Journal of Engineering Science and Technology
Special Issue on DICAS Vol. 21, No. 1 (2026) 36 - 44

COMPARATIVE ANALYSIS OF LIGHTGBM, RANDOM FOREST, AND XGBOOST WITH HYPERPARAMETER TUNING FOR PREDICTING ELECTRICITY CONSUMPTION

MICHELLE PANDOJO LUKMAN*, CALVIN,
MARCHELLE IMANUEL, MEDIANA ARYUNI

School of Information Systems, BINUS University,
Jl. Raya Kebon Jeruk No. 27, 11530, Jakarta Barat, DKI Jakarta, Indonesia
*Corresponding Author: michellepandojo@gmail.com

Abstract

Electricity consumption forecasting has increasingly leveraged machine learning models, yet most of the current studies focus on large-scale environments rather than small rental housing units with highly variable usage patterns. This research is aimed at comparing the performance of LightGBM, Random Forest, and XGBoost models with and without Hyperparameter Tuning, in predicting electricity consumption in rental houses in Jakarta. This study follows the CRISP-DM framework to predict electricity consumption. In this study, the dataset used was collected through an online questionnaire that targeted rental house occupants in Jakarta. The dataset consists of the following variables, including the electricity consumption of each device and one dependent variable representing the total electricity consumption per day (in kWh). All three baseline models were tuned by using the RandomizedSearchCV technique during the modeling phase. Each model was assessed based on the R^2, MAE, and RMSE metrics to determine which model best predicted the electricity consumption. Results of this study demonstrate that the performance of all three models improves after Hyperparameter Tuning. Among the results obtained, LightGBM with Hyperparameter Tuning achieved a score of 1.4805 RMSE, which is the highest. Findings of this study show the effectiveness of RandomizedSearchCV in improving the performance of electricity consumption prediction models. These findings can provide insight into the determination of an appropriate model for predicting electricity consumption in small-scale residential areas. This study is limited by its reliance on self-reported consumption data from respondents. Its originality lies in applying ensemble-learning models and Hyperparameter Tuning to small-scale residential settings using survey-based household electricity consumption data, an approach that remains largely absent from existing research.

Keywords: Hyperparameter tuning, LightGBM, Random forest, RandomizedCV, Rental house electricity use prediction, XGBoost.

1. Introduction

Electricity is the primary source of energy that has turned into a basic human need. In Indonesia, electricity is produced and distributed by Perusahaan Listrik Negara (PLN) [1]. PLN supplies various varieties of power subscriptions to the public, ranging from 450 VA, 900 VA, 1300 VA, and many others in accordance with the consumer's needs [2]. Most customers nowadays follow the prepaid system, where electricity tokens must be purchased first before electricity is consumed. These tokens can be purchased through PLN Mobile, e-commerce applications, and convenience stores. These adjustments have pointed out the importance of energy management in Indonesia.

In recent times, machine learning algorithms have widely been used as prediction tools in each field, including forecasting electricity consumption. Within the different types of machine learning algorithms, ensemble learning has proven to be effective in various tasks, from combining several base models that together yield a much stronger model. Examples of these include LightGBM, Random Forest, and XGBoost. Random Forest generates a forest of decision trees and sums up the output of each to arrive at a final prediction. In contrast, XGBoost is a tree-boosting method where each new tree tries to correct the mistakes from previous trees. Most researchers use this to handle regression tasks. LightGBM handles regression tasks as well. This algorithm uses Gradient-based One-Side Sampling, which speeds up the training while taking less storage with good performance [3]. Besides, recently, it has been shown that model performance for predicting electricity consumption can significantly be improved by using Hyperparameter Tuning [4]. RandomizedSearchCV is a widely applied technique of Hyperparameter Tuning since its competitive performance with lower computation time allows the near-optimal parameter combinations in a range to be found [5].

Although these studies have brought about remarkable findings, most of these research focus on large-scale environments such as smart grids and industrial sectors [6]. In addition, a very limited number of studies have been devoted to electricity consumption forecasting in rental housing units. As it is forecasted that by 2025, 60% of the population will be living in urban areas, Jakarta as the capital city of Indonesia will witness an increase in electricity demand, mainly coming from informal housing [7]. Due to this fact, rental housing or so-called ‘kost’ has become a sector with significant potential for improved energy efficiency. These accommodations can be occupied either by students, workers, or even families, therefore giving rise to diverse electricity consumption patterns.

Research that compared Random Forest and Decision Tree in predicting household electricity bill indicates that Random Forest has better performance than Decision Tree, with a higher accuracy of 90.05% [8]. Random Forest have also shown optimal MAE and accuracy in short-term daily electricity consumption predictions (1 day ahead) [3].

Additionally, a study that developed an energy consumption forecasting model using LightGBM in a household dataset revealed that it achieved the lowest RMSE compared to other models [9]. LightGBM outperforms other regressor models for very short-term predictions using high-frequency data 10 minutes ahead [3]. Another study revealed that the DWT-XGBoost model performed well in predicting industrial electricity consumption.

Moreover, the different approaches for Hyperparameter Tuning have been used by researchers to improve the model performance. For example, RandomizedSearchCV is recognized as balancing efficient training time and high accuracy. It works by randomly sampling across a predefined parameter range in order to get the best combination of hyperparameters [4]. RandomizedSearchCV also successfully refined model accuracy in the said research [5].

While many researchers have compared different machine learning algorithms to predict electricity consumption, only a few have implemented Hyperparameter Tuning in order to show its effectiveness in improving the performance of a model. In addition, prior research was focused on predicting electricity consumption in industrial environments, without looking into electricity consumption prediction in small-scale areas with limited access to real-time electricity data. This paper addresses these gaps by comparing LightGBM, Random Forest, and XGBoost, with and without Hyperparameter Tuning using RandomizedSearchCV in order to find out which model performed best in predicting electricity consumption for rental houses in Jakarta.

This study compares three different baseline models: LightGBM, Random Forest, and XGBoost in predicting the electricity consumption of rental houses. Hyperparameter Tuning using RandomizedSearchCV was also applied to all baseline models. These predictions will provide insights on how to find the most suitable model for predicting electricity on a small scale. It is expected to help occupants of rental houses in managing their electricity usage more efficiently.

This manuscript is structured as follows: the next section describes the method used in this study. Section III discusses the results and the models used. Section IV summarizes the overall findings of the research.

2. Methods

The method applied for the current study is the CRISP-DM framework, which includes the following steps: Business Understanding, Data Preparation, Modeling, Evaluation, and Deployment, as shown in Fig. 1. CRISP-DM ensures a full elaboration and exploitation of the data, with the aim of producing accurate predictions in electricity consumption. This method has also been adopted in similar research [6].

2.1. Description of the dataset

The dataset was collected through an online questionnaire via Google Forms, targeting 193 respondents who were the occupants of rental houses in Jakarta. The collected data consists of the number of occupants, electricity consumption for each device, and one dependent variable representing total electricity consumption in kWh.

2.2. Dependent variables

Electric Consumption per Day (kWh) was set as the dependent variable. It represents the daily consumption of electricity in total. This is done by adding together the consumption of energy for each electrical device.

2.3. Independent variables

Electricity consumption of each device was obtained by asking the respondents to fill out the wattage and average use (hours) of every appliance. Independent variables of this study are shown in Table 1.

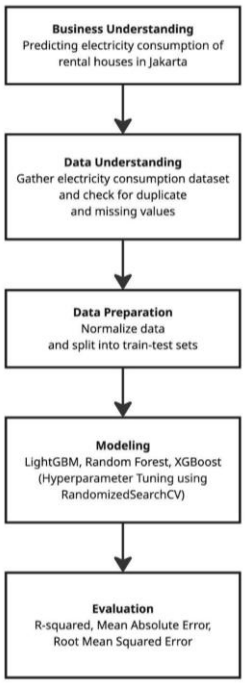

Fig. 1. Flowchart of the methods.

Table 1. Independent variables.

Variable Name	Explanation
Number of Occupants	Number of people who lives in one rental room
Light Bulbs Consumption (kWh)	Electricity consumption from all light bulbs that are attached to a wall
Air Conditioner Consumption (kWh)	Electricity consumption from air conditioner
Refrigerator Consumption (kWh)	Electricity consumption from refrigerator
Electric Stove Consumption (kWh)	Electricity consumption from electric stove
Lamp Consumption (kWh)	Electricity consumption from non-ceiling lighting (lamps)
Laptop Charger Consumption (kWh)	Electricity consumption from laptop charger
Phone Charger Consumption (kWh):	Electricity consumption from phone charger
Fan Consumption (kWh)	Electricity consumption from fan
Magic Jar Consumption (kWh)	Electricity consumption from magic jar
Water Heater Consumption (kWh)	Electricity consumption from water heater

2.4. Data analysis process

This process removes duplicate records, handles null values, encodes categorical variables into numerical formats, and normalizes numerical features. This process ensures that the dataset is clean before moving on to the modeling phase.

2.4.1. Duplicate data

The duplicated() function in Python was used to check any duplicated data in the dataset. Results show that no duplicate data is present in any of the attributes

2.4.2. Null values

The is.null().sum function was run to verify the completeness of the dataset. Results show that there is no missing value across all attributes.

2.4.3. Encode

The number of occupants columns in the dataset was transformed using the Label Encoding technique. This step was done because LightGBM, Random Forest, and XGBoost can only perform mathematical operations when input features are in numeric form.

2.4.4. Normalization

Numerical features were normalized using the Min-Max Scaling technique, an approach that converts the values of each feature to a common scale within the range [0,1][0, 1][0,1], as shown in Eq. (1).

$$X' = \frac{X - Xmin}{Xmax - Xmin} \tag{1}$$

This step is necessary to ensure features with larger numerical ranges do not dominate those with smaller ranges during the training process.

2.4.5. Data splitting

Splitting has been done using the 80/20 ratio through the function train_test_split() with a random state of 42; this ensures that the model has enough data to learn but does not overfit [10]. In previous findings, it was also established that 80/20 is the most successful ratio as compared to other splitting ratios [11].

2.4.6. Modeling

This study compares LightGBM, Random Forest, and XGBoost, with and without Hyperparameter Tuning, in predicting electricity consumption in rental houses. LightGBM implements the leaf-wise tree growth approach to build decision trees and features efficiently for large datasets [12]. Random Forest constructs multiple decision trees using random subsets of the training data and input variables. Each decision tree makes an individual prediction, and these predictions are averaged to produce a final prediction [13]. XGBoost constructs a series of decision trees sequentially, where each new tree tries to correct the mistakes made by the previous trees [14].

2.4.7. Hyperparameter tuning

Hyperparameter Tuning using RandomizedSearchCV was performed in all baseline models. RandomizedSearchCV samples a specified number of parameter combinations. This technique has been proven to enhance the accuracy of a model while maintaining training time [15]. For LightGBM, the tuning process modifies several parameters such as maximum tree depth, learning rate, number of estimators, minimum child samples, subsample, columns used for each tree, reg_alpha, reg_lambda. Random Forest modifies the number of estimators, maximum tree depth, maximum features, minimum samples required for splitting and leaves, and bootstrap as parameters to tune the model. The XGBoost model modifies the number of estimators, learning rate, maximum depth, subsample ratio, column sampling ratio, gamma, reg_alpha, and reg_lambda. RandomizedSearchCV was set to test 100 random parameter combinations with 5-fold cross-validation. The aim is to find the optimal settings that can improve model performance.

Mean Squared Error (RMSE) [16]. R-Squared is known as the proportion of variance in the dependent variable which can be explained by the independent variable. The formula is shown in Eq. (2).

$$R^2 = 1 - \frac{SSres}{SStot} \tag{2}$$

where $SSres = \sum(y_i - \hat{y}_i)^2$, $SStot = \sum(y_i - \bar{y})^2$. y_i represents the actual value, $\hat{y}_i$ represents predicted value, and $\underline{y}$ represents the mean of actual values.

MAE is the average absolute difference between predicted and actual values, obtained by the equation shown in Eq. (3).

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{3}$$

where n is the total number of data. RMSE is the root square of MAE. To compare the performance of each model, a higher R^2 value means a better model fit, while a lower value of MAE and RMSE indicates that the model has a higher accuracy.

3. Results and Discussion

The performance of LightGBM, Random Forest, and XGBoost, with and without Hyperparameter Tuning are assessed based on the R^2, MAE, and RMSE measures. Findings show that all three baseline models performed better after Hyperparameter Tuning. This is proven by previous studies that show the effectiveness of Hyperparameter Tuning in refining model accuracy [5]. Performance of each model is shown in Table 2.

Table 2. Model performance comparison.

Model	R^2	MAE	RMSE
LightGBM	0.7956	1.6466	2.4870
Random Forest	0.8931	1.0907	1.7981
XGBoost	0.8300	1.1223	2.2680
LightGBM and Hyperparameter Tuning	0.9275	0.9953	1.4805
Random Forest and Hyperparameter Tuning	0.8988	0.9855	1.7494
XGBoost and Hyperparameter Tuning	0.8980	1.0781	1.7566

Based on the results above, LightGBM with Hyperparameter Tuning achieved the highest performance compared to other models with a R^2 of 0.9275, MAE of 0.9953, and RMSE of 1.4805. Correspondingly, LightGBM without Hyperparameter Tuning performed the weakest, reflecting its poor performance as a baseline model in predicting rental housing electricity consumption. This agreed with previous studies explaining that this baseline model is more suitable when applied to large datasets [13]. However, the significant improvement after Hyperparameter Tuning shown in Fig. 2. demonstrated the effectiveness of the RandomizedSearchCV method in optimizing its performance for smaller data sets. This behaviour can be explained by the sensitivity of LightGBM to hyperparameter settings such as learning rate, number of leaves, and maximum depth, which directly influence model complexity. Without proper tuning, LightGBM may underfit or overfit the data, particularly in rental housing electricity consumption datasets.

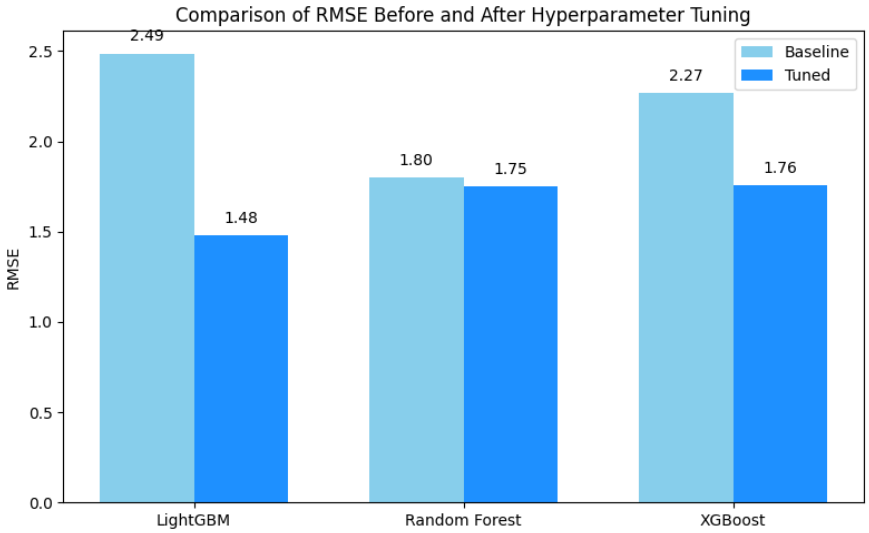

Fig. 2. Comparison of RMSE before and after Hyperparameter tuning.

Moreover, Random Forest exhibits stable performance, with only a slight reduction in RMSE after Hyperparameter Tuning, indicating that it is inherently robust to parameter configuration. This finding is consistent with previous studies describing Random Forest as a robust model with good accuracy [3].

4. Conclusions

The purpose of this study is to compare LightGBM, Random Forest, and XGBoost, with and without Hyperparameter Tuning using RandomizedSearchCV in predicting electricity consumption of rental houses in Jakarta. In this study, the primary dataset was used, which is collected through an online questionnaire distributed to rental house occupants in Jakarta. It follows the CRISP-DM framework to ensure that data is processed before the modeling phase. To find which model performed best in predicting the consumption of electricity, each of these models was evaluated with regard to the R^2, MAE, and RMSE metrics.

The results showed that all three algorithms significantly performed better after Hyperparameter Tuning. LightGBM with Hyperparameter Tuning had the highest score with a R^2 of 0.9275, MAE of 0.9953, and RMSE of 1.4805, while the baseline LightGBM model had the worst performance. This shows how effective RandomizedSearchCV can be in fine-tuning the performance of an ensemble model, even on smaller datasets. Using these results, ensemble models with Hyperparameter Tuning by RandomizedSearchCV can be considered to predict electricity consumption for small-scale residential areas. This study is, however, limited by dependent variables used for training as the data was collected mainly from respondents.

Future research can be done by comparing various techniques of Hyperparameter Tuning in the model of electricity consumption prediction. The use of time series data will also provide more reliable data. With both improvements, better predictions of rental housing electricity consumptions can be expected.

Nomenclatures

n	Total number of data
R^2	Coefficient of Determination
VA	Volt-Ampere
X	Numerical feature
y_i	Actual value
$\hat{y}_i$	Predicted value

Abbreviations

CRISP-DM	Cross-Industry Standard Process for Data Mining
MAE	Mean Absolute Error
RMSE	Root Mean Squared Error

References

1. PT PLN (Persero). (2026). What is PLN?. Retrieved January 26, 2026, from https://web.pln.co.id/en/bki/what-is-pln
2. PT PLN (Persero). (2019). Promo Gebyar Kemerdekaan 2019. Retrieved January 26, 2026, from https://web.pln.co.id/promo-gebyar-kemerdekaan-2019
3. Allal, Z.; Noura, H.N.; Salman, O.; and Chahine, K. (2024). Power consumption prediction in warehouses using variational autoencoders and tree-based regression models. *Energy and Built Environment*.
4. Kavitha, M.; and Revathy, N. (2025). *Predictive modeling of forest cover types using XGBoost and hyperparameter tuning*. In Senjyu, T.; So-In, C.; and Joshi, A. (Eds.), *Smart Trends in Computing and Communications*. Lecture Notes in Networks and Systems, vol 1463. Springer Nature Singapore.
5. Alamsyah, N.; Budiman, B.; Yoga, T.P.; and Alamsyah, R.Y.R. (2024). XGBoost hyperparameter optimization using RandomizedSearchCV for accurate forest fire drought condition prediction. *Jurnal Pilar Nusa Mandiri*, 20(2), 103-110.

6. Herawati, N.A.; Gary, A.A.P.; Hikmawati, E.; and Surendro, K. (2024). A hybrid predictive model as an emission reduction strategy based on power plants' fuel consumption activity. *IEEE Access*, 12, 47119-47133.
7. Wulandri, D.W.; and Mori, S. (2015). Nature and operation of kost private rental housing in urban settlement development of Jakarta, Indonesia. *Journal of Civil Engineering and Architecture*, 9(11), 1362-1369.
8. Harshavardhan, T.V.; and Loganayagi, S. (2025). Comparative analysis of random forest algorithm over decision tree algorithm with improved accuracy in predicting the household electricity bill. *Proceedings of the 3rd International Conference on Engineering and Science to Achieve the Sustainable Development Goals*, Istanbul, Turkey, 20112.
9. Munir, S.; Pradhan, M.R.; Abbas, S.; and Khan, M.A. (2024). Energy consumption prediction based on LightGBM empowered with eXplainable artificial intelligence. *IEEE Access*, 12, 91263-91271.
10. Andrika, M.Y.; and Rahardi, M. (2025). Comparative study of linear regression, SVR, and XGBoost for stock price prediction after a stock split. *Journal of Applied Informatics and Computing*, 9(4), 1817-1824.
11. Haque, A.; Raza, S.; Ahmad, S.; Hossain, A.; Abdeljaber, H.A.M.; Eljialy, A.E.M; Alanazi, S.; and Nazeer, J. (2024). Implication of different data split ratio on the performance of model in price prediction of used vehicles using regression analysis. *Data & Metadata*, 3, 425.
12. Yan, J.; Xu, Y.; Cheng, Q.; Jiang, S.; Wang, Q.; Xiao, Y.; Ma, C.; Yan, J.; and Wang, X. (2021). LightGBM: Accelerated genomically designed crop breeding through ensemble learning. *Genome Biology*, 22(1), 271.
13. Zhou, Z.; Qiu, C.; and Zhang, Y. (2023). A comparative analysis of linear regression, neural networks and random forest eegression for predicting air ozone employing soft sensor models. *Scientific Reports*, 13(1), 22420.
14. Chen, T.; and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, United States, 785-794.
15. Zhao, Y.; Zhang, W.; and Liu, X. (2024). Grid search with a weighted error function: hyper-parameter optimization for financial time series forecasting. *Applied Soft Computing*, 154, 111362.
16. Chai, T.; and Draxler, R.R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? - Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247-1250.