KOMPUTASI: JURNAL ILMIAH ILMU KOMPUTER DAN MATEMATIKA

VOL. 23 (1) (2026), 50-58 p-ISSN: 693-7554, e-ISSN:2654-3990

Sentiment Analysis of Electric Vehicles on Social Media Using Bidirectional Encoder Representations from Transformers (BERT) and Long Short-Term Memory (LSTM)

Muhammad Fadhillah Harahap [1], Prihastuti Harsani [2], Yusma Yanti [3]

[1, 2, 3] Department of Computer Science, Faculty of Mathematics and Natural Science, Pakuan University, Bogor, West Java, 16143, Indonesia

---

Abstract

Electric vehicles (EVs) are widely recognized as an environmentally sustainable alternative capable of reducing greenhouse gas emissions; however, their adoption in Indonesia remains limited. Data from the Indonesian Ministry of Transportation, as recorded in the Type Approval Registration System (SRUT), indicate that approximately 195,084 Battery Electric Vehicles (BEVs) were registered nationwide by early 2024. Furthermore, the sentiment analysis results indicate that public opinions toward electric vehicles on social media are predominantly negative and very negative, reflecting concerns related to cost, performance, and maintenance. This finding highlights the gap between government promotion efforts and public perception of electric vehicle adoption in Indonesia. This study investigates public sentiment toward electric vehicles using social media data from X, Instagram, and TikTok, while also comparing the effectiveness of two text classification approaches: Bidirectional Encoder Representations from Transformers (BERT) and Long Short-Term Memory (LSTM). A total of 5,172 Indonesian-language comments were collected through crawling and scraping techniques using electric-vehicle-related keywords over the period January 2021 to January 2025. The comments were categorized into five sentiment classes: very positive, positive, neutral, negative, and very negative. The analytical process followed the Knowledge Discovery in Databases (KDD) framework, including data preprocessing, transformation, classification, and evaluation using a confusion matrix. This study employs a BERT-based model specifically tailored for the Indonesian language, known as IndoBERT. The results indicate that IndoBERT substantially outperformed LSTM, achieving an accuracy of 91% compared to 36% for LSTM. Sentiment analysis reveals a dominance of negative and very negative opinions, primarily reflecting public concerns regarding cost, performance, and maintenance of electric vehicles. These findings offer important insights for policymakers and the automotive industry in designing targeted promotion strategies, improving public awareness, and strengthening supporting infrastructure. Future research is encouraged to explore data augmentation techniques to improve model performance, particularly for deep learning models such as LSTM, in order to better support evidence-based electric vehicle adoption policies.

Keywords: BERT; LSTM; sentiment analysis; electric vehicles; social media

---

1. Introduction

Electric vehicles have emerged as a key component of the global response to environmental degradation associated with fossil fuel–based transportation systems. Compared to conventional vehicles, electric vehicles are widely perceived as offering superior energy efficiency, reduced environmental impact, and lower long-term operating costs [1]. In Indonesia, the promotion of electric vehicle adoption has been incorporated into national policies aimed at supporting sustainable development and mitigating carbon emissions. Official statistics from the Ministry of Transportation indicate that by November 2024, a total of 195,084 electric vehicles including electric motorcycles and passenger cars had been registered nationwide [2].

Despite these developments, the diffusion of electric vehicles has not progressed without obstacles. Social factors such as public perception, limited technological awareness, and inadequate supporting infrastructure continue to hinder broader acceptance. In this context, understanding how public attitudes toward electric vehicles are shaped and communicated has become increasingly important. Social media platforms, including X, Instagram, and TikTok, play a central role in facilitating public discourse. Owing to their extensive user bases, real-time interaction, and continuously evolving content, these platforms constitute rich and dynamic sources of public opinion data, making them highly suitable for sentiment analysis studies [3].

Sentiment analysis is a computational technique used to assess subjective information in textual data in order to identify users' attitudes or emotional orientations toward a particular subject, typically categorized as

*Corresponding author. *E-mail address:* muhammadfadhillahharahap@gmail.com

Received: 7 January 2026, Accepted: 30 Januari 2026 and available online 31 January 2026

DOI: https://doi.org/10.33751/komputasi.v19i2.5260

positive, neutral, or negative [4]. With the rapid advancement of artificial intelligence and natural language processing, sentiment analysis methodologies have evolved from traditional machine learning approaches such as Naïve Bayes and Support Vector Machines (SVM) [5]to more sophisticated deep learning models, including Long Short-Term Memory (LSTM) networks [6] and transformer-based architectures such as Bidirectional Encoder Representations from Transformers (BERT) [7].

Previous studies have applied these techniques in various sentiment analysis contexts. Kusuma et al. [5] utilized an SVM classifier to analyze public sentiment toward electric vehicles and achieved an accuracy of 68.7%. Ananda et al. [6] combined SVM with Term Frequency–Inverse Document Frequency (TF–IDF) features to classify electric vehicle–related tweets, reporting an accuracy of up to 84%. Verawati and Jaelani [8] employed the Naïve Bayes algorithm to examine Twitter users' sentiment regarding electric buses, with findings indicating a predominance of positive sentiment. Furthermore, Khadapi and Pakpahan [9] conducted a comparative study of LSTM and BERT models in sentiment analysis of discussions surrounding the 2024 Indonesian general election, concluding that LSTM demonstrated greater stability, while BERT exhibited superior performance in capturing complex contextual information.

Building upon these studies, the present research conducts sentiment analysis on public opinions regarding electric vehicles in Indonesia using deep learning approaches, specifically IndoBERT and LSTM models. The dataset was collected from X, Instagram, and TikTok, comprising a total of 5,172 Indonesian-language comments. These comments were categorized into five sentiment classes: very positive, positive, neutral, negative, and very negative. The primary novelty of this study lies in the application of IndoBERT, a BERT-based language model pre-trained on large-scale Indonesian corpora, which is expected to address the limitations of general-purpose language models in capturing local linguistic and contextual nuances. Additionally, this research provides a systematic performance comparison between IndoBERT and LSTM to evaluate their respective effectiveness in sentiment classification tasks involving Indonesian-language social media data. The objective of this study is to offer deeper insights into public perceptions of electric vehicles in Indonesia while identifying the most accurate sentiment classification approach for Indonesian textual data. The findings are expected to contribute to evidence-based policymaking, public communication strategies, and the future development of environmentally sustainable transportation technologies

2. Methodology

The methodological framework adopted in this study is based on the Knowledge Discovery in Databases (KDD) approach, which consists of several sequential stages, namely data selection, data preprocessing, data transformation, sentiment classification, and performance evaluation. The classification stage employs two deep learning models, namely IndoBERT and Long Short-Term Memory (LSTM), to analyze sentiment patterns in textual data. The dataset comprises user-generated comments related to electric vehicles collected from social media platforms X, Instagram, and TikTok. A total of 5,172 Indonesian-language comments were gathered over the period from January 2021 to January 2025 and subsequently utilized for sentiment analysis.

2.1 Data Prepocessing and Data Transformation

The preprocessing procedures were tailored to the characteristics of public statements regarding electric vehicles collected from social media platforms X, Instagram, and TikTok. These steps were implemented to improve data consistency, reduce noise, and enhance the quality of textual inputs prior to sentiment analysis. The preprocessing pipeline consists of the following stages:

1. Case Folding,

   This stage involves converting all characters in the text into lowercase to ensure uniform text formatting and to prevent duplication caused by differences in letter capitalization.
2. Stopword Removal

   In this step, commonly occurring words that do not convey significant semantic information are removed. Additionally, irrelevant or unused hashtags generated during the data crawling process are eliminated to minimize noise and improve data relevance.
3. Tokenization

   Tokenization is the process of segmenting sentences into individual tokens, which serve as the fundamental units for text analysis. During this stage, punctuation marks and numerical values are typically discarded.
4. Stemming

   Stemming reduces words to their root or base forms, allowing different word variations with similar meanings to be treated as a single term, thereby improving consistency in textual representation.

After completing these four preprocessing stages case folding, stopword removal, tokenization, and stemming the resulting cleaned dataset is subsequently utilized for manual sentiment labeling, which serves as the reference data for training and evaluating the classification models. Following the preprocessing stage, the textual data were transformed into numerical representations suitable for model input. For the LSTM model, the processed text was encoded using a one-hot encoding scheme. In contrast, the IndoBERT model employed BERT-based tokenization, which incorporates special tokens such as [CLS], [SEP], and [PAD] to capture sentence-level information, segment boundaries, and sequence padding, respectively. Word embeddings and model fine-tuning were performed using the Hugging Face Transformers library in conjunction with the PyTorch framework. This approach enables the optimization of pre-trained IndoBERT parameters for the specific sentiment classification task while preserving contextual representations learned from large-scale Indonesian language corpora.

2.2 Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) is a deep learning model introduced by Google and represents a major advancement of the Transformer architecture. Unlike traditional sequential models, BERT processes text bidirectionally, allowing each word in a sentence to be interpreted based on its surrounding context from both preceding and succeeding tokens. This capability enables BERT to capture rich semantic and contextual information, making it highly effective for natural language understanding tasks. Although BERT demonstrates optimal performance on English-language datasets, direct application to Indonesian text is less effective due to linguistic differences. To address this limitation, this study employs IndoBERT, a variant of the BERT Base model that has been pre-trained on large-scale and diverse Indonesian language corpora, enabling improved contextual understanding of Indonesian text [10]. The overall IndoBERT-based classification architecture is illustrated in Figure 1.

As shown in Figure 2, the input text first undergoes a preprocessing stage to normalize and clean the raw social media data. The processed text is then tokenized using the BERT encoder within the IndoBERT model. During this stage, the input sequence is converted into subword tokens based on the IndoBERT vocabulary and encoded into contextual embeddings. A dropout layer is subsequently applied to reduce overfitting by randomly deactivating a portion of neurons during training. Finally, the output representations generated by the encoder are passed to a dense classifier layer, which determines the sentiment class of each input text. This study utilizes the Hugging Face Transformers library, which provides a comprehensive collection of pre-trained transformer models for text classification across more than 100 languages, with full support for PyTorch and TensorFlow frameworks. Tokenization is performed using the BertTokenizer, which maps input text to the appropriate vocabulary indices, applies special tokens required by the BERT architecture, and prepares the input in a format compatible with the IndoBERT encoder

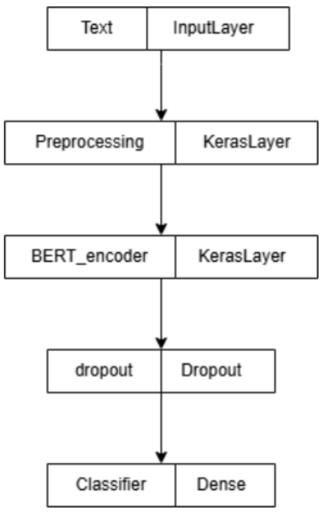

Figure 1. BERT Flowchart

2.3 Long short term memory (LSTM)

Long Short-Term Memory (LSTM) is an extension of the Recurrent Neural Network (RNN) architecture designed to address the limitations of standard RNNs in modeling long-range dependencies. By incorporating a memory cell mechanism, LSTM is capable of retaining information over extended time intervals, making it particularly suitable for capturing temporal patterns and sequential relationships in textual data. In sentiment analysis tasks, LSTM effectively models word order and contextual dependencies to determine sentiment polarity within a sequence [11]. The overall workflow of the LSTM model employed in this study is illustrated in Figure 2. As depicted in Figure 2, each LSTM unit (represented by the blue blocks) consists of three primary gating mechanisms namely the forget gate, input gate, and output gate along with a cell state that serves as long-term memory. The operational steps of an LSTM unit can be described as follows:

1. Input and Previous Hidden State, At each time step, the current input vector and the hidden state from the previous time step are used jointly to compute the current state of the network
2. Forget Gate, the forget gate determines which information from the previous cell state should be retained or discarded, allowing the model to remove irrelevant or outdated information.

3. Input Gate and Candidate State
   a. The input gate regulates the amount of new information that will be written into the cell state.
   b. A candidate cell state is generated using a hyperbolic tangent (*tanh*) activation function, representing potential new content to be stored.
4. Cell State Update is updated by combining the retained information from the previous state with the newly selected candidate information.
5. Output Gate and Hidden State
   a. The output gate determines which parts of the updated cell state are exposed as output.
   b. The current hidden state (h_t) is computed based on the updated cell state and serves as input to the next time step or subsequent network layers.

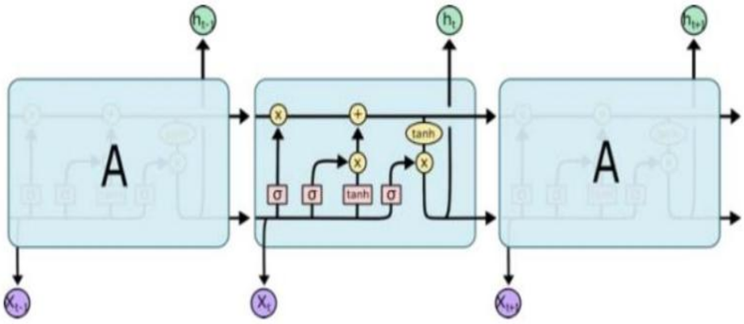

Figure 2. LSTM Flowchart

2.4 Confusion Matrix

A confusion matrix is a tabular representation used to summarize the performance of a classification model by displaying the number of correctly and incorrectly classified instances for each class [12]. In this study, model evaluation is conducted using a confusion matrix in conjunction with standard performance metrics, including accuracy, precision, recall, and F1-score, to provide a comprehensive assessment of classification effectiveness. The dataset is partitioned using an 80:20 split, where 80% of the data are allocated for model training and the remaining 20% are reserved for testing. The sentiment classification task is formulated as a five-class problem, consisting of *very positive*, *positive*, *neutral*, *negative*, and *very negative* sentiment categories.

3. Result and Discution

Data were collected from social media platforms X, Instagram, and TikTok using the Insta Data Scraper browser extension. This tool was employed to crawl and extract user-generated comments from publicly accessible posts related to electric vehicles. As an illustrative case, a TikTok post uploaded by the account *@Slowgan* containing the hashtag #kendaraanlistrik was selected, and the associated comments were retrieved to capture public responses toward electric vehicles. Overall, the data collection process yielded a total of 5,172 Indonesian-language comments obtained from X, Instagram, and TikTok. Prior to sentiment classification, the collected comments underwent a preprocessing stage to clean and standardize the textual data. Subsequently, the preprocessed data were classified into five sentiment categories, namely *very positive*, *positive*, *neutral*, *negative*, and *very negative*

Table 1. Distribution of Sentiment Categories

Sentiment Category	Number of Comments
Very Positive	895
Positive	515
Neutral	663
Negative	1,47
Very Negative	1,629
Total	**5,172**

Based on the sentiment distribution presented in Table 1, negative and very negative sentiments dominate public discussions on electric vehicles across X, Instagram, and TikTok. Specifically, more than half of the collected comments express unfavorable perceptions, indicating substantial public concern regarding electric vehicle adoption. These negative sentiments are primarily associated with economic factors such as high initial costs, as well as technical concerns related to vehicle performance and maintenance. This dominance of negative sentiment suggests that, despite increasing policy support and infrastructure development, public acceptance of electric vehicles in Indonesia remains limited. Therefore, sentiment analysis in this study not only serves as a comparative evaluation of classification models but also provides substantive insights into public attitudes toward electric vehicles.

The data transformation stage applies word embedding techniques to convert textual data into numerical vector representations, enabling the text to be processed by deep learning models. This transformation facilitates the utilization of textual information by both BERT and LSTM during the sentiment classification process. As illustrated in Figure 3, BERT embeddings are generated through a contextual embedding mechanism that captures bidirectional semantic relationships within the text. In contrast, Figure 4 depicts the embedding process for the LSTM model, where word representations are mapped into a sequential vector space to preserve temporal dependencies across tokens.

The tokenization and data encoding stages are essential for preparing textual input for the BERT model. Prior to classification, the input text must be segmented into numerical tokens, which are subsequently mapped to their corresponding indices in the tokenizer's vocabulary. Through this process, the original text is transformed into vectorized numerical representations suitable for deep learning models. In this encoding scheme, the value 3 represents the [CLS] token, 1603 corresponds to the [SEP] token, and 0 denotes the [PAD] token used for sequence padding. The tokenized and embedded representations obtained from earlier stages are then fed into the BERT model for further processing.

```
The BERT model has 201 different named parameters.
==== Embedding Layer ====
bert.embeddings.word_embeddings.weight                      (31923, 768)
bert.embeddings.position_embeddings.weight                    (512, 768)
bert.embeddings.token_type_embeddings.weight                    (2, 768)
bert.embeddings.LayerNorm.weight                                  (768,)
bert.embeddings.LayerNorm.bias                                    (768,)
==== First Transformers ====
bert.encoder.layer.0.attention.self.query.weight              (768, 768)
bert.encoder.layer.0.attention.self.query.bias                    (768,)
bert.encoder.layer.0.attention.self.key.weight                (768, 768)
bert.encoder.layer.0.attention.self.key.bias                      (768,)
bert.encoder.layer.0.attention.self.value.weight              (768, 768)
bert.encoder.layer.0.attention.self.value.bias                    (768,)
bert.encoder.layer.0.attention.output.dense.weight            (768, 768)
bert.encoder.layer.0.attention.output.dense.bias                  (768,)
bert.encoder.layer.0.attention.output.LayerNorm.weight            (768,)
bert.encoder.layer.0.attention.output.LayerNorm.bias              (768,)
bert.encoder.layer.0.intermediate.dense.weight               (3072, 768)
bert.encoder.layer.0.intermediate.dense.bias                     (3072,)
bert.encoder.layer.0.output.dense.weight                     (768, 3072)
bert.encoder.layer.0.output.dense.bias                            (768,)
bert.encoder.layer.0.output.LayerNorm.weight                      (768,)
bert.encoder.layer.0.output.LayerNorm.bias                        (768,)
==== Output Layer ====
bert.pooler.dense.weight                                      (768, 768)
bert.pooler.dense.bias                                            (768,)
classifier.weight                                               (5, 768)
classifier.bias                                                     (5,)
==== Total Parameters ====
Total Parameters: 110562053
```

Figure 3. BERT Embedding

```
Label yang ada dalam data pelatihan: [0 1 2 3 4]
Label train one-hot encoded:
 [[0. 0. 0. 0. 1.]
 [0. 0. 0. 1. 0.]
 [0. 1. 0. 0. 0.]
 ...
 [0. 0. 0. 0. 1.]
 [0. 0. 0. 1. 0.]
 [0. 0. 0. 0. 1.]]

Label test one-hot encoded:
 [[0. 1. 0. 0. 0.]
 [1. 0. 0. 0. 0.]
 [0. 0. 0. 1. 0.]
 ...
 [0. 0. 0. 1. 0.]
 [0. 0. 1. 0. 0.]
 [0. 0. 0. 0. 1.]]
```

Figure 4. LSTM Embedding

This stage constitutes the experimental training phase of the study, where model performance is evaluated under different hyperparameter configurations, particularly variations in the number of epochs and batch sizes, in order to obtain optimal results. Three data-splitting scenarios were examined, namely 70:30, 80:20, and 90:10 ratios for training and testing datasets.

Across all three scenarios, the BERT model exhibited a consistent decrease in training loss, indicating a stable and effective learning process. However, under the 80:20 data split, the model not only demonstrated a smooth and stable loss reduction but also achieved the lowest final training loss among the evaluated configurations. This finding suggests that the 80:20 split provides a favorable balance between sufficient training data and an adequate test set for performance evaluation. The stable training behavior further indicates that the model progressively captures meaningful patterns within the data, as illustrated in Figure 5.

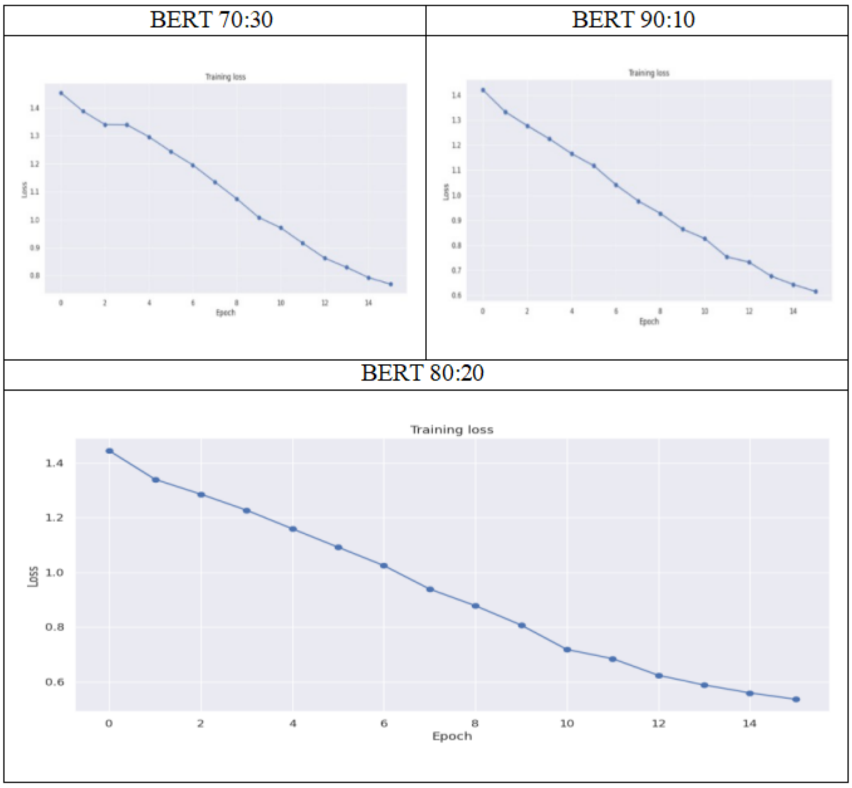

Figure 5. BERT Training Loss over Epochs

Furthermore, the classification performance of the BERT model using the 80:20 split was analyzed through a confusion matrix visualization. As shown in Figure 6, the confusion matrix presents the prediction results on the test dataset across five sentiment classes, providing a detailed view of class-wise classification performance.

Based on Figure 6, the confusion matrix evaluation yields four fundamental classification outcomes: True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). These values form the basis for assessing the performance of the classification model. In this study, model performance is evaluated using three key metrics, namely accuracy, precision, and recall, which are derived from the confusion matrix components. The results of the performance metric calculations are presented in Figure 7, providing a quantitative summary of the model's classification effectiveness.

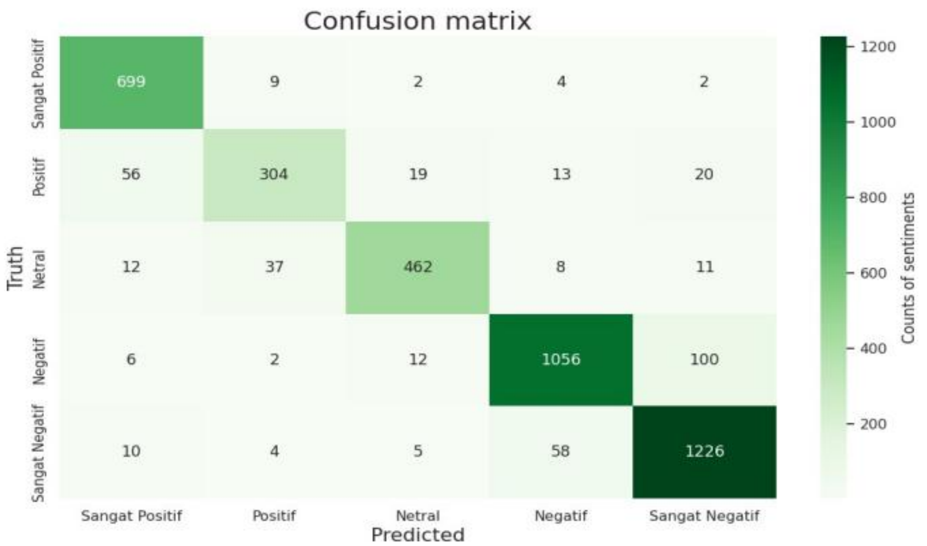

Figure 6. Confusion Matrix Heatmap of the BERT Model

	precision	recall	f1-score	support
0	0.89	0.98	0.93	716
1	0.85	0.74	0.79	412
2	0.92	0.87	0.90	530
3	0.93	0.90	0.91	1176
4	0.90	0.94	0.92	1303
accuracy			0.91	4137
macro avg	0.90	0.88	0.89	4137
weighted avg	0.91	0.91	0.90	4137

Figure 7. Confusion Matrix Results of the BERT Model

The embedded data obtained from the previous transformation stage were subsequently fed into the LSTM model for further processing. This stage represents the experimental training phase of the study, in which various hyperparameter configurations—specifically the number of epochs and batch sizes were evaluated to obtain optimal model performance. The training loss curves of the LSTM model are illustrated in Figure 6, which presents the loss trajectories under three different data-splitting scenarios: 70:30, 80:20, and 90:10 for training and testing datasets. Under the 80:20 split, both training and validation loss exhibit a significant decrease during the initial epochs, followed by a stable convergence with minor fluctuations. This behavior indicates that the model learns effectively while maintaining good generalization performance on the validation data.

In contrast, the 90:10 split demonstrates signs of overfitting, as the validation loss continues to increase despite a decreasing training loss. Meanwhile, the 70:30 split shows relatively good performance; however, its stability is less consistent when compared to the 80:20 configuration, as further illustrated in Figure 8. These observations suggest that the 80:20 data split provides the most balanced and stable learning condition for the LSTM model. Furthermore, the classification performance of the LSTM model under the 80:20 split was evaluated using a confusion matrix. The visualization of the confusion matrix, presented in Figure 9, depicts the prediction results on the test dataset across five sentiment classes, offering detailed insights into class-wise classification performance.

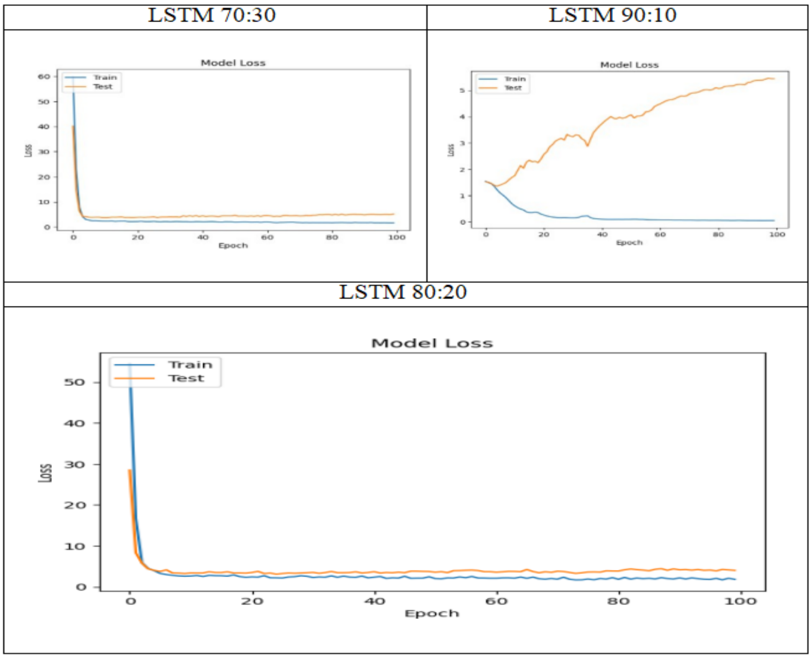

Figure 8. Training Loss Curve of the LSTM Model Across Epochs

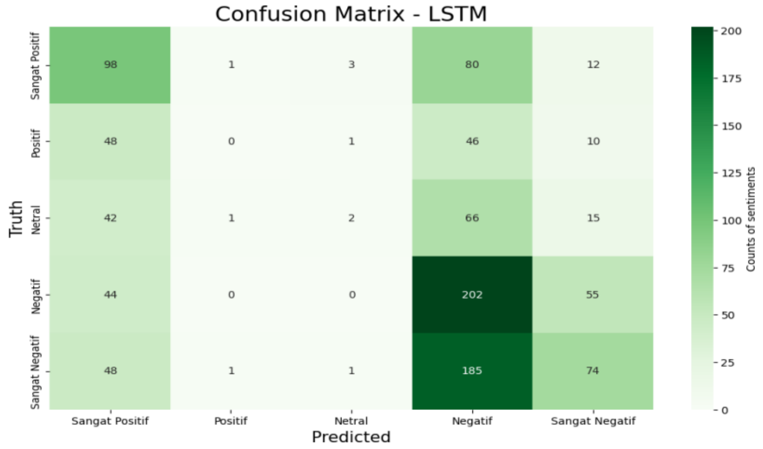

Figure 9. Confusion Matrix Heatmap of the LSTM Model

Based on Figure 9, the confusion matrix evaluation yields four fundamental classification outcomes, namely True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). These components serve as the foundation for assessing the performance of the classification model. In this study, model performance is evaluated using three key metrics: accuracy, precision, and recall, all of which are derived from the confusion matrix values. The results of these performance metric calculations are presented in Figure 10, providing a quantitative summary of the model's classification effectiveness.

```
Classification Report:
              precision    recall  f1-score   support

           0       0.35      0.51      0.41       194
           1       0.00      0.00      0.00       105
           2       0.29      0.02      0.03       126
           3       0.35      0.67      0.46       301
           4       0.45      0.24      0.31       309

    accuracy                           0.36      1035
   macro avg       0.29      0.29      0.24      1035
weighted avg       0.33      0.36      0.31      1035
```

Figure 10. Confusion Matrix Results of the LSTM Model

The experimental results obtained using the IndoBERT and LSTM models are summarized in Table 2. Overall, IndoBERT consistently outperforms LSTM across all evaluated train–test split scenarios, namely 70:30, 80:20, and 90:10. The most notable performance is observed under the 80:20 data split, where IndoBERT achieves the highest classification accuracy of 91%, whereas the LSTM model attains an accuracy of only 36%. This performance disparity can be attributed to fundamental differences in model architecture and learning mechanisms. IndoBERT is based on a transformer architecture and benefits from pre-training on large-scale Indonesian language corpora, enabling the model to acquire rich linguistic and contextual knowledge prior to the classification task. As a result, IndoBERT is capable of delivering optimal performance even with a relatively small number of training epochs (16 epochs in this study). In contrast, LSTM processes input sequences sequentially and learns representations from scratch, relying heavily on input encoding schemes such as one-hot encoding. This limitation makes it more challenging for LSTM to capture long-range contextual dependencies within the text.

Moreover, increasing the number of training epochs for LSTM up to 100 epochs in this study does not necessarily lead to improved classification accuracy, indicating inefficiencies in learning complex contextual patterns. These findings demonstrate that transformer-based models such as IndoBERT are substantially more effective than LSTM in capturing contextual semantics for text classification tasks, particularly in sentiment analysis of Indonesian-language social media data.

Table 2. Comparison of Experimental Results

Splitting	**IndoBERT**			**LSTM**		
	70:30:00	80:20:00	90:10:00	70:30:00	80:20:00	90:10:00
Epoch	16	16	16	100	100	100
Num_Batch	259	259	259	229	229	229
Optimize	*Adam*	*Adam*	*Adam*	*Adam*	*Adam*	*Adam*
Accuracy	0.79	0.91	0.84	0.32	0.36	0.34

4. Conclusion

This study conducted sentiment analysis on public opinions regarding electric vehicles expressed on social media platforms, namely X, Instagram, and TikTok, using Indonesian-language data. Based on the analysis of 5,172 comments, the findings reveal that public sentiment toward electric vehicles in Indonesia is predominantly negative and very negative, indicating widespread concerns related to purchase cost, vehicle performance, and maintenance issues. This result suggests that public acceptance of electric vehicles remains a significant challenge despite ongoing government initiatives and infrastructure development.

In addition to identifying public sentiment patterns, this study also evaluated the performance of two deep learning models, IndoBERT and LSTM, for sentiment classification. The experimental results demonstrate that IndoBERT consistently outperforms LSTM across all evaluated scenarios. Under the optimal 80:20 train–test split, IndoBERT achieves an accuracy of 91%, whereas LSTM attains only 36%. The superior performance of IndoBERT can be attributed to its transformer-based architecture and pre-training on large-scale Indonesian corpora, which enable more effective contextual understanding of Indonesian-language text.

Overall, these findings confirm that IndoBERT is a robust and effective model for sentiment analysis of Indonesian social media data, while also providing meaningful insights into public perceptions of electric vehicles. The results of this study can support policymakers and industry stakeholders in formulating more targeted communication strategies and addressing public concerns to encourage broader adoption of electric vehicles in Indonesia.

References

[1] A. Alfarizi and R. Fitriani, "Energi Terbarukan dan Kendaraan Listrik," *Jurnal Energi*, vol 5, no. 2, pp. 101–109, 2023.

[2] Outlook 2025: "*Menanti Perkembangan Infrastruktur Kendaraan Listrik di Indonesia*," [Online]. Available: databoks.katadata.co.id

[3] Pradana, "Analisis Sentimen Kendaraan Listrik," *Jurnal Informatika Sosial*, vol. 2, no. 1, 2024.

[4] Alita and Isnain, "Sentiment Analysis in Bahasa Indonesia Text," *J. Komputer & AI*, vol. 4, no. 1, 2020.

[5] G.H. Kusuma et al., "Pendekatan Machine Learning: Analisis Sentimen Terhadap Kendaraan Listrik," *Jurnal Teknologi dan Sains*, vol. 6, 2023.

[6] N. Ananda et al., "Klasifikasi Sentimen Tweet Menggunakan SVM," *J. Data Science Indonesia*, vol. 5, no. 3, 2023.

[7] Kurniawan et al., "IndoBERT for Indonesian NLP," *IEEE Conf. on Computational Linguistics*, 2022.

[8] I. Verawati and S.N. Jaelani, "Sentimen Bus Listrik Menggunakan Naïve Bayes," *Jurnal Sistem Informasi*, vol. 8, no. 2, 2024.

[9] M. Khadapi and V. Pakpahan, "Analisis Sentimen LSTM vs BERT untuk Pemilu," *J. NLP dan Politik Digital*, vol. 3, no. 2, 2024. Karimah, N., & Baita, A. (2024).

[10] Multi-Aspect Sentiment Analysis Pada Review Film Menggunakan Metode Bidirectional Encoder Representations From Transformers (BERT) Multi-Aspect Sentiment Analysisof Film Review Using Bidirectional Encoder Representations from Transformers (BERT). *Jurnal Sistem Komputer*, *13*(1), 2020. https://doi.org/10.34010/komputika.v13i1.11098

[11] Isnain, A. R., Sulistiani, H., Hurohman, B. M., Nurkholis, A., & Styawati, S.(2022).AnalisisPerbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen.*Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *8*(2), 299. https://doi.org/10.26418/jp.v8i2.54704

[12] Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan ConfusionMatrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, *5*(2), 697–7.