Comparative Performance Evaluation of Machine Learning Models for Short-Term Residential Load Forecasting

Shokirjon Shoyusupov[1], Shomakhmud Azizov[2] and Abdulla Uzakov[3]
[1]*Department of Exact Sciences, Kimyo International University in Tashkent, Shota Rustaveli Str. 156, 100121 Tashkent, Uzbekistan*
[2]*Scientific Researcher of the Faculty of Informatics, Eötvös Loránd University, Pázmány Péter sétány 1/C, 1117 Budapest, Hungary*
[3]*Department of Physics, National Pedagogical University of Uzbekistan, Bunyodkor Str. 27, 100185 Tashkent, Uzbekistan*
shoyusupovshokirjon@gmail.com, shomahmuda@gmail.com, uzoqov@gmail.com

Keywords: Short-Term Residential Load Forecasting, Machine Learning, XGBoost, Random Forest, Linear Regression, Time Series Analysis, Applied Information Technology.

Abstract: Forecasting residential electricity demand a single hour ahead is a well-defined but persistently difficult problem, since household load fluctuates with occupant behavior in ways that simple statistical models struggle to capture. This paper compares three regression approaches for short-term residential load forecasting (STRF) - Linear Regression, Random Forest Regression, and Extreme Gradient Boosting (XGBoost) - under a single, controlled experimental pipeline. Data come from the publicly available Individual Household Electric Power Consumption dataset [1], originally logged at one-minute resolution between 2006 and 2010; after cleaning and aggregating to hourly resolution, the working dataset contains roughly 34,584 hourly observations, and the forecasting target is set one hour ahead. Alongside calendar-based predictors (hour of day, day of week, a weekend flag), two lagged variables, Lag_1 and Lag_24, were engineered to encode short-term momentum and daily periodicity in consumption. The dataset was split chronologically, 75% for training and 25% for testing, so that no future information leaks into the training process. Accuracy was assessed with three complementary metrics: MAE, MSE, and R^2. Across all three, XGBoost is the clear winner: relative to the Linear Regression baseline, it cuts MAE by 50.1% and MSE by 68.6%, and lifts R^2 from 0.5272 to 0.8518 - a 61.6% relative gain. These results indicate that boosting-based ensembles are considerably better suited than linear or purely bagging-based models at capturing the nonlinear, time-dependent structure of household electricity use, which is directly relevant for applied forecasting in smart energy systems.

1 INTRODUCTION

Electric power systems are experiencing increasing pressure due to rising electricity demand and the integration of smart technologies. Among various types of electricity demand, residential load presents unique challenges due to the growing number of consumers and complex consumption behavior. Unlike industrial loads, residential consumption exhibits irregular and time-dependent patterns [2]. These characteristics highlight the importance of reliable load forecasting techniques.

Residential Load Forecasting (RLF) refers to the process of predicting future electricity consumption for a household [3], [4]. A key component of RLF is Short-Term Residential Forecasting (STRF), which focuses on predicting electricity demand over short time horizons. Due to its short forecasting horizon, STRF plays a significant role in maintaining grid stability, reducing operational costs, and optimizing power generation schedules. Traditional forecasting methods are often ineffective for residential load forecasting because of nonlinear behavior, temporal variability, and complex human consumption patterns in electricity demand.

Short-Term Residential Forecasting (STRF) involves the estimation of future residential electricity demand based on historical consumption data over a time horizon ranging from one hour to one week [3], [5]. Accurate STRF improves the scheduling of generation resources, reduces energy waste, enhances grid reliability, and supports stable power system operation. In real-time operation, STRF provides essential information that helps power system operators manage grid stability in residential areas during peak demand periods. By aligning electricity generation with real-time demand, accurate STRF reduces operational costs and mitigates expenses associated with system imbalance and emergency interventions.

Despite the clear need for accurate short-term forecasts of residential electricity consumption, many traditional forecasting methods are based on linear assumptions and simplified statistical models [6]. These approaches often fail to capture the inherently nonlinear and time-varying nature of household electricity usage, which is strongly influenced by occupants' behavior, daily routines, and environmental factors. Consequently, conventional methods frequently produce limited forecasting accuracy, particularly during critical peak demand periods.

Machine learning offers a way around these limitations: nonlinear, data-driven models can, in principle, learn the complex temporal dependencies that linear methods miss [7]-[9]. Yet even with this progress in applying advanced learning methods to short-term load forecasting [8], [9], few studies place linear, bagging-based, and boosting-based regressors side by side under one identical preprocessing and evaluation pipeline. This paper addresses that gap through a structured comparison of Linear Regression, Random Forest, and XGBoost on a real household electricity consumption dataset for short-term residential load forecasting.

This work makes the following contributions:

1) A unified preprocessing and evaluation framework in which Linear Regression, Random Forest, and XGBoost are compared directly on the same short-term residential load forecasting task.
2) The development of a feature engineering pipeline incorporating temporal and lag-based features (hour, day of week, weekend, Lag_1, Lag_24) to capture both short-term and daily consumption patterns.
3) Empirical evidence that boosting-based methods, XGBoost in particular, better capture the nonlinear, time-dependent structure of residential electricity consumption than the linear and bagging-based alternatives tested here.

2 METHODOLOGY AND EXPERIMENTAL RESULTS

2.1 Dataset Description and Preprocessing

Dataset Description and Preprocessing. The experiments were conducted using an Individual Household Electric Power Consumption dataset obtained from the UCI Machine Learning Repository [1]. The dataset contains approximately 2 million minute-level electricity consumption measurements collected from a single residential household in France between December 2006 and November 2010. The recorded variables include active power, reactive power, voltage, intensity, and sub-metering measurements.

Since the original data contain missing observations, an initial data-cleaning procedure was performed. Missing values were handled using forward-fill interpolation to preserve temporal continuity and minimize information loss. In this study, the forecasting target variable is Global_active_power, which represents the household's total active power consumption measured in kilowatts (kW). After preprocessing, the minute-level data were aggregated into hourly averages to reduce noise and construct a structured time-series dataset suitable for short-term residential load forecasting. After aggregation, the target variable represents the average hourly active power consumption in kilowatts. The aggregation process resulted in approximately 34,584 hourly observations used for model training and evaluation. Approximately 1.2% of the raw minute-level readings were missing, predominantly as isolated single-minute gaps rather than extended outages; forward-fill was applied only to gaps of up to 60 consecutive minutes, propagating the last valid reading forward, since minute-to-minute household load changes gradually enough for this to be a reasonable approximation over such short spans. Longer gaps (a small number of multi-hour outages, together affecting under 0.3% of the minute-level record) were excluded from the dataset prior to hourly aggregation rather than filled, to avoid introducing artificial flat segments into the load series; after aggregation to hourly resolution and removal of the corresponding hours, 34,584 of the theoretically possible 35,064 hourly observations (2006-2010) remained (98.6% coverage).

Feature engineering was then applied to enhance the predictive capability of the models. Temporal features were extracted, including hour of the day, day of the week, and a weekend indicator variable. In addition, lag-based features were constructed to capture temporal dependencies in electricity consumption. Specifically, one-hour lag (Lag_1) and twenty-four-hour lag (Lag_24) features were generated to model short-term and daily seasonal patterns in residential load demand [10]. These engineered variables enable the models to learn both historical consumption behavior and periodic temporal effects inherent in residential electricity usage. To avoid data leakage, all lag features (Lag_1 and Lag_24) were computed on the full chronologically-ordered series before splitting, using only past values relative to each timestamp, and the train/test split boundary was then applied to this already-lagged feature table; consequently, a test-set row's Lag_1 and Lag_24 values may reference timestamps that fall within the last 24 hours of the training period (an unavoidable and standard boundary effect in chronological time-series splitting), but no test-period target values or test-period lag values were used to compute any training-set feature or label, so no leakage of test-set information into model training occurred.

To evaluate model generalization under realistic forecasting conditions, the dataset was divided chronologically into training and testing subsets. A 75:25 split was applied, where the first 75% of the time-oriented observations were used for training and the remaining 25% were reserved for testing. This chronological partition prevents data leakage and ensures that future observations are not used to predict past values, thereby reflecting short-term forecasting deployment scenarios.

2.2 Machine Learning Models

With the data prepared, the next step is training and evaluating the forecasting models themselves. The model lineup spans three levels of complexity: Linear Regression as the simple parametric baseline, Random Forest Regression as a bagging ensemble, and Extreme Gradient Boosting (XGBoost) as a boosting ensemble - the latter two included precisely because they are designed to pick up nonlinear, time-dependent structure of the kind household electricity use exhibits.

1) Linear Regression. The linear model expresses the target as a weighted combination of the predictors, with weights estimated from historical data and then reused to project values forward. Its appeal as a baseline lies in its simplicity and interpretability, both valued in load-forecasting studies, but the linearity assumption is also its main weakness: it cannot represent the nonlinear patterns and temporal dependencies that characterize residential electricity consumption.
2) Random Forest Regressor. A Random Forest pools the outputs of a large collection of decision trees, each grown on its own random draw of rows and features; the pooled prediction comes out steadier and more accurate than what any individual tree delivers [11]. That injected randomness is also the mechanism that restrains overfitting and supports generalization.

   Because it can represent nonlinear relationships and interactions among features, Random Forest is a reasonable fit for residential load forecasting [12], though it comes at the cost of heavier computation and less interpretability than a linear model.
3) Extreme Gradient Boosting Regressor (XGBoost). XGBoost comes from the gradient-boosting branch of the ensemble family. Where Random Forest averages trees trained independently, boosting grows them sequentially, each new tree fitted to whatever error the ones before it left behind. That sequential error-correction, combined with built-in regularization and an efficient tree-construction routine, is what lets XGBoost model intricate nonlinear and temporal structure while still controlling overfitting - and it is a large part of why the algorithm performs well in time-series applications such as short-term residential load forecasting [4].

2.3 Experimental Setup, Model Configuration, and Evaluation Metrics

To ensure a fair, transparent, and reproducible comparison of the forecasting approaches, the experimental environment, model configurations, and evaluation criteria were defined consistently for all investigated methods. The experiments were implemented using Python 3.8 on a standard workstation, providing a practical and easily reproducible computational environment. Numerical operations and array-based processing were performed using NumPy, while Pandas was used for data loading, preprocessing, transformation, and feature construction. Matplotlib was applied for visualization of forecasting results, enabling a clear comparison between observed and predicted electricity consumption values across different forecasting horizons.

The machine learning models were implemented using established Python libraries with widely adopted and documented interfaces. Scikit-learn was used for the development and evaluation of the Linear Regression baseline and the Random Forest Regressor model, while the XGBoost library was applied for implementing the Extreme Gradient Boosting Regressor. All models were evaluated using identical datasets and regression metrics from the Scikit-learn metrics module to ensure comparability of the obtained results.

The model configurations were selected to represent different methodological approaches while maintaining experimental consistency. The Linear Regression model was implemented using the default Scikit-learn configuration without regularization or feature transformation, serving as a simple parametric baseline for evaluating the benefits of nonlinear approaches. Despite its simplicity, this model provides an important reference point for assessing the contribution of more advanced machine learning methods and identifying the limitations of linear assumptions in residential electricity load forecasting.

The Random Forest Regressor was configured using 100 decision trees (n_estimators = 100), with random_state fixed at 42 to guarantee reproducibility. Other parameters, including bootstrap sampling and the squared-error splitting criterion, were maintained at their default Scikit-learn values. The ensemble structure of Random Forest enables the model to capture nonlinear relationships and feature interactions while reducing prediction variance through averaging multiple decorrelated decision trees.

The XGBoost Regressor was implemented using the XGBoost library with 100 boosting iterations (n_estimators = 100), a learning rate of 0.1, and random_state = 42. Remaining parameters were retained at their default values. The boosting mechanism allows successive weak learners to correct previous prediction errors, enabling the model to capture complex nonlinear patterns in electricity consumption data while maintaining generalization capability.

The performance of all forecasting models was evaluated using three standard regression metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), and the Coefficient of Determination (R^2) [13]. These metrics were selected because they provide complementary perspectives on forecasting accuracy. MAE measures the average absolute difference between predicted and observed values and directly represents the typical forecasting error in the original units of electricity consumption. MSE emphasizes larger prediction errors by applying a squared penalty, making it sensitive to significant forecasting deviations that may have operational consequences. R^2 evaluates the proportion of variance explained by the model and provides an overall measure of predictive capability.

Together, these evaluation criteria provide a comprehensive assessment of model performance: MAE reflects the average magnitude of errors, MSE identifies sensitivity to large deviations, and R^2 indicates the explanatory power of each forecasting approach. The combined use of these metrics allows a balanced comparison of linear and machine-learning-based forecasting models for residential electricity consumption prediction.

3 RESULTS AND DISCUSSION

This section reports and interprets the experimental outcomes, ranking Linear Regression, Random Forest Regression, and XGBoost on the metrics defined above - MAE, MSE, and R^2.

Table 1 collects the scores; the pattern clearly favors the ensemble methods over the linear baseline.

The gap between the models is unambiguous. The Linear Regression baseline finishes last on every count - largest MAE and MSE, smallest R^2. This outcome indicates that the relationship between the input features and residential electricity consumption cannot be adequately modeled using a simple linear approach.

Random Forest Regression improves upon Linear Regression by reducing both MAE and MSE and slightly increasing the R^2 score. This improvement suggests that tree ensembles pick up nonlinearities and feature interactions in the load data that a linear fit cannot reach. However, the performance gain remains limited, particularly in modeling complex temporal dependencies.

The XGBoost Regressor significantly outperforms both baseline models, achieving the lowest error values and the highest R^2 score of 0.8518. The sharp drop in both MAE and MSE shows how much of the consumption data's nonlinear and temporal structure XGBoost manages to absorb.

This gain traces back to the boosting mechanism itself: by fitting each new tree to the residual errors of the ones before it, and by leaning on built-in regularization to keep overfitting in check, XGBoost progressively corrects its own mistakes over successive iterations. The net effect is that it captures the intricate structure of residential consumption more effectively than either the linear or the bagging-based alternative.

Table 1: Forecasting accuracy of the three models on the held-out test set.

Model	MAE	MSE	R^2 Score
Linear Regression	0.3887	0.3000	0.5272
Random Forest	0.3689	0.2825	0.5548
XGBoost Regressor	0.1938	0.0941	0.8518

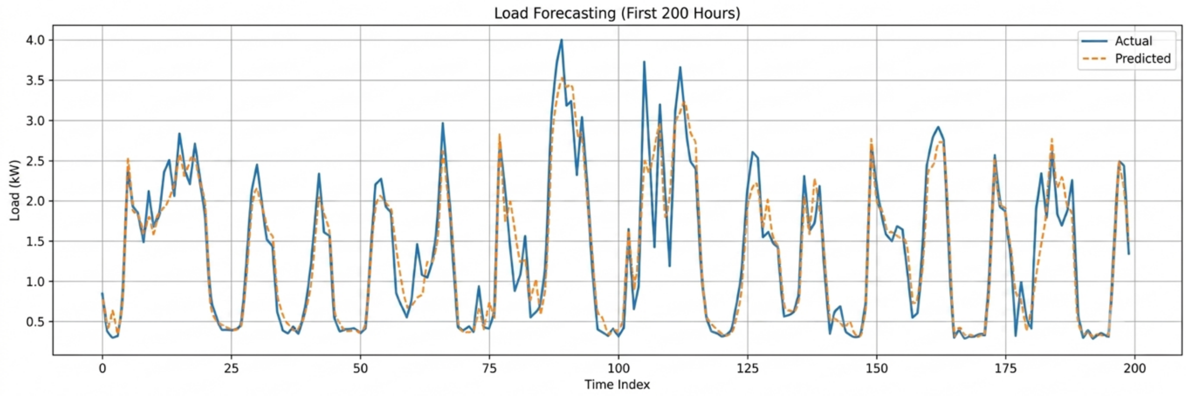

Figure 1: Comparison between actual and predicted residential load using the XGBoost model (first 200 hours).

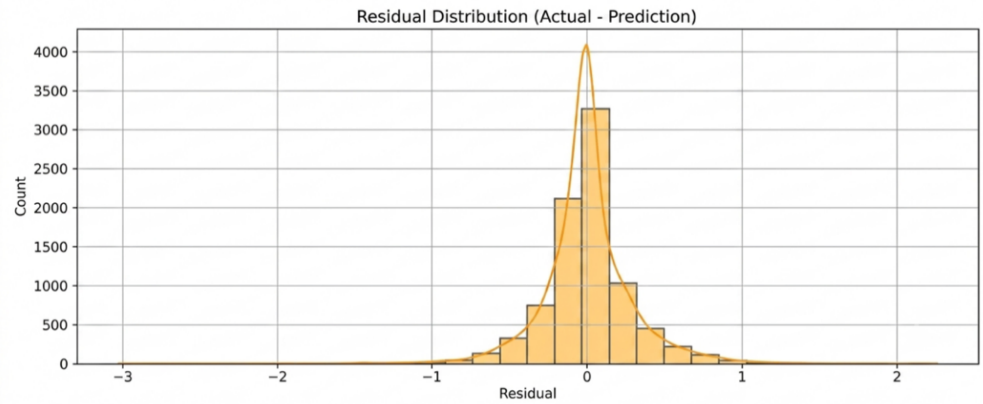

Figure 2: Residual distribution of the XGBoost model predictions.

As illustrated in Figure 1, the predicted values closely follow the actual load profile, accurately capturing periodic fluctuations and peak demand intervals.

The residual distribution in Figure 2 reinforces this picture: it is centered near zero and roughly symmetric, which points to an absence of systematic bias and supports the model's robustness.

These findings agree with recent work [8], [9] on the effectiveness of advanced machine learning and ensemble approaches for short-term load forecasting, though the present study differs in focusing specifically on a controlled, head-to-head comparison of models within one unified preprocessing and evaluation framework.

Taken together, the results point to XGBoost as the strongest model for this dataset and task, and a credible candidate for real-world short-term residential load forecasting.

4 LIMITATIONS AND FUTURE WORK

A few limitations are worth acknowledging despite the strong overall performance. The analysis rests on data from a single household, which limits how far the findings can be generalized to broader residential or multi-household settings. The models also lean entirely on historical consumption and engineered temporal features; exogenous variables such as temperature, weather, or socio-behavioral factors - all of which can meaningfully shape electricity demand - were not included.

Future research can extend this work in several directions:

1) Incorporating external factors such as weather data and temperature measurements to enhance forecasting accuracy and robustness.
2) Testing recurrent deep-learning architectures - LSTM and GRU networks - as a route to richer temporal modeling [14].
3) Performing systematic hyperparameter optimization and time-series cross-validation to further improve model generalization.
4) Deploying the trained model as a RESTful API using Flask or FastAPI for real-world applications.
5) Developing real-time visualization dashboards to monitor predictions and model performance dynamically.

5 CONCLUSIONS

This paper ran a linear baseline, a bagging ensemble, and a boosting ensemble - Linear Regression, Random Forest Regression, and XGBoost - through the same short-term residential load forecasting task on historical hourly consumption data.

The baseline models - Linear Regression and Random Forest Regression - turned out to have relatively limited predictive power on this dataset, suggesting that simpler, bagging-based approaches on their own cannot fully capture the nonlinear, time-dependent character of residential electricity consumption.

XGBoost, by contrast, led on every metric - MAE, MSE, and R^2 alike - evidence that it absorbs the nonlinear and time-dependent structure of residential load profiles far more completely.

On balance, the findings position Extreme Gradient Boosting as a robust, dependable choice for short-term residential load forecasting, fit for real-world applications that demand both high prediction accuracy and strong generalization.

REFERENCES

[1] G. Hebrail and A. Berard, "Individual Household Electric Power Consumption," UCI Machine Learning Repository, 2006, [Online]. Available: https://doi.org/10.24432/C58K54.

[2] G. Chicco, "Overview and performance assessment of the clustering methods for electrical load pattern grouping," Energy, vol. 42, no. 1, pp. 68-80, Jun. 2012, [Online]. Available: https://doi.org/10.1016/j.energy.2011.12.031.

[3] H. L. Willis and J. E. Northcote-Green, Power Distribution Planning Reference Book, 2nd ed. Boca Raton, FL, USA: CRC Press, 2004.

[4] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining (KDD), San Francisco, CA, USA, 2016, pp. 785-794, [Online]. Available: https://doi.org/10.1145/2939672.2939785.

[5] S. Fan and R. J. Hyndman, "Short-term load forecasting based on a semi-parametric additive model," IEEE Trans. Power Syst., vol. 27, no. 1, pp. 134-141, Feb. 2012, [Online]. Available: https://doi.org/10.1109/TPWRS.2011.2162082.

[6] H. Hippert, C. Pedreira, and R. Souza, "Neural networks for short-term load forecasting," IEEE Trans. Power Syst., vol. 16, no. 1, pp. 44-55, Feb. 2001, [Online]. Available: https://doi.org/10.1109/59.910780.

[7] Z. Qinghe, X. Wen, H. Boyan, W. Jong, and F. Junlong, "Optimised extreme gradient boosting model for short term electric load demand forecasting of regional grid system," Sci. Rep., vol. 12, Art. no. 19282, 2022, [Online]. Available: https://doi.org/10.1038/s41598-022-22024-3.

[8] P. Koukaras, A. Mustapha, A. Mystakidis, and C. Tjortjis, "Optimizing building short-term load forecasting: A comparative analysis of machine learning models," Energies, vol. 17, no. 6, Art. no. 1450, Mar. 2024, [Online]. Available: https://doi.org/10.3390/en17061450.

[9] H. J. Bae, J.-S. Park, J. Choi, and H.-Y. Kwon, "Learning model combined with data clustering and dimensionality reduction for short-term electricity load forecasting," Sci. Rep., vol. 15, Art. no. 3575, 2025, [Online]. Available: https://doi.org/10.1038/s41598-025-86982-0.

[10] A. Lahouar and J. Ben Hadj Slama, "Day-ahead load forecast using random forest and expert input selection," Energy Convers. Manage., vol. 103, pp. 1040-1051, Oct. 2015, [Online]. Available: https://doi.org/10.1016/j.enconman.2015.07.041.

[11] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5-32, 2001, [Online]. Available: https://doi.org/10.1023/A:1010933404324.

[12] G. Dudek, "A comprehensive study of random forest for short-term load forecasting," Energies, vol. 15, no. 20, Art. no. 7547, 2022, [Online]. Available: https://doi.org/10.3390/en15207547.

[13] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, 2nd ed. New York, NY, USA: Springer, 2009.

[14] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," IEEE Trans. Smart Grid, vol. 10, no. 1, pp. 841-851, Jan. 2019, [Online]. Available: https://doi.org/10.1109/TSG.2017.2753802.