ISSN 2407-4322
E- ISSN 2503-2933


# Analysis Of Public Sentiment Towards The Free Nutritious Meal Program In Schools Based On Tweets Using The K-Nearest Neighbors Method

**Aiga Rizki Maharani*[1], Rendra Gustriansyah [2], Muhammad Hafiz Irfani [3]**
[1,2] University Indo Global Mandiri; Jl. Jend. Sudirman Km.4 No. 62, 20 Ilir D. IV, Kec. Ilir Tim. I, Kota Palembang, Sumatera Selatan 30129, telp. 0811-7322-710
[3] Information Technology, Computer Science and Science, University Indo Global Mandiri
e-mail: *[1] 2021110024@students.uigm.ac.id , [2] rendra@uigm.ac.id , [3] m.haviz@uigm.ac.id

***Abstract***

*The public sentiment analysis of the free nutritious meal program in schools was conducted based on data from the social media platform Twitter (X). This program is an initiative by the Indonesian government aimed at improving the nutritional quality of children, particularly those from underprivileged families, as well as reducing stunting rates. The data used consisted of 3,007 tweets that had undergone preprocessing, manual labeling, and class balancing using oversampling techniques. The K-Nearest Neighbors (K-NN) method was applied to classify sentiment into three categories: positive, negative, and neutral. The data was split with 80% used for training and 20% for testing. The analysis process included data representation using TF-IDF and model evaluation using metrics such as accuracy, precision, recall, and F1-score. Evaluation results showed that the K-NN model with K=3 achieved an accuracy of 82%, with the best performance in classifying negative sentiment tweets (recall = 1.00, F1-score = 0.93). These findings indicate that public opinion toward the program tends to be negative, mainly due to concerns over budget allocation and food distribution. This study is expected to provide input for the government in designing more effective and responsive communication strategies and public policies.*

***Keywords***— Sentiment Analysis, Twitter, Free Nutritious Meal Program, *K-Nearest Neighbors* , Stunting

## 1. INTRODUCTION

Child nutrition is one of the strategic issues that is a major concern in social and educational policy in Indonesia. Unmet nutritional needs have a direct impact on health, cognitive development, and the quality of human resources in the future. One indicator of nutritional problems is the high rate of stunting, which remains a national challenge to this day. To address this, the government has initiated a free nutritious meal program in schools aimed at improving children's nutrient intake, particularly for those from low-income families, while also boosting students' motivation to learn [1]. According to a report [2], this program is designed to have a significant impact on reducing stunting rates. However, Indonesia remains one of the countries with the highest stunting prevalence in Southeast Asia. Data [3] shows that Indonesia's stunting rate (21%) remains higher than Malaysia's (17%) and Thailand's (14%), which have successfully reduced stunting prevalence through integrated nutrition policies and equitable food distribution. Although Indonesia's stunting rate decreased from 24% in 2021 to 21% in 2024, this figure still exceeds the WHO's ideal threshold of below 20% [4].

To achieve the stunting reduction target, a multidimensional approach involving the health, education, and economic sectors, as well as active community participation, is required [5]. In line with this, the President and Vice President of Indonesia have made the free nutritious meals program a key strategy for stunting prevention and part of the Indonesia's Golden Generation 2045 vision [6]. The success of this program heavily depends on meeting nutritional standards, ensuring equitable food distribution, and securing sustainable funding [7]. In its implementation, the program has sparked a variety of public responses, both positive and negative. Social media, particularly Twitter (now X), has become one of the main platforms for the public to express opinions and criticism. Based on a study analyzing over 9,000 tweets, it was found that negative sentiment dominated with over 8,000 tweets, while positive sentiment accounted for only around 430 tweets [7]. This negative sentiment is generally related to public concerns about the program's budget, estimated to reach Rp450 trillion per year, as well as challenges in food distribution that remain uneven [8].

Although public opinion about this program is abundant on social media, data-based studies that systematically measure public sentiment are still limited. This is important because poorly managed perceptions can affect the acceptance and success of policy implementation [9]. Therefore, sentiment analysis of public opinion is one effective method for understanding how the public views the program. Previous studies have utilized machine learning algorithms to analyze public sentiment toward national policies. For example, [10] compared the K-Nearest Neighbors (KNN) and Naïve Bayes algorithms to analyze public sentiment regarding the relocation of the National Capital (IKN). The results showed that KNN had the highest accuracy of 88.12%, making it superior to Naïve Bayes. [11] used KNN to classify public sentiment regarding the use of e-commerce in Indonesia and achieved an accuracy of 82%. Another study by [12] examined public comments on the free lunch program uploaded through the YouTube channel “tempodotco” using the KNN method and SMOTE technique. The results showed that negative sentiment dominated, with the best performance achieved at K=4 with an accuracy of 76%.

Based on a review of these studies, there are no studies that specifically utilize Twitter data to analyze public sentiment toward free nutritious school meal programs. In addition, previous studies generally focus on other platforms such as YouTube or different policies. Thus, there is a research gap that can be filled by this study. This study is novel because it examines public sentiment toward free nutritious school meal programs using Twitter data, which has not been widely done, and applies the K-Nearest Neighbors (KNN) algorithm for sentiment classification based on tweets. It is hoped that this study will produce a simple yet accurate classification model.

The purpose of this study is to analyze public sentiment toward free nutritious school meal programs using Twitter data and the KNN algorithm. This study is expected to contribute academically by adding to the literature on the use of machine learning algorithms in social media sentiment analysis, as well as providing practical contributions in the form of an objective picture of public perception that can serve as a basis for consideration in formulating more effective communication strategies and public policies. The limitations of this study include the use of only tweets relevant to the free nutritious school meal program, thus excluding opinions from other social media platforms. Additionally, the analysis was conducted using only the KNN method without comparing it to other algorithms.

## 2. METHODS

This study was conducted through systematic stages to produce valid results. The background of this study is based on the pros and cons of the community regarding the free nutritious meals program in schools, which has been widely discussed on social media, particularly Twitter (X). Observations were made on public conversations on Twitter during January–April 2025, when this issue became a national concern. The research process included collecting tweet data and evaluating sentiment classification to obtain an objective representation of public opinion. The complete research stages are shown in Figure 1

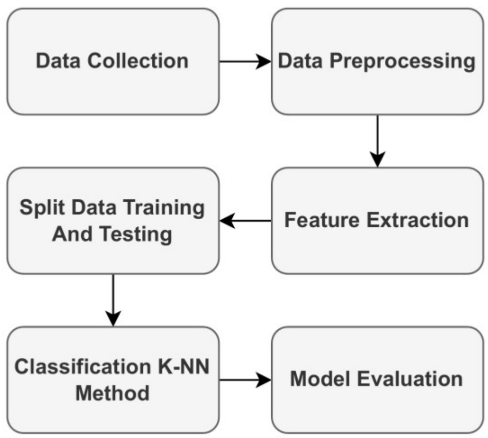


Fig. 1 Research Stages

### *2.1 Data Collection*

The research data was sourced from Twitter (X), which was chosen due to its high level of public interaction and open data access. Data collection was carried out using the Instant Data Scraper extension to retrieve tweets based on keywords such as “free nutritious meals” and variations thereof. The data was then exported to CSV format and processed in Google Colab using Python with libraries such as pandas, nltk, scikit-learn, and matplotlib.

### *2.2 Pre-processing Test Data*

Pre-processing is an important step in text-based data analysis, especially for data from Twitter, which generally contains noise, irrelevant information, and unstructured formats. This process aims to clean and normalize data so that it is ready for use in sentiment analysis and classification. In this study, preprocessing was performed on the collected tweets, including text cleaning, duplicate removal, and word normalization. The implementation was carried out using Python on Google Colab, utilizing libraries such as re (regular expressions) for special character removal, nltk for tokenization, and Sastrawi for Indonesian language stemming.

### *2.3 Sentiment Labeling*

After going through the pre-processing stage, the next step is manual sentiment labeling, which forms the basis for building the K-Nearest Neighbors (K-NN) model, as this method requires labeled training data. The labeling process is carried out by analyzing each tweet based on the meaning of the words and the context of the sentence, then classifying it into three sentiment categories: positive, negative, or neutral. Label determination is based on keywords and contextual interpretation to ensure accuracy. This labeled data is then used as training data for the K-NN model, enabling the model to automatically label new tweets by comparing their similarity to previously labeled data.

*2.4 Data Balancing (Random Oversampling)*

Manual labeling results show an unbalanced distribution of sentiment data, with more neutral and positive tweets than negative ones. This imbalance can introduce bias into the classification model, particularly in recognizing minority classes. To address this, the Random Oversampling technique is used to double the samples of the minority class until its proportion is equal to that of the other classes. With balanced data, the K-Nearest Neighbors (K-NN) model can be trained more fairly, thereby improving accuracy and the ability to recognize negative sentiment, which was previously limited.

*2.5 Feature Extraction (Text Representation)*

In text classification, raw data cannot be directly used by machine learning algorithms, so it must be converted into a numerical representation. One popular method is Term Frequency–Inverse Document Frequency (TF-IDF), which assigns weights to words based on their frequency of occurrence in a document and their rarity across the entire corpus. In this study, feature extraction was performed using TF-IDF with Python on the Google Colaboratory platform. This representation converts documents into fixed-dimensional vectors that indicate the importance of words for use in the K-Nearest Neighbors (K-NN) classification algorithm. TF-IDF weights are calculated using the following three components:

a) Term Frequency (TF) Measures the frequency of occurrence of term t in document d:

$$TF(t,d) = \frac{f_{t,d}}{\sum_k f_{t,d}} \quad (1)$$

Where :
$f_{t,d}$ : number of words $t$ in document *d*
$\sum_k f_{t,d}$ : total number of words in the document *d*

b) Inverse Document Frequency (IDF) Measures the rarity of words in the entire document:

$$IDF(t,d) = log_{10}\left(\frac{N}{df(t)}\right) \quad (2)$$

Where :
$N$ : total number of documents
$df(t)$ : number of documents containing word $t$

c) TF-IDF Combine both values to calculate word weight:

$$TF - IDF(t,d) = TF(t,d) \times IDF(t) \quad (3)$$

This TF-IDF representation is an important basis in the text classification process, as it helps machine learning algorithms recognize patterns and distinguish document characteristics based on meaningful word features.

*2.6 Training and Testing Data Distribution*

After the pre-processing and labeling stages are complete, the data is then divided into two parts, namely training data to build the model and test data to measure its performance. Some of the data is set aside as test data to evaluate the model's performance on data that has not been studied before. The division is done randomly to ensure that the distribution of each class remains balanced. To see the effect of the division ratio on the performance of the K-NN model, experiments were conducted with several variations in the test data ratio, namely 10%, 20%, 30%, 40%, and 50%. The results of these variations were used to determine the best configuration for producing classification accuracy.

*2.7 KNN Classification*

This study uses the K-Nearest Neighbors (K-NN) method to classify public sentiment toward the Free Nutritious Meals Program in schools based on collected tweet data. K-NN is a supervised learning algorithm that classifies new data by referring to its proximity to labeled training data. In the context of this study, the proximity between tweets is measured using Cosine Similarity, which calculates the similarity between two vectors based on the angle between them. The cosine similarity value ranges from -1 to 1, where a value close to 1 indicates a high degree of similarity between two texts. The higher the value, the more similar the content of the tweets.

All data that has gone through the pre-processing stage is then converted into vector representations using the Term Frequency–Inverse Document Frequency (TF-IDF) technique. TF-IDF is used to highlight words that have a high level of importance and reduce the weight of common words, so that the data representation becomes more informative. This representation allows the K-NN algorithm to measure the similarity between tweets more accurately.

Selecting the optimal K value is an important step in the K-NN method because it affects the classification results. The K value indicates the number of nearest neighbors that are used as a reference in determining the class of a data point. If K is too small, the model risks overfitting, while if it is too large, underfitting may occur. Therefore, this study tests K values in the range of 1 to 10. Each K value is evaluated using the 5-fold cross-validation technique with accuracy metrics. This process is carried out using cosine distance as a measure of proximity between data, as it is considered more effective in capturing similarities in text-based content.

*2.8 Model Evaluation*

The model was evaluated to measure the extent of the classification model's performance in identifying public sentiment toward free nutritious meal programs based on tweet data. This evaluation is an important step to ensure that the model not only performs well on training data but is also capable of generalizing to new data that has not been studied before. In this study, the evaluation was conducted using a confusion matrix to identify the number of correct and incorrect predictions for each sentiment class. Additionally, several performance metrics such as accuracy, precision, recall, and F1-score were calculated, reflecting the model's accuracy and completeness in classification.

In binary classification, the confusion matrix is a 2x2 matrix consisting of four main components, namely True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) [13]. However, since this study uses a multi-class approach with three sentiment categories—positive, negative, and neutral—the confusion matrix is expanded to a 3x3 matrix to represent all sentiment classes. An illustration of the confusion matrix for sentiment classification can be seen in Table 2.

Table 1 Confusion Matrix

| Prediction Class / Current Classes | Positive | Negative | Neutral |
|---|---|---|---|
| Positive | TP (Positive) | FN (Negative) | FN (Neutral) |
| Negative | TP (Positive) | FN (Negative) | FN (Neutral) |
| Neutral | TP (Positive) | FN (Negative) | TN (Neutral) |

- True Positive (TP): The number of tweets classified by the model as positive sentiment and which actually match their original positive label.
- False Positive (FP): The number of tweets predicted as positive sentiment by the model, but which actually have a negative or neutral label (also known as (Type I Error).
- False Negative (FN): The number of tweets that should be labeled as positive, but are classified as negative or neutral by the model (Type II Error).
- True Negative (TN): The number of tweets that are correctly predicted as not being positive sentiment negative or neutral according to their original label.

As an illustration, suppose a tweet expresses support for a free nutritious school meal program and is categorized as positive sentiment. If the model predicts the tweet as positive, then this is a TP. Conversely, if a neutral tweet is classified as positive, then this becomes an FP, and so on. From this confusion matrix, various evaluation metrics can be calculated to measure the performance of the K-NN model's sentiment classification in more detail, namely:

a) Accuracy indicates the proportion of correct predictions (positive, negative, or neutral) compared to the total number of predictions. In the case of sentiment analysis, accuracy helps to understand the overall performance of the model. However, for unbalanced data, this metric may be less representative [14].

$$accuracy = \frac{TP+T}{TP+TN+FP+FN} \quad (4)$$

b) Precision measures how accurate the model is in predicting a particular sentiment (for example, how many tweets predicted to be positive are actually positive). This metric is particularly important when FP errors need to be minimized [15].

$$precision = \frac{TP}{TP+FP} \quad (5)$$

c) Recall shows how well the model finds all tweets that actually belong to a particular sentiment class (for example, positive). This is especially important when FN errors are considered more serious [16].

ISSN 2407-4322
E- ISSN 2503-2933


$$recall = \frac{TP}{TP+FN} \quad (6)$$

d) The F1-Score shows the harmonic mean between precision and recall, which provides a balanced assessment of model performance, especially when the distribution of data between classes is uneven. This metric is very useful in sentiment analysis, as it emphasizes the importance of balancing prediction accuracy and the model's ability to recognize all classes [17].

$$f1 - score = 2\,x\,\frac{precision\;x\;recall}{precision + r} \quad (7)$$

## 3. RESULTS AND DISCUSSION

This section presents the results of data processing and analysis that have been carried out, accompanied by a discussion linking the research findings to previous theories and research. The results presented include sentiment data distribution, evaluation of the performance of the K-Nearest Neighbors (K-NN) classification model, and analysis of accuracy and other performance metrics. The discussion is conducted to interpret these findings, identify emerging patterns, and explain their implications for the research and the context of the free nutritious school meal program policy.

### *3.1 Data Collection*

This research dataset comes from Twitter (X) and focuses on tweets related to the free nutritious school meal program as an initiative of the Indonesian government. Data was collected using the Instant Data Scraper extension with keywords such as "free nutritious meals" and variations thereof, during the period January–April 2025 to ensure up-to-date opinions. This process yielded 3,007 tweets that were used in training and testing the sentiment classification model. Figure 2 shows an example of raw data from the scraping process.

| no | created_at | author_id | text |
|---|---|---|---|
| 1 | 2025-04-23 13:20:19+00:00 | 17206651002108371 | Program #MakanBergiziGratis ini sangat membantu anak-anak sekolah yang kurang mampu. Semoga berlanjut terus! |
| 2 | 2025-04-23 13:17:53+00:00 | 10301048837165257 | Salut untuk pemerintah! Program makan gratis di sekolah bikin anak-anak lebih semangat belajar. 👏 |
| 3 | 2025-04-23 13:01:31+00:00 | 18129942547280098 | Katanya gratis tapi di sekolah anak saya tetap diminta bawa bekal gimana sih programnya nggak merata. |
| 4 | ... | .... | .... |
| 2999 | 2025-04-23 13:23:26+00:00 | 13111869789954212 | Program makan bergizi gratis bakal menghabiskan anggaran Rp 71 triliun tahun ini. Siapa para penikmatnya? 🤷 |
| 3000 | 2025-04-23 13:23:25+00:00 | 26967018856123110 | Ketua DPR RI Puan Maharani mendukung sikap Presiden Prabowo Subianto yang akan mengusut dugaan penggelapan dana program Makan Bergizi Gratis (MBG) oleh mitra Badan Gizi Nasional (BGN) |

Fig. 2 Raw Data from Twitter Scraping

### *3.2 Pre-processing Text Data*

After collecting 3,007 tweets, the next step is pre-processing to clean the data of irrelevant elements such as links, symbols, and special characters. This process is done to normalize the text so that it is ready to be processed by classification algorithms such as K-Nearest Neighbors (K-NN). All stages are implemented using Python in Google Colaboratory with the re, nltk, Sastrawi, and pandas libraries. The stages are as follows:

a) Cleaning

At this stage, irrelevant characters such as links/URLs (example https://t.co/xyz), mentions (example @username), hashtags, numbers, punctuation marks, and emoji symbols are eliminated. The aim is to remove "noise" from the text so that it focuses more on meaningful words that form the core of the user's opinion. An example of the cleaning results is presented in Table 2.

Table 2 Cleaning Result

| No | After Cleaning |
|---|---|
| 1 | Program MakanBergiziGratis ini sangat membantu anak-anak sekolah yang kurang mampu Semoga berlanjut terus |
| 2 | Salut untuk pemerintah Program makan gratis di sekolah bikin anakanak lebih semangat belajar |
| 3 | Katanya gratis tapi di sekolah anak saya tetap diminta bawa bekal gimana sih programnya nggak merata |
| 4 | ……. |
| 2999 | Program makan bergizi gratis bakal menghabiskan anggaran Rp triliun tahun ini Siapa para penikmatnya |
| 3000 | Ketua DPR RI Puan Maharani mendukung sikap Presiden Prabowo Subianto yang akan mengusut dugaan penggelapan dana program Makan Bergizi Gratis MBG oleh mitra Badan Gizi Nasional BGN |

*b) Lowercasing*

At this stage, all characters in the text are converted to lowercase so that the model does not treat identical words differently simply because of differences in capitalization. For example, the words "Program" and "program" will be considered the same after this process. An example of the results of the lowercasing process is shown in Table 3.

Table 3 Lowercasing Results

| No | After Lowercasing |
|---|---|
| 1 | program makanbergizigratis ini sangat membantu anak anak sekolah yang kurang mampu semoga berlanjut terus |
| 2 | salut untuk pemerintah program makan gratis di sekolah bikin anak anak lebih semangat belajar |
| 3 | katanya gratis tapi di sekolah anak saya tetap diminta bawa bekal gimana sih programnya nggak merata |
| 4 | ……. |
| 2999 | program makan bergizi gratis bakal menghabiskan anggaran rp triliun tahun ini siapa para penikmatnya |
| 3000 | ketua dpr ri puan maharani mendukung sikap presiden prabowo pubianto yang akan mengusut dugaan penggelapan dana program makan bergizi gratis mbg oleh mitra badan gizi nasional bgn |

c) Tokenization

The tokenization process aims to break down text into smaller pieces of words, known as tokens. This stage is very helpful in the feature extraction process because each word can be analyzed individually. The tokenization function in this study was implemented using simple Python code, namely by utilizing the *text.split()* method to separate words based on spaces. An example of the tokenization results is shown in Table 4

Table 4 Tokenization Results

| No | After Tokenization |
|---|---|
| 1 | ['program', 'makan', 'bergizi', 'gratis', 'ini', 'sangat', 'membantu', 'anak', 'anak', 'sekolah', 'yang', 'kurang', 'mampu', 'semoga', 'berlanjut', 'terus'] |
| 2 | ['salut', 'untuk', 'pemerintah', 'program', 'makan', 'gratis', 'di', 'sekolah', 'bikin', 'anak', 'anak', 'lebih', 'semangat', 'belajar'] |
| 3 | ['katanya', 'gratis', 'tapi', 'di', 'sekolah', 'anak', 'saya', 'tetap', 'diminta', 'bawa', 'bekal', 'gimana', 'sih', 'programnya', 'nggak', 'merata'] |
| 4 | ……. |
| 2999 | ['program', 'makan', 'bergizi', 'gratis', 'bakal', 'menghabiskan', 'anggaran', 'rp', 'triliun', 'tahun', 'ini', 'siapa', 'para', 'penikmatnya'] |
| 3000 | ['ketua', 'dpr', 'ri', 'puan', 'maharani', 'mendukung', 'sikap', 'presiden', 'prabowo', 'subianto', 'yang', 'akan', 'mengusut', 'dugaan', 'penggelapan', 'dana', 'program', 'makan', 'bergizi', 'gratis', 'mbg', 'oleh', 'mitra', 'badan', 'gizi', 'nasional', 'bgn'] |

.

d) Stopword Removal

Stopwords are common words such as “di”, “yang”, “dan”, “dari”, “untuk” which often appear in sentences but have no significant meaning in the context of sentiment analysis. Therefore, these words are removed so that only words that contribute significantly to the meaning of the sentence remain. Examples of stopword removal results are shown in Table 5.

Table 5 Stopword Removal Results

| No | After Stopword Removal |
|---|---|
| 1 | ['program', 'makan', 'bergizi', 'gratis', 'sangat', 'membantu', 'anak', 'anak', 'sekolah', 'kurang', 'mampu', 'semoga', 'berlanjut', 'terus'] |
| 2 | ['salut', 'pemerintah', 'program', 'makan', 'gratis', 'sekolah', 'bikin', 'anak', 'anak', 'semangat', 'belajar'] |
| 3 | ['katanya', 'gratis', 'sekolah', 'anak', 'saya', 'tetap', 'diminta', 'bawa', 'bekal', 'gimana', 'programnya', 'nggak', 'merata'] |
| 4 | ……. |
| 2999 | ['program', 'makan', 'bergizi', 'gratis', 'bakal', 'menghabiskan', 'anggaran', 'rp', 'triliun', 'tahun', 'siapa', 'penikmatnya'] |
| 3000 | ['ketua', 'dpr', 'ri', 'puan', 'maharani', 'mendukung', 'sikap', 'presiden', 'prabowo', 'subianto', 'mengusut', 'dugaan', 'penggelapan', 'dana', 'program', 'makan', 'bergizi', 'gratis', 'mbg', 'mitra', 'badan', 'gizi', 'nasional', 'bgn'] |

e) Stemming

Stemming is the process of transforming words into their base form or root word. For example, the words "membantu", "dibantu", and 'bantuan' will all be changed to "bantu." The goal is to ensure that words with the same meaning are not counted as different words by the system. The stemming process is performed using the Sastrawi library, a Python library specifically developed to support natural language processing in Indonesian. Examples of stemming results are shown in Table 6.

Table 6 Stemming Results

| No | After Stemming |
|---|---|
| 1 | ['program', 'makan', 'gizi', 'gratis', 'sangat', 'bantu', 'anak', 'anak', 'sekolah', 'kurang', 'mampu', 'semoga', 'lanjut', 'terus'] |
| 2 | ['salut', 'pemerintah', 'program', 'makan', 'gratis', 'sekolah', 'bikin', 'anak', 'anak', 'semangat', 'ajar'] |
| 3 | ['kata', 'gratis', 'sekolah', 'anak', 'saya', 'tetap', 'minta', 'bawa', 'bekal', 'gimana', 'program', 'nggak', 'rata'] |
| 4 | ……. |
| 2999 | ['program', 'makan', 'gizi', 'gratis', 'bakal', 'habis', 'anggar', 'rp', 'triliun', 'tahun', 'siapa', 'nikmat'] |
| 3000 | ['ketua', 'dpr', 'ri', 'puan', 'maharani', 'dukung', 'sikap', 'presiden', 'prabowo', 'subianto', 'usut', 'duga', 'gelap', 'dana', 'program', 'makan', 'gizi', 'gratis', 'mbg', 'mitra', 'badan', 'gizi', 'nasional', 'bgn'] |

f) Sentiment Labeling

After pre-processing is complete, the next step is sentiment labeling to group each tweet into positive, negative, or neutral categories. This step is important because the K-Nearest Neighbors (K-NN) method used is supervised learning and requires labeled data. As an initial step, the first 500 tweets were manually labeled based on the meaning and context of the sentences, then used to train the K-NN model. This model then automatically classified the remaining 2,507 tweets, successfully labeling all 3,007 tweets. After the automatic predictions were completed, a review was conducted to ensure all data had labels.

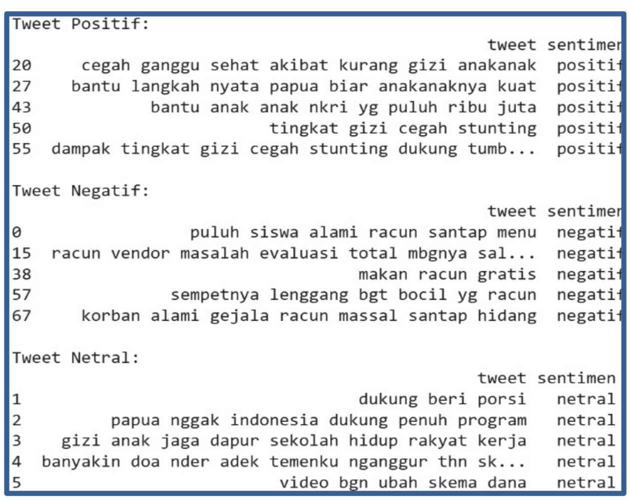


Fig. 3 Example of Tweet Sentiment Classification Results into Three Categories

Figure 3 shows an example of tweets grouped into three sentiments. Positive tweets generally contain support for nutrition improvement or stunting prevention programs, negative tweets contain criticism related to program implementation or specific incidents, while neutral tweets are informative without showing clear emotional sentiment. This example is provided to clarify the characteristics of each class and demonstrate the model's ability to recognize language context functionally.

g) Data Balancing

After all tweets were successfully labeled, a data distribution check was performed on each sentiment class. The results showed a significant imbalance, with the neutral and positive classes each having more than 1,400 tweets, while the negative class had only about 150 tweets in Figure 4.

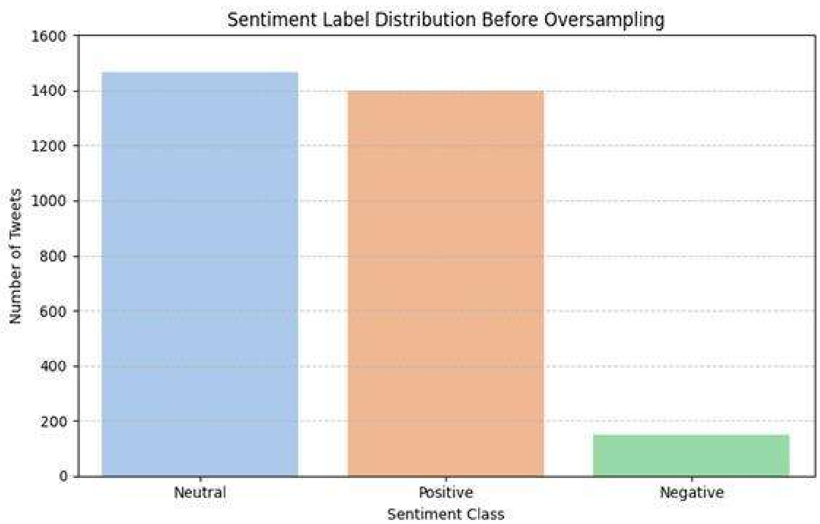


Fig. 4 Sentiment Label Distribution Before Oversampling

This imbalance has the potential to make the model more inclined to recognize the majority class and reduce accuracy for the minority class. To overcome this, the Random Oversampling technique was used, implemented through the imblearn.over_sampling.RandomOverSampler library. This method doubles the minority class data until the distribution is balanced. After the oversampling process, each class has 1,465 tweets, bringing the total data to 4,395 tweets, as shown in Figure 5. This balancing is expected to improve the generalization ability of the K-NN model across all three classes without bias toward any single class.

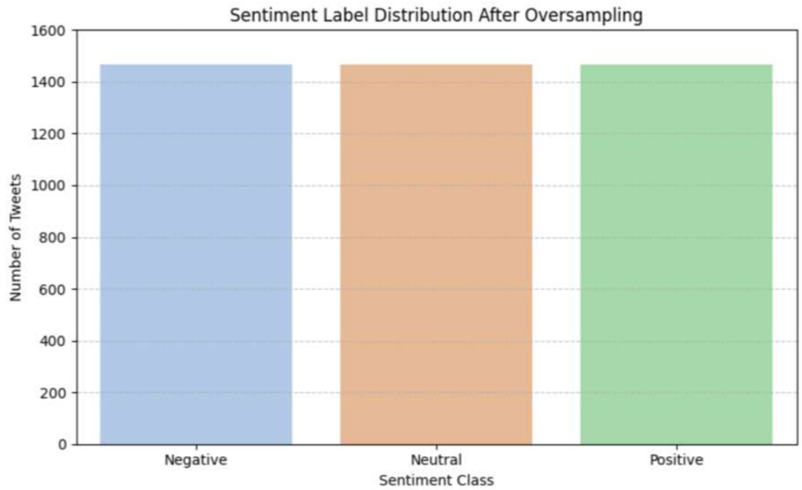

Fig. 5 Sentiment Label Distribution After Oversampling

h) Feature Extraction (Text Representation)

At this stage, feature extraction is performed using the Term Frequency-Inverse Document Frequency (TF-IDF) method to convert the pre-processed text data into numerical representations. This method assigns weights to each word based on its frequency in a document relative to the entire corpus, enabling its use by the K-Nearest Neighbors (K-NN) algorithm. The vectorization process yields TF-IDF numerical weights for each word. To understand the distribution and contribution of words, a heatmap visualization is created, as shown in Figure 6.

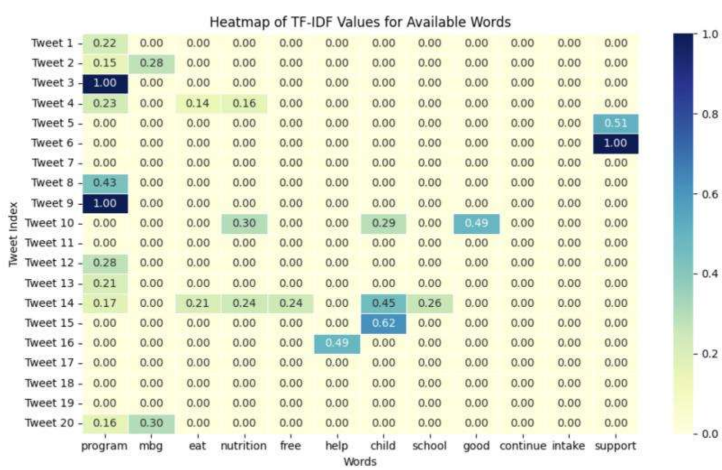

Fig. 6 TF-IDF Value Heatmap

This image shows relevant words such as “program,” “mbg,” “eat,” “nutrition,” and “free,” with colors representing TF-IDF weights. A value of 0.00 indicates that the word does not appear in a particular tweet, while a value close to 1.0 indicates that the word is dominant. For example, the word “program” has the highest weight (1.00) in the 2nd and 9th indexed tweets. This process ensures that the text data is successfully converted into a numerical form ready for sentiment classification using K-NN.

i) Training and Testing Data Distribution

After the data is represented in vector form using the TF-IDF method, the next step is to divide the dataset into training data and test data to evaluate the model's performance on data it has not been trained on. This division is performed using the train_test_split function from the Scikit-learn library. In addition to the main proportion of 80% training data and 20% test data, experiments were also conducted with variations in test size of 10%, 20%, 30%, 40%, and 50% to see their effect on model accuracy. The evaluation results for each proportion are presented in Figure 7.

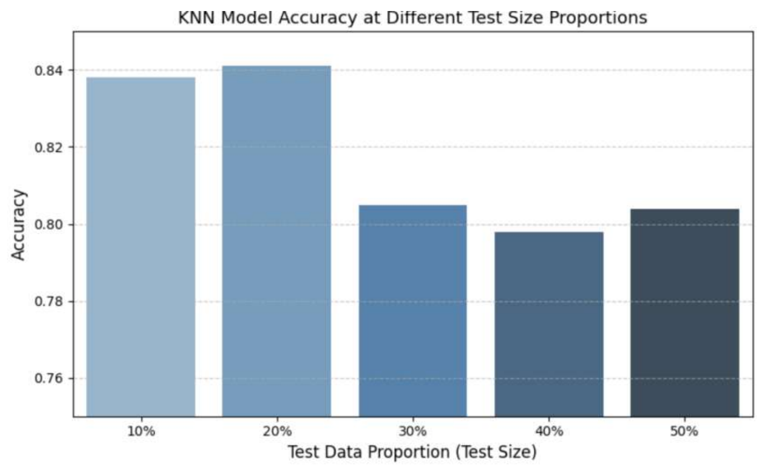


Fig. 7 Visualization of Model Accuracy against Variations in Test Data Proportions

Based on the test results visualized in Figure 7, the highest accuracy was achieved at a ratio of 80:20 with a value of 0.8407, followed by a ratio of 90:10 with a value of 0.8364. Conversely, increasing the test data portion to 50% caused a decrease in accuracy, although not significant. Based on these findings, the 80:20 ratio was selected as the primary configuration because it provides an optimal balance between training effectiveness and evaluation validity.

j) Determining the Optimal K Value

The selection of the K value in the K-Nearest Neighbors (K-NN) method is an important step because it greatly affects the accuracy and generalization of the model. A K value that is too small tends to cause the model to be too sensitive to the training data (overfitting), while a K value that is too large can cause the model to lose its ability to distinguish patterns (underfitting). Therefore, testing is carried out to determine the most optimal K value.

This test uses the k-fold cross-validation technique on data that has undergone oversampling, measuring the average accuracy for each K value in the range of 1 to 10. The distance between data points is calculated using the cosine similarity metric, which is considered more suitable for text-based data. The test results are visualized in the form of a curve in Figure 8.

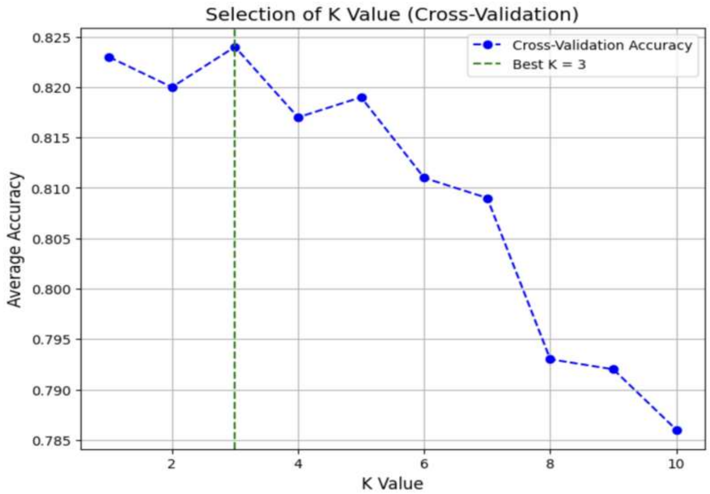


Fig. 8 Optimal K Value Selection Graph (Cross-Validation)

From the graph, it can be seen that a value of K = 3 produces the highest accuracy of 0.8268, slightly higher than K = 1, which has an accuracy of 0.8264. Although the difference is small, K = 3 is chosen because it is more stable and less sensitive to outliers, so the risk of overfitting is lower than with very small K values. Additionally, there is a clear trend of decreasing accuracy as the K value increases, supporting the selection of K at this optimal point.

To clarify the comparison, Table 8 below presents the average accuracy results from cross-validation for each K value tested.

Table 7 Average Cross-Validation Accuracy At Various K Values

| K value | Cross-Validation Accuracy |
|---|---|
| 1 | 0.8264 |
| 2 | 0.8184 |
| 3 | **0.8268** |
| 4 | 0.8159 |
| 5 | 0.8184 |
| 6 | 0.8105 |
| 7 | 0.8068 |
| 8 | 0.7961 |
| 9 | 0.7941 |
| 10 | 0.7845 |

Based on the table, a K value that is too large causes a significant decrease in accuracy. Conversely, K=3 not only provides the highest accuracy value, but also shows better stability than K=1, so it is chosen as the optimal value in this study. This selection is also supported by the results of the initial analysis, where smaller K values show a tendency toward overfitting on the training data, while larger K values exhibit a decline in performance due to overgeneralization. Thus, K=3 becomes the parameter used in the final classification stage.

k) Model Evaluation

Model performance evaluation is an important step in assessing the effectiveness of classification algorithms. In this study, the K-Nearest Neighbors (K-NN) model with K = 3 was evaluated using test data divided into 80% training data and 20% test data (3,516 training data and 879 test data). The evaluation process utilized accuracy, precision, recall, F1-score, and confusion matrix metrics. The confusion matrix visualization is shown in Figure 9, illustrating the distribution of predictions for the three sentiment classes: positive, negative, and neutral.

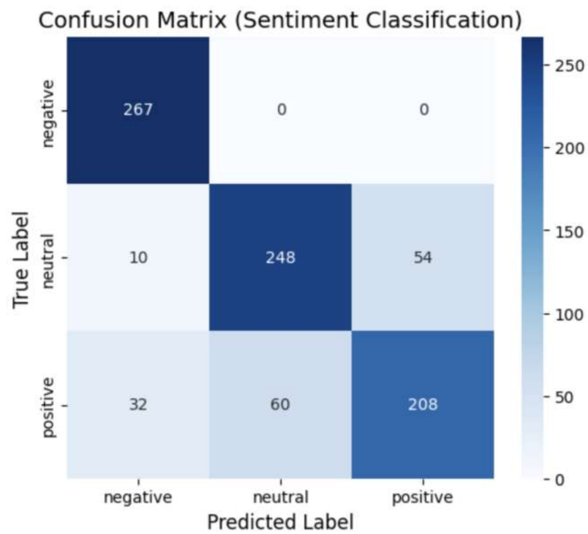


Fig. 9 Confusion Matrix Results

Based on the results of the confusion matrix visualization, the following information can be obtained:

- Negative sentiment: Of the total 267 tweets, all were correctly predicted without any classification errors, meaning that the accuracy rate for this class reached 100%.
- Neutral sentiment: Out of 312 tweets, 248 were correctly classified, while 10 were incorrectly classified as negative and 54 as positive.
- Positive sentiment: Out of 300 tweets, 208 tweets were classified correctly, while 32 tweets were classified as negative and 60 as neutral.

From these results, the negative class showed the best performance based on evaluation metrics such as precision, recall, and f1-score. The following are the manual evaluation metrics calculations:

$$Precision_{negative} = \frac{267}{267 + 42} = \frac{267}{309} \approx 0.864$$

$$Recall_{negative} = \frac{267}{267 + 0} = \frac{267}{267} \approx 1.000$$

$$F1 - Score_{negative} = 2 \times \frac{0.864 \times 1.000}{0.864 + 1.000} = \frac{1.728}{1.864} \approx 0.927$$

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Prediction} = \frac{267 + 248 + 208}{879} = \frac{723}{879} \approx 0.822$$

Based on the evaluation results, the accuracy obtained in the test data was 0.82 or 82%. This shows that the model successfully classified tweets with a fairly high level of accuracy. In addition to accuracy, other metrics used for evaluation were precision, recall, and f1-score, which were displayed in the form of a classification report. The following is Table 9, which shows the classification report results for each sentiment class.

Table 8 K-NN Model Classification Results Report

| Sentiment Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Negative | 0.86 | 1.00 | 0.93 | 267 |
| Neutral | 0.81 | 0.79 | 0.80 | 312 |
| Positive | 0.79 | 0.69 | 0.74 | 300 |
| Accuracy | | | **0.82** | **879** |
| Macro Avg | 0.82 | 0.83 | 0.82 | 879 |
| Weighted Avg | 0.82 | 0.82 | 0.82 | 879 |

The overall accuracy score of the model reached 0.82 or 82%, indicating that the majority of K-Nearest Neighbors (K-NN) model predictions were successful in matching the actual sentiment labels. In addition to accuracy, other evaluation metrics such as macro average and weighted average for precision, recall, and f1-score also showed fairly high values, ranging from 0.82 to 0.83.

## 4. CONCLUSIONS

This study analyzes public sentiment toward free nutritious school meal programs using data from Twitter (X) and the K-Nearest Neighbors (K-NN) method. Out of 3,007 tweets that underwent pre-processing and labeling, it was found that public sentiment tends to be negative, although there are positive and neutral opinions. After applying oversampling to address data imbalance, the class distribution became more balanced, making the analysis more representative. The K-NN model showed that negative sentiment was easier to identify compared to neutral and positive sentiment, which tended to be difficult to distinguish due to similar language structures. With an optimal K value of 3 and an 80:20 data split, the model achieved an accuracy of 82%. Further evaluation showed the best performance on negative sentiment with a recall of 1.00 and an F1-score of 0.93, while performance on neutral and positive sentiment still has room for improvement.

## 5. SUGGESTIONS

This study has several important contributions for various parties. For the government and policymakers, the results of this study can be used as consideration to understand public opinion and concerns regarding free nutritious meal programs, as well as a basis for designing more responsive communication strategies and adjusting program implementation to be more open and equitable. For future researchers, this study still has room for development, either through increasing the amount and period of data, applying advanced pre-processing techniques to overcome language ambiguity, or exploring other algorithms such as SVM, Random Forest, or deep learning models such as BERT to compare their performance with the K-NN method. Meanwhile, for academics and students, this research shows that simple methods such as K-NN remain relevant and effective in text classification, especially when supported by good data processing and feature representation, making them an economical and practical alternative in building social media-based sentiment analysis systems.

## DAFTAR PUSTAKA

[1] T. Andryansyah and S. Dewi, “Analysis of the Role of Government in Efforts to Prevent Stunting to Achieve Quality Human Resources in Ampek Angkek Subdistrict” *Indonesia*, vol. 3, no. 4, 2024, doi: 10.30640/inisiatif.v3i4.2943.

[2] Ministry of Health of the Republic of Indonesia, “Prevalence of Stunting in Indonesia Declines to 21.6% from 24.4%.” 2024. [Online]. Available: https://sehatnegeriku.kemkes.go.id/baca/rilismedia/20230125/3142280/prevalensi-stunting-di-indonesia-turun-ke-216-dari-244/

[3] UNICEF, “Nutrition deprivation in early childhood.” 2024. [Online]. Available: https://www.unicef.org/media/157661/file/Child-food-poverty-2024.pdf

[4] A. Ramadhani, I. Permana, M. Afdal, and M. Fronita, “Sentiment Analysis of Public Responses on Twitter Related to Prabowo’s Free Lunch Program – Gibran Using the Naïve Bayes Classifier and Support Vector Machine Algorithms” *BITS*, vol. 6, no. 3, pp. 1509–1516, 2024, doi: 10.47065/bits.v6i3.6188.

[5] A. Sitanggang, Y. Umaidah, and R. I. Adam, “Analysis of Public Sentiment Toward the Free Lunch Program on Social Media X Using the Naïve Bayes Algorithm” *J. Inform. Dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.4902.

[6] F. R. Yuliardi, F. Fauzi, and T. W. Utami, “Analysis of Public Opinion Sentiment Towards Stunting After the First Vice Presidential Debate of 2024 Using the Bootstrap Aggregating Naïve Bayes Algorithm” in *SEMNAS*, 2024, vol. 3, pp. 1129–1139.

[7] M. Azhari, “Analysis of Public Opinion Sentiment on the Free Lunch Program Using Random Forest on Media X” *BITS*, vol. 6, no. 3, pp. 1932–1942, 2024, doi: 10.47065/bits.v6i3.6423.

[8] R. Saputra and F. N. Hasan, “Sentiment Analysis of the Free Lunch & Milk Program Using the Naive Bayes Algorithm,” *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 6, no. 3, pp. 411–419,

2024, doi: 10.47233/jteksis.v6i3.1378.

[9] B. Liu, *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions, Second Edition*. 2020. doi: 10.1017/9781108639286.

[10] P. et al., “Sentiment Analysis of the Relocation of the Capital City of Nusantara” *J. KomtekInfo*, vol. 10, pp. 1–7, 2023, doi: 10.35134/komtekinfo.v10i1.330.

[11] H. D. Herawati, H. K. Rahayu, R. K. Triastanti, and R. Rusiyono, “Preventing Malnutrition in Preschool Children through Training in Nutritional Status Assessment for Early Childhood Education Teachers” *Media Karya Kesehat.*, vol. 6, no. 1, pp. 157–168, 2023, doi: 10.24198/mkk.v6i1.40800.

[12] V. H. Irsansaputra and D. Udjulawa, “Sentiment analysis of public comments on the ‘free lunch program’ using the KNN (K-nearest neighbor) and SMOTE algorithms” *J. XYZ*, vol. 3, no. 2, pp. 1–8, 2024.

[13] S. Sathyanarayanan and B. R. Tantri, “Confusion Matrix-Based Performance Evaluation Metrics” *AJBR*, Dec. 2024, doi: 10.53555/AJBR.v27i4S.4345.

[14] R. Yunitarini, J. F. Gultom, Stefany, S. Helmi, and M. Fakhry, “Classification of Traditional Madura Herbal Medicine Using the K-Nearest Neighbors (KNN) Method and Term Frequency-Inverse Document Frequency (TF-IDF)” *J. Inform. Polinema*, vol. 11, pp. 99–106, 2022, doi: 10.33795/jip.v11i1.6456.

[15] F. T. Admojo and Ahsanawati, “Classification of Alcohol Aromas Using the KNN Method” *Indones. J. Data Sci.*, vol. 1, no. 2, pp. 34–38, 2020, doi: 10.33096/ijodas.v1i2.12.

[16] A. Dwiki, A. Putra, Juanita, and B. Luhur, “Sentiment Analysis of User Reviews of the seed and Bareksa Apps Using the KNN Algorithm” *J. XYZ*, vol. 8, no. 2, pp. 636–646, 2021.

[17] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, “The impact of class imbalance in classification performance metrics based on the binary confusion matrix,” *Pattern Recognit.*, vol. 91, pp. 216–231, 2019.