SMATIKA : STIKI Informatika Jurnal
Accredited Sinta 4 Ministry of Higher Education, Science and Technology Republic of Indonesia SK No. 79/E/KPT/2023
E-ISSN: 2580-6939 || P-ISSN: 2087-0256

Sentiment analysis on public opinion trends on #MakanBergiziGratis programs on platform X using Long Short-Term Memory (LSTM) networks

Veny Dwi Wahyuningsih[1*], Yayak Kartika Sari[2], Agung Prasetya[3]

[1]*Universitas Bhinneka PGRI, Fakultas Sains dan Teknologi, Informatika, Jl. Mayor Sujadi No.7, Manggisan, Plosokandang, Kec. Kedungwaru, Kabupaten Tulungagung, Jawa Timur 66229, Indonesia*
[2,3] *Universitas Bhinneka PGRI, Fakultas Sains dan Teknologi, Informatika, Jl. Mayor Sujadi No.7, Manggisan, Plosokandang, Kec. Kedungwaru, Kabupaten Tulungagung, Jawa Timur 66229, Indonesia*

Keywords

BERT; Free Nutritious Meal Program; Long Short-Term Memory (LSTM); Sentiment Analysis; SMOTE

Corresponding Author:
dwiveny13@gmail.com

Abstract

The Free Nutritious Meal Program (Makan Bergizi Gratis/MBG) has sparked diverse public reactions on X (formerly Twitter), making it important to understand sentiment dynamics at scale. This study analyzes sentiment in 5,516 Indonesian-language tweets collected between January to November 2025 using a hybrid embedding of Bidirectional Encoder Representations from Transformers (BERT) with a Long Short-Term Memory (LSTM) model. Initial labeling using a lexicon-based approach reveals an imbalanced distribution (positive 34.34%, neutral 60.80%, negative 4.86%). To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) is used to build a synthetic representation for minority classes in the embedding space. This approach helps improve class balance while maintaining the overall data representation. The proposed model achieves good performance, with 82.98% accuracy, 83.27% precision, 82.98% recall, and an F1-score of 83.01%. Cross-validation indicates relatively consistent performance, with accuracy ranging from 80.51% to 84.41%. Despite these results, the negative class remains challenging (F1-score 0.57), highlighting the impact of linguistic complexity, including implicit and nuanced expressions in social media text. Overall, the findings suggest that integrating BERT embeddings with LSTM and feature-level SMOTE can be a suitable approach for handling imbalanced sentiment classification tasks. However, further improvements, particularly in advanced transformer fine-tuning and deeper linguistic modeling, are needed to better capture subtle sentiment patterns.

1. Introduction

The rapid growth of social media has transformed it into a dynamic platform for expressing public opinions on social issues and government policies. In Indonesia, social media usage reached approximately 139 million users in 2024, representing nearly 50% of the population [1]. Among various platforms, X (formerly Twitter) plays a significant role as a medium for real-time public discourse through short text-based interactions and hashtag-driven discussions.

One topic that is frequently discussed on platform X is the #MakanBergiziGratis (MBG) program, a strategic initiative introduced by the Indonesian government to improve nutritional intake and reduce stunting rates

[2]. According to data from Badan Gizi Nasional and Kementerian Kesehatan Republik Indonesia, the MBG program has reached 36.773.520 beneficiaries as of October 2025 [3].

The program has generated diverse responses from the public, ranging from support to criticism, reflecting various perspectives on its implementation, effectiveness, and sustainability. This is reflected in the collected data, where a number of posts exhibit negative sentiment highlighting various issues during the period from January to November 2025 [4]. This diversity of opinions indicates the importance of applying a systematic and data-driven approach to comprehensively analyze public sentiment.

Sentiment analysis is a popular method for categorizing opinions in textual data. Although traditional machine learning techniques like Naïve Bayes, Support Vector Machine (SVM), and Logistic Regression have demonstrated encouraging outcomes, their ability to capture contextual relationships within text is restricted [5]. Deep learning techniques like Long Short-Term Memory (LSTM), which can represent long-term sequential dependencies in textual data, have been developed to overcome these constraints [6]. A number of previous studies have reported that LSTMs perform well in a variety of text classification tasks. Research by [7] showed that LSTM achieved an accuracy of 85.3% in the classification of Halodoc application reviews, exceeding Support Vector Machine (SVM) with an accuracy of 78.6% and Naïve Bayes with an accuracy of 74.1%. Similar results are shown by a study [8] which reported that LSTM was able to analyze Indonesian texts with an accuracy of 90.05% and an F1-score value of 98.66%, but SVM only achieved roughly 85%. However, the use of LSTM in sentiment analysis is the main focus of this work rather than method comparison.

Additionally, Transformer-based models have shown good performance in generating contextual word embeddings, such as Bidirectional Encoder Representations from Transformers (BERT). This model is suitable for sentiment classification tasks on social media data because it can capture semantic context and sequential patterns thanks to the combination of BERT and LSTM.

Despite these advancements, several challenges remain in sentiment analysis, particularly when dealing with imbalanced datasets and complex linguistic expressions. Social media data often exhibit class imbalance, where certain sentiment classes dominate others, potentially biasing the model toward majority classes. Moreover, minority classes especially negative sentiment often involves more complex, implicit, or sarcastic expressions, making them harder to classify accurately.

Therefore, this study uses an LSTM approach with BERT-based embedding to analyze public opinion sentiment regarding the #MakanBergiziGratis program. To mitigate data imbalance, the Synthetic Minority Oversampling Technique (SMOTE) was applied to the training data. Apart from evaluating the efficacy of the suggested model, this study also examines how language complexity and data imbalance affect classification results, especially for minority classes. It is hoped that the findings of this study will provide insights into public opinion patterns and assist in the development of more robust sentiment analysis models for Indonesian social media data. This study contributes by applying SMOTE in the BERT embedding space and demonstrating that linguistic complexity plays a more critical role than class imbalance in sentiment classification.

2. Research Methods

This study proposes a hybrid sentiment classification framework integrating BERT embeddings and LSTM, with SMOTE applied in the embedding space to address class imbalance. This approach was chosen because it can handle sequential text data processing and identify long-term dependencies, making it effective for sentiment classification tasks [7]. The research stages include the process of data collection, preprocessing, labeling, data splitting text representation using BERT, classification using the LSTM model, and model evaluation. All of these stages are presented in detail in Figure 1.

The data collection stage of this study began with a web scraping technique to obtain data from the X platform with the help of Node.js-based tweet-harvest software. The text is then cleaned and normalized during the data preprocessing stage. The sentiments were then divided into three groups: positive, negative, and neutral. After labeling, the data was divided into training and testing components.

Text representations are obtained through tokenization and embedding processes using pre-trained BERT models, resulting in contextual vector representations. In this study, In the sentiment classification process, BERT is used as a feature extractor to generate embeddings, which are then fed into an LSTM model. The model's predicted output is then evaluated using metrics such as accuracy, precision, recall, F1-score, and confusion matrix.

2.1 Data Collection

The initial step in this research is the data collection process. Data is collected from platform X, which was selected due to its open and publicly accessible conversations, and is often used by users to write, deliver and send opinions on public policy issues. Data retrieval was conducted from platform X through web scraping techniques with the help of Node.js-based tweet-harvest software as well as authentication using auth tokens. Data collection was carried out using #MakanBergiziGratis keywords in the period from January 1, 2025 to November 30, 2025 and divided into several time intervals for each month. The data collected is in the form of text uploads that include public comments or responses to the program. The data obtained is then stored in CSV format, then all files from the same interval are combined into one monthly dataset. Furthermore, a data cleaning and selection process is carried out which includes the removal of duplicate data, the filtering of tweets that contain advertisements or promotions, the deletion of foreign-language tweets, and the elimination of tweets that are not in accordance with the topic of the Free Nutritious Meal Program which is then processed at the analysis stage.

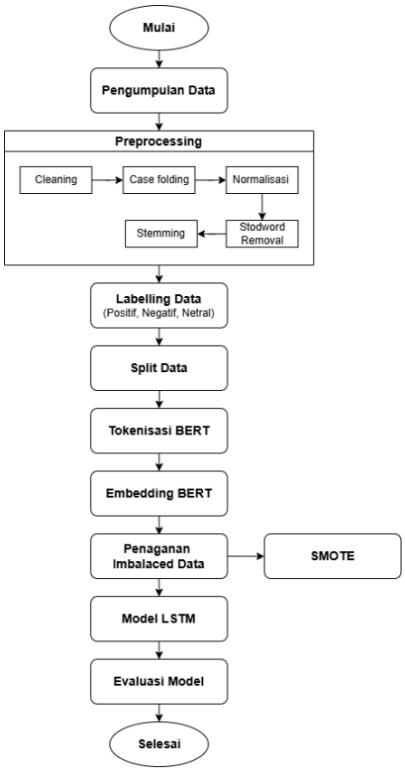

Figure 1. Research Model Flow

2.2 Preprocessing Data

In the early stages of implementing text mining, the preprocessing process is carried out to produce clean and structured data [9]. This process involves cleaning the data of unnecessary characters such as URLs, mentions, hashtags, numbers, punctuation, and special symbols. Then, case folding is performed, converting all text to lowercase. The normalization stage is used to replace slang words or non-standard words with standard words according to the Big Indonesian Dictionary (KBBI). Next, the stopword removal stage removes conjunctions

and common terms such as in, which, and not. Stemming is the final step that involves removing suffixes from words and returning them to their basic form.

2.3 Labelling Data

Sentiment labeling in this study was conducted using a lexicon-based approach by matching each word in a tweet with a predefined sentiment dictionary. The lexicon used in this study was adapted from publicly available Indonesian sentiment lexicons and further refined manually to better capture the characteristics of informal language commonly found in social media text, including slang and colloquial expressions. A sentiment score indicating positive, negative, or neutral polarity is assigned to each word in the lexicon. A tweet's sentiment score is determined by summing the scores of each word in the text. Tweets are categorized into three groups based on their final score: neutral (score = 0), negative (score < 0), and positive (score > 0) [10]. Although this approach enables efficient labeling of large datasets, it has limitations in capturing contextual meanings such as sarcasm, irony, and implicit sentiment, which may introduce noise in the labeling process. This limitation is considered in the analysis and discussion of the model's performance.

2.4 Split Data

The dataset was split into training and testing sets at an 80:20 ratio, consisting of 4,412 training samples and 1,104 testing samples. To ensure that the class distribution remains consistent across both subsets, a stratified sampling approach was applied to preserve the balance of sentiment class distribution in each dataset [11]. Stratified sampling is particularly important in this study due to the imbalanced class distribution, where the neutral class dominates the dataset. This method helps reduce bias in model evaluation and provides a more accurate assessment of model performance by maintaining the percentage of each class in both the training and test sets.

2.5 Tokenization of BERT

The tokenization process using the BERT (Bidirectional Encoder Representations from Transformers) tokenization method is used to generate context-based token representations. Furthermore, the embedding generated by BERT is used as a word vector representation which then becomes an input for the LSTM model to carry out the sentiment classification process. Tokenization uses the WordPiece approach, allowing the breakdown of words into subwords so that they are able to handle uncommon words and morphological variations. To simplify the process of classifying and separating text fragments, tokenization in BERT involves adding special tokens such as [CLS] and [SEP]. To ensure uniform input length, padding and truncation are used. This approach assists BERT in dealing with rare words, morphological variations, and typos without eliminating the structure of important information in the text [12].

2.6 Embedding BERT

BERT produces contextualized embeddings that are two-way contextual, so that they are able to represent the meaning of words based on the context of sentences more accurately than static embedding. Tokenized tokens are given vector representations through a pre-trained BERT model, where each token will have a fixed-dimensional representation (e.g. 768 for the base model). These representations can then be used as inputs for the LSTM model to classify sentiment [13].

2.7 Handling Imbalanced Data

The performance of sentiment classification models is significantly affected by imbalanced class distributions, where certain classes dominate over others. This situation can cause the model to be biased towards the majority class, thus impairing its ability to detect the minority class. This problem also exists in Long Short-Term Memory (LSTM) models, where class dominance can affect learning and reduce the model's ability to generalize well [14]. To address this problem, this study utilizes training data to create synthetic samples for the minority class using the Synthetic Minority Over-sampling Technique (SMOTE). Unlike direct oversampling techniques, SMOTE improves class balance and increases the diversity of the training data by interpolating between existing minority class samples to generate new examples [15]. This approach aims to improve the model's ability to identify representative patterns of underrepresented classes while reducing bias towards the majority class.

In this study, SMOTE is applied after the text representation stage, where textual data has been transformed into numerical vectors using BERT embeddings. By operating in the embedding space, SMOTE generates synthetic samples based on high-dimensional semantic representations rather than raw textual input. This result, patterns in the minority class in the feature space can be captured more effectively by the model. However, since SMOTE operates on numerical vectors, it does not fully preserve the linguistic structure and contextual meaning of the original text. As a result, the generated samples may not always reflect natural language expressions and can introduce noise, particularly in complex cases such as sarcasm or implicit sentiment. Despite these shortcomings, Using SMOTE in the embedding space improves the class balance and increases the model's sensitivity to the minority class.

2.8 Model Long Short-Term Memory (LSTM)

The sentiment classification model in this study uses a Long Short-Term Memory (LSTM) network to overcome the long-term memory problem in sequential data and avoid the vanishing gradient problem [16]. The LSTM model is chosen for the sentiment classification task including contextual relatedness because of its ability to identify sequential dependencies in text data. To generate three sentiment classifications (negative, neutral, positive), the model structure is constructed with a combination of embedding layers, LSTM layers, dropouts, and dense layers that use softmax activation. This model was trained using the Adam optimizer and the categorical cross-entropy loss function, and the softmax activation function was used to calculate the probability of each sentiment class. The formula for the softmax function is described as follows:

$$P(y = i \mid x) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{1}$$

Description:
a. $P(y = i \mid x)$: Probability of data x belongs to class i
b. z_i: Model Output Score for Class i
c. K: Number of classes

2.9 Model Evaluation

The model's capacity to generalize data under different circumstances was tested using 5-fold cross validation [17]. The classification performance of the test data was analyzed using a confusion matrix. This analysis is reinforced by the standard evaluation metrics in sentiment analysis, namely F1-score, accuracy, precision, and recall. Here is a breakdown of the evaluation metrics used:

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{2}$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{3}$$

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

$$F1\text{-Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

Model evaluation using 5-fold cross-validation was performed on the test data to assess model stability and generalization. At this stage, model performance was evaluated using weighted F1 score and accuracy metrics.

3. Result and Discussions

3.1 Data Collection

The web scraping technique was applied to collect research data from public uploads on platform X that contained opinions about the Free Nutritious Meal Program. Data capture is done using Node.js-based tweet-harvest with token authentication. The data collection period lasted from January 1 to November 30, 2025, by applying the keywords "Free Nutritious Meals", Indonesian filters (*lang:id*) and time restriction parameters

(since and until) that were applied on a monthly basis. The scraping process resulted in 5,571 tweets. After the data cleansing stage was carried out to eliminate duplication based on the amount of data to 5,516 tweets which were then applied in the preprocessing process and sentiment analysis was implemented with the Long Short-Term Memory (LSTM) technique.

3.2 Preprocessing Data

In this stage, the data undergoes preprocessing, including cleaning, casefolding, normalization, stopword removal, and stemming, to standardize the text prior to classification, as presented in Table 1.

Table 1. Results of the preprocessing stage

Tweet	*Cleaning*	*Casefolding*	Normalisasi	*Stopword Removal*	*Stemming*
@ardisatriawan Bullshit. Gaada namanya makan bergizi gratis	Bullshit Gaada namanya makan bergizi gratis	bullshit namanya makan bergizi gratis	omong kosong tidak ada namanya makan bergizi gratis	['omong', 'kosong', 'tidak', 'namanya', 'makan', 'bergizi', 'gratis']	omong kosong tidak nama makan gizi gratis

After the preprocessing stage is thoroughly carried out, cleaner, structured, and representative data is produced. The data is ready to be used at the sentiment analysis and classification modeling stages to achieve a more optimal level of accuracy.

3.3 Labelling Data

The sentiment labeling stage is carried out after the data goes through the preprocessing process. The data labeling process applies a lexicon-based approach, where data is grouped into three classes, namely positive, negative, and neutral. The word positive represents support or a good response to the Free Nutritious Meal program, while the word negative represents criticism or rejection, while the word neutral is informative without a particular opinion bias. In the labeling process, sentiment classification is not solely determined by the presence of words in the lexicon but also considers the overall semantic context of the sentence. For sentences containing contrastive conjunctions such as *"but", "however",* or *"although"*, the sentiment label is assigned based on the dominant clause that conveys the main opinion. Additionally, sentences that are descriptive and do not contain explicit evaluative expressions are categorized as neutral. The goal of this method is to ensure that the labeling results correspond to the actual meaning and are logically consistent in the text.

Figure 2 presents the distribution of sentiment labels based on the labeling results of 5,516 tweets, showing that neutral sentiment dominates with 3,354 tweets (60.80%), followed by positive sentiment with 1,894 tweets (34.34%), and negative sentiment with 268 tweets (4.86%).

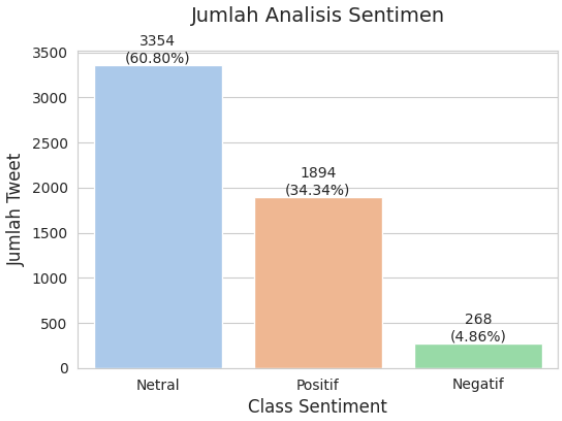

Figure 2. Sentiment Label Distribution

The sentiment distribution indicates a strong dominance of neutral sentiment, which suggests that public discourse surrounding the MBG program is largely informational rather than evaluative. This condition may affect the classification process, particularly in distinguishing between neutral and positive sentiments that often share similar linguistic patterns. Furthermore, the limited proportion of negative sentiment introduces class imbalance issues, which can reduce the model's sensitivity in detecting minority class patterns. Examples of tweets for each sentiment class are presented in Table 2.

Table 2. Tweets Based on Labelling Results

Tweet	***Sentiment Score***	***Sentiment Label***
program makan bergizi gratis berdampak besar terhadap perkembangan generasi muda indonesia	1	Positive
edukasi dan kesehatan setelah makan bergizi gratis	0	Neutral
program makan bergizi gratis yg jdi prioritas utama aja bentukannya gak jelas aplagi yg hanya prioritas pendukung	-1	Negative

3.4 Split Data

The dataset was divided into training and testing data with a ratio of 80:20 after preprocessing and labeling. A total of 4,412 (79.99%) of the total 5,516 datasets were used for training, while 1,104 (20.01%) were used for testing. This data division was carried out to maximize the model's ability to analyze data patterns from the training data, while the testing data was used to assess model performance. Figure 3 displays the distribution of data divisions.

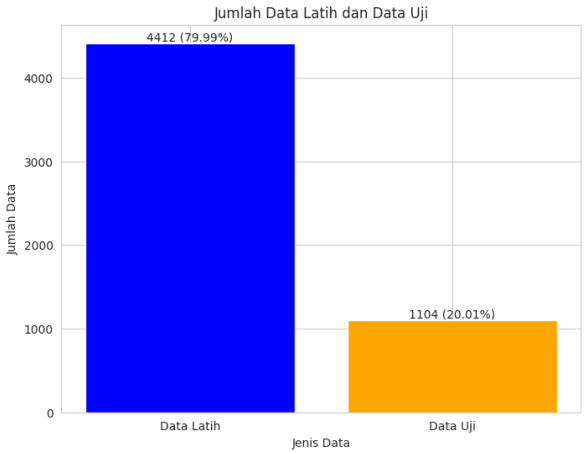

Figure 3. Split Data Distribution

3.5 Tokenization of BERT

Tokenization in this study was carried out using BERT by applying the WordPiece method, which is a technique of breaking words into subwords to overcome vocabulary limitations (out-of-vocabulary). Based on the tokenization results of the example sentences, representations in the form of input_ids, attention masks, and token lists are obtained. Each sequence begins with a special token [CLS] (ID 101) that serves as an aggregate representation of the sentence for classification purposes, and ends with [SEP] (ID 102) as the end marker of the sequence. The [PAD] token (ID 0) is added until it reaches the maximum predetermined length so that all data has uniform dimensions when processed in batches. The value on the attention mask indicates the model's processed token (1) and the ignored padding token (0).

Some words are broken down into subwords. For example, "republic" is segmented into rep and ##ublik, while "indonesia" is segmented into indo and ##nesia. The ## sign indicates that the token is a continuation of the previous subword. Through this process, the model gains a more flexible and contextual understanding of the morphological structure of words. Table 3 displays the BERT tokenization results.

Table 3. BERT Tokenization Results

No	*Token*	*Input ID*	*Attention Mask*
1	[CLS]	101	1
2	Rep	76456	1
3	##ublik	77186	1
4	Indo	74502	1
5	##nesia	66887	1
....	....	....	....
31	[SEP]	102	1
32	[PAD]	0	0

Overall, the tokenization results show that the WordPiece method is capable of representing text in structured numerical form with a consistent sequence length.

3.6 Embedding BERT

The embedding of BERT in this study resulted in tensors of sizes (4412, 64, 768) for the training data and (1104, 64, 768) for the test data. The first number indicates the number of samples, the second number represents the maximum length of the sequence (64 tokens), and the third number is the embedding vector dimension (768) in the BERT-Base model. Token embedding, segment embedding, and position embedding are the three

primary components that are added to each token to produce its representation, this is to provide a two-way contextual representation that can capture the relationship between words in a sentence. In text classification tasks, vectors on the [CLS] token are generally used as representations of the entire sentence. Thus, the final dimensions used as a classification feature become (4412, 768) for the trained data and (1104, 768) for the testing data.

3.7 Handling Imbalanced Data

The distribution of the research dataset is dominated by the neutral class in contrast to the positive and negative classes, indicating significant class imbalance. Following the BERT embedding procedure, which converts textual data into high dimensional numeric vectors, SMOTE was applied to the training data to address this issue. This allows SMOTE to operate in the feature space rather than on raw text. A comparative visualization of the data distribution before and after SMOTE is shown in Figure 4.

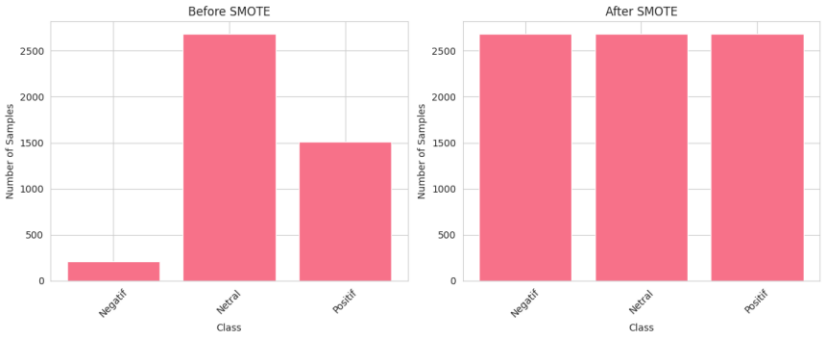

Figure 4. Comparison of Label Distribution Before and After SMOTE

Before applying SMOTE, the class distribution is highly imbalanced, with approximately 2,700 neutral samples, 1,500 positive samples, and only 200 negative samples. After applying SMOTE with a k-nearest neighbors parameter of 5, the distribution becomes balanced, with each class consisting of approximately 2,700 samples. The model performs better when SMOTE is applied in the embedding space, especially when it comes to identifying minority class patterns. The model attains an accuracy of roughly 83%, which is reflected in the total performance gain. However, despite the balanced distribution, the performance of the negative class remains lower compared to the neutral and positive classes.

This limitation indicates that although SMOTE successfully addresses the issue of class imbalance in terms of quantity, it does not fully capture the semantic complexity of minority class expressions. Since SMOTE generates synthetic samples through interpolation between embedding vectors, the resulting representations may not always correspond to naturally occurring linguistic patterns. As a result, complex expressions such as sarcasm, irony, or implicit sentiment remain difficult for the model to classify accurately.

3.8 Model Long Short-Term Memory (LSTM)

This work employed a Long Short-Term Memory (LSTM) model with BERT-based text embedding to categorize sentiment into three groups: neutral, negative, and positive. A single LSTM layer with 128 units makes up the model architecture, which is capable of identifying both short-term and long-term dependencies in textual input. A Dropout layer is used before the output layer and after the LSTM to lessen overfitting. The recorded features are processed by a dense layer with 64 neurons and ReLU activation before being classified by the softmax output layer. 5-Fold Cross-Validation, 10–20 epochs, a batch size of 32, the Adam optimizer, categorical cross-entropy loss, and a learning rate of 0.001 were used to train the model. The model has 467,715 trainable parameters overall.

Based on the architectural illustration (Figure), the processing flow shows that BERT embeddings provide rich contextual representations before being passed to the LSTM. The LSTM then extracts important sequential patterns from the text, while the combination of Dropout and Dense layers functions as a regularization mechanism and feature enhancement. This configuration enables the model to balance complexity and generalization. Analytically, this architecture supports stable classification performance in cross-validation and explains the model's ability to capture

linguistic context, although limitations remain in distinguishing classes with imbalanced data distributions, particularly in the negative sentiment class.

Model: "sequential_6"

Layer (type)	Output Shape	Param #
lstm_6 (LSTM)	(None, 128)	459,264
dropout_12 (Dropout)	(None, 128)	0
dense_12 (Dense)	(None, 64)	8,256
dropout_13 (Dropout)	(None, 64)	0
dense_13 (Dense)	(None, 3)	195

Total params: 467,715 (1.78 MB)
Trainable params: 467,715 (1.78 MB)
Non-trainable params: 0 (0.00 B)

Figure 5. LSTM Model Architecture

3.9 Model Evaluation

This study evaluates sentiment classification models using performance criteria such as accuracy, precision, recall, F1-score, and loss using the Stratified 5-Fold Cross Validation method. This evaluation method provides a more accurate assessment of model stability and generalization ability to new data by ensuring that the class distribution is maintained across each fold.

Based on the evaluation findings shown in Table 4, the model accuracy values ranged from 80.51% to 84.41%. This relatively small range indicates consistent model performance across various data subsets. Fold 4 had the lowest performance (80.51%), while Fold 2 had the highest performance (84.41%). This indicates that the differences are within a reasonable range, despite the data variation, indicating that the model is not overfitting.

Table 4. Evaluation Results per Fold

Fold	*Accuracy* (%)	*Precision* (%)	*Recall* (%)	*F1-Score* (%)	*Loss*
1	84.15	84.21	84.15	84.18	0.4307
2	84.41	84.86	84.41	84.58	0.4495
3	83.95	84.43	83.95	83.97	0.4350
4	80.51	80.87	80.51	80.60	0.4866
5	81.87	82.00	81.87	81.72	0.4648

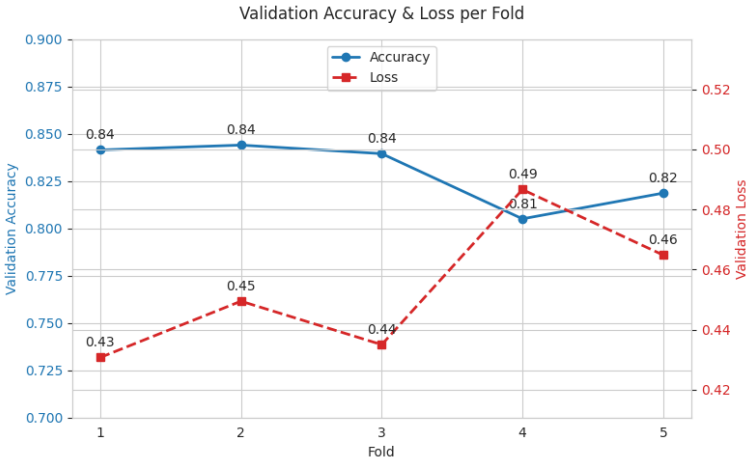

Figure 6. 5-Fold Cross Validation Results

Based on Figure 6, with validation accuracy ranging from 0.81 to 0.84, the cross-validation results show that the model performs reasonably consistently across all folds. The highest accuracy is observed in fold 2 (0.84), while the lowest occurs in fold 4 (0.81). Although a decrease appears in fold 4, the performance improves again in fold 5, indicating that the variation is not systematic but rather influenced by the characteristics of each data subset.

In terms of loss, the model shows moderate fluctuations, with the lowest loss in fold 1 (0.43) and the highest in fold 4 (0.49). The increase in loss in fold 4 is consistent with the drop in accuracy, suggesting that the model encounters more complex or less representative patterns in that particular fold. However, the relatively small gap between folds indicates that the model maintains good learning stability. Overall, the model does not exhibit substantial overfitting, as evidenced by the limited variation in both accuracy and loss. The use of SMOTE in the embedding space, It supports this stability by balancing the training data and enabling the model to learn more representative features across various sentiment classes.

Furthermore, the average evaluation results of the model are presented in Table 5.

Table 5. Average Results of Model Evaluation

Model	*Accuracy* (%)	*Precision* (%)	*Recall* (%)	*F1-Score* (%)	*Mean Loss*
LSTM + BERT	82.98	83.27	82.98	83.01	0.4533

Based on the research results, the LSTM + BERT model achieved an average accuracy of 82.98% by balancing the precision, recall, and F1-score values. This demonstrates that the model can accurately categorize sentiment data and perform consistently across various evaluation metrics. The relatively balanced metric values also suggest that the model does not overly favor a particular class during prediction. This condition is supported by the application of SMOTE after BERT embedding, it strengthens the model's capacity to identify minority class patterns and increases class balance at the feature level.

The final evaluation based on the classification report in Table 6 shows that the model achieves an overall accuracy of 83%.

Table 6. Classification Report

Class	*Precision* (%)	*Recall* (%)	*F1-Score* (%)	*Support*
Negative	0.54	0.62	0.57	268
Neutral	0.86	0.87	0.86	3354
Positive	0.83	0.78	0.81	1894
Accuracy	-	-	0.83	5516
Macro Avg	0.74	0.76	0.75	5516
Weighted Avg	0.83	0.83	0.83	5516

The results showed that, with an F1 score of 0.86, the neutral class performed better than the positive class, which had an F1 score of 0.81, according to the data. However, the negative class performed the worst, with an F1 score of 0.57. Although SMOTE has been applied to balance the dataset in the embedding space, the performance gap between classes still exists. This suggests that data balancing alone is not sufficient to fully address the challenges of sentiment classification, particularly for the negative class. The difference between the weighted average (0.83) and the macro average (0.75) further emphasizes that the dominance of the neutral class still impacts model performance. This suggests that despite the improvements made by SMOTE, the model generally performs better on majority patterns.

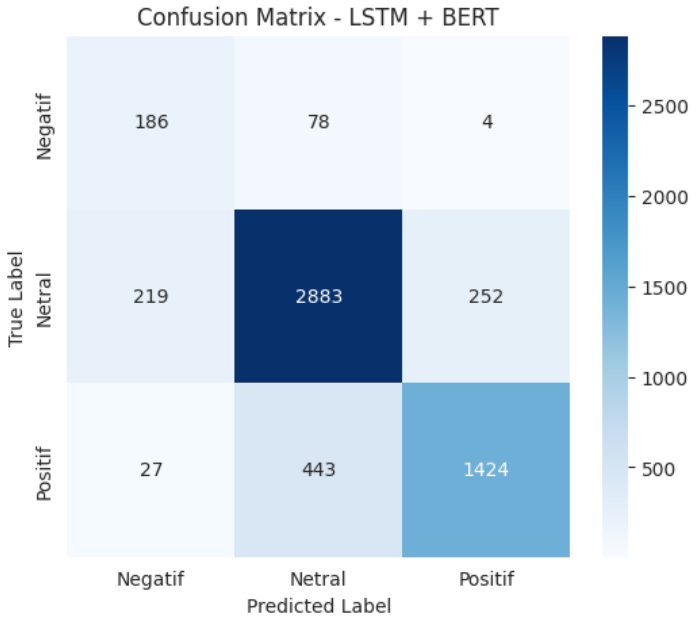

Figure 7. Confusion Matrix

Based on Figure 7, according to the confusion matrix, the model with the most accurate predictions in the neutral class was 2,883, compared to the positive class of 1,424 and negative 186. Misclassifications are still observed, particularly where neutral data is predicted as positive (252) and negative (219), indicating overlapping linguistic characteristics between classes. In the negative class, the model shows the weakest performance, with several instances misclassified as neutral, reflecting the complexity and ambiguity of negative sentiment expressions. In the positive class, although the performance is relatively strong, a number of instances (443) are misclassified as neutral. This suggests that some positive sentiments are expressed implicitly, making them harder to distinguish from neutral expressions.

Overall, Figure 7 this shows how effectively the model can capture dominant patterns, especially in the neutral class, but still has difficulty distinguishing more complex and ambiguous sentiments, especially in the negative class. Although the SMOTE technique has been applied to address data imbalance, the results indicate that balancing the data distribution alone is not sufficient to overcome the complexity of patterns in the negative class. This suggests that the main challenge lies not only in the quantity of data but also in the diversity and linguistic characteristics of negative sentiment.

On the other hand, the combination of BERT and LSTM enhances the model's understanding of semantic context and sequential relationships in the text, as demonstrated by the model's excellent performance on the neutral class and somewhat decent performance on the positive class. However, further approaches are still needed, such as exploring more specific semantic features or applying context-based data augmentation, to improve accuracy in the negative class.

3.10 Discussions

This study shows that the LSTM model combined with BERT-based embeddings achieves stable and relatively strong performance, with an average accuracy of 82.98% across 5-fold cross-validation. This result indicates that the integration of contextual representation and sequential modeling is effective in capturing both semantic and structural patterns in social media text. This result is in line with recent research demonstrating that transformer-based embeddings, like BERT, can boost feature representation in hybrid deep learning models and dynamically capture contextual meaning, hence improving sentiment classification accuracy [18].

The predominance of the neutral sentiment class (60.80%) is one of the study's main conclusions, indicating that public discourse tends to be more informational rather than strongly opinionated. From a modeling perspective, this class imbalance can lead to bias toward the majority class, which remains a common challenge in sentiment classification tasks [19]. To address this problem, SMOTE was applied after the BERT embedding process, meaning that oversampling was performed in the feature space (embedding space). This approach enables the generation of synthetic samples based on semantically rich representations, thereby improving class balance and supporting more stable model learning.

However, the results show that the negative sentiment class remains the most difficult to classify, even after applying SMOTE. This suggests that the challenge in sentiment classification is not solely caused by data imbalance but also by the inherent linguistic complexity of social media text. Negative sentiment is often expressed implicitly through sarcasm, irony, or indirect language, making it challenging for the model to pick up discriminative and consistent patterns. This finding is supported by recent studies highlighting that complex linguistic expressions in social media require deeper contextual understanding and advanced modeling approaches [20].

Furthermore, although SMOTE improves minority class representation, it has limitations in capturing the semantic diversity of natural language. This is because SMOTE generates synthetic data through interpolation between vectors in the embedding space, which may not fully reflect realistic linguistic variations. As a result, the generated samples may lack the richness and variability found in real-world data. This limitation is reflected in the confusion matrix, where misclassifications frequently occur between the neutral class and both positive and negative classes, indicating overlapping linguistic features and ambiguity in sentiment expression.

Overall, the results of this study support the idea that linguistic complexity and class distribution both influence sentiment classification performance. While the combination of BERT, SMOTE, and LSTM provides a robust approach for handling imbalanced data, further improvements are needed. To capture more subtle nuances of sentiment, especially in minority classes, future research could explore more sophisticated methods such as context-aware data augmentation or transformer-based model refinement.

4. Conclusion

This study analyzed public sentiment toward the #MakanBergiziGratis program using an LSTM model with BERT-based embeddings on 5,516 tweets. The results indicate a highly imbalanced dataset, with neutral sentiment dominating. With an average accuracy of 82.98% across cross-validations, the suggested model shows stable performance.

The application of SMOTE in the embedding (feature) space improved minority class representation and overall model stability. However, the negative class remained the most difficult to classify, indicating that challenges in sentiment classification are not solely due to data imbalance but also linguistic complexity, such as implicit and nuanced expressions in social media text.

Overall, the combination of BERT and LSTM is effective for sentiment analysis. However, improving performance particularly for minority classes requires not only data balancing but also enhanced modeling of complex linguistic patterns. Future work should focus on advanced approaches such as transformer fine-tuning and improved data quality to better capture sentiment nuances.

5. References

[1] DataReportal, “Digital 2024: Indonesia.” [Online]. Available: https://datareportal.com/reports/digital-2024-indonesia

[2] A. Atikah Merlinda and Y. Yusuf, “Analisis Program Makan Gratis Prabowo Subianto Terhadap Strategi Peningkatan Motivasi Belajar Siswa di Sekolah Tinjauan dari Perspektif Sosiologi Pendidikan,” *Ranah Res. J. Multidiscip. Res. Dev.*, vol. 7, no. 2, pp. 1364–1373, 2025, doi: https://doi.org/10.38035/rrj.v7i2.1360.

[3] K. R. I. Kesehatan, "Kemenkes Tegaskan Keamanan Pangan sebagai Kunci Keberhasilan Program Makan Bergizi Gratis," *Kementerian Kesehatan Republik Indonesia.*, 2025. [Online]. Available: https://kemkes.go.id/id/kemenkes-tegaskan-keamanan-pangan-sebagai-kunci-keberhasilan-program-makan-bergizi-gratis?utm_source=chatgpt.com

[4] B. Suwastoyo, "Dilema Program Makan Siang Gratis, Antara Manfaat dan Tantangan," 2024, [Online]. Available: https://www.cips-indonesia.org/post/dilema-program-makan-siang-gratis-antara-manfaat-dan-tantangan?lang=id

[5] B. Siswoyo, N. Azka, and P. Utomo, "Pemanfaatan Machine learning untuk Klasifikasi Sentimen Pelanggan pada Media Sosial," vol. 1, no. 1, pp. 29–34, 2025.

[6] D. Wahyuni, N. Fadhillah, W. Ariestya, and U. Gunadarma, "Metode Long Short-Term Memory dan Lexicon Based Untuk Analisis Sentimen Aplikasi tiktok," vol. 23, pp. 173–180, 2024.

[7] R. Refianti, A. B. Mutiara, and R. A. Putra, "A Lexicon-Based Long Short-Term Memory (LSTM) Model for Sentiment Analysis to Classify Halodoc Application Reviews on Google Playstore," *J. Appl. Data Sci.*, vol. 5, no. 1, pp. 146–157, 2024, doi: 10.47738/jads.v5i1.160.

[8] A. Prasetya, Y. K. Sari, J. Iskandar, and M. K. Ansor, "Identifikasi Jenis Operasi Data Manipulation Language Berbasis Bilstm Pada Kalimat Berbahasa Indonesia," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 4, pp. 2552–2557, 2024, doi: 10.29100/jipi.v9i4.8695.

[9] S. Chohan, A. Nugroho, A. Maezar, B. Aji, and W. Gata, "Analisis Sentimen Aplikasi Duolingo Menggunakan Metode Naïve Bayes dan Synthetic Minority Over Sampling Technique," vol. 22, no. 2, 2020, doi: https://doi.org/10.31294/p.v21i2.

[10] Miftakhul Rahman, Mantri Kromo Fandith Fili, and Wardianto, "Analisis Hasil Rekapitulasi Pilkada Daerah Khusus Jakarta (DKJ) 2024 Menggunakan Metode Support Vector Machine," *JICode J. Inform. dan Komput.*, vol. 2, no. 1, pp. 100–111, 2025, doi: 10.30599/c0zqdw84.

[11] S. G. Wijaya, "Analisis Sentimen Pengguna Twitter Terhadap Kebijakan Royalti Restoran dan Kafe Dengan Multinomial Naive Bayes," vol. 9, pp. 49–58, 2026, doi: https://doi.org/10.36080/idealis.v9i1.3698.

[12] L. B. Hutama and D. Suhartono, "Indonesian Hoax News Classification with Multilingual Transformer Model and BERTopic," vol. 46, pp. 81–90, 2022, doi: https://doi.org/10.31449/inf.v46i8.4336.

[13] N. Mushtaq, G. Ali, D. Muhammad, K. Malik, and A. Bukhari, "BERT applications in natural language processing : a review," 2025, doi: https://doi.org/10.1007/s10462-025-11162-5.

[14] R. Efendi, T. Wahyono, and I. R. Widiasari, "DBSCAN SMOTE LSTM : Effective Strategies for Distributed Denial of Service Detection in Imbalanced Network Environments," 2024, doi: https://doi.org/10.3390/bdcc8090118.

[15] C. Sintiya, G. H. Hutagaol, D. Bate, and S. Irviantina, "Evaluasi Teknik Resampling untuk Class Balancing dalam Analisis Sentimen Kesehatan Mental Berbasis Bi-LSTM," vol. 26, no. 2, pp. 257–274, 2025, doi: https://doi.org/10.55601/jsm.v26i2.1799.

[16] K. S. Nugroho *et al.*, "Deteksi Depresi dan Kecemasan Pengguna Twitter Menggunakan Bidirectional LSTM," no. Ciastech, pp. 287–296, 2021, doi: https://doi.org/10.48550/arXiv.2301.04521.

[17] E. Salim and A. Solichin, "Analisis Sentimen Pada Media Sosial Twitter Terhadap Pelayanan Dinas Kependudukan dan Pencatatan Sipil Menggunakan Algoritma Naive Bayes," vol. 5, pp. 79–86, 2022, doi: https://doi.org/10.36080/idealis.v5i2.2961.

[18] C. H. Lin and U. Nuha, "Sentiment analysis of Indonesian datasets based on a hybrid deep - learning strategy," *J. Big Data*, vol. 10, p. 88, 2023, doi: 10.1186/s40537-023-00782-9.

[19] J. Rahman, A. Riaz, P. Malakar, and M. Kabir, "Recent advancements and challenges of NLP-based

sentiment analysis : A state-of-the-art review," *Nat. Lang. Process. J.*, vol. 6, no. January, p. 100059, 2024, doi: 10.1016/j.nlp.2024.100059.

[20] K. Jia, F. Meng, J. Liang, and P. Gong, "Text sentiment analysis based on BERT-CBLBGA," *Comput. Electr. Eng.*, vol. 112, p. 109019, 2023, doi: https://doi.org/10.1016/j.compeleceng.2023.109019.