



Comparison of Modern NLP with Classical Machine Learning Algorithms in Evaluating Food Security Programs

Anita Sindar Sinaga^{1*}, Dameria Esterlina Sijabat², Bella Saputri³, Nadia Aulia⁴

^{1,3,4} Information Technology, STMIK Pelita Nusantara, Indonesia

² Digital Business, STMIK Pelita Nusantara, Indonesia

¹haito_ita@yahoo.com, ²sijabatdame@gmail.com, ³bella@gmail.com, ⁴nadia1@gmail.com

Abstract: The success of food security programs faces various challenges. Most of the available data is in the form of unstructured text reports, news, and policy documents. The BERT (Bidirectional Encoder Representations from Transformers) model allows the system to read reports and news by considering the relationship between words in sentences. Compared to Support Vector Machines (SVMs) that rely on numerical data. The dataset is expanded to improve the generalization of the IndoBERT Classifier. There are 6 commodity data and 3 labels used in IndoBERT Modeling, represented by a 768-dimensional feature vector resulting in Accuracy 0.8333 (83.33%) indicating 5 correct predictions, with one misclassification. Tuned Min-Max on Support Vector Machines (SVM) is used in each dimension to find the optimal hyperplane contributing. The feature matrix x with size (39,10) and the target variable y with size (39) show Accuracy 0.92 (92.0%) that the data division process maintains the class proportion consistently. SVM performed better than IndoBERT. Classification evaluation of the models showed IndoBERT with Accuracy 83% and SVM Scuracy 87%.

Keywords: Food Availability; Modeling; Modern NLP; SVM; IndoBERT

1. INTRODUCING

The food security program is a strategic priority for the government to achieve food independence and maintain national economic stability. The Food and Agriculture Organization (FAO) and the Global Report on Food Crisis report noted that more than 282 million people in 59 countries face acute food insecurity due to geopolitical conflict, climate change, the energy crisis, and the pandemic. Measuring the success of food security programs faces various challenges, because most of the available data is unstructured [1]. Plant-based foods are sourced from rice, corn, sugar, cooking oil, shallots, garlic, chilies, and cayenne pepper, while animal-based foods are sourced from beef, broiler chicken, and eggs. Estimating the success of the national food security program is carried out using text-based and numerical data-based modeling [2]. Numerical data is inherently structured and ready to use. Text data is unstructured and requires significant pre-processing steps to convert words into meaningful numerical representations (vectorization). Algorithms for tabular data tend to focus on extracting statistical patterns from predetermined features [3]. Modern text algorithms focus on understanding the sequential context and semantics of language. Natural Language Processing (NLP) technology, specifically the BERT (Bidirectional Encoder Representations from Transformers) model, understands the contextual and semantic meaning of words [4]. Numerical input data (vectors or numerical features) is used by a Support Vector Machine (SVM) to perform classification or regression. Combining the two allows machine learning models to gain a more holistic and in-depth understanding of the phenomenon under study, compared to using only one type of data [5].





BERT is based on the Transformer architecture which consists entirely of self-attention layers. The IndoBERT model has better performance compared to classical methods such as SVM, the BERT model has higher accuracy and relevance in estimating the success of food security programs [6]. This finding not only contributes to the development of Indonesian language artificial intelligence, but also supports the implementation of data-driven policy analysis in the agricultural sector and national food security. The development of the BERT model can estimate the success of the national food security program with higher accuracy than conventional models. BERT can process a word by looking at the context of the words to the left and right simultaneously. Classical algorithms for numerical data model a linear relationship between a dependent variable (target) and one or more features. This algorithm assumes that the data has a straight-line relationship. SVM works by finding the optimal hyperplane that separates data classes with the largest margin. BERT provides higher classification accuracy than SVM, in capturing all sentiments.

A previous study integrated Natural Language Processing (text mining) with traditional machine learning models to analyze narrative reports from FEWS NET (Famine Early Warning Systems Network) to identify drivers of global food insecurity by applying topic classification and ML predictors to textual early warning data, which revealed market- and climate-related factors as key determinants [7]. Another study mapping key NLP applications in food security policymaking showed that insights gained from unstructured documents cannot be effectively captured using classical machine learning alone. Further comparative evaluation applies pre-trained language models based on modern NLP embeddings in conjunction with ML classifiers and compares their performance with classical bag-of-words-based ML and structured nutrition features in predicting food categories and nutrition scores, demonstrating that modern NLP representations provide superior generalization and classification accuracy. Modern NLP approaches such as IndoBERT are specifically designed to process unstructured text through contextual embeddings that capture semantic meaning, whereas classical machine learning relies on structured data or shallow text representations such as TF-IDF or bag-of-words [8]. Although NLP models automatically learn deep semantic relationships and perform better in complex food safety analyses, classical machine learning remains valuable as a basic approach for structured data [9].

This study aims to determine whether IndoBERT contextual embeddings can capture semantic meaning, local linguistic nuances, and latent indicators of food insecurity more accurately than traditional ML methods. The study's contribution lies in its role as one of the first studies to apply IndoBERT to evaluate food security in the Indonesian context, providing empirical evidence on the advantages and limitations of its performance relative to classical ML. This study provides a comparative framework that can guide researchers and policymakers in selecting appropriate analytical methods for Indonesia's food security monitoring system.

2. RESEARCH METHODOLOGY

2.1 Research Stages

Modern NLP models such as BERT can estimate the success of national food security programs compared to conventional algorithmic approaches. Numerical data-based modeling uses a quantitative and structured approach. Text data-based modeling uses a qualitative approach and involves unstructured data [10]. Numeric data measures quantitative facts while text data describes events. BERT learns from language patterns that appear in food texts. BERT automatically understands signals from the dataset. SVM (Support Vector Machine) works to find the optimal hyperplane that maximizes the margin between data classes for classification. Research stages see Fig. 1.



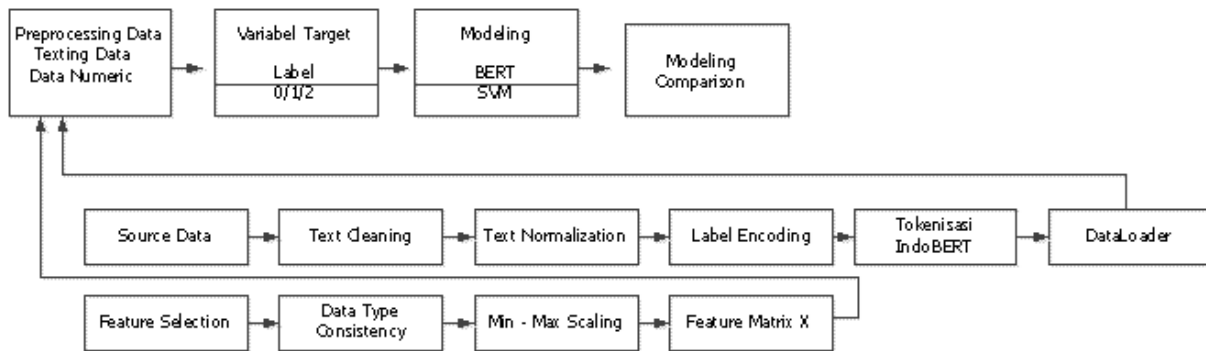


Figure 1. Research Stages

Food availability data was retrieved from portal links using scraping techniques. The scraped text data (taken from the links) was cleaned of punctuation, numbers, special characters, and stopwords [11]. Text in columns was converted to lowercase, punctuation, numbers, and special characters were removed, and common Indonesian words (stopwords) were also removed. Tokenization was performed using the BERT tokenizer. Each text was broken down into smaller units (tokens) using the IndoBERT tokenizer. Embeddings capture the meaning and semantic context of the text. Texts with similar meanings will have closer embedding vectors in multidimensional space [12]. Data text preprocessing see Fig. 2.

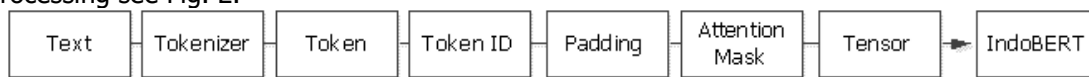


Figure 2. Text Preprocessing

2.2 Data Preprocessing

The food security dataset used in this study is derived from Indonesian food-related data sources, including official food security and agricultural reports from Badan Pusat Statistik (BPS), Kementerian Pertanian Republik Indonesia, and Badan Pangan Nasional, as well as textual policy documents, press releases, and public reports related to food availability, prices, and distribution. These data sources provide unstructured textual information describing food production, market dynamics, climate impacts, and food access conditions, which are suitable for analysis using IndoBERT-based NLP models. Each document was collected as a randomized text string from 14 URLs. Noise was removed by removing excess spaces and non-alphabetic characters. Numbers, units (tons, ha, %, rupiah), commas, periods, and stop words related to production were not removed. Text normalization was performed by converting the text to lowercase. The tokenizer splits subwords (WordPiece), adds [CLS] and [SEP] tokens, converts tokens to numeric IDs, padding (to ensure equal length), and truncation (limited to 512 tokens). The tokenizer generates input_ids, attention_masks, token_type_ids, and labels as classification targets. SVM data preprocessing includes ensuring all data variables are numeric and then handling missing values using the mean or median to avoid modeling errors. The data is divided into training and testing sets to avoid data leakage, followed by feature scaling using StandardScaler or MinMaxScaler. SVM is very sensitive to differences in variable scale. Target normalization (y) improves performance if the target values have a wide range. The BERT model represents each word as a real-valued vector in a low-dimensional space [13].

2.3 Variable Target

This representation is learned from a large amount of data and can capture semantic and contextual relationships between words with similar contexts and similar vector representations. Each text is labeled according to its production level [14]. All compiled text data is categorized by commodity (corn, chicken eggs, rice, beef, broiler chicken, fresh milk) using predefined keywords, as shown in Table 1.



Table 1. Texting Data Sources

Types of Food	Keywords	Id_label	Availability
Corn	['corn availability', 'corn production', 'corn price', 'corn stock']	0	2 (High)
Eggs	['chicken egg availability', 'chicken egg production', 'chicken egg price', 'chicken egg stock']	1	1 (Medium)
Rice	['rice availability', 'rice production', 'rice price', 'rice stock']	2	2 (High)
Beef	['beef availability', 'beef production', 'beef price', 'beef stock']	3	0 (Low)
Chicken	['broiler chicken availability', 'broiler chicken production', 'broiler chicken price', 'broiler chicken stock']	4	2 (High)
Fresh Milk	['fresh milk availability', 'fresh milk production', 'fresh milk price', 'fresh milk stock']	5	1 (Medium)

Text data in NLP is processed by transforming raw text data into structured numeric representations that can be understood and analyzed by machine learning algorithms. The text is converted to a numeric format, such as a vector or matrix. The text data is processed using BERT embedding, which represents text vectors (words, tokens, or sentences) generated by the BERT model. In BERT contextual embedding, the meaning of a word depends on the sentence. Each word is represented as a real-valued vector in a low-dimensional space. Each text is labeled according to its availability/production level. When text is processed using BERT, feature extraction is obtained from the word pieces and contextual embedding. Label 0 = Low, low production/availability. Label 1 = Medium, stable production/availability. Label 2 = High, high production/availability. Feature extraction in the context of SVM can be understood as a combination of internal feature transformations through kernels and external feature extraction techniques to improve the quality of data representation before it enters modeling. Feature extraction in SVM relies on feature transformation through the use of kernels to map the data to a higher-dimensional space. By mapping features to this new space, the non-linear relationship between the input and target variables can be made more linear, making them easier to separate or regress. All feature data is normalized using StandardScaler so that the SVM can optimally separate the classes in the high-dimensional vector space.

2.4 Modeling

BERT transforms each token into a 768-dimensional vector (for IndoBERT Base). BERT embedding consists of Token Embedding, Positional Embedding, and Segment Embedding. The combination of these three produces a contextual representation. BERT understands what fell, its cause, and whether it is an indicator of food success or failure [15]. BERT generates a single primary CLS embedding vector. The resulting vector represents the entire text and is used as input to the classifier. The implementation of the SVM model begins with preparing the numerical data. After the data is cleaned and standardized, the next step is to organize the data into features (X) and targets (y), then separate it into training and testing sections. The SVM model is built by selecting an appropriate kernel. The RBF kernel is capable of capturing non-linear patterns. After preprocessing, these features were used as input in an SVM with a multiclass classification scheme using the One-vs-Rest approach to predict food security success categories, consisting of low (0), medium (1), and high (2).

2.5 Modeling Comparison

The model was evaluated using accuracy metrics, F1 score, Confusion Matrix, and Precision and Recall per class. BERT modeling builds a text-based model that utilizes BERT to predict the success of food security programs, with the output being the probability of success (classification) through targets and labels. Model selection took into account language, hardware resources, and dataset size. BERT transforms each token into a 768-dimensional vector (for IndoBERT Base). BERT embedding consists of Token Embedding, Positional Embedding, and Segment Embedding. The combination of these three produces a contextual representation. BERT understands what falls, its causes, and whether it is an indicator of food security success or failure. BERT generates a single CLS embedding vector. The resulting vector represents the entire text and is used as input to the classifier. BERT





improves multiclass classification. The implementation of the SVM model begins with preparing the numerical data after cleaning and standardizing it. The next step is to organize the data into features (X) and targets (y), then separate it into training and testing sections.

3. RESULT AND DISCUSSIONS

This research employs a comparative modeling approach using two different data representations: unstructured textual data and structured numerical data. The overall workflow consists of data collection, preprocessing, feature representation, model training, and performance evaluation. Textual data related to food availability were collected from multiple online sources using web scraping techniques. The collected texts include reports and news articles discussing food production, supply, and market conditions. Initial preprocessing focused on noise reduction, including the removal of irrelevant characters and normalization through lowercase conversion. Unlike classical NLP pipelines, extensive stemming and stopword removal were not applied, as the IndoBERT tokenizer is designed to preserve contextual information.

The initial stage of IndoBERT modeling involves converting six text units into six numeric vectors (embeddings) that represent the semantic meaning of each text. Each embedding represents a single text document about food availability within the high-dimensional semantic space generated by the IndoBERT model. The IndoBERT tokenizer segments text into subword units using the WordPiece algorithm and generates token identifiers, attention masks, and segment indicators. Each input sequence is truncated or padded to a fixed length to ensure consistency. The IndoBERT base model transforms each token into a contextual vector representation, with the final classification relying on the [CLS] embedding, which represents the semantic summary of the entire text. The resulting embeddings of fourteen articles/texts about food availability have been successfully represented numerically, as shown in Fig. 3.

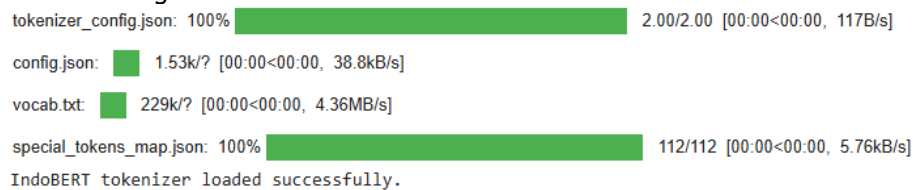


Figure 3. IndoBERT Tokenization

Text tokenization extracts the CLS token embedding, which represents the entire text sequence, resulting in an embedding. The feature (X) of the form (6, 768) indicates that there are six text datasets (documents/articles), each represented by 768 numeric features, which are IndoBERT Base embeddings generated from the semantic representation of the text. The matching number of rows between X and y indicates that each IndoBERT embedding has been paired with the correct label, so the data is ready to be used for the training and evaluation of the classification model, as can be seen in Table 2.

Table 2. Structure Data Set

Column Name	Number of Data (Non-Null)	Type Data	Short Description
URLs	6	Object	Address of news source or article related to food security
text_content	6	Object	Original text from news collection (raw text)
food_category	6	Object	Food commodity category (rice, corn, chicken eggs, etc.)
cleaned_text_content	6	Object	Text that has undergone normalization and cleaning stages
indobert_embeddings	6	Object	Text vector representation from

Anita Sindar Sinaga: *Corresponding Author



Copyright © 2025, Anita Sindar Sinaga, Dameria Esterlina Sijabat, Bella Saputri, Nadia Aulia.



			IndoBERT (without additional cleaning)
<code>indobert_embeddings_cleaned</code>	6	Object	IndoBERT embedding of cleaned text

In classical NLP, text normalization is performed intensively through lowercase conversion, stopword removal, and stemming or lemmatization to reduce word variation before feature extraction, such as TF-IDF. Normalization is minimal because the tokenizer and model already handle lowercase conversion and understand word context without the need for stemming or stopword removal. IndoBERT emphasizes understanding the contextual meaning of the entire text, as seen in Table 3.

Table 3. Differences Between Classical NLP – IndoBERT

Process	Classic NLP	IndoBERT
Lowercase	Manuals	Automatic
Stopword removal	Yes	No
Stemming	Yes	No
Lemmatization	Yes	No
Cleaning ringan	Optional	Yes
Feature scaling	Yes	No

The most critical improvement would be to collect a much larger and more diverse dataset for each commodity and each availability label. This could involve scraping more varied sources, including news archives, government reports, and social media, and potentially manual annotation for higher quality labels. Explore more sophisticated NLP methods for automated labeling, such as weak supervision, transfer learning with other labeled datasets, or rule-based systems refined by domain experts, to reduce reliance on simple keyword matching. Implement a proper train-validation-test split (or cross-validation) if the dataset can be sufficiently expanded. This will provide a more reliable measure of the model's generalization performance. Perform a detailed analysis of misclassified samples to understand common patterns or features that confuse the model, guiding future data collection or feature engineering efforts. If further data expansion and tuning yield limited improvements, consider exploring alternative model architectures or pre-trained models. Investigate text data augmentation techniques to artificially increase the size and diversity of the training data, especially for under-represented classes. Numerical data representing food production and availability indicators were processed separately. Missing values were handled using statistical imputation, and all features were transformed into numeric format. To prevent bias caused by scale differences, feature normalization was applied using Min-Max Scaling.

A multi-class SVM classifier with a radial basis function (RBF) kernel was employed due to its ability to model non-linear relationships. The One-vs-Rest strategy was used to handle multi-class classification. Hyperparameter tuning was conducted to optimize classification performance across all availability categories. The dataset has two main components tokenized `commodity_data`, containing numeric representations and attention masks of the text for each commodity, ready for the NLP model. `commodity_availability_labels` contains category labels for each commodity based on text analysis. The shape of `X_train` (4, 768) indicates that there are 4 text data sets used for model training, each data set represented by 768 numeric features from IndoBERT embedding. Meanwhile, `X_test` (2, 768) indicates that 2 text data sets are set aside as test data with the same number of features, thus being consistent with the training data. These results indicate that the model has a strong bias toward the High class (2), one sample originally labeled Low (0) and one originally labeled Medium (1) were both incorrectly predicted as High, while the four samples actually labeled High were all correctly predicted. Consequently, the model failed to recognize the Low and Medium classes in the analyzed samples and tended to default to the High class, as seen in Fig. 4.



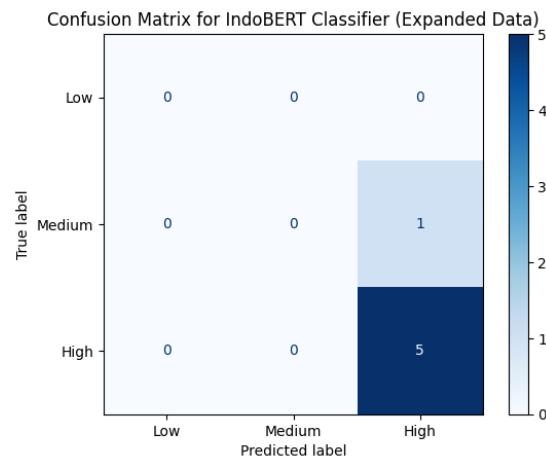


Figure 4. IndoBERT Classification Confusion Matrix

The expanded dataset was successfully categorized and labeled with the following availability statuses: corn: High, chicken eggs: High, rice: High, beef: Medium, broiler chicken: High, and fresh milk: High. Figure 4 shows IndoBERT's classification performance: Accuracy 0.8333 (83.33%), F1-Score 0.7576 (75.76%), Precision 0.6944 (69.44%), Recall 0.8333 (83.33%). The IndoBERT model successfully generated 768-dimensional contextual embeddings for each commodity.

The initial stage begins by loading the dataset into a DataFrame. Data cleaning is performed to ensure input quality. Empty values in the Province column are labeled Unknown, while empty values in numeric columns such as rice, corn, beef, broiler chicken, fresh milk, and others are filled using the average value of each column after converting the number format from text (changing commas to periods and removing non-numeric characters). Histogram of predicted vs. actual class distributions generated, as seen in Figure 5.

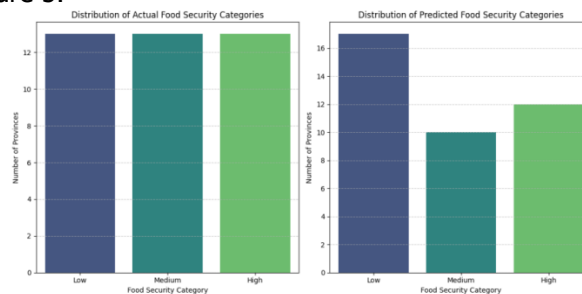


Figure 5. Histogram of Distributions Generated

The distribution of provinces by food security/availability category, divided into three classes: the Low category has the largest number of provinces (approximately 17 provinces), followed by the High category (approximately 12 provinces), and the Medium category has the smallest number (approximately 10 provinces). The data.info() output confirms that all columns now have a Non-Null Count equal to the total number of entries (39), indicating no remaining NaN values. The top five most important features are rice and corn (0.20), total food (0.19), rice (0.175), corn (0.15), and broiler chicken (0.108). The results of the dataset division after the normalization process using MinMaxScaler and the separation of training and test data. X_train_minmax of size (27, 10) indicates that there are 27 training data with 10 numeric features as input variables, while X_test_minmax of size (12, 10) indicates 12 test data with the same number of features, y_train_minmax of size (27,) and y_test_minmax of size (12,) indicate that each training and test data has one target label for each observation. This structure confirms that the train-test split process with a ratio of approximately 70% training data and 30% test data has been carried out consistently, and the data is ready to be used for training and evaluating classification models. Accuracy increased from 0.83 to 0.92. This

improvement is clearly visible in the classification report, where recall for class 1 (Medium) increased from 0.50 to 0.75 and its F1 score increased from 0.67 to 0.86. The improvement in model performance after tuning and scaling across all classes or only dominated by certain classes, also validates that the model has been able to distinguish each food security category accurately and equally. Confusion Matrix for Min-Max Scaled SVM Model generated and displayed, as seen in Figure 6.

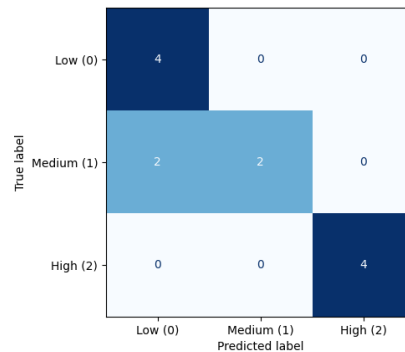


Figure 6. Confusion Matrix for Min-Max Scaled SVM Model

Confusion Matrix of SVM Min-Max Scaled Model shows shows Row 0 (Actual Class : Low): The model correctly predicted 4 instances as Low (0). There was no misclassification of this class. Row 1 (Actual Class : Medium): This is where the misclassification occurred. 2 instances that were actually Medium (2) were incorrectly predicted as Low (0). 3 instances that were actually Medium (1) were correctly predicted as Medium (1). Row 2 (Actual Class : High): The model correctly predicted 4 instances as High (2). Confusion Matrix of Newly Tuned Min-Max Scaled Model's shows Row 0 (Actual Class : Low): The model correctly predicted 4 instances as Low (0). There was no misclassification of this class. Row 1 (Actual Class : Medium). This is where the misclassification occurred. One instance that was actually Medium (1) was incorrectly predicted as Low (0). The model correctly predicted four instances as High (2), as seen in Table 4.

Table 4. SVM Modeling Result

Class	Min-Max Model SVM				Tuned Min-Max Model SVM			
	Precision	Recall	F1-Score	Support	Precision	Recall	F1-Score	Support
0	0.67	1.00	0.80	4	0.80	1.00	0.89	4
1	1.00	0.50	0.67	4	1.00	0.75	0.86	4
2	1.00	1.00	1.00	4	1.00	1.00	1.00	4
Accuracy	—	—	0.83	12	—	—	0.92	12
Macro Avg	0.89	0.83	0.82	12	0.93	0.92	0.92	12
Weighted Avg	0.89	0.83	0.82	12	0.93	0.92	0.92	12

A comparison of SVM modeling evaluations shows that the tuned model consistently improved its performance compared to the original model across almost all evaluation metrics. For class 0, the precision increased from 0.67 to 0.80, and the F1-score increased from 0.80 to 0.89, indicating more accurate predictions without compromising the ability to detect all data for that class (recall remained at 1.00). For class 1, Recall increased from 0.50 to 0.75, and the F1-score increased from 0.67 to 0.86, indicating that the tuned model significantly improved its ability to recognize the previously difficult-to-classify intermediate class. Class 2 performance remained excellent for both models, with Precision, Recall, and F1-score all at 1.00. Overall, model accuracy increased from 0.83 to 0.92, and macro and weighted average values also increased significantly, confirming that the tuning process successfully improved the model's ability to classify all classes more evenly and accurately, as seen in Table 5.

Table 5. SVM Modeling Evaluation

Class	Precision	Recall	F1-Score	Support
0	0.76	1.00	0.87	13

1	0.90	0.69	0.78	13
2	1.00	0.92	0.96	13
Accuracy	—	—	0.87	39
Macro Avg	0.89	0.87	0.87	39
Weighted Avg	0.89	0.87	0.87	39

The Confusion Matrix shows 13 'Low' instances were correctly predicted, 9 'Medium' instances were correctly predicted, but 4 'Medium' instances were incorrectly classified as 'Low'. 12 'High' instances were correctly predicted, but 1 'High' instance was incorrectly classified as 'Medium'. The prediction accuracy for the full min-max scale dataset is 87%. The SVM model trained with these optimal parameters on the min-max scale data achieved a test accuracy of 0.92 (92%). Comparing the new tuning for min-max scale data with previous results. Accuracy on the test set increased significantly from 0.83 (83%) to 0.92 (92%). Recall for class 1 ('Medium') increased from 0.50 (50%) to 0.75 (75%), and its F1 score increased from 0.67 (67%) to 0.86 (86%). The number of misclassifications for class 1 (Medium) decreased from 2 cases to 1. The Indobert model produced a Confusion Matrix that was robust to predicting the High class, as one sample with the actual label Low and one with Medium were both incorrectly classified as High. All four samples actually labeled High were correctly classified, as seen in Figure 7.

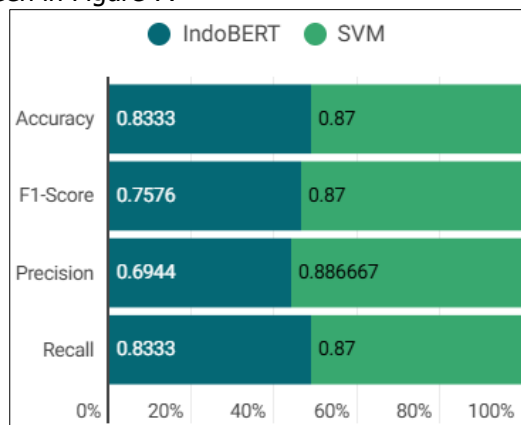


Figure 7. Comparison of IndoBERT and SVM Estimation Evaluation

Figure 8 shows a comparison of the performance of two classification models, IndoBERT and SVM. IndoBERT's accuracy was 0.8333 (83.33%), while SVM's was higher, 0.87 (87%), indicating SVM is more accurate overall. IndoBERT's F1-Score reached 0.7576, while SVM's was 0.87, indicating a better balance between precision and recall in SVM. IndoBERT's precision was 0.6944, while SVM's was much higher at 0.8867, indicating SVM's positive predictions were more accurate and had fewer misclassifications. IndoBERT's Recall was 0.8333, and SVM's was 0.87, indicating SVM was more capable of correctly capturing all positive data.

4. CONCLUSION

The SVM based on numerical data that has been normalized with Min-Max Scaling shows a more stable and balanced performance, with an increase in accuracy from 83% to 92%, as well as significant improvements in the Medium class (recall increased from 50% to 75% and F1-score from 67% to 86%), that SVM is able to learn the decision boundary between classes better and reduce the classification error of the middle class. The SVM model trained on normalized numerical data achieved more balanced and consistent classification performance across all food availability categories. Performance improvements were particularly evident in the medium availability class, indicating that feature scaling and hyperparameter tuning enhanced the model's ability to distinguish borderline cases. The IndoBERT-based text classification model achieved relatively high overall accuracy but showed a tendency to favor the high availability class. This bias can be attributed to the limited



dataset size and uneven label distribution, which constrained the model's ability to learn nuanced semantic differences across categories. While contextual embeddings provide rich semantic representations, their effectiveness is highly dependent on data volume and diversity.

5. REFERENCES

- [1] S. Kannan, "Transforming Community Engagement with Generative AI: Harnessing Machine Learning and Neural Networks for Hunger Alleviation and Global Food Security," *Cuest. Fisioter.*, vol. 54, no. 4, pp. 953–963, 2025.
- [2] K. N. Khikmah, B. Sartono, B. Susetyo, and G. A. Dito, "Performance Comparative Study of Machine Learning Classification Algorithms for Food Insecurity Experience by Households in West Java," *J. Online Inform.*, vol. 9, no. 1, pp. 128–137, 2024, doi: 10.15575/join.v9i1.1012.
- [3] A. Çalışkan, S. Demirhan, and R. Tekin, "Comparison of different machine learning methods for estimating compressive strength of mortars," *Constr. Build. Mater.*, vol. 335, no. 1, pp. 12–21, 2022, doi: 10.1016/j.conbuildmat.2022.127490.
- [4] K. Kumari, A. Mirzakhani Nafchi, S. Mirzaee, and A. Abdalla, "AI-Driven Future Farming: Achieving Climate-Smart and Sustainable Agriculture," *AgriEngineering*, vol. 7, no. 3, pp. 1–30, 2025, doi: 10.3390/agriengineering7030089.
- [5] W. R. Mgonezulu, M. M. N. Chitete, B. B. Maonga, and M. A. R. Phiri, "Machine learning approaches for grain seed quality assessment: a comparative study of maize seed samples in Malawi," *Discov. Appl. Sci.*, vol. 7, no. 6, 2025, doi: 10.1007/s42452-025-07191-y.
- [6] R. T. Al Zahrawi, S. N. S. Abdullah, A. Sarirete, and M. A. R. Abdullah, "Advancing Literary Analysis With Python: A Comprehensive Study of Simile Detection and Classification in the Translation of *Al-Abrat*," *SAGE Open*, vol. 15, no. 4, pp. 1–20, 2025, doi: 10.1177/21582440251378859.
- [7] B. Carneiro et al., "Text mining and machine learning reveal global determinants of food insecurity," *Sci. Rep.*, vol. 15, no. 1, pp. 1–19, 2025, doi: 10.1038/s41598-025-20670-x.
- [8] M. Meeske, F. Cruijssen, and C. Van Der Lee, *The role of language technology and artificial intelligence in food security policymaking*. 2026.
- [9] J. Arévalo-Royo, F. J. Flor-Montalvo, J. I. Latorre-Biel, R. Tino-Ramos, E. Martínez-Cámara, and J. Blanco-Fernández, "AI Algorithms in the Agrifood Industry: Application Potential in the Spanish Agrifood Context," *Appl. Sci.*, vol. 15, no. 4, pp. 1–37, 2025, doi: 10.3390/app15042096.
- [10] A. Rejeb, K. Rejeb, and A. Hassoun, "The impact of machine learning applications in agricultural supply chain: a topic modeling-based review," *Discov. Food*, vol. 5, no. 1, 2025, doi: 10.1007/s44187-025-00419-1.
- [11] G. Sabapathy, "Artificial Intelligence in Pharmaceutical Commercial Operations: Transforming Sales Performance and Patient Engagement in the Digital Era," *J. Comput. Sci. Technol. Stud.*, vol. 7, no. 8, pp. 636–651, 2025, doi: 10.32996/jcsts.
- [12] H. Resources and A. Officer, "ARTIFICIAL INTELLIGENCE FOR GLOBAL FOOD SECURITY: DATA-DRIVEN STRATEGIES FOR CLIMATE-RESILIENT AGRICULTURAL SYSTEMS *International Journal of Engineering Technology Research & Management*," *Int. J. Eng. Technol. Res. Manag.*, no. 12, 2019, [Online]. Available: file:///D:/RRC/Referenceing/Safety and Resilience/Ref/Sep-2019-02-1756779191-DEC201923.pdf.
- [13] A. S. R. M. Sinaga, R. E. Putra, and A. S. Girsang, "Prediction measuring local coffee production and marketing relationships coffee with big data analysis support," *Bull. Electr. Eng. Informatics*, vol. 11, no. 5, pp. 2764–2772, 2022, doi: 10.11591/eei.v11i5.4082.
- [14] C. P. Ng, T. Y. Tang, and A. Pariatamby, "How urgent environmental challenges and tech advancements reshape commercial food systems: A bibliometric analysis leveraging natural-language processing," *Int. J. Eng. Bus. Manag.*, vol. 18, pp. 1–23, 2026, doi: 10.1177/18479790251394768.
- [15] P. Bhairo and R. Maharaj, "Harnessing artificial intelligence for food security in a changing climate," *Discov. Food*, vol. 5, no. 1, 2025, doi: 10.1007/s44187-025-00609-x.

