

Sentiment Analysis of Tokopedia Customer Reviews Using BiLSTM and IndoBERT with Comparative Analysis of Preprocessing and Labeling Methods

Rahmi Anadra¹, Hari Wijayanto^{2*}, Kusman Sadik³

^{1,2,3}School of Statistics and Data Science, Mathematics, and Informatics, IPB University, Indonesia

Article Info

Article history:

Received Jul 26, 2025

Revised Oct 22, 2025

Accepted Nov 18, 2025

Keywords:

Sentiment analysis

BiLSTM

IndoBERT

Text preprocessing

Data labeling

ABSTRACT

This study addresses key challenges in Indonesian sentiment analysis related to preprocessing, labeling strategies, and class imbalance. It compares the performance of BiLSTM and IndoBERT using user reviews collected from Tokopedia. The dataset was manually and automatically labeled, then processed under three preprocessing schemes. Both models were trained with tuned hyperparameters and imbalance-handling techniques and evaluated through twenty rounds of stratified five-fold cross-validation. Performance was assessed using balanced accuracy and F1-score. IndoBERT achieved the highest results, with balanced accuracy up to 0.85 and F1-scores up to 0.83, while BiLSTM reached balanced accuracy up to 0.78 and F1-scores up to 0.76. Applying class weight and focal loss improved model performance by approximately 2% to 11% over the baseline. BiLSTM demonstrated greater training efficiency, requiring only 1 to 2.5 minutes per epoch, compared with IndoBERT's 2.6 to 3.6 minutes. Although manual labeling remained superior in capturing contextual nuance and emotional cues, GPT-based labeling showed strong agreement with the human annotations. A four-way ANOVA revealed that all main factors and several interactions significantly influenced classification outcomes. Overall, BiLSTM provides faster training efficiency, whereas IndoBERT delivers higher predictive accuracy.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Hari Wijayanto,

School of Statistics and Data Science, Mathematics, and Informatics, IPB University,

Meranti Street, IPB Dramaga Campus, Bogor 16680, West Java, Indonesia

Email: hari@apps.ipb.ac.id

1. INTRODUCTION

Indonesia is projected to experience the fastest global e-commerce growth by 2024 at 30.5%, nearly tripling the global average of 10.4% [1]. This rapid expansion has increased digital transactions and generated a large volume of user-generated reviews across platforms. Although mostly unstructured, these reviews strongly reflect customer opinions and influence purchase decisions [2], [3], making them valuable for assessing customer preferences, seller credibility, and product quality. However, their scale and linguistic variability make manual analysis challenging. Many reviews contain code-mixing, emojis, informal expressions, and abbreviations [4], underscoring the need for text analysis methods capable of capturing Indonesian-specific linguistic characteristics [5].

In the last years, sentiment analysis has obtained considerable interest within Natural Language Processing because of its potential ability for automatic opinion and attitude detection expressed in text [6], [7]. The practical contribution brought about by this method is that consumer perceptions will be better understood by service providers in a more constructive and systematic

way, hence smoothing the way for informed and data-driven decision-making. However, challenges persist due to the abundance of unstructured text and the limited availability of high-quality labeled datasets. Efforts have been made towards the development of the review methods of sentiment analysis to reflect local linguistic characteristics more effectively, a current drive in Indonesian-language NLP research [4], [5].

Recently, sentiment analysis has seen much development in deep learning architectures such as CNN and BiLSTM. This is because they deal with the task of representing sequences with many interdependencies, much like those found in text. Of these, BiLSTM has lately seen quite considerable attention due to its ability to process information from both past and future contexts—either way, giving a much better representation of sentence-level semantics compared to the traditional one-directional LSTM. Wu et al. [8], proved that the BiLSTM model performed much better than CNN when combined with FastText embeddings in the social media text classification. Similarly, Suhaeni et al. [9], reported that BiLSTM outperformed the performance of LSTM and GRU models as it showed higher accuracy with a higher F1-score on Indonesian political sentiment data during the 2024 election. This emphasizes that BiLSTM will be more suitable and effective for the customer reviews, which always contain a lot of informal and non-standard expressions.

The introduction of transformer-based architectures added BERT and its derivatives, one of which is IndoBERT, designed for the Indonesian language. IndoBERT has been performing well in Indonesian NLP tasks due to better captures of local linguistics compared to RNN-based and conventional models. Nissa and Yulianti [10], conducted an experiment on a multi-label customer review classification task and obtained up to 19.19% improvement in accuracy, whereas Baharuddin and Naufal [11], achieved the best accuracy of 97% in exam question classification. Therefore, IndoBERT remains at the state-of-the-art status for Indonesian-language applications, including the task at hand—sentiment analysis of e-commerce customer reviews.

Apart from model selection, text preprocessing is a crucial step in cleaning the input data by removing elements bearing little semantic value, such as stops, symbols, and non-standard words, to improve the model in catching more meaningful linguistic patterns. While transformer-based models generally tend to be more robust to noisy input [12], previous studies suggest that different preprocessing strategies might still influence the performance of classification, particularly for morphologically and lexically complex languages such as Indonesian [4], [13]. This challenge is even greater in e-commerce reviews, which may include slang, abbreviations, code-mixing, and emojis for emotional expressions. For this reason, this study evaluates three different preprocessing scenarios to investigate their impact on the sentiment analysis model's performance.

While preprocessing plays a critical role in sentiment analysis, the precision of data labeling is equally crucial for shaping model performance. Manual annotation is generally considered the most reliable method because human annotators can capture nuanced meaning, implicit intention, and emotional cues. However, it is time-consuming and resource-intensive. To address large-scale datasets, recent studies [14]–[17], recently proposed automated labeling methods using large pre-trained models like ChatGPT and XLM-RoBERTa. Previous works indeed suggested that an automated approach to solving such problems may achieve good approximations to human-level annotations although their effectiveness often depends on the linguistic characteristics and contextual complexity of datasets. Despite the interest in the subject, there is still a lack of systematic comparisons between manual and automatic labeling when it comes to Indonesian e-commerce reviews.

In sentiment analysis, the class imbalance problem is another big challenge faced due to the unbalanced population of dominance of positive reviews in the dataset, which often results in a reduced population of negative or neutral samples. This reduces the balanced accuracy and F1-score. Previous works, such as, resampling techniques, class weighting, and focal loss [16], [18], may boost these cases toward more equal classification, which have shown improvements in achieving more equitable classification results. However, this still remains relatively underexplored in the application of these methods in the comparative assessment with BiLSTM and IndoBERT within e-commerce sentiment analysis.

Although several Indonesian studies have compared deep learning models for sentiment analysis, most of them evaluate models separately or focus on limited components such as model architecture alone. This leaves a gap for a more integrated evaluation that considers the interaction between preprocessing, labeling, and class-imbalance handling. Building on this, the present study offers a more comprehensive comparison of BiLSTM and IndoBERT for Indonesian e-commerce sentiment analysis by examining three preprocessing strategies, manual and automatic labeling using GPT and XLM-RoBERTa, and two techniques for handling class imbalance. By combining these elements within a unified experimental design, this study provides a deeper understanding of the factors that influence model performance on Tokopedia reviews and offers practical insights for developing more robust e-commerce analytics systems.

2. RESEARCH METHOD

This study uses a comparative experimental approach to evaluate BiLSTM and IndoBERT on Indonesian e-commerce reviews collected from Tokopedia. The research follows a standard data mining framework covering the full workflow of sentiment analysis, from data preparation to model training and evaluation. An overview of this process is presented in Figure 1.

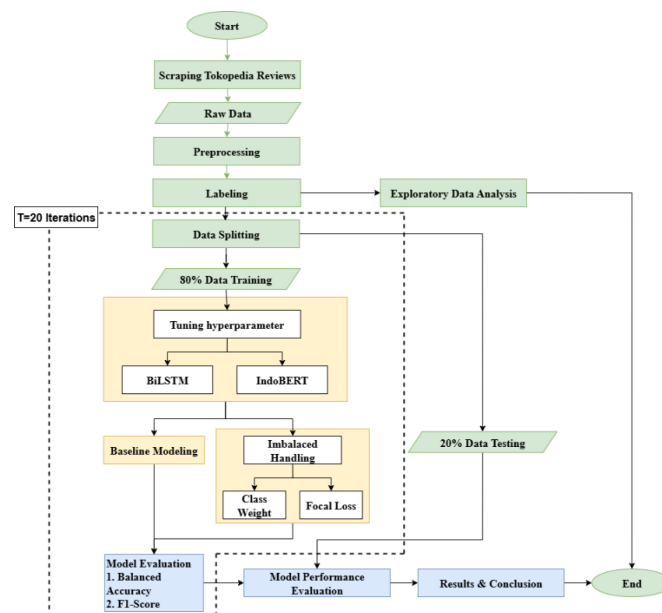


Figure 1. Flowchart of data analysis procedure

2.1. Dataset

Data collection in this study uses Indonesian customers reviews from the e-commerce website Tokopedia through a Python-based web scraping approach. The data were for 28 categories of products and review texts with their respective product names, identities of reviewers, and ratings to show customer experiences and views about the services offered. Data was collected in the year 2025, and 8943 reviews were prepared for analysis.

2.2. Preprocessing

Cleaning and standardization of Tokopedia customer reviews were conducted through several preprocessing steps. These steps included case folding to convert all text into lowercase, cleaning to remove symbols, numbers, and other irrelevant characters, normalization to convert informal or slang words into their standard forms, stopwords removal to eliminate less informative words, stemming to reduce words to their root forms, and tokenization to break the text into word-level units [19]–[21]. These procedures were applied to reduce noise and produce cleaner and more structured text that is more suitable for processing by sentiment analysis models [12].

To evaluate how preprocessing choices affect model performance, three preprocessing scenarios were implemented to represent different levels of text transformation. Scenario 1 applied

only case folding and tokenization, serving as a minimal-preprocessing baseline that preserves most linguistic and non-linguistic cues. Scenario 2 used the full preprocessing pipeline commonly adopted in Indonesian NLP studies, including emoji removal. Scenario 3 followed the same steps as Scenario 2 but retained emojis as separate tokens because they carry affective cues relevant for sentiment classification. Emojis were kept in their original form rather than converted into textual descriptors because they were intended to be processed directly by the model in later stages. These three scenarios together provide a controlled basis for assessing how text cleaning and the preservation of emoji-based emotional signals influence sentiment analysis outcomes. Examples of the processed text for each scenario are presented in Table 1.

Table 1. Preprocessing steps

Steps	Scenario 1	Scenario 2	Scenario 3
Row Data	barang2 yg dipesan diproses & dikirim dg cepat. diterima dalam keadaan masih beku👍👍		
Case folding	barang2 yg dipesan diproses & dikirim dg cepat. diterima dalam keadaan masih beku👍👍		
Cleaning	-	barang yg dipesan diproses dikirim dg cepat diterima dalam keadaan masih beku	barang yg dipesan diproses dikirim dg cepat diterima dalam keadaan masih beku👍👍
Normalization	-	barang yang dipesan diproses dikirim dengan cepat diterima dalam keadaan masih beku	barang yang dipesan diproses dikirim dengan cepat diterima dalam keadaan masih beku👍👍
Stopword Removal	-	barang dipesan diproses dikirim cepat diterima beku	barang dipesan diproses dikirim cepat diterima beku👍👍
Stemming	-	barang pesan proses kirim cepat terima beku	barang pesan proses kirim cepat terima beku👍👍
Tokenization	['barang', '2', 'yg', 'dipesan', 'diproses', '&', 'dikirim', 'dg', 'cepat', 'diterima', 'dalam', 'keadaan', 'masih', 'beku', '👍', '👍']	['barang', 'pesan', 'proses', 'kirim', 'cepat', 'terima', 'beku']	['barang', 'pesan', 'proses', 'kirim', 'cepat', 'terima', 'beku', ' ', '👍', '👍']

TweetTokenizer from NLTK was used in Scenario 1 and Scenario 3 because it preserves non-alphabetic characters such as numbers, punctuation, symbols, and emojis during tokenization. These elements therefore appear as separate tokens in both scenarios in accordance with the intended preprocessing design.

2.3. Labeling

Labeling was conducted by assigning each review to one of three sentiment categories: positive, neutral, or negative, using two approaches. The first approach was manual annotation, which served as the gold standard because human annotators are generally better at capturing contextual meaning and emotional cues. Manual labeling was performed by the researcher, who has an academic background in Statistics and Data Science with training in natural language processing. A structured guideline was applied: reviews expressing satisfaction were labeled positive, those containing complaints were labeled negative, and descriptive reviews without clear polarity were labeled neutral. To maintain consistency, the annotator conducted a second-pass review of all labels to reassess ambiguous cases, serving as an internal quality-control mechanism commonly used in single-annotator studies.

The second approach consisted of automatic labeling using ChatGPT [15], and XLM-RoBERTa [22], chosen for their efficiency in processing large datasets. These automatic labels were not intended to replace the manual labels but were used to assess their alignment with the gold standard. Because manual annotation was carried out by only one annotator, inter-human agreement metrics could not be computed. Instead, reliability was evaluated by calculating Cohen's Kappa between the manual and automatically generated labels, as reported in the Results section. This procedure aligns with recent practices for assessing the reliability of automatic labeling methods through comparison with human annotation. Because the manual labeling was conducted by a single annotator, inter-human agreement metrics could not be computed. Nevertheless, labeling consistency was maintained through repeated self-review, and Cohen's Kappa was calculated between the manual and automatic labels as an alternative indicator of reliability.

2.4. BiLSTM

BiLSTM enhances the traditional LSTM by analyzing text sequences in both forward and backward directions to serve better contextual information inside a sentence. For the study at hand, BiLSTM has been implemented using combined 300-dimensional FastText and 300-dimensional Emoji2Vec embeddings, which produce a 600-dimensional input representation. Emojis retained during preprocessing were represented using Emoji2Vec, enabling the BiLSTM model to incorporate emoji-based affective cues as part of its input features. Hence, the model consists of an embedding layer, a BiLSTM layer, a dense layer, and a softmax output layer for classifying reviews into positive, neutral, or negative sentiments. Training was conducted using optimized hyperparameters, including batch size, learning rate, dropout rate, and a maximum of 15 epochs, with early stopping (patience = 3) to prevent overfitting.

Mathematically, the hidden state in a BiLSTM is computed by concatenating the forward LSTM hidden state ($h_t^{forward}$) and the backward LSTM hidden state ($h_t^{backward}$), as expressed in the following equation [23]:

$$h_t^{bilstm} = h_t^{forward} \oplus h_t^{backward} \quad (1)$$

The architecture of the BiLSTM model used in this study is illustrated in Figure 2 [24]. With its ability to capture bidirectional contextual information and utilize rich embedding representations, BiLSTM is expected to improve sentiment classification accuracy on Tokopedia customer reviews.

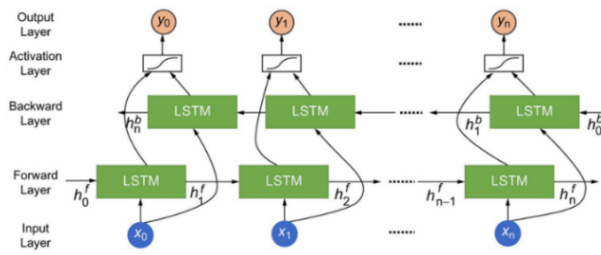


Figure 2. Architecture of the BiLSTM model

2.5. IndoBERT

The Bidirectional Encoder Representations from Transformers (BERT) is a transformer-based model designed to produce bidirectional contextual language representations [25]–[27]. IndoBERT is an adaptation of this architecture, pretrained on a large Indonesian corpus consisting of Wikipedia, news articles, and various web texts, enabling it to better capture Indonesian linguistic characteristics [28]. The model follows the BERT-base configuration with 12 encoder layers, 12 attention heads, and a 768-dimensional hidden size [29]. In this study, the *indobenchmark/indobert-base-pl* model was fine-tuned on the Tokopedia review dataset by adding a dense–softmax classification layer on top of the pretrained encoder. For scenarios that retained emojis, the IndoBERT vocabulary was expanded by adding emoji tokens so that these symbols were represented explicitly rather than mapped to unknown tokens. This extension ensured that emojis received trainable embeddings during fine-tuning and contributed to the model’s contextual understanding of sentiment. The model was trained using the AdamW optimizer with tuned learning rate, batch size, and dropout values for up to five epochs, with early stopping (patience = 2). Figure 3 illustrates the IndoBERT architecture, where the [CLS] token embedding is passed through a linear layer and softmax function to generate the three sentiment classes.

Training was done with an AdamW optimizer with tuned learning rate, batch size, and dropout values, a maximum of five epochs with early stopping on patience value two. Figure 3 overall architecture. The overall architecture of IndoBERT is depicted in the figure by passing [CLS] token representation through a linear layer followed by a softmax function that classifies reviews into three categories as per sentiment.

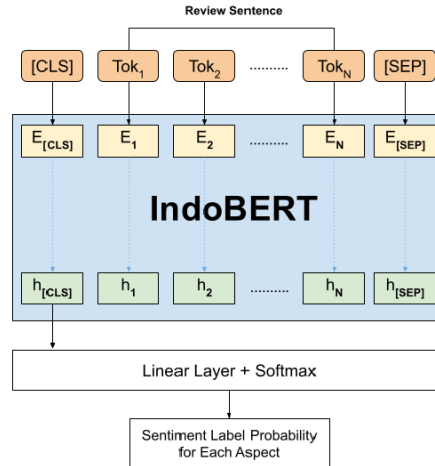


Figure 3. Architecture of the IndoBERT model

2.6. Handling Class Imbalanced

Class imbalance is a common problem in machine learning tasks, including sentiment analysis, where models often get biased toward one majority class, while underrepresenting the minority classes that can potentially represent valuable information [30]. Therefore, two strategies concerning loss were implemented to handle this issue.

1. Class Weight

Instead, this approach changes the loss function by weighting up the minority classes so that a misclassification in these classes weighs a lot more during training. The class weight is generally determined by the following formula:

$$w_c = \frac{N}{n_c \times C} \quad (2)$$

In this formula, N denotes the total number of samples, n_c represents the number of samples in class c , and C is the total number of classes [31]. This weighting scheme gives more importance to misclassification in minority classes, encouraging the model to learn more coherently over all categories.

2. Focal Loss

Focal loss is a variant of the cross-entropy loss which has built into it a focusing factor that delves lesser on easily classified samples, hence putting more emphasis on harder examples or underrepresented classes [32]. For multi-class classification with the number of classes $C > 2$, the formula is expressed as follows:

$$\mathcal{L}_{FC} = -\sum_{i=1}^C \alpha_i (1 - p_{t,i})^\gamma \log(p_{t,i}) \quad (3)$$

In this formulation, $p_{t,i}$ denotes the predicted probability of the correct class i , α_i is the balancing factor for each class, and γ is the focusing parameter that emphasizes hard-to-classify samples [33]. This mechanism encourages the model to focus more on minority or infrequent samples, leading to more balanced classification performance across sentiment classes.

2.7. Evaluation Metrics

In classification tasks, system performance is typically evaluated using metrics derived from the confusion matrix. In this study, these metrics assess both the quality of automatic labels generated by GPT and XLM-RoBERTa compared with manual annotations, and the performance of the final classification models. For multiclass classification, the confusion matrix forms an $N \times N$ table, where N is the number of classes [34], with evaluation based on True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). While accuracy reflects the proportion of correct predictions, it is insufficient under class imbalance; therefore, precision, recall, F1-score, and balanced accuracy are included. Macro-averaging is applied because it

weights all classes equally and is better suited for imbalanced data, whereas micro-averaging behaves similarly to accuracy and is less sensitive to minority classes [35].

a. Accuracy

Accuracy is the ratio of correctly predicted instances in the overall dataset:

$$\text{Accuracy} = \frac{\sum_{i=1}^3 C_{i,i}}{\sum_{i=1}^3 \sum_{j=1}^3 C_{i,j}} \quad (4)$$

b. Balanced Accuracy

Balanced accuracy defines the average recall of all classes and is suitable for model performance evaluation in imbalanced datasets:

$$\text{Balanced Accuracy} = \frac{1}{N} \sum_{i=1}^3 \frac{TP_i}{TP_i + FN_i} \quad (5)$$

c. Precision

Precision is the ratio of truly predicted positive instances by all instances that were predicted as positive for each class:

$$\text{Precision} = \frac{1}{N} \sum_{i=1}^3 \frac{TP_i}{TP_i + FP_i} \quad (6)$$

d. Sensitivity

Sensitivity reflects the model's ability to correctly identify real positive cases:

$$\text{Recall} = \frac{1}{N} \sum_{i=1}^3 \frac{TP_i}{TP_i + FN_i} \quad (7)$$

e. F1-Score

F1-score in general is the harmonic average of precision and recall:

$$\mathbf{F1 - Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

2.8 Analysis of Variance (ANOVA)

ANOVA is a parametric statistical method used to determine whether the mean values of two or more groups differ significantly. It partitions total variation into between-group and within-group components and compares them using an F-statistic. Depending on the number of factors, ANOVA can be conducted as one-way, two-way, or extended to multi-factor designs, enabling simultaneous evaluation of multiple effects. The calculated F-value at a chosen significance level is compared with a critical value; if it exceeds the critical value, the null hypothesis of equal group means is rejected [36].

3. RESULTS AND DISCUSSION

3.1. Preprocessing

The preprocessing step cleaned the text but also decided on the final number of valid samples. In all 8,943 reviews remained for Scenario 1 because only case folding was used. In Scenario 2, cleaning removed invalid or empty texts, which reduced the dataset by 55 to 8,888 reviews. Scenario 3 resulted in 8,922 reviews because emojis were retained while the rest were removed. Table 2 shows a detailed data distribution for each scenario.

Table 2. Comparison in volume by different preprocessing scenarios

Preprocessing Scenario	Data Volume
1	8943
2	8888
3	8922

3.2. Labeling

Labeling was into three sentiment classes, namely positive, neutral, and negative, which was constant for all preprocessing scenarios. The gold standard was manual since it could give very contextual and highly accurate annotations. Additionally, there was automatic labeling that was performed for comparison purposes through GPT using the zero-shot prompt method and XLM-RoBERTa, a multilingual contextual representation.

The positive sentiment is always the most dominant category in all scenarios, while the negative sentiment is the least represented, as presented in Table 3. On one hand, GPT generated a label distribution closer to the manual ones, especially preserving the share of neutral sentiment.

On the other hand, XLM-RoBERTa is predisposed to generating fewer positive labels while increasing the share of neutral and even the negative labels in some cases. These results confirm once more the status of manual labeling as a main reference, and automatic labeling methods as useful efficient alternatives if the class distribution shift is carefully analyzed.

Table 3. Data labeling results by preprocessing scenario

Preprocessing Scenario	Labeling Method	Label			Total
		Positive	Neutral	Negative	
1	Manual	77.64%	12.80%	9.56%	8943
	GPT	76.07%	12.77%	11.17%	
	XLM-RoBERTa	75.05%	12.58%	12.37%	
2	Manual	75.69%	14.05%	10.27%	8888
	GPT	71.72%	15.93%	12.36%	
	XLM-RoBERTa	63.59%	24.87%	11.54%	
3	Manual	76.69%	13.13%	10.16%	8922
	GPT	73.20%	14.46%	12.35%	
	XLM-RoBERTa	65.75%	22.91%	11.32%	

In overall interpretation, the results show that GPT-based labeling is more robustly consistent in its approximation to manual labeling in all scenarios of pre-processing compared to XLM-RoBERTa. This trend in the results is undergirded by the higher values of accuracy, F1-Score, and Cohen's Kappa that GPT shows consistently, reflecting stronger agreement with manual labeling and overall predictive reliability. GPT's performance is at its best in scenario 1, although slight degradations have been observed in scenarios 2 and 3, which remain relatively minor and do not compromise the overall stability of the model. In contrast, larger fluctuations across scenarios demonstrate XLM-RoBERTa's higher sensitivity to changes in preprocessing. The summary of detailed evaluation metrics across the scenarios and comparisons is contained in Table 4 for labeling.

Table 4. Comparison of evaluation metrics between manual and automatic labeling across preprocessing scenarios

Labeling Method	Preprocessing Scenario	Evaluation Metrics					
		Accuracy	Bal. Accuracy	Precision	Recall	F1-Score	Cohen's Kappa
Manual vs GPT	1	0.9356	0.8985	0.8585	0.8985	0.8767	0.8316
	2	0.9220	0.9020	0.8333	0.9020	0.8639	0.8153
	3	0.9262	0.8989	0.8329	0.8989	0.8623	0.8188
Manual vs XLM-RoBERTa	1	0.8281	0.7069	0.6570	0.7069	0.6778	0.5586
	2	0.7606	0.7199	0.6555	0.7199	0.6728	0.4927
	3	0.7789	0.7298	0.6637	0.7298	0.6829	0.5128

3.3. Exploratory Data Analysis

In this direction, the exploratory data analysis was the word cloud, highlighting the most frequently used terms when customers left reviews about Tokopedia, hence giving some first ideas about the main linguistic patterns that dominate in this dataset.



Figure 4. Compare word cloud (a) before preprocessing and (b) after preprocessing

Figure 4 compares the word clouds before and after preprocessing. In the raw text of (a), there were still many common words, colloquial expressions, abbreviations, and non-alphabetic

characters present, which created noise in the dataset. After preprocessing, as shown in (b), the words became cleaner, more consistent, and more representative, and most customer-related terms such as *barang*, *bagus*, and *sesuai* received higher weights, indicating that the text became more suitable for sentiment analysis.

Further from the text-based word cloud, there is an emoji cloud based on the analysis of visual symbols usage. Indeed, as seen in Figure 5 above, the positive emojis such as thumbs-up, hearts, and smiling faces were the most dominant, indicating high levels of customer satisfaction. The negativity was used less, further cementing the notion that the sentiment of the reviews was mostly positive.

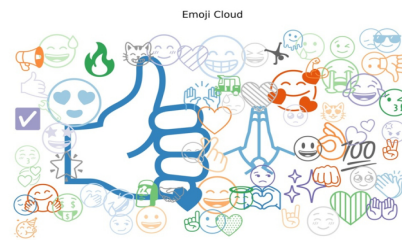


Figure 5. Emoji cloud

3.4. Modeling

This work investigates the performance of three different preprocessing scenarios on two key architectures, namely BiLSTM and IndoBERT. Each model was trained under two configurations, namely baseline and class balancing, using class weight and focal loss, making six experimental variations for each scenario. In this work, we labeled our dataset with three approaches, consisting of manual annotation as the gold standard and automatic labeling with GPT and XLM-RoBERTa. Model performance was measured by using balanced accuracy and macro F1-score, which gives a more representative setting for an imbalanced dataset than overall accuracy. The best hyperparameters used in the experiments have been tuned using grid search and are presented in Table 5.

Table 5. Best tuned hyperparameters for the models of BiLSTM and IndoBERT

Parameter	BiLSTM	IndoBERT
Bacth Size	16	32
Learning Rate	0.0001	0.000005
Dropout	0.3	0.3
Unit	64	-
Epoch	15 (patience=3)	5 (patience=2)

Two widely used architectures were considered, namely BiLSTM and IndoBERT, for which hyperparameters were tuned. Experiments were conducted by considering three preprocessing scenarios, three labeling methods, and two class-balancing techniques. To ensure the reliability of the results, each configuration ran 20 times with stratified fivefold cross-validation. The performance of the models was computed by using the balanced accuracy and F1-score, whereas the obtained results are presented as boxplots to provide a clear view of the consistency across different configurations.

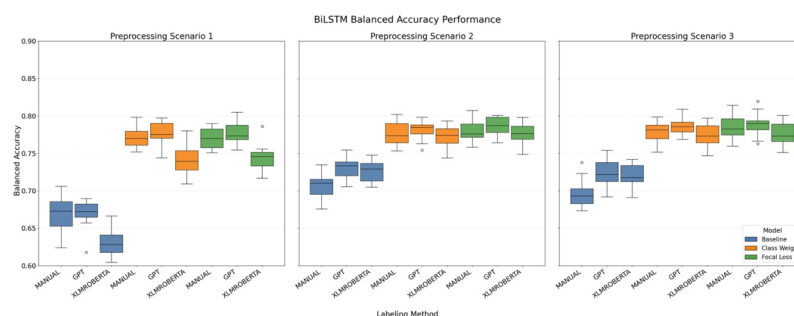


Figure 6. Model performance of BiLSTM based on balanced accuracy

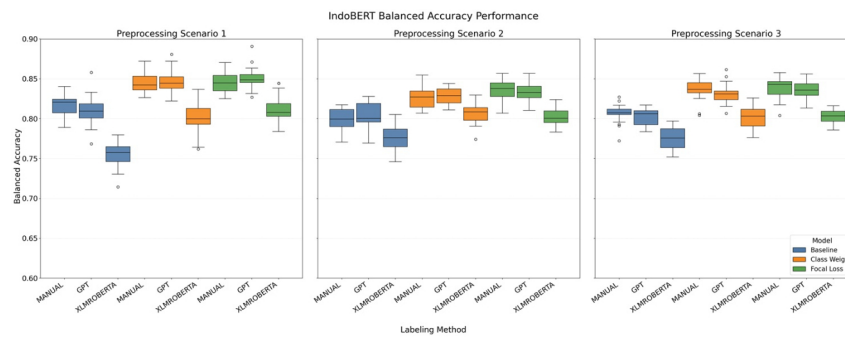


Figure 7. Model performance of IndoBERT based on balanced accuracy

Figures 6 and 7 show that IndoBERT consistently achieves higher balanced accuracy and lower variance than BiLSTM across all preprocessing and labeling configurations. This pattern indicates that IndoBERT is more robust to noisy text, labeling inconsistencies, and class imbalance because its contextual embeddings can better capture semantic cues. In contrast, BiLSTM exhibits lower accuracy and wider spread, especially under the baseline setting, suggesting that the model is more easily affected by minority-class sparsity and the loss of contextual information. Both class weight and focal loss improve BiLSTM performance by reducing its bias toward the majority class, although IndoBERT still maintains superior stability. Performance also varies by preprocessing scenario; results are generally higher when sentiment-bearing cues such as emojis are preserved, while more aggressive cleaning reduces accuracy for both models because important emotional and lexical signals are removed.

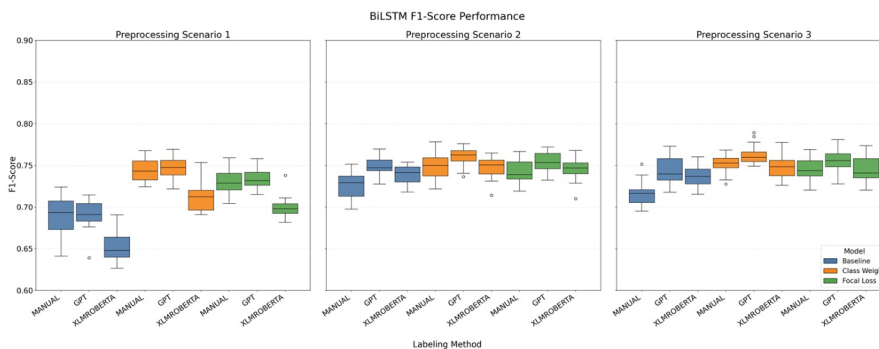


Figure 8. Model performance of BiLSTM based on F1-score

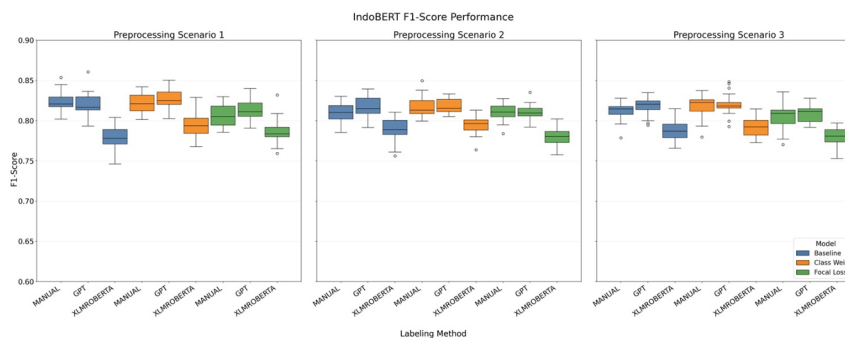


Figure 9. Model performance of IndoBERT based on F1-score

Figures 8 and 9 show a pattern consistent with the balanced accuracy results. IndoBERT maintains higher and more stable F1-scores across all preprocessing and labeling configurations, indicating its ability to capture minority-class signals more effectively. BiLSTM exhibits lower F1-scores and larger variability, particularly under the baseline condition, reflecting its difficulty in

identifying minority sentiments when contextual information is reduced. The use of class weight and focal loss improves BiLSTM substantially by reducing its majority-class bias, although IndoBERT continues to outperform it overall. Performance also tends to be higher when emojis are preserved, suggesting that sentiment-bearing cues enhance the models ability to correctly classify low-frequency negative and neutral reviews.

Quantitatively, the performance of both models is summarized in Table 6, which provides average performance, in terms of balanced accuracy and F1-score, each combination of preprocessing scenario, labeling approach, and classbalancing method. Every value represents the mean of the 20 experimental iterations and becomes more reliable for comparisons across different configurations.

Table 6. Balanced accuracy and F1-score results of the BiLSTM model

Labeling Method	Imbalance Handling	Preprocessing Scenario 1		Preprocessing Scenario 2		Preprocessing Scenario 3	
		Bal. Acc	F1-Score	Bal. Acc	F1-Score	Bal. Acc	F1-Score
Manual	Baseline	0.6695	0.6895	0.6695	0.7246	0.6952	0.7167
	Class Weight	0.7713	0.7442	0.7713	0.7486	0.7791	0.7519
	Focal Loss	0.7710	0.7304	0.7710	0.7424	0.7844	0.7463
GPT	Baseline	0.6714	0.6909	0.6714	0.7497	0.7234	0.7426
	Class Weight	0.7762	0.7475	0.7762	0.7602	0.7868	0.7624
	Focal Loss	0.7766	0.7344	0.7766	0.7540	0.7894	0.7556
XLM-RoBERTa	Baseline	0.6297	0.6527	0.6297	0.7394	0.7210	0.7363
	Class Weight	0.7404	0.7109	0.7404	0.7477	0.7741	0.7490
	Focal Loss	0.7435	0.6991	0.7435	0.7446	0.7758	0.7447

Table 6 summarizes the quantitative performance of the BiLSTM model across all experimental configurations. The baseline consistently yields the lowest balanced accuracy and F1-scores, particularly under XLM-RoBERTa labeling, confirming that BiLSTM is highly sensitive to class imbalance and reductions in contextual clarity. Applying class weight and focal loss substantially improves performance, increasing average balanced accuracy from approximately 0.67 to around 0.77–0.79 across scenarios. Scenario 3 produces the highest scores, indicating that preserving emojis helps BiLSTM capture affective cues that support sequential modeling. Overall, the numerical results reinforce the pattern observed in the boxplots and show that BiLSTM benefits strongly from imbalance-handling techniques and performs better when emotional and lexical cues are retained during preprocessing.

Table 7. Balanced accuracy and F1-score results of the IndoBERT model

Labeling Method	Imbalance Handling	Preprocessing Scenario 1		Preprocessing Scenario 2		Preprocessing Scenario 3	
		Bal. Acc	F1-Score	Bal. Acc	F1-Score	Bal. Acc	Bal. Acc
Manual	Baseline	0.8165	0.8238	0.7986	0.8096	0.8064	0.8116
	Class Weight	0.8453	0.8223	0.8260	0.8173	0.8365	0.8186
	Focal Loss	0.8461	0.8065	0.8358	0.8102	0.8392	0.8047
GPT	Baseline	0.8098	0.8202	0.8048	0.8177	0.8023	0.8176
	Class Weight	0.8458	0.8275	0.8280	0.8182	0.8315	0.8206
	Focal Loss	0.8510	0.8132	0.8337	0.8103	0.8355	0.8089
XLM-RoBERTa	Baseline	0.7548	0.7786	0.7761	0.7884	0.7744	0.7882
	Class Weight	0.8003	0.7937	0.8064	0.7947	0.8018	0.7928
	Focal Loss	0.8120	0.7864	0.8031	0.7802	0.8031	0.7800

Table 7 summarizes the quantitative performance of the IndoBERT model across all configurations. The baseline already achieves strong results, with balanced accuracy and F1-scores around 0.81 and 0.82, showing that IndoBERT can capture sentiment cues effectively even without imbalance handling. Class weight and focal loss provide small but consistent improvements, raising balanced accuracy to approximately 0.84–0.85 across scenarios and confirming IndoBERT's lower sensitivity to class imbalance. Scenario 1 produces the highest scores, indicating that minimal preprocessing and the preservation of original cues such as emojis allow IndoBERT to retain richer contextual information. Overall, the numerical results reflect the boxplot trends: IndoBERT maintains stable and high performance across configurations and benefits most when linguistic and affective signals are preserved.

The results also show that GPT-based labeling achieves high agreement with manual annotations for both models, although manual labeling remains the main reference in this study. This is consistent with the findings of [37], which indicate that automatic labeling can, in some cases, produce classification performance comparable to or even better than manually annotated data. Nonetheless, performance metrics mainly capture technical alignment and do not fully reflect semantic nuances or contextual fidelity. Manual annotation retains advantages in interpreting subtle emotional cues, linguistic patterns, and implicit meaning that automatic methods may overlook. Therefore, although automated labeling is efficient and scalable for large datasets, the manually annotated labels provide a deeper and more contextually grounded reference and are used as the primary benchmark in evaluating model performance.

Figures 10 and 11 illustrate the distribution of computation time per epoch for BiLSTM and IndoBERT. BiLSTM shows the lowest and most efficient training time, reflecting its lighter sequential architecture and CPU execution. IndoBERT requires higher training time, which is consistent with the computational demands of transformer-based contextual encoding even when optimized on GPU. For BiLSTM, the second preprocessing scenario trains faster than scenarios 1 and 3, indicating that more homogeneous and standardized text allows sequential models to update parameters more efficiently. In contrast, IndoBERT displays relatively stable computation times across all configurations, suggesting that its computational cost is governed mainly by internal subword modeling rather than by differences in preprocessing. Overall, BiLSTM is more efficient in terms of training time, while IndoBERT offers stable computational behavior aligned with its richer representational capacity.

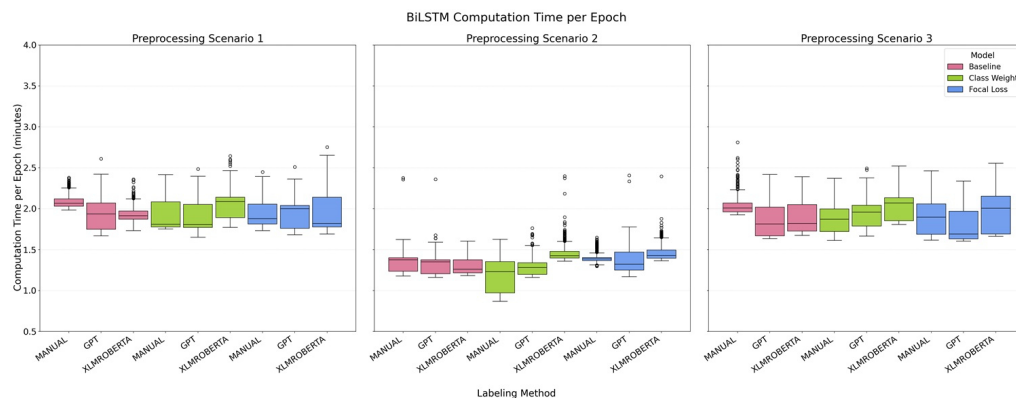


Figure 10. Training time distribution per epoch for BiLSTM

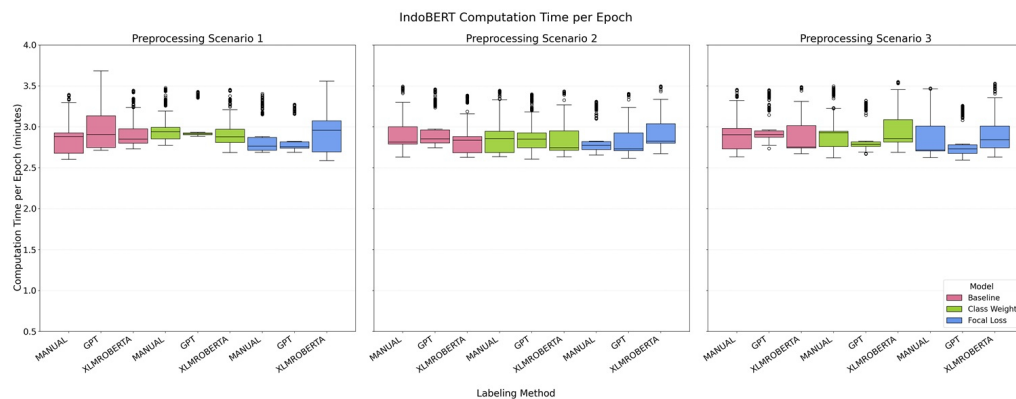


Figure 11. Training time distribution per epoch for IndoBERT

Coming full circle and supporting the above visual analysis, Table 8 gives the average training time per epoch for every model at various combinations of preprocessing scenarios, labeling strategies, and imbalance handling techniques.

Table 8. Average train time per epoch

Labeling Method	Imbalance Handling	Training Time (Minutes)					
		Preprocessing Scenario 1		Preprocessing Scenario 2		Preprocessing Scenario 3	
		BiLSTM	IndoBERT	BiLSTM	IndoBERT	BiLSTM	IndoBERT
Manual	Baseline	2.087	2.864	1.352	2.907	2.035	2.917
	Class Weight	1.923	2.990	1.197	2.883	1.877	2.926
	Focal Loss	1.936	2.853	1.397	2.853	1.903	2.879
GPT	Baseline	1.936	2.951	1.323	2.948	1.857	2.975
	Class Weight	1.893	3.008	1.289	2.887	1.940	2.869
	Focal Loss	1.952	2.847	1.365	2.852	1.788	2.809
XLM-RoBERTa	Baseline	1.928	2.948	1.298	2.857	1.887	2.909
	Class Weight	2.044	2.939	1.454	2.857	2.040	2.968
	Focal Loss	1.946	2.938	1.459	2.934	1.980	2.913

Table 8 shows the average training time per epoch for all configurations and supports the patterns observed in the boxplots. BiLSTM consistently trains faster, with times ranging from roughly 1.2 to 2.1 minutes per epoch, and scenario 2 tends to run the quickest because the text becomes more homogeneous after full cleaning. IndoBERT, which was trained with GPU acceleration, exhibits highly stable training times of about 2.85 to 3.01 minutes across all scenarios and labeling methods. This stability reflects the transformer's parallelized computation, where runtime is influenced more by model structure than by variations in text preprocessing. The use of class weight or focal loss does not substantially increase training time for either model because these techniques adjust gradient weighting without altering the overall computational graph. In summary, BiLSTM remains more time-efficient, whereas IndoBERT is computationally heavier but maintains consistent training speed across preprocessing, labeling, and imbalance-handling configurations, reinforcing the robustness patterns previously observed in the performance metrics.

To check the observed performances were statistically significant across configurations or not, a four-way ANOVA was performed. The result from this is given in Table 9.

Table 9. Results of Four-Way ANOVA

Source of Variation	df	F-Statistic	p-value
Model	1	6035.51	0.000*
Preprocessing	2	92.22	0.000*
Labeling	2	350.32	0.000*
Balancing	2	1833.67	0.000*
Model × Preprocessing	2	218.39	0.000*
Model × Labeling	2	85.40	0.000*
Preprocessing × Labeling	4	44.07	0.000*
Model × Balancing	2	297.88	0.000*
Preprocessing × Balancing	4	48.43	0.000*
Labeling × Balancing	4	1.94	0.102
Model × Preprocessing × Labeling	4	5.73	0.000*
Model × Preprocessing × Balancing	4	19.32	0.000*
Model × Labeling × Balancing	4	5.36	0.000*
Preprocessing × Labeling × Balancing	8	6.56	0.000*
Model × Preprocessing × Labeling × Balancing	8	0.74	0.657

*) Significant at the 5% level

The four-way ANOVA results in Table 9 show that all main factors (classification model, preprocessing strategy, labeling method, and imbalance-handling technique) have significant effects on balanced accuracy ($p < 0.05$). The model is the strongest factor, followed by the balancing technique, indicating that architectural characteristics and the approach used to address class imbalance are the primary drivers of performance, while preprocessing and labeling also play meaningful roles. Several two-way interactions, including Model × Preprocessing, Model × Labeling, and Model × Balancing, are significant, which means that the effectiveness of these components depends on the model applied. The Labeling × Balancing interaction is not significant, showing that these two factors do not produce a combined effect. Some three-way interactions, particularly Model × Preprocessing × Balancing, are significant and indicate that the best outcomes occur when these elements align properly. The four-way interaction is not significant, suggesting that the combined influence of all factors does not add variation beyond the effects already captured by the lower-order interactions.

4. CONCLUSION

This study achieved its objectives by evaluating the effects of preprocessing strategies, labeling approaches, and imbalance-handling techniques on the performance of BiLSTM and IndoBERT for Indonesian e-commerce sentiment classification. The results showed that IndoBERT provided stronger and more stable performance across configurations, particularly when sentiment-bearing cues such as emojis and informal expressions were preserved. BiLSTM remained more computationally efficient and achieved competitive results when supported by class-balancing techniques. GPT-based automatic labeling yielded high agreement with manual annotations, although manual labeling remains more reliable for capturing contextual nuance. The four-way ANOVA confirmed that all main factors, along with several two- and three-way interactions, significantly influenced classification outcomes, indicating that performance depends on how model architecture, preprocessing decisions, and imbalance-handling methods align with the linguistic properties of user-generated text. Future work may explore hybrid or domain-adaptive architectures and extend the dataset to other review platforms to improve generalization and practical applicability.

REFERENCES

- [1] E. F. Santika, "ECDB: Proyeksi Pertumbuhan e-Commerce Indonesia Tertinggi Sedunia pada 2024," *Databoks*, 2024. <https://databoks.katadata.co.id/datapublish/2024/04/29/ecdb-proyeksi-pertumbuhan-e-commerce-indonesia-tertinggi-sedunia-pada-2024> (accessed May 17, 2024).
- [2] T. Chen, P. Samaranayake, X. Y. Cen, M. Qi, and Y. C. Lan, "The Impact of Online Reviews on Consumers' Purchasing Decisions: Evidence From an Eye-Tracking Study," *Front. Psychol.*, vol. 13, no. June, 2022, doi: 10.3389/fpsyg.2022.865702.
- [3] Y. Ahn and J. Lee, "The Impact of Online Reviews on Consumers' Purchase Intentions: Examining the Social Influence of Online Reviews, Group Similarity, and Self-Construal," *J. Theor. Appl. Electron. Commer. Res.*, vol. 19, no. 2, pp. 1060–1078, 2024, doi: 10.3390/jtaer19020055.
- [4] C. H. Lin and U. Nuha, "Sentiment analysis of Indonesian datasets based on a hybrid deep-learning strategy," *J. Big Data*, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00782-9.
- [5] L. Wang, G. Che, J. Hu, and L. Chen, "Online Review Helpfulness and Information Overload: The Roles of Text, Image, and Video Elements," *J. Theor. Appl. Electron. Commer. Res.*, vol. 19, no. 2, pp. 1243–1266, 2024, doi: 10.3390/jtaer19020064.
- [6] K. L. Tan, C. P. Lee, and K. M. Lim, "A Survey of Sentiment Analysis: Approaches, Datasets, and Future Research," *Appl. Sci.*, vol. 13, no. 7, 2023, doi: 10.3390/app13074550.
- [7] R. N. Amalia, K. Sadik, and K. A. Notodiputro, "a Preliminary Study of Sentiment Analysis on Covid-19 News: Lesson Learned From Data Acquisition, Pre-Processing, and Descriptive Analytics," *Barekeng*, vol. 17, no. 4, pp. 1901–1914, 2023, doi: 10.30598/barekengvol17iss4pp1901-1914.
- [8] M. Wu, Z. Wen, P. Xiao, and W. Li, "Performance Comparison of CNN and BiLSTM with FastText Word Embeddings for Chinese Weibo Sentiment Analysis," vol. 13, no. 7, pp. 100–106, 2025.
- [9] C. Suhaeni, H. Wijayanto, and A. Kurnia, "Sentiment Classification on the 2024 Indonesian Presidential Candidate Dataset Using Deep Learning Approaches," *Indones. J. Stat. Its Appl.*, vol. 8, no. 2, pp. 83–94, 2024, doi: 10.29244/ijsa.v8i2p83-94.
- [10] N. K. Nissa and E. Yulianti, "Multi-label text classification of Indonesian customer reviews using bidirectional encoder representations from transformers language model," *Int. J. Electr. Comput. Eng.*, vol. 13, no. 5, pp. 5641–5652, 2023, doi: 10.11591/ijece.v13i5.pp5641-5652.
- [11] F. Baharuddin and M. F. Naufal, "Fine-Tuning IndoBERT for Indonesian Exam Question Classification Based on Bloom's Taxonomy," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 9, no. 2, pp. 253–263, 2023, doi: 10.20473/jisebi.9.2.253-263.

- [12] M. Siino, I. Tinnirello, and M. La Cascia, "Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers," *Inf. Syst.*, vol. 121, no. March 2023, p. 102342, 2024, doi: 10.1016/j.is.2023.102342.
- [13] M. A. Palomino and F. Aider, "Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis," *Appl. Sci.*, vol. 12, no. 17, 2022, doi: 10.3390/app12178765.
- [14] S. Biswas, K. Young, and J. Griffith, "A Comparison of Automatic Labelling Approaches for Sentiment Analysis," no. Data, pp. 312–319, 2022, doi: 10.5220/0011265900003269.
- [15] F. Gilardi, M. Alizadeh, and M. Kubli, "ChatGPT outperforms crowd workers for text-annotation tasks," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 120, no. 30, pp. 1–3, 2023, doi: 10.1073/pnas.2305016120.
- [16] A. N. Ma'aly, D. Pramesti, A. D. Fathurahman, and H. Fakhurroja, "Exploring Sentiment Analysis for the Indonesian Presidential Election Through Online Reviews Using Multi-Label Classification with a Deep Learning Algorithm," *Inf.*, vol. 15, no. 11, pp. 1–33, 2024, doi: 10.3390/info15110705.
- [17] Y. Asri, D. Kuswardani, W. N. Suliyanti, Y. O. Manullang, and A. R. Ansyari, "Sentiment analysis based on Indonesian language lexicon and IndoBERT on user reviews PLN mobile application," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 38, no. 1, p. 677, 2025, doi: 10.11591/ijeecs.v38.i1.pp677-688.
- [18] T. Cai and X. Zhang, "Multi - Channel BLTCN - BLSTM Self - Attention," *Sensors*, pp. 2–15, 2023.
- [19] M. F. Juna and M. Hayaty, "The observed preprocessing strategies for doing automatic text summarizing," *Comput. Sci. Inf. Technol.*, vol. 4, no. 2, pp. 119–126, 2023, doi: 10.11591/csit.v4i2.pp119-126.
- [20] M. A. Rosid, A. S. Fitriani, I. R. I. Astutik, N. I. Mulloh, and H. A. Gozali, "Improving Text Preprocessing for Student Complaint Document Classification Using Sastrawi," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 874, no. 1, 2020, doi: 10.1088/1757-899X/874/1/012017.
- [21] L. Zhu and D. Luo, "A Novel Efficient and Effective Preprocessing Algorithm for Text Classification," *J. Comput. Commun.*, vol. 11, no. 03, pp. 1–14, 2023, doi: 10.4236/jcc.2023.113001.
- [22] A. Conneau *et al.*, "Unsupervised Cross-lingual Representation Learning at Scale," 2020.
- [23] G. Xu, Z. Chen, and Z. Zhang, "Aspect category sentiment analysis based on pre-trained BiLSTM and syntax-aware graph attention network," *Sci. Rep.*, vol. 15, no. 1, pp. 1–15, 2025, doi: 10.1038/s41598-025-86009-8.
- [24] C. Liu, "Long short-term memory (LSTM)-based news classification model," *PLoS One*, vol. 19, no. 5 May, pp. 1–23, 2024, doi: 10.1371/journal.pone.0301835.
- [25] F. Alkomah and X. Ma, "A Literature Review of Textual Hate Speech Detection Methods and Datasets," *Inf.*, vol. 13, no. 6, pp. 1–22, 2022, doi: 10.3390/info13060273.
- [26] K. Chatsiou, "Text Classification of Manifestos and COVID-19 Press Briefings using BERT and Convolutional Neural Networks," pp. 1–12, 2020, [Online]. Available: <http://arxiv.org/abs/2010.10267>
- [27] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.
- [28] M. K. Shaik Vadla, M. A. Suresh, and V. K. Viswanathan, "Enhancing Product Design through AI-Driven Sentiment Analysis of Amazon Reviews Using BERT," *Algorithms*, vol. 17, no. 2, p. 59, 2024, doi: 10.3390/a17020059.
- [29] Anugerah Simanjuntak *et al.*, "Research and Analysis of IndoBERT Hyperparameter Tuning in Fake News Detection," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 13, no. 1, pp. 60–67, 2024, doi: 10.22146/jnteti.v13i1.8532.
- [30] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, *A survey on imbalanced learning: latest research, applications and future directions*, vol. 57, no. 6. 2024. doi: 10.1007/s10462-024-10759-6.
- [31] M. N. Razali, N. Arbaiy, P. C. Lin, and S. Ismail, "Optimizing Multiclass Classification Using Convolutional Neural Networks with Class Weights and Early Stopping for

- Imbalanced Datasets,” *Electron.*, vol. 14, no. 4, pp. 1–14, 2025, doi: 10.3390/electronics14040705.
- [32] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal Loss for Dense Object Detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020, doi: 10.1109/TPAMI.2018.2858826.
- [33] Y. N. Kunang, S. Nurmaini, D. Stiawan, and B. Y. Suprpto, “Deep learning with focal loss approach for attacks classification,” *Telkomnika (Telecommunication Comput. Electron. Control.)*, vol. 19, no. 4, pp. 1407–1418, 2021, doi: 10.12928/TELKOMNIKA.v19i4.18772.
- [34] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, “Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem,” *Technologies*, vol. 9, no. 4, 2021, doi: 10.3390/technologies9040081.
- [35] S. Farhadpour, T. A. Warner, and A. E. Maxwell, “Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices,” *Remote Sens.*, vol. 16, no. 3, pp. 1–22, 2024, doi: 10.3390/rs16030533.
- [36] K. Nand Kumar, “Mikailalsys Journal of,” *Mikailalsys J. Math. Stat.*, vol. 2, no. 3, pp. 102–113, 2024.
- [37] A. S. Rizkia, Wufon, and F. F. Roji, “Sentiment Analysis of Coretax : A Comparison of Manual , Transformers- Based , and Lexicon-Based Data Labeling on IndoBERT Performance Analisis Sentimen Coretax : Perbandingan Pelabelan Data Manual ,” *Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. July, pp. 1037–1048, 2025.