

Comparative Analysis of GridSearchCV and RandomSearchCV for LightGBM-Based Stunting Prediction Using (MICS6) Bangladesh 2019 Data

Fitri Mintarsih¹, Nisrina Nurani Anwar², Andrew Fiade³, Victor Amrizal⁴,
Hendra Bayu Suseno^{5*}

^{1,2,3,4,5} Department of Informatics, Faculty of Science and Technology, Syarif Hidayatullah
State Islamic University Jakarta, Indonesia

ABSTRACT

Open Access

Type: Research Paper

Article:

Received: April 12, 2026

Revised: May 29, 2026

Accepted: August 04, 2026

Issued: October 30, 2026

© Mintarsih et al. (2026).



This is an open-access article
under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license

*Correspondence Address:

hendra.bayu@uinjkt.ac.id

Stunting among children under five is a major public health concern that can impair physical growth and cognitive development. This study evaluates and compares GridSearchCV and RandomSearchCV for hyperparameter tuning of the LightGBM algorithm in predicting stunting using data from the Multiple Indicator Cluster Surveys (MICS6) Bangladesh 2019. The dataset comprises demographic and child health variables, and model performance was evaluated using training-to-testing ratios of 70:30, 80:20, and 90:10. Accuracy, precision, recall, F1-score, and execution time were used as evaluation metrics. The results showed that the 90:10 split yielded the best overall performance. Among the evaluated configurations, LightGBM with RandomSearchCV achieved an accuracy of 98.09%, precision of 95.93%, recall of 97.08%, and an F1-score of 96.50%, with a reported execution time of 0.1 seconds. Compared with the untuned model, hyperparameter tuning provided only marginal improvements in classification performance, while RandomSearchCV offered a favorable balance between predictive performance and computational efficiency. These findings demonstrate the potential of LightGBM-based stunting prediction using (MICS6) Bangladesh 2019 data. However, further evaluation using independent datasets is necessary to assess the model's generalizability and its potential application in early stunting detection.

Keywords: *LightGBM, Hyperparameter Tuning, GridSearchCV, RandomSearchCV, Stunting*

1. INTRODUCTION

Stunting, a condition in which a child's height falls below the average height of other children of the same age based on anthropometric growth measurement standards, has become a significant global health issue [1]. This condition is influenced by various factors such as age, weight and height, maternal education, as well as social and child health conditions, and most commonly occurs during the First 1,000 Days of Life (HPK) [2]. According to UNICEF, stunting is defined as children aged 0–59 months with a Z-score less than minus two standard deviations ($-2SD$) from the median [3].

In Bangladesh, the stunting rate remains high and alarming. Based on the Bangladesh Demographic and Health Survey (BDHS), in 2019, the prevalence of stunting among children under five years was reported at 36.1% [3]. An analysis of 23,099 children in Bangladesh [4] indicated that the risk of stunting is higher among children aged 0–59 months, infants with low birth weight and length, children from families with low socioeconomic status, certain geographic regions, and those born to mothers with low educational attainment. UNICEF, through its global Multiple Indicator Cluster Surveys (MICS6) in Bangladesh 2019, provided a dataset with comprehensive and relevant variables for detecting stunting in children under five years old. The dataset includes individual child health conditions along with diverse variables such as age, gender, maternal education level, and socioeconomic status [5].

The role of Machine Learning (ML) in detecting stunting and malnutrition has become increasingly significant in response to the growing need for innovative solutions to this global health problem [6][7][8]. One ML algorithm that has demonstrated outstanding performance in various health-related classification and prediction studies is the Light Gradient Boosting Machine (LightGBM) [9]. The strength of LightGBM lies in its use of histogram-based decision trees and leaf-wise growth, which enable faster training and lower memory consumption [10]. Although LightGBM has been widely applied across various domains, its use in research related to stunting or malnutrition remains relatively rare, and many existing studies rely on small datasets [11][12].

The performance of LightGBM depends heavily on the combination of hyperparameters used. Default hyperparameters are often suboptimal, and further optimization is required, as each health dataset has unique characteristics such as data size, feature variation, and the complexity of relationships between variables [13][14][15][16]. Several studies have shown that hyperparameter tuning significantly improves the accuracy and precision of LightGBM. For example, GridSearchCV has achieved an accuracy of 97.10% in airline review classification tasks [13]. On the other hand, RandomSearchCV has also proven effective in improving model accuracy with more efficient computation time, as demonstrated in a study with a stroke patient dataset that reported significant improvements in accuracy and precision [15].

GridSearchCV explores all predefined parameter combinations, making it capable of identifying the best configuration, though at the cost of higher computational time [17]. Meanwhile, RandomSearchCV randomly samples parameter combinations, making it more time-efficient while still having the potential to identify competitive configurations [18]. This study applies GridSearchCV and RandomSearchCV to the LightGBM model using 24,686 data records and 15 variables, testing hyperparameter combinations tailored to stunting data in order to optimize performance and compare the effectiveness of both methods.

2. METHODS

The research methodology employed for the analysis of LightGBM algorithm optimization with hyperparameter tuning using GridSearchCV and RandomSearchCV in predicting stunting among children under five years old begins with data collection methods and a quantitative approach to analyze stunting data..

2.1 Machine Learning

Machine learning is a field of study that enables computers to learn without being explicitly programmed while processing data more efficiently. Its applications include prediction, pattern recognition, and recommendation systems [18]. Machine learning has several main categories, such as supervised learning, which predicts specific outcomes using labeled data [19]; unsupervised learning, which analyzes unlabeled data to

discover patterns [18]; and ensemble learning, which combines multiple models to improve accuracy, particularly in imbalanced datasets [20].

Ensemble learning is a technique that combines two or more machine learning algorithms to produce better performance than a single algorithm [21]. One method is boosting, which transforms weak learners into strong classifiers by focusing on errors made in the early iterations [22]. Another method is bagging, which trains multiple random subsets of data and combines the results using majority voting [23].

2.2 LightGBM

LightGBM employs two main techniques, Gradient-Based One-Sided Sampling (GOSS) and Exclusive Feature Bundling (EFB), ensuring faster training and higher accuracy (Mienye & Sun, 2021). GOSS modifies the gradient boosting technique by prioritizing training instances with larger gradients, thereby accelerating learning and reducing model complexity. Meanwhile, EFB bundles sparse and mutually exclusive features, reducing the number of attributes to process. This feature reduction speeds up training while preserving essential information in the LightGBM model [24]

2.3 Hyperparameter Tuning

The process of finding the most suitable hyperparameters is known as hyperparameter tuning. In general, machine learning requires hyperparameter optimization to select the most optimal set for improving model performance and effectiveness [25].

a. GridSearchCV

GridSearchCV is a hyperparameter optimization method that determines the optimal values by exhaustively exploring all points in the search space, forming a “grid” of all parameter combinations. It tries every possible hyperparameter combination in the predefined search space [26]

b. RandomSearchCV

RandomSearchCV is a hyperparameter optimization method that randomly selects parameter combinations from the predefined search space and evaluates model performance for each. It is far more time- and

resource-efficient, especially when dealing with large parameter spaces [27].

2.4 Cross-Validation

Cross-validation is a technique that splits data into multiple subsets (folds) to ensure that the model generalizes well to unseen data [28]. The dataset is divided into several folds, where the model is trained on some folds and tested on the remaining fold. This process is repeated until the final result is the average performance across all folds, thereby reducing bias and variance in model evaluation [29].

2.5 SMOTE (Synthetic Minority Oversampling Technique)

Imbalanced data distribution can affect classification results [30]. Resampling effectively addresses this issue, which can be done through oversampling (adding samples to the minority class) or undersampling (reducing samples from the majority class). SMOTE is a statistical technique that increases the number of minority class samples by generating synthetic instances.

2.6 Evaluation Methods

The confusion matrix is a table used to evaluate the performance of classification models in machine learning. It displays the number of correct and incorrect predictions made by the model by comparing actual labels and predicted labels for each class .

- Accuracy, measures the proportion of correct predictions across all test data ($TP + TN$) divided by the total predictions
- Recall, (sensitivity) focuses on the model's ability to correctly identify positive cases: TP divided by $(TP + FN)$.
- Precision, measures the accuracy of positive predictions: TP divided by $(TP + FP)$.
- F1-Score, harmonizes precision and recall into a single metric, useful when there is a trade-off between the two.

2.7 Data Preprocessing

Before applying the dataset to the algorithm, preprocessing is necessary. Data cleaning removes anomalies, selects correlated features with the target, and transforms dataset values.

- Missing data: handled by ignoring or imputing values using the mean.
- Outliers: identified using the Interquartile Range (IQR), where data points below $Q1 - 1.5 \times IQR$ or above $Q3 + 1.5 \times IQR$ are considered outliers [31].
- Duplicate Data: detected when records have identical values across all columns.
- Normalization: scales numerical values to a uniform range using the formula min-max

2.8 Split Ratio

The dataset was split into training and testing sets using ratios of 70:30, 80:20, and 90:10. As in [32], the 80:20 ratio achieved the best performance with Decision Tree Regression, reaching 100% accuracy, while the 70:30 ratio produced consistent results, and the 90:10 ratio also maintained stable performance.

2.9 Method of Data Collection

A literature study was conducted by reviewing previous research relevant to this context, providing a foundation for designing more appropriate approaches. The review included journals, articles, and e-books related to machine learning, particularly LightGBM optimization. This exploration allowed for a deeper understanding of the issues and potential solutions tested in prior studies, strengthening the theoretical basis and conceptual framework of the research. In this study, the dataset used comes from the Multiple Indicator Cluster Surveys (MICS6) Bangladesh 2019, specifically the "ch" (Child Under Five Years) dataset.

3. RESULTS AND DISCUSSION

After training the model using default parameters without applying hyperparameter tuning, evaluations with GridSearchCV and RandomSearchCV were conducted to assess the model's performance under different train-test split ratios, namely 70:30, 80:20, and 90:10.

3.1. LightGBM without Tuning

The evaluation of the model without hyperparameter tuning across several train-test split scenarios is presented in Table 1.

Table 1. Without Tuning

Ratio	Accuracy	Precision	Recall	F1-Score	Execute time (/s)
80:20	0,9645	0,9454	0,9227	0,9339	0,04
70:30	0,9675	0,9497	0,9302	0,9398	0,06
90:10	0,9795	0,9544	0,9708	0,9625	0,03

The results indicate that the 90:10 split ratio achieved the best performance across all evaluation metrics

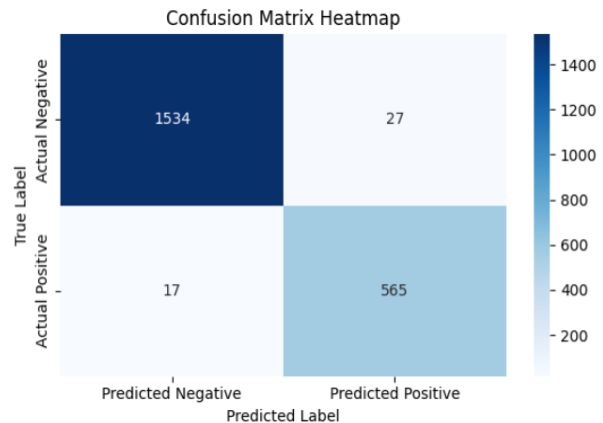


Figure 1. Confusion matrix without tuning (90:10)

The confusion matrix for the 90:10 split is shown in Figure 1. The matrix demonstrates that the model performed very well in classification. A total of 1,534 negative cases were correctly predicted as negative (true negatives), while only 27 negative cases were misclassified as positive (false positives). On the other hand, 565 positive cases were correctly predicted (true positives), and only 17 positive cases were misclassified as negative (false negatives). The high values of true positives and true negatives, combined with the relatively small number of misclassifications, indicate that the model achieved high accuracy, precision, recall, and F1-score. Therefore, the model can be considered reliable in distinguishing between the two target classes with minimal classification error. The results indicate that the 90:10 split ratio achieved the best performance across all evaluation metrics.

3.2. LightGBM with GridSearchCV

The evaluation of the model tuned with GridSearchCV across different train-test split scenarios is presented in Table 2.

Table 2. With GridSearchCV

Hyperparameter	Tuning Range	Hyperparameter Optimal Values		
		Ratio		
		80:20	70:30	90:10
boosting_type	goss	goss	goss	goss
n_estimators	100, 200, 500	500	500	100
learning_rate	0.01, 0.1, 0.2	0,05	0.05	0,1
num_leaves	30,50,100,150	50	50	50
max_depth	3, 5, 10, 15	9	12	12
min_child_sample	10, 30, 50	10	10	10

The results show that the 90:10 ratio consistently yielded the best performance. Each split ratio generated different sets of optimal hyperparameters.

Table 3. Best Parameters GridSearchCV

Rati o	Accuracy	Precisi on	Reca ll	F1- Score	Execut e time (/s)
80:20	0,9645	0,9454	0,9227	0,9339	0,04
70:30	0,9675	0,9497	0,9302	0,9398	0,06
90:10	0,9795	0,9544	0,9708	0,9625	0,03

The confusion matrix with GridSearchCV for the 90:10 split is shown in Figure 2.

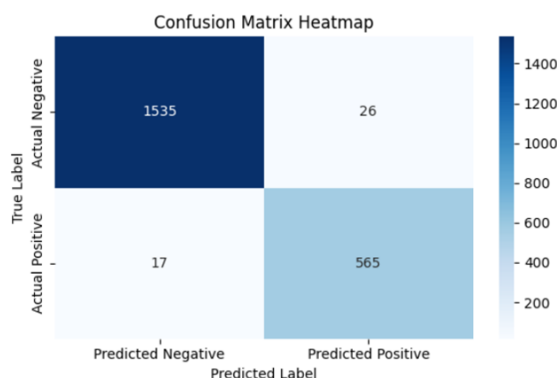


Figure 2. Confusion matrix with GridSearchCV (90:10)

accurate predictions: 1,535 cases were correctly classified as negative (true negatives), and 565 cases were correctly classified as positive (true positives). Meanwhile, 26 negative cases were misclassified as positive (false positives), and 17 positive cases were misclassified as negative (false negatives). The proportion of correct classifications was substantially higher than incorrect ones, suggesting that the model demonstrated strong performance in distinguishing both classes. These results reflect high accuracy, precision, and recall, with a relatively balanced error

distribution, highlighting the model's reliability for binary classification tasks.

3.3. LightGBM with RandomSearchCV

The evaluation results of the LightGBM model optimized using RandomSearchCV across several training-to-testing data split scenarios are presented in Table 4. The table provides a comparative overview of the model's performance under different data partition configurations, allowing the classification results obtained from each scenario to be examined systematically. This evaluation aims to illustrate how variations in the training-to-testing split ratios are associated with differences in the performance of the model after hyperparameter optimization using RandomSearchCV. The results summarized in Table 4 serve as the basis for comparing the model's evaluation performance across the experimental scenarios considered in this study.

Table 4. Best Parameters RandomSearchCV

Hyperparameter	Tuning Range	Hyperparameter Optimal Values		
		Ratio		
		80:20	70:30	90:10
boosting_type	goss	goss	goss	goss
n_estimators	100, 250, 500	500	500	250
learning_rate	0.01, 0.05, 0.1	0,05	0.05	0,05
num_leaves	31, 40, 50	50	50	50
max_depth	3, 9, 12	12	12	9
min_child_sample	10, 30, 50	30	30	10

The optimal hyperparameter values obtained through RandomSearchCV varied across the different training-to-testing data split ratios evaluated in this study. This variation indicates that each data partition may result in a different set of hyperparameter configurations that optimize the model's performance under the corresponding experimental conditions. The following section presents the optimal hyperparameter values identified by RandomSearchCV for each data split ratio, providing a detailed overview of the configurations selected during the optimization process.

Table 5. Best Parameters RandomSearchCV

Ratio	Accur acy	Precisio n	Recal l	F1- Score	Executi on time(/s)
80:20	0,9704	0,9568	0,933 0	0,9448	0,06
70:30	0,9676	0,9508	0,929 0	0,9398	0,9
90:10	0,9809	0,9593	0,970 8	0,9650	0,1

Each data split ratio yielded different hyperparameter values. The following presents the optimal hyperparameter values obtained using GridSearchCV. In addition, the confusion matrices generated using RandomSearchCV for the various training-to-testing data split scenarios are presented below.

The confusion matrix for the 90:10 split is shown in Figure 3

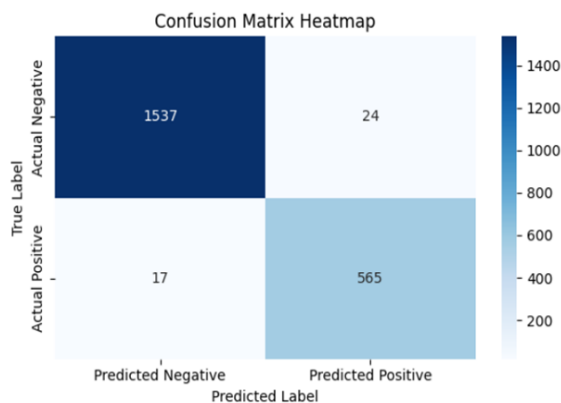


Figure 3. Confusion Matrix with RandomSearchCV (90:10)

The confusion matrix indicates that the model achieved strong classification performance on the evaluated dataset. A total of 1,537 negative cases were correctly classified as negative (true negatives), while 565 positive cases were correctly classified as positive (true positives). These results suggest that the model was able to distinguish between the two classes effectively within the evaluated sample.

The model produced 24 false positives, in which negative cases were incorrectly classified as positive, and 17 false negatives, in which positive cases were incorrectly classified as negative. Compared with the number of correct predictions, these errors were relatively limited. However, their potential implications depend on the context of the classification task, particularly the consequences of incorrectly identifying positive or negative cases.

Overall, the confusion matrix provides evidence of favorable classification performance on the evaluated dataset, with relatively few misclassifications in both classes. Nevertheless, these results alone are insufficient to establish the model's reliability or generalizability to unseen populations. Further evaluation using appropriate performance metrics, repeated data partitioning or cross-validation, and independent validation data is needed to assess the consistency of the model's predictions.

3.4. Best Ratio (LightGBM without Hyperparameter Tuning)

At the 90:10 training-to-testing ratio, the model achieved an accuracy of 0.9795, demonstrating a high level of overall classification performance. The precision value of 0.9544 indicates that approximately 95.44% of all instances predicted as positive were correctly classified, reflecting the model's ability to minimize false-positive predictions. Meanwhile, the recall value of 0.9708 indicates that the model successfully identified 97.08% of all actual positive instances, demonstrating its effectiveness in detecting positive cases. The combination of these precision and recall values resulted in an F1-score of 0.9625, indicating a strong balance between the model's ability to make accurate positive predictions and its capacity to identify actual positive instances.

Beyond its strong classification performance, the model configured with the 90:10 ratio also demonstrated superior computational efficiency in terms of prediction execution time. Specifically, the model required only 0.03 seconds to complete the prediction process, outperforming the configurations with 80:20 and 70:30 ratios, which required 0.04 and 0.06 seconds, respectively. These results indicate that the 90:10 ratio provided the shortest execution time among the three evaluated configurations, suggesting that this data-splitting strategy enabled the model to deliver predictions more efficiently under the experimental conditions. Furthermore, the relatively short execution time indicates that the model can perform the prediction process with minimal computational delay.

Overall, the experimental results demonstrate that the 90:10 training-to-testing ratio provided the best combination of classification performance and computational efficiency under the no-tuning scenario. The model achieved an accuracy of 97.95%, a precision of 95.44%, a recall of 97.08%, and an F1-score of 96.25%, while requiring only 0.03 seconds for prediction. These findings suggest that the 90:10 ratio was the most effective configuration among the evaluated data-splitting strategies for the experimental setup considered in this study. Nevertheless, the selection of this configuration as the optimal approach should be interpreted within the scope of the experimental design, as further evaluation using repeated experiments or cross-validation may be necessary to establish the robustness and generalizability of the observed performance.

3.5. Best Ratio (LightGBM with GridSearchCV)

The evaluation conducted using GridSearchCV with training-to-testing data split ratios of 80:20, 70:30, and 90:10 demonstrated that the 90:10 ratio achieved the best overall classification performance. Under this configuration, the model attained an accuracy of 97.99%, a precision of 95.69%, a recall of 97.08%, and an F1-score of 96.33%. These results indicate that the model exhibited a high level of predictive accuracy and maintained a strong balance between precision and recall, reflecting its ability to generate reliable positive predictions while effectively identifying actual positive instances. Furthermore, the model required only 0.5 seconds to complete the execution process, representing the shortest execution time among the three evaluated configurations.

In comparison, the model configured with an 80:20 training-to-testing ratio achieved an accuracy of 96.80%, a precision of 94.69%, a recall of 93.47%, and an F1-score of 94.08%. Meanwhile, the 70:30 ratio produced an accuracy of 96.76%, a precision of 95.08%, a recall of 92.90%, and an F1-score of 93.98%. Although both configurations demonstrated relatively high classification performance, their accuracy, recall, and F1-score values were lower than those obtained using the 90:10 ratio. These differences suggest that the proportion of

data allocated for training may have influenced the model's ability to learn relevant patterns and generalize its predictions to unseen data. Nevertheless, the comparatively higher precision achieved by the 70:30 configuration indicates that its positive predictions were proportionally more accurate than those of the 80:20 configuration.

Overall, the experimental findings indicate that the 90:10 training-to-testing ratio provided the most favorable results among the evaluated configurations when GridSearchCV was applied. The combination of the highest accuracy, precision, recall, and F1-score, together with the shortest execution time of 0.5 seconds, suggests that this configuration offered the most effective balance between predictive performance and computational efficiency in the conducted experiments. Although the 80:20 and 70:30 ratios provided larger testing sets, their lower performance on several key evaluation metrics indicates that allocating a greater proportion of data for training may have benefited the model under the conditions examined in this study. However, further validation using repeated experiments or cross-validation would be necessary to determine whether the observed advantage of the 90:10 ratio remains consistent across different data partitions.

3.6. Best Ratio (LightGBM with RandomSearchCV)

The LightGBM model evaluated using RandomSearchCV with three training-to-testing data split ratios (80:20, 70:30, and 90:10) demonstrated the best overall performance at the 90:10 ratio. Under this configuration, the model achieved an accuracy of 98.09%, a precision of 95.93%, a recall of 97.08%, and an F1-score of 96.50%, indicating a high level of classification performance and a strong balance between precision and recall. In addition to its superior predictive performance, the model required only 0.1 seconds of training time, demonstrating its computational efficiency under the experimental conditions. These results suggest that the model was able to learn relevant patterns from the training data and generate accurate predictions while maintaining a favorable balance between false-positive and false-negative classifications.

In comparison, the model configured with an 80:20 training-to-testing ratio achieved an accuracy of 97.04% and an F1-score of 94.48%, with an execution time of 0.06 seconds. Although this configuration demonstrated relatively high classification performance and a shorter execution time than the 90:10 configuration, its accuracy and F1-score remained lower. Meanwhile, the model evaluated using a 70:30 ratio achieved an accuracy of 96.76% and an F1-score of 93.98%, with an execution time of 0.9 seconds. The results indicate a slight decline in classification performance under this configuration, accompanied by a longer execution time. These differences suggest that the choice of training-to-testing data split ratio may influence both the predictive performance and computational efficiency of the LightGBM model when optimized using RandomSearchCV.

Overall, the experimental findings indicate that the 90:10 training-to-testing ratio provided the most favorable combination of classification performance and training efficiency among the three evaluated configurations. The highest accuracy of 98.09%, precision of 95.93%, recall of 97.08%, and F1-score of 96.50% demonstrate the model's strong classification capability under this configuration. These findings suggest that allocating a larger proportion of the available data for training may improve the model's ability to identify relevant patterns and enhance its predictive performance. Nevertheless, because the 90:10 ratio leaves a relatively small proportion of data for testing, its superior results should be interpreted cautiously. Further validation through repeated data partitioning or cross-validation is recommended to assess the stability and generalizability of the model's performance.

3.7. Comparison Result

The training and evaluation processes were conducted comprehensively to ensure optimal performance. Through these experiments, it became evident that each tuning method had different impacts on the quality of the model's predictions. Evaluations were performed using various train-test split ratios, with the best results consistently obtained at the 90:10 ratio. This demonstrates that allocating a larger proportion of data for training provides the model with greater opportunity to learn from

sufficient information, leading to optimal evaluation performance. In other words, the 90:10 ratio strikes an effective balance between sufficient training data and representative test data, enabling each tuning method to achieve its best performance under this configuration.

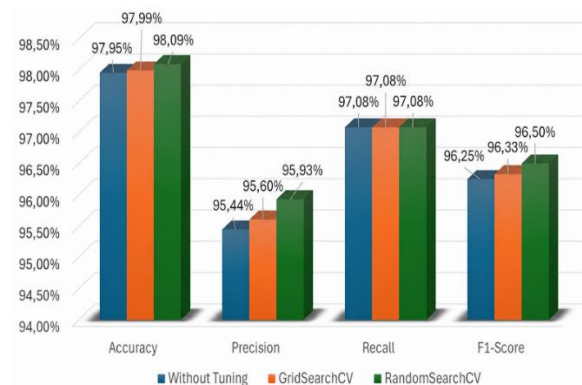


Figure 4. Comparison Result (90:10)

Figure 4 presents a comparison of evaluation results. The LightGBM model evaluated using three configurations—default (without tuning), GridSearchCV, and RandomSearchCV—under the 90:10 split showed excellent performance across key metrics. The evaluation results indicate:

- Accuracy: RandomSearchCV achieved the highest at 98.09%, demonstrating strong overall predictive capability.
- Precision: RandomSearchCV produced the best result at 95.60%, indicating fewer false positives.
- Recall: All three methods achieved the same recall value of 97.08%, showing consistency in detecting most positive cases.
- F1-score: RandomSearchCV achieved the highest at 96.50%, representing the best balance between precision and recall.
- Execution Time: RandomSearchCV required 0.1 seconds, which was faster than GridSearchCV (0.5 seconds) but slightly slower than the untuned model (0.03 seconds).

Although hyperparameter tuning with GridSearchCV and RandomSearchCV resulted in performance improvements, the gains were relatively small.

- The accuracy of the untuned model already reached 97.95%, increasing only

slightly to 97.99% and 98.09% after tuning—an improvement of just 0.04% to 0.14%.

- b. This phenomenon suggests that when data is already clean, balanced, and the chosen algorithm has strong baseline performance, tuning does not necessarily yield substantial performance gains. This finding aligns with [33], where LightGBM without tuning achieved 97.07% accuracy after SMOTE+RUS preprocessing, and tuning improved it only marginally to 97.11%.
- c. Thus, tuning primarily contributes to efficiency and stability in parameter search rather than significantly enhancing performance metrics.

3.8. Comparison with Previous Studies

This study compares the findings obtained by the authors with those reported in the previous study conducted by Intan et al. (2024), focusing on the application of hyperparameter optimization techniques for stunting prediction. Both studies employed GridSearchCV as the hyperparameter optimization method and used a dataset comprising 10,000 records. The use of the same optimization technique and an equivalent dataset size provides a basis for examining the methodological similarities between the two studies. However, despite these similarities, the studies differed in the machine learning algorithms used to develop the predictive models. The present study applied the LightGBM algorithm, whereas Intan et al. (2024) employed the Multilayer Perceptron (MLP) algorithm.

These methodological differences provide an opportunity to examine the performance of different machine learning algorithms when applied to stunting prediction under the respective experimental settings. Although both studies utilized GridSearchCV to identify suitable hyperparameter configurations, the characteristics of LightGBM and MLP may influence their respective classification performance. Therefore, comparing the findings of the present study with those of Intan et al. (2024) can provide additional context for understanding the application of different

machine learning algorithms to stunting prediction. Nevertheless, any conclusions regarding the relative performance of the two algorithms should be drawn cautiously, taking into account the evaluation metrics, experimental configurations, and validation procedures reported in each study.

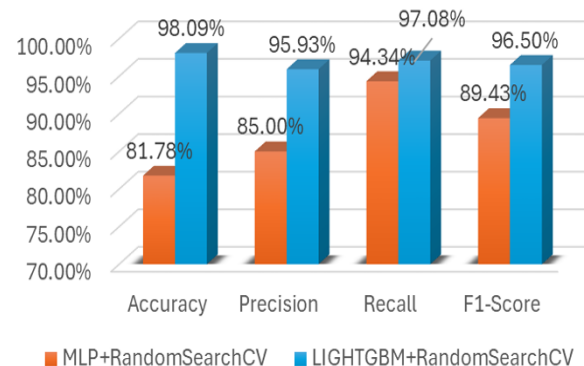


Figure 5. Comparison with Previous Studies

From the evaluation results it can be seen that:

- a. LightGBM accuracy reached 98.09%, significantly higher than MLP's 81.78%. This indicates that LightGBM is capable of classifying data more correctly across almost all samples.
- b. In the precision metric, which measures the proportion of correct positive predictions, LightGBM also outperformed MLP with a score of 95.93%, compared to MLP's 85.00%. This indicates that LightGBM is more effective at avoiding false positives.
- c. Recall, which indicates the model's ability to capture all true positive data, was also better for LightGBM (97.08%) than MLP (94.34%), indicating that LightGBM was more accurate in detecting all positive cases without many missed false negatives.
- d. The F1-score, a combined metric that balances precision and recall, confirms the superiority of LightGBM (96.50%) over MLP (89.43%), indicating balanced and consistent performance.

CONCLUSION

This study compared the performance of LightGBM without hyperparameter tuning, LightGBM optimized using GridSearchCV, and

LightGBM optimized using RandomSearchCV for stunting prediction among children under five years of age using the (MICS6) Bangladesh 2019 dataset. The experimental results showed that the 90:10 training-to-testing ratio achieved the best overall classification performance among the evaluated data-splitting configurations. Under this setting, LightGBM with RandomSearchCV obtained an accuracy of 98.09%, precision of 95.93%, recall of 97.08%, and an F1-score of 96.50%. These results indicate that RandomSearchCV delivered the best predictive performance among the evaluated configurations, particularly in balancing precision and recall.

The comparison also revealed that hyperparameter tuning yielded only marginal improvements over the untuned LightGBM model, which had already achieved an accuracy of 97.95% and an F1-score of 96.25%. This finding suggests that the effectiveness of hyperparameter optimization depends on the baseline model performance and the characteristics of the dataset. Although RandomSearchCV achieved the highest accuracy and F1-score in this study, the observed differences should be interpreted within the experimental setting and do not establish the universal superiority of this optimization method over GridSearchCV.

Despite the promising results, several limitations should be acknowledged. The evaluation was conducted using a single dataset and a limited number of training-to-testing ratios. Furthermore, the 90:10 split allocated only 10% of the data for testing, which may limit the representativeness of the evaluation and the assessment of model generalizability. Future research should employ repeated stratified experiments or cross-validation, evaluate the model on independent datasets from different populations, and examine the effects of class distribution and relevant predictive variables on model performance. Such investigations would provide stronger evidence of the model's robustness and applicability.

In conclusion, LightGBM optimized using GridSearchCV demonstrated promising performance for stunting classification using the (MICS6) Bangladesh 2019 dataset. However, the model should be regarded as a potential analytical tool rather than a validated clinical or public health screening system. Further external validation, model interpretability analysis, and assessment in relevant public health settings are

necessary before considering its practical application in early stunting risk identification.

DECLARATIONS

Author Contributions: The first and second authors contributed equally to conceptualization, methodology, data curation, software implementation, formal analysis, and writing the original draft. The third, fourth, and fifth authors contributed equally to validation, investigation, data visualization, interpretation of the results, supervision, and writing review and editing. All authors reviewed and approved the final version of the manuscript.

Funding: This research received no external funding.

Conflict of Interest: The authors declare no conflict of interest.

Generative Artificial Intelligence Use Statement: No Generative AI or Large Language Model was used to generate the scientific content, research data, analysis, or interpretation of the findings presented in this manuscript. The authors take full responsibility for the originality, accuracy, integrity, citations, and conclusions of the research.

Ethics Approval and Informed Consent: Not applicable. This study used secondary data from the Multiple Indicator Cluster Surveys (MICS6) Bangladesh 2019 dataset and did not involve direct recruitment or interaction with human participants.

Data Availability: The data used in this study were obtained from the Multiple Indicator Cluster Surveys (MICS6) Bangladesh 2019 dataset. The dataset is available through the official UNICEF MICS website, subject to the applicable data access conditions and terms of use.

REFERENCES

- [1] Yuliasri, N. (2024). From Womb to World: Immunological Pathways Influencing Stunting and 2030 Sustainability. RSF Conference Series: Business, Management and Social Sciences. <https://doi.org/10.31098/bmss.v4i1.886..>
- [2] Kawo, M. A., Idris, M. N., Haruna, K., & Kaita, F. U. (2024). Predicting

- Undernutrition Risk Factors Using Machine Learning Techniques in Nigerian Under Five Children. *International Journal of Computer Science and Mobile Computing*, 13(7), 56–70.
<https://doi.org/10.47760/ijcsmc.2024.v13i07.006..>
- [3] Chowdhury, T. R., Chakrabarty, S., Rakib, M., Winn, S., & Bennie, J. (2022). Risk factors for child stunting in Bangladesh: an analysis using MICS 2019 data. *Archives of Public Health*, 80. <https://doi.org/10.1186/s13690-022-00870-x>.
- [4] Ekka Pujo Ariesanto Akhmad. (2025). Enhancing the Accuracy of Airline Review Classification Using SMOTE and Grid Search with Cross Validation for Hyperparameter Tuning. *Journal of Information Systems Engineering and Management*, 10(25s), 850–861. <https://doi.org/10.52783/jisem.v10i25s.4174>.
- [5] Hartanto, A. D., Kholik, Y. N., & Pristyanto, Y. (2023). Stock Price Time Series Data Forecasting Using the Light Gradient Boosting Machine (LightGBM) Model. *International Journal on Informatics Visualization*, 7(4), 2270–2279. <https://doi.org/10.30630/joiv.7.4.1740>.
- [6] Intan, P. A., Ardhia, I. C., & Sri, C. K. A. (2024). Klasifikasi Penyakit Stunting Menggunakan Algoritma Multi-Layer Perceptron. *Journal MIND Journal | ISSN*, 9(1), 52–63. <https://doi.org/10.26760/mindjournal.v9i1.52-63>.
- [7] Massari, H. El, Gherabi, N., Qanouni, F., & Mhammedi, S. (2024). Diabetes Prediction Using Machine Learning with Feature Engineering and Hyperparameter Tuning. 15(8), 171–179.
- [8] Ndagijimana, S., Kabano, I., Masabo, E., & Ntaganda, J. (2023). Prediction of Stunting Among Under-5 Children in Rwanda Using Machine Learning Techniques. *Journal of Preventive Medicine and Public Health*, 56, 41–49. <https://doi.org/10.3961/jpmph.22.388>.
- [9] Saarela, M., & Jauhiainen, S. (2021). Comparison of feature importance measures as explanations for classification models. *SN Applied Sciences*, 3. <https://doi.org/10.1007/s42452-021-04148-9>.
- [10] Shao, H. Z., Liu, X., Zong, D. S., & Song, Q. J. (2024). Optimization of diabetes prediction methods based on combinatorial balancing algorithm. *Nutrition and Diabetes*, 14(1), 1–13. <https://doi.org/10.1038/s41387-024-00324-z>, doi: 10.12928/TELKOMNIKA.v12i4.537.
- [11] Zhang, Y., Wang, C., Xia, Q., Jiang, W., Zhang, H., Amiri-Ardekani, E., Hua, H., & Cheng, Y. (2023). Machine learning-based prediction of candidate gene biomarkers correlated with immune infiltration in patients with idiopathic pulmonary fibrosis. *Frontiers in Medicine*, 10(February). <https://doi.org/10.3389/fmed.2023.1001813>.
- [12] Khan, S., & Hancioglu, A. (2019). Multiple Indicator Cluster Surveys: Delivering Robust Data on Children and Women across the Globe. *Studies in Family Planning*, 50(3), 279–286. <https://doi.org/10.1111/sifp.12103>.
- [13] Ji, Z., & Zhang, Y. (2022). Predictive Analysis of Cryptocurrency Residual Log-returns Based on LightGBM. *Academic Journal of Business & Management*, 4(4), 96–100. <https://doi.org/10.25236/ajbm.2022.040419>.
- [14] Hastuti, N. T., & Budiman, F. (2024). Optimasi Klasifikasi Stunting Balita dengan Teknik Boosting pada Decision Tree. *Edumatic: Jurnal Pendidikan Informatika*, 8(2), 655–664. <https://doi.org/10.29408/edumatic.v8i2.27913>.
- [15] Gujjar, J., & Kumar, V. (2021). Google Colaboratory : Tool for Deep Learning and Machine Learning Applications. *International Journal in Computer Simulation*, 6, 23–26. <https://doi.org/10.17010/IJCS/2021/V6/I3-4/165408>.

- [16] Fajri, M., & Primajaya, A. (2023). Komparasi Teknik Hyperparameter Optimization pada SVM untuk Permasalahan Klasifikasi dengan Menggunakan Grid Search dan Random Search. *Journal of Applied Informatics and Computing*, 7(1), 14–19. <https://doi.org/10.30871/jaic.v7i1.5004>.
- [17] Y. S. Deraman, I. Ahmad, A. F. Ramli, and N. A. M. Mortar, “Optimization performance of unmanned aerial vehicle in wireless sensor network,” *AIP Conf. Proc.*, vol. 2291, no. November, 2020, doi: 10.1063/5.0025092.
- [18] Bradshaw, T. J., Huemann, Z., Hu, J., & Rahmim, A. (2023). A Guide to Cross-Validation for Artificial Intelligence in Medical Imaging. *Radiology: Artificial Intelligence*, 5(4). <https://doi.org/10.1148/ryai.220232>.
- [19] Amin. (2022). Confusion Matrix in Binary Classification Problems: A Step-by-Step Tutorial. In *Journal of Engineering Research*. <https://doi.org/10.21608/erjeng.2022.274526>.
- [20] Bairamova, D. (2024). Predicting Course Grades of Students Academic Performance Using The LightGBM Regressor. *Suleyman Demirel University Bulletin Natural and Technical Sciences*. <https://doi.org/10.47344/sdubnts.v62i1.952>.
- [21] Ahamad, G. N., Shafiullah, Fatima, H., Imdadullah, Zakariya, S. M., Abbas, M., Alqahtani, M. S., & Usman, M. (2023). Influence of Optimal Hyperparameters on the Performance of Machine Learning Algorithms for Predicting Heart Disease. *Processes*, 11(3). <https://doi.org/10.3390/pr11030734>.
- [22] Delgado-Panadero, Á., García-Ordás, M. T., & Benítez-Andrades, J. (2022). Implementing local-explainability in Gradient Boosting Trees: Feature Contribution. *ArXiv*, abs/2402.0. <https://doi.org/10.1016/j.ins.2021.12.111>.
- [23] Khan, S., & Hancioglu, A. (2019). Multiple Indicator Cluster Surveys: Delivering Robust Data on Children and Women across the Globe. *Studies in Family Planning*, 50(3), 279–286. <https://doi.org/10.1111/sifp.12103>.
- [24] Mohammed, A., & Kora, R. (2022). An effective ensemble deep learning framework for text classification. *Journal of King Saud University - Computer and Information Sciences*, 34(10), 8825–8837. <https://doi.org/10.1016/j.jksuci.2021.11.001>
- [25] Nardkulpat, A., Boonpook, W., Sitthi, A., & Tan, Y. (2025). Machine learning-based and synthetic aperture radar time-series data for rice classification over Sentinel-1 imagery. *Bulletin of Electrical Engineering and Informatics*, 14(2), 1138–1150. <https://doi.org/10.11591/eei.v14i2.7833>
- [26] Piegorsch, W. (2020). *Confusion Matrix*. Wiley StatsRef: Statistics Reference Online. <https://doi.org/10.1002/9781118445112.stat08244>.
- [27] Ravichandran, T., Gavahi, K., Ponnambalam, K., Burtea, V., & Mousavi, S. J. (2021). leak detection in water mains. 307–323. <https://doi.org/10.2166/hydro.2021.093>.
- [28] Ravichandran, T., Gavahi, K., Ponnambalam, K., Burtea, V., & Mousavi, S. J. (2021). leak detection in water mains. 307–323. <https://doi.org/10.2166/hydro.2021.093>
- [29] Yan, J., Xu, Y., Cheng, Q., Jiang, S., Wang, Q., Xiao, Y., Ma, C., Yan, J., & Wang, X. (2021). LightGBM: accelerated genomically designed crop breeding through ensemble learning. *Genome Biology*, 22(1), 1–24. <https://doi.org/10.1186/s13059-021-02492-y>.
- [30] Noor, M., Syarwani, M., Putri, R. M., Kalimantan, I., & Arsyad, M. (2024). Healthy Environment As A Human Right: Law Enforcement To Overcome Stunting In Children. *At-Tadbir : Jurnal Ilmiah Manajemen*. <https://doi.org/10.31602/piuk.v0i0.15389>,
- [31] Deng, L., & Xiao, M. (2024). Hyperparameter recommendation via

- automated meta-feature selection embedded with kernel group Lasso learning. *Knowl. Based Syst.*, 306, 112706.
<https://doi.org/10.1016/j.knosys.2024.112706>.
- [32] Hapsari, G. I., Munadi, R., Erfianto, B., & Irawati, I. D. (2025). Feature Selection Using Pearson Correlation for Ultra-Wideband Ranging Classification. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*.
<https://doi.org/10.29207/resti.v9i2.6281>.
- [33] Shao, H. Z., Liu, X., Zong, D. S., & Song, Q. J. (2024). Optimization of diabetes prediction methods based on combinatorial balancing algorithm. *Nutrition and Diabetes*, 14(1), 1–13.
<https://doi.org/10.1038/s41387-024-00324-z>.
- [34] S. M. J. Rahman, N. A. M. F. Ahmed, M. M. Abedin, B. Ahammed, M. Ali, M. J. Rahman, and M. Maniruzzaman, “Investigate the risk factors of stunting, wasting, and underweight among under-five Bangladeshi children and its prediction based on machine learning approach,” *PLOS ONE*, vol. 16, no. 6, Art. no. e0253172, 2021.
<https://doi.org/10.1371/journal.pone.0253172>.
- [35] AF. H. Bitew, C. S. Sparks, and S. H. Nyarko, “Machine learning algorithms for predicting undernutrition among under-five children in Ethiopia,” *Public Health Nutrition*, vol. 25, no. 2, pp. 269–280, 2022.
<https://doi.org/10.1017/S1368980021004262>.
- [36] O. N. Chilyabanyama, R. Chilengi, M. Simuyandi, C. C. Chisenga, M. Chirwa, K. Hamusonde, R. K. Saroj, N. T. Iqbal, I. Ngaruye, and S. Bosomprah, “Performance of machine learning classifiers in classifying stunting among under-five children in Zambia,” *Children*, vol. 9, no. 7, Art. no. 1082, 2022.
<https://doi.org/10.3390/children9071082>.
- [37] H. Shen, H. Zhao, and Y. Jiang, “Machine learning algorithms for predicting stunting among under-five children in Papua New Guinea,” *Children*, vol. 10, no. 10, Art. no. 1638, 2023.
<https://doi.org/10.3390/children10101638>.
- [38] E. Indrisari, H. Febiansyah, and B. Adiwinoto, “A systematic literature review on the application of machine learning for predicting stunting prevalence in Indonesia (2020–2024),” *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 14, no. 3, pp. 277–283, Jul. 2025.
<https://doi.org/10.32736/sisfokom.v14i3.2366>.
- [39] E. Ramadanti, D. A. Dinathi, C. S. K. Aditya, and D. R. Chandranegara, “Diabetes Disease Detection Classification Using Light Gradient Boosting (LightGBM) With Hyperparameter Tuning,” *Sinkron: Jurnal dan Penelitian Teknik Informatika*, vol. 8, no. 2, 2024.
<https://doi.org/10.33395/sinkron.v8i2.13530>.