JURNAL STATISTIKA DAN APLIKASINYA

e-ISSN 2620 – 8369
VOLUME 9 (ISSUE 2) DECEMBER 2025: 57-70
https://doi.org/10.21009/JSA.09206

Performance Evaluation of Word Embedding Techniques in Twitter Sentiment Analysis using LSTM

Faroh Ladayya*, Widyanti Rahayu, Siti Rohmah Rohimah, Ferdiansyah Rizki Saputra, Thoriq Akbar Maulana, Najwa Nur Madinah

Department of Statistics, Faculty of Mathematics and Natural Sciences, Universitas Negeri Jakarta Jl. Rawamangun Muka No.11, Rawamangun, East Jakarta 13220, Indonesia

Corresponding author's e-mail: ***faroh.ladayya@unj.ac.id***

Article History:
Received: 3, November 2025
Revised: 15, November 2025
Revised
Accepted: 29, December 2025
Published: 31, December 2025
Available online.

Keywords:
Long-Short Term Memory; Sentiment Analysis; Word Embedding.

ABSTRACT

Opinions expressed on social media can be used as feedback on a product, both goods and services. The sentiment analysis was utilized for analyzing opinions given by the public via social media. The sentiment contained in an opinion can be positive, negative, or neutral. This study aims to compare the performance of three word embedding techniques—Word2Vec, GloVe, and FastText—when combined with a Long Short-Term Memory (LSTM) model for sentiment classification of Indonesian Twitter data. LSTM was selected due to its ability to model sequential text data and capture long-term contextual dependencies that are often present in natural language. To enable sentiment classification using LSTM, textual data from social media were transformed into numerical vectors. Thus, the word embedding technique is used to convert text into a vector. The vector that had been obtained will be used as input for LSTM. All embeddings were evaluated under the same preprocessing pipeline and LSTM architecture to ensure a fair comparison. Model performance was assessed using accuracy, precision, recall, F1-score, and ROC/AUC metrics. The results indicate that the LSTM model effectively captures sentiment patterns in Indonesian tweets, with Word2Vec achieving the best overall performance, followed by GloVe and FastText. These findings suggest that domain-adapted word embeddings remain highly effective for sentiment analysis in Indonesian social media contexts.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-ShareAlike 4.0 International License.

How to cite this article:
F. Ladayya, W. Rahayu, S. R. Rohimah, F. R. Saputra, T. A. Maulana, N. N. Madinah, "Performance Evaluation of Word Embedding Techniques in Twitter Sentiment Analysis using LSTM", Jurnal Statistika dan Aplikasinya, vol. 9, iss. 2, pp. 57 – 70, December 2025

Copyright © 2025 Author(s)
Journal e-mail: *jsa@unj.ac.id*
Research Article · Open Access

1. INTRODUCTION

In the era of digitalization, data processing is no longer limited to structured numerical data but increasingly involves unstructured data such as text, images, and videos. Unstructured data dominate digital information, most of which is textual [1]. The rapid growth of social media has transformed how people communicate and express opinions on various issues. In Indonesia, with an internet penetration rate of 79% and 65% of the population actively using social media, textual data have become a valuable resource for large-scale analysis and data-driven decision-making [2].
Natural Language Processing (NLP) enables computers to analyze and interpret human language computationally [3]. One of the most widely applied NLP tasks is sentiment analysis, which aims to identify opinions, emotions, and attitudes expressed in text [4]. Sentiment analysis plays an important role in various domains, including public policy evaluation and consumer behavior analysis [5]. In supervised sentiment analysis, machine learning models learn from labeled data to classify text into sentiment categories such as positive, negative, or neutral. Conventional approaches such as logistic regression, Support Vector Machines (SVM), and Naïve Bayes have been widely applied and shown satisfactory performance in Indonesian sentiment analysis studies [6]–[8]. However, model performance is highly dependent on the quality of text preprocessing [9].

Text preprocessing is therefore a critical step in sentiment analysis, as social media text often contains noise and informal language. Common preprocessing steps include tokenization, stopword removal, stemming or lemmatization, and word normalization [6], [10]. After preprocessing, text must be converted into numerical form so that it can be processed by machine learning algorithms. Traditional feature extraction methods such as Bag of Words (BoW) and TF-IDF rely on word frequency and fail to capture semantic relationships between words [11]. In contrast, word embedding techniques map words into continuous vector spaces, allowing semantically similar words to be represented closer together [12], [13], and forming the basis of deep learning-based sentiment analysis.

Among deep learning models, Long Short-Term Memory (LSTM) networks have been widely used for sentiment analysis due to their ability to model sequential text data and capture long-term dependencies [14]. Several studies have demonstrated that LSTM effectively captures contextual relationships in text and often outperforms conventional machine learning models such as SVM and Naïve Bayes [6], [15]. However, the performance of LSTM-based models is strongly influenced by the choice of word embedding techniques used as input representations. The comparison of Word2Vec, GloVe, and FastText is therefore important because each embedding method captures semantic information differently and exhibits distinct strengths and limitations, particularly when applied to informal social media text. While Word2Vec and GloVe rely on word-level representations [12], [13], FastText incorporates subword information that can better handle misspellings and out-of-vocabulary words [16], which are common in Indonesian Twitter data.

Most previous sentiment analysis studies focus on a single word embedding method or employ different datasets and experimental configurations, making objective performance comparison across methods difficult [16], [17]. In the context of Indonesian sentiment analysis, prior studies—including those using the same dataset—are still largely dominated by conventional machine learning approaches and rarely examine deep learning models combined with multiple embedding techniques in a controlled setting [7], [8]. To address this gap, this study deliberately employs the same dataset as previous work [7] to conduct a fair and controlled comparison of Word2Vec, GloVe, and FastText within an identical LSTM framework. By isolating the effect of word embedding methods, this study aims to identify the most suitable representation technique for sentiment classification of Indonesian social media text.

2. METHODS

Material and Data

This research utilizes a dataset from a previous study by [7], consisting of tweets obtained from the social media platform *Twitter* containing the keyword “JakLingko.” The data were collected between

June 1, 2022, and August 31, 2022, using a Python-based scraping tool called "snscrape." A total of 2,340 tweets were retrieved and used as the primary dataset for the analysis conducted in this study.

Research Method

Word Embedding

Word embedding is a fundamental technique in Natural Language Processing (NLP) that represents words as dense, low-dimensional numerical vectors. This representation allows words with similar meanings or contexts to appear closer to one another in the vector space, enabling models to capture semantic and syntactic relationships between them [12], [13]. By transforming textual data into numerical form, word embeddings facilitate the integration of language information into deep learning models and significantly improve performance in various NLP tasks—such as sentiment analysis, text classification, and machine translation—due to their ability to effectively encode both semantic and contextual information [15], [18].

Word2Vec

Word2Vec, introduced by [12], is a neural-based model designed to learn distributed vector representations of words from large text corpora. The fundamental assumption underlying this approach is that words appearing in similar contexts tend to share similar meanings. Word2Vec employs a shallow neural network to learn word embeddings through one of two architectures: Continuous Bag of Words (CBOW) and Skip-gram. In the CBOW model, the objective is to predict a target word w_t based on its surrounding context words $w_{t-c}, \dots, w_{t+c}$, where c represents the context window size. The model maximizes the conditional probability:

$$P(w_t \mid w_{t-c}, \dots, w_{t+c}) = \frac{\exp(v'_{w_t} \cdot h)}{\sum_{w=1}^{V} \exp(v'_w \cdot h)} \tag{1}$$

where:

$$h = \frac{1}{2c} \sum_{-c \le j \le c, j \neq 0} v_{w_{t+j}}$$

Here, v_w and v'_w denote the input and output vector representations of word w, respectively, and V is the vocabulary size. The training objective is to maximize the log-likelihood over the entire corpus:

$$\max_{\theta} \sum_{t=1}^{T} \log P(w_t \mid w_{t-c}, \dots, w_{t+c}) \tag{2}$$

Conversely, the Skip-gram model reverses this process by using the current word to predict its surrounding context words. Its objective is to maximize the probability:

$$\max \sum_{t=1}^{T} \sum_{-c \le j \le c, j \neq 0} \log P\left(w_{t+j} \mid w_t \right) \tag{3}$$

with

$$P(w_O \mid w_I) = \frac{\exp(v'_{w_O} \cdot v_{w_I})}{\sum_{w=1}^{V} \exp(v'_w \cdot v_{w_I})}$$

Since calculating the denominator for every word in the vocabulary is computationally expensive, both architectures typically employ negative sampling or hierarchical softmax to approximate the probability distribution efficiently.

Global Vectors for Word Representation (GloVe)

GloVe, introduced by [19], is a statistical word embedding model that learns vector representations of words by leveraging global word co-occurrence information from a large corpus. Unlike Word2Vec, which relies on local context windows to predict word relationships, GloVe constructs embeddings based on the global co-occurrence probability of words across the entire text corpus.

The main idea behind GloVe is that the ratios of co-occurrence probabilities between words can reveal meaningful semantic relationships. If two words i and j frequently appear together in similar contexts, their corresponding word vectors should be closely related in the embedding space.

Mathematically, GloVe minimizes the following weighted least squares objective function:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij})^2 \quad (4)$$

where X_{ij} is the number of times word $j$appears in the context of word i, $w_i$and $\tilde{w}_j$ are the word and context vectors, respectively, $b_i$and $\tilde{b}_j$ are the bias terms, and $f(X_{ij})$ is the weighting function.

Fast Text

FastText, introduced by [16], is an extension of the Word2Vec model designed to enhance the representation of words by incorporating subword information. FastText represents words as the sum of the vector representations of their constituent character n-grams. This approach allows the model to capture morphological and subword-level information, making it particularly effective for languages with rich morphology or for handling rare and out-of-vocabulary (OOV) words. Unlike Word2Vec, which treats each word as an independent token, FastText enables the generation of meaningful representations even for unseen words by utilizing shared subword structures. This makes it highly suitable for sentiment analysis tasks involving informal or morphologically diverse languages such as Indonesian.

Formally, a word w is represented as a collection of character n-grams G_w, for example, the word "transport" with n = 3 (trigram) would be represented as:

$$G_{transport} = \{< tr, tra, ran, ans, nsp, spo, por, ort, rt >\}$$

Each n-gram $g \in G_w$has its own vector representation z_g.
The word vector v_w is then obtained by summing the vectors of all its n-grams:

$$v_w = \sum_{g \in G_w} z_g \quad (5)$$

During training, FastText typically uses the Skip-gram architecture (similar to Word2Vec), where the objective is to maximize the probability of predicting a context word $w_o$given a target word w_i:

$$P(w_o \mid w_i) = \frac{\exp(v'_{w_o} \cdot v_{w_i})}{\sum_{w=1}^{V} \exp(v'_w \cdot v_{w_i})} \quad (6)$$

where $v_{w_i}$and $v'_{w_o}$are the input and output vector representations respectively.

Long-Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture proposed by [14] to overcome the *vanishing gradient* problem commonly encountered in conventional RNNs. LSTM introduces a *memory cell* equipped with three primary gates — the *input gate* (i_t), *forget gate* (f_t), and *output gate* (o_t) — which regulates the flow of information by determining which data should be stored, updated, or discarded.

The operations of an LSTM unit can be mathematically represented as follows:

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t * \tanh(C_t) \end{aligned} \tag{7}$$

where x_t denotes the input at time step t, h_t is the hidden state, C_t is the cell state, and σ represents the sigmoid activation function. This formulation enables the model to selectively preserve long-term dependencies and capture contextual relationships within sequences [20],[21]

3. RESULTS

The dataset used in this study consists of tweets written in Bahasa Indonesia that were manually labeled into two sentiment categories: positive and negative. Neutral or ambiguous tweets were excluded to maintain a clear binary classification task. The final dataset comprised 2,340 tweets, with 1,283 (54.83%) positive and 1,057 (45.17%) negative samples, as illustrated in Figure 1. The proportion between the two sentiment classes can be considered relatively balanced, ensuring that the learning process of the classification model is not biased toward a particular sentiment category.

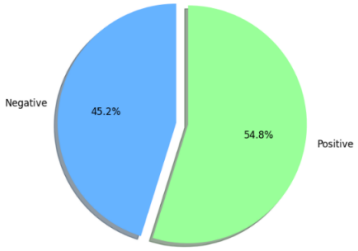

Figure 1. Distribution of Sentiment Labels

Before model training, this study utilized a dataset that had already undergone a comprehensive text preprocessing pipeline developed by Ladayya (2022) specifically for Indonesian social media text. The preprocessing process included converting all characters to lowercase, removing URLs, mentions, hashtags, punctuation, and stop words, as well as normalizing informal words and abbreviations into standard Indonesian vocabulary. This prior preprocessing ensured that the dataset used in this research was already clean, standardized, and semantically consistent, allowing the present study to focus primarily on the modeling and comparative analysis of word embedding techniques.

Both stemming and filtering preprocessing variants were initially tested during the experiments. However, the performance difference between the two methods was found to be minimal. Therefore, this study employed the filtering approach for the final modeling, as it retains the original lexical form of words, leading to better compatibility with pretrained Indonesian embeddings such as Word2Vec, GloVe, and FastText. Several studies have shown that extensive text normalization, including stemming, does not always lead to improved performance in sentiment analysis, particularly for Indonesian social media text. In some cases, stemming may remove important morphological information that contributes to sentiment polarity, while modern word embedding models are able to capture semantic patterns effectively without aggressive stemming [16], [23]. Table 1 presents several examples of tweets after preprocessing using the filtering approach, showing how raw informal texts were transformed into clean, tokenized sequences suitable for embedding-based modelling.

Table 1. Example Original Tweet and Filtered Tweet

Original Tweet	Filtered Tweet
Wah, jaklingko gratis malah klo pake kartunya. Skrg ada tarif integritas malah murah banget klo liat review yg udh pake, bisa ampe 50% kepotong. https://t.co/qxLw1oMkpm	['jaklingko', 'gratis', 'pakai', 'kartu', 'tarif', 'integritas', 'murah', 'banget', 'lihat', 'review', 'pakai', 'potong']
Ada yg udh bisa akses apk jaklingko? 🥹	['akses', 'apk', 'jaklingko', 'wajah', 'pusing', 'mabuk']
@imandani w brangkat ngantor bisa 2000 doang berkat jaklingko. 🫵🏻	['brangkat', 'ngantor', 'doang', 'berkat', 'jaklingko', 'tangan', 'angkat', 'rapat', 'warna', 'kulit', 'cerah']
@AnchaAncha12345 Daftar aja d aplikasi jaklingko \nFree kok	['daftar', 'aplikasi', 'jaklingko', 'fre']

In this study, three pretrained word embedding models were employed — Word2Vec, GloVe, and FastText — each representing different strategies of learning distributed word representations. The purpose of integrating multiple embedding methods was to evaluate how distinct semantic learning mechanisms affect sentiment classification in Bahasa Indonesia, particularly within informal text domains such as Twitter.

The Word2Vec model used in this study was initialized with pretrained vectors trained on the Indonesian Wikipedia corpus (commonly known as *idwiki embeddings*) [22]. These pretrained embeddings were then fine-tuned using the collected tweet dataset to allow the model to adapt to domain-specific vocabulary and stylistic nuances typical of social media language. This hybrid approach preserved the general semantic relationships learned from Wikipedia while enhancing the model's ability to capture sentiment-bearing expressions prevalent in Twitter discourse.

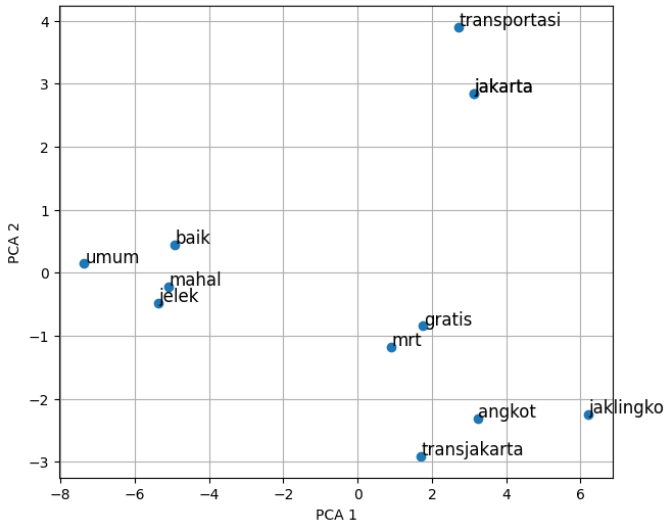

Figure 2. PCA Visualizations of Word2Vec

The visualization of selected word vectors projected into a two-dimensional space using Principal Component Analysis (PCA) is presented in Figure 2. The plot illustrates meaningful semantic groupings — for instance, words such as *"transportasi," "jakarta," "mrt,"* and *"transjakarta"* cluster closely, reflecting their contextual similarity related to public transportation. Conversely, sentiment-related words such as *"jelek," "mahal,"* and *"baik"* appear distinctly separated, indicating the model's ability to differentiate opposing sentiment polarities in the embedding space.

The GloVe embeddings used in this study were obtained from a publicly available Indonesian corpus on Kaggle (2024), originally trained on large-scale co-occurrence statistics of Indonesian text. Since GloVe representations are static and not directly trainable, the vectors were first converted into the Word2Vec format and subsequently fine-tuned using the same tweet corpus employed in this research. This fine-tuning allowed the embeddings to better capture domain-specific nuances and sentiment-bearing expressions that frequently appear in social media discussions about public transportation.

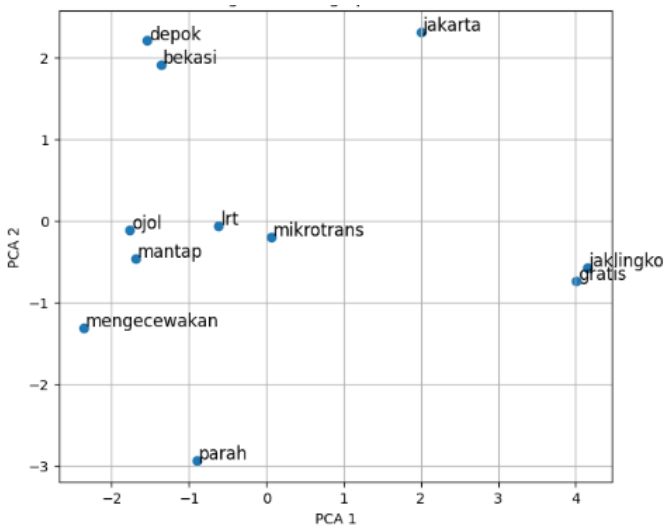

Figure 3. PCA Visualizations of GloVe

The two-dimensional PCA visualization of several representative words, presented in Figure 3, shows that semantically related words are positioned close to each other. For instance, *"jaklingko"* appears near *"gratis"*, which aligns with public discussions emphasizing the free-fare aspects of the JakLingko service. Similarly, positive expressions such as *"mantap"* are located close to terms like *"ojol"*, reflecting positive sentiment often associated with online transportation services. Conversely, negative expressions such as *"parah"* and *"mengecewakan"* occupy distant regions of the semantic space, highlighting the model's ability to distinguish between contrasting emotional tones within transportation-related discourse.

For FastText, the pretrained embeddings were obtained from the Facebook AI Research repository (cc.id.300.vec), which provides subword-level representations capable of handling out-of-vocabulary and morphologically rich tokens. Similar to the previous embeddings, the model was fine-tuned using the collected tweet dataset, enabling the subword mechanism to adapt to the stylistic and orthographic variations common in Indonesian social media text—such as repetitive characters or informal spellings (e.g., *"jaklingko"*, *"jaklingkocc"*, *"jaklingkologo"*). This fine-tuning process allows FastText to better generalize across noisy or non-standard lexical forms while retaining its pretrained semantic knowledge.

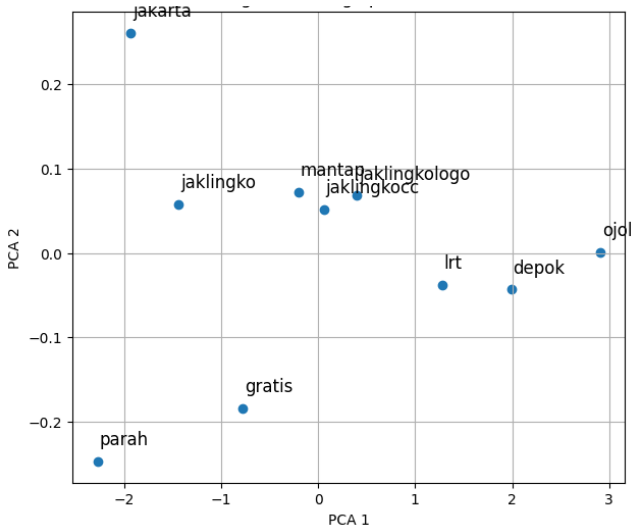

Figure 4. PCA Visualizations of FastText

The visualization of word vectors in Figure 4 shows that semantically related or contextually similar tokens tend to cluster closely within the embedding space. Words such as *"jaklingko," "jaklingkocc,"* and *"jaklingkologo"* appear in nearly identical positions, indicating that the model effectively captures their shared subword structures and orthographic similarities. Interestingly, the token *"mantap"* is positioned near this cluster, suggesting that user discussions about *Jaklingko* often carry a positive sentiment or approval tone. In contrast, other terms such as *"gratis," "parah,"* and *"ojol"* occupy more distant regions, reflecting distinct semantic and affective contexts within Jakarta's transportation discourse. These findings indicate that FastText's subword modeling performs well in representing informal and noisy Indonesian text, particularly for domain-specific terms that frequently appear in multiple surface variations on social media.

After constructing and fine-tuning the three word embedding models (Word2Vec, GloVe, and FastText), each embedding representation was integrated into a Long Short-Term Memory (LSTM) architecture for sentiment classification. In this study, the same LSTM architecture and hyperparameters were applied across all embedding types to ensure comparability. The model consisted of a single LSTM layer with 64 hidden units, followed by a dense layer with ReLU activation and a final softmax output for binary classification (positive vs. negative). Regularization techniques, including dropout (rate = 0.5) and L2 weight decay ($\lambda = 0.001$), were employed to prevent overfitting. Each model was trained for up to 30 epochs with early stopping based on validation loss, using a batch size of 32 and the Adam optimizer (learning rate = 0.0005).

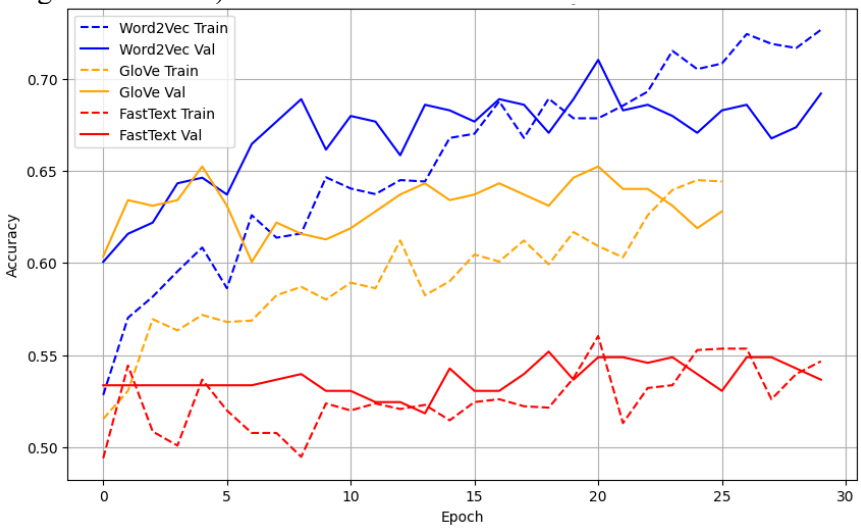

Figure 5. Train vs Validation Accuracy

The training and validation accuracy curves for the three LSTM models are presented in Figure 5. All models show a clear upward trend in both training and validation accuracy, indicating that the learning process was stable and free from severe overfitting. However, the degree of convergence differs across embeddings. The Word2Vec-based model demonstrates the most consistent improvement, with its validation accuracy closely following the training curve throughout the epochs — suggesting good generalization and well-aligned learning between training and unseen data. In contrast, the GloVe model exhibits moderate performance, with validation accuracy slightly lower and more variable, indicating slower adaptation to the domain-specific language of social media. The FastText model performs the weakest and most unstable across epochs, with training and validation curves showing minimal growth, reflecting difficulties in capturing consistent sentiment representations despite its subword modeling advantage.

When focusing specifically on validation performance (Figure 5), the Word2Vec-based LSTM again achieved the highest validation accuracy throughout training, maintaining a clear margin above the GloVe and FastText models. The GloVe model followed with moderate and relatively stable accuracy, while the FastText model consistently showed the lowest validation scores, suggesting that its representations were less effective for sentiment discrimination.

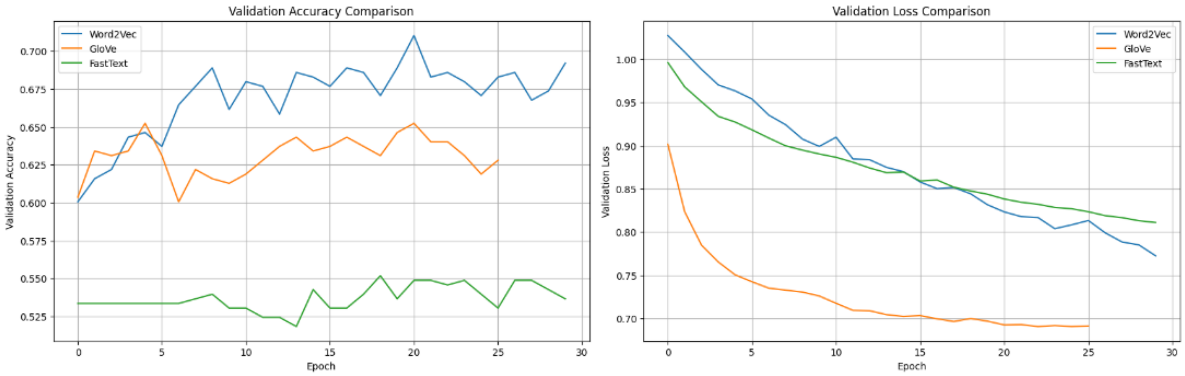

Figure 6. Comparison of Validation Accuracy and Validation Loss

A similar trend can be seen in the validation loss curves (Figure 6). Although GloVe achieved a lower absolute validation loss, the Word2Vec model exhibited a steadier downward trajectory, indicating more stable convergence and stronger generalization capacity. Meanwhile, the FastText model's slower loss reduction aligns with its lower validation accuracy, confirming its weaker adaptation to the tweet corpus.

Table 2. Evaluation Metrics of LSTM Models with Word2Vec, GloVe, and FastText Embedding

Embedding Type	Accuracy	Precision	Recall	F1 Score	AUC
Glove	0.675214	0.665263	0.820779	0.734884	0.746954
Fasttext	0.548433	0.551051	0.953247	0.698382	0.561145
Word2Vec	0.742165	0.757576	0.779221	0.768246	0.800328

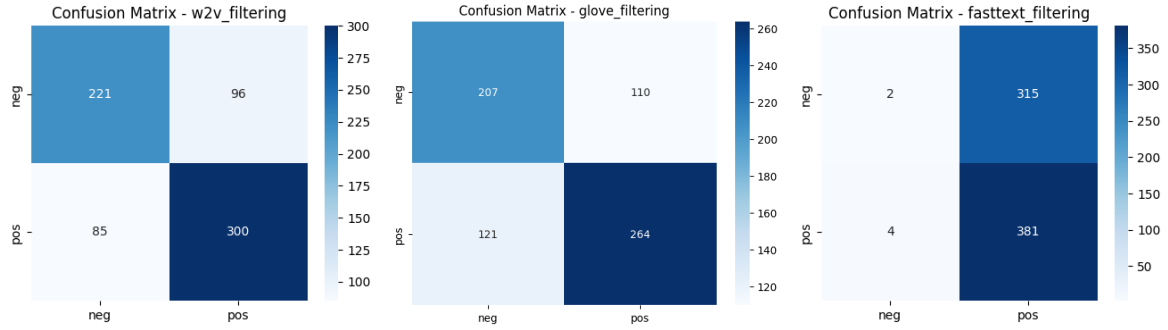

Figure 7. Confusion Matrix of LSTM Models with Word2Vec, GloVe, and FastText Embedding

The quantitative evaluation results of the three embedding models are summarized in Table 2, while the corresponding confusion matrices are shown in Figure 7. Overall, the Word2Vec-based LSTM achieved the highest performance across all metrics, with an accuracy of 0.74, an F1-score of 0.77, and an AUC of 0.80. This model produced a well-balanced classification between positive and negative sentiments, as reflected in its confusion matrix, where both classes were predicted with relatively similar precision and recall values. The GloVe-based model ranked second, yielding an accuracy of 0.68 and F1-score of 0.73. Although slightly lower than Word2Vec, its confusion matrix shows a moderate balance between the two sentiment classes, suggesting that the static nature of GloVe embeddings still preserves sufficient semantic information for sentiment discrimination. In contrast, the FastText-based model performed notably worse, with accuracy and AUC values of 0.55 and 0.56, respectively. Its confusion matrix reveals a strong bias toward the positive class, where nearly all samples were classified as positive regardless of their true labels. This imbalance indicates that while FastText's subword mechanism helps in capturing morphological variations, it tends to overfit to frequent surface patterns (e.g., positive expressions such as "bagus," "mantap," "terbaik") and fails to adequately represent less frequent or more context-dependent negative terms. Such behavior highlights a limitation of FastText embeddings in handling sentiment polarity when trained on informal and domain-specific social media text.

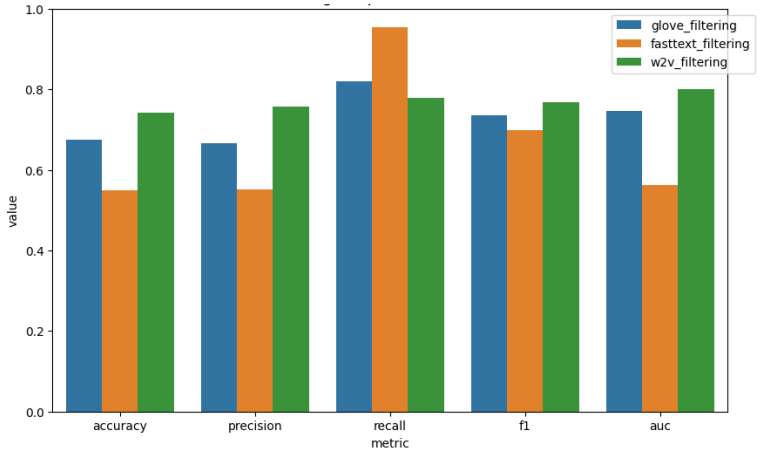

Figure 8. Embedding Comparison Across Metric

Figure 8 presents a comparative visualization of model performance across five metrics: accuracy, precision, recall, F1-score, and AUC. Consistent with previous findings, the Word2Vec-based model achieved the best overall performance, showing strong results in all metrics, particularly in AUC (0.80) and F1-score (0.77). The GloVe model performed moderately well, maintaining balanced precision and recall values, while the FastText model, despite having the highest recall, exhibited substantially lower precision—indicating that it tended to overpredict the positive class, as seen in its confusion matrix. These results suggest that Word2Vec embeddings provide the most stable and discriminative representation for sentiment classification on Indonesian tweets.

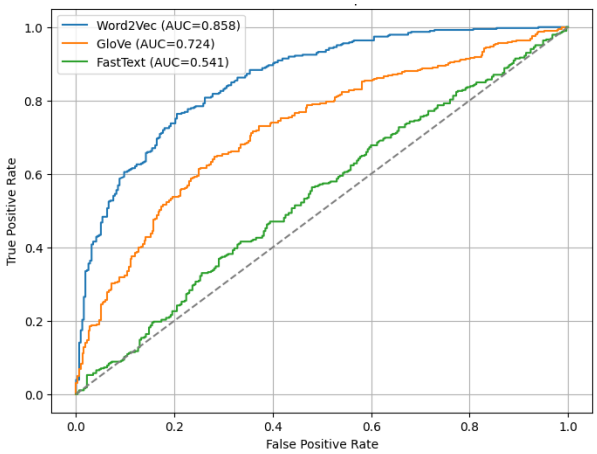

Figure 9. ROC Curve Comparison

Figure 9 shows the Receiver Operating Characteristic (ROC) curves for the three models. The ROC curve illustrates the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR) at different classification thresholds. A model with a curve closer to the top-left corner indicates better discriminative ability, as it achieves a higher TPR with fewer false positives. Among the three models, Word2Vec shows the steepest and highest ROC curve with an AUC of 0.86, confirming its superior classification capability. The GloVe model follows with an AUC of 0.72, reflecting fair performance, while the FastText model has an AUC of 0.54, which is only slightly above random guessing (AUC = 0.5). This indicates that the FastText model struggled to effectively separate positive and negative sentiments, likely due to its bias toward the dominant class observed earlier.

4. DISCUSSIONS

The results of this study demonstrate that the choice of word embedding has a substantial impact on the performance of LSTM models for sentiment analysis in Bahasa Indonesia tweets. Among the three embeddings evaluated—Word2Vec, GloVe, and FastText—the Word2Vec-based model consistently achieved the best performance across all metrics, including accuracy, F1-score, and AUC.

This indicates that Word2Vec embeddings, when fine-tuned with domain-specific data, can effectively capture the semantic nuances and sentiment polarity of Indonesian social media text. Such findings align with previous studies [22], [23] that highlight the adaptability of Word2Vec for Indonesian NLP tasks due to its capacity to preserve contextual relationships between words. In contrast, the GloVe model demonstrated competitive but slightly lower results. Although its co-occurrence-based training provides a strong general representation of Indonesian text, the lack of contextual updating during fine-tuning may have limited its sensitivity to domain-specific slang and informal expressions often found in social media. This pattern is consistent with observations by [24] who noted that pretrained GloVe models tend to perform well on formal corpora but require additional adaptation to capture sentiment in noisy text domains.

Meanwhile, the FastText model, despite its subword-level representations, showed the weakest performance, particularly in distinguishing negative sentiment. The confusion matrix revealed a bias toward predicting positive classes, suggesting that the subword mechanism may have overemphasized frequent positive morphemes (e.g., “baik,” “bagus,” “mantap”) and failed to generalize effectively to less frequent negative patterns. This finding contrasts with reports in multilingual sentiment studies [12], [16] where FastText generally performs well for morphologically rich languages. The discrepancy in this study could be attributed to the high lexical variability and informal orthography in Indonesian tweets, which pose additional challenges for subword-based models.

Another key insight from this research is that fine-tuning pretrained embeddings using domain-specific data significantly improves model performance across all embedding types. This approach allows models to retain general linguistic knowledge while adapting to the syntactic and semantic peculiarities of social media language—an advantage over purely pretrained or purely data-specific embeddings. This hybrid training strategy aligns with recent findings by [26], who reported that integrating pretrained vectors with in-domain fine-tuning enhances contextual understanding and improves generalization on Indonesian sentiment datasets.

Overall, the superior performance of the Word2Vec-based LSTM underscores its robustness for sentiment analysis in Bahasa Indonesia, particularly for user-generated text with informal language. Compared to prior works that relied solely on pretrained or monolingual embeddings, this study contributes a systematic comparison of three major embedding methods under identical modeling conditions, providing empirical evidence that Word2Vec remains a strong and efficient baseline for Indonesian sentiment analysis tasks.

5. CONCLUSION

This study compared the performance of three word embedding models—Word2Vec, GloVe, and FastText—combined with an identical LSTM architecture for sentiment classification of Indonesian tweets. The experiments revealed that the Word2Vec-based model consistently outperformed the others in accuracy, F1-score, and AUC, followed by GloVe and FastText. This superior performance reflects Word2Vec's strong capability to capture semantic relationships through contextual word co-occurrence, particularly after fine-tuning with domain-specific tweet data.

The integration of pretrained embeddings from the Indonesian Wikipedia corpus with the collected tweet dataset proved highly beneficial. This hybrid approach enabled the model to maintain general semantic understanding while adapting to the informal and noisy language characteristic of social media. GloVe embeddings also performed well but were constrained by their static nature, which limited their flexibility in handling linguistic variations. Meanwhile, FastText, though designed to manage subword information, showed lower accuracy, possibly due to excessive noise and inconsistent spelling in the dataset that reduced its effectiveness.

Overall, these findings highlight that domain adaptation through fine-tuning significantly enhances sentiment classification performance, especially for low-resource languages like Bahasa Indonesia. Word2Vec remains a reliable and efficient choice, balancing interpretability and computational simplicity. Future work should explore contextualized models such as IndoBERT to assess their advantages in representing nuanced sentiment. In conclusion, this research provides empirical evidence that careful embedding selection and fine-tuning strategies are key to improving LSTM-based sentiment analysis on Indonesian social media data.

6. ACKNOWLEDGMENTS

The authors would like to thank the Faculty of Mathematics and Natural Sciences, Universitas Negeri Jakarta (FMIPA UNJ) for supporting this research under Research Grant No. 153/PPK-PEN/FMIPA/2025.

7. REFERENCES

[1] J. Gantz and D. Reinsel, "The Digital Universe in 2020: Big Data, Bigger Digital Shadows, and Biggest Growth in the Far East," IDC White Paper, Dec. 2012.
[2] S. Kemp, "Digital 2024: Indonesia," DataReportal, 2024. [Online]. Available: https://datareportal.com/reports/digital-2024-indonesia
[3] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning-based natural language processing,"IEEE Comput. Intell. Mag., vol. 13, no. 3, pp. 55–75, Aug. 2018.

[4] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Eng. J., vol. 5, no. 4, pp. 1093–1113, Dec. 2014.
[5] M. Ramadhani and H. S. Goo, "Twitter sentiment analysis using deep learning methods," in *Proc. 2017 7th Int. Annual Engineering Seminar (InAES)*, Yogyakarta, Indonesia, Aug. 2017, pp. 1–6, doi: 10.1109/InAES.2017.8068556
[6] A. Agarwal and N. Mittal, "Machine learning approach for sentiment analysis," in *Prominent Feature Extraction for Sentiment Analysis*, Socio-Affective Computing Series, Cham, Switzerland: Springer, 2016, pp. 21–45, doi: 10.1007/978-3-319-25361-0_2.
[7] F. Ladayya, D. Siregar, W. E. Pranoto, and H. D. Muchtar, "Analisis sentimen pada program transportasi publik JakLingko dengan metode Support Vector Machine," Jurnal Statistika dan Aplikasinya, vol. 6, no. 2, pp. 93–104, 2022.
[8] D. Siregar, F. Ladayya, N. Z. Albaqi, and B. M. Wardana, "Penerapan metode Support Vector Machine dan Naïve Bayes Classifier dalam analisis sentimen publik terhadap konsep child-free di Twitter,"Jurnal Statistika dan Aplikasinya, vol. 7, no. 1, pp. 109–120, 2023.
[9] H.-T. Duong and T.-A. Nguyen-Thi, "A review: preprocessing techniques and data augmentation for sentiment analysis," Computational Social Networks, vol. 8, no. 1, pp. 1–16, 2021, doi: 10.1186/s40649-020-00083-6.
[10] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Engineering Journal, vol. 5, no. 4, pp. 1093–1113, 2014.
[11] D. E. Cahyani and I. Patasik, "Performance comparison of TF-IDF and Word2Vec models for emotion text classification," Bull. Electr. Eng. Informatics, vol. 10, no. 5, pp. 2647–2654, 2021.
[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv:1301.3781, 2013.
[13] C. Wang, P. Nulty, and D. Lillis, "A comparative study on word embeddings in deep learning for text classification," Proc. 4th Int. Conf. NLPIR, pp. 37–46, 2021.
[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[15] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, "Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis," J. Comput. Sci., vol. 27, pp. 386–393, 2018.
[16] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, "Sentiment analysis using Word2Vec and long short-term memory (LSTM) for Indonesian hotel reviews," Procedia Comput. Sci., vol. 179, pp. 728–735, 2021.
[17] A. Zouzou and I. El Azami, "Text sentiment analysis with CNN & GRU model using GloVe," in Proc. 5th Int. Conf. Intell. Comput. Data Sci. (ICDS), Fez, Morocco, Oct. 2021.
[18] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning-based natural language processing," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, pp. 55–75, 2018.
[19] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1532–1543.
[20] Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[21] C. C. Aggarwal, *Neural Networks and Deep Learning: A Textbook*. Cham, Switzerland: Springer, 2023.
[22] Kurniawan and S. Louvan, "IndoSum: A new benchmark dataset for Indonesian text summarization," in *Proceedings of the 2018 International Conference on Asian Language Processing (IALP)*, Bandung, Indonesia, 2018, pp. 15–17, doi: 10.1109/IALP.2018.8629109.
[23] H. Juwiantho, E. I. Setiawan, J. Santoso, and M. H. Purnomo, "Sentiment analysis Twitter Bahasa Indonesia berbasis Word2Vec using deep convolutional neural network," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 7, no. 1, pp. 218–225, 2020, doi: 10.25126/jtiik.202071218.
[24] P. Gupta, N. Roy, G. Batra, and A. K. Dubey, "Decoding emotions in text using GloVe embeddings,"

in Proc. 2021 Int. Conf. on Computing, Communication, and Intelligent Systems (ICCCIS), Feb. 2021, pp. 1–6, doi: 10.1109/ICCCIS51004.2021.9397132.

[25] R. I. Perwira and V. A. Permadi, "Domain-specific fine-tuning of IndoBERT for aspect-based sentiment analysis in Indonesian travel user-generated content," *Journal of Information Systems Engineering and Business Intelligence*, vol. 11, no. 1, pp. 30–40, 2025, doi: 10.20473/jisebi.11.1.30-40.