OPEN ACCESS
ISSN 2356-5462
http://journals.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 11, No. 1, Jun 2025. pp. 26-35
doi: 10.21108/ijoict.v11i1.9111

StuntCare: Digital Innovation for Early Warning of Stunting-Risk Families in Sigi Regency

Maulidyani Abu [1], Muh. Al-Fath Salsabilah [2], Juni Wijayanti Puspita [3*], Resnawati [5], Abdul Mahatir Najar [6], Rina Ratianingsih [7], Agus Indra Jaya [8], Abunawas Tjaija [9]

[1,2,3,4,5,6,7,8]*Mathematics Study Program, Tadulako University, Indonesia*
[9]*Public Administration Study Program, Muhammadiyah Palu University, Indonesia*

[*] juni.wpuspita@yahoo.com

Abstract

The Prevalence of Stunting in Sigi Regency remains notably high at 36.8%, significantly above the national target. Stunting is frequently caused by recurrent infections, poor sanitation, and chronic nutritional deficiencies. Since stunting is a condition of chronic malnutrition that impairs a child's physical and cognitive development, an early warning system is essential for prevention. This study proposes the development of a web-based application to predict the risk of stunting in vulnerable families. Families are the primary focus as they serve as the first environment where children grow and develop. If risk factors are present within a family, the likelihood of stunting increases. Therefore, early detection is crucial for mapping family health conditions. By predicting stunting risks, families can take preventive measures before the condition severely impacts the child. This early warning system serves as a critical alarm, encouraging families to be more vigilant in maintaining the health of all household members. The stunting prediction system is developed as a web-based application, utilizing 11 variables for early stunting detection and employing the K-Nearest Neighbor (KNN) method. The model's accuracy is evaluated using a Confusion Matrix, achieving an accuracy rate of 99.991%.

Keywords: Early Warning System, Stunting, Classification, K-Nearest Neighbor, Confusion Matrix

Received on 15 May 2025. Revised on 11 Jun 2025. Accepted and Published on 2 July 2025.

I. INTRODUCTION

Stunting remains the most significant nutritional challenge in Indonesia, particularly among toddlers (BADUTA/BALITA), commonly referred to as growth retardation in children. As of 2017, Indonesia ranked fifth globally in stunting prevalence, with a rate of 29.6% [1]. This condition stems from multiple factors, including poor childcare practices, insufficient maternal knowledge about health and nutrition before and after childbirth, limited access to healthcare services, inadequate household availability of nutritious food, and restricted access to clean water and sanitation.

Recent data from the 2022 National Nutritional Status Survey indicate a declining trend in stunting prevalence among toddlers in 13 districts and cities across Central Sulawesi Province. Banggai Laut showed the most notable improvement, with a 6.1% reduction compared to 2021 [2]. In Sigi Regency, the stunting rate also declined, from 40.7% to 36.8% [2]. Despite these positive developments, the rate in Sigi remains significantly above the national target of 14%. To address this gap, the regency has established a goal to reduce stunting prevalence to 23% by 2024 [3].

The National Population and Family Planning Board (BKKBN) identifies families at risk of stunting based on 11 key factors. These include having children under two (BADUTA) or five (BALITA) years old, couples of reproductive age (PUS), pregnant women, inadequate water sources, poor sanitation facilities, marriage before age 19 or after 35, birth intervals of less than two years, high parity (excessive number of children), and non-participation in modern family planning programs. These variables form the basis of this study. Currently, stunting risk assessments are conducted manually, which is inefficient and prone to errors given the large number of households in each region. This highlights the urgent need for an automated, reliable system to improve accuracy and efficiency in identifying at-risk families.

Advancements in technology offer a viable solution through the development of a web-based application for stunting risk prediction. Classification, a process used to categorize data with unknown class labels, serves as the foundation for this system [4]. Among various classification methods, the K-Nearest Neighbor (K-NN) algorithm is particularly suitable due to its simplicity, rapid training process, ease of interpretation, and effectiveness in handling large datasets [5]. Implementing this approach will enable more accurate and efficient identification of families at risk of stunting, facilitating timely interventions to mitigate long-term health impacts on children. Moreover, with standardized indicators from BKKBN (Badan Kependudukan dan Keluarga Berencana Nasional), this system can be scaled nationally, ensuring consistent and data-driven stunting prevention efforts across Indonesia.

II. LITERATURE REVIEW

Previous research related to stunting risk prevention in infants and toddlers has been conducted by several researchers. [6] and [7] conducted stunting-focused education and training for community health workers. Ningrum and Ikawati [8] in their study performed early stunting detection in toddlers using a feature selection approach with Multilayer Perceptron. Sudiati et al [9] also conducted early detection accompanied by IoT-based stunting education, but their detection focused only on infants. Kartinawati et al [10] similarly performed early stunting detection in toddlers using machine learning models including Random Forest, Support Vector Machine, and Multilayer Perceptron, identifying influencing factors such as breastfeeding practices, parental education, and infant weight and length. Additionally, Lukmana et al [11] developed an expert system for early stunting detection in children using the Forward Chaining method where stunting assessment was based on factors including body weight, height, and head circumference, and as well as Purwati and Sulistyo [12] that using KNN Machine Learning.From those studies, none have examined early detection of families at risk of stunting as a preventive measure.

In this study, early detection is performed by examining factors established by BKKBN: having children under two or five years old, couples of reproductive age (PUS), pregnant women, inadequate water sources, poor sanitation facilities, marriage before age 19 or after 35, birth intervals of less than two years, high parity (excessive number of children), and non-participation in modern family planning programs. These key factors will form the basis for developing a detection system using one of the machine learning models, K-Nearest Neighbour, with accuracy tested using Confusion Matrix. The resulting model will then be implemented in a website where users can perform early detection and receive follow-up for those identified as at risk of stunting.

TABLE 1
RESEARCH ABOUT STUNTING

No.	Method and Data	[6]	[7]	[8]	[9]	[10]	[11]	[12]	Our Research
1	Multilayer Perceptron	-	-	√	-	√	-	-	-
2	IoT-based (mobile based) detection (manual)	-	-	-	√	-	-	-	-
3	Random Forest	-	-	-	-	√	-	-	-
4	Support Vector Machine	-	-	-	-	√	-	-	-
5	Forward Chaining	-	-	-	-	-	√	-	-
6	KNN	-	-	-	-	-	-	√	√
7	Web-based detection	-	-	-	-	-	√	√	√
8	Using children measurement for indicator	-	-	√	√	√	√	√	-
9	Using family condition for indicator	-	-	-	-	-	-	-	√

III. RESEARCH METHOD

The web based application for predicting families at risk of stunting in Sigi Regency will be implemented through the stages given in Fig. 1.

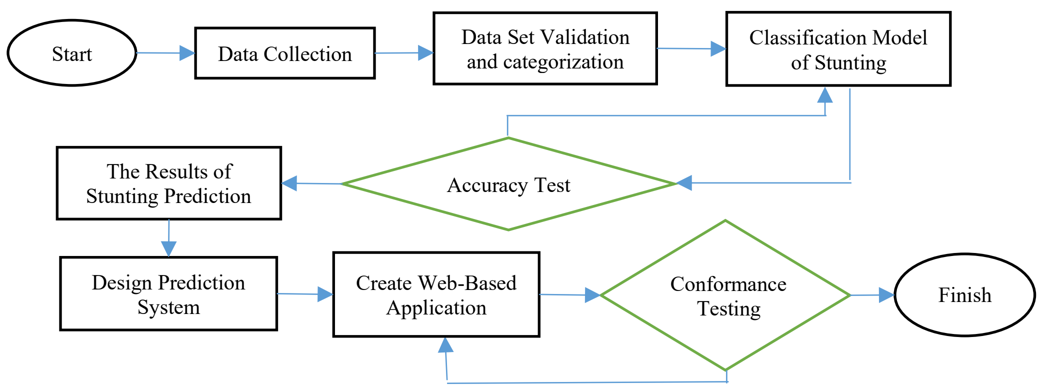

Fig. 1. Research Flowchart

A. Dataset

The dataset was obtained from the National Population and Family Planning Board (BKKBN) in Central Sulawesi Province, comprising family records across the Sigi Regency. The collected data was then checked for completeness of attributes. The Incomplete data removed from the dataset.

B. Classification Model of Stunting

From 40,379 complete records, we allocated 70% (28,265) as training data and 30% (12,114) as test data, evaluating model performance on 11 attributes. The K-Nearest Neighbour algorithm was then applied. The KNN algorithm determines class labels by identifying the most frequent class among k-nearest neighbors, where 'nearness' is quantified using these following distances as written as (1) and (2).

1) Euclidean Distance

$$Dist(X,Y) = \sqrt{\sum_{i=1}^{D} (x_i - y_i)^2} \tag{1}$$

where $Dist\ (X, Y)$ represents the inter-object distance (Euclidean Distance), x_i as training data, y_i as testing data, i as data variable or attribute, and D as data dimension [13].

2) Manhattan Distance

$$d(X,Y) = \sum_{i=1}^{n} |x_i - y_i| \tag{2}$$

where $d\ (X,Y)$ represents distance between x and y, i denotes the attribute index of the data, n as total number of data points, x_i is the i-th data point of the cluster center, and y_i is the i-th data point in the dataset [14].

These distances are the frequent formula that always use for calculating distance in K-NN and easy to implement. After obtaining the nearest neighbour distances, they are sorted in ascending order. The test data's class is then assigned based on majority voting among its k-nearest neighbours.

C. Accuracy Test

After class assignment, we evaluated the classifier's performance using a Confusion Matrix (TABLE 2). This matrix quantifies the model's ability to discriminate between classes by comparing predicted and actual labels [15].

TABLE 2
CONFUSION MATRIX [15]

Class	Predicted Positive	Predicted Negative
Actually Positive	TP	FN
Actually Negative	FP	TN

TP : Number of positive-class data correctly classified by the system.
TN : Number of negative-class data correctly classified.
FN : Number of positive-class data misclassified as negative.
FP : Number of negative-class data misclassified as positive.
The accuracy is calculated using the following formula:

$$Accuracy = \frac{TP + TN}{total} \tag{3}$$

After accuracy validation, the model is tested by classifying new stunting data to evaluate its predictive performance on unseen cases.

D. Create Web-Based Application

The Web-Based Application was developed to facilitate stunting risk prediction for families in Sigi Regency. The system was implemented using Python programming language within the Visual Studio Code development environment.

IV. RESULTS AND DISCUSSION

A. Classification and Prediction

The KNN algorithm was deployed with $k \in \{1, \dots, n\}$ (where n = training data size) for hyperparameter optimization. Initial proximity measurements employed:

1) Euclidean Distance

The Euclidean distance was calculated using Equation (1). Let training data X_i and test data Y_j (where $i = 1, \dots, 28265$ and $j = 1, \dots, 12114$) be represented as vectors $X_i = (v_{1i}, v_{2i}, v_{3i}, v_{4i} \dots, v_{11i})$

and $Y_j = (v_{1j}, v_{2j}, v_{3j}, v_{4j},, .. \ v_{11j})$ where v_l $(l = 1,2,3,4,5,6, \ldots 11)$ denotes the l-th variable (attribute). The Euclidean distance between the first test data $Y_1 = (0,0,0,0,0,0,0,0,0,1,0)$ and the first training data $X_1 = (0,0,0,0,0,0,0,0,0,0,0)$ is computed as follows:

$$d(X_1, Y_1) = \sqrt{\begin{array}{l}(0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-0)^2 + (0-1)^2 + (0-0)^2 \\ +(0-1)^2 + (0-0)^2\end{array}}$$
$$= 1.414$$

The distance calculation between test and training data was performed sequentially until all 28,265 training data were processed. This procedure was repeated for each test data, from the second to the 12,114th test case. The complete distance computation results are presented in TABLE 3.

TABLE 3
CALCULATED EUCLIDEAN DISTANCES

Training Data and Test Data (X_i, Y_j)	**Euclidean Distance**
(X_1, Y_1)	1.414214
(X_2, Y_1)	1.414214
(X_3, Y_1)	1.732051
(X_4, Y_1)	1.732051
(X_5, Y_1)	1
….	…..
(X_{28265}, Y_1)	0

2) Manhattan Distance

The Manhattan distance was computed using Equation (2). The Manhattan distance between the first test data $Y_1 = 0,0,0,0,0,0,0,0,0,1,0$ and the first training data $X_1 = 0,0,0,0,0,0,0,0,0,0,0$ is calculated as follows:

$$\begin{aligned} d(X_1, Y_1) &= |0-0| + |0-0| + |0-0| + |0-0| + |0-0| + |0-0| + |0-0| + |0-1| + |0-0| \\ &\quad +|0-1| + |0-0| \\ &= 2 \end{aligned}$$

The distance calculation between test data and training data was performed sequentially until processing all training data points. This procedure was repeated for each subsequent test data point, from the second test data point to the 12,114th test data point. The complete distance calculation results are presented in TABLE 4.

TABLE 4
CALCULATED EUCLIDEAN DISTANCES

Training Data and Test Data (X_i, Y_j)	**Manhattan Distance**
(X_1, Y_1)	2
(X_2, Y_1)	2
(X_3, Y_1)	3
(X_4, Y_1)	3
(X_5, Y_1)	0
….	…..
(X_{28265}, Y_1)	1

After calculating the distances between test data and training data, the next step is to sort these distances from smallest to largest. Once the data is sorted by distance, the class is determined by majority voting among the k-nearest neighbours (here using $k = 1$).

TABLE 5
CLASSIFICATION RESULTS FOR K=1 NEAREST NEIGHBOR DATA

Distance	$k = 1$	**Status**	
		At-Risk	**Non-At-Risk**
Euclidean	2.067	0	2.067
Manhattan	2.067	0	2.067

According to TABLE 5, it can be concluded that the first test data (Y_1) belongs to the non-stunting-risk family category. The same process is performed for the second test data through the last test data A Confusion Matrix was then constructed to visualize the comparison between model predictions and actual data, as well as to calculate model accuracy. The Confusion Matrix and accuracy values for each distance calculation are presented below.

TABLE 6
CONFUSION MATRIX OF EUCLIDEAN DISTANCE

		Prediction Label	
		Non-At-Risk	At-Risk
Actual Label	Non-At-Risk	6588	0
	At-Risk	2	5524

Based on TABLE 6, out of 12,114 data points, only two were misclassified. The model correctly predicted 6,588 data points as "Non-At-Risk" (True Negative) and 5,524 data points as "At-Risk" (True Positive), with no False Positive prediction errors. However, there were two False Negative prediction errors where two "At-Risk" data points were incorrectly predicted as "Non-At-Risk". The accuracy value was then calculated using Equation (3) as follows.

$$Accuracy = \frac{6588 + 5524}{6588 + 0 + 2 + 5524} = 0{,}99983$$

The accuracy value of the model using Euclidean distance is 0,99983 or 99,983%

TABLE 7
CONFUSION MATRIX OF MANHATTAN DISTANCE

		Prediction Label	
		Non-At-Risk	At-Risk
Actual Label	Non-At-Risk	6588	0
	At-Risk	2	5524

According to TABLE 7, out of 12,114 data points, only two were misclassified. The model correctly predicted 6,588 data points as "Non-Risk" (True Negative) and 5,524 data points as "At-Risk" (True Positive), with no False Positive prediction errors. However, there were two False Negative cases where two "At-Risk" data points were incorrectly classified as "Non-Risk". The accuracy value was then calculated using Equation (3).

$$Accuracy = \frac{6588 + 5524}{6588 + 0 + 2 + 5524} = 0.99983$$

The accuracy value of the model using Manhattan distance is 0.99983 or 99.983%.

The same procedure was repeated from calculating distances between test and training data using different k-values through to computing accuracy. The accuracy results for k-values ranging from 1 to 34 (as accuracy decreases for $k > 34$) are shown in Fig. 2.

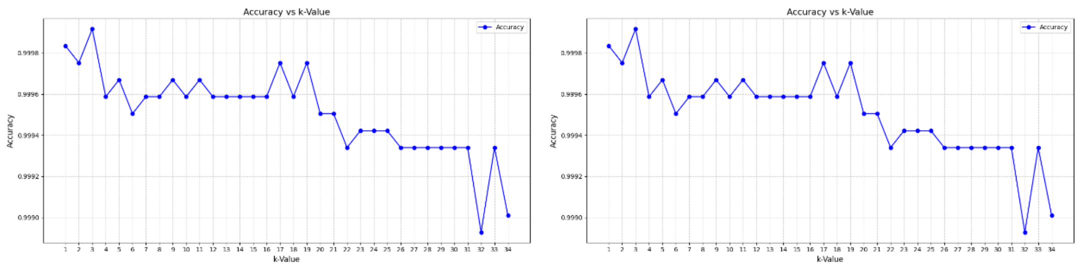

Fig. 2. Accuracy vs. k-Value Plot Using Euclidean Distance (Left) and Manhattan Distance (Right)

Fig. 2 demonstrates that the k-value yielding the highest accuracy for both Euclidean and Manhattan distances is $k = 3$. These results are evident in the Confusion Matrix, which shows the accuracy achieved with $k = 3$ using both Euclidean and Manhattan distance methods (they have the same results).

TABLE 8
CONFUSION MATRIX OF EUCLIDEAN AND MANHATTAN DISTANCE

		Prediction Label	
		Non-At-Risk	At-Risk
Actual Label	Non-At-Risk	6588	0
	At-Risk	1	5525

Based on TABLE 8, out of 12,114 data points, only one was misclassified—a single "At-Risk" data point incorrectly predicted as "Non-At-Risk." The accuracy calculation yields the following results:

$$Accuracy = \frac{6588 + 5524}{6588 + 0 + 2 + 5524} = 0.99991$$

The accuracy value of the model using Euclidean and Manhattan distance is 0.99991 or 99.991%.

B. Web-Based Application

To facilitate stunting risk prediction for families in Sigi Regency, we developed a web-based application using Python programming language within the Visual Studio Code development environment. The resulting programming outputs are displayed as follows:

1) Login Page Interface

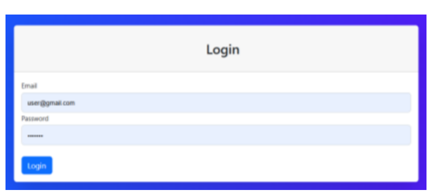

Fig 3. Login Page Interface

The login page serves as the authentication gateway to the website, requiring users to input their registered email address and password for system access.

2) Dashboard Page

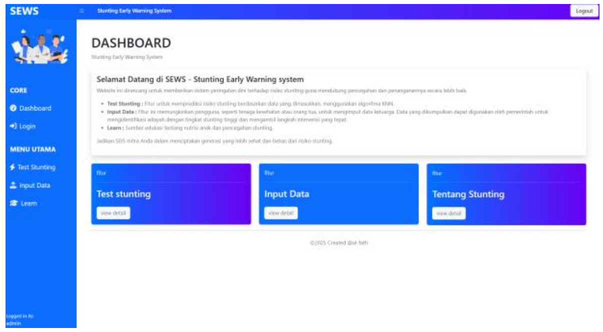

Fig 4. Dashboard Page

This page serves as the default interface that presenting the appropriate dashboard for either regular users or administrators based on their login credentials. On the dashboard, user can select which one of the features that they want which are Test Stunting, Input Data, and About Stunting. These three features' function can be known as below.

3) Test Stunting Form Page for Prediction

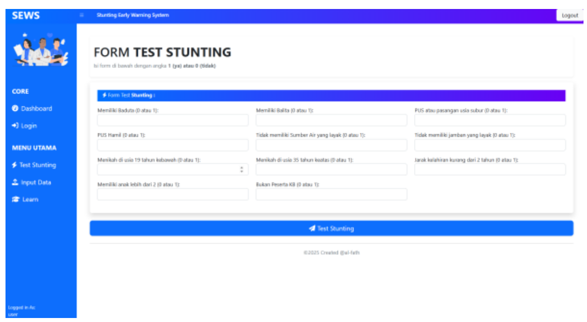

Fig 5. Form Test Stunting Page

This interface serves as the primary data entry portal for family information to generate stunting risk predictions. The user can input the form according to the condition of their family. Below the input form, a dedicated "Test Stunting" button triggers the system to process and display prediction results.

4) Prediction Result

The input data on the prediction page will be divided into two main categories: "At-Risk" and "Non-At-Risk" families. These categories provide information about stunting risk based on the entered family data.

To provide a clearer representation, these prediction results can be visually displayed in Fig. 6 and 7.

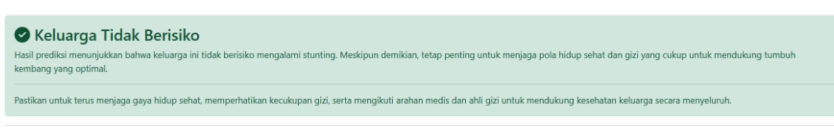

Fig 6. Non-At-Risk Family Results Interface

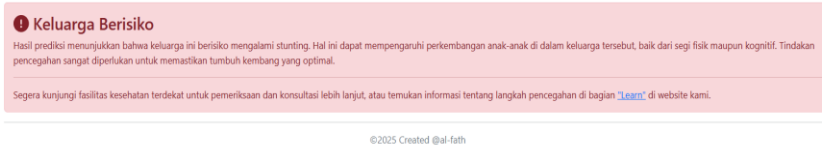

Fig 7. At-Risk Family Results Interface

When predictions classify a family as "At-Risk", clicking the "learn" button triggers navigation to the subsequent information page.

5) Data Input Page

The data input page displays two different interfaces depending on user privileges. For regular users, the page only shows the family data input form containing fields for "Family Head Name," "Address," "National ID Number (NIK)," and "Status." The "Status" field is automatically populated with either "At-Risk" or "Non-Risk" based on the prediction results. For administrators, the page additionally displays all previously submitted family data records alongside the input form. This dual-view functionality ensures users only see relevant information while allowing administrators full access to collected data for monitoring and analysis purposes.

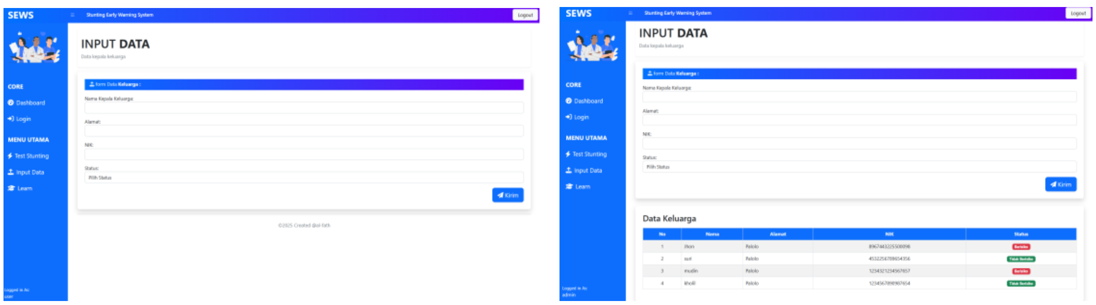

Fig. 8. Data Input Page for User (Left) and Admin (Right)

6) *Learning About Stunting Page*

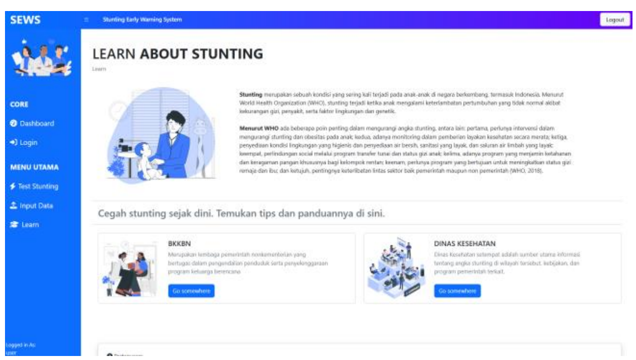

Fig. 9. Learn Page

This page is designed to provide relevant information and guidance for stunting risk management and reduction efforts. It serves as an educational resource to support prevention initiatives, offering recommendations for “At-Risk” family.

7) *Logout*

Fig. 10. Logout Icon

The interface includes a logout button positioned in the top-right corner, accessible to both regular users and administrators when they need to securely exit the website.

V. CONCLUSION

The study demonstrates that K-Nearest Neighbor (KNN) predictions using $k = 3$ (the value yielding highest accuracy) identify 5,525 families at risk of stunting and 6,589 non-risk families, based on both Euclidean and Manhattan distance metrics. The model achieves 99.991% accuracy when evaluated via Confusion Matrix at $k = 3$. As part of this research, a web-based application was developed to facilitate practical stunting risk prediction. This tool is designed to streamline the identification of families at risk of stunting, offering an efficient alternative to manual assessment methods.

ACKNOWLEDGMENT

This research was supported by faculty of Mathematics and Natural Science, Tadulako University.

REFERENCES

[1] S. Lonang, A. Yudhana, and M. K. Biddinika, "Analisis Komparatif Kinerja Algoritma Machine Learning untuk Deteksi Stunting," *Jurnal Media Informatika Budidarma*, vol. 7, no. 4, p. 2109, Oct. 2023, doi: 10.30865/mib.v7i4.6553.

[2] TPPS (Tim Percepatan Penurunan Stunting), "Laporan Penyelenggaraan Percepatan Penurunan Stunting," Palu, 2023.

[3] Kominfosantik, "Turunkan Prevelensi Stunting Kab. Sigi 2024, Pemprov. Sulteng Buat Program Tangguh Bersinar," PEMPROV Sulawesi Tengah . Accessed: May 14, 2025. [Online]. Available: https://topikterkini.com/2024/01/12/turunkan-prevelensi-stunting-kab-sigi-2024-pemprov-sulteng-buat-program-tangguh-bersinar/

[4] L. andiani, J. Ilmu Komputer, and D. Palupi Rini, *Analisis Penyakit Jantung Menggunakan Metode KNN Dan Random Forest*, vol. 5 (1). pp. 191-196. ISSN 979-587-573-6, 2019.

[5] D. Cahyanti, A. Rahmayani, and S. Ainy Husniar, "Indonesian Journal of Data and Science Analisis performa metode Knn pada Dataset pasien pengidap Kanker Payudara," vol. 1, no. 2, pp. 39–43, 2020.

[6] N. A. Fajar *et al.*, "Deteksi dini stunting berbasis edukasi dan pendampingan kader kesehatan," *SELAPARANG : Jurnal Pengabdian Masyarakat Berkemajuan*, vol. 8, no. 4, pp. 4387–4396, Dec. 2024.

[7] R. D. Aisyah, S. Suparni, and E. Subowo, "Pemberdayaan Kader SMART Stunting (Pencegahan dan Deteksi Pada Calon Pengantin, Ibu Hamil dan Balita)," *Journal of Community Development*, vol. 5, no. 3, pp. 434–445, Oct. 2024, doi: 10.47134/comdev.v5i3.289.

[8] A. A. Ningrum and Y. Ikawati, "Early Detection of Stunting Based on Feature Engineering and Machine Learning Algorithm Approach," *Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 3, Aug. 2024, doi: 10.35882/ijeeemi.v6i3.6.

[9] L. E. Sudiati, W. Aditama, and D. A. Puryono, "IoT-based Stunting Education and Early Detection System for Stunting-Free Indonesia," *East Asian Journal of Multidisciplinary Research*, vol. 2, no. 12, pp. 4855–4866, Dec. 2023, doi: 10.55927/eajmr.v2i12.6767.

[10] K. T. Kartinawati, L. G. Pradnyawati, and I. M. E. Dwipayana, "Early Detection Of Children's Development On Stunting Toddlers," *Jurnal Kedokteran*, vol. 7, no. 2, p. 57, Jul. 2022, doi: 10.36679/kedokteran.v7i2.521.

[11] H. Hen Lukmana, M. Al-Husaini, I. Hoeronis, and L. Desi Puspareni, "Pengembangan Sistem Informasi Deteksi Dini Stunting Berbasis Sistem Pakar Menggunakan Metode Forward Chaining".

[12] N. Purwati and G. B. Sulistyo, “Stunting Early Warning Application Using KNN Machine Learning Method,” *Jurnal Riset Informatika*, vol. 5, no. 3, pp. 373–378, Jun. 2023, doi: 10.34288/jri.v5i3.228.

[13] R. Dwi Yulian Prakoso, B. Soejono Wiriaatmadja, and F. Wahyu Wibowo, *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS) Sistem Klasifikasi Pada Penyakit Parkinson Dengan Menggunakan Metode K-Nearest Neighbor*.

[14] M. Nishom, “Perbandingan Akurasi Euclidean Distance, Minkowski Distance, dan Manhattan Distance pada Algoritma K-Means Clustering berbasis Chi-Square,” *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 4, no. 1, pp. 20–24, Jan. 2019, doi: 10.30591/jpit.v4i1.1253.

[15] S. Proboningrum and A. Sidauruk, “Sistem Pendukung Keputusan Pemilihan Supplier Kain Dengan Metode Moora,” *Sistem Informasi* |, vol. 8, no. 1, pp. 43–48, 2021.