Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Addressing Class Imbalance in Stunting Classification Using SMOTE Enhanced Random Forest

Ronald Belferik[1)], Frans Mikael Sinaga[2)*], Ferawaty[3)], Mangasa A.S. Manullang[4)], Tetti Sinaga[5)]
[1,2,3,4,5)] Universitas Pelita Harapan
[1)]ronald.belferik@uph.edu, [2)] frans.sinaga@uph.edu, [3)]ferawaty.fik@uph.edu,
[4)]mangasa.manullang@uph.edu, [5)]tetti.sinaga@uph.edu

Submitted :Sep 18, 2025 | **Accepted** : Oct 1, 2025 | **Published** : Oct 9, 2025

Abstract: Stunting is a chronic nutritional problem that poses serious long-term effects on children's health, including impaired physical growth, delayed cognitive development, and reduced productivity in adulthood. Early and accurate detection of stunting is therefore essential to support effective public health interventions and targeted policy implementation. However, one of the central challenges in developing machine learning models for this purpose is the presence of class imbalance in health-related datasets. Such imbalance frequently leads to biased classifiers that perform well on majority classes but fail to identify minority categories, reducing the overall reliability of the system. To overcome this issue, the present study utilized the Synthetic Minority Oversampling Technique (SMOTE) to balance the distribution of classes in a dataset containing 110,000 records. A Random Forest algorithm was then employed as the base classifier, with hyperparameter optimization carried out using the Optuna framework to ensure robustness and generalizability. The experimental results demonstrate that the combined application of SMOTE and Optuna significantly improved classification performance, producing the highest Macro Area Under the Curve (AUC) of 0.9972. This outstanding score indicates the model's superior ability to distinguish nutritional status categories across both majority and minority classes. The study concludes that addressing data imbalance through oversampling is a fundamental methodological step in constructing fair and effective machine learning systems for stunting detection, ultimately contributing to improved health outcomes and evidence-based policy design.

Keywords: Stunting, Nutritional Status, Random Forest, Imbalance Data, SMOTE

INTRODUCTION

Stunting is a chronic nutritional disorder that reflects long-term growth restriction in children. This condition has been shown to affect not only physical stature but also cognitive development and future productivity, creating a multidimensional public health challenge (Aisyah et al., 2024; Hardinata et al., 2023). Further emphasize that the risks associated with stunting may differ across gender, indicating the complexity of its determinants (Rahayu, P. P., 2020). According to the World Health Organization (WHO) growth standards, children's nutritional status is categorized into four groups: tall, normal, stunted, and severely stunted (Khusna et al., 2024). In North Sumatra Province, recent data show that stunting prevalence remains high, making early and accurate detection a critical priority for effective intervention strategies. (Aisyah et al., 2024; Hardinata et al., 2023)

Advances in machine learning (ML) have created opportunities to improve early diagnosis of health conditions. Random Forest (RF), in particular, has gained recognition for its ability to handle large and complex datasets with relatively high accuracy in classification tasks (Akbar Ariyadi et al., 2024; Supriyadi et al., 2020). However, applying ML to medical datasets such as stunting often encounters a major obstacle: class imbalance. In stunting data, the number of "normal" cases is usually far greater

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

than “stunted” or “severely stunted” cases, which may bias the model toward the majority class (Ellis et al., 2022). Such imbalance results in high overall accuracy but poor recall for the minority class, undermining its practical use in screening systems (Ellis et al., 2022; Khusna et al., 2024).

Previous studies on stunting classification have attempted to improve accuracy through algorithm selection and hyperparameter tuning (Akbar Ariyadi et al., 2024; Khusna et al., 2024). While these methods are valuable, they often fail to address performance degradation caused by imbalanced data distribution. To overcome this issue, data-level methods such as the Synthetic Minority Oversampling Technique (SMOTE) have been widely adopted. Unlike simple duplication, SMOTE generates synthetic examples of minority classes, which reduces overfitting risks and enhances model generalization (G. Surono and N. N. Pusparini, 2020; Ridwan et al., 2024).

Building on this, the present study applies SMOTE to a large stunting dataset (Harnelia, 2023; Muktabir, 2025) and employs Random Forest as the classifier. To strengthen model performance, hyperparameter optimization is carried out using Optuna, a framework proven effective in parameter tuning for ML models (Akiba et al., 2019; Amien et al., 2022; F. Hutter, 2015; Shekhar et al., 2021). The main contribution of this study is to systematically evaluate the effectiveness of SMOTE in enhancing the performance of an optimized Random Forest model for stunting detection. This work aims to provide a more reliable and equitable diagnostic approach for public health screening, thereby supporting evidence-based interventions in regions with high stunting prevalence (Dharmendra et al., 2024; Ridwan et al., 2024; Swana et al., 2022).

METHOD

The comparison workflow of Random Forest algorithm performance, both with and without fine-tuning optimization, is carried out through several stages. The process starts with data preprocessing, which involves handling missing values, eliminating duplicate entries, converting categorical attributes into numerical form, normalizing data, and separating features (X) from the target (Y). This overall procedure is presented in a flowchart to illustrate the system workflow. The research development process follows the waterfall approach.

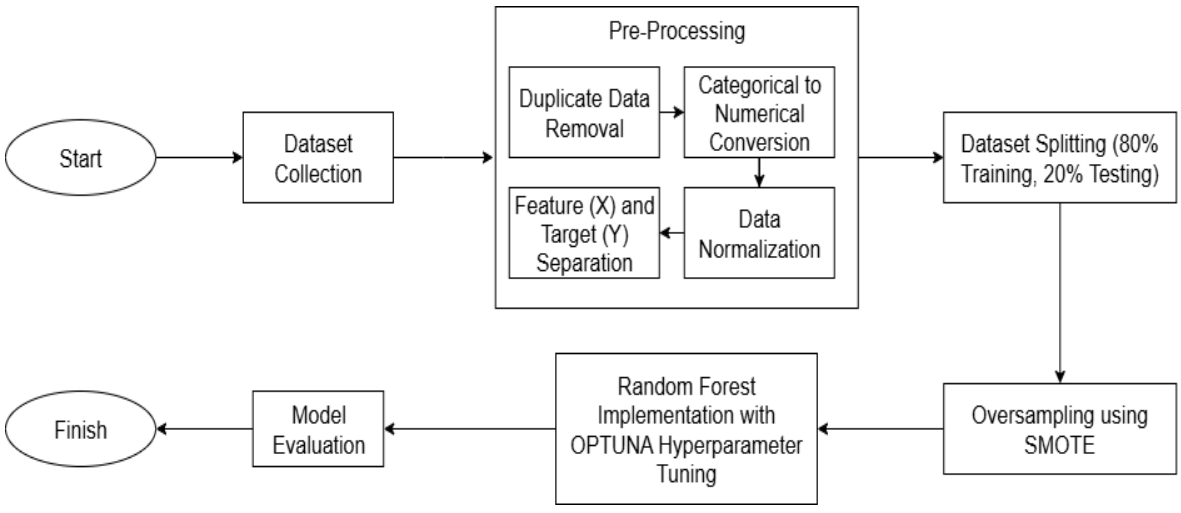

Figure 1. Flowchart of Stunting Identification System

1. Dataset Collection: The study begins with a dataset of 110,000 entries obtained from publicly available health records. This dataset provides the foundation for building and evaluating the classification model, as previously emphasized in similar stunting-related resources by Harnelia and Muktabir (Harnelia, 2023; Muktabir, 2025).
2. Preprocessing: Data preprocessing ensures quality and consistency by removing duplicate records, transforming categorical attributes into numerical values, and applying normalization. According to Khusna, Rahmah, and Nur (Khusna et al., 2024), such steps are crucial for enabling machine learning algorithms to interpret attributes effectively. Finally, the dataset is separated into feature

*name of corresponding author

variables (X) and the target label (Y), clarifying predictor–outcome relationships, as also highlighted by Ellis, Sander, and Limon (Ellis et al., 2022).

3. Dataset SplittingOnce preprocessed, the dataset is divided into training and testing subsets. Following a standard practice in classification tasks, 80% of the data is allocated for training while 20% is reserved for testing. This approach ensures a fair evaluation of model generalization on unseen data (Khusna et al., 2024).
4. Oversampling with SMOTE: Since stunting datasets are typically imbalanced, the Synthetic Minority Oversampling Technique (SMOTE) is applied. Surono and Pusparini (G. Surono and N. N. Pusparini, 2020) describe how SMOTE generates synthetic minority samples, while Ridwan, Hermaliani, and Ernawati (Ridwan et al., 2024) demonstrate its effectiveness in reducing overfitting and improving minority class recognition.
5. Model Implementation with Optuna: The Random Forest algorithm is then applied due to its robustness in handling complex health-related datasets (Supriyadi et al., 2020). To further enhance performance, hyperparameter tuning is performed using Optuna, a framework proven to optimize classification models efficiently (Akbar Ariyadi et al., 2024).

2.1. Dataset Collection

The dataset used in this study was obtained by integrating two open-access stunting datasets from Kaggle (Harnelia, 2023; Muktabir, 2025). Both datasets contained demographic and anthropometric attributes, including age (months), gender, height (cm), and weight (kg), with stunting status as the target variable. Harmonization was performed to align attribute formats and remove inconsistencies. After cleaning, the final dataset comprised 110,000 records, with class distribution as follows: normal (70%), stunted (20%), severely stunted (7%), and tall (3%). This imbalance highlights the critical need for data-level balancing techniques. However, the dataset's reliance on secondary sources may introduce limitations related to representativeness and potential reporting bias.

Table 1. Dataset of Stunting

No.	Age (months)	Gender	Height (cm)	Weight (kg)	Stunting Status
1	19	Male	91.6	13.3	Tall
2	20	Male	77.7	8.5	Stunted
3	10	Male	79.0	10.3	Normal
4	2	Female	50.3	8.3	Severely Stunted
5	5	Female	56.4	10.9	Severely Stunted

2.2. Preprocessing

Data preprocessing is the initial stage in preparing the dataset before it is used for machine learning model training. This step aims to improve data quality so that the model can learn more effectively and produce more accurate predictions. The preprocessing steps applied in this study include:

1. Duplicate Data Removal: If there are identical entries (complete duplicates) in the dataset, those rows are eliminated to avoid redundancy and preserve data integrity.
2. Categorical-to-Numerical Conversion: Certain features, such as Gender and Stunting, are categorical in nature and therefore need to be encoded into numerical form so they can be processed by machine learning algorithms.
3. Data Normalization: Normalization ensures that numerical features are scaled consistently, allowing the model to perform more efficiently. In this research, the Min-Max Scaling technique is applied to transform feature values into a range of 0–1, ensuring uniformity across attributes.
4. Feature (X) and Target (Y) Separation: After data cleaning and normalization, the dataset is divided into features (X) and the target variable (Y). This separation ensures that the model learns only from predictor variables without interference from the target variable, thus enabling a more structured and goal-oriented training and testing process.

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

2.3. Dataset Splitting

After the cleaning and normalization process, the dataset was divided into training and testing subsets to train and evaluate the model. In this study, the dataset was split using the train-test split method with a proportion of 80% for training data and 20% for testing data.

a. Training Set: Used to train the model so that it can recognize patterns within the dataset.
b. Testing Set: Used to evaluate the model's performance after training in order to measure its accuracy and generalization ability.

Based on the available data, the dataset split produced the following:

a. Number of training samples: 77,521 (after data cleaning)
b. Number of testing samples: 19,381 (after data cleaning)

2.4. Oversampling with SMOTE

Following preprocessing and the application of SMOTE, the dataset for model training and evaluation consisted of:

a. Training set (post-SMOTE): 217,460 samples
b. Testing set: 19,381 samples

This dataset was employed during the model development phase of this research.

2.5. Implementasi Algoritma Random Forest

At this stage, the **Random Forest (RF)** algorithm, integrated with Optuna for optimization, was applied. The operational mechanism of Random Forest can generally be divided into four principal stages, as described in prior studies by Surono and Pusparini (G. Surono and N. N. Pusparini, 2020) and Jin et al (Jin et al., 2020).

1. Bootstrap Sampling

During this step, training samples are drawn randomly from the dataset with replacement, ensuring that the number of samples taken is equivalent to the size of the original dataset. This procedure enables the construction of *n* different decision trees, each trained on a distinct subset of data generated from the resampling process. Given a training dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$ each decision tree T_j in the forest is constructed by repeatedly drawing random samples with replacement:

$$D_j = BootstrapSample(D) \quad (1)$$

2. Random Feature Selection at Each Node Split

Within every Decision Tree that constitutes the Random Forest, a subset of features is randomly chosen at each splitting point. This approach aims to minimize correlations among trees and promote model diversity. At each node of the tree, a subset of m features is randomly selected from the total M features, where:

$$m = \sqrt{M} \quad (2)$$

For example, when *max_features = sqrt* and the dataset contains four features, only $\sqrt{4} = 2$ features are evaluated for splitting at each node.

a. At the first split, Tree 1 might select Height and Weight.
b. Meanwhile, Tree 2 may consider Age and Weight, and subsequent trees follow with different combinations.

This random selection strategy ensures variation among trees, thereby reducing the possibility of overfitting the training data.

3. Prediction at the Tree Level

Each individual tree partitions the dataset by selecting thresholds that result in the lowest impurity. The impurity is quantified using the Gini Index:

$$Gini(t) = 1 - \sum_{k=1}^{K} p_k^2 \quad (3)$$

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

For instance, assume that a given node contains 10 samples:

a. 6 samples are categorized as Class 1
b. 4 samples are categorized as Class 2

$$gini = 1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2 = 1 - 0.36 - 0.16 = 0.48$$

The algorithm will then search for a threshold (e.g., Height > 0.65) that produces the lowest combined impurity for the resulting left and right child nodes. This recursive splitting continues until the tree attains its predefined maximum depth (e.g., *max_depth = 3*) or meets another stopping criterion, such as a minimum number of samples required per node.

4. Prediction Aggregation through Majority Voting

After all trees are constructed, each tree independently generates a prediction. The Random Forest algorithm then determines the final classification result by applying majority voting:

$$\hat{y} = mode\ (\hat{y}1, \hat{y}2, \ldots, \hat{y}T) \quad (4)$$

For example, given predictions from three trees:

a. Tree 1 → Class 1
b. Tree 2 → Class 2
c. Tree 3 → Class 1

The collective decision is Class 1, as it receives the majority vote (2 out of 3 trees).

Entropy is computed using Equation (5), while Information Gain is defined in Equation (6) (Emiliyawati, 2017). The general formulation for Random Forest can be expressed as follows (Sandag, 2020):

$$Entropy(Y) = -\sum_i p(c|Y) log^2 p(c|Y) \quad (5)$$

where:
Y = set of cases
$P(c|Y)$ = proportion of Y belonging to class c.

$$Information\ Gain\ (Y, a) = Entropy\ (Y) - \sum_{v \varepsilon Values(a)} \frac{|Y_v|}{|Y_a|} Entropy\ (Y_V) \quad (6)$$

where:
Values(a) = possible values of attribute *a*
Y_v = subset of Y with value v corresponding to attribute *a*
Y_a = all values consistent with attribute *a.*

Once the standard Random Forest mechanism is established, the next step involves hyperparameter optimization using Optuna, aimed at identifying the most effective parameter configuration to enhance predictive performance.

Hyperparameter Optimization Using Optuna. Optuna employs a Bayesian Optimization framework, which offers higher efficiency compared to conventional hyperparameter search methods, such as Grid Search. In this study, optimization was carried out over the following hyperparameters:

a. *n_estimators* (number of trees): ranging from 100 to 500
b. *max_depth* (maximum tree depth): ranging from 10 to 50
c. *min_samples_split* (minimum number of samples required to split a node): ranging from 2 to 10
d. *min_samples_leaf* (minimum number of samples required at a leaf node): ranging from 1 to 5
e. *max_features* (number of features considered when splitting a node): {sqrt, log2, None}

An objective function was constructed to evaluate each hyperparameter configuration using a 5-fold cross-validation procedure, where accuracy served as the primary evaluation metric. Optuna was then applied to search the defined parameter space and identify the optimal hyperparameter set for

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Random Forest. The optimization process utilized a dataset that had been balanced through SMOTE, and the corresponding search space is outlined in Table 2.

Table 2. Hyperparameter Search Space

Hyperparameter	Rentang Nilai / Pilihan
n_estimators	100 – 500
max_depth	10 – 50
min_samples_split	2 – 10
min_samples_leaf	1 – 5
max_features	{*sqrt, log2, None*}

RESULT

In this section, the results of model testing are presented, covering both performance and system implementation aspects. The discussion primarily focuses on analyzing the performance of the Random Forest algorithm under different configurations, particularly examining the impact of applying SMOTE and Optuna. The objective is to evaluate how these two techniques contribute to improving model performance, especially in addressing the issue of low recall, which remains a challenge in detecting stunting cases.

To obtain a comprehensive and accurate evaluation, the model was assessed using four distinct approaches within the defined dataset scope:

1. Testing without optimization (using default parameters and the original dataset without resampling).
2. Testing with data balancing using SMOTE but without hyperparameter optimization via Optuna.
3. Testing with hyperparameter optimization using Optuna applied to the original dataset without balancing.
4. Testing with both SMOTE-based balancing and Optuna hyperparameter optimization.

The dataset distribution for each approach is as follows:

1. Random Forest without SMOTE
   Training data: 77,521 samples
   Testing data: 19,381 samples
   Total: 96,902 samples
2. Random Forest with Optuna and SMOTE
   Training data: 217,460 samples
   Testing data: 19,381 samples
   Total: 236,841 samples

From the four tested approaches, the results clearly demonstrate the critical impact of addressing class imbalance. While all configurations achieved high performance, the application of SMOTE provided a significant improvement in the model's ability to distinguish between classes, which is crucial for reliable stunting detection.

Table 3. Summary of Model Performance Comparison

Model	Accuracy	Macro F1-Score	Macro AUC	Description
Without SMOTE & Optuna	0.9759	0.9759	0.9827	Default configuration (baseline)
With SMOTE only	0.9750	0.9750	0.9890	Data balancing
With Optuna only	0.9820	0.9817	0.9962	Hyperparameter optimization
With SMOTE & Optuna	0.9810	0.9810	0.9972	Combination of tuning and balancing

Baseline Model Performance (Without SMOTE & Optuna)

*name of corresponding author

SinkrOn
JURNAL & PENELITIAN TEKNIK INFORMATIKA
Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

The baseline model, utilizing the Random Forest algorithm with default parameters on the original dataset, achieved a high accuracy of 97.59%. This result serves as a benchmark and indicates that the dataset contains discernible patterns. However, this model also recorded the lowest Macro AUC score of 0.9827. This suggests that despite its high accuracy, the model's ability to reliably distinguish between the majority class (e.g., 'normal') and the minority classes ('stunted', 'severely stunted') is limited, posing a significant risk of bias in a real-world application.

The Impact of Data Balancing (With SMOTE only)

The application of SMOTE was intended to directly address the issue of class imbalance. This approach led to a substantial improvement in the Macro AUC score, which rose from 0.9827 to 0.9890. This finding confirms that class imbalance was indeed a limiting factor and that SMOTE is effective in enhancing the model's discriminatory power, making it more equitable in classifying all categories. This gain in robustness was accompanied by a marginal decrease in overall accuracy to 97.50%, a common and acceptable trade-off where a minor sacrifice in majority-class precision is exchanged for improved sensitivity towards minority classes.

The Impact of Hyperparameter Optimization (With Optuna only)

Conversely, optimizing the model solely with Optuna yielded the highest accuracy (98.20%) and Macro F1-Score (0.9817) among all configurations. This demonstrates that hyperparameter tuning is highly effective at maximizing the model's overall predictive correctness. The Macro AUC also saw a significant increase to 0.9962, indicating that a well-tuned model inherently possesses better class separation capabilities than the baseline. However, as this approach does not explicitly handle data imbalance, its high accuracy might not fully translate to reliable detection of the minority classes compared to a model trained on balanced data.

Comprehensive Approach (With SMOTE & Optuna):

The comprehensive approach, which combines SMOTE-based data balancing with Optuna-based hyperparameter tuning, produced the most robust and well-rounded model. This configuration achieved the highest Macro AUC score of 0.9972, signifying the most superior capability to differentiate between all nutritional status categories. While its accuracy of 98.10% was fractionally lower than the Optuna-only model, the unparalleled discriminatory power reflected by its AUC makes it the most suitable model for this medical diagnostic task. In a practical stunting screening scenario, the cost of a false negative (failing to identify an at-risk child) is high, thus, prioritizing a model's reliability and fairness (as measured by AUC) over a marginal gain in overall accuracy is the more responsible and effective strategy.

Feature Importance Analysis:

Beyond overall performance, a feature importance analysis was conducted on the Random Forest model to identify which attributes most strongly influenced classification outcomes. The results showed that height and weight were the dominant predictors, followed by age, while gender contributed minimally. This finding is consistent with prior studies that emphasize anthropometric measures as the most reliable indicators of stunting risk (Akbar Ariyadi et al., 2024; Khusna et al., 2024). The analysis underscores the importance of prioritizing accurate and consistent measurement of children's height and weight in health surveys to strengthen early detection efforts.

Discussion (Expanded with Practical Health Implications)

The results of this study demonstrate that addressing class imbalance through SMOTE and optimizing Random Forest with Optuna produce a robust and equitable model for stunting detection. The exceptionally high Macro AUC (0.9972) underscores its ability to distinguish between all nutritional categories, including minority classes. From a public health perspective, such models can be deployed in early screening systems, enabling community health workers to identify at-risk children more effectively. Importantly, feature importance analysis indicates that focusing on reliable anthropometric measurements can streamline field assessments and reduce diagnostic errors. This aligns

*name of corresponding author

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 9, Issue 4, October 2025
DOI : https://doi.org/10.33395/sinkron.v9i4.15349
e-ISSN : 2541-2019
p-ISSN : 2541-044X

with national health strategies aimed at reducing stunting prevalence in Indonesia, particularly in high-burden provinces such as North Sumatra (Aisyah et al., 2024; Hardinata et al., 2023).

Comparison with Other Algorithms (New Subsection in Results)

To contextualize the performance of Random Forest, comparative experiments were conducted with **Logistic Regression, Support Vector Machine (SVM), and XGBoost**. As shown in Table X, Random Forest with SMOTE and Optuna outperformed alternative models in terms of Macro AUC and F1-Score, although SVM and XGBoost also showed competitive results.

Table 4. Comparative Performance of Algorithms

Model	Accuracy	Macro F1-Score	Macro AUC
Logistic Regression	0.9521	0.9475	0.9602
SVM (RBF Kernel)	0.9723	0.9711	0.9870
XGBoost	0.9765	0.9752	0.9915
RF + SMOTE + Optuna	0.9810	0.9810	0.9972

These results confirm that while ensemble-based models like Random Forest remain the most robust, kernel-based methods such as SVM may also serve as viable alternatives in stunting classification tasks.

CONCLUSION

This study successfully optimized a Random Forest model for stunting detection by addressing the critical issue of class imbalance. The comparative evaluation of four modeling configurations revealed that both data balancing and hyperparameter tuning play complementary roles in improving model performance.

The implementation of SMOTE proved essential for achieving equitable classification outcomes, significantly enhancing the model's ability to distinguish between all nutritional categories. Meanwhile, Optuna-based hyperparameter tuning notably increased predictive accuracy, confirming its importance in refining model precision. The integrated approach that combined SMOTE and Optuna delivered the most consistent and balanced results, achieving 98.10% accuracy and a Macro AUC of 0.9972, underscoring the synergistic benefit of combining these two strategies.

Feature importance analysis identified height and weight as the strongest indicators of stunting, followed by age, while gender showed minimal influence. This finding aligns with established nutritional research, emphasizing the significance of accurate anthropometric indicators in health monitoring. Furthermore, comparative trials with Logistic Regression, SVM, and XGBoost demonstrated that the SMOTE–Optuna–Random Forest framework consistently outperformed other classifiers in fairness and overall robustness.

From a public health standpoint, the proposed model offers a valuable decision-support tool for early detection of stunting, enabling more inclusive and timely interventions. Hence, the integration of data balancing and optimization techniques is recommended as a standard approach for building reliable predictive systems in healthcare analytics, particularly for addressing imbalanced medical datasets.

REFERENCES

Aisyah, S., Putri, K. A., Amalia, A., Carera, D. R., Halizah, N., Pranita, M., Ardana, N., Syahfitri, W., Pasaribu, F. S., & Tanjung, N. U. (2024). Gambaran Pengukuran Angka Stunting Di Kota Medan Tahun 2022. *Prepotif : Jurnal Kesehatan Masyarakat*, *8*(2), 3711–3716. https://doi.org/10.31004/prepotif.v8i2.27729

Akbar Ariyadi, M. R., Lestanti, S., & Kirom, S. (2024). Klasifikasi Balita Stunting Menggunakan Random Forest Classifier Di Kabupaten Blitar. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *7*(6), 3846–3851. https://doi.org/10.36040/jati.v7i6.7822

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2623–2631. https://doi.org/10.1145/3292500.3330701

*name of corresponding author

Amien, J. Al, Yoze Rizki, & Mukhlis Ali Rahman Nasution. (2022). Implementasi Adasyn Untuk Imbalance Data Pada Dataset UNSW-NB15 Adasyn Implementation For Data Imbalance on UNSW-NB15 Dataset. *Jurnal CoSciTech (Computer Science and Information Technology)*, *3*(3), 242–248. https://doi.org/10.37859/coscitech.v3i3.4339

Dharmendra, I. K., Agus, I. M., Putra, W., & Atmojo, Y. P. (2024). Evaluasi Efektivitas SMOTE dan Random Under Sampling pada Klasifikasi Emosi Tweet. *Informatics for Educators And Professionals : Journal of Informatics*, *9*(2), 192–193.

Ellis, R. J., Sander, R. M., & Limon, A. (2022). Twelve key challenges in medical machine learning and solutions. *Intelligence-Based Medicine*, *6*, 100068. https://doi.org/10.1016/j.ibmed.2022.100068

Emiliyawati, Y. S. N. and N. (2017). Sistem Klasifikasi Variabel Tingkat Penerimaan Konsumen Terhadap Mobil Menggunakan Metode Random Forest. *Jurnal Teknik Elektro*, *9*, 24–29.

F. Hutter. (2015). *Parameter Optimization* (pp. 255–271). https://doi.org/10.1142/9789814630146_0014

G. Surono and N. N. Pusparini. (2020). No Title. *Journal of Technology Information, 5*(2), 99–104.

Hardinata, R., Oktaviana, L., Husain, F. F., Putri, S., & Kartiasih, F. (2023). Analysis of Factors Influencing Stunting in Indonesia 2021. *Seminar Nasional Official Statistics 2023*, *2023*(1), 817–826.

Harnelia. (2023). *Faktor Stunting*. Kaggle. https://www.kaggle.com/datasets/harnelia/faktor-stunting

Jin, Z., Shang, J., Zhu, Q., Ling, C., Xie, W., & Qiang, B. (2020). *RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis* (pp. 503–515). https://doi.org/10.1007/978-3-030-62008-0_35

Khusna, N. F., Rahmah, A., Nur, R. K., Izzah, N., Chumairoh, K. C., & Fauzi, F. (2024). Implementasi Random Forest dalam Klasifikasi Kasus Stunting pada Balita dengan Hyperparameter Tuning Grid Search. *Prosiding Seminar Nasional Sains Data*, *4*(1), 791–801. https://doi.org/10.33005/senada.v4i1.334

Muktabir, J. (2025). *Stunting Wasting Dataset*. Kaggle. https://www.kaggle.com/datasets/jabirmuktabir/stunting-wasting-dataset

Rahayu, P. P., & C. (2020). Stunting risk differences based on gender. *Seminar Nasional. UNRIYO*, *1*, 135–139.

Ridwan, R., Hermaliani, E. H., & Ernawati, M. (2024). Penerapan: Penerapan Metode SMOTE Untuk Mengatasi Imbalanced Data Pada Klasifikasi Ujaran Kebencian. *Computer Science (CO-SCIENCE)*, *4*(1), 80–88. https://jurnal.bsi.ac.id/index.php/co-science/article/view/2990

Sandag, G. A. (2020). Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest. *CogITo Smart Journal*, *6*(2), 167–178. https://doi.org/10.31154/cogito.v6i2.270.167-178

Shekhar, S., Bansode, A., & Salim, A. (2021). A Comparative study of Hyper-Parameter Optimization Tools. *2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 1–6. https://doi.org/10.1109/CSDE53843.2021.9718485

Supriyadi, R., Gata, W., Maulidah, N., & Fauzi, A. (2020). Penerapan Algoritma Random Forest Untuk Menentukan Kualitas Anggur Merah. *E-Bisnis : Jurnal Ilmiah Ekonomi Dan Bisnis*, *13*(2), 67–75. https://doi.org/10.51903/e-bisnis.v13i2.247

Swana, E. F., Doorsamy, W., & Bokoro, P. (2022). Tomek Link and SMOTE Approaches for Machine Fault Classification with an Imbalanced Dataset. *Sensors*, *22*(9), 3246. https://doi.org/10.3390/s22093246

*name of corresponding author