AI-Based Stunting Classification Model And Distribution Of Stunting Zone Using Streamlit Framework

Tjut Awaliyah Zuraiyah[a,*], Nurdina Widanti[b], Yamato[c], Andi Chairunnas[d], Kriti Mauludin[e], Bira Arya Setha[f]

[a] Computer Science Study Program, Pakuan University, Jl. Pakuan P.O. BOX 452, Bogor City, 16129, Indonesia
[b] Jayabaya University, Jl. Raya Jakarta Bogor No. 28, East Jakarta, 16452, Indonesia

Corresponding author: tjut.awaliyah@unpak.ac.id

Abstract*— Stunting is a public health problem that requires serious attention. Stunting in children is not only a height problem, but most dangerously, it can lead to low learning capacity, mental retardation and the onset of chronic diseases. Artificial intelligence (AI) can help make decisions by analysing available data. Deep learning (DL) models are AI models that can automatically provide systems to help make intelligent decisions. The purpose of this study is to explore the performance of AI in classifying stunting in young children using the DL algorithm with two classes, namely stunting or severaly stunting and displaying the number of distribution of stunting zones in each village using the streamlit framework with the folium library. The method used is data collection and uses the Cross-Industry Standard Process for Data Mining (CRISP DM) method which includes the following stages: Business Problem, Data Understanding, Exploration Data Analysis, Data Modelling and Model Deployment. The material used is stunting data from the Bogor City Health Office for 2022-2024. SMOTE is used to balance the dataset to increase accuracy. The DL models used are LSTM and BiLSTM. The model performance is evaluated by assessing accuracy, precision, recall and F1 score and the results show that the BiLSTM model is better after balancing the data with SMOTE. This intelligent system is dynamic and provides automatic stunting classification and distribution zone display, so that stakeholders receive the right recommendations for stunting decision making.

Keywords*— classification; stunting; deep learning; SMOTE; LSTM ; BILSTM; streamlit; folium.

Manuscript received 15 Oct. 2020; revised 29 Jan. 2021; accepted 2 Feb. 2021. Date of publication 17 Feb. 2021.
International Journal on Informatics Visualization is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. INTRODUCTION

Stunting is an important factor in assessing the growth and development rate of infants and toddlers [1] The relationship between nutrition and development is a reciprocal relationship, which means that nutrition will determine the success of a nation. But unfortunately, until now nutrition is still a major problem, especially in developing countries [2]–[4] Stunted children are at risk of increased morbidity and mortality, decreased immune system immunity and increased risk of infection [5]. The long-term effects cause a child's failure to reach his or her cognitive potential and physical abilities, thus affecting his or her future work capacity and socioeconomic status [6].

Toddlers at risk of stunting can have a long-term impact on children's growth and development. By utilizing artificial intelligence [7], it is hoped that a classification model can be developed that can predict the potential for toddlers to experience stunting. AI can make a significant contribution to handling stunting in toddlers through the identification of risk factors, prediction of potential stunting cases, and the design of more appropriate interventions. With careful data analysis using artificial intelligence technology, stunting handling can be carried out more efficiently and effectively. AI can also help in the early detection of stunting cases so that interventions can be carried out faster, so that the risk of complications due to stunting can be minimized [8].

AI can be interpreted as intelligence that is integrated into a system that can be regulated in the context of the intelligence of scientific entities. Based on its taxonomy, AI is implemented into several techniques [9] : ML, neural network (NN) [10], and DL [11]. The data collection method is carried out through surveys and observations of toddlers in certain zones. The collected data is then processed and analyzed using machine learning algorithms [12]–[14]. Some of the machine learning algorithms that are often used in handling stunting in toddlers include Decision Tree [15], [16], Random Forest [4], [17], XGBoost [18], [19].

The development trend of malnutrition research with ML algorithms has an exponential trend, and it is decreasing in 2023 [11]. An interesting trend is shown in research with DL. DL. There are several DL algorithm models that can be used to classify *time series* data, one of which is *Long Short-Term Memory* (LSTM). LSTM is a *deep learning* algorithm introduced by Hochreiter and Schmidhuber. LSTM arises because it can remember long-term information (*long term dependency*) that can be used for classification. The main goal of LSTM in the case of forecasting is to make an accurate prediction of a variable. The best forecasting is based on the prediction error rate, where the smaller the error rate, the more accurate a method is in predicting the variable [20]. Furthermore, another *deep learning algorithm* was introduced by Graves and Schmidhuber, namely *Bidirectional* LSTM (BiLSTM) which is a development algorithm of LSTM variants. BiLSTM is a developed LSTM that can overcome the weaknesses in the LSTM method [21], BiLSTM [22] has two additional inputs, namely backward and *forward*, which are used to improve the performance of LSTM which only has one input, namely *forward*.

For stunting classification using unbalanced datasets, it has been found that with the utilization of the Synthetic Minority Synthetic Oversampling Technique (SMOTE) to handle unbalanced target variables has been used [23], [24]. SMOTE works by adding synthetic data to the minority class to make the data balanced. The application of the SMOTE oversampling method tends to increase the accuracy to be higher and can even reach twice the accuracy without SMOTE [25]. The results of this study compare the results of the model that applies preprocessing with those without preprocessing. The model that uses only LSTM has the best accuracy of 78.35%; models with normalization produce an accuracy of 81.53%; models using SMOTE produce an accuracy of 81.66%; and models using normalization and SMOTE produce the best accuracy of 85.79%

Based on the problems mentioned above by utilizing artificial intelligence technology, a deep learning classification model will be created using the LSTM and BiLSTM methods of stunting data in Bogor city. Meanwhile, the visualization of the results of the best method will be displayed using the streamlit framework application in the form of a distribution of stunting zones. It is hoped that this model of classification and distribution of stunting zones can increase the effectiveness of interventions and reduce stunting rates. This approach makes a valuable contribution in understanding and addressing the problem of stunting in toddlers and can be a guideline for more targeted and data-driven health policies.

II. MATERIALS AND METHODS

This section describes the methodology used covering *Business understanding*, Data Understanding, data preparation, modelling, evaluation and deployment. Figure 1 shows the research steps using the Cross-Industry Standard Process for Data Mining (CRISP-DM) method. CRISP-DM is a method that provides a standard process in data mining that is easier to apply because each stage or phase is clearly defined and structured and has a complete and well-documented data mining methodology.

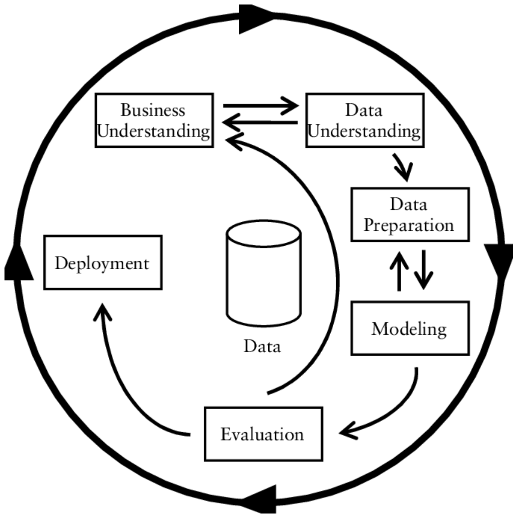

Fig. 1 Research Methods [26]–[28]

A. *Business Understanding*

The business objective of this reasearch is to create an intelligent stunting classification system automatically and dynamically with the distribution of distribution zones. The objective specification looks at the best accuracy values of ML and DL as well as visualization of the distribution of stunting zones.

B. *Data Understanding*

After going through the business understanding stage, several things that must be done in this stage include recognizing and understanding the data to be used, taking or collecting data, and analyzing the data. This study uses datasets collected and obtained from the Bogor City Health Office. The details of the dataset will be shown in more depth in Table 1.

TABLE 1

DATASETS, DATA COUNTS AND FEATURE COUNTS

Dataset	Amount of Data	Number of Features
BNBA Stunting 2022 – 2024	6032	32

The dataset shown in Table 1 comes from the Bogor City Health Office which was previously used as basic data to provide assistance to stunted children. Dataset 2 features are added, namely *latitude, longitude* which is useful for the process of visualizing stunting distribution zones at *the deployment* stage using *Streamlit* so that the total features in the dataset are 32 features.

The dataset is collected annually by the Bogor City Health Office to monitor stunting trends, especially in the Bogor City area and the data provided for this study is stunting data for 3 years starting from 2022 to 2023. The amount of data for the 3 years is 6032 data.

After understanding the characteristics of the data, the next process analyzes the dataset. The analysis process showed that the distribution of stunting classes between 'Short' (stunted) and 'Very Short' (severaly stunted) was significantly unbalanced, this was shown by the 'Very Short' class amounting to 1638 and the 'Short' class amounting to 4394. The distribution of stunting classes will be shown in Figure 2.

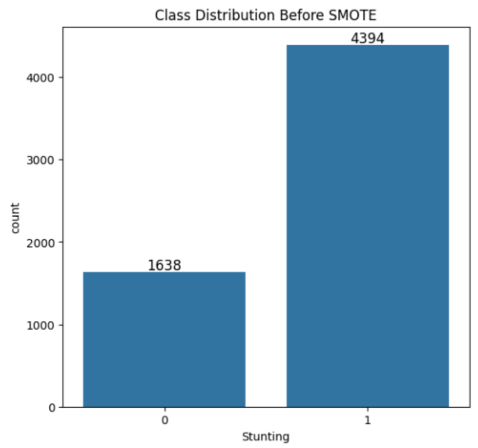

Fig. 2 Distribution of Classes Before SMOTE

C. Data Preparation

Data preparation is carried out in 2 stages, namely, by means of *data cleaning* and *data transformation*. After the data is obtained, *data cleaning* is carried out by deleting data that is not suitable or not needed in the study, such as deleting or filling in the missing value data in the research data The *Data Preparation* stage in the CRISP-DM method includes the process of preparing initial data whose characteristics are known to be more ideal data to enter into the modeling process.

The 'Age at Measurement' column whose initial format of the data (Year - Month - Day) is changed to a numerical form, namely in the format (month). The column was also renamed to make it easier to become 'Age When Measuring (Month)'.

In addition to changing the data format and column names, in this stage the dataset also needs to go through feature selection. The feature selection process aims to ensure that the features included in the model are the most relevant features and have the highest correlation value to the classification process. Variable correlation will show the correlation between two variables, either the linear correlation is towards -1 or positively correlated towards 1. Figure 3 will show the correlation between all variables, especially numerical variables.

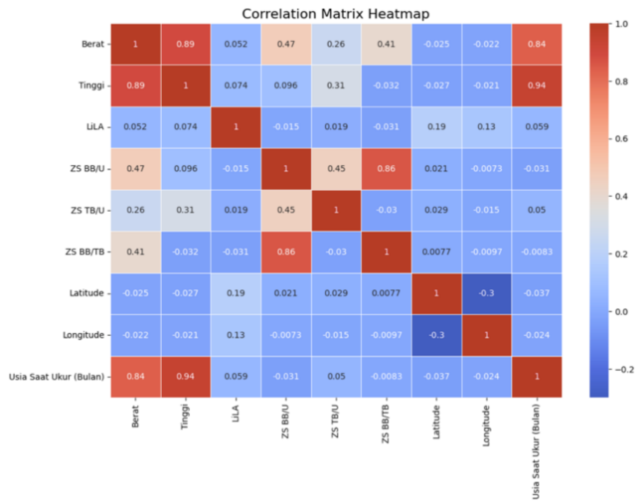

Fig. 3 Correlation of Stunting Dataset Variables

The selection of features is based on the correlation value of positive variables and looks at the Regulation of the Minister of Health of the Republic of Indonesia related to nutritional status thresholds, especially in the stunting category. So that the variables selected are 4 to be entered into the model, namely Weight, Height, Age When Measured (Month), and ZS TB/U. So that Table 2 will show the overall features and Table 3 will display the features selected based on the two things above.

TABLE 2

DATASET BEFORE FEATURE SELECTION

No	Column	Data Type
1	JK, Date of Birth, BB Born, TB Born, Prov, Regency/City, Kec, Puskesmas, Village/Kel, Posyandu, RT, RW, Address, Age at Uku, Date of Measurement, Lila, BB/U, TB/U, Weight Gain, PMT Received (kg), KPSP, KIA,	Object
2	Berat, Tinggi, ZS BB/U, BB/TB, ZS TB/U, ZS BB/TB, JML VIT A, Latitude, Longitude, Usia Saat Ukur (Bulan)	Float64

After the feature selection process and the features that will be used in the modeling process are selected. The following is shown the dataset that has gone through the feature selection process.

TABLE 3

DATASET AFTER FEATURE SELECTION

No	Column	Data Type
1	Heavy	Float64
2	Tall	Float64
3	TB/U	Object
4	ZS TB/U	Float64
5	Age at Measurement (Months)	Float64

The selected features need to go through the last stage of processing, namely data scaling. Scale the data using the function of *StandardScaler().* This function is useful for reducing outliers and after this stage, the dataset preparation is complete and ready to enter the modeling stage.

D. Modeling

This stage involves direct *deep learning* to determine the *data mining* technique to be used. At this stage, the best modeling scenario is built by selecting the algorithm to be used, dividing the data according to the availability of data into training data and test data.

1. Long Short-Term Memory (LSTM)

Long short-term memory is a type of RNN architecture that was first introduced by Hochreiter & Schmidhuber in 1997. According to Dr. Budi Raharjo (2022:70) in his book entitled *Deep Learning* with *Python. LSTM* is a modified RNN architecture that addresses the problem of gradient loss as well as the problem of long data sequences to preserve memory. Figure 4 shows *the neurons on the LSTM* to see the architecture of the *long short-term memory algorithm*.

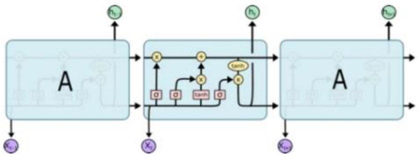

Fig. 4 Neurons on LSTM

2. Bidirectional LSTM

BiLSTM is one of the variants of LSTM that has two inputs. Forward and backward inputs are the two types of inputs that are fed into the BiLSTM architecture and the outputs of this architecture have their bias values combined into one. BiLSTM utilizes the previous information and the information after it by processing data from two directions. The forward layer serves to present the information beforehand, while the backward layer serves to present the information afterwards. The following is an illustration of the BiLSTM architecture which can be seen in Figure 5.

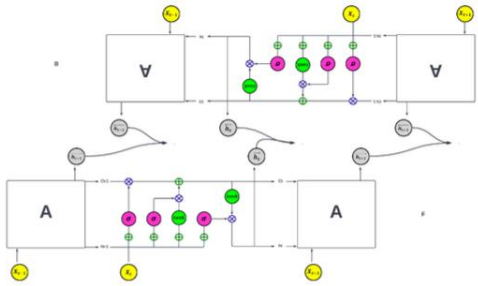

Fig. 5 BiLSTM Architecture

The SMOTE process is carried out first to balance the distribution of classes. Deep *Learning* algorithms are very sensitive to data distribution. Figure 6 shows the distribution after SMOTE and Table 4 shows the distribution of classes before and after the *implementation of SMOTE.*

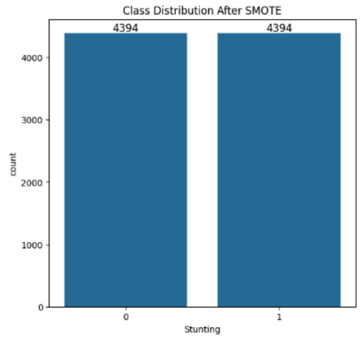

Fig. 6 Distribution After SMOTE

TABLE 4

DISTRIBUTION BEFORE AND AFTER SMOTE

No	Column	'Short' Class	'Very Short' Classes
1	Dataset Awal	4394	1638
2	SMOTE	4394	4394

This modeling has 2 processes that include *Deep Learning* using *Long Short-Term Memory (LSTM)* and *LSTM Bidirectioan* which can be seen in Figure 10 for LSTM modeling and Figure 7 modeling using *the BILSTM algorithm model.*

```
# Membangun dan melatih model LSTM untuk data setelah SMOTE
model_after = Sequential()
model_after.add(LSTM(50, activation='relu', input_shape=(X_train_after.shape[1], X_train_after.shape[2])))
model_after.add(Dropout(0.2))
model_after.add(Dense(4, activation='softmax'))

model_after.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
history_after = model_after.fit(X_train_after, y_train_after, epochs=100, batch_size=32, validation_split=0.2)
```

Fig. 7 Deep Learning Model with LSTM

The next process is modeling with *the BILSTM algorithm.* This model is a development of *the previous* LSTM model where *the LSTM network* is created in two layers or repeatedly. So that this model can overcome the limitations of the traditional artificial neural network, this model can be seen in Figure 8.

```
if len(X_train_bilstm.shape) == 2:
    X_train_bilstm = np.expand_dims(X_train_bilstm, axis=-1)


# Inisialiasi model algoritma BiLSTM
model_BiLSTM = Sequential()
model_BiLSTM.add(Bidirectional(LSTM(50, activation='relu', input_shape=(X_train_bilstm.shape[1], X_train_bilstm.shape[2]))))
model_BiLSTM.add(Dropout(0.4))
# output untuk klasifikasi
model_BiLSTM.add(Dense(4, activation='softmax'))

# Build model compiler
model_BiLSTM.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Melatih model dengan menggunakan set data Training
history_BiLSTM = model_BiLSTM.fit(X_train_bilstm, y_train_bilstm, epochs=100, batch_size=32, validation_split=0.2)
```

Fig. 8 Deep Learning Model with BILSTM

E. Evaluation

This stage is carried out to see the performance level of the pattern generated by the confusion matrix algorithm to find out the accuracy level of *the recall value* starting from correct positive predictions, false positive predictions, correct negative predictions and incorrect negative predictions.

F. Deployment

The last stage that will be carried out is the creation of a web application to make it easier for interested parties to use. The best model based on the *evaluation* stage is used at this stage. This stage uses the streamlit framework [29], [30] where 2 main menus are made, namely classification (Early Detection of Stunting and Tunning Model) and visualization of stunting distribution zones. Figure 9 will show the display of the stunting classification menu in the early detection feature by inputting 3 variables, namely Height, Age and Gender of the child [31].

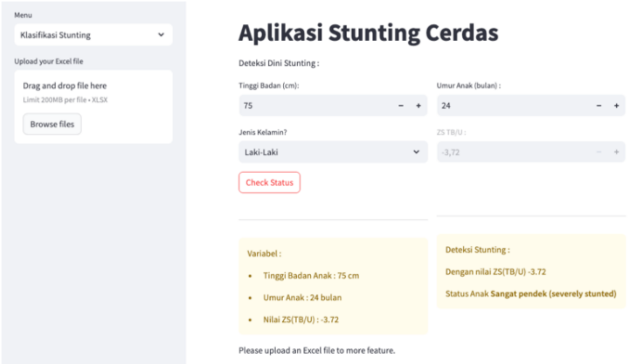

Fig. 9 Stunting Detection Results

Another menu display in the *deployment stage* is the visualization of stunting distribution zones. The display of the stunting distribution map in the Bogor City area will be shown in Figure 10.

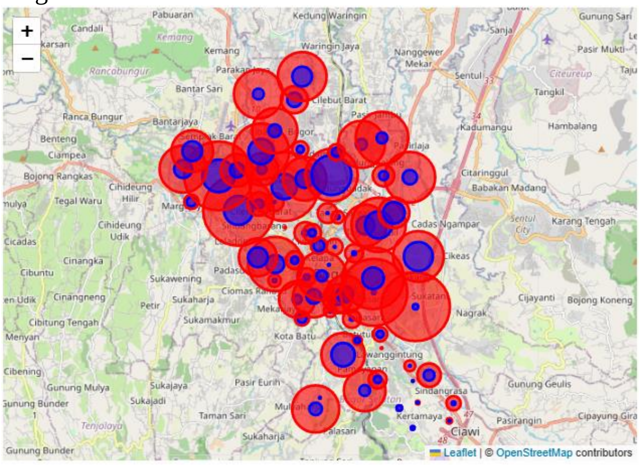

Fig. 10 Stunting Distribution Map in Bogor City

III. RESULTS AND DISCUSSION

This study emphasizes the use of deep learning algorithms to improve accuracy in classifying stunting. The best algorithm will be used in the deployment process for the visualization of the distribution zone.

A. Classification with Deep Learning

Furthermore, evaluation is also needed in the *deep learning* model where the evaluation metrics use confusion matrix, accuracy, precision, recall, and F1-Score. This evaluation was shown in 3 experiments, namely when the dataset used without SMOTE balancing and the dataset to be used was balanced using SMOTE. The dataset after SMOTE uses 2 algorithms, namely LSTM and BILSTM Table 6 will show the results of a detailed evaluation [32].

TABLE 5

COMPARISON OF DEEP LEARNING MODELS

Evaluation Metrics	No SMOTE with LSTM	With SMOTE	
		LSTM	BILSTM
Accuracy	98.84%	99.20%	99.43%
Precision	0.99	0.99	1.00
Recall	0.96	0.99	1.00
F1-Score	0.97	1.00	0.99

From the results of the evaluation of the five models, *BILSTM* was chosen to be used in the deployment stage because it was superior to the other four algorithms, namely Decison Tree, Random Forest, XGBoost, and LSTM with an accuracy of 99.43%. The BILSTM with the best accuracy will also show the *confusion matrix* value in Figure 11.

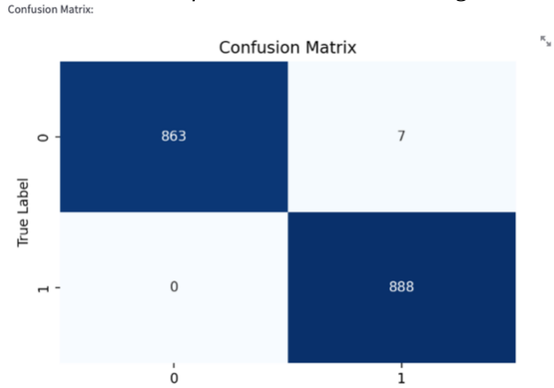

Fig. 11 Confusion Matrix BILSTM

B. Discussion

This study uses the BNBA (By Name By Address) dataset of Stunting in Bogor City in 2022-2024 with unbalanced data. Unbalanced data sets can be overcome with several strategies. One of the strategies in overcoming unbalanced data is to use SMOTE.

SMOTE provides balanced and representative data on stunting conditions. So that the two models built with deep learning provide quite good accuracy by setting some of the necessary parameters.

The classification process with high accuracy can detect children at risk of stunting by considering the characteristics of the child's weight, height and age. This feature is calculated and the ZScore (TB/U) value is the result that determines whether a child is stunted and severaly stunted. This model has the potential to save children and future generations and of course help avoid the possibility of disabilities in children in an early way.

At the deployment stage, the streamlit framework is used by importing the folium library with the command import folium from streamlit_folium import st_folium. Furthermore, to visualise two classes of stunting and severaly stunting data per village, the TB / U column, latitude and longitude are used at show in figure 12.

```python
# Fungsi untuk visualisasi spasial
def spatial_visualization(data):
    # Pisahkan kolom Latitude dan Longitude
    data['lat'] = pd.to_numeric(data['Latitude'], errors='coerce')
    data['lon'] = pd.to_numeric(data['Longitude'], errors='coerce')

    # Hapus baris dengan nilai latitude atau longitude yang tidak valid
    data = data.dropna(subset=['lat', 'lon'])

    # Konversi kolom Tanggal Pengukuran menjadi format datetime
    data['date'] = pd.to_datetime(data['Tanggal Pengukuran'], errors='coerce')

    # Filter data untuk baris di mana 'TB/U' yang termasuk status 'Pendek' atau 'Sangat Pendek'
    filtered_data = data[data['TB/U'].isin(['Pendek', 'Sangat Pendek'])]

    # Hitung frekuensi kemunculan setiap kombinasi lat dan lon
    location_counts = filtered_data.groupby(['lat', 'lon']).size().reset_index(name='count')

    # Hitung jumlah status 'Pendek' dan 'Sangat Pendek' di setiap lokasi
    status_counts = filtered_data.groupby(['lat', 'lon', 'TB/U']).size().unstack(fill_value=0)

    # Ambil desa dan puskesmas yang unik sebagai fungsi untuk setiap lokasi (lat, lon)
    location_info = filtered_data.groupby(['lat', 'lon']).agg({
        'Desa/Kel': 'first',
        'Pukesmas': 'first'
    }).reset_index()
```
Fig. 12 Data per village, the TB / U column, latitude and longitude code

A base map is then created with folium and counted using the count command. It is also coloured red to visualise the amount of stunting data and blue to visualise severaly stunting data as shown in Figure 13.

```python
        # Tambahkan lingkaran untuk 'Pendek' dengan warna merah
        if count_pendek > 0:
            folium.Circle(
                location=(lat, lon),
                radius=count_pendek * 10,  # Radius yang ditentukan oleh jumlah 'Pendek'
                color='red',
                fill=True,
                fill_color='red',
                fill_opacity=0.6
            ).add_to(m)

        # Tambahkan lingkaran untuk 'Sangat Pendek' dengan warna biru
        if count_sangat_pendek > 0:
            folium.Circle(
                location=(lat, lon),
                radius=count_sangat_pendek * 10,  # Radius yang ditentukan oleh jumlah 'Sangat Pendek'
                color='blue',
                fill=True,
                fill_color='blue',
                fill_opacity=0.6
            ).add_to(m)
```
Fig. 13 Coloured red and blue to visualization code

This stunting application is dynamic, by running experiments using Aceh stunting data as seen in Figure 14, the application can display a map of stunting distribution zones in Aceh province.

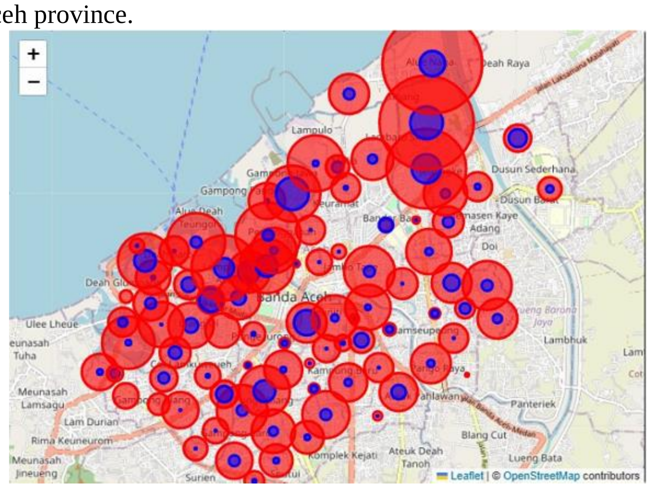

Fig. 14 Stunting Distribution Map in Aceh Povince

The training on the dataset is done twice in parallel, where both trainings use two different algorithms, namely LSTM and Bi-LSTM. Both trainings use the same parameters so that the final results can show which algorithm has the best performance. The parameters used in both training are
Model Epoch Batch Dropout Learning Rate

TABLE 6
MODEL EPOCH BATCH DROPOUT LEARNING RATE

Model	Epoch	Batch	Dropout	Learning Rate
LSTM	30	32	0.2	0.1
Bi-LSTM	30	32	0.2	0.1

The above algorithm and parameters were then implemented using several Python libraries, including Pandas, NumPy, Seaborn, Scikit-learn, Keras Tensorflow and Imbalanced-learn for the SMOTE implementation.

The main factor in Bi-LSTM's high precision and recall is its superior architecture and way of working. In the LSTM architecture, the algorithm processes data sequentially from start to finish, whereas Bi-LSTM processes data both backwards and forwards in parallel. This is because Bi-LSTM relies not only on past data, but also on future data in time series. As a result, the understanding of data elements and sequences is better, especially in stunting classification, which requires more complex time series predictions. The two architectures in this algorithm can be seen again in Figure 14 and Figure 15. we will show the process of Bi-LSTM in learning the pattern of stunting data, as shown in the epoch graph and performance table.

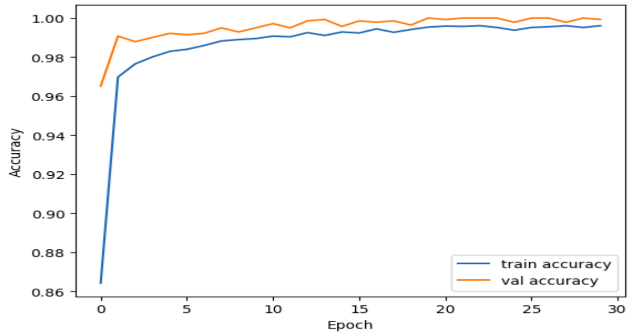

Fig. 14 Train and val accuracy

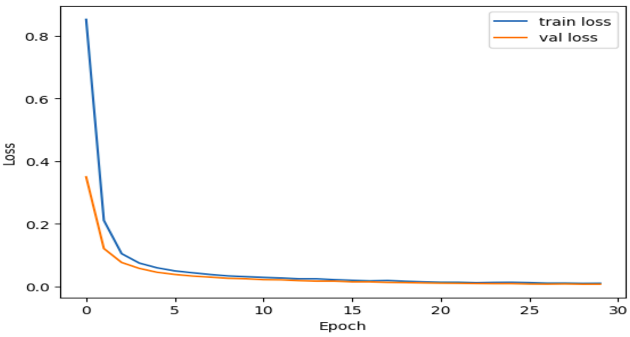

Fig. 15 Train and val loss

TABLE 7
PERFORMANCE AND ACCURACY IMPROVEMENT IN EACH EPOCH

Epoch	Learning Time (seconds)	Akurasi (%)
1	4	95%
2	5	96%
3	8	97%
4	9	97%
5	11	97%
~	~	~
10	16	98%
~	~	~

20	29	98%
~	~	~
30	50	99%

CONCLUSION

In this study, classification was carried out using deep learning with LSTM and BILSTM algorithms. The first stage is carried out, namely the data cleaning and transformation stage, after which the division of training data and test data is carried out, which is 80:20. The correlation feature showed that the variables that greatly influenced were "Age at Measurement (Month)", "Weight" and "Height" which had correlation values of 0.84, 0.89 and 0.94, while the poor correlation values were found in the variables "BB/TB" and "ZS BB/TB", which had a value of -0.83, which indicated that these variables had no effect on stunting outcomes.

The model evaluation using the confusion matrix obtained the accuracy value of each model, namely LSTM without SMOTE 98.84%, LSTM with SMOTE 99.20% and the model with the best accuracy, namely BiLSTM with SMOTE with an accuracy value of 99.43%. The comparison shows that the BiLSTM algorithm produces more accurate classification values than other algorithms.

The model with the best accuracy is then selected to create a web application system using the Streamlit framework. Based on a series of processes that have been carried out in this study, it can be concluded that this study has succeeded in building the best model and web application for stunting classification using the BiLSTM algorithm. This intelligent system is dynamic because it can use stunting datasets for all regions and will provide automatic stunting classification along with the distribution of zones so that stakeholders can get the right recommendations in decision-making.

ACKNOWLEDGMENT

We would like to thank LPPM Pakuan University and the Indonesian Ministry of Education and Culture for financing this research through the BIMA Program of the 2024 Fundamental scheme with contract numbers 26/LPPM-UP/PFR/VI/2024 and 004/LL$/SP2H/RT-MONO/2024. We would also like to thank the reviewers who have provided valuable criticism and suggestions for the improvement of our manuscript.

REFERENCE

[1] E. R. Arumi, S. A. Subrata, and A. Rahmawati, "Implementation of Naïve bayes Method for Predictor Prevalence Level for Malnutrition Toddlers in Magelang City," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 7, no. 2, pp. 201–207, 2023, doi: 10.29207/resti.v7i2.4438.

[2] F. Ahmmed *et al.*, "Association between short birth spacing and child malnutrition in Bangladesh : a propensity score matching approach," 2024, doi: 10.1136/bmjpo-2023-002240.

[3] E. A. Turjo and H. Rahman, "Assessing risk factors for malnutrition among women in Bangladesh and forecasting malnutrition using machine learning approaches," *BMC Nutr.*, pp. 1–25, 2024, doi: 10.1186/s40795-023-00808-8.

[4] Sabbir Ahmed Hemo and M. I. Rayhan, "Classification tree and random forest model to predict under-five malnutrition in Bangladesh," *Biometrics Biostat. Int. J.*, vol. 10, no. 3, pp. 116–123, 2021, doi: 10.15406/bbij.2021.10.00337.

[5] S. M. J. Rahman *et al.*, "Investigate the risk factors of stunting , wasting , and underweight among under-five Bangladeshi children and its prediction based on machine learning approach," pp. 1–11, 2021, doi: 10.1371/journal.pone.0253172.

[6] M. Usman, "Spatial and Machine Learning Approach to Model Childhood Stunting in Pakistan : Role of Socio-Economic and Environmental Factors," 2022.

[7] B. Watkins, L. Odallo, and J. Yu, "Artificial intelligence for the practical assessment of nutritional status in emergencies," no. December 2023, pp. 1–12, 2024, doi: 10.1111/exsy.13550.

[8] B. Vichave, N. Jain, P. Garad, N. Gandhi, and P. D. Meshram, "Malnutrition Detection using AI," *Int. J. Adv. Res. Sci. Commun. Technol.*, pp. 285–291, 2023, doi: 10.48175/ijarsct-9692.

[9] A. A. Permana, B. Raharja, and A. T. Perdana, "Artificial Intelligence for Diagnosing Child Stunting: A Systematic Literature Review," *J. Syst. Manag. Sci.*, vol. 13, no. 6, pp. 605–621, 2023, doi: 10.33168/JSMS.2023.0635.

[10] A. R. Lakshminarayanan, B. Pavani, V. Rajeswari, S. Parthasarathy, A. A. A. Khan, and K. J. Sathick, "Malnutrition Detection using Convolutional Neural Network," in *Proceedings of 2021 IEEE 7th International Conference on Bio Signals, Images and Instrumentation, ICBSII 2021*, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/ICBSII51839.2021.9445188.

[11] H. Yuliansyah, T. W. Sukesi, S. A. Mulasari, and W. N. S. W. Ali, "Bulletin of Social Informatics Theory and Application Artificial intelligence in malnutrition research: a bibliometric analysis," vol. 7, no. 1, pp. 32–42, 2023, doi: 10.31763/businta.73i1.605.

[12] M. Deprez and E. C. Robinson, "Machine learning basics," *Mach. Learn. Biomed. Appl.*, pp. 41–65, 2024, doi: 10.1016/B978-0-12-822904-0.00007-8.

[13] S. Ndagijimana, I. H. Kabano, E. Masabo, and J. M. Ntaganda, "Prediction of Stunting Among Under-5 Children in Rwanda Using Machine Learning Techniques," pp. 41–49, 2023.

[14] E. Darnila, K. Mawardi, M. Sinambela, and I. Pahendra, "Supervised Models to Predict the Stunting in East Aceh," vol. 2, no. 3, pp. 34–39, 2022.

[15] A. I. Putri, Y. Syarif, P. Jayadi, F. Arrazak, and F. N. Salisah, "Implementasi Algoritma Decision Tree dan Support Vector Machine (SVM) untuk Prediksi Risiko Stunting pada Keluarga," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 2, pp. 349–357, 2024, doi: 10.57152/malcom.v3i2.1228.

[16] A. Nugroho, H. L. Hendric, S. Warnars, F. L. Gaol, and T. Matsuo, "Trend of Stunting Weight for Infants and Toddlers Using Decision Tree."

[17] P. Handayani and A. C. Fauzan, "KLIK: Kajian Ilmiah Informatika dan Komputer Machine Learning Klasifikasi Status Gizi Balita Menggunakan Algoritma Random Forest," *Media Online)*, vol. 4, no. 6, pp. 3064–3072, 2024, doi: 10.30865/klik.v4i6.1909.

[18] H. Shen, H. Zhao, and Y. Jiang, "Machine Learning Algorithms for Predicting Stunting among Under-Five Children in Papua New Guinea," 2023.

[19] O. N. Chilyabanyama *et al.*, "Performance of Machine Learning Classifiers in Classifying Stunting among Under-Five Children in Zambia," *Children*, vol. 9, no. 7, 2022, doi: 10.3390/children9071082.

[20] L. Wiranda and M. Sadikin, "PENERAPAN LONG SHORT TERM MEMORY PADA DATA TIME SERIES UNTUK MEMPREDIKSI PENJUALAN PRODUK PT . METISKA

FARMA," vol. 8, pp. 184–196, 2019.
[21] D. R. Alghifari, M. Edi, and L. Firmansyah, "Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia Bidirectional LSTM Implementation for Sentiment Analysis Against Grab Indonesia Services," vol. 12, pp. 89–99, 2022.
[22] T. Chawla, S. Mittal, and H. K. Azad, "MobileNet-GRU fusion for optimizing diagnosis of yellow vein mosaic virus," *Ecol. Inform.*, vol. 81, no. August 2023, p. 102548, 2024, doi: 10.1016/j.ecoinf.2024.102548.
[23] F. M. Amin, D. Candra, and R. Novitasari, "Identification of Stunting Disease using Anthropometry Data and Long Short-Term Memory (LSTM) Model," vol. 11, no. 1, pp. 25–36, 2025.
[24] A. A. G. Y. Pramana, M. F. Maulana, M. C. Tirtayasa, and D. A. Tyas, "Enhancing Early Stunting Detection: A Novel Approach using Artificial Intelligence with an Integrated SMOTE Algorithm and Ensemble Learning Model," in *Proceedings - 2024 IEEE Conference on Artificial Intelligence, CAI 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 486–493. doi: 10.1109/CAI59869.2024.00098.
[25] C. Haryawan and Y. M. K. Ardhana, "ANALISA PERBANDINGAN TEKNIK OVERSAMPLING SMOTE," vol. 6, no. 1, pp. 73–78, 2023.
[26] M. J. Nodeh, M. H. Calp, and İ. Şahİn, "Analyzing and Processing of Supplier Database Based on the Cross-Industry Standard Process for Data Mining (CRISP-DM) Algorithm Analyzing and Processing of Supplier Database Based on the Cross-Industry Standard Process for Data Mining (CRISP-DM) Algorithm," no. January, 2020, doi: 10.1007/978-3-030-36178-5.
[27] A. Purbasari, U. Pasundan, F. Rinawan, U. Padjadjaran, and A. Zulianto, "CRISP-DM for Data Quality Improvement to Support Machine Learning of Stunting Prediction in Infants and Toddlers," no. September, 2021, doi: 10.1109/ICAICTA53211.2021.9640294.
[28] R. Wirth and J. Hipp, "CRISP-DM : Towards a Standard Process Model for Data Mining," no. 24959.
[29] Y. Akkem, B. S. Kumar, and A. Varanasi, "Streamlit Application for Advanced Ensemble Learning Methods in Crop Recommendation Systems – A Review and Implementation," *Indian J. Sci. Technol.*, vol. 16, no. 48, pp. 4688–4702, 2023, doi: 10.17485/IJST/v16i48.2850.
[30] A. Tolba and N. Nabil, "Innovative Approach for Early Detection and Diagnosis of Tomato Leaf Diseases," no. June, 2024, doi: 10.61356/j.oia.2024.1197.
[31] R. N. Rachmawati and N. H. Pusponegoro, "Spatial Bayes Analysis on Cases of Malnutrition in East Nusa Tenggara, Indonesia," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 337–343. doi: 10.1016/j.procs.2021.01.014.
[32] S. Ankalaki, V. G. Biradar, P. K. K. Naik, and G. S. Hukkeri, "A Deep Learning Approach for Malnutrition Detection," *Int. J. online Biomed. Eng.*, vol. 20, no. 6, pp. 116–138, 2024, doi: 10.3991/ijoe.v20i06.46919.