Volume 11 Issue 1 Year 2026 | Page 393-403 ISSN: 2527-9866
Received: ***18-02-2026*** | *Revised:* ***24-02-2026*** | *Accepted:* ***28-02-2026***

Sentiment Analysis of the Free Nutrious Meal Program using IndoBERT and RCNN Methods

Ancilla Nebrisca Valonika[1], Kristoko Dwi Hartomo[2]
[1,2] Satya Wacana Christian University, Salatiga, Indonesia, 50711
e-mail: 682022026@student.uksw.edu[1], kristoko@uksw.edu[2]
*Correspondence: 682022026@student.uksw.edu

Abstract: *This study examines public sentiment regarding the Free Nutritious Meal Program through a deep learning-based sentiment classification methodology applied to X and TikTok. The suggested method uses a hybrid IndoBERT RCNN architecture, with IndoBERT being used to extract features and RCNN being used to classify sentiment. There are 10,000 comments from each platform in the dataset. These comments went through preprocessing and sentiment labeling steps. Model evaluation was conducted using stratified K-fold cross-validation with different combinations of learning rate, batch size, and epochs. The best configuration achieved an accuracy and F1-score of 78% on X and 83% on TikTok. The model performs well in identifying overall sentiment patterns, although neutral sentiment remains challenging to classify, particularly in X data containing sarcastic or indirect language. These findings provide empirical insights into cross-platform sentiment characteristics and highlight the potential of this approach for testing sentiment monitoring strategies across.*

Keywords: *Sentiment analysis, Free Nutritious Meal Program, IndoBERT, RCNN, hyperparameters*.

1. Introduction

Stunting is still a major nutritional problem that has long-term effects on the quality of human resources. As a result, the elimination of all forms of malnutrition is an important goal in the Sustainable Development Goals (SDGs) [1]. In Indonesia, even though the prevalence of stunting is going down, the number is still below the target of 19% set by the Rencana Pembangunan Jangka Menengah Nasional (RPJMN) 2020–2024 [1]. To expedite the decrease of stunting, the Indonesian government initiated the Free Nutritious Meal Program (MBG) in January 2025, focusing on students and other vulnerable populations [2]. Nonetheless, the execution of the MBG Program continues to encounter numerous challenges, such as regulatory complications, stakeholder coordination, and public acceptance [3]. In the digital age, social media sites like X and TikTok are the best places for people to quickly share their thoughts, opinions, and criticisms of government policy. To find out what people think about the MBG program, you need to read what people are saying on both platforms. But social media data is informal and full of language problems, so we need an analytical approach that can help us understand the context well enough to measure sentiment reliably.

Natural Language Processing (NLP)-based sentiment analysis has been extensively utilized to assess public opinion regarding government policies. Prior research has predominantly utilized traditional machine learning algorithms, including Naïve Bayes [4][5], Support Vector Machine [6][7], and Random Forest [6][8]. These methods are effective for straightforward datasets. However, they inadequately capture semantic relationships and contextual significance, particularly in the complex and rapidly evolving domain of social media communications. As a result, deep learning–based NLP methods are needed to better capture the meaning of text in context.

Language models based on transformers, like IndoBERT, have shown better results in many Indonesian NLP tasks because they can learn contextual representations from large amounts of data [9]. Numerous studies have integrated IndoBERT with various deep learning architectures, including Recurrent Convolutional Neural Networks (RCNN), to improve sentiment classification efficacy [10]. Nevertheless, the majority of previous studies have concentrated on a singular social media platform and have not comprehensively examined the influence of hyperparameter variations on model performance, especially regarding public policy evaluation in Indonesia.

Previous studies on public policy sentiment analysis still exhibit limitations in three main aspects. First, there aren't many cross-platform studies yet, even though the way people talk about things on X and TikTok may be very different. Second, the investigation of hyperparameter combinations within the IndoBERT-RCNN architecture concerning public policy has not been extensively studied. Third, the precision of neutral sentiment classification continues to pose a challenge because of its unclear distinctions from other categories, although it seldom receives focused scrutiny. Given the identified gaps, this study investigates the impact of various combinations of learning rates (5e-5, 3e-5, 2e-5) and batch sizes (16, 32) on the performance of the IndoBERT-RCNN model. It also looks at how the data and public opinion patterns are different on the X and TikTok platforms. This study also looks at how well the model can correctly classify neutral sentiment.

This study is novel due to its integration of cross-platform analysis with the examination of IndoBERT-RCNN hyperparameters, alongside a distinct emphasis on the difficulties associated with neutral class classification. This research seeks to assess model performance across diverse hyperparameter combinations using a manually labeled dataset and to examine variations in sentiment patterns among platforms. The results should help build Indonesian-language sentiment analysis models in theory and help with public policy evaluation on social media in practice.

2. Literature Review (optional)

Previous studies have explored the use of IndoBERT and its hybrid architectures for Indonesian sentiment analysis. Jayadianti et al. [9] demonstrated that fine-tuned IndoBERT integrated with RCNN achieved higher performance than the standalone IndoBERT model, reporting an accuracy of 95.16% and an F1-score of 93.27% under specific hyperparameter settings (batch size 16, learning rate 3e-5, epoch 3). These findings indicate that combining contextual transformer representations with recurrent convolutional layers can enhance classification performance. Given the empirical evidence provided in that study, IndoBERT–RCNN has been established as a competitive architecture for Indonesian sentiment analysis. However, the study did not systematically evaluate the impact of alternative hyperparameter configurations.

Further investigation by Kusoema et al. [10] confirmed that hyperparameter selection significantly influences IndoBERT–RCNN performance, identifying learning rate and batch size as critical factors. Although their results support the importance of parameter tuning, the analysis was conducted on a single issue-specific dataset and did not examine cross-platform sentiment characteristics. Other related works remain platform-dependent. Riyadi et al. [11] reported high classification accuracy on YouTube comments (up to 98%), yet generalizability beyond that platform was not assessed. Ridho et al. [12] showed IndoBERT outperforming traditional machine learning approaches in fake news detection, but their work focused on binary classification rather than multiclass sentiment tasks. Similarly, Chamid et al. [13] combined IndoBERT with topic modeling to examine public opinion, emphasizing thematic exploration rather than predictive sentiment evaluation.

Taken together, existing research confirms the effectiveness of IndoBERT and IndoBERT–RCNN architectures. Nonetheless, systematic hyperparameter experimentation across various social media platforms, especially in the context of public policy discourse—continues to be constrained. In this context, the current study does not seek to reaffirm the superiority of IndoBERT–RCNN over its base model, as this has been empirically validated in previous research. Instead, it focuses on testing how stable and sensitive IndoBERT–RCNN is to changes in learning rate and batch size on different platforms (X and TikTok). This study aims to deliver a more thorough evaluation of model robustness and generalization by integrating manual labeling validation and comparative analysis across platforms.

3.Methods

This study implements transfer learning for sentiment analysis by leveraging the IndoBERT model that has undergone pre-training and fine-tuning, where all output tokens are used as inputs for the RCNN classification model [9]. Figure 1 illustrates the sequential research stages, starting from dataset collection, data labeling, preprocessing, model training, to result evaluation and model testing.

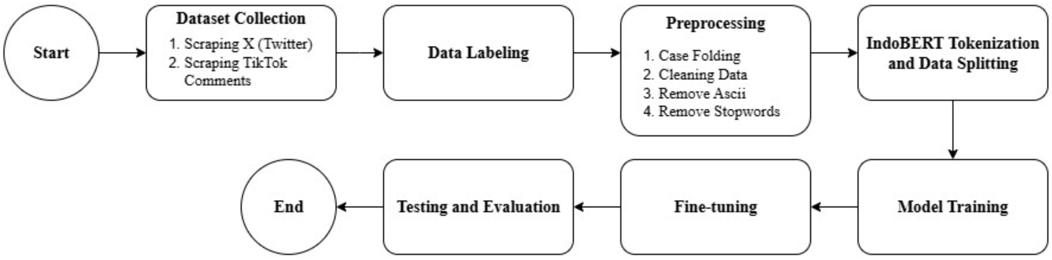

Figure 1. Research Stages

A. Dataset Collection

This study collected publicly available comments related to the Free Nutritious Meal Program (Program Makan Bergizi Gratis, MBG) from X (Twitter) and TikTok between January 1, 2025 and October 30, 2025, utilizing the official APIs and access mechanisms provided by each platform. Data were obtained through the official access mechanisms provided by each platform in accordance with their applicable terms of service and data use policies. For X, comments were retrieved using the keywords "MBG" and "makan bergizi gratis," with language filtering restricted to Indonesian ("lang:id"). For TikTok, comments were gathered from publicly accessible videos discussing the implementation of the MBG program, particularly in elementary, junior high, and senior high school contexts. A total of 10,000 public comments were collected from each platform. Only publicly accessible content was included in this study, and no private or restricted data were accessed. All user identifiers were removed during preprocessing to ensure anonymity and protect user privacy. The dataset was used exclusively for academic research purposes, and all data were exported in CSV format for further analysis.

B. Data Labeling

Before preprocessing, all data were manually annotated by the author. A total of 10,000 comments from X and 10,000 from TikTok were labeled using predefined guidelines to ensure consistency. Each comment was classified into one of three categories: positive (supportive or approving statements), negative (criticism, dissatisfaction, rejection, or sarcastic disapproval), and neutral (factual statements, questions, or emotionally unexpressive opinions). The classification was based on the overall contextual meaning of each comment rather than isolated keywords. To evaluate annotation consistency, an intra-annotator reliability test was performed by randomly re-labeling 10% of the data after a time interval without referring to the initial labels. The agreement between the two rounds was high, indicating stable and internally consistent annotations.

C. Preprocessing

The preprocessing stage is necessary to minimize the noise typically present in crawled data, which is generally unstructured [14]. This study employs four preprocessing steps as follows:

1. Case Folding, which makes the text uniform by changing all the letters to lowercase.
2. Data Cleaning gets rid of URLs, hashtags, characters that start with "@" and repeated characters. It also gets rid of extra spaces, leaving only one space between words.
3. Remove ASCII, which gets rid of extra characters like emojis, punctuation marks, numbers, and words that are only one character long.
4. Remove Stopwords, which are words that don't add any information, like conjunctions and prepositions.

D. Model Architecture

This study employs sentiment analysis via a transfer learning methodology, utilizing a pre-trained IndoBERT model, subsequently augmented by incorporating all output tokens as inputs for the RCNN classification model [9]. The modeling process consists of the following steps:

1. IndoBERT is used to extract contextual representations of each token based on pre-training and fine-tuning. The model employed is IndoBERT-base (indobenchmark/indobert-base-p1) consisting of 12 Transformer layers with a hidden size of 768.
2. All output tokens from IndoBERT (768 dimensions) are utilized as inputs to the RCNN to capture bidirectional contextual information through a two-layer bidirectional LSTM with a hidden size of 256 and a dropout rate of 0,3.
3. The RCNN builds word representations by combining a bidirectional RNN to model left–right contextual dependencies and a convolutional layer to extract salient semantic features [15]. In this study, the convolutional layer is implemented as a one-dimensional Conv1D with 512 input channels, 256 output channels, kernel size 3, padding 1, and ReLU activation.
4. A global max-pooling layer is applied to select the most informative features, followed by a fully connected layer with 256 input units and 128 output units, using ReLU activation and a dropout rate of 0,2.
5. The final features are processed by an output layer with 128 input units and 3 output units and classified using a softmax function implemented through CrossEntropyLoss.

E. Data Splitting and Validation

The dataset was stratified into a development set (80%) and a holdout test set (20%). Stratified K-Fold Cross-Validation was then applied to the development set, where each fold alternately served as the validation set while the remaining folds were used for training. This approach resulted in a final split of 64% of the total data for training, 16% for validation, and 20% for testing, with the test set kept completely independent throughout the entire model development process.

F. Hyperparameter Settings

This study examine various combinations of hyperparameters, including learning rates (5e-5, 3e-5, and 2e-5) and batch sizes (16 and 32). Some choices are based on earlier studies that used the IndoBERT–RCNN architecture for text classification and showed that it worked well in some setups [9][10]. Furthermore investigates epoch variations (5, 10, and 15) to ascertain the optimal number of training repetitions, considering that the cited studies employed differing epoch counts [16]. The best hyperparameter combination is chosen based on how well it works with accuracy, precision, recall, and F1-score, and then it is used in all model experiments.

G. Model Evaluation

Model evaluation is performed by examining different combinations of learning rate and batch size hyperparameters within the IndoBERT–RCNN architecture on both platforms, utilizing accuracy and F1-score metrics to evaluate prediction quality. The goal of this process is to find the hyperparameter settings that give the best performance before full validation. Stratified k-Fold Cross-Validation with k = 3 is used to check the model. The selection of a relatively small k value because the dataset only has 10,000 samples. This will give model a more stable performance estimate and lower the chance of overfitting, while still making sure that each fold has enough test samples [17]. The dataset is split into three equal-sized groups (folds) at random, but the proportions of classes in each fold stay the same. After that, training and testing are done three times [18], with each data sample being used as part of the test set only once. The two best combinations of hyperparameters are then re-evaluated using the IndoBERT–RCNN architecture and checked with this method to make sure there is no bias and that the model's performance stays stable.

4. Result and Discussion

This study assesses the efficacy of the IndoBERT–RCNN model in discerning public sentiment regarding the Free Nutritious Meal Program (MBG) through the analysis of user comment data sourced from X and TikTok social media platforms. The analysis starts with processing the raw data, which includes labeling the sentiment, preprocessing, tokenization, and training the model with different hyperparameter settings. Next, we use measures like accuracy and F1-score to see how well the model works. Confusion matrices make this clear in a way that is easy to understand.

A. Dataset

This study uses public comments about the Free Nutritious Meal Program that were posted on the X and TikTok social media sites. There are 10,000 comments on each platform that are used to test the model. There are three groups for the comments based on their tone: positive, neutral, and negative. Table 1 shows how tagged sentiment data is spread out across different platforms.

Table 1. Sentiment Class Distribution for X and TikTok Datasets

Platform Dataset	Positive	Neutral	Negative
X (Twitter)	4534	2221	3945
TikTok	4055	1625	4320

Prior to the model training process, the labeled data are processed through a preprocessing stage. This stage aims to reduce noise in textual data, thereby improving the quality of input used for model training. Table 2 presents a comparison of dataset characteristics before and after the application of preprocessing, as described below.

Table 2. Comparison of Dataset Before and After Preprocessing

Platform Dataset	Before Preprocessing	After Preprocessing
X (Twitter)	@bartenderrrrrr mending investasi pendidikan gratis dah nanti feedback kok: kuliah gratis > berpendidikan dan berpikiran luas/terbuka > menaikkan taraf hidup > sukses > sejahtera > bisa makan bergizi dari hasil sendiri gak usah yang namanya ngasih mbg tapi sejahterakan aja rakyatnya.	mending investasi pendidikan gratis dah feedback kuliah gratis berpendidikan berpikiran luas terbuka menaikkan taraf hidup sukses sejahtera makan bergizi hasil sendiri gak namanya ngasih mbg sejahterakan aja rakyatnya
TikTok	BERMANFAAT MIN, apalagi untuk ibu2 yg pekerja aktif/karyawan, gk perlu masak bekal lg, 😁😁😁	bermanfaat min apalagi untuk ibu yg pekerja aktif karyawan gk perlu masak bekal lg

B. IndoBERT Tokenazation

Tokenization is performed using the indobenchmark/indobert-base-p1 model, a BERT variant that has been pre-trained on Indonesian language corpora. This process splits the input text into the smallest linguistic units in the form of tokens, which are then converted into numerical representations (token IDs) so that they can be processed by the RCNN model [19].

Table 3. Tokenization Results

Process	Sentence 1 (X/ Twitter)	Sentence 2 (TikTok)
Main Input	*komisi ii dpr ri pemda sukseskan program makan siang bergizi gratis*	*makan siang bergizi gratis jaga anak anak aktif bersemangat*
Tokenization	'komisi' 'ii' 'dpr' 'ri' 'pemda' 'sukses' '##kan' 'program' 'makan' 'siang' 'bergizi' 'gratis'	'makan' 'siang' 'bergizi' 'gratis' 'jaga' 'anak' 'anak' 'aktif' 'bersemangat'
Encoding	[3892, 1693, 2441, 949, 9615, 2152, 32, 986, 521, 3346, 15738, 1243]	[521, 3346, 15738, 1243, 7773, 436, 436, 1786, 10457]
Adding Special Tokens	[2, 3892, 1693, 2441, 949, 9615, 2152, 32, 986, 521, 3346, 15738, 1243, 3, 0, 0, 0, 0, 0, 0]	[2, 521, 3346, 15738, 1243, 7773, 436, 436, 1786, 10457, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BERT Embeddings	[0.5568, 1.2416, 0.4262, 0.0741, 1.7319, -2.4398, 0.0471, -1.007, 0.5294, 0.7248, ….]	[-0.7449, 2.1274, 0.0717, -0.2732, 1.8633, 0.4521, -1.2345, 0.7890, -0.5432, 1.0987, ….]

The modeling process presented in Table 3 begins with text tokenization using IndoBERT. At this stage, the input text is segmented into subword units. For example, in Sentence 1, the word *"sukseskan"* is split into two tokens, namely *"sukses"* and "*##kan*", where the prefix ## indicates that the token is a continuation subword of the preceding token. After this tokenization process, the result is trimmed or padded to a maximum length of 128 tokens to keep the model input consistent. Then, an encoding process turns the resulting tokens into numbers based on the vocabulary learned during IndoBERT pre-training. Each token is given a different integer index. Table 3 shows that the encoding values for the words *"sukses"* and *"##kan"* in Sentence 1 are 2152 and 32, respectively. The word *"anak"* in Sentence 2 has an ID of 436, which happens twice.

After that, special tokens are added to the encoded sequences to show specific structural and functional responsibilities. The row titled "Step 3: Adding Special Tokens" shows that the [CLS] token (ID: 2), which is used for classification, is always added to the beginning of each sequence. The [SEP] token (ID: 3), which is a separator token, is added at the end of the original sequence before padding, as shown in both examples ([2, ..., 3, 0, 0, ...]). The [PAD] token (ID: 0) is used to make all sequences the same length by filling in the spaces after the [SEP] token or the end of the sentence. IndoBERT then processes all encoded sequences that have special tokens using different transformer encoder layers. This creates BERT embeddings, which are continuous-valued vector representations for each token position. These vectors contain information about the meaning, structure, and context of the input text. They will be used in the next steps of modeling.

C. Model Evaluation

During the training of the IndoBERT and RCNN models, different hyperparameter settings were tested. These included learning rates of 5e-5, 3e-5, and 2e-5; batch sizes of 16 and 32; and epoch settings of 5, 10, and 15. Table 4 shows the ten best combinations out of the eighteen that were made during hyperparameter optimization. Based on these numbers, Combination 6, which has a batch size of 32, a learning rate of 2e-05, and 15 epochs, does the best, with an accuracy and an F1-score of 0.78. After that, model validation was done on the test data using a three-fold stratified K-Fold cross-validation method. The results of the evaluation show that the third fold had the highest accuracy, which was 0.78, as shown in Table 5.

Table 4. X (Twitter) Evaluation Results

Batch Size	Learning Rate	Epochs	Accuracy	Precision	Recall	F1-Score
16	2e-05	5	0.7873	0.7874	0.7873	0.7872
		10	0.7878	0.7845	0.7878	0.7854
		15	0.7742	0.7928	0.7742	0.7793
	5e-05	10	0.7878	0.8003	0.7878	0.7920
32	**2e-05**	10	0.7869	0.7995	0.7869	0.7912
		15	**0.7887**	**0.7892**	**0.7887**	**0.7876**
	3e-05	5	0.7775	0.7707	0.7775	0.7728
		10	0.7785	0,7775	0.7785	0.7753
		15	0.7850	0.7902	0.7850	0.7856
	5e-05	5	0.7817	0.7785	0.7817	0.7789

Table 5. X (Twitter) Validation Results

Fold	Training Accuracy	Best validation Accuracy
1	**0.7721**	**0.7870**
2	0.7668	0.7715
3	0.7586	0.7905

The average confusion matrix from 3-fold cross-validation on the test dataset is shown in Figure 2. The model is very good at finding the negative class, with an average of 1084.7 correct predictions. However, it does make mistakes by putting an average of 210.3 and 109.3 instances into the neutral and positive groups, respectively. The model makes an average of 368.0 correct predictions for the neutral class, but there is still a lot of overlap with the negative and positive classes. The positive class shows the best performance, with an average of 1278.7 correct predictions. This shows that the model is very good at recognizing positive sentiment. The evaluation results show that the model does a good job of classifying things, especially when it comes to positive mood. However, its performance on the neutral class is the lowest compared to the positive and negative classes.

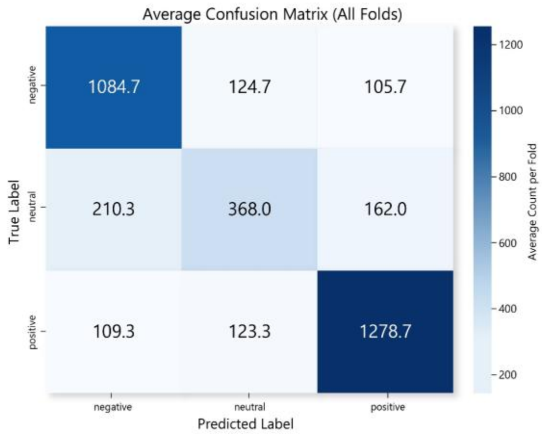

Figure 2. Average Confusion Matrix Data Validation X (Twitter)

Table 6 shows the ten best combinations out of the eighteen that were made during hyperparameter optimization. Based on this information, Combination 9, which has a batch size of 32, a learning rate of 3e-05, and 5 epochs, gets the best results, with an accuracy and an F1 score of 0.83. After that, K-Fold cross-validation is used to test the results, and the first fold gets the highest accuracy of 0.80, as shown in Table 7.

Table 6. TikTok Evaluation Results

Batch Size	Learning Rate	Epochs	Accuracy	Precision	Recall	F1-Score
16	2e-05	10	0.8240	0.8214	0.8240	0.8205
		15	0.8185	0.8197	0.8185	0.8183
	3e-05	10	0.8220	0.8260	0.8220	0.8231
	5e-05	10	0.8215	0.8215	0.8215	0.8243
		15	0.8235	0.8242	0.8235	0.8238
32	2e-05	10	0.8190	0.8183	0.8190	0.8186
	3e-05	**5**	**0.8350**	**0.8328**	**0.8350**	**0.8336**
	5e-05	5	0.8155	0.8300	0.8155	0.8202
		10	0.8280	0.8269	0.8280	0.8274
		15	0.8215	0.8244	0.8215	0.8228

Table 7. TikTok Validation Results

Fold	Training Accuracy	Best validation Accuracy
1	0.7951	0.8081
2	**0.8071**	**0.8058**
3	0.8056	0.8186

Figure 3 shows the average confusion matrix from 3-fold cross-validation on the test data. The model achieves the highest classification accuracy for the positive class, with an average of 1,163.3 correct predictions. Performance on the negative class is also robust, yielding 1,179.3 correct predictions. In contrast, the neutral class exhibits the lowest classification performance, with only 332.7 correct predictions on average, and is frequently misclassified as either negative or positive. These results indicate that while the model effectively identifies positive and negative sentiments, its discriminative ability for the neutral class remains limited.

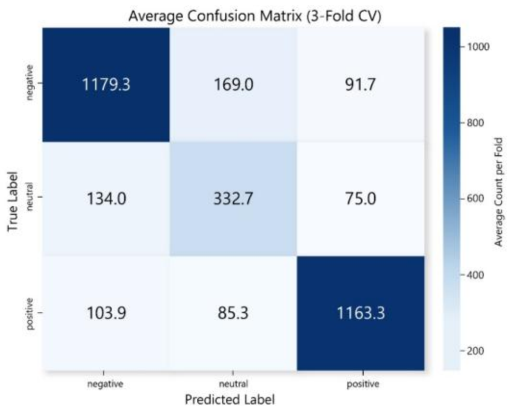

Figure 3. Average Confusion Matrix Data Validation TikTok

D. Model Evaluation

This study not only evaluates the model but also tests input sentences to gauge prediction performance. The test uses data from both X (Twitter) and TikTok, and the results are then looked at separately for each platform.

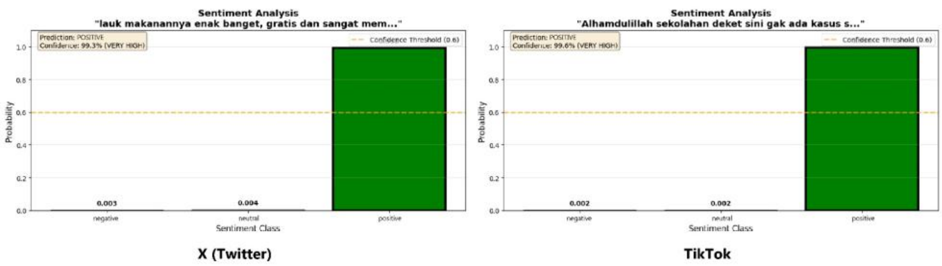

Figure 4. Positive Sentiment Prediction Testing on X and TikTok Comments

Figure 4 shows that the IndoBERT-RCNN model can always tell if comments on both X (Twitter) and TikTok are good. The phrase *"lauk makanannya enak banget, gratis, dan sangat membantu"* was correctly predicted to have a positive emotion in the X comment data. This line makes it clear how much people like the Free Nutritious Meal Program (MBG), which helps the model correctly figure out what it means when it says something positive. On the TikTok platform, the model also demonstrates the ability to identify positive sentiment in longer and more narrative sentences, such as *"Alhamdulillah sekolahan deket sini gak ada kasus seperti ini.. dg adanya program MBG ini sangat membantu para orang tua yang blm sempat memasak untuk anaknya.. anak saya pun jadi suka menabung karena uang saku jadi sisa gak habis.. sangat terbantu sekali program ini buat saya."* Even though the sentence structure is complicated and the language is casual, the model can still convey the overall positive meaning of the statement because it uses a lot of appreciative words. This finding shows that the model learns explicitly expressed positive sentiment more easily, even when the text's attributes differ between platforms.

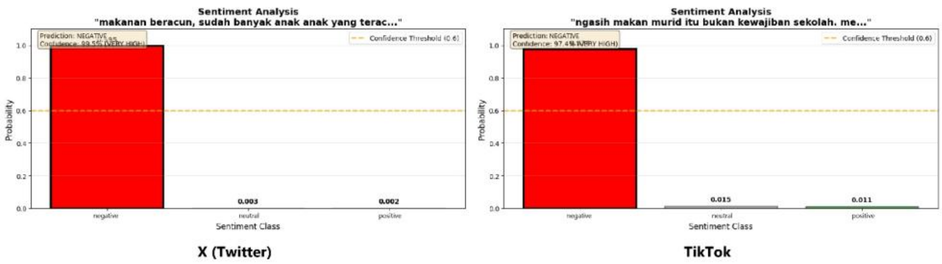

Figure 5. Negative Sentiment Prediction Testing on X and TikTok Comments

Figure 5 shows the negative sentiment category, where the model consistently works to find comments that include criticism and strong negative feelings. The phrase *"makanan beracun, sudah banyak anak-anak yang teracuni oleh MBG"* in the X comment data was correctly classified as negative sentiment. This statement uses strong language and clear accusations about the policy, which helps the model find the negative emotional cues.

Similar results are also observed in TikTok comments, such as *"ngasih makan murid itu bukan kewajiban sekolah. mending program MBG ini dihentikan aja daripada bikin kasus keracunan kayak begini di mana-mana. terus alirkan uangnya ke hal lain kayak perbaikannya sekolah."* This comment is an extended argument against the MBG program. The model's ability to classify this statement as negative sentiment suggests that linguistic patterns featuring explicit criticism and causal narratives are typically more readily discernible than sentences with ambiguous sentiment content.

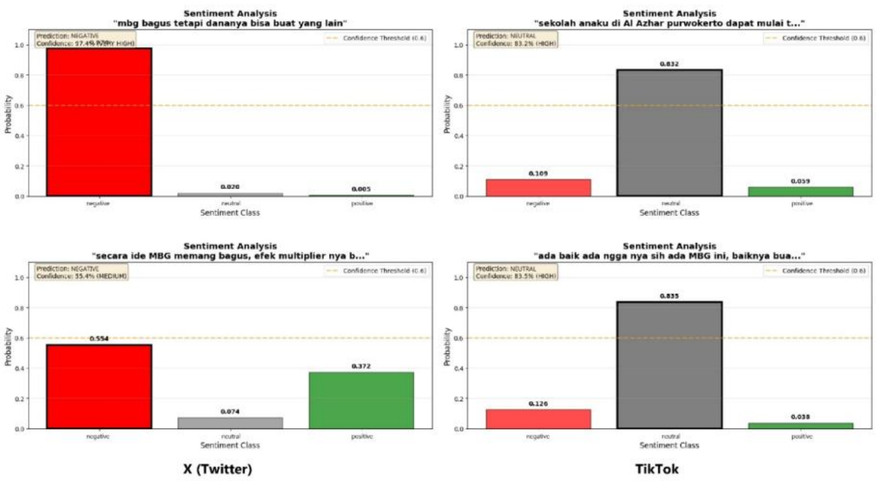

Figure 6. Neutral Sentiment Prediction Testing on X and TikTok Comments

Figure 6 presents additional testing examples to examine neutral sentiment classification on X and TikTok. The results indicate that the model still encounters difficulty distinguishing genuinely neutral comments from those containing mixed or contrastive evaluations. It's very likely that the statement *"MBG bagus tetapi dananya bisa buat yang lain"* on X is incorrectly labeled as negative. The adversative conjunction *"tetapi"* shifts the focus to the important sentence, making the model stress negative polarity indicators. This shows appreciation for the MBG program. A similar pattern appears in the longer comment *“secara ide MBG memang bagus, efek multiplier nya besar, membangkitkan ekonomi di sektor pertanian dan pangan, tapi pelaksanaannya menurut sy ga bagus karena akhirnya yg punya dapur mbg org2 politik, pengawasan kejaksaan dan kpk jg belum jelas utk program ini.”* Even though the first reviews were good, the forecast is mostly made up of clear negative phrases like "ga bagus," "org2 politik," and "belum jelas." This means that the model puts more weight on strong negative lexical cues, especially when they are used to make connections that are different from each other.

The true sentence *"sekolah anakku di Al Azhar Purwokerto dapat mulai tanggal 4 Agustus kemarin"* is correctly classified as neutral on TikTok. This shows that the model works well when language signals are clearly not meant to be evaluative. Likewise, the comment *“ada baik ada ngga nya sih ada MBG ini, baiknya buat ibu yg punya anak sekolah ga repot buatin bekal, dan anak pun uang jajan nya awet karena udh kenyang makan nasi. tapi bagi pedagang sekolahan itu berdampak bgt, suamiku jualan cireng isi di sekolah, dan biasa habis 4 box kalo pagi. semenjak ada MBG hanya habis 2 box”* is also predicted as neutral, despite noticeable probabilities in the positive and negative classes. The balanced presentation of benefits and disadvantages enables the model to capture proportional pro–contra discourse rather than defaulting to the final clause. The analysis of these cases demonstrates that inaccuracies in the neutral category are associated with various linguistic and distributional factors. Contrastive discourse structures that place evaluative clauses at the end of a sentence usually make certain polarity cues more important in predictive contexts. The presence of explicit negative lexical items and the concurrent incorporation of both positive and negative evaluations within a single discourse influence the probability distribution across categories. Another issue that might be making things worse is that the training data doesn't do a good job of modeling complex neutral expressions. This makes the model more sensitive to strong polarity signals than to subtle differences. Differences in linguistic style and platform features also seem to play a role in the differences in how well neutral classification works.

5. Conclusion

This study reveals that the IndoBERT-RCNN model delivers varying results across platforms when hyperparameters are tuned differently, with 79% accuracy and F1-score on X (batch size 32, learning rate 2e-05, 10 epochs) and 78% on TikTok (batch size 32, learning rate 3e-05, 5 epochs), confirming that platform-specific language patterns should guide model tuning. However, neutral sentiment identification remains problematic, mainly due to comments that blend opposing views or contain mixed evaluations as identified in the error analysis. Future work could explore emotion-based methods [20]. to better capture subtle language cues, examine specific policy aspects more deeply, and test approaches across different policy contexts to build monitoring systems that adapt better across platforms and provide more complete pictures of public opinion.

References

[1] BKPK, *Survei Kesehatan Indonesia*. Jakarta, 2023. [Online]. Available: https://www.badankebijakan.kemkes.go.id/laporan-tematik-ski/
[2] Suardi and A. S. Purmadani, “Makan Bergizi Gratis Dan Dampak Bagi Pertumbuhan Ekonomi,”

Bisnis-Net J. Ekon. dan Bisnis, vol. 8, no. 1, pp. 323–327, 2025, doi: https://doi.org/10.46576/bn.v8i1.6450.

[3] D. Herdiana, "Implementasi Kebijakan Makan Bergizi Gratis (MBG): Faktor-faktor Pendorong dan Penghambat," *J. Ilm. Multidisiplin*, vol. 3, no. 2, pp. 470–477, 2025, doi: https://doi.org/10.5281/zenodo.15063556.

[4] Y. Z. Vebrian and Kustiyono, "A Sentiment Analysis of Free Meal Plans on Social Media Using Naïve Bayes Algorithms Analisis Sentimen Terhadap Rencana Makan Gratis Di Sosial Media X Menggunakan Alogaritma Naive Bayes," vol. 10, no. 1, p. 2025, 2025, doi: https://doi.org/10.35314/3m2fcz69.

[5] A. Patue, G. F. Sidik, A. Affandy, and A. R. Ismail, "Sentiment Analysis Of Public Comments On Licensing Services Seeks To Use The Naive Bayes Algorithm With Genetic Selection Algorithm," *Sistemasi*, vol. 13, no. 1, p. 252, 2024, doi: 10.32520/stmsi.v13i1.3550.

[6] G. G. F. Djema and O. Suria, "Public Sentiment Analysis of Danantara Policy through Social Media X Using SVM and Random Forest," *INOVTEK Polbeng - Seri Inform.*, vol. 10, no. 2, pp. 1207–1217, 2025, doi: 10.35314/rcr21h75.

[7] M. R. Maulana and R. A. Putri, "Sentiment Analysis of the 2024 General Election Commission in Indonesia through Twiter using the Support Vector Machine (SVM) Algorithm," *Sistemasi*, vol. 13, no. 3, p. 1226, 2024, doi: 10.32520/stmsi.v13i3.4127.

[8] I. Muhandhis and A. S. Ritonga, "Public Sentiment Analysis on TikTok about Tapera Policy using Random Forest Classifier," *Sistemasi*, vol. 14, no. 1, p. 354, 2025, doi: 10.32520/stmsi.v14i1.4878.

[9] H. Jayadianti, W. Kaswidjanti, A. T. Utomo, S. Saifullah, F. A. Dwiyanto, and R. Drezewski, "Sentiment analysis of Indonesian reviews using fine-tuning IndoBERT and R-CNN," *Ilk. J. Ilm.*, vol. 14, no. 3, pp. 348–354, 2022, doi: 10.33096/ilkom.v14i3.1505.348-354.

[10] W. Jaya Kusoema *et al.*, "Analisis Sentimen dalam Kasus Korupsi PT. Pertamina menggunakan Metode indoBERT dan RCNN Sentiment Analysis on the PT Pertamina Corruption Case using IndoBERT and RCNN Methods," vol. 14, pp. 2246–2257, 2025, doi: https://doi.org/10.32520/stmsi.v14i5.5392.

[11] S. Riyadi, L. K. Salsabila, C. Damarjati, and R. A. Karim, "Sentiment Analysis of YouTube Users on Blackpink Kpop Group Using IndoBERT," *INTENSIF J. Ilm. Penelit. dan Penerapan Teknol. Sist. Inf.*, vol. 8, no. 2, pp. 233–245, 2024, doi: 10.29407/intensif.v8i2.22678.

[12] M. Y. Ridho and E. Yulianti, "From Text to Truth: Leveraging IndoBERT and Machine Learning Models for Hoax Detection in Indonesian News," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 10, no. 3, pp. 544–555, 2024, doi: 10.26555/jiteki.v10i3.29450.

[13] A. A. Chamid, R. Nindyasari, N. Azizah, and A. Hariyadi, "Analysis of Public Opinion on The Governor Candidate Debate Using LDA and IndoBERT," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, no. 3, 2025, doi: 10.22219/kinetik.v10i3.2221.

[14] A. Z. Abdullah and E. B. Setiawan, "Sentiment Analysis Accuracy for 2024 Indonesian Election Tweets Using CNN-LSTM With Genetic Algorithm Optimization," *INTENSIF J. Ilm. Penelit. dan Penerapan Teknol. Sist. Inf.*, vol. 9, no. 1, pp. 1–14, 2025, doi: 10.29407/intensif.v9i1.22999.

[15] Sumit, S. Bisht, S. Joshi, and U. Rana, "Comprehensive Review of R-CNN and its Variant Architectures," *Int. Res. J. Adv. Eng. Hub*, vol. 2, no. 04, pp. 959–966, 2024, doi: 10.47392/irjaeh.2024.0134.

[16] T. Balasubramanian and R. Dr.S, "Deep Learning Approaches for Precision Detection, Optimized Classification, and Diagnostic Accuracy Enhancement in Pox Disease Using Transfer Learning," pp. 0–11, 2025, doi: 10.20944/preprints202504.2586.v1.

[17] O. Chamorro Atalaya *et al.*, "K-Fold Cross-Validation through Identification of the Opinion Classification Algorithm for the Satisfaction of University Students," *Int. J. Online Biomed. Eng.*, vol. 19, no. 9, pp. 122–130, 2023, doi: https://doi.org/10.1007/s41664-018-0068-2.

[18] H. Azis, F. Fattah, and P. Putri, "Performa Klasifikasi K-NN dan Cross-validation pada Data Pasien Pengidap Penyakit Jantung," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 81–86, 2020, doi: https://doi.org/10.33096/ilkom.v12i2.507.81-86.

[19] W. A. Hidayat and V. R. S. Nastiti, "Perbandingan Kinerja Pre-Trained IndoBERT-Base dan IndoBERT-Lite pada Klasifikasi Sentimen Ulasan TikTok Tokopedia Seller Center dengan Model IndoBERT," vol. 11, no. 2, pp. 13–20, 2024, doi: 10.30656/jsii.v11i2.9168.

[20] A. T. B. W and D. H. Fudholi, "Klasifikasi Emosi pada Teks dengan Menggunakan Metode Deep Learning," *J. Ilm. Indones.*, vol. 6, no. 1, 2021, doi: https://doi.org/10.36418/syntax-literate.v6i1.