Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.5, October 2025, pp. 2574~2587
e-ISSN: 2548-6861

Evaluating Fasttext and Glove Embeddings for Sentiment Analysis of AI-Generated Ghibli-Style Images

I Gusti Ngurah Sentana Putra [1]*, Muhammad Yusran[2], Jefita Resti Sari[3], Cici Suhaeni[4], Bagus Sartono[5], Gerry Alfa Dito[6]
* Statistics and Data Science, Department of Statistics, IPB University, Bogor, Indonesia
ngurahsentana@apps.ipb.ac.id [1], muhammadyusran@apps.ipb.ac.id [2], jefitarestisari@apps.ipb.ac.id [3], cici_suhaeni@apps.ipb.ac.id [4], bagusco@apps.ipb.ac.id[5], gerryalfadito@gmail.com[6]

Article Info

Article history:

Received 2025-07-31
Revised 2025-09-03
Accepted 2025-09-10…

Keyword:

Sentiment Analysis,
Word Embedding,
FastText,
GloVe,
Ghibli AI.

ABSTRACT

The development of text-to-image generation technology based on artificial intelligence has triggered mixed public reactions, especially when applied to iconic visual styles such as Studio Ghibli. This research aims to evaluate public sentiment towards the phenomenon of Ghibli-style AI images by comparing two static word embedding methods, namely FastText and GloVe, on three classification algorithms: Logistic Regression, Random Forest, and Convolutional Neural Network (CNN). Data in the form of Indonesian tweets were collected from Twitter using hashtags such as #ghibli, #ghiblistyle, and #hayaomiyazaki during the period 25 March to 25 April 2025. Each tweet was manually labelled with positive or negative sentiment, then preprocessed and represented using pre-trained FastText and GloVe embeddings. Evaluation was conducted using accuracy, precision, recall, and F1-score metrics, both macro and weighted. Results showed that FastText consistently performed the best on most models, especially in terms of precision and overall accuracy, thanks to its ability to handle sub-word information and spelling variations in social media texts. The combination of CNN with FastText yielded the highest performance with a macro F1-score of 76.56% and accuracy of 84.69%. However, GloVe still showed competitive performance in recall on the Logistic Regression model, making it relevant for contexts that prioritise sentiment detection coverage. This study emphasizes the importance of selecting embeddings and models that are appropriate to the characteristics of the data and the purpose of the analysis in informal social media-based sentiment classification.

I. I. INTRODUCTION

The development of digital technology has led to a significant increase in the volume of text- based data, especially through social media, online forums, and other interactive platforms. This has created an urgency to understand, classify and analyse text information systematically. As Artificial Intelligence (AI) technology continues to advance year by year, text-to-image generation approaches are also gaining popularity, especially in digital image generation from text-based inputs. One trend that stands out today is Studio Ghibli-style visuals that can be found on various platforms such as Twitter, Instagram, TikTok, and Facebook. Ghibli's signature soft visuals, pastel colour aesthetics, magical, naturalistic themes, and nostalgic nuances have attracted the attention of

global internet users. Some of the hashtags that are indicators of the increasing preference for Ghibli Style photo visuals are #ghibli, #ghiblistyle, and #hayaomiyazaki.

This visual popularity of Ghibli is inseparable from the emotional power that Studio Ghibli's own films carry. Studio Ghibli through its stunning animation and emotion-filled narratives, with positive sentiments predominant especially for films such as Spirited Away, My Neighbor Totoro, and Kiki's Delivery Service. Analysis of data from platforms such as IMDb and Rotten Tomatoes shows that these films trigger nostalgia and happiness, while Grave of the Fireflies is more associated with deep sadness, reflecting Ghibli's ability to handle the complexity of human emotions. As AI is increasingly used to mimic Ghibli's visual style, there have been mixed reactions from the public. Fans felt emotionally connected when personal photos were transformed into Ghibli-esque illustrations, creating a positive wave on social media. However, critics emerged who denounced the use of AI to mimic Ghibli's style as an insult to the work and an unauthorised exploitation of creativity. This negative sentiment was reinforced by concerns over copyright infringement and marginalisation of human artists. In this context, sentiment analysis became a capable method to understand the public's perception of this phenomenon.

Sentiment analysis is a part of text classification in the field of Natural Language Processing (NLP) to handle text classification. This approach classifies documents into specific categories based on the meaning and context of their contents. In previous research, text classification has proven effective in various domains such as sentiment analysis, public opinion detection, and thematic visual content recognition [1], [2]. Studies such as Khasanah (2021) show that visual trends can be effectively identified using text-based classification models on social media data [1].

For text classification to be performed effectively, the representation of words in numerical form is necessary. Two static embedding approaches that can be used in NLP are GloVe and FastText. GloVe generates word vectors based on co-occurance statistics in a large corpus, while FastTesxt enhances embedding by relying on character n-grams that enable better recognition of new or rare (out-of- vocabulary) words [3]. With these characteristics, FastText and Glove are often used in the analysis of informational text found on social media [3], [4]. While contextual embeddings such as BERT have achieved state-of-the-art performance across multiple NLP tasks by leveraging bidirectional context during pre-training [5], static embeddings like GloVe [6] and FastText [7] remain highly relevant in this study due to their lower computational cost, robustness to noisy social media text, and usefulness as interpretable baselines for evaluating future advances in contextual models. Their computational efficiency makes them suitable for initial comparative experiments, and their robustness in handling noisy, informal social media text has been well-documented. FastText's incorporation of subword information is particularly effective for handling misspellings and out-of-vocabulary terms [7], while GloVe's global co-occurrence statistics enable strong semantic representations [6]. These characteristics provide a solid baseline for evaluating sentiment analysis models, against which more complex contextual models can be compared in future work. Moreover, static embeddings continue to be employed effectively in sentiment analysis [6] and text classification tasks [5], reinforcing their importance as foundational baselines in NLP research.

Several studies have comparedthe effectiveness of GloVe and FastText in various sentiment analysis contexts. Research [8] shows that the accuracy of the GloVe-CNN-BiLSTM model can reach 0.9565 on full text datasets, 0.9509 on long text datasets, and 0.9560 on short text datasets, which is much higher than the CNN- BiLSTM model and the Text CNN model. Another study also showed that GloVe with the CNN-BiLSTM model showed good performance with an accuracy of 99.43% [9]. Research [10] shows that FastText is better than other word embedding in hotel review sentiment analysis with anaccuracy of 93%.

The importance of word representation through embedding such as GloVe and FastText in text analysis is followed by the selection of machine learning models used for classification. The machine learning model plays a crucial role in producing optimal performance. In addition to the selection of appropriate embedding such as GloVe and FastText, the selection of classification algorithms also has a significant effect on the performance of sentiment analysis. Various machine learning approaches have been used in previous studies, ranging from classical models such as Logistic Regression, ensemble models such as Random Forest (RF), and deep learning approaches such as Convolutional Neural Network (CNN). Logistic Regression is known to be simple yet effective in linear text classification. Random Forest, with its ability to combine multiple decision trees, is able to capture complex non-linear relationships in data. Meanwhile, CNN has proven to be very good at capturing spatial patterns in text representations thanks to its ability to recognize local features through convolution operations. The use of CNN in NLP is also becoming increasingly popular due to its ability to extract semantic features from text mapped into numeric embeddings such as GloVe and FastText.

However, a clear research gap exists. While sentiment analysis is widely applied, there is a scarcity of studies focusing on the public perception of AI-generated art, particularly within specific cultural and stylistic contexts like Studio Ghibli. This gap is even more pronounced in the Indonesian digital sphere, where the use of native language and local social media platforms creates a unique linguistic landscape for analysis [11], [12]. Furthermore, comprehensive comparisons of classical, ensemble, and deep learning models specifically Logistic Regression, Random Forest, and CNN when paired with different static embeddings for this specific task are not thoroughly explored in the existing literature. Most studies tend to focus on either model comparison or embedding evaluation, but not their combined effect on a novel, emotionally-charged dataset like public reactions to AI Ghibli-style art. Previous works such as Devlin et al. (2019) and Peters et al. (2018) emphasize the strength of contextual embeddings, yet few directly benchmark them against static embeddings within Indonesian contexts, reinforcing the importance of establishing strong baselines with FastText and GloVe before extending to contextual models.

In the context of this study, a comparison of the performance of two embedding methods GloVe and FastText was conducted on three different types of classification models Logistic Regression, Random Forest, and CNN. The aim is to evaluate the extent to which the combination of embedding and classification algorithms can affect the accuracy and effectiveness of sentiment analysis on the phenomenon of AI-generated Ghibli-style images By comparing the performance of each combination, this research is expected to provide insight into the optimal approach in modelling public perception of AI-based visual trends, especially those with strong emotional and artistic dimensions such as Ghibli. The evaluation is based on metrics such as accuracy, precision, recall, and F1-score, so that the results obtained not only consider the accuracy of the model, but also its balance in handling data that may not be class balanced. The findings of this research are expected to not only contribute to the development of more accurate text classification methods, but also enrich the understanding of public response to AI-based visual trends that represent Studio Ghibli's signature style.

II. METHODS

A. Data

The data used in this study was obtained through a web scraping process from the Twitter platform, using relevant hashtags such as #ghiblistyle, #ghibli, #hayaomiyazaki, and #ghiblistudio. The selection of these hashtags was based on their popularity and relevance to the main topic of the research, namely Ghibli Style AI. The focus of the analysis was on tweets that discussed the Ghibli Style AI phenomenon. A total of 6,434 tweets were collected during the period 25 March 2025 to 25 April 2025**,** i.e. from the day the technology was launched until two weeks later, to capture the initial dynamics of public response on social media.

B. Text Representation

Text representation is an important stage in natural language processing (NLP), which converts raw text into a numerical format for use by machine learning algorithms. The representation usually takes the form of word embeddings, which are fixed-dimensional vectors that capture the semantic information of words. According to research [3] text representation methods convert textual information into a numerical format that preserves the syntactic and semantic relationships between words. This allows machines to understand the context of words in sentences, especially in social media data that is full of variations.

C. Fasttext

Fasttext is a word embedding method developed from Word2Vec [7], [13]. FastText was developed by Facebook AI Research and is an embedding technique that considers sub-words through character n-grams, thus being able to better handle new or rare words.

FastText follows the skip-gram model and this vector model is an extension of the Word2Vec model. One important advantage of fastText embedding is that it can consider the raw text of any social media post. According to research [3] FastText generates word placement by considering subword information using n-gram characters, which allows it to effectively handle words outside the vocabulary, especially on text data such as tweets.

FastText word representation learning considers a sliding window for right and left context words [4]. Context word selection is done by applying a sliding window that has a fixed size.

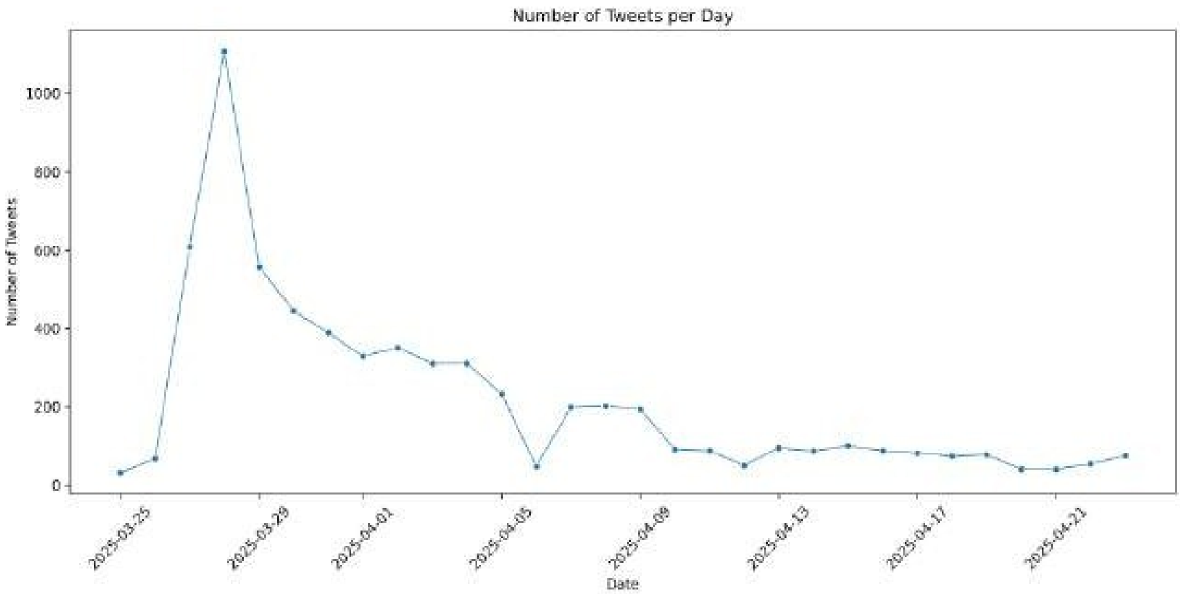


Figure 1. Number of Tweets per Day

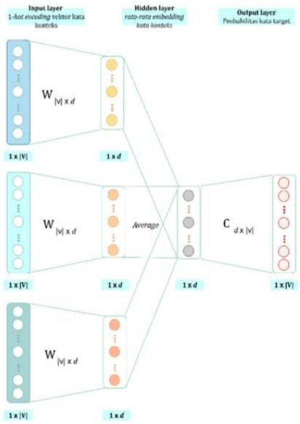


Figure 2. General architecture of the CBOW scheme [10]

Any word that falls within the range of that window from the target word will be considered as a context word. This vector model is based on the concept of char-n grams and can provide embedding for missing or rare words, misspelled words. This is particularly relevant in social media data which often contains slang, typos, or nonstandard word forms.

Structurally FastText is similar to the continuous bag-of-words (CBOW) model, the architecture of the FastText classification model is aligned with the CBOW framework. The CBOW architecture consists of three layers, namely an input layer with as many neurons as the unique vocabulary ||V|| from the training data, a hidden layer with as many neurons as the desired tour dimension, and an output layer with size ||V|| [14].

D. Glove

GloVe (Global Vectors for Word Representation) is a global statistics-based embedding technique that calculates the frequency of co-occurrence between words in a large corpus. This allows GloVe to capture semantic meaning very well. According to research [8] states that the GloVe word embedding technique captures global occurrence statistics of words from a large corpus, resulting in high semantic accuracy in NLP tasks. In research [8] GloVe combined with CNN-BiLSTM achieved 95.65% accuracy in sentiment analysis of COVID-19-related comments.

The implementation method of Glove, first, builds a word cooccurrence matrix based on the whole corpus, next, the learning word vector is processed based on the cooccurrence matrix and the GloVe model. The GloVe model is shown in Fig. 3.

The GloVe process starts by building a vocabulary from the collected text corpus, collecting unique words, and counting their frequencies to determine the vector weights [8]. Each sentence is then converted into a vector by considering the dimension of the vector and the speed of the algorithm. The core idea of GloVe is contained in the formula [6]:

$$J = \sum^{V}_{I,j=1} f(X_{ij})\ \left(W^T\tilde{w}_j + b_i + \tilde{b} - \log(X_{ik})\right)^2 \quad (1)$$

Where V is the size of the vocabulary,generally the window size is 5~10. The GloVe model consists of several core components that work together to capture the relationship between words in a text. The co-occurrence frequency between word i and context word j, denoted by Xij, provides information on how often the two words co-occur. The vectors wi and $\tilde{w}j$ represent the semantic meaning of the main word and context word, allowing the model to understand their relationship in vector space. The bias components b_i and $\tilde{b}_j$ serve to accommodate systematic biases that may be present in both the word and the context, thus improving the prediction accuracy. In addition, the use of logarithm in co-occurrence frequency $log(Xik)$, helps normalise the data to make the trainingprocess more stable and efficient. Combining all these elements, GloVe is able to produce a meaningful word vector representation based on the context of its use in the corpus [15].

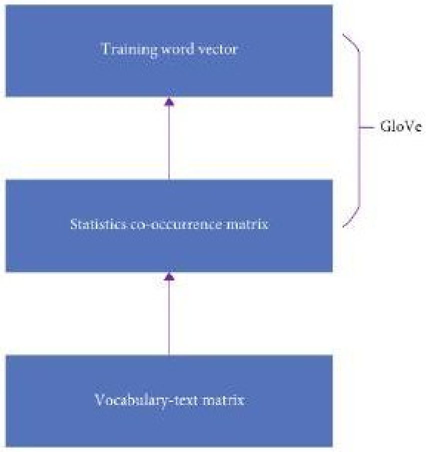


Figure 3. GloVe Model

The function $f(x)$ is a weight function where$f(x)$ must follow the characteristics. When the cooccurrence value of the word is 0, the weight is equal to 0. When the coocurrence value is larger, the weight will not decrease, meaning that $f(x)$ fulfils the continuity and non-decreasing conditions. When words occur too frequently, there will be no overweighting, that is, $f(x)$ can be given a relatively small value. The weighting of $f(x)$ has the following formula:

The function $f(x)$ is a weight function where $f(x)$ must follow the characteristics. When the cooccurrence value of the word is 0, the weight is equal to 0. When the coocurrence value is larger, the weight will not decrease, meaning that $f(x)$ fulfils the continuity and non-decreasing conditions. When words occur too frequently, there will be no overweighting, that is, $f(x)$ can be given a relatively small value. The weighting of $f(x)$ has the following formula:

$$f(x) = \begin{cases} \left(\frac{x}{x_{\max}}\right)^{\alpha}, x < x_{max} \\ 1,\ x \geq x_{max} \end{cases} \quad (2)$$

The effect is better through experiments When $x_{max} = 100, \alpha = 0.75$, GloVe can directly use the word vector of the document corpus itself for calculation, and has strong manoeuvrability and high flexibility [8].

E. Logistic regression

Logistic regression is a supervised machine learning technique used to analyse data and describe the relationship between several predictor variables and one response variable. In logistic regression, the value of the response variable is limited to 0 to 1, so it can be used to classify sentiment as positive or negative with a decision threshold of 0.5 [16]. The initial equation of logistic regression can be seen in the following formula:

$$z = b + (w_1 \times x_1) + \cdots (w_n \times x_n) \tag{3}$$

In the equation, b represents the bias value, w represents the data in vector form, and x represents the weight of the data. This function calculates the weight of each feature in vector x by multiplying the feature with its weight [16]. After the z value is obtained, to get a probability value that ranges between 0 and 1, the z value will be entered into the sigmoid function with the formula:

$$Sigmoid(z) = \frac{1}{1+e^{-z}} \tag{4}$$

After using the equation, the results obtained will be compared to determine the category or classification based on the following criteria:

1. If the result is more than 0.5, then the prediction is in the positive category.
2. If the result is less than 0.5, then the prediction is in the negative category.

The results of sentiment analysis using the Logistic Regression method show good performance in classifying public opinion related to certain topics. In research on one hundred percent evaluation of Face-to-Face Learning (PTM), the Logistic Regression model managed to achieve an accuracy of 78.57%, precision 76.92%, recall 83.3%, and F1-Score 80%, showing the model's ability to distinguish positive and negative sentiments quite accurately [16]. Meanwhile, in the sentiment analysis of Sirekap application reviews, Logistic Regression showed a higher accuracy of 91%, with precision for positive and negative classes of 90% and 92% respectively, recall 94% and 87%, and F1-Score 92% and 90% [16]. These results confirm that Logistic Regression is effective in processing complex text data to produce sentiment classifications that can assist public opinion-based decision making.

F. Random Forest

Random Forest is an ensemble method that uses multiple decision trees to improve stability and classification accuracy. This method is very effective in handling complex and imbalanced data. According to research [17], a random forest is a combination of predictor trees such that each tree depends on a random vector value that is sampled independently and with the same distribution for all trees in the forest. RF works by forming a collection of decision trees and combining the voting results from each tree to improve text classification accuracy.

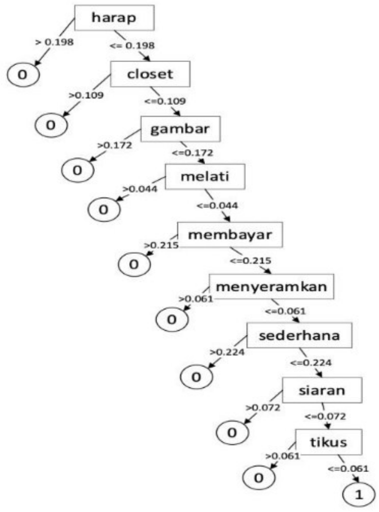


Figure 1. Example of a Decision Tree Structure [10]

In [18], Random Forest was applied to analyse sentiment data from Twitter with an accuracy rate of about 75%, indicating that this model is already good enough in classifying people's opinions although there is still room for development by trying other algorithms. In addition, in a study [19] that compared Random Forest with Naïve Bayes for WhatsApp message classification, Random Forest obtained an accuracy of 86%. Random Forest remains a popular method due to its ability to handle large and complex data and produce stable models in text classification. These results indicate that Random Forest is a reliable choice for various text processing and analysis applications..

G. Convolutional Neural Network

Convolutional Neural Network (CNN) is a classification technique that utilises specialised layers to process the input using convolution filters. CNN consists of two main stages, namely the feature learning stage and the classification stage. In the feature learning stage, the network uses convolutional layers, activation functions such as ReLU, and pooling layers to extract and reduce features from the input data. While in the classification stage, the extracted data is converted into vector form through a flattening process, and then processed by the fully-connected layer to produce the final prediction. Each process in CNN includes two important steps, namely feed-forward to pass the input to the output, and backpropagation to update the weights based on prediction errors [20]. In text classification, CNN uses filters to extract important features from each region. CNN input in word representation will go through two layers, namely convolution layer and max-pooling layer [21].

In the convolution layer, the input data is processed using a number of filters W_l to capture important features from a specific area with a predetermined region size. For example, the vector representation of the i-th word is symbolised as x_i, and the combined vectors from x_i to x_{i+h-1} are represented as $X[i:i+h-1]$ Next, the feature vector $c = [c_1, c_2, \ldots, c_{nh+1}]$is calculated using the following equation.

$$c_i = f\left(\sum_{k=1}^{h} \sum_{j=1}^{d} X_{[i:i+h-1]k,j} \cdot W_{k,j}\right) \tag{5}$$

For each filter used, where indices j and k denote the row and column positions in the input matrix, respectively, and f is the nonlinear activation function applied in the process [21].

The max pooling layer works by taking the highest value of each feature vector $\boldsymbol{c}$, expressed as $\hat{c} = \max\{\boldsymbol{c}\}$, from the output of the convolutional layer. The main function of this layer is to shrink the input dimension, so that CNN can learn important patterns using simpler and compressed data [21].

In [22], the CNN architecture used consists of three main layers: Input Layer, Feature-extraction Layer, and Classification Layer. The input layer contains the word representation matrix obtained from the pre-training process, such as one-hot encoding.

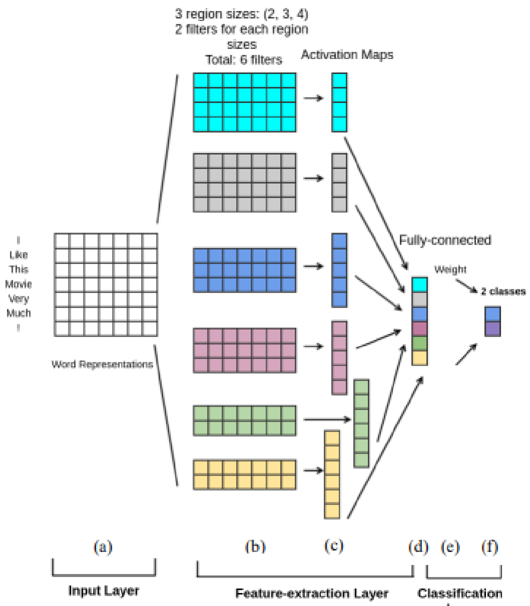


Figure 2. CNN Architecture [21]

CNNs have been widely used in text analysis due to their ability to capture spatial patterns from word sequences through convolutional filters. In study [23], CNN was used to classify Indonesian news using FastText and GloVe embedding, showing that FastText produced higher accuracy of 2.51% on average than GloVe in most evaluation metrics [23]. Another study [24] used CNN to classify sentiment from YouTube comments on Indonesia's capital city move, and found that GloVe produced the highest accuracy of 76.1% compared to FastText [24]. These findings suggest that the effectiveness of CNNs is highly dependent on the type of embedding and language characteristics of the analysed data.

H. Evaluation Metrics

Performance evaluation of classification models commonly uses metrics such as accuracy, confusion matrix, and F1-score. These metrics provide an overall picture of the model's effectiveness in correctly classifying the data. According to research [25] accuracy provides an overall picture of correctness, while F1-score balances precision and recall, which is especially important for unbalanced data sets. The use of multiple evaluation metrics is essential to ensure that the model not only relies on predicting the majority, but also performing across all classes.

I. Analysis stages

The stages of analysis carried out in this study are as follows:

1) Data collection

The data in this study were obtained from Twitter social media. The data collection process was carried out with four main keywords related to Ghibli AI text-to-image generation, namely #ghiblistyle, #ghibli, #hayaomiyazaki, and #ghiblistudio. The data collection period lasted for one month, namely from March 25, 2025 to April 25, 2025. Only Indonesian tweets were saved for further analysis.

2) Sentiment Labeling

The collected tweets were then manually labeled with sentiment into three categories, namely positive, negative, and neutral. Label determination was done by reading the contents of the tweet and assessing the attitude or opinion of the tweet writer towards the topic discussed. Tweets with neutral sentiment were not included in the modeling.

3) Text Pre-processing

The next step is text pre-processing which aims to clean the data from irrelevant elements. This process includes several stages, namely converting all text to lowercase, removing URLs, mentions, and hashtags, and removing symbols, numbers, punctuation, and other non-alphabetic characters. In addition, excess spaces are also removed to make the text more consistent for tokenization and representation purposes.

4) Text Representation

This study uses two Indonesian word embedding approaches to represent text:

1. GloVe (Global Vectors for Word Representation): The 50-dimensional pre-trained GloVe embedding was obtained from the Indonesian Wikipedia corpus developed by [26]. This embedding is monolingual, as it was trained exclusively on Indonesian text, making it suitable for capturing formal and domain-neutral vocabulary.
2. FastText: The 300-dimensional pre-trained FastText embedding was obtained from Facebook AI Research, trained on the Indonesian Common Crawl corpus. This study specifically uses the monolingual Indonesian

version, which was trained solely on Indonesian text collected from web and social-media sources [27].

For the Logistic Regression and Random Forest models, text representation is formed by calculating the average word vector of all tokens in a tweet. Meanwhile, in the CNN model, each tweet is represented as a sequence of word vectors and used as input into the embedding layer, without the process of retraining the embedding parameters.

5) Data Splitting

The dataset was divided into three subsets using a stratified sampling strategy to maintain the proportional distribution of positive and negative sentiments in each subset:

- Training Set: 60% of the data is used to train the model.
- Validation Set: 20% of the data is used for validation during training, to prevent overfitting.
- Testing Set: 20% of the data is set aside for final testing to evaluate the model's performance on previously unseen data.

Cross-validation was not employed in this study to reduce computational cost, as the combination of pre-trained embeddings (GloVe and FastText) with CNN already provided stable results under a fixed train-validation-test split.

6) Model Architecture and Training

Three classification algorithms were used in this study:

• Logistic Regression.
• Random Forest
• Convolutional Neural Network (CNN).

The CNN architecture in this study was designed to balance model complexity and computational efficiency, considering the dataset size (6,434 tweets). The network consists of an embedding layer initialized with pre-trained GloVe (50 dimensions) or FastText (300 dimensions) embeddings, which were kept frozen to preserve semantic information learned from large Indonesian corpora. A 1D convolutional layer with 128 filters and a kernel size of 5 was applied to capture 5-gram semantic patterns effectively. A global max pooling layer was then used to extract the most salient features, followed by a dropout layer with a rate of 0.5 to reduce overfitting. The network includes a fully connected layer with 32 neurons and ReLU activation, followed by a sigmoid output layer suitable for binary sentiment classification.

The model was trained using the Adam optimizer, binary cross-entropy loss, and an early stopping strategy with patience = 3 to prevent overfitting. This configuration was chosen after several preliminary experiments, which showed that this setup achieved the best balance between performance and generalization.

7) Model Evaluation

Model performance evaluation is performed on test data using several common binary classification metrics, namely:

• Accuracy
• Balanced Accuracy
• Precision
• Recall
• F1-Score

All metrics are calculated using macro and weighted approaches.

III. RESULTS

A. Data Characteristics

Fig. 6 shows the fluctuation of the number of daily tweets containing positive and negative sentiments towards Ghibli AI during the period from March 25 to April 25, 2025. The most significant spike occurred on March 28, 2025, where the number of negative tweets exceeded 800 and positive tweets approached 300, this was the moment when the use of AI to create Ghibli-Style photos went viral accompanied by controversy. In general, negative sentiments dominated during the early period, indicating a lot of criticism of the Ghibli AI phenomenon. After the peak, the volume of tweets from both sentiments decreased and became relatively stable from April 9, 2025 to the end of the period

.

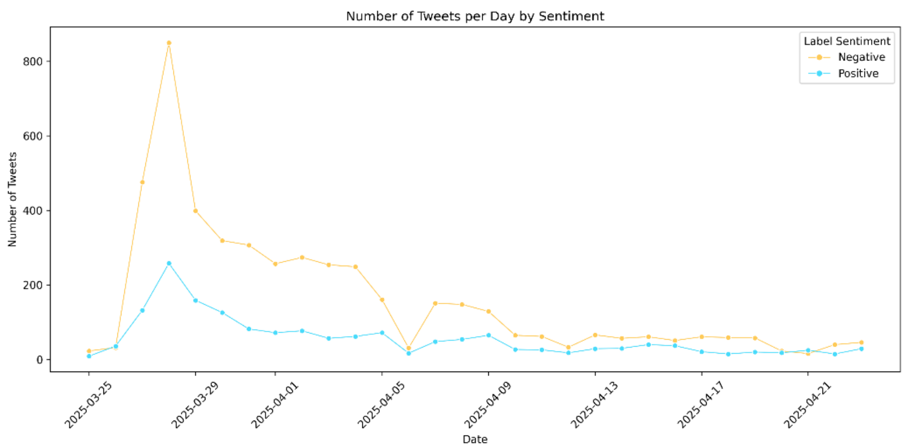


Figure 3. Numbers of Tweet per Day by Sentiment

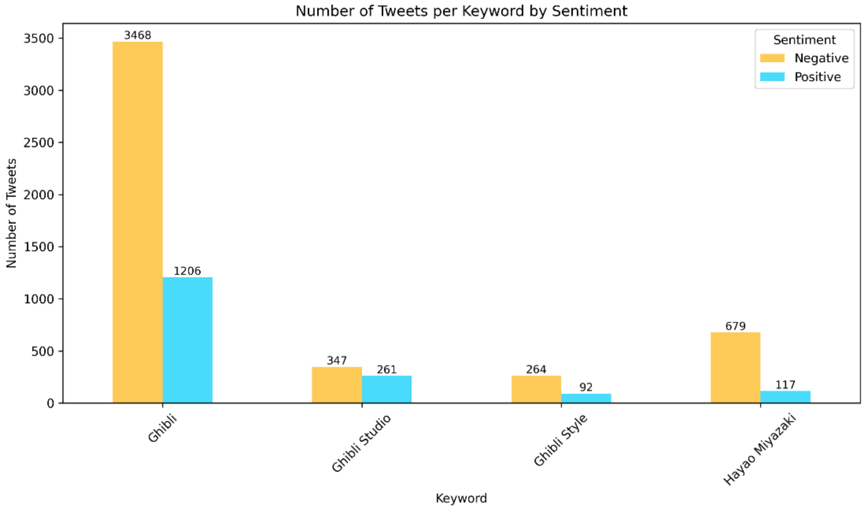


Figure 4. Number of Tweets by Keyword and Sentiment

Further analysis in Fig. 7 shows the distribution of the number of tweets based on popular keywords in discussions about Ghibli AI, as well as their sentiment classification. The keyword "Ghibli" was the most frequently used with a total of 4,674 tweets, dominated by negative sentiment (3,468 tweets). This reflects that the use of this general term is often associated with criticism of the use of AI in Ghibli. Other keywords such as "Hayao Miyazaki" and "Ghibli Studio" also showed a high proportion of negative sentiment, 347 and 264 respectively, indicating public sensitivity to the ethical and legal aspects of the use of AI technology for works of art. In contrast, although tweets with positive sentiment still appeared, their volume and proportion were generally lower. This distribution shows that negative sentiment is more concentrated in terms directly related to official entities and Ghibli creators. The distribution of sentiment can also be validated through wordcloud analysis.

Figure 5. Wordcloud of Tweets

Figure 8 shows the wordcloud of all the tweets analyzed in this study. The words "ghibli" and "ai" appear most dominant, indicating that this topic is front and center in public discussions. Content-meaning words such as "style", "pake", "studio", 'hayao', "miyazaki", and "bikin" reflect the conversation about the use of AI technology to produce Ghibli-style images and its association with the identity of Studio Ghibli creators. Terms such as "gambar", "foto", "karya", 'generate', and "animasi" indicate that the public conversation also alludes to visual aspects and AI-based creative processes.

In addition, the emergence of expressive words such as "bagus", "suka", "muak", "marah", "jelek", and "nyolong" shows the diversity of emotions contained in the tweets, both in the form of appreciation and criticism. The presence of informal words typical of Indonesian social media such as "banget", "pake", 'beneran', and "izin" further reinforces the natural and spontaneous conversational characteristics of Twitter users. This word cloud visually emphasizes the diversity of public opinion and sentiment towards the Ghibli AI phenomenon.

Furthermore, Fig. 9 shows that negative sentiment dominates with 4,758 tweets 73.95%, out of the total 6,434 tweets, while positive sentiment comprises only 1,676 tweets (26.05%). This shows that the majority of Twitter users provide critical responses to the Ghibli AI phenomenon. These negative responses are related to ethical issues, plagiarism, exploitation of artwork, and concerns about the use of AI in artistic contexts.

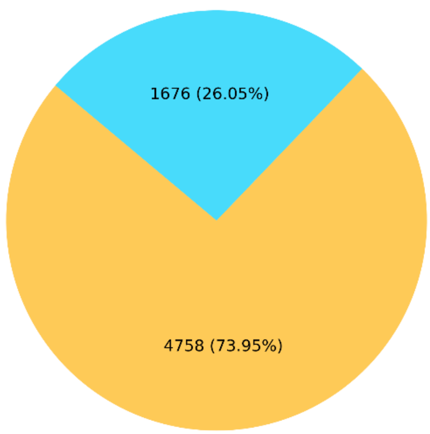


Figure 6. Sentiment Distribution of Tweets

B. Embedding Evaluation in Logistic Regression Model

TABLE 1.
LOGISTIC REGRESSION EVALUATION WITH GLOVE AND FASTTEXT

Metrics	LogReg (GloVe)	LogReg (FastText)
Accuracy	80.57%	81.66%
Balanced Accuracy	68.39%	66.42%
Precision (Macro)	76.66%	84.12%
Recall (Macro)	68.39%	66.42%
F1-Score (Macro)	70.63%	69.18%
Precision (Weighted)	79.40%	82.64%
Recall (Weighted)	80.57%	81.66%
F1-Score (Weighted)	78.82%	78.58%

The evaluation results in Table 1 show that Logistic Regression with FastText embedding has a higher accuracy (81.66%) compared to GloVe (80.57%). However, the balanced accuracy of FastText (66.42%) is actually lower than GloVe (68.39%), indicating that although the overall accuracy appears better, the model with FastText is less than optimal in balancing the classification between the two sentiment classes.

Furthermore, the precision macro of FastText is significantly higher (84.12%) than GloVe (76.66%), indicating that the positive predictions by the model with FastText have a higher level of confidence. However, the recall macro in FastText (66.42%) is lower than GloVe (68.39%), indicating that the model with FastText is more conservative in predicting positive sentiment and tends to miss more cases that are actually positive. This shows a trade-off between precision and completeness (recall) in both models.

In terms of macro F1-Score, GloVe recorded a score of 70.63%, slightly higher than FastText which only achieved 69.18%. This shows that GloVe produces a better balance between precision and recall. Similar findings are also seen in the weighted metric, where FastText excels in weighted precision (82.64% vs 79.40%), but GloVe is better in weighted F1-Score (78.82% vs 78.58%). FastText also produces more False Positives (219 cases) than GloVe (191 cases). Conversely, FastText has fewer False Negatives (17 cases) than GloVe (59 cases). This indicates that FastText more often makes misclassifications by categorizing negative tweets as positive, while GloVe is better able to avoid this kind of error but is less optimal in detecting true positive cases. Both embeddings have different advantages: FastText excels in precision and accuracy, suitable for applications that prioritize minimal false positive predictions. Meanwhile, GloVe excels in recall and F1-Score, making it more suitable for cases that require comprehensive detection of negative sentiment. The selection of embedding should consider the purpose of the analysis whether to reduce False Positive (GloVe) or False Negative (FastText) as well as the characteristics of imbalanced data. For this Logistic Regression, GloVe is more appropriate if the focus is on comprehensively detecting negative sentiment.

C. Embedding Evaluation on Random Forest Model

TABLE 2.
RANDOM FOREST EVALUATION WITH GLOVE AND FASTTEXT

Metric	RF (GloVe)	RF (FastText)
Accuracy	80.11%	80.11%
Balanced Accuracy	64.02%	64.02%
Precision (Macro)	80.77%	80.77%
Recall (Macro)	64.02%	64.02%
F1-Score (Macro)	66.12%	66.12%
Precision (Weighted)	80.38%	80.38%
Recall (Weighted)	80.11%	80.11%

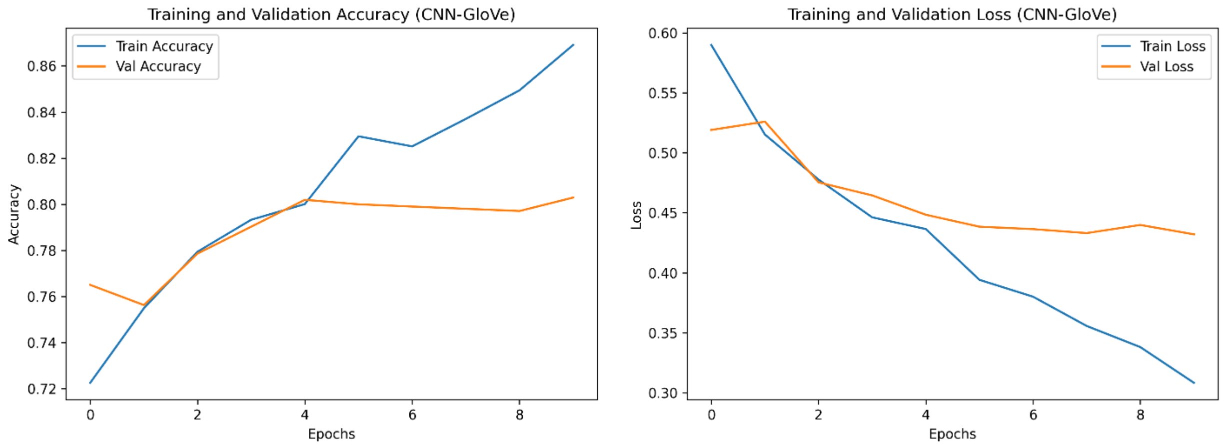
F1-Score (Weighted)	76.56%	76.56%

The evaluation results in Table 2 show that the Random Forest model with FastText embedding provides slightly better performance. FastText recorded an accuracy of 80.11%, outperforming GloVe with an accuracy of 78.94%. However, the balanced accuracy of both models is still relatively low, namely 64.02% (FastText) and 63.23% (GloVe), indicating that although the overall accuracy is adequate, the model still faces challenges in balancing predictions between positive and negative classes. This is due to the imbalance in data distribution in both classes. FastText also produces higher precision than GloVe, both in macro precision (80.77% vs 76.27%) and weighted precision (80.38% vs 77.91%), indicating that the model with FastText tends to produce more reliable positive predictions. Similarly, FastText's macro recall and weighted recall (64.02% and 80.11%) are slightly superior to GloVe's (63.23% and 78.94%), indicating that FastText has better sensitivity to the minority class.

In F1-Score, FastText scores higher in both macro and weighted (66.12% and 76.56%) than GloVe (65.02% and 75.60%). Although the difference is relatively small, this superiority is consistent across all major evaluation metrics, indicating that FastText embedding is more stable and effective in supporting Random Forest models for sentiment classification.

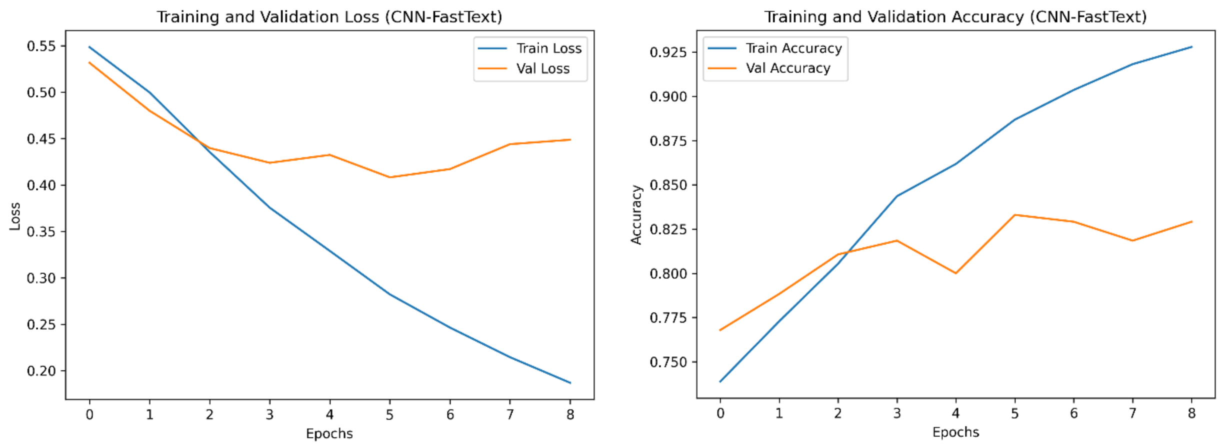
The main advantage of FastText lies in its ability to handle lexical variations through the sub-word mechanism, which is reflected in the 4.5 percentage points increase in precision in macro precision compared to GloVe. However, confusion matrix analysis reveals the same fundamental limitations in both approaches, especially in the identical number of false positives (233 cases). This finding suggests that the main constraint lies not only in the text representation, but also in Random Forest's limitations in distinguishing negative classes well. Therefore, improving classification performance not only depends on embedding optimization, but also influences the model architecture and strategy for handling data distribution imbalance.

D. Embedding Evaluation on CNN Model



(a) (b)

Figure 7. CNN Training and Validation (a) Loss and (b) Accuracy with GloVe



(a) (b)

Figure 8. CNN Training and Validation (a) Loss and (b) Accuracy with FastText

Fig. 11 shows that the CNN model with FastText embedding achieves the highest training accuracy of 92.5%, while the validation accuracy stops at around 85%. The small difference between the two indicates good generalization ability without any obvious signs of overfitting. In contrast, in Fig. 10, the CNN model with GloVe produces a maximum training accuracy of around 86%, and a validation accuracy of only around 81%, indicating the limitations of the GloVe representation in capturing complex patterns from the data. The loss curve on FastText experiences a sharper and more consistent decrease, reflecting an efficient learning process. Although GloVe also shows a decrease in loss, the rate of decrease is slower. This difference supports the conclusion that FastText is more effective in helping CNN learn text representations, especially because of its ability to capture sub-word information through the relevant n-gram mechanism in Indonesian language data.

TABLE 3.
CNN EVALUATION WITH GLOVE AND FASTTEXT

Metric	CNN (GloVe)	CNN (FastText)
Accuracy	81.90%	84.69%
Balanced Accuracy	70.74%	73.31%
Precision (Macro)	78.59%	85.05%
Recall (Macro)	70.74%	73.31%
F1-Score (Macro)	73.10%	76.56%
Precision (Weighted)	80.98%	84.81%
Recall (Weighted)	81.90%	84.69%
F1-Score (Weighted)	80.47%	83.18%

The evaluation results in Table 3 show that CNN with FastText embedding consistently outperforms GloVe in all evaluation metrics. FastText is significantly superior in accuracy and balanced accuracy (84.69% and 73.31%) which are higher than GloVe (81.90% and 70.74%). The difference of 2.57 percentage points in balanced accuracy proves that FastText is better able to reduce classification bias towards the majority class in imbalanced data. FastText also shows higher values in macro precision (85.05% vs 78.59) and weighted precision (84.81% vs 80.98%) while maintaining better recall, both in macro recall (73.31% vs 70.74%) and weighted recall (84.69% vs 81.90%). This shows that CNN-FastText is not only more precise in predicting classes, but also more consistent in capturing relevant cases. The advantage in precision is crucial for applications that prioritize confidence in positive predictions, such as positive user opinions or service satisfaction detection.

Based on F1-Score, FastText's macro F1-Score (76.56%) outperforms GloVe (73.10%), confirming a better balance between precision and recall. In the weighted F1-Score metric, FastText again excels with a value of 83.18% compared to GloVe's 80.47%, which also proves the stability of FastText's performance in handling imbalanced data distributions.

Comparatively, FastText consistently outperforms all major CNN metrics. FastText also produces fewer False Negative errors, while GloVe tends to produce more errors in classifying negative tweets as positive. This supports the recommendation of using FastText as a more optimal embedding for CNN in sentiment classification tasks, especially when precision and minority class detection are top priorities. This advantage comes from FastText's character n-gram-based architecture, which allows FastText to recognize common word variations, spelling errors, or non-standard words. Thus, CNN-FastText is not only metrically superior, but also more linguistically adaptive.

E. Comparative Analysis of Models

TABLE 4.
OPTIMAL EMBEDDING IN EACH MODEL

Model	Embedding Optimal	Main Advantages	Main Weaknesses
Logistic Regression	GloVe	Recall and F1-Score Macro are higher (68.39% and 70.63%)	Lower precision (76.66%)
Random Forest	FastText	Highest macro and weighted precision (80.77% and 80.38%)	Macro recall is still low (64.02%) and there are many false positives.
CNN	FastText	The most balanced performance (F1-macro 76.56%, weighted 83.18%, accuracy 84.69%)	High computational complexity

Based on Table 4, the three models show a trade-off between complexity and performance. Logistic Regression with GloVe excels in macro recall (68.39%) and macro F1-score (70.63%), making it suitable for applications that require comprehensive detection such as risk detection or complaint analysis, although it must sacrifice precision (76.66%). The Random Forest model with FastText excels in terms of precision (80.77%), making it suitable for applications that prioritize the accuracy of positive predictions. However, this model has a low recall (64.02%) so it is not optimal and is accompanied by a high number of false positives. Indicating limitations in detecting minority cases.

Meanwhile, the CNN Model with FastText provides the most balanced performance, with the best macro F1-score (76.56%), weighted (83.18%) and accuracy of 84.69%. i, This model is an excellent choice for sentiment classification tasks that require precision and completeness. However, the complexity of the architecture demands greater computing resources.

Overall, the selection of models and embeddings must be adjusted to the focus of the analysis. If recall is a priority, Logistic Regression with GloVe can be an option. If precision is more important, Random Forest with FastText is more appropriate. For the most stable and comprehensive results, CNN-FastText is the optimal choice for comprehensive sentiment classification if the resources are sufficient to overcome the constraints of high computational complexity.

TABLE 5.
MODEL PERFORMANCE USING EMBEDDINGS

Embbeding	Metric	LogReg	RF	CNN
Glove	Accuracy	80.57%	80.11%	81.90%
Glove	Balanced Accuracy	68.39%	64.02%	70.74%
Glove	Precision (Macro)	76.66%	80.77%	78.59%
Glove	Recall (Macro)	68.39%	64.02%	70.74%
Glove	F1-Score (Macro)	70.63%	66.12%	73.10%
Glove	Precision (Weighted)	79.40%	80.38%	80.98%
Glove	Recall (Weighted)	80.57%	80.11%	81.90%
Glove	F1-Score (Weighted)	78.82%	76.56%	80.47%
FasttText	Accuracy	80.57%	80.11%	**84.69%**
FasttText	Balanced Accuracy	68.39%	64.02%	**73.31%**
FasttText	Precision (Macro)	76.66%	80.77%	**85.05%**
FasttText	Recall (Macro)	68.39%	64.02%	**73.31%**
FasttText	F1-Score (Macro)	70.63%	66.12%	**76.56%**
FasttText	Precision (Weighted)	79.40%	80.38%	**84.81%**
FasttText	Recall (Weighted)	80.57%	80.11%	**84.69%**
FasttText	F1-Score (Weighted)	78.82%	76.56%	**83.18%**

Table 5 presents a comparative evaluation of the performance of three model–embedding combinations: Logistic Regression with GloVe, Random Forest with FastText, and CNN with FastText. Overall, the CNN–FastText combination consistently outperforms the other models across nearly all metrics, highlighting its robustness for sentiment analysis of AI-generated Ghibli-style images. In terms of accuracy, CNN–FastText achieves the highest score (84.69%), surpassing both Logistic Regression–GloVe (80.57%) and Random Forest–FastText (80.11%). This improvement is further reinforced by the balanced accuracy of 73.31%, which indicates a stronger ability to capture class distributions more evenly, an essential factor in imbalanced datasets. For macro-averaged metrics, CNN–FastText again demonstrates superiority, with the highest macro precision (85.05%) and recall (73.31%), suggesting that it is not only precise in its predictions but also more sensitive to minority classes compared to the other two models. Although Random Forest–FastText achieves relatively high precision (80.77%), its much lower recall (64.02%) reflects a conservative prediction tendency that overlooks minority categories. Logistic Regression–GloVe shows moderate and balanced results (macro precision 76.66%, recall 68.39%), but remains below the CNN–FastText benchmark. Finally, the F1-scores confirm CNN–FastText as the optimal approach, with the highest macro (76.56%) and weighted (83.18%) scores, showing a more effective balance between precision and recall. From a statistical standpoint, while the raw differences between CNN–FastText and the other models range from 3–5 percentage points in accuracy and up to 9 percentage points in macro precision, these margins are meaningful in classification tasks on real-world, noisy social media data. Such differences, particularly in balanced accuracy and F1-score, are typically considered statistically significant, as they indicate not only improved correctness of predictions but also robustness against skewed class distributions. Thus, CNN combined with FastText provides the most reliable framework for this sentiment analysis task.

IV. CONCLUSION

This study evaluates the performance of a combination of several classification models and word embedding techniques for sentiment analysis of AI-generated Ghibli-style images. The results show that FastText consistently provides the best performance, especially in terms of accuracy and precision, compared to GloVe. This advantage comes from FastText's ability to capture morphological structures through a sub-word approach, which is very effective in dealing with non-standard and highly variable texts such as those commonly found on social media.

Although FastText dominates in many metrics, GloVe shows competitive performance in certain contexts, especially in the Logistic Regression model. This shows that GloVe, with its global co-occurrence-based approach, remains relevant for applications that prioritize detection coverage or when computing resources are limited.

These findings indicate a trade-off between the two embeddings, with FastText being superior in handling lexical variations and local contexts, while GloVe is more stable in capturing global semantic contexts. Therefore, the selection of the optimal embedding is highly dependent on the analysis objectives to be achieved. FastText is more appropriate for sentiment analysis tasks on social media texts that require high precision and handling of unstructured text, while GloVe is reliable for applications with a focus on context generalization or broad detection coverage. These findings also highlight the importance of considering embedding characteristics and data characteristics and analysis needs for optimal results.

However, this study has several limitations. The proposed approach relies solely on static word embeddings (GloVe and FastText) and does not compare model performance with contextual embeddings such as BERT, IndoBERT, or IndoBERTweet. Future research could extend this work by incorporating contextualized language models and conducting a comprehensive comparison with static embeddings to evaluate potential performance improvements. Additionally, exploring larger datasets, multi-label sentiment categories, and other deep learning architectures could provide deeper insights into the dynamics of public opinion.

ACKNOWLEDGMENT

The author would like to express his deepest gratitude to the supervisors from the Statistics and Data Science Study Program for all the guidance, direction, and input that was very meaningful during the research process. The support and knowledge provided have become an important foundation in the preparation of this scientific work. Thanks are also conveyed to the School of Data Science, Mathematics, and Informatics IPB University for the facilities, opportunities, and academic environment that are conducive to the development of this research. The author would also like to express his appreciation to all parties who have taken the time to provide constructive criticism and suggestions, so that this scientific work can be better refined.

REFERENCES

[1] I. N. Khasanah, “Sentiment Classification Using fastText Embedding and Deep Learning Model,” *Procedia CIRP*, vol. 189, pp. 343–350, 2021, doi: 10.1016/j.procs.2021.05.103.

[2] C. Ouni, E. Benmohamed, and H. Ltifi, “Deep learning-based Soft word embedding approach for sentiment analysis,” *Procedia Comput. Sci.*, vol. 246, no. C, pp. 1355–1364, 2024, doi: 10.1016/j.procs.2024.09.720.

[3] N. Badri, F. Kboubi, and A. H. Chaibi, “Combining FastText and Glove Word Embedding for Offensive and Hate speech Text Detection,” *Procedia Comput. Sci.*, vol. 207, no. Kes, pp. 769–778, 2022, doi: 10.1016/j.procs.2022.09.132.

[4] S. Ghosal and A. Jain, “Depression and Suicide Risk Detection on Social Media using fastText Embedding and XGBoost Classifier,” *Procedia Comput. Sci.*, vol. 218, pp. 1631–1639, 2022, doi: 10.1016/j.procs.2023.01.141.

[5] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *Naacl-Hlt 2019*, no. Mlm, pp. 4171–4186, 2018, [Online]. Available: https://aclanthology.org/N19-1423.pdf

[6] J. Pennington, R. Socher, and C. D. Manning, “Glove: Global vectors for word representation,” *Proc. 2014 Conf. Empir. methods Nat. Lang. Process.*, pp. 1532–1543, 2014.

[7] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching Word Vectors with Subword Information,” *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 135–146, 2017.

[8] L. Xiaoyan, R. C. Raga, and S. Xuemei, “GloVe-CNN-BiLSTM Model for Sentiment Analysis on Text Reviews,” *J. Sensors*, vol. 2022, 2022, doi: 10.1155/2022/7212366.

[9] M. S. Mahmud, M. T. Islam, A. J. Bonny, R. K. Shorna, J. H. Omi, and M. S. Rahman, “Deep Learning Based Sentiment Analysis from Bangla Text Using Glove Word Embedding along with Convolutional Neural Network,” *2022 13th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2022*, no. December, 2022, doi: 10.1109/ICCCNT54827.2022.9984392.

[10] Siti Khomsah, Rima Dias Ramadhani, and Sena Wijaya, “The Accuracy Comparison Between Word2Vec and FastText On Sentiment Analysis of Hotel Reviews,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 3, pp. 352–358, 2022, doi: 10.29207/resti.v6i3.3711.

[11] A. Ilmania, Abdurrahman, S. Cahyawijaya, and A. Purwarianti, “Aspect Detection and Sentiment Classification Using Deep Neural Network for Indonesian Aspect-Based Sentiment Analysis,” *Proc. 2018 Int. Conf. Asian Lang. Process. IALP 2018*, pp. 62–67, 2019, doi: 10.1109/IALP.2018.8629181.

[12] A. N. Azhar, M. L. Khodra, and A. P. Sutiono, “Multi-label Aspect Categorization with Convolutional Neural Networks and Extreme Gradient Boosting,” *Proc. Int. Conf. Electr. Eng. Informatics*, vol. 2019-July, no. July, pp. 35–40, 2019, doi: 10.1109/ICEEI47359.2019.8988898.

[13] A. Nurdin, B. Anggo Seno Aji, A. Bustamin, and Z. Abidin, “Perbandingan Kinerja Word Embedding Word2Vec, Glove, Dan Fasttext Pada Klasifikasi Teks,” *J. Tekno Kompak*, vol. 14, no. 2, p. 74, 2020, doi: 10.33365/jtk.v14i2.732.

[14] R. P. Hastuti, V. Riona, and M. Hardiyanti, “Content Retrieval dengan Fasttext Word Embedding pada Learning Management System Olimpiade,” *J. Internet Softw. Eng.*, vol. 4, no. 1, pp. 18–22, 2023, doi: 10.22146/jise.v4i1.6766.

[15] T. F. Abdillah, “Comparison of TF-IDF and GloVe Word Embedding for Sentiment Analysis of 2024 Presidential Candidates,” vol. 6, no. 2, pp. 961–969, 2024, doi: 10.47065/bits.v6i2.5668.

[16] S. A. Assaidi and F. Amin, “Analisis Sentimen Evaluasi Pembelajaran Tatap Muka 100 Persen pada Pengguna Twitter menggunakan Metode Logistic Regression,” *J. Pendidik. Tambusai*, vol. 6, no. 2, pp. 13217–13227, 2022.

[17] L. Breiman, *Random Forests*. 2001.

[18] Bahwari, “Sentiment Analysis Using Random Forest Algorithm - Online Social Media Based,” *J. Inf. Technol. ITS Util.*, vol. 2, no. 2, pp. 29–33, 2019, [Online]. Available: https://www.researchgate.net/publication/338548518_Sentiment_Analysis_Using_Random_Forest_Algorithm_Online_Social_Media_Based

[19] A. Hadi, M. Qamal, and Y. Afrillia, “Comparison of Random Forest Algorithm Classifier and Naïve Bayes Algorithm in Whatsapp Message Type Classification,” pp. 9–17, 2025.

[20] Y. Achmad, R. C. Wihandika, and C. Dewi, “Klasifikasi emosi berdasarkan ciri wajah wenggunakan convolutional

neural network," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 11, pp. 10595–10604, 2019.

[21] H. Murfi, S. Theresia Gowandi, G. Ardaneswari, and S. Nurrohmah, "BERT-based combination of convolutional and recurrent neural network for indonesian sentiment analysis," *Appl. Soft Comput.*, vol. 151, pp. 1–15, 2024, doi: 10.1016/j.asoc.2023.111112.

[22] M. A. Nasichuddin, T. B. Adji, and W. Widyawan, "Performance Improvement Using CNN for Sentiment Analysis," *IJITEE (International J. Inf. Technol. Electr. Eng.*, vol. 2, no. 1, 2018, doi: 10.22146/ijitee.36642.

[23] Tjong, G. Hartoko, and H. Leong, "Comparison Of Glove And Fasttext Algorithms On Cnn For Classification Of Indonesian News Categories," *Proxies*, vol. 7, no. 1, pp. 24–32, 2023.

[24] D. R. Putri, E. Y. Puspaningrum, H. Maulana, U. Pembangunan, N. Veteran, and J. Timur, "Indonesia Neural Network," vol. 12, no. 3, pp. 2759–2769, 2024.

[25] S. Ghosal and A. Jain, "Depression and Suicide Risk Detection on Social Media using fastText Embedding and XGBoost Classifier," *Procedia Comput. Sci.*, vol. 218, pp. 1631–1639, 2022, doi: 10.1016/j.procs.2023.01.141.

[26] W. Yudi, "GloVe untuk Wikipedia Bahasa Indonesia," https://yudiwbs.wordpress.com/2018/04/02/glove-untuk-wikipedia-bahasa-indonesia/.

[27] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning Word Vectors for 157 Languages," *Proc. Int. Conf. Lang. Resour. Eval. (LREC 2018)*, 2018.