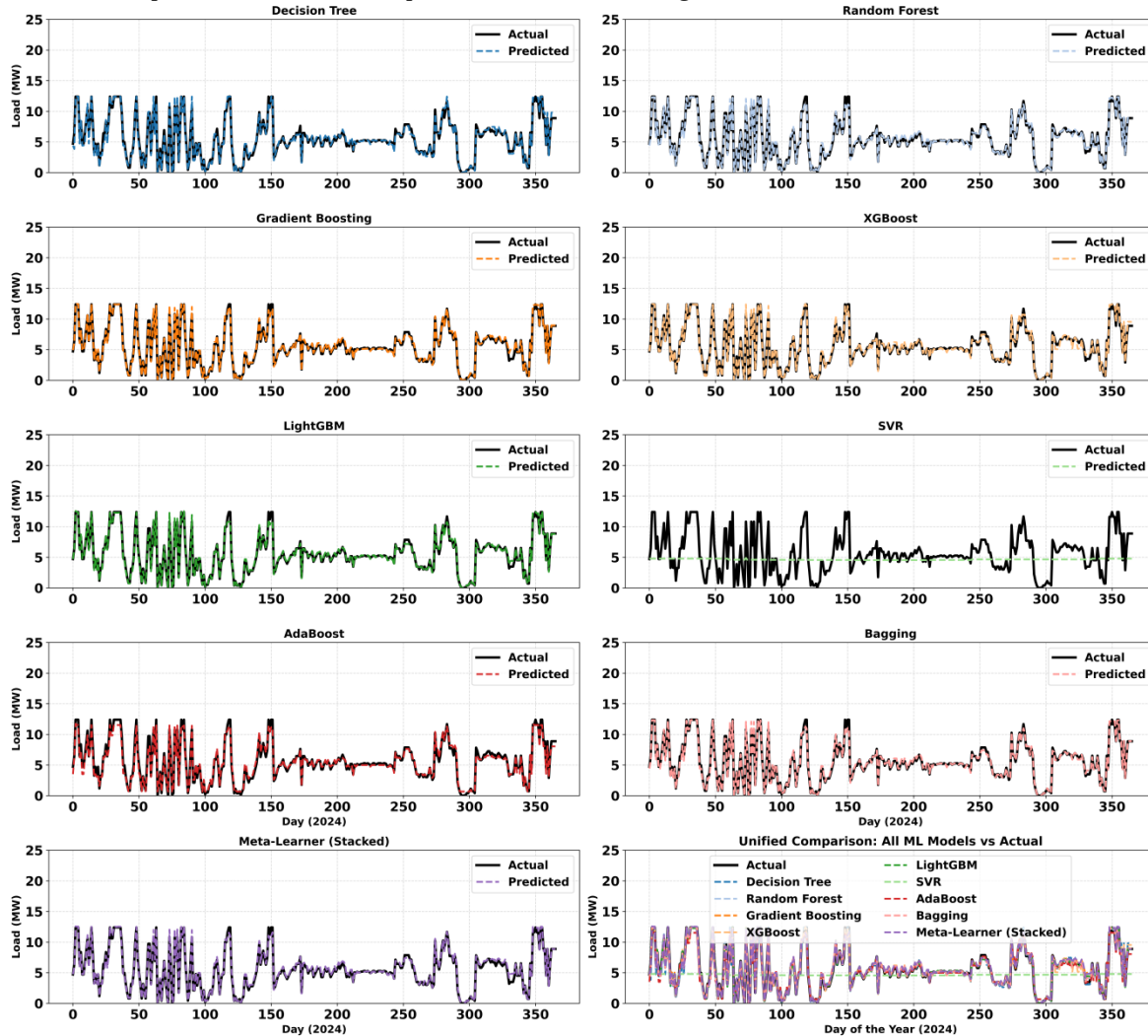


Fig. 5 Forecasting results of nine machine learning models vs actual load (2024).

4.2 Correlation-Based Feature Analysis and Interpretability

Fig. 6 presents the Pearson correlation heatmap of all engineered and original input features used in the forecasting models. This analysis was conducted as part of the feature selection process to ensure minimal redundancy and maximize informational diversity across the feature space. For example, in the heatmap, we find strong positive correlations among features such as temperature and avg temp_7 ($\rho = 0.96$) and between season and temperature ($\rho = 0.58$), a sign of seasonal climate conditions and their effect on load shape. LAG features like lag_1, avg_last_7, and avg_last_14 also showed significant correlations among themselves and with the target

variable, and thus were relevant for capturing the autocorrelation patterns in the daily load profile. On the other hand, time-related indicators (day_of_week, is_weekend, and Population) showed very small or zero correlations, reflecting their independence and possibly the model's generalization power. Low multicollinearity was generally observed among these features, suggesting that they were valid and worked in harmony without redundancy or instability during learning. Such correlation-based analysis also served as the basis for the design of a structurally sound feature set that was conducive to the interpretability and robustness of the predicted models, as demonstrated by the performance measures reported in Section 4.3.

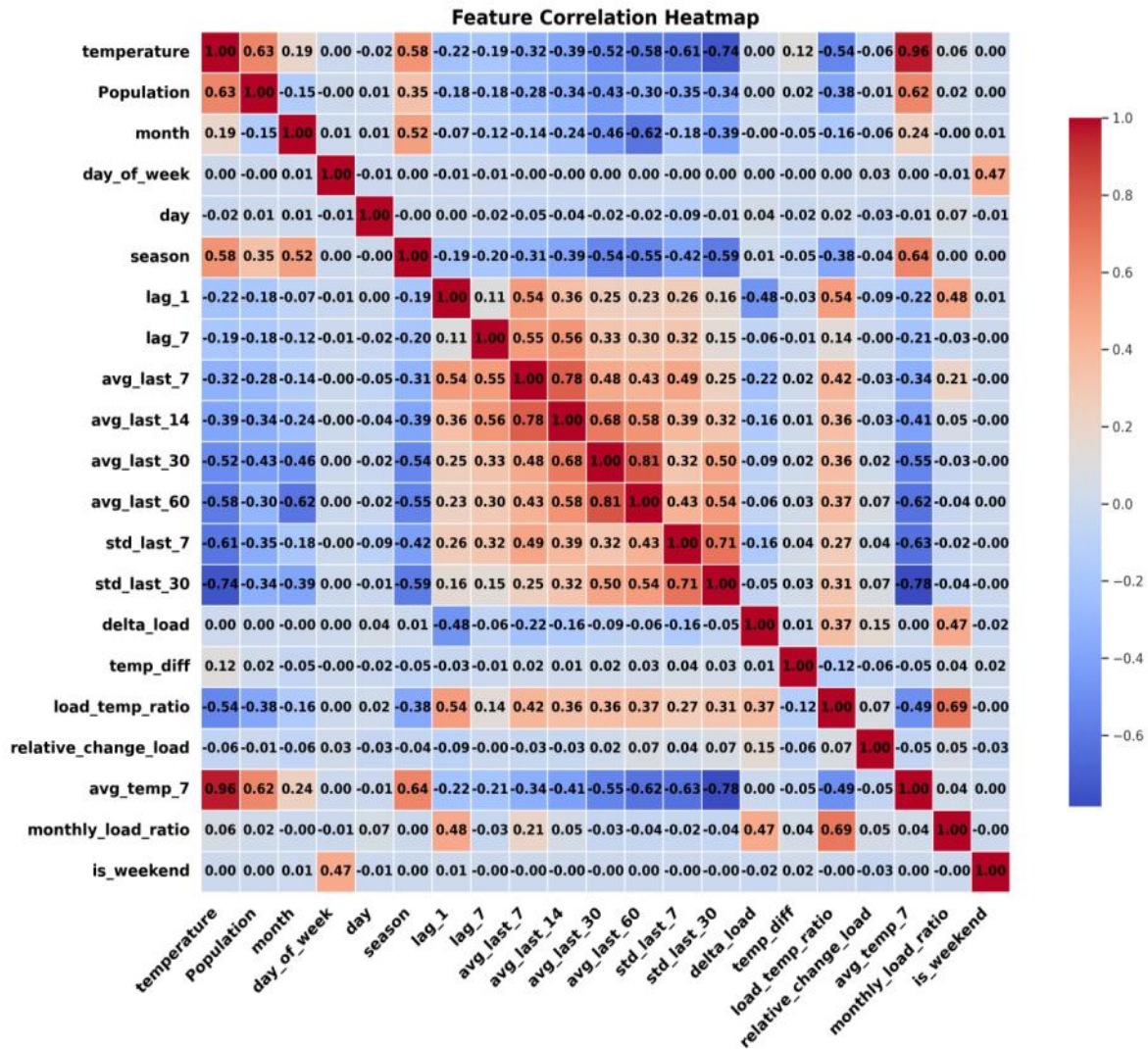


Fig. 6 Heatmap showing the correlation between the feature inputs used in the forecast.

4.3 Performance Evaluation Using SMAPE, MAE, RMSE, and R^2 Metrics

Fig. 7 shows the SMAPE values for each model, which are the proportional accuracy values for each forecast. The bagging model corresponds to the smallest SMAPE of 5.527028% (visually marked in green to indicate its relative superiority). On the contrary, SVR achieved the worst SMAPE of 52.648748% (indicated in red) due to its poor generalization and unpredictable predictions.

Fig. 8 extends the analysis by showing absolute error measures. Here again, Bagging and Random Forest demonstrated consistently strong results, with low MAE and RMSE values and high R^2 scores, suggesting accurate tracking of the target signal. Gradient Boosting achieved the

highest R^2 value (0.986305), indicating robust explanatory power. In contrast, SVR produced the highest absolute errors (MAE = 2.733530, RMSE = 3.991705) and a negative R^2 (-0.085160), confirming its failure to capture the underlying temporal dynamics.

The use of color-coded bars in the figures—green for the most accurate and red for the least—visually reinforces the statistical outcomes. Taken together, the results clearly highlight the strength of ensemble learning methods, particularly Bagging and Gradient Boosting, in delivering accurate and reliable long-term load forecasts under fluctuating and data-sensitive conditions.

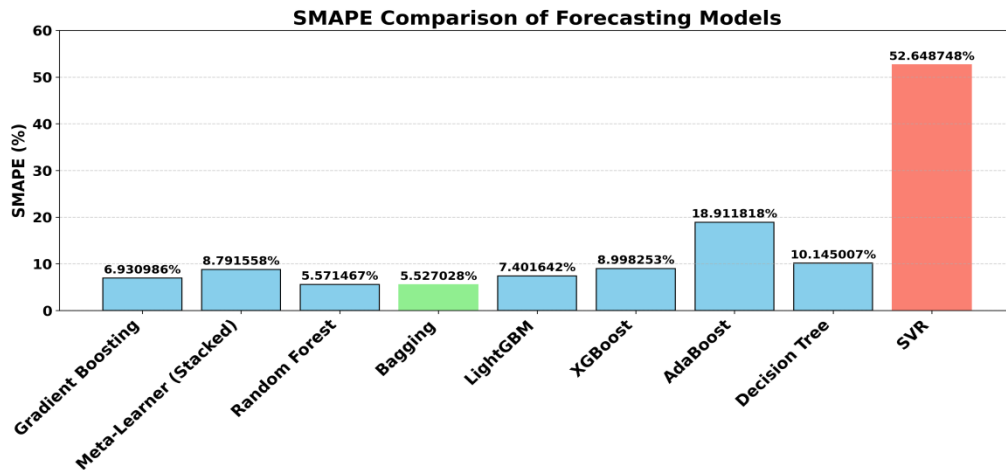


Fig. 7 Relative Forecast Error (SMAPE) Comparison.

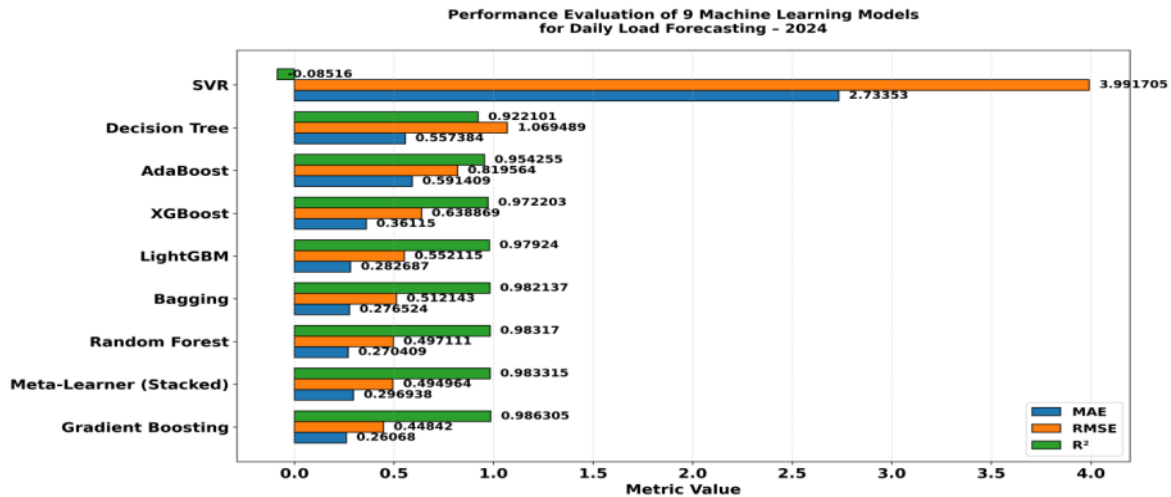


Fig. 8 Model Accuracy Metrics (MAE, RMSE, R²)

4.4 Residual Error Distribution Analysis

To supplement the absolute and relative error measures above, analysis of the residual error distribution was done on all nine forecasting models. Fig. 9 plots the distribution and central tendency of the prediction residuals, which are the differences between actual and predicted load values (Actual - Predicted). In contrast to scalar measures like MAE or RMSE, residual boxplots give a distributional view, allowing the evaluation of variability, asymmetry (skewness), and outlier behavior of each model. The mean residual in this figure is represented by an “x” mark, and the number of outliers (according to the 1.5xIQR criterion) is given in each model.

Gradient Boosting, Meta-Learner (Stacked), and Random Forest models have compact residual distributions with medians close to zero, indicating predictable, consistent

forecasting behavior with minimal variation. Their interquartile ranges are quite narrow, and their mean values are close to the median, indicating little bias and limited skewness across different demand conditions. Conversely, the Support Vector Regression model appears to have a significantly wider interquartile range and a highly asymmetric error distribution, with residuals exceeding the presented limits (as denoted by the arrows). This increased dispersion indicates reduced stability and poor generalisation across the dataset's temporal patterns. Although some models (e.g., XGBoost) can identify more outliers based on the adopted criterion, Support Vector Regression is clearly defined by the size and spread of its residuals, which highlights its relatively poor robustness in the forecasting task.

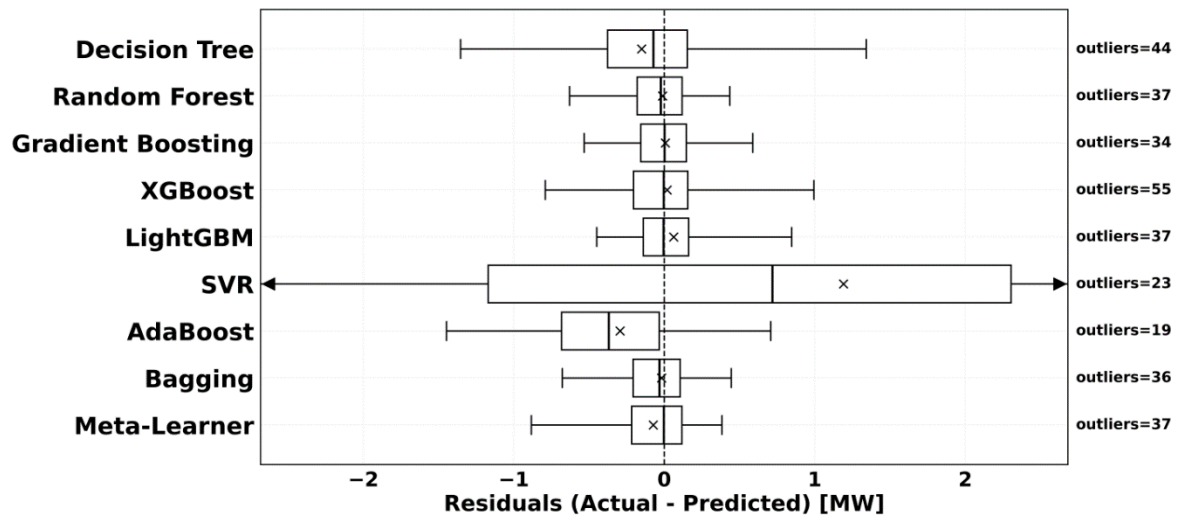


Fig. 9 Residual error distributions (Actual – Predicted) for all forecasting models, with mean markers and outlier counts.

4.5 Computational Efficiency Analysis of the Machine Learning Models

Fig. 10 illustrates a side-by-side comparison of the training durations required for each of the nine machine learning models examined in this research. The outcome clearly underscores the trade-off between model complexity and training efficiency. The Meta-Learner (Stacked) stood out with the longest training duration (4.60 seconds), which is understandable considering its ensemble architecture combining three powerful base learners—Gradient Boosting, XGBoost, and Random Forest—alongside a RidgeCV meta-regressor.

Conversely, both Decision Tree and SVR models demonstrated very short training times (0.037 and 0.087 seconds, respectively), yet their predictive outcomes, detailed earlier in Section 4.3, fell well below expectations. Among the more accurate algorithms, Gradient Boosting achieved a compelling balance between accuracy and computational cost, requiring only 0.67 seconds to train while attaining the highest R^2 score of 0.986. Altogether, these observations highlight how crucial it is to weigh both model precision and training speed when deploying models in real-time forecasting settings for energy systems

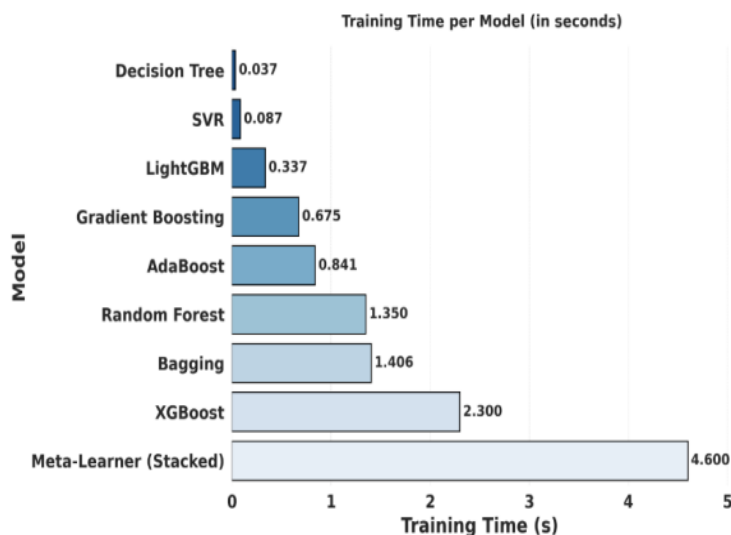


Fig. 10 Training time (s) of machine learning models ranked by computational efficiency.

4.6 Selected Models Justification for SHAP-Based Interpretability Analysis

While the study originally implemented and evaluated nine distinct machine learning algorithms, the SHAP-based interpretability analysis—illustrated in Fig. 11—was intentionally limited to the three most accurate and reliable models: Gradient Boosting, Random Forest, and the Meta-Learner (Stacked). This focused selection was guided by both scientific rigor and technical feasibility. First, these models consistently demonstrated superior forecasting performance, with R^2 values exceeding 0.98 and notably low MAE and RMSE scores, establishing them as the most trustworthy for feature attribution analysis. Second, SHAP's TreeExplainer, which ensures high interpretability and computational efficiency, is natively

optimized for tree-based models. Applying SHAP to other models, such as SVR, AdaBoost, or LightGBM, would have required using KernelExplainer. This more resource-intensive alternative is often unstable in ensemble settings and with high-dimensional datasets. Additionally, models with weaker generalization—evidenced by negative or low R^2 values—could lead to misleading interpretations if included. Therefore, to preserve the clarity, reliability, and scientific integrity of the interpretability results, the analysis was strategically centered on the most performant and SHAP-compatible models. This approach reinforces the transparency of the forecasting framework without compromising analytical depth or computational tractability.

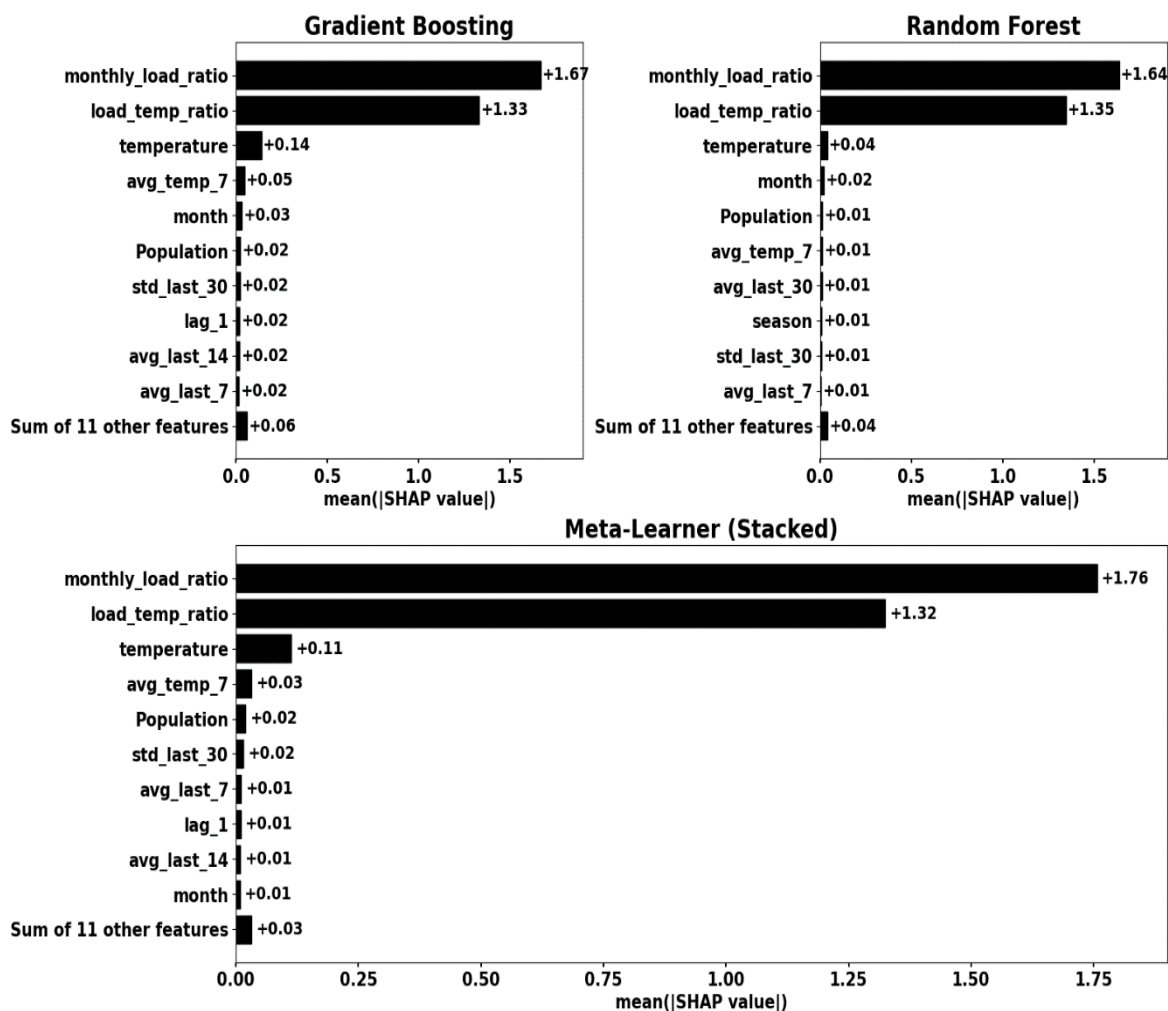


Fig. 11. SHAP feature importance for top-performing machine learning models.

5. CONCLUSION

The evaluation outcomes show that the point accuracy of Gradient Boosting was higher

than all the investigated models, and it also had the lowest MAE and the highest R^2 score ($R^2 = 0.986$, MAE = 0.260 MW, RMSE = 0.448 MW).

Nevertheless, the stacked ensemble model (Gradient Boosting + XGBoost + Random Forest with RidgeCV as a meta-learner) provided a good trade-off among accuracy, model interpretability, and model stability. SHAP

analysis also found that calendar features and short-term lags were the most influential

predictors of the predictions. These observations reinforce the importance of well-designed features and interpretable ensembles in cases of extremely small data and noisy grid environments like Mosul. The results of the study imply that interpretable, hybrid models may well be of great assistance for energy planning and operations decision-making in similar, resource-restricted environments.

ACKNOWLEDGEMENTS

The authors would like to thank University of Mosul, College of Engineering, Electrical department, for support given during this work.

Table 1: Model Abbreviations and Full Names

Abbreviation	Full Name
kV	kilovolt
MW	megawatt
R ²	Coefficient of Determination
MAE	Mean Absolute Error
SHAP	Shapley Additive Explanations
ML	Machine Learning
XGBoost	eXtreme Gradient Boosting
LightGBM	Light Gradient Boosting Machine
LSTM	Long Short-Term Memory
AI	Artificial Intelligence
ISIS	Islamic State of Iraq and Syria
RidgeCV	Ridge Regression with Cross-Validation
NaN	Not a Number
CART	Classification and Regression Tree
SVR	Support Vector Regression
SVM	Support Vector Machine
RBF	Radial Basis Function
GB	Gradient Boosting
XGB	eXtreme Gradient Boosting
RF	Random Forest
L1	L1 regularization (Lasso)
L2	L2 regularization (Ridge)
RMSE	Root Mean Squared Error
SMAPE	Symmetric Mean Absolute Percentage Error
MAPE	Mean Absolute Percentage Error
CV	Cross-Validation
SMAPE	Symmetric Mean Absolute Percentage Error
RMSE	Root Mean Squared Error
IQR	Interquartile Range

REFERENCES

[1] A. T. Mustafa and O. S. A.-D. Al-Yozbaky, "Forecasting Energy Demand and Generation Using Time Series Models: A

Comparative Analysis of Classical, Grey, Fuzzy, and Intelligent Approaches," *Frankl. Open*, p. 100350, Aug. 2025.

- [2] C. Kaur, J. Siddiki, and P. Singh, "Climate crisis, war and pandemic: How weather and global events have shaped UK electricity demand," *J. Environ. Manage.*, vol. 394, p. 126937, Nov. 2025.
- [3] I. Milstein, A. Tishler, and C.-K. Woo, "The effect of demand uncertainty on the optimal capacity mix of a wholesale electricity market," *Energy Econ.*, p. 109165, Jan. 2026.
- [4] W. Waheed, Q. Xu, M. Aurangzeb, S. Iqbal, S. H. Dar, and Z. M. S. Elbarbary, "Empowering data-driven load forecasting by leveraging long short-term memory recurrent neural networks," *Heliyon*, vol. 10, no. 24, p. e40934, Dec. 2024.
- [5] A. Th. Mustafa and O. Sh. Al-Yozbaky, "A Hybrid GBM-LSTM and Feature Engineering Residual Learning Approach for Medium-Term Load Forecasting Framework in Resource-Constrained Power Grids," in *2025 4th International Conference on Power Systems and Electrical Technology (PSET)*, Tokyo, Japan, 2025, pp. 831–836.
- [6] C. Yin and S. Mao, "Fractional multivariate grey Bernoulli model combined with improved grey wolf algorithm: Application in short-term power load forecasting," *Energy*, vol. 269, p. 126844, Apr. 2023.
- [7] W. F. Mbasso, A. Harrison, P. Jangir, I. Dagal, M. Khishe, A. Smerat, S. Chebaane, and T. Saidani, "Reliability-conscious power flow optimization in hybrid renewable microgrids: a case study in Sub-Saharan Africa using Gauss-Seidel and metaheuristic techniques," *Int. J. Electr. Power Energy Syst.*, vol. 173, p. 111350, Dec. 2025.
- [8] B. Biswal, S. Deb, S. Datta, T. S. Ustun, and U. Cali, "Review on smart grid load forecasting for smart energy management using machine learning and deep learning techniques," *Energy Rep.*, vol. 12, pp. 3654–3670, Dec. 2024.
- [9] F. Yaprakdal and M. Varol Arisoy, "A Multivariate Time Series Analysis of Electrical Load Forecasting Based on a Hybrid Feature Selection Approach and Explainable Deep Learning," *Appl. Sci.*, vol. 13, no. 23, p. 12946, Dec. 2023.
- [10] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

- [11] W. Liao, J. Fang, L. Ye, B. Bak-Jensen, Z. Yang, and F. Porte-Agel, "Can we trust explainable artificial intelligence in wind power forecasting?," *Appl. Energy*, vol. 376, p. 124273, Dec. 2024.
- [12] B. Assadhan, A. Bashaiwth, and H. Binsalleeh, "Enhancing Explanation of LSTM-Based DDoS Attack Classification Using SHAP With Pattern Dependency," *IEEE Access*, vol. 12, pp. 90707–90725, 2024.
- [13] M. Alizamir, M. Wang, R. M. A. Ikram, A. Gholampour, K. O. Ahmed, S. Heddami, and S. Kim, "An interpretable XGBoost-SHAP machine learning model for reliable prediction of mechanical properties in waste foundry sand-based eco-friendly concrete," *Results Eng.*, vol. 25, p. 104307, Mar. 2025.
- [14] R. Akpahou, L. D. Mensah, D. A. Quansah, and F. Kemausuor, "Energy planning and modeling tools for sustainable development: A systematic literature review," *Energy Rep.*, vol. 11, pp. 830–845, Jun. 2024.
- [15] G. Kobenkins, M. Marinbahs, A. Bizans, N. Rilevs, and O. Sliskis, "The influence of dynamic loads on the vibration level of rotating units of traction drives," *Energy Rep.*, vol. 9, pp. 131–137, May 2023.
- [16] S. Koshy, S. Rahul, R. Sunitha, and E. P. Cheriyan, "8 Smart grid-based big data analytics using machine learning and artificial intelligence: a survey," in *Artificial Intelligence and Internet of Things for Renewable Energy Systems*, N. Priyadarshi, S. Padmanaban, K. K. Hiran, J. B. Holm-Nielson, and R. C. Bansal, Eds. De Gruyter, 2021, pp. 241–278.
- [17] N. N. Thilakarathne, M. K. Kagita, S. Lanka, and H. Ahmad, "Smart Grid: A Survey of Architectural Elements, Machine Learning and Deep Learning Applications and Future Directions," *J. Intell. Syst. Internet Things*, pp. 32–42, 2021.
- [18] W. Loh, "Fifty Years of Classification and Regression Trees," *Int. Stat. Rev.*, vol. 82, no. 3, pp. 329–348, Dec. 2014.
- [19] J. Heinermann and O. Kramer, "On Heterogeneous Machine Learning Ensembles for Wind Power Prediction," *AAAI Workshop: Computational Sustainability*, 2015.
- [20] H. Chamlal, F. Aaboub, and T. Ouaderhman, "A preordnance-based decision tree method and its parallel implementation in the framework of Map-Reduce," *Appl. Soft Comput.*, vol. 167, p. 112261, Dec. 2024.
- [21] T. Januschowski, Y. Wang, K. Torkkola, T. Erkkilä, H. Hasson, and J. Gasthaus, "Forecasting with trees," *Int. J. Forecast.*, vol. 38, no. 4, pp. 1473–1481, Oct. 2022.
- [22] A. Brunello, E. Marzano, A. Montanari, and G. Sciavico, "J48SS: A Novel Decision Tree Approach for the Handling of Sequential and Time Series Data," *Computers*, vol. 8, no. 1, p. 21, Mar. 2019.
- [23] O. Gulaydin and M. Mourshed, "Machine learning for subnational residential electricity demand forecasting to 2050 under shared socioeconomic pathways: Comparing tree-based, neural and kernel methods," *Energy*, vol. 336, p. 138195, Nov. 2025.
- [24] T. Hastie, J. H. Friedman, and R. Tibshirani, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer, 2001.
- [25] M. Fellah, S. Ouhaibi, N. Belouaggadia, and K. Mansouri, "Energy consumption forecasting and thermal insulator selection with random forest regression," *Sci. Afr.*, vol. 29, p. e02870, Sep. 2025.
- [26] H. Z. Anonto, M. I. Hossain, A. Shufian, Md. S. Sayem, S. M. T. H. Shovon, P. P. Sheikh, and S. S. Alam, "Intelligent energy management of microgrids using machine learning: Leveraging random forest models for solar and wind power," *Results Eng.*, vol. 27, p. 106539, Sep. 2025.
- [27] I. D. Mienye and Y. Sun, "A Survey of Ensemble Learning: Concepts, Algorithms, Applications, and Prospects," *IEEE Access*, vol. 10, pp. 99129–99149, 2022.
- [28] A. K. Nayak, K. C. Sharma, R. Bhakar, and H. Tiwari, "Probabilistic online learning framework for short-term wind power forecasting using ensemble bagging regression model," *Energy Convers. Manag.*, vol. 323, p. 119142, Jan. 2025.
- [29] X. Yu, Z. Xu, X. Zhou, J. Zheng, Y. Xia, L. Lin, and S.-H. Fang, "Load Forecasting Based on Smart Meter Data and Gradient Boosting Decision Tree," in *2019 Chinese Automation Congress (CAC)*, Hangzhou, China, 2019, pp. 4438–4442.
- [30] J. C. M. Sánchez, H. G. A. Mesa, A. T. Espinosa, S. R. Castilla, and F. G. Lamont, "Improving wheat yield prediction through variable selection using Support Vector Regression, Random Forest, and Extreme Gradient Boosting," *Smart Agric. Technol.*, vol. 10, p. 100791, Mar. 2025.
- [31] M. Zambudio Martínez, L. H. M. D. Silveira, R. Marin-Perez, and A. F. S. Gomez, "Development and Comparison of

- Artificial Neural Networks and Gradient Boosting Regressors for Predicting Topsoil Moisture Using Forecast Data,” *AI*, vol. 6, no. 2, p. 41, Feb. 2025.
- [32] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco California USA, 2016, pp. 785–794.
- [33] L. Zhang and D. Jánosik, “Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches,” *Expert Syst. Appl.*, vol. 241, p. 122686, May 2024.
- [34] E. Aguilar Madrid and N. Antonio, “Short-Term Electricity Load Forecasting with Machine Learning,” *Information*, vol. 12, no. 2, p. 50, Jan. 2021.
- [35] U. von Luxburg, I. Guyon, S. Bengio, H. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett, and Neural Information Processing Systems Foundation, Eds., *Advances in neural information processing systems 30: 31st Annual Conference on Neural Information Processing Systems (NIPS 2017): Long Beach, California, USA, 4-9 December 2017*. Red Hook, NY: Curran Associates, Inc, 2018.
- [36] S. Simaiya, M. Dahiya, S. Tomar, N. Faujdar, Y. K. Sharma, and N. Nuristani, “A transfer learning-based hybrid model with LightGBM for smart grid short-term energy load prediction,” *Energy Explor. Exploit.*, vol. 42, no. 5, pp. 1853–1876, Sep. 2024.
- [37] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “LightGBM: A Highly Efficient Gradient Boosting Decision Tree,” in *Advances in Neural Information Processing Systems*, 2017, vol. 30.
- [38] S. K. Koteswararao and T. Rajendran, “Higher accuracy of drought prediction using logistic regression algorithm over adaboost algorithm,” presented at the *INNOVATIONS IN THERMAL, MANUFACTURING, STRUCTURAL AND ENVIRONMENTAL ENGINEERING: ICITMSEE’24*, Trichy, India, 2025, p. 020095.
- [39] S. Fan and L. Chen, “Short-Term Load Forecasting Based on an Adaptive Hybrid Method,” *IEEE Trans. Power Syst.*, vol. 21, no. 1, pp. 392–401, Feb. 2006.
- [40] C.-H. Chien, A. J. C. Trappey, and C.-C. Wang, “ARIMA-AdaBoost hybrid approach for product quality prediction in advanced transformer manufacturing,” *Adv. Eng. Inform.*, vol. 57, p. 102055, Aug. 2023.
- [41] J. M. Durango, C. González-Castaño, C. Restrepo, and J. Muñoz, “Application of Support Vector Machine to Obtain the Dynamic Model of Proton-Exchange Membrane Fuel Cell,” *Membranes*, vol. 12, no. 11, p. 1058, Oct. 2022.
- [42] P. Vrablecová, A. Bou Ezzeddine, V. Rozinajová, S. Šárik, and A. K. Sangaiah, “Smart grid load forecasting using online support vector regression,” *Comput. Electr. Eng.*, vol. 65, pp. 102–117, Jan. 2018.
- [43] J. Zhou, X. Tong, J. Zhou, R. Liu, and S. Bai, “Feature selection of power system data for frequency prediction with BO-LightGBM-Boruta,” *Energy AI*, vol. 21, p. 100581, Sep. 2025.
- [44] A. Mahboob, M. Rashad, G. Abbas, Z. Mushtaq, T. Mazhar, and A. U. Rehman, “Fortifying Industry 4.0 Solar Power Systems: A Blockchain-Driven Cybersecurity Framework with Immutable LightGBM,” *Comput. Mater. Contin.*, vol. 85, no. 2, pp. 3805–3823, 2025.
- [45] J. Moon, S. Jung, J. Rew, S. Rho, and E. Hwang, “Combination of short-term load forecasting models based on a stacking ensemble approach,” *Energy Build.*, vol. 216, p. 109921, Jun. 2020.
- [46] G. Fan, J. Xu, M. Li, T. Wei, and S. M. M. Nassabeh, “Implications of hot chemical–thermal enhanced oil recovery technique after water flooding in shale reservoirs,” *Energy Rep.*, vol. 6, pp. 3088–3093, Nov. 2020.
- [47] R. Mohan and N. Pachauri, “An ensemble model for the energy consumption prediction of residential buildings,” *Energy*, vol. 314, p. 134255, Jan. 2025.
- [48] J. T. Villalonga Palou, J. Serrano González, J. M. Riquelme Santos, and J. M. Roldán Fernández, “A novel weight-based ensemble method for emerging energy players: an application to electric vehicle load prediction,” *Energy AI*, vol. 20, p. 100510, May 2025.
- [49] M. Dostmohammadi, M. Z. Pedram, S. Hoseinzadeh, and D. A. Garcia, “A GA-stacking ensemble approach for forecasting energy consumption in a smart household: A comparative study of ensemble methods,” *J. Environ. Manage.*, vol. 364, p. 121264, Jul. 2024.
- [50] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, “Current methods and advances in forecasting of wind power

- generation,” *Renew. Energy*, vol. 37, no. 1, pp. 1–8, Jan. 2012.
- [51] J. Michańków, P. Sakowski, and R. Ślepaczuk, “Mean Absolute Directional Loss as a new loss function for machine learning problems in algorithmic investment strategies,” *J. Comput. Sci.*, vol. 81, p. 102375, Sep. 2024.
- [52] S. Xu, T. Li, X. Jia, H. Yang, and J. Liu, “A novel compensation method for absolute positioning accuracy errors of the end-effector of mobile robot systems considering the flexible deformations of tires and manipulator,” *Measurement*, vol. 254, p. 117772, Oct. 2025.
- [53] C. Fry and M. Brundage, “The M4 forecasting competition – A practitioner’s view,” *Int. J. Forecast.*, vol. 36, no. 1, pp. 156–160, Jan. 2020.
- [54] T. Uyanık, Ç. Karatug̃, and Y. Arslanoğlu, “Machine learning approach to ship fuel consumption: A case of container vessel,” *Transp. Res. Part Transp. Environ.*, vol. 84, p. 102389, Jul. 2020.
- [55] G. Bontempi, S. Ben Taieb, and Y.-A. Le Borgne, “Machine Learning Strategies for Time Series Forecasting,” in *Business Intelligence*, vol. 138, M.-A. Aufaure and E. Zimányi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 62–77.
- [56] B. E. Flores, “A pragmatic view of accuracy measurement in forecasting,” *Omega*, vol. 14, no. 2, pp. 93–98, Jan. 1986.
- [57] A. Wang, Q. Yu, J. Wang, X. Yu, Z. Wang, and Z. Hu, “Electric Load Forecasting Based on Deep Ensemble Learning,” *Appl. Sci.*, vol. 13, no. 17, p. 9706, Aug. 2023.
- [58] C. Tofallis, “A better measure of relative prediction accuracy for model selection and model estimation,” *J. Oper. Res. Soc.*, vol. 66, no. 8, pp. 1352–1362, Aug. 2015.
- [59] S. Liu and W. Sun, “Knowledge- and data-driven prediction of blast furnace gas generation and consumption in iron and steel sites,” *Appl. Energy*, vol. 390, p. 125819, Jul. 2025.
- [60] W. Chen, H. Huang, X. Ma, X. Xu, Y. Guan, G. Wei, L. Xiong, C. Zhong, D. Chen, and Z. Wu, “The short-term wind power prediction based on a multi-layer stacked model of BO CNN-BiGRU-SA,” *Digit. Signal Process.*, vol. 156, p. 104838, Jan. 2025.