Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics e-ISSN: 2656-8624
Homepage: https://ijeeemi.org/; Vol. 7, No. 2, pp. 344-357, April 2025

RESEARCH PAPER

OPEN ACCES

Analysis of the Effect of Feature Extraction on Sentiment Analysis using BiLSTM: Monkeypox Case Study on X/Twitter

Noryasminda, Triando Hamonangan Saragih, Rudy Herteno, Mohammad Reza Faisal, and Andi Farmadi
Department of Computer Science, Faculty of Mathematics and Natural Science, Lambung Mangkurat University, Banjarbaru, Indonesia

ABSTRACT
The monkeypox outbreak has again become a global concern due to its widespread spread in various countries. Information related to the disease is widely shared through social media, especially Twitter which is a major source of public opinion. However, the complexity of language and the diverse viewpoints of users often pose challenges in accurately analyzing sentiment. Therefore, sentiment analysis of tweets about monkeypox is important to understand public perception and its impact on the dissemination of health information. This research contributes to identifying the most effective word embedding-based feature extraction method for sentiment analysis of health issues on social media. The purpose of this study is to compare the performance of word embedding methods namely Word2Vec, GloVe, and FastText in sentiment analysis of tweets about monkeypox using the BiLSTM model. Data totaling 1511 tweets were collected through a crawling process using the Twitter. After the data is collected, manual labeling is done into three sentiment categories, namely positive, negative, and neutral. Furthermore, the data is processed through a preprocessing stage which includes data cleaning, case folding, tokenization, stopword removal, and stemming. The evaluation results show that FastText with BiLSTM produces the highest accuracy of 89.8%, followed by Word2Vec at 88.6%, and GloVe at 87.4%. FastText proved to be more effective in reducing classification errors, especially in distinguishing between negative and positive sentiments due to its ability to capture subword information and broader context. These findings suggest that the use of FastText can improve the accuracy of sentiment analysis, especially on health issues that develop on social media, so that it can support data-driven decision making by relevant parties in handling information dissemination.

PAPER HISTORY
Received April 02, 2024
Revised May 31, 2024
Accepted May 31, 2024

KEYWORDS
Monkeypox;
Sentiment Analysis;
Twitter;
Word Embedding;
BiLSTM

CONTACT:
noryasminda@gmail.com
triando.saragih@ulm.ac.id
rudy.herteno@ulm.ac.id
reza.faisal@ulm.ac.id
andifarmadi@ulm.ac.id

1. INTRODUCTION

The monkeypox outbreak has again become a global concern due to its widespread spread in various countries. The World Health Organization (WHO) has designated mpox as a global Public Health Emergency of International Concern (PHEIC). The disease mostly occurs in the Central and West African regions in wild rodents [1]. From January 1 to June 22, 2022, a total of 3,413 laboratory-confirmed cases of monkeypox and one death were reported to WHO from 50 countries or territories across five WHO Regions [2]. The disease is caused by a virus of the genus Orthopoxvirus in the family Poxviridae. The virus can be transmitted from animals to humans, and in some cases can also be transmitted between humans [3]. Transmission between humans can occur through direct contact with an infected person, respiratory droplets, contaminated materials, as well as through sexual contact [4]. The re-emergence of this outbreak sparked widespread attention, both from the medical community and the general public who actively discussed this topic through social media, particularly on X/Twitter.

Twitter is a popular microblogging platform where users can share updates, express themselves or express opinions through tweets [5]. This activity makes Twitter an important data source in analyzing public responses to global health issues. Understanding public sentiment related to disease outbreaks such as monkey pox is crucial for governments and policymakers in designing effective mitigation strategies to control the spread of the disease [6]. Previous research has shown that Twitter data is relevant for analyzing public responses to monkeypox outbreaks. For example, Ng et al. (2022) analyzed 352.182 English tweets and found that public discussions about mpox included concerns for safety, stigma against minority communities, and distrust of public institutions [7]. These findings indicate the importance of a deeper understanding of public perceptions through a systematic approach. Therefore, the application of computational methods such as

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

sentiment analysis is relevant and crucial to assess public opinion regarding the spread of this outbreak.

Sentiment analysis is a branch of Natural Language Processing (NLP) that aims to understand and classify opinions or emotions in a text [8]. However, text in its raw form cannot be directly processed by the model. Therefore, a word embedding technique is required to convert the text into a numerical vector that can be understood by the model [9]. Some popular methods in word embedding include Word2Vec, GloVe, and FastText. Word2Vec builds word representations based on their context in the text using Continuous Bag of Words (CBOW) and Skip-Gram approaches [10]. GloVe learns word representations by factorizing the co-occurrence matrix, making it effective in capturing linear relationships between words [11]. FastText is an extension of Word2Vec that not only represents the whole word but also considers sub-words (n-grams), making it more effective in handling words that occur infrequently or have spelling variations [12].

Once the text is represented in vector form through word embedding, the data can be used as input in deep learning models for sentiment analysis. One method that is often used is Bidirectional Long Short-Term Memory (BiLSTM). BiLSTM is able to capture context from both directions in a text sequence, so as to understand the meaning of the text more thoroughly [13]. The study by Li et al. (2021) showed that the BiLSTM model with an attention mechanism achieved up to 96% accuracy in sentiment analysis of Amazon product reviews [14]. Another study by Rahman et al. (2024) introduced a RoBERTa-BiLSTM hybrid model that achieved 92.36% accuracy on the IMDb dataset, outperforming other baseline models [15]. In addition, a study by Glenn et al. (2023) showed that BiLSTM trained with FastText embedding achieved an average accuracy of 70.83% in emotion classification on Indonesian tweets, surpassing traditional models such as logistic regression and random forest [16].

Although the context of the topics studied in these studies is different, the results show that the choice of word embedding method and the use of BiLSTM models significantly affect the performance of sentiment analysis or emotion classification. However, to date, no study has specifically compared the three word embedding methods Word2Vec, FastText, and GloVe with BiLSTM in the context of sentiment analysis related to monkeypox. Therefore, this study aims to evaluate the effect of various Word2Vec, FastText, and GloVe feature extraction methods on the performance of the BiLSTM model in sentiment analysis related to monkeypox on Twitter and determine the method that provides the highest accuracy. The findings of this research are expected to contribute as follows: 1) Provide in-depth information related to public perceptions of monkeypox, so that it can be used as a reference in formulating strategic policies to deal with the spread of infectious diseases, 2) Become a reference for further studies in the field of sentiment analysis and Natural Language Processing (NLP), especially those using the BiLSTM method with various feature extraction techniques such as Word2Vec, FastText, and GloVe, 3) Provide insight into the effectiveness and accuracy of various word embedding methods in improving the performance of deep learning models for text analysis, 4) Provide a dataset of tweets related to monkeypox that have gone through manual labeling and preprocessing processes as a data source that can be used by other researchers in the development of sentiment analysis models and studies in the field of digital health.

This study is structured as follows: section II presents the materials and methods, including dataset collection, data labeling, preprocessing, word embedding techniques, and the BiLSTM model used for sentiment analysis. Section III reports the evaluation results of BiLSTM combined with Word2Vec, GloVe, and FastText along with performance metrics such as accuracy, precision, recall, and F1-score. Section IV discusses the comparative performance, interpretation of results, limitations of the study, and comparison with previous research. Section V concludes the paper by summarizing the objectives, key findings, limitations, and future research directions.

2. MATERIALS AND METHOD

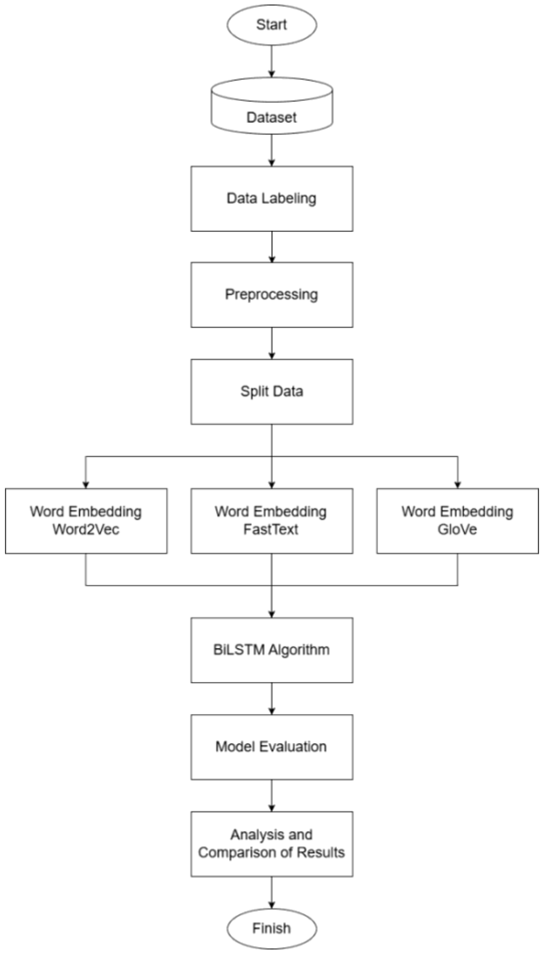

Fig. 1. Flowchart of monkeypox sentiment analysis research with word embedding method comparison using BiLSTM.

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

The entire study process was conducted using a laptop with Intel(R) Core(TM) i5-10300H processor specifications and 8 GB RAM. To support the computational needs in training deep learning models, the Google Colaboratory platform was used, which provides a cloud-based Python environment with GPU support for free. This platform was chosen because of its flexibility and efficiency in running deep learning experiments such as BiLSTM on medium-scale datasets. The programming language used is Python with the help of various libraries such as scikit-learn, matplotlib, seaborn, fasttext-wheel, gensim, and tensorflow. By utilizing these resources, this study compares three word embedding-based feature extraction methods namely Word2Vec, FastText, and GloVe using BiLSTM. The research stages include collecting monkeypox datasets through crawling from Twitter, manually labeling data, preprocessing, splitting data, model evaluation, and comparative analysis of results. Fig. 1 illustrates the research flow diagram used in this study.

A. Dataset

The dataset used in this study is monkeypox related tweets obtained through a web crawling process using tweet-harvest, a Node.js-based tool that allows data extraction from Twitter search pages without using the official API. The crawling process was performed by filtering Indonesian tweets using the keywords “monkey pox OR mpox lang:id”, with the search mode set on the “LATEST” tab to obtain the latest tweets chronologically. The number of tweets targeted in data retrieval is around 1500, and the crawling results are saved in CSV format using the -o parameter “mpox.csv” for further processing. The authentication process is done using Twitter guest token, which is temporary but sufficient to access public data through tweet-harvest. The final dataset obtained consists of 1511 tweets and includes 15 attributes: conversation_id_str, created_at, favorite_count, tweet, id_str, image_url, in_reply_to_screen_name, lang, location, quote_count, reply_count, retweet_count, tweet_url, user_id_str, and username. Information about the data retrieval parameters is shown in Table 1 below:

Table 1. Data collection parameters of tweets about monkeypox on twitter platform.

Parameters	Value
Keywords	cacar monyet OR mpox lang:id
Target number of tweets	1500
Search tab	LATEST
Tool	tweet-harvest versi 2.6.1
Token	Twitter Guest Token

B. Data Labeling

After the data collection process is complete, the next step is to manually label the tweets based on the sentiment contained in them. The data labeling process is divided into three main categories: positive, negative, and neutral. These three categories are common classifications used in sentiment analysis, as applied in various previous studies such as in the research of Afuan and Hidayat (2024), Villavicencio et al. (2021), and Acosta et al. (2021) [17][18][19]. Positive labels were assigned to tweets that contained sentiments of support, encouragement, or positive narratives towards prevention, treatment, or optimistic attitudes. Conversely, a negative category is given if the tweet contains expressions of fear, anxiety, complaints, or words with negative connotations. The neutral category is given to tweets that only convey factual information or news without clear emotional expression [20]. These criteria were developed by referring to emotion and opinion-based sentiment classification approaches in social media that have been widely used in previous literature.

The labeling process was done by the researchers themselves manually. To maintain consistency and reduce subjective bias, researchers first compiled labeling guidelines in the form of definitions and examples for each sentiment category. This guide was used as a reference during the labeling process. In addition, researchers also rechecked a random portion of the data after the initial process was complete, to ensure the consistency of classification between labels. Examples of tweet data labeling results can be seen in Table 2.

Table 2. The study dataset that has been labeled on each tweet data.

No	conversation_id_str	created_at	favorite_count	tweet	...	label
1	1.84E+18	Mon Sep 16 09:18:07 +0000 2024	0	Warga Diimbau Kenali Gejala Cacar Monyet & Penularannya https://t.co/8iE7K8hJHd	...	positive
2	1.84E+18	Mon Sep 16 03:48:37 +0000 2024	0	Menkes: Perlindungan dari vaksin cacar efektif dan tidak perlu khawatir soal Mpox. #Cegah Mpox #TakPerluKhawatirMpox https://t.co/xBwhGij5wh	...	positive
…	…	…	…	…	…	…

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

No	conversation_id_str	created_at	favorite_count	tweet	...	label
1510	1.83E+18	Thu Aug 22 07:22:58 +0000 2024	0	@lgabyt Capekk banget woyhy Agustus nih Belum lagi cacar monyet https://t.co/W01MIQhriX	...	negative
1511	1.83E+18	Thu Aug 22 07:16:26 +0000 2024	0	MPOX atau cacar monyet merupakan penyakit yang masih menjadi perhatian di Indonesia terutama setelah merebaknya wabah global pada tahun 2022. https://t.co/GcDjINpjfu	...	neutral

The distribution of the number of tweets per sentiment category is shown in Fig. 2.

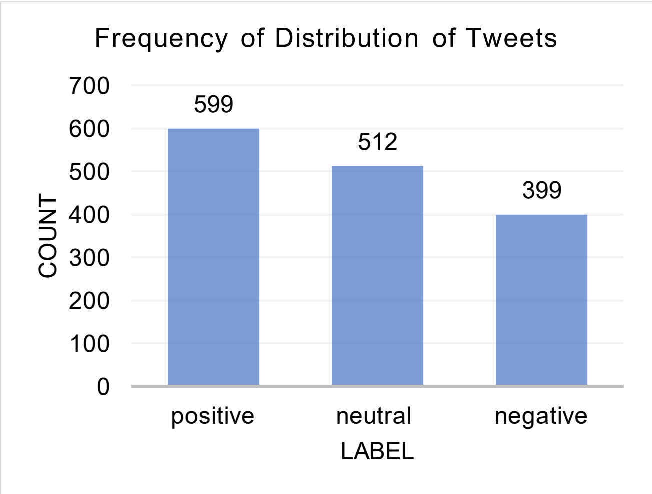

Fig 2. Distribution of tweet data based on positive, negative, and neutral sentiment labels.

C. Preprocessing

Preprocessing is an important stage in sentiment analysis because it has a major effect on the accuracy and performance of the resulting model [17]. In addition, preprocessing also serves as the first step to normalize the data by removing irrelevant words in the sentiment analysis process [21]. Preprocessing stages in each study may vary depending on the characteristics of the data and the purpose of the analysis. The preprocessing stages carried out in this study include:

1) Data Cleaning

The data cleaning stage aims to remove irrelevant elements in the dataset, thus improving the quality of the data before it is used in sentiment analysis [22]. In this study, the data cleaning process includes several important steps, namely the removal of unnecessary columns or attributes such as conversation_id_str, created_at, favorite_count, id_str, image_url, in_reply_to_screen_name, lang, location, quote_count, reply_count, retweet_count, tweet_url, user_id_str, and username. At this stage, only the attributes "tweet" and "label" are retained. In addition, data that has empty values (NaN) and duplicate data are also removed. Further cleaning is done on the text content by removing URLs, usernames, mentions, as well as other special characters that are not relevant in sentiment analysis [23]. This step is important as clean data can help reduce noise and improve the accuracy of the classification model. After the entire cleaning process was completed, the amount of tweet data was reduced to 1510.

2) Case Folding

Case folding is the process of converting all letters in a text into uniform lowercase or uppercase [24]. This action is simple but important in avoiding duplication of tokens that should be the same, such as the words "Monkeypox" and "monkeypox". In this research, all text will be uniformed to lowercase form.

3) Tokenization

Tokenization is the process of breaking a sentence into separate words, so that each word can be analyzed individually [21][25]. This step is crucial in forming a vector representation of the text. The choice of tokenization technique affects how the model understands the word context. For example, space-based tokenization is unable to capture certain idioms or phrases. Therefore, accurate tokenization can improve the understanding of context by the model, although it sometimes requires customization for informal language commonly used in social media.

4) Stopword Removal

Stopword removal is the process of removing common words such as "and", "the", or "is" that often appear in large amounts of text and are considered to have no important meaning in text analysis [26]. Although stopwords do not carry a strong sentiment meaning, the removal of these words can affect the context in the sentence. Therefore, the selection of stopwords that are appropriate to the health domain is important to ensure that relevant information is retained.

5) Stemming

The stemming process is a step to find the basic form of words that contain affixes, by removing prefixes, inserts, suffixes, or combinations of prefixes and suffixes [21]. The purpose of stemming is to simplify various forms of words that have similar meanings into one basic form. For example, the words "penularan" and "menular" will be reduced to "tular". This is beneficial for reducing sparsity in data representation. However, overly aggressive stemming can inaccurately change the meaning of words or produce unnatural base forms, especially in the context of Indonesian. Therefore, the application of stemming

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

needs to be tailored to an algorithm that supports the characteristics of the target language.

Overall, each preprocessing step makes an important contribution in shaping data that is better prepared to be analyzed by the model. However, the selection and application of each step must be adjusted to retain the semantic meaning relevant to the sentiment. The output generated from preprocessing can be seen in Table 3.

Table 3. Example of preprocessing results from one of the tweets.

Preprocessing	Result
Basic Sentence	Waspada Penularan Penyakit Cacar Monyet di Singkawang #BNetwork https://t.co/rFD9UUn7tU
Data Cleaning	Waspada Penularan Penyakit Cacar Monyet di Singkawang
Case Folding	waspada penularan penyakit cacar monyet di singkawang
Tokenization	['waspada', 'penularan', 'penyakit', 'cacar', 'monyet', 'di', 'singkawang']
Stopword Removal	['waspada', 'penularan', 'penyakit', 'cacar', 'monyet', 'singkawang']
Stemming	['waspada', 'tular', 'sakit', 'cacar', 'monyet', 'singkawang']

D. Split Data

Split data is the process of dividing data into training and testing data [27]. The training data is used to train the algorithm in building the model, while the testing data is used to assess the performance of the model and measure the extent to which the model is able to classify the data correctly [28]. The division of the amount of training data and testing data is a factor that affects accuracy [29]. Therefore, errors in determining the composition of the two types of data can affect the accuracy value obtained. In this study, the division is done with a proportion of 80% (1208 data) for training data and 20% (302 data) for testing data.

E. Word Embedding

Word embedding is one of the basic techniques in natural language processing (NLP) that serves to preserve the semantic meaning of words. This technique converts words into numerical vectors, where words that have meaning relationships will be close to each other in a multidimensional space [30]. In 2013, Tomas Mikolov and his colleagues introduced the Word2Vec algorithm, a word embedding technique capable of representing each word in the form of a vector with semantic meaning. Word2Vec has two main architectures, namely Continuous Bag-of-Words (CBOW) and Skip-Gram. Later, Pennington and his colleagues from Stanford developed another word embedding model called GloVe (Global Vectors). GloVe combines a global matrix factorization approach with local context information. Furthermore, Bojanowski and his colleagues extended the Word2Vec approach by developing FastText. FastText is a word embedding model, where each word is represented through an n-gram character [10]. Along with the increasing popularity of word embedding in the field of Natural Language Processing (NLP), there is a motivation to compare the performance of various existing word embedding models. In this study, the word embedding methods used are Word2Vec, GloVe, and FastText, each of which uses a vector size of 100 to represent words in the form of numeric vectors. The selection of these three methods is based on their ability to capture various aspects of word meaning and context that are highly relevant for our sentiment analysis.

1) Word2Vec

In 2013, Mikolov and his colleagues introduced Word2Vec which is a simple artificial neural network-based model with one hidden layer. The model receives input in the form of chunks of text from the training corpus, which are obtained through a sliding window mechanism. In each window, a single word is selected as the target word, while the surrounding words are considered as context [12]. Word2Vec consists of two main training approaches, namely the CBOW (Continuous Bag-of-Words) model and the Skip-gram model, both of which are built using a shallow neural network consisting of an input layer, a hidden layer, and an output layer [31]. An illustration of both models is shown in Fig. 3 [30].

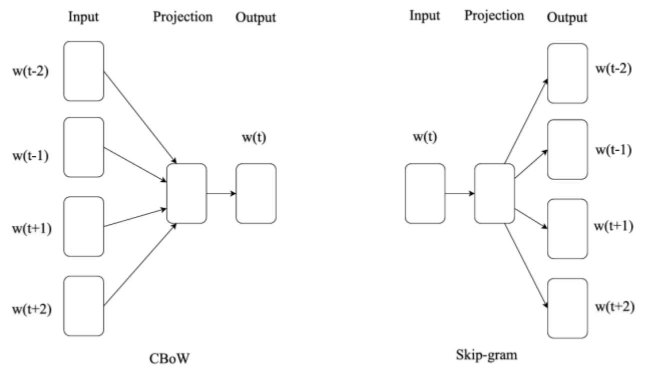

Fig. 3. Architecture of the Continuous Bag of Words (CBOW) model and the Skip-Gram model.

CBOW and skip-gram models utilize the weights of the hidden layer obtained during the training process to represent words in vector form. Although the goal of both is to form a word representation, the training method is different. The CBOW model works by predicting the target word based on the surrounding words (context), using multiple words as input to estimate the center word. This approach is suitable for small-sized data and has a fast

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

training time as it utilizes word distribution information from the context. In contrast, the Skip-gram model works by predicting context words based on a single input word (the middle word). Although Skip-gram training tends to be slower, it is more effective in handling large datasets and infrequently occurring words or phrases [31].

To better understand how Word2Vec works in building a word vector representation, Algorithm 1 presents the pseudocode of Word2Vec [32].

Algorithm 1. Pseudocode of the Word2Vec algorithm for generating vector representations.

Word2Vec

Input: *d*: dataset

Output: Matrix $W_{(256,\ 50)}$ consisting of one-hot encoded vectors representing each possible byte value (0–255)

Initialize a list *f* to store tuples of the form (byte_value, frequency)
for *i* := 0 to 255 **do**
freq ← 0
for each item *j* **in** the dataset *d*
freq ← freq + $\text{frequencyOfOccurence}_{(i,\ j)}$
end for
append the tuple (*i*, freq) to the list *f*
end for
f ← organize f by sorting it by frequency value
W ← word2vec(*f*, 50)
return *W*

2) GloVe

GloVe (Global Vectors) is an algorithm developed by Pennington and colleagues in 2014 as an unsupervised learning method that aims to generate word representations in vector form. The approach begins by building a co-occurrence matrix that records the frequency of co-occurrence between words in the entire corpus.

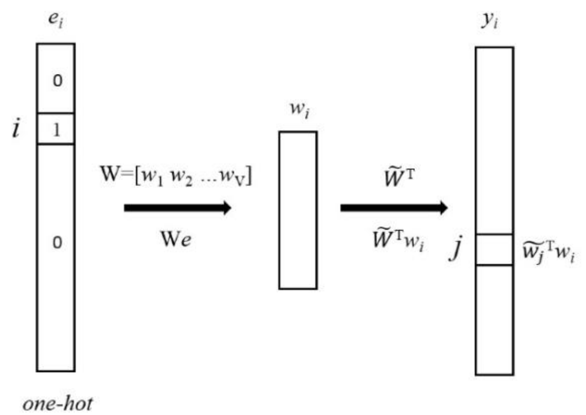

Fig. 4. GloVe model architecture for capturing global word co-occurrence statistics.

GloVe combines local and global context information of word occurrences to capture semantic relationships between words more accurately. Furthermore, it constructs an objective function that approximates the logarithm of the co-occurrence probability, utilizing the probability ratio to represent the relationship between words more efficiently [30]. Fig. 4 shows the architecture of the GloVe model [33]. Algorithm 2 presents the pseudocode of GloVe [34].

Algorithm 2. Pseudocode of the GloVe algorithm for generating word vectors based on word co-occurrence statistics.

GloVe

Input: Corpus *D*, sliding window size *ws*

Output: word vector v^s

Begin

For each word c_i in corpus *D*:
let c_i be the centre, *ws* be the radius, D_i is the co-occurrence word set
for D_i in *D*:
f_i = count(c_i) + 1, where f_j is the word frequency of c_i
update the co-occurrence matrix *X* using c_i and f_i
For *i* in *X*:
train the GloVe model using equation

$$J = \sum_{i,j}^{N} f(X_{ij})(v_i^T v_j + b_i + b_j - \log(X_{ij}))^2$$

update the parameters of the objective function
word vector v^s is obtained until the model converges
Return v^s

End

3) FastText

FastText is an extension of Word2Vec developed by Facebook AI Research in 2016 [12]. FastText provides a vector representation for each n-gram, and a word is represented as the sum of these n-gram vectors. This approach is particularly useful for languages with complex morphological structures, where a word can have many different forms. In its training process, FastText uses CBOW or Skip-gram models, and applies techniques such as negative sampling or hierarchical softmax to reduce computational load and handle large vocabularies efficiently [30]. Fig. 5 illustrates how FastText works using the Skip-Gram architecture [17].

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

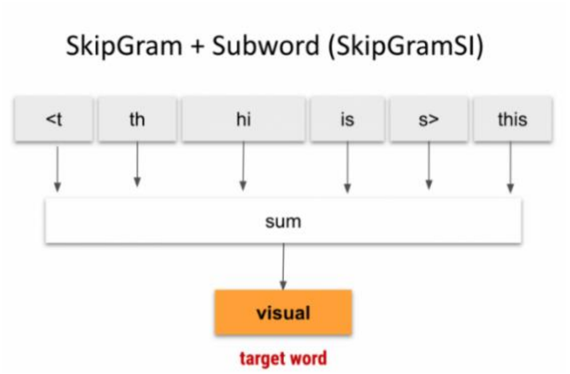

Fig. 5. Illustrates how FastText works using the Skip-Gram architecture.

The symbols '<' and '>' are used to mark the beginning and end of a word [35]. FastText is able to recognize and generate vector representations for words that have never appeared in the training data, thus increasing its flexibility and effectiveness in various natural language processing applications [30]. Algorithm 3 shows the steps performed in the FastText method [35].

Algorithm 3. Steps of the FastText algorithm illustrating subword-based word embedding using the Gensim library.

FastText

Start

1: Using fastText from the Gensim library

2: Input the Content column in FastText

3: Set iteration parameters, windows, and dimensions

4: Running fastText

End

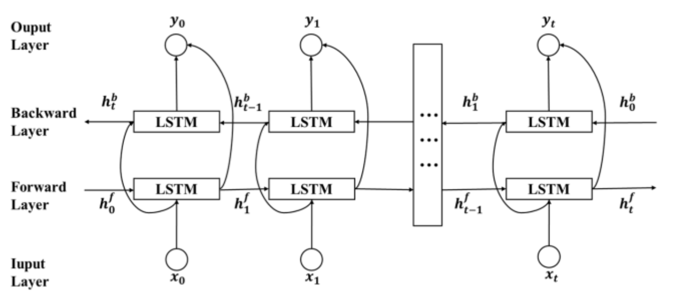

Fig. 6. Model architecture of Bidirectional Long Short-Term Memory (BiLSTM).

F. Bidirectional Long Short-Term Memory

BiLSTM (Bidirectional Long Short-Term Memory) is a variant of LSTM network designed to process data in two directions, namely forward and backward. With this capability, BiLSTM is very effective in capturing long-term dependencies in a sequence, as it considers the context from both directions. This bidirectional approach is very beneficial in understanding the context thoroughly, especially in tasks such as sequential recommendation systems that require understanding the order of user preferences [36]. The architecture of BiLSTM can be seen in Fig. 6 [37].

In the BiLSTM architecture, the process in the forward LSTM ($\overrightarrow{h_f}$) that processes the sequence of features from L_{C1} to L_{C300} is expressed as Eq. (1) [38].

$$\overrightarrow{h_f} = \overrightarrow{LSTM}(Lc_n), n \in [1, 300] \quad (1)$$

while the process on the backward LSTM ($\overleftarrow{h_b}$) that processes the sequence from L_{C300} to L_{C1} is expressed as Eq. (2).

$$\overleftarrow{h_b} = \overleftarrow{LSTM}(Lc_n), n \in [300, 1] \quad (2)$$

The final output of BiLSTM is the combined result of forward and backward which is expressed as Eq. (3).

$$h_n = [\overrightarrow{h_f}, \overleftarrow{h_b}] \quad (3)$$

In this study, the BiLSTM model was built using the TensorFlow-Hard framework. The model architecture is designed to classify sentiments in three categories, namely positive, negative, and neutral. The model configuration consists of several layers designed sequentially with the following details:

1) The embedding layer uses pre-trained word embedding from the feature extraction method (Word2Vec, GloVe, or FastText) with dimensions according to the vector representation of each method. This layer is non-trainable so the weights are not updated during training.
2) SpatialDropout1D is applied after the embedding layer with a dropout rate of 20% to prevent overfitting by removing features spatially.
3) Bidirectional LSTM Layer consists of one bidirectional LSTM layer, each with 100 units. The dropout and recurrent dropout are each set at 0.2 to reduce overfitting on both the input and recurrent connections.
4) The dense output layer consists of three neurons with softmax activation function, which corresponds to the number of sentiment classes.
5) The model was compiled using the sparse_categorical_crossentropy loss function since the labels were encoded as integers, as well as using the Adam optimizer with default parameters.

The configuration of each layer is shown in Table 4.

G. Evaluation

Model evaluation is the process of knowing the performance of the model in understanding the extent to which the model can produce the expected predictions, as well as to determine whether the model works well or vice versa. In this study, the method used to measure model performance is confusion matrix. Confusion matrix is a table used to represent the amount of test data that is classified correctly or incorrectly, thus helping to evaluate the accuracy of a classification system [17]. By using

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

confusion matrix, the performance of a classification system can be analyzed in detail including identifying where classification errors occur. This technique is a simple yet effective method to measure the performance of a classification system. The following is an illustration of the confusion matrix shown in Table 5.

Table 4. Architecture Configuration of Bidirectional Long Short-Term Memory (BiLSTM) Model.

Layer	Parameters
Embedding	input_dim: vocabulary size output_dim: embedding dimension (100) weights: pre-trained embedding matrix input_length: input length (number of tokens per tweet) trainable: False
SpatialDropout1D	rate: 0.2
Bidirectional LSTM	units: 100 (for each direction) dropout: 0.2 recurrent_dropout: 0.2
Dense	units: 3 (number of sentiment classes) activation: softmax
Model Compilation	loss: sparse_categorical_crossentropy optimizer: Adam metrics: accuracy

Table 5. Confusion matrix for negative, neutral, and positive sentiment classification.

Confusion Matrix		Prediction		
		Positive	Neutral	Negative
Actual	Positive	True Positive (TP)	False Neutral (FNeu)	False Negative (FNeg)
	Neutral	False Positive (FP)	True Neutral (TNeu)	False Negative (FNeg)
	Negative	False Positive (FP)	False Neutral (FNeu)	True Negative (TNeg)

The confusion matrix used in this study classifies sentiments into three categories, namely positive, neutral, and negative. In this context, True Positive (TP) refers to the number of data that belong to the positive category and are correctly classified by the system. True Neutral (TNeu) indicates the amount of data that are actually neutral and are accurately identified as such, while True Negative (TNeg) refers to the number of data from the negative category that are correctly classified. On the other hand, False Positive (FP) describes the amount of data that are actually not positive, such as neutral or negative, but are incorrectly classified as positive by the system. False Neutral (FNeu) refers to data from the positive or negative category that are misclassified as neutral. Lastly, False Negative (FNeg) represents the number of data that come from the positive or neutral category but are wrongly predicted as negative. This detailed breakdown provides a more comprehensive evaluation of model performance by identifying specific areas where misclassification occurs. To calculate the accuracy, precision, and recall values, the formulas formulated in Eq. (4), Eq. (5), Eq. (6), and Eq. (7) are used.

$$Total = \mathrm{TP} + \mathrm{FNeu} + \mathrm{FNeg} + \mathrm{FP} + \mathrm{TNeu} + \mathrm{FNeg} + \mathrm{FP} + \mathrm{FNeu} + \mathrm{TNeg} \quad (4)$$

$$accuracy = \frac{\mathrm{TP+TNeu+TNeg}}{\mathrm{Total}} \quad (5)$$

$$precision = \frac{\mathrm{TP}}{\mathrm{TP+FP+FP}} \quad (6)$$

$$recall = \frac{\mathrm{TP}}{\mathrm{TP+FNeu+FNeg}} \quad (7)$$

3. RESULTS

In this study, the evaluation results of sentiment analysis related to monkeypox using BiLSTM were presented with a comparison of three word embedding methods, namely Word2Vec, FastText, and GloVe. The evaluation was conducted to determine which model provided the highest accuracy among the three word embedding methods. The evaluation results included performance metrics such as accuracy, precision, recall, and F1-Score for each model tested. The research results regarding the performance of each word embedding with the BiLSTM algorithm were shown in Table 6.

Table 6. Performance metric results of Word2Vec, GloVe, and FastText using BiLSTM.

	Model	Perfomance Metrics			
		Acuracy	Precision	Recall	F1-Score
BiLSTM	Word2Vec	0.886	0.887	0.886	0.886
	GloVe	0.874	0.875	0.874	0.874
	FastText	0.898	0.898	0.898	0.897

The evaluation results in Table 6 showed that the BiLSTM model with FastText embedding achieved the highest performance in terms of accuracy, precision, recall, and F1-score. These metrics were important in sentiment analysis for public health issues such as monkeypox. Accuracy provided a general measure of the model's correctness, while precision ensured that the classified sentiment (positive, negative, or neutral) was relevant and not misleading, especially important in

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

preventing the spread of misinformation related to disease outbreaks. Recall reflected the model's ability to identify all relevant sentiments, which was crucial for comprehensively capturing public opinion. F1-score, as the harmonic mean of precision and recall, offered a balanced indicator of model performance and was particularly useful in cases with class imbalance. Overall, these four metrics provided a reliable framework for evaluating the model's ability to support sentiment-based health information analysis. Fig. 7 shows the performance comparison of the three word embedding models.

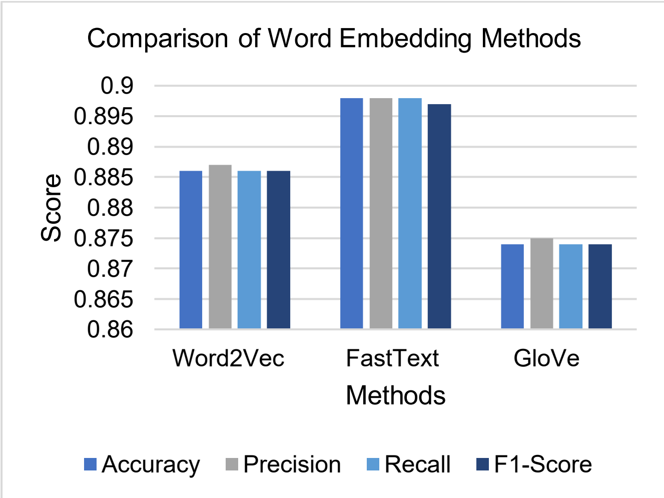

Fig. 7. Performance comparison of Word2Vec, FastText, and GloVe word embedding methods.

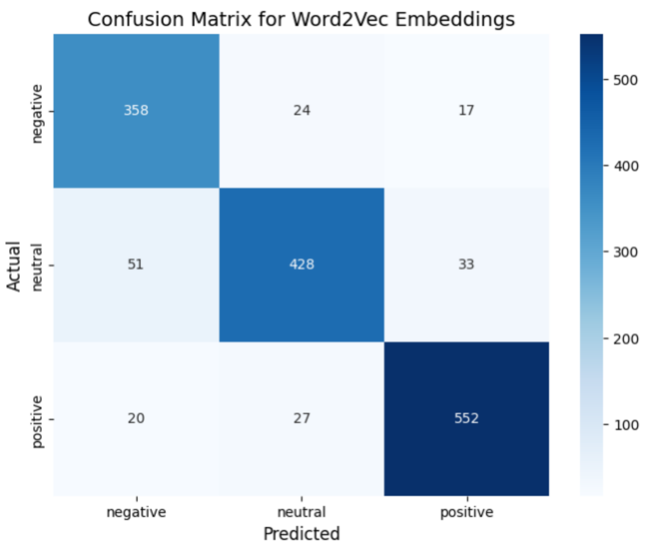

Fig. 8. Visualization of confusion matrix from the result of applying Word2Vec using BiLSTM model.

4. DISCUSSION

This research used tweet data related to monkeypox to compare several word embedding methods, namely Word2Vec, FastText, and GloVe, using the BiLSTM algorithm. Each word embedding method used a vector dimension size of 100. The tweet data used in the research had gone through a manual labeling process to determine the sentiment of each tweet namely positive, negative, and neutral. After labeling, the data also underwent a preprocessing stage which included data cleaning, case folding, tokenization, stopword removal, and stemming. This process aimed to clean the data from irrelevant elements and homogenize the text format to make it easier to be processed by the model. Next, the data was divided into two parts, 80% (1208 data) for training data and 20% (302 data) for testing data. This division aimed to ensure that the model could be optimally trained and objectively tested.

Based on the research results, a confusion matrix was obtained that illustrated the differences in model performance in classifying negative, neutral, and positive sentiments. In the Word2Vec method, the BiLSTM model was able to classify sentiment quite well, although there were still some misclassifications. In Fig. 8, 358 negative data were classified correctly, while 24 negative data were mispredicted as neutral and 17 as positive. For neutral sentiment, there were 428 correctly classified data, but 51 data were wrongly predicted as negative and 33 as positive. Meanwhile, for positive sentiment, 552 data were correctly classified, but 27 data were incorrectly predicted as neutral and 20 data as negative. These results show that the model has high accuracy, but still has difficulty distinguishing neutral sentiment from other categories, which may be caused by context similarity or ambiguity in sentiment expressions in the tweet data.

In the GloVe method, the BiLSTM model was able to classify the sentiment quite well, although there were still a number of misclassifications. In Fig. 9, 344 negative data were classified correctly, while 39 negative data were mispredicted as neutral and 16 as positive. For neutral sentiment, there were 444 correctly classified data, but 29 data were incorrectly predicted as negative and 39 as positive. As for the positive sentiment, 532 data were classified correctly, but 53 data were wrongly predicted as neutral and 14 data as negative.

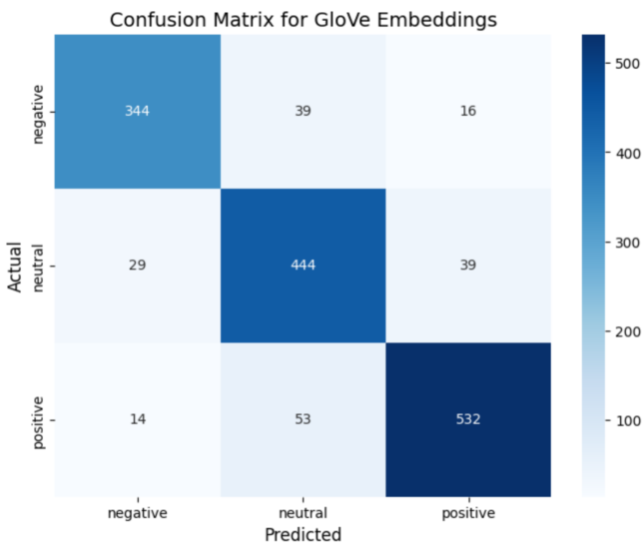

Fig. 9. Visualization of confusion matrix from the result of applying GloVe using BiLSTM model.

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

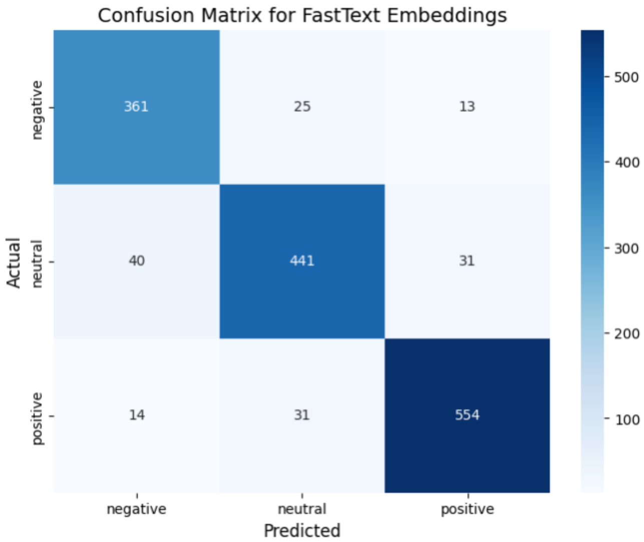

Fig. 10. Visualization of confusion matrix from the result of applying FastText using BiLSTM model.

Meanwhile, the FastText method shows the best performance compared to the previous two methods, Word2Vec and GloVe. In Fig. 10, 361 negative data were correctly classified, while 25 negative data were incorrectly predicted as neutral and 13 data as positive. For neutral sentiment, there were 441 correctly classified data, but there were still 40 data that were wrongly predicted as negative and 31 data as positive. For positive sentiment, the model successfully classified 554 data correctly, while 31 data were incorrectly predicted as neutral and 14 data as negative. One of the reasons why FastText shows the best results is because of its ability to capture sub-word information. Unlike Word2Vec and GloVe, which represent words as a single unit, FastText breaks words into multiple n-grams of characters, so it can recognize word meanings that have never been seen before. This is especially helpful in the context of social media like Twitter, which tends to have many variations in spelling, abbreviations, or typos. For example, variations in the writing of "cacar monyet" such as cacrmonyet, cacar_monyet, or other forms can still be recognized by FastText because the model understands the internal structure of words. This ability makes FastText better at capturing the nuances of sentiment, especially in words that rarely appear or are uncommon.

Based on the confusion matrix results, it can be concluded that the FastText method combined with the BiLSTM model provides the best classification performance in this study, with an accuracy of 89.8% as shown in Table 7. This finding is in line with several previous studies that have also shown the effectiveness of combining deep learning models and embedding methods in sentiment analysis tasks. For example, Iparraguirre-Villanueva et al. [39] used a CNN-LSTM hybrid model to analyze public sentiment towards the monkeypox virus based on Twitter data, and obtained an accuracy of 83%. Cai et al. [40] used a combination of BERT-BiLSTM to analyze investor and consumer sentiment in the energy market, with accuracy results reaching 86.20%. Furthermore, Saha et al. [41] applied Deep RNN-based LSTM architecture for sentiment

Table 7. Comparison of performance metrics results with previous research.

Study	Method & Model	Accuracy (%)	Precision	Recall	F1-Score
Iparraguirre-Villanueva et al. [39]	CNN-LSTM	83%	-	85%	83%
Cai et al. [40]	BERT	85.59%	57.96%	55.76%	56.84%
	BiLSTM	77.75%	16.34%	7.47%	10.25%
	BERT-BiLSTM	86.20%	57.70%	70.78%	63.57%
Saha et al. [41]	Naïve Bayes	78%	77%	81%	-
	SVM	77%	75%	79%	-
	Decision Tree	74%	73%	74%	-
	LSTM + 1-gram	81%	77%	82%	-
	LSTM + CBOW	83%	81%	81%	-
	LSTM + Skip-gram	83%	79%	80%	-
	LSTM + Context Encoder	85%	83%	84%	-
Afuan and Hidayat [17]	SVM +TF-IDF	72%	81%	72%	-
	SVM + FastText	73%	81%	73%	-
This Study (2025)	BiLSTM + Word2Vec	88.6%	88.7%	88.6%	88.6%
	BiLSTM + GloVe	87.4%	87.5%	87.4%	87.4%
	BiLSTM + FastText	89.8%	89.8%	89.8%	89.7%

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

analysis of Bengali-language political news and achieved an accuracy of 85%, outperforming other methods such as Naive Bayes, SVM, and Decision Tree. In addition, Afuan et al. [17] used the Support Vector Machine (SVM) algorithm to analyze public sentiment towards the Kampus Merdeka program, by comparing two feature extraction methods, namely TF-IDF and FastText. As a result, the FastText model obtained 73% accuracy, while TF-IDF was 72%. Although these studies showed good performance, the approach in this study using FastText and BiLSTM resulted in a higher accuracy of 89.8%. This improvement most likely stems from BiLSTM's ability to understand context in depth, as well as FastText's advantage in sub-word-based word representation, which is particularly effective for handling the informal and diverse language commonly found on social media platforms.

Despite the excellent results obtained, this study still has some limitations that need to be considered. One of them is the use of a limited vector dimension of 100. Although this dimension was quite representative for the purposes of this analysis, the representation of word meaning can be deeper if higher dimensions are used, although of course it requires more computational resources. In addition, the amount of data used in this study was also relatively small, with only 1510 tweets for training and testing, which may have affected the model's ability to generalize, especially when applied to more diverse data or on a larger scale in the real world. The possibility of bias in the manual labeling process was also a factor to consider, as it could have affected the accuracy and validity of the model.

Furthermore, the use of data from social media such as Twitter also raises broader implications regarding representation and potential bias. Platforms like Twitter tend to represent certain segments of the population, such as users who are younger, tech-savvy, and tend to be more vocal in voicing opinions in the digital public sphere. This can lead to sampling bias that affects the representativeness of the sentiment distribution in the data. In addition, social media content is informal and highly contextual, with the use of sarcasm, slang, or a mix of languages, which poses challenges for manual labeling and automatic classification. These biases may limit the model's ability to generalize beyond the current dataset, especially when applied to different populations or platforms. Future research should consider integrating more diverse data sources and applying bias mitigation approaches, such as user profiling, contextual analysis, or fairness-aware modeling approaches.

The findings of this study make an important contribution to the development of sentiment analysis models, particularly in the selection of optimal word embedding methods for unstructured text data such as tweets. This result shows that the selection of the right word representation can have a significant impact on the accuracy of the model, so it can be a reference for further research in the field of natural language processing (NLP). In addition, these findings have broad potential applications, especially in assisting policy makers in the field of public health. By utilizing deep learning-based sentiment analysis of public conversations on social media, authorities can gain real-time insight into public perceptions of health issues such as monkeypox. This information can be used to design more effective communication strategies, increase public awareness, and develop more responsive and data-driven intervention policies.

5. CONCLUSION

Based on the research results, three word embedding methods Word2Vec, FastText, and GloVe were evaluated using the BiLSTM algorithm for sentiment analysis related to monkeypox. The dataset used was 1511 tweets that had been collected through the crawling process on Twitter, then manually labeled into three sentiment categories namely positive, negative, and neutral. Furthermore, the data was processed through a preprocessing stage which included data cleaning, case folding, tokenization, stopword removal, and stemming. After the process, the final data amounted to 1510 tweets. The model was evaluated using accuracy, precision, recall, and F1-Score metrics. The results showed that FastText with BiLSTM gave the best performance with 89.8% accuracy, 89.8% precision, 89.8% recall, and 89.7% F1-Score. Word2Vec produced 88.6% accuracy, 88.7% precision, 88.6% recall, and 88.6% F1-Score. Meanwhile, GloVe obtained an accuracy of 87.4%, precision 87.5%, recall 87.4%, and F1-Score 87.4%. FastText proves to be more effective in capturing the meaning of words with more complex contexts, especially in reducing the misclassification of negative and positive sentiments. The results of this study confirm that choosing the right word embedding method has a significant influence on improving accuracy in sentiment analysis. However, this study still has some limitations that need to be considered. One of them is the relatively small dataset size, which may limit the model's ability to generalize to more varied data. In addition, the dimension of the embedding vectors used is also limited, which may limit the model's capacity to capture more complex semantic representations. The manual labeling process also has the potential to contain subjective bias, especially in distinguishing between neutral and negative or positive sentiments which are often ambiguous. For this reason, future research is recommended to use a larger and more diverse dataset, both in terms of number and variety of content. In addition, future research directions could also explore the use of more sophisticated models such as BERT, GPT, or other transformer-based approaches that have been shown to excel in understanding sentence context in depth. Semi-supervised or active learning approaches can also be considered to improve the quality of data labeling with more focused human intervention. Furthermore, it would be relevant to explore the applicability of hybrid models or test the effectiveness of these approaches on other health issues beyond monkeypox, such as COVID-19, air pollution, or vaccination issues. It is important to see to what extent the findings in this study can be generalized to different

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

types of public health crises. By doing so, this research is expected to become a broader reference in the development of real-time health issue monitoring systems based on sentiment analysis, especially in understanding public perceptions on social media.

REFERENCES

[1] S. Srivastava *et al.*, "The Global Monkeypox (Mpox) Outbreak: A Comprehensive Review," *Vaccines (Basel)*, vol. 11, no. 6, p. 1093, Jun. 2023, doi: 10.3390/vaccines11061093.

[2] World Health Organization, "Multi-country monkeypox outbreak: situation update," World Health Organization. Accessed: Jan. 10, 2025. [Online]. Available: https://www.who.int/emergencies/disease-outbreak-news/item/2022-DON396

[3] A. Letafati and T. Sakhavarz, "Monkeypox virus: A review," *Microb Pathog*, vol. 176, p. 106027, Mar. 2023, doi: 10.1016/j.micpath.2023.106027.

[4] Y.-H. Luo, T. Zhang, J.-L. Cao, W.-S. Hou, A.-Q. Wang, and C.-H. Jin, "Monkeypox: An outbreak of a rare viral disease," *Journal of Microbiology, Immunology and Infection*, vol. 57, no. 1, pp. 1–10, Feb. 2024, doi: 10.1016/j.jmii.2023.12.006.

[5] A. Karim, M. Mansab, M. Shahroz, M. F. Mushtaq, and I. cheol Jeong, "Anticipating impression using textual sentiment based on ensemble LRD model," *Expert Syst Appl*, vol. 263, p. 125717, Mar. 2025, doi: 10.1016/j.eswa.2024.125717.

[6] S. Bengesi, T. Oladunni, R. Olusegun, and H. Audu, "A Machine Learning-Sentiment Analysis on Monkeypox Outbreak: An Extensive Dataset to Show the Polarity of Public Opinion From Twitter Tweets," *IEEE Access*, vol. 11, pp. 11811–11826, 2023, doi: 10.1109/ACCESS.2023.3242290.

[7] Q. X. Ng, C. E. Yau, Y. L. Lim, L. K. T. Wong, and T. M. Liew, "Public sentiment on the global outbreak of monkeypox: an unsupervised machine learning analysis of 352,182 twitter posts," *Public Health*, vol. 213, pp. 1–4, Dec. 2022, doi: 10.1016/j.puhe.2022.09.008.

[8] S. A. Devi, M. S. Ram, P. Dileep, S. R. Pappu, T. S. M. Rao, and M. Malyadri, "Positional-attention based bidirectional deep stacked AutoEncoder for aspect based sentimental analysis," *Big Data Research*, vol. 39, p. 100505, Feb. 2025, doi: 10.1016/j.bdr.2024.100505.

[9] G. Sun, Y. Cheng, Z. Zhang, X. Tong, and T. Chai, "Text classification with improved word embedding and adaptive segmentation," *Expert Syst Appl*, vol. 238, p. 121852, Mar. 2024, doi: 10.1016/j.eswa.2023.121852.

[10] A. B. Nassif, A. Elnagar, I. Shahin, and S. Henno, "Deep learning for Arabic subjective sentiment analysis: Challenges and research opportunities," *Appl Soft Comput*, vol. 98, p. 106836, Jan. 2021, doi: 10.1016/j.asoc.2020.106836.

[11] D. Cao and M. K. Chan, "Enhancing chemical synthesis research with NLP: Word embeddings for chemical reagent identification—A case study on nano-FeCu," *iScience*, vol. 27, no. 10, p. 110780, Oct. 2024, doi: 10.1016/j.isci.2024.110780.

[12] F. Rollo, G. Bonisoli, and L. Po, "A Comparative Analysis of Word Embeddings Techniques for Italian News Categorization," *IEEE Access*, vol. 12, pp. 25536–25552, 2024, doi: 10.1109/ACCESS.2024.3367246.

[13] P. Anki and A. Bustamam, "Measuring the accuracy of LSTM and BiLSTM models in the application of artificial intelligence by applying chatbot programme," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, no. 1, pp. 197–205, Jul. 2021, doi: 10.11591/ijeecs.v23.i1.pp197-205.

[14] X. Li, X. Sun, Z. Xu, and Y. Zhou, "Explainable Sentence-Level Sentiment Analysis for Amazon Product Reviews," in *2021 5th International Conference on Imaging, Signal Processing and Communications (ICISPC)*, IEEE, Jul. 2021, pp. 88–94. doi: 10.1109/ICISPC53419.2021.00024.

[15] M. M. Rahman, A. I. Shiplu, Y. Watanobe, and M. A. Alam, "RoBERTa-BiLSTM: A Context-Aware Hybrid Model for Sentiment Analysis," *Journal of Latex Class Files*, vol. 14, no. 8, 2021, doi: https://doi.org/10.48550/arXiv.2406.00367.

[16] A. Glenn, P. LaCasse, and B. Cox, "Emotion classification of Indonesian Tweets using Bidirectional LSTM," *Neural Comput Appl*, vol. 35, no. 13, pp. 9567–9578, May 2023, doi: 10.1007/s00521-022-08186-1.

[17] L. Afuan and N. Hidayat, "Sentiment Analysis of the Kampus Merdeka Program on Twitter Using Support Vector Machine and a Feature Extraction Comparison: TF-IDF vs. FastText," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1738–1753, Dec. 2024, doi: 10.47738/jads.v5i4.436.

[18] C. Villavicencio, J. J. Macrohon, X. A. Inbaraj, J.-H. Jeng, and J.-G. Hsieh, "Twitter Sentiment Analysis towards COVID-19 Vaccines in the Philippines Using Naïve Bayes," *Information*, vol. 12, no. 5, p. 204, May 2021, doi: 10.3390/info12050204.

[19] M. J. Acosta, G. Castillo-Sánchez, B. Garcia-Zapirain, I. de la Torre Díez, and M. Franco-Martín, "Sentiment Analysis Techniques Applied to Raw-Text Data from a Csq-8 Questionnaire about Mindfulness in Times of COVID-19 to Improve Strategy Generation," *Int J Environ Res Public Health*, vol. 18, no. 12, p. 6408, Jun. 2021, doi: 10.3390/ijerph18126408.

[20] M. S. Gyimah, J. B. H. -Acquah, R.-M. M. Gyening, M. Asante, U. F. I. Abdulrahman, and E. Kotei, "AsanteTwiSenti: A Sentiment dataset of Ghanaian Asante Twi Tweets in a multilingual context," *Data Brief*, vol. 60, p. 111460, Jun. 2025, doi: 10.1016/j.dib.2025.111460.

[21] S. H. Imanuddin, K. Adi, and R. Gernowo, "Sentiment Analysis on Satusehat Application Using

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

Support Vector Machine Method," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 5, no. 3, pp. 143–149, Jul. 2023, doi: 10.35882/jeemi.v5i3.304.

[22] M. F. Akbar, M. I. Mazdadi, Muliadi, T. H. Saragih, and F. Abadi, "Implementation of Information Gain Ratio and Particle Swarm Optimization in the Sentiment Analysis Classification of Covid-19 Vaccine Using Support Vector Machine," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 5, no. 4, pp. 261–270, Sep. 2023, doi: 10.35882/jeeemi.v5i4.328.

[23] A. B. Alawi and F. Bozkurt, "A hybrid machine learning model for sentiment analysis and satisfaction assessment with Turkish universities using Twitter data," *Decision Analytics Journal*, vol. 11, p. 100473, Jun. 2024, doi: 10.1016/j.dajour.2024.100473.

[24] Rianto, A. B. Mutiara, E. P. Wibowo, and P. I. Santosa, "Improving the accuracy of text classification using stemming method, a case of non-formal Indonesian conversation," *J Big Data*, vol. 8, no. 1, p. 26, Dec. 2021, doi: 10.1186/s40537-021-00413-1.

[25] N. Darraz, I. Karabila, A. El-Ansari, N. Alami, and M. El Mallahi, "Integrated sentiment analysis with BERT for enhanced hybrid recommendation systems," *Expert Syst Appl*, vol. 261, p. 125533, Feb. 2025, doi: 10.1016/j.eswa.2024.125533.

[26] A. A. Aladeemy *et al.*, "Advancements and challenges in Arabic sentiment analysis: A decade of methodologies, applications, and resource development," *Heliyon*, vol. 10, no. 21, p. e39786, Nov. 2024, doi: 10.1016/j.heliyon.2024.e39786.

[27] N. Z. Al Habesyah, R. Herteno, F. Indriani, I. Budiman, and D. Kartini, "Sentiment Analysis of TikTok Shop Closure in Indonesia on Twitter Using Supervised Machine Learning," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 2, pp. 148–156, Apr. 2024, doi: 10.35882/jeeemi.v6i2.381.

[28] S. Gupta, K. Saluja, A. Goyal, A. Vajpayee, and V. Tiwari, "Comparing the performance of machine learning algorithms using estimated accuracy," *Measurement: Sensors*, vol. 24, p. 100432, Dec. 2022, doi: 10.1016/j.measen.2022.100432.

[29] A. Rácz, D. Bajusz, and K. Héberger, "Effect of Dataset Size and Train/Test Split Ratios in QSAR/QSPR Multiclass Classification," *Molecules*, vol. 26, no. 4, p. 1111, Feb. 2021, doi: 10.3390/molecules26041111.

[30] M. M. Czajka, D. Kubacka, and A. Świetlicka, "Embedding representation of words in sign language," *J Comput Appl Math*, vol. 465, p. 116590, Sep. 2025, doi: 10.1016/j.cam.2025.116590.

[31] J. Zhou, Z. Ye, S. Zhang, Z. Geng, N. Han, and T. Yang, "Investigating response behavior through TF-IDF and Word2vec text analysis: A case study of PISA 2012 problem-solving process data," *Heliyon*, vol. 10, no. 16, p. e35945, Aug. 2024, doi: 10.1016/j.heliyon.2024.e35945.

[32] N. Bakhshinejad and A. Hamzeh, "Parallel-CNN network for malware detection," *IET Inf Secur*, vol. 14, no. 2, pp. 210–219, Mar. 2020, doi: 10.1049/iet-ifs.2019.0159.

[33] C. Liu, P. Zhang, T. Li, and Y. Yan, "Semantic Features Based N-Best Rescoring Methods for Automatic Speech Recognition," *Applied Sciences*, vol. 9, no. 23, p. 5053, Nov. 2019, doi: 10.3390/app9235053.

[34] C. Zhang, W. Su, S. Chen, S. Zeng, and H. Liao, "A Combined Weighting Based Large Scale Group Decision Making Framework for MOOC Group Recommendation," *Group Decis Negot*, vol. 32, pp. 537–567, 2023, doi: https://doi.org/10.1007/s10726-023-09816-2.

[35] N. H. Pribadi, R. Sarno, A. S. Ahmadiyah, and K. R. Sungkono, "Semantic Recommender System Based on Semantic Similarity Using FastText and Word Mover's Distance," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 2, pp. 377–385, Apr. 2021, doi: 10.22266/ijies2021.0430.34.

[36] M. Valera and Dr. R. Mehta, "Advanced Deep Learning Models for Improving Movie Rating Predictions: A Benchmarking Study," *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*, vol. 4, no. 4, p. 100200, Dec. 2024, doi: 10.1016/j.tbench.2025.100200.

[37] Y. Fan, Q. Tang, Y. Guo, and Y. Wei, "BiLSTM-MLAM: A Multi-Scale Time Series Prediction Model for Sensor Data Based on Bi-LSTM and Local Attention Mechanisms," *Sensors*, vol. 24, no. 12, p. 3962, Jun. 2024, doi: 10.3390/s24123962.

[38] P. Wu, X. Li, C. Ling, S. Ding, and S. Shen, "Sentiment classification using attention mechanism and bidirectional long short-term memory network," *Appl Soft Comput*, vol. 112, p. 107792, Nov. 2021, doi: 10.1016/j.asoc.2021.107792.

[39] O. Iparraguirre-Villanueva *et al.*, "The Public Health Contribution of Sentiment Analysis of Monkeypox Tweets to Detect Polarities Using the CNN-LSTM Model," *Vaccines (Basel)*, vol. 11, no. 2, p. 312, Jan. 2023, doi: 10.3390/vaccines11020312.

[40] R. Cai *et al.*, "Sentiment Analysis About Investors and Consumers in Energy Market Based on BERT-BiLSTM," *IEEE Access*, vol. 8, pp. 171408–171415, 2020, doi: 10.1109/ACCESS.2020.3024750.

[41] B. N. Saha, A. Senapati, and A. Mahajan, "LSTM based Deep RNN Architecture for Election Sentiment Analysis from Bengali Newspaper," in *2020 International Conference on Computational Performance Evaluation (ComPE)*, IEEE, Jul. 2020, pp. 564–569. doi: 10.1109/ComPE49325.2020.9200062.

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88

AUTHOR BIOGRAPHY

Noryasminda is from Rangga Ilung, Central Kalimantan. After graduating from high school, she continued her education to the university level. Since 2021, she has been studying as a student of the Computer Science Study Program at Lambung Mangkurat University. Her current research focuses on data science, especially in the field of text analysis or sentiment analysis based on natural language processing (NLP). In addition, her final project centered on research on sentiment analysis related to monkeypox disease on X/Twitter social media. This research aims to understand the pattern of public perception of monkeypox and provide insights that can support the dissemination of more effective health information. Email: noryasminda@gmail.com.

Triando Hamonangan Saragih currently holding the position of a lecturer within the Department of Computer Science at Lambung Mangkurat University, is heavily immersed in the realm of academia, with a profound focus on the multifaceted domain of Data Science. His academic pursuits commenced with the successful completion of his bachelor's degree in Informatics at the esteemed Brawijaya University, located in the vibrant city of Malang, back in the year 2016. Building upon this foundational achievement, he proceeded to further enhance his scholarly credentials by enrolling in a master's program in Computer Science at Brawijaya University, Malang, culminating inthe conferral of his advanced degree in 2018. The research field he is involved in is Data Science. Email: triando.saragih@ulm.ac.id. Orcid ID: 0000-0003-4346-3323.

Rudy Herteno received his bachelor's degree in Computer Science from Lambung Mangkurat University in 2011. After completing his studies, he worked as a software developer for several years to gain more experience in the field. During this period, he developed various software applications, particularly to support the needs of local governments. In 2017, he obtained a master's degree in Informatics from STMIK Amikom University. Currently, he is a lecturer in the Computer Science program at Lambung Mangkurat University. His research interests include software engineering, software defect prediction, and deep learning, aiming to improve software quality, optimize error detection in systems, and develop artificial intelligence- based solutions. He can be contacted at email: rudy.herteno@ulm.ac.id. Orcid ID: 0000-0003-0637-8090.

Mohammad Reza Faisal was born in Banjarmasin. Following his graduation from high school, he pursued his undergraduate studies in the Informatics department at Pasundan University in 1995, and later majored in Physics at Bandung Institute of Technology in 1997. After completinghis bachelor's program, he gained experience as a training trainer in the field of information technology and software development. Since 2008, he has been a lecturer in computer science at Universitas Lambung Mangkurat, while also pursuing his master's program in Informatics at Bandung Institute of Technology in 2010. In 2015, he furthered his education by pursuing a doctoral degree in Bioinformatics at Kanazawa University, Japan. To this day, he continues his work as a lecturer in Computer Science at Universitas Lambung Mangakurat. His research interests encompass Data Science, Software Engineering, and Bioinformatics. He can be contacted at email: reza.faisal@ulm.ac.id. Orcid ID: 0000-0001-5748-7639.

Andi Farmadi is a senior lecturer in the Computer Science program at Lambung Mangkurat University. He has been teaching since 2008 and currently serves as the Head of the Data Science Lab since 2018. He completed his undergraduate studies at Hasanuddin University and his graduate studies at Bandung Institute of Technology. His research area, up to the present, focuses on Data Science. One of his research projects, along with other researchers, published in the International Conference of Computer and Informatics Engineering (IC2IE), is titled "Hyperparameter tuning using GridsearchCV on the comparison of the activation function of the ELM method to the classification of pneumonia in toddlers," and this research was published in 2021. Email: andifarmadi@ulm.ac.id. Orcid Id: 0009-0009-0926-8082.

Corresponding author: Triando Hamonangan Saragih, triando.saragih@ulm.ac.id, Department of Computer Science, Lambung Mangkurat University, Banjarbaru, Indonesia. **DOI:** https://doi.org/10.35882/ijeeemi.v7i2.88