Klasifikasi Stunting pada Balita dari Bandarharjo Menggunakan Algoritma K-Nearest Neighbors dan Random Forest

Tugas Akhir

diajukan untuk memenuhi salah satu syarat

memperoleh gelar sarjana

dari Program Studi Informatika

Fakultas Informatika

Universitas Telkom

1301213269

Ratin Kani

Program Studi Sarjana Informatika

Fakultas Informatika

Universitas Telkom

Bandung

2024

LEMBAR PENGESAHAN

Klasifikasi Stunting pada Balita dari Bandarharjo Menggunakan Algoritma K-Nearest Neighbors dan Random Forest

Classification of Stunting in Toddlers from Bandarharjo Using K-Nearest Neighbors and Random Forest Algorithms

NIM: 1301213269

Ratin Kani

Tugas akhir ini telah diterima dan disahkan untuk memenuhi sebagian syarat memperoleh gelar pada Program Studi Sarjana Informatika

Fakultas Informatika

Universitas Telkom

Bandung, 3 Desember 2024

Menyetujui

Pembimbing I

Dr. Putu Harry Gunawan, S.Si., M.Si., M.Sc.

NIP: 16860043

Ketua Program Studi

Sarjana Informatika,

Dr. Erwin Budi Setiawan, S.Si., M.T.

NIP: 00760045

LEMBAR PERNYATAAN

Dengan ini saya, Ratin Kani, menyatakan sesungguhnya bahwa Tugas Akhir saya dengan judul "**Klasifikasi Stunting pada Balita dari Bandarharjo Menggunakan Algoritma K-Nearest Neighbors dan Random Forest**" beserta dengan seluruh isinya adalah merupakan hasil karya sendiri, dan saya tidak melakukan penjiplakan yang tidak sesuai dengan etika keilmuan yang belaku dalam masyarakat keilmuan. Saya siap menanggung resiko/sanksi yang diberikan jika dikemudian hari ditemukan pelanggaran terhadap etika keilmuan dalam buku TA atau jika ada klaim dari pihak lain terhadap keaslian karya.

Bandung, 3 Desember 2024

Yang Menyatakan,

Ratin Kani

Classification of Stunting in Toddlers from Bandarharjo Using K-Nearest Neighbors and Random Forest Algorithms

Ratin Kani
School of Computing
Telkom University
Bandung, Indonesia
dhafanur@student.telkomuniversity.ac.id

Putu Harry Gunawan
School of Computing
Telkom University
Bandung, Indonesia
phgunawan@telkomuniversity.ac.id

***Abstract*—Stunting is a major health priority for children in Indonesia, with a prevalence of 21.6% in 2022, prompting the government to aim for a reduction to 14% by 2024 in accordance with WHO standards. Stunting, caused by prolonged malnutrition, affects children's physical and cognitive development. Machine learning is needed to develop predictive models that can identify stunting early, enabling timely interventions. This study aims to develop a machine learning-based predictive model for early intervention in stunting. It evaluates the performance of Random Forest (RF), K-Nearest Neighbors (KNN), and a proposed ensemble learning algorithm called Stacked RFKNN. Using a dataset from Bandarharjo Health Center that includes toddler measurements and growth indicators for training and testing, RF excels at handling large feature sets and identifying important predictors, KNN effectively recognizes local patterns, and Stacked RFKNN combines the strengths of both. Given the dataset's imbalance, with only 4.1% of data related to stunting, oversampling was applied to the minority class. Results show the proposed Stacked RFKNN algorithm outperformed both RF and KNN, achieving a classification accuracy of 99.21% and an F1-score of 95.42%. In comparison, RF achieved an accuracy of 99.12% and an F1-score of 95.05%, while KNN attained an accuracy of 97.19% and an F1-score of 86.14%. This approach offers an accurate predictive system for early stunting intervention, aiding the reduction of stunting rates in Indonesia.**

***Keywords*—ensemble learning, k-nearest neighbors, machine learning, random forest, stunting**

I. Introduction

Stunting is a significant public health issue in developing countries, including Indonesia, where the prevalence was 21.6% in 2022 according to the Asian Development Bank [3]. This condition impairs children's growth and development due to chronic malnutrition [13] and has long-term cognitive, health, and economic impacts [20]. Reducing stunting is a national priority, with the Indonesian government aiming to lower its prevalence significantly by 2024 [12]. Advanced predictive models are essential for early detection and timely intervention to achieve this goal.

Machine learning techniques have demonstrated significant potential in predicting health outcomes, such as stunting. Studies have shown that Random Forest (RF) outperforms Naïve Bayes (NB) and Logistic Regression in predicting stunting among toddlers in Bojongsoang, highlighting RF's robustness and widespread applicability [9], [10]. K-Nearest Neighbors (KNN) has also demonstrated strong performance in stunting prediction, with Sibuea *et al.* [19] reporting an accuracy of 98.00%, supported by Khansa and Gunawan [14], who found KNN consistently outperforms NB. Pebrianti *et al.* [16] further validated KNN's reliability, achieving an accuracy of 92.00% with data from Bojongemas Village. These results emphasize both RF and KNN's ability to capture local data patterns, which is particularly useful in cases where localized health data is crucial. Additionally, Daffa and Gunawan [6] utilized a boosted version of the KNN algorithm to classify stunting status in toddlers, achieving an accuracy of 97.94%, illustrating the effectiveness of ensemble methods in enhancing model performance.

Research on RFKNN, an ensemble learning method combining KNN and RF, has demonstrated promising results in predictive tasks [15]. Berliana *et al.* [4] proposed a stacking classifier utilizing both RF and KNN as base learners, successfully leveraging their diverse strengths to enhance the overall performance of the ensemble model. The effectiveness of stacking is further highlighted by Rahim *et al.* [17], who applied a stacked ensemble method for type-2 diabetes prediction, showcasing the method's versatility across various domains. These studies underscore the ability of stacking classifiers to outperform individual models, providing a strong basis for its potential application in predicting stunting in toddlers [8].

This study aims to develop an efficient and effective ML model to detect stunting in toddlers using a dataset from the Bandarharjo Health Center in Central Java, consisting of over 3,700 records of toddlers' physical measurements. Bandarharjo was chosen for its significant stunting challenges and less developed status compared to cities like Jakarta, offering a unique view on stunting in different socio-economic conditions. Building on previous research, this study will primarily use RF and KNN, and will also propose a Stacked RF-KNN ensemble method to determine the most effective model for early stunting detection. By providing timely alerts through ML models, this research aims to contribute to stunting prevention in Indonesia, potentially mitigating this critical health issue.

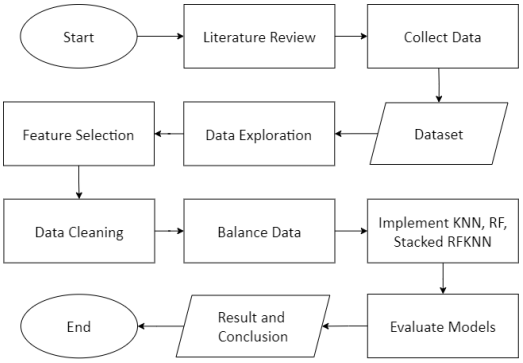

Fig. 1. Flowchart of Research

II. METHODS

A. Research Design

The research follows a systematic approach to classify stunting in toddlers using machine learning. Initially, a literature review is conducted to understand stunting and identify effective classification methodologies. Data on toddler measurements is collected from the Bandarharjo Health Center and analyzed to select relevant features, which are then visualized for clarity. During the preprocessing stage, the data is cleaned and normalized to enhance the accuracy, reliability, and efficiency of the subsequent analysis or modeling process. Furthermore, the dataset is divided into training and test sets, with the training set being oversampled to address the imbalance in stunted cases. Following this, the KNN, RF, and Stacked RFKNN models are implemented and trained using the preprocessed dataset. The performance of each model is evaluated and compared to determine the most effective approach for stunting classification. A flowchart of the research process is provided in Fig. 1.

B. K-Nearest Neighbors

KNN is an algorithm used for data classification and regression, leveraging the proximity of data points for classification [1], [7]. For a given query instance q, the algorithm identifies the K nearest instances and determines the class by majority vote. The distance between new data and training data is calculated using the Euclidean distance formula:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{1}$$

where x is a data sample, y is the test data, n is the number of training data points, and i is a data variable. This method classifies data points within a multidimensional space by finding the closest neighbors. Algorithm 1 presents a simple implementation of the KNN algorithm.

Algorithm 1 K-Nearest Neighbors

Input: X: Training data, Y: Class labels of X, K: Number of nearest neighbors.
Output: Predicted class of test data x.

Start
Classification (X, Y, x)
1. *for* each x test data point *do*
   Calculate the Euclidean distance between x and each point in X.
   end for
2. Classify each test data point x into the majority class of its nearest neighbors.

End

C. Random Forest

RF is a versatile machine learning algorithm that addresses classification problems by creating an ensemble of decision trees. RF constructs multiple decision trees from random subsets of data and features, with each tree making an independent prediction of the class [18]. The final prediction is then determined by majority voting across all the trees. Algorithm 2 presents a simple implementation of the RF algorithm [11].

Algorithm 2 Random Forest

Input: D: training data, c: number of decision trees.
Output: *Random Forest* classification.

To generate c classifications.
for i = 1 to c ***do***
  Randomly select training data D with replacement to create D_i.
  Generate a root node, N_i, containing D_i.
  Call BuildTree(N_i).
end for

procedure BuildTree(N)
  if N contains only one class ***then***
    return
  else
    Randomly select x% of the available split features in N.
    Choose feature F with the highest information gain for splitting.
    Generate child nodes from N, N_1, ..., N_f, where F has a number of possible values F_1, ..., F_f.
    for i = 1 to f ***do***
      Update the contents of N to become D_i, where D_i is all occurrences in N that match F_i.
      Call BuildTree(N_i).
    end for
  end if
end procedure

D. Stacked RFKNN

Stacked RFKNN uses ensemble learning to boost prediction accuracy by combining multiple models [17]. In this approach, the ensemble utilizes two base models: RF, which handles complex feature interactions, and KNN, which focuses on local patterns, with each model being trained separately on the data. The predictions from these base learners are then used as inputs for a meta-learner, which is another RF model. This meta-RF synthesizes the base learners' outputs to provide a final prediction. Stacking improves predictive performance by integrating the strengths of RF and KNN while mitigating their individual weaknesses [2]. The decision to limit the ensemble to these two models strikes a balance between enhancing performance and avoiding the added complexity, longer

load times, and potential overfitting that could arise from incorporating more base learners. This ensures the predictions are both effective and efficient.

E. Evaluation Metrics

In assessing the performance of algorithms for classifying stunting in toddlers, the metrics used are accuracy, precision, recall, and F1-score. These metrics are computed as follows:

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN}, \tag{2}$$

$$precision = \frac{TP}{TP + FP}, \tag{3}$$

$$recall = \frac{TP}{TP + FN}, \tag{4}$$

$$F1\text{-}score = 2 \times \left(\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \right). \tag{5}$$

In equations (2) to (5), *TP* (True Positive) denotes correctly identified stunted cases, *TN* (True Negative) represents correctly identified non-stunted cases, *FP* (False Positive) indicates cases incorrectly predicted as stunted, and *FN* (False Negative) represents cases incorrectly predicted as non-stunted.

III. Results and Discussions

A. Exploratory Data Analysis

The dataset utilized in this study comprises health information of toddlers from Bandarharjo Health Center. After feature selection, it includes key attributes such as *Gender*, *Age_at_Measurement*, *Weight*, *Height*, *Weight/Age*, *Weight/Height*, *ZS_Weight/Age*, *ZS_Weight/Height*, and *ZS_Height/Age*. The *Height/Age* feature is used as the target variable for model training.

In Fig. 2, the distribution of the target feature reveals a pronounced imbalance. Specifically, the majority class, 'Normal', significantly outnumbers the minority class, 'Stunting'. This imbalance poses a challenge as it may bias the model towards the majority class, potentially impairing its ability to accurately predict the minority class and affecting overall performance. It is further revealed in Fig. 3 that the *Weight* feature has numerous extreme upper bound points, while all *Z-Score* features show extreme values on both bounds. These outliers can introduce noise and skew results, underscoring the need for careful handling to optimize model performance and ensure reliable predictions.

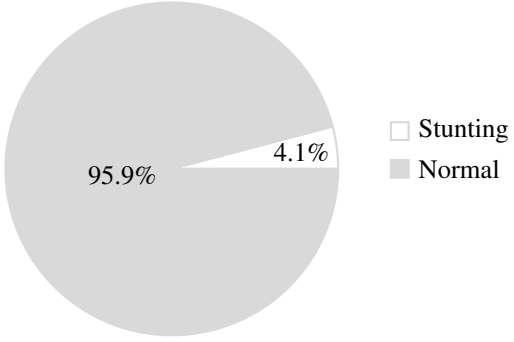

Fig. 2. Class Distribution of the Target Feature

B. Data Preprocessing

Feature selection is a critical step in data preprocessing to enhance model accuracy and robustness by eliminating redundant and irrelevant features. This study reduces the original 27 features to 10, including the target feature.

To ensure the features can be used, several encoding techniques were applied. The *Age_at_Measurement* feature was converted to months for consistency. The *Gender* feature was encoded by removing any whitespace and mapping 'M' to 0 and 'F' to 1. The *Weight/Age* feature was encoded as follows: 'Severely Underweight' to 0, 'Underweight' to 1, 'Normal' to 2, and 'Risk of Overweight' to 3. For the *Height/Age* feature, 'Normal' was mapped to 0, while both 'Stunted' and 'Severely Stunted' were mapped to 1. Finally, the *Weight/Height* feature was encoded by mapping 'Severely Wasted' to 0, 'Wasted' to 1, 'Normal' to 2, 'At Risk of Overweight' to 3, 'Overweight' to 4, and 'Obese' to 5.

To address outliers in the numerical features, infrequent outliers at both ends were replaced with the median, while more prevalent outliers were capped using winsorization. Winsorization, a statistical technique that mitigates the impact of outliers, involves capping extreme values at a specified percentile closer to the center of the data distribution [21]. Subsequently, numerical features were scaled to a range between 0 and 1 using MinMaxScaler. The MinMaxScaler adjusts feature values by first subtracting the minimum value, then dividing by the difference between the maximum and minimum values, effectively normalizing the data within a specified range. This uniform scaling is advantageous for models sensitive to feature

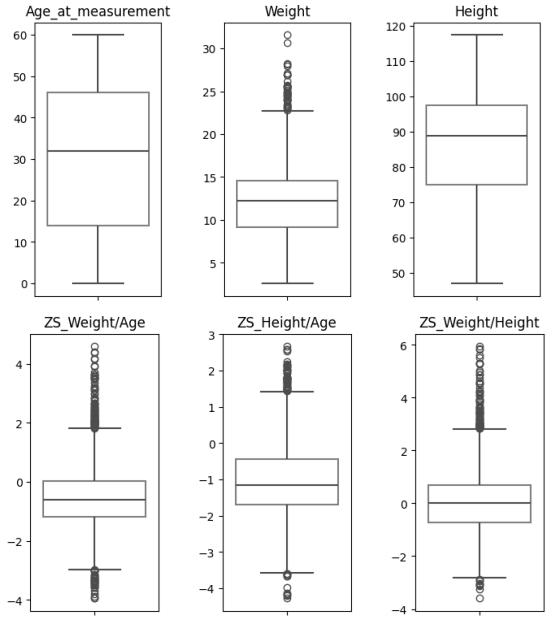

Fig. 3. Visual of Numerical Features

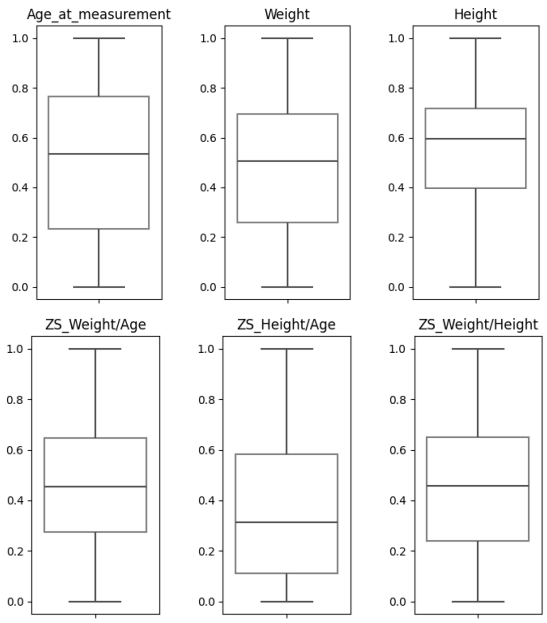

Fig. 4. Visual of Numerical Features After Preprocessing

TABLE I
CONFUSION MATRIX FOR THE KNN MODEL

		Predicted Class		
		Negative	**Positive**	**Total**
Actual Class	**Negative**	1060	24	1084
	Positive	8	45	53
	Total	1068	69	1137

scaling, such as KNN. The results of these preprocessing steps are shown in Fig. 4.

The dataset exhibited a significant class imbalance with a ratio of 1:23, comprising 155 stunted cases versus 3,632 normal cases. To address this imbalance, the Synthetic Minority Over-sampling Technique (SMOTE) was applied. SMOTE generates synthetic samples for the minority class, balancing the class distribution [5]. However, before applying SMOTE, the data is split into 70% training and 30% testing sets to ensure the testing data remains genuine and unaltered. The training set initially contained 2,548 normal cases and 102 stunted cases. After applying SMOTE to the training data, the set was balanced with 2,548 instances of both normal and stunted cases. This balanced distribution is crucial for improving the performance of machine learning models, which might otherwise be biased towards the majority class.

C. Implementation of KNN

The KNN algorithm was implemented using the scikit-learn library in Python with the parameters: metric='euclidean', n_neighbors=5, and weights='distance'. After fitting, KNN proved to be an effective baseline model due to its simplicity and robustness in handling the stunting dataset. The model's performance is presented in Table I, demonstrating its capability to identify patterns in the data.

TABLE II
CONFUSION MATRIX FOR THE RF MODEL

		Predicted Class		
		Negative	**Positive**	**Total**
Actual Class	**Negative**	1079	5	1084
	Positive	5	48	53
	Total	1084	53	1137

TABLE III
CONFUSION MATRIX FOR THE STACKED RFKNN MODEL

		Predicted Class		
		Negative	**Positive**	**Total**
Actual Class	**Negative**	1081	3	1084
	Positive	6	47	53
	Total	1087	50	1137

D. Implementation of RF

The RF algorithm was subsequently implemented using the scikit-learn library in Python, the RF model was configured with the parameters: bootstrap=False, criterion='gini', max_depth=None, and n_estimators=100. Demonstrating superior performance over the KNN model, the RF model effectively handled the complexities of the stunting dataset. The detailed results are outlined in Table II.

E. Implementation of Stacked RFKNN

Finally, a Stacked RFKNN model was constructed to leverage the strengths of both RF and KNN algorithms. The predictions from the previously fitted RF and KNN models were then used as input features for the meta-learner, which is another RF model in this case. The meta-learner was trained on these predictions to determine the optimal way to combine them and correct any individual errors made by the base models. The Stacked RFKNN model demonstrated superior performance, achieving the highest predictive accuracy and F1-score among the tested models. The results, presented in Table III, highlight the effectiveness of ensemble learning in improving model performance on the stunting dataset.

F. Model Evaluation

Table IV presents the accuracy and F1-score of the models evaluated in this study. The KNN model achieved an accuracy and F1-score of 97.19% and 86.14%, respectively. This demonstrates that while KNN serves as a reliable baseline model, it has limitations in handling the complexity of the dataset, particularly in achieving a high F1-score, which balances precision and recall.

The RF model showed a significant improvement, accurately predicting 99.12% of the data with an F1-score of 95.05%. This improvement can be attributed to the RF model's capability to capture intricate patterns and

TABLE IV
ACCURACY AND F1-SCORE OF MODELS

Model	**Accuracy**	**F1-score**
KNN	97.19%	86.14%
RF	99.12%	95.05%
Stacked RFKNN	99.21%	95.42%

interactions within the data, leveraging its ensemble of decision trees to enhance predictive performance.

The Stacked RFKNN model, which combines the strengths of both RF and KNN through a stacking ensemble technique, outperformed the other models achieving an accuracy of 99.21% and an F1-score of 95.42%. This superior performance highlights the effectiveness of the stacking approach, where the base models' predictions are used to train a meta-learner, in this case, further improving the model's ability to generalize and accurately predict outcomes. The Stacked RFKNN model's overall higher performance, particularly its elevated F1-score, indicates an enhanced capability to balance precision and recall, making it the best performing model in this study.

IV. Conclusion

Stunting remains a significant public health concern in Indonesia, making effective early detection methods crucial. This study has demonstrated the efficacy of machine learning models in classifying stunting among toddlers, focusing on the implementation and evaluation of the Stacked RFKNN model. This model was designed to enhance performance by integrating the strengths of both RF and KNN through a stacking ensemble technique, which combines the predictive power of multiple models to improve accuracy, with MinMaxScaler normalization ensuring consistent feature scaling, and SMOTE playing a crucial role in addressing class imbalance. The Stacked RFKNN model demonstrated superior performance compared to the individual RF and KNN models. The Stacked RFKNN achieved the highest accuracy of 99.21% and an F1-score of 95.42%, as shown in Table IV. In comparison, the RF model achieved an accuracy of 99.12% and an F1-score of 95.05%, while the KNN model achieved an accuracy of 97.19% and an F1-score of 86.14%. These results highlight why the Stacked RFKNN model outperforms both individual RF and KNN models in stunting detection. Taking advantage of the strengths of RF and KNN, the model addresses the limitations of each algorithm. RF captures the overall data structure, while KNN focuses on local patterns, allowing the model to refine decision boundaries and correct errors. This study highlights the critical role of machine learning techniques in addressing complex health challenges and supports the broader implementation of these methods to improve the effectiveness of health interventions aimed at reducing stunting rates. Future research could explore the integration of additional data sources, such as genetic or environmental factors, to further refine these models and improve their applicability across diverse populations.

References

[1] A. Almomany, W. R. Ayyad, and A. Jarrah, "Optimized implementation of an improved KNN classification algorithm using Intel FPGA platform: Covid-19 case study," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, pp. 3815-3827, 2022. doi: 10.1016/j.jksuci.2022.04.006.

[2] A. Alzubaidi, S. Halawani, and M. Jarrah, "Towards a stacking ensemble model for predicting diabetes mellitus using a combination of machine learning techniques," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, Dec. 2023. doi: 10.14569/IJACSA.2023.0141236.

[3] Asian Development Bank, "Stunting prevalence," Asian Development Bank, 2024. [Online]. Available: https://kidb.adb.org/

[4] A. U. Berliana and A. Bustamam, "Implementation of stacking ensemble learning for classification of COVID-19 using image dataset CT scan and lung X-ray," in *Proc. 3rd Int. Conf. Inf. Commun. Technol. (ICOIACT)*, 2020, pp. 148-152. doi: 10.1109/ICOIACT50329.2020.9332112.

[5] R. C. Bhagat and S. S. Patil, "Enhanced SMOTE algorithm for classification of imbalanced big-data using random forest," in *Proc. IEEE Int. Adv. Comput. Conf. (IACC)*, 2015, pp. 403-408. doi: 10.1109/IADCC.2015.7154739.

[6] M. G. Daffa and P. H. Gunawan, "Stunting classification analysis for toddlers in Bojongsoang: A data-driven approach," in *Proc. Int. Conf. Sci. Eng. Inf. Technol. (ICoSEIT)*, Feb. 2024, pp. 42–46. doi: 10.1109/ICoSEIT60086.2024.10497515.

[7] R. Devi and P. Sumanjani, "Improved classification techniques by combining KNN and random forest with Naive Bayesian classifier," in *Proc. IEEE Int. Conf. Eng. Technol. (ICETECH)*, Mar. 2015, pp. 1-4. doi: 10.1109/ICETECH.2015.7274997.

[8] S. Džeroski and B. Ženko, "Is combining classifiers with stacking better than selecting the best one?," *Mach. Learn.*, vol. 54, pp. 255–273, 2004. doi: 10.1023/B:MACH.0000015881.36452.6e.

[9] C. Fannany, P. H. Gunawan and N. Aquarini, "Machine learning classification analysis for proactive prevention of child stunting in Bojongsoang: A comparative study," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Kuta, Indonesia, 2024, pp. 1-5. doi: 10.1109/ICoDSA62899.2024.10651698.

[10] H. Janawisuta, P. H. Gunawan and Indwiarti, "Early detection of stunting in Indonesian toddlers: A machine learning approach," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Kuta, Indonesia, 2024, pp. 12-16. doi: 10.1109/ICoDSA62899.2024.10651637.

[11] H. Guo, H. Nguyen, D.-A. Vu, and X.-N. Bui, "Forecasting mining capital cost for open-pit mining projects based on artificial neural network approach," *Resour. Policy*, vol. 74, pp. 101474, 2021. doi: 10.1016/j.resourpol.2019.101474.

[12] Kementerian Sekretariat Negara Republik Indonesia, "Presiden targetkan angka stunting di Indonesia turun hingga 14 persen pada 2024," Jan. 2023. [Online]. Available: https://www.setneg.go.id/

[13] Kementerian Kesehatan Republik Indonesia, "Apa itu stunting," Kementerian Kesehatan Republik Indonesia, Sept. 2022. [Online]. Available: https://yankes.kemkes.go.id/

[14] G. A. F. Khansa and P. H. Gunawan, "Predicting stunting in toddlers using KNN and Naïve Bayes methods," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Kuta, Indonesia, 2024, pp. 17-21. doi: 10.1109/ICoDSA62899.2024.10651676.

[15] J. Liang, Q. Liu, N. Nie, B. Zeng, and Z. Zhang, "An improved algorithm based on KNN and random forest," in *Proc. 3rd Int. Conf. Comput. Sci. Appl. Eng. (CSAE)*, Sanya, China, 2019, pp. 74. doi: 10.1145/3331453.3360963.

[16] S. Pebrianti, R. Astuti, and F. Basysyar, "Penerapan algoritma K-nearest neighbor dalam klasifikasi status stunting balita di Desa Bojongemas," *Jurnal Mahasiswa Tek. Inf.*, vol. 8, no. 2, pp. 2479-2488, Apr. 2024. doi: 10.36040/jati.v8i2.8448.

[17] M. Rahim, M. Hossain, M. Hossain, J. Shin, and K. Yun, "Stacked ensemble-based type-2 diabetes prediction using machine learning techniques," *AETiC*, vol. 7, pp. 30-39, Jan. 2023. doi: 10.33166/AETiC.2023.01.003.

[18] A. K. Sandhu and R. S. Batth, "Software reuse analytics using integrated random forest and gradient boosting machine learning algorithm," *Softw. Pract. Exp.*, vol. 51, no. 22, pp. 735–747, 2020. doi: 10.1002/spe.2921.

[19] A. T. A. Sibuea, P. H. Gunawan and Indwiarti, "Classifying stunting status in toddlers using K-nearest neighbor and logistic regression analysis," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Kuta, Indonesia, 2024, pp. 6-11. doi: 10.1109/ICoDSA62899.2024.10652063.

[20] World Health Organization, "Stunting in a nutshell," Nov. 2015. [Online]. Available: https://www.who.int/

[21] F. Zubedi, B. Sartono, and K. Notodiputro, "Implementation of winsorizing and random oversampling on data containing outliers and unbalanced data with the random forest classification method," *J. Nat.*, vol. 22, 2022. doi: 10.24815/jn.v22i2.25499.