

RESEARCH ARTICLE

OPEN ACCESS

Manuscript received July 27, 2023; revised August 20, 2023; accepted September 21, 2023; date of publication October 30, 2023
Digital Object Identifier (DOI): <https://doi.org/10.35882/jeemi.v5i4.319>

Copyright © 2023 by the authors. This work is an open-access article and licensed under a Creative Commons Attribution-ShareAlike 4.0 International License ([CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/)).

How to cite: Rahmi Yunida, Mohammad Reza Faisal, Muliadi, Fatma Indriani, Friska Abadi, Irwan Budiman, and Septyan Eka Prastya, "LSTM and Bi-LSTM Models For Identifying Natural Disasters Reports From Social Media, vol. 5, no. 4, pp. 241-249, October 2023.

LSTM and Bi-LSTM Models For Identifying Natural Disasters Reports From Social Media

Rahmi Yunida¹, Mohammad Reza Faisal¹, Muliadi¹, Fatma Indriani¹, Friska Abadi¹,
Irwan Budiman¹, and Septyan Eka Prastya²

¹Computer Science Department, Lambung Mangkurat University, Banjarbaru, South Kalimantan, Indonesia

²Information Technology Department, Sari Mulia University, Banjarmasin, South Kalimantan, Indonesia

Corresponding author: Mohammad Reza Faisal (e-mail: reza.faisal@ulm.ac.id).

ABSTRACT Natural disaster events are occurrences that cause significant losses, primarily resulting in environmental and property damage and in the worst cases, even loss of life. In some cases of natural disasters, social media has been utilized as the fastest information bridge to inform many people, especially through platforms like Twitter. To provide accurate categorization of information, the field of text mining can be leveraged. This study implements a combination of the word2vec and LSTM methods and the combination of word2vec and Bi-LSTM to determine which method is the most accurate for use in the case study of news related to disaster events. The utility of word2vec lies in its feature extraction method, transforming textual data into vector form for processing in the classification stage. On the other hand, the LSTM and Bi-LSTM methods are used as classification techniques to categorize the vectorized data resulting from the extraction process. The experimental results show an accuracy of 70.67% for the combination of word2vec and LSTM and an accuracy of 72.17% for the combination of word2vec and Bi-LSTM. This indicates an improvement of 1.5% achieved by combining the word2vec and Bi-LSTM methods. This research is significant in identifying the comparative performance of each combination method, word2vec + LSTM and word2vec + Bi-LSTM, to determine the best-performing combination in the process of classifying data related to earthquake natural disasters. The study also offers insights into various parameters present in the word2vec, LSTM, and Bi-LSTM methods that researchers can determine. The findings of the present study have demonstrated that, upon thorough examination and analysis, the Bi-Directional Long Short-Term Memory (Bi-LSTM) approach has resulted in superior performance when compared to the Long Short-Term Memory (LSTM) approach, particularly in conjunction with the word2vec feature extraction methodology.–

INDEX TERMS Bi-LSTM, earthquake, LSTM, natural disaster, word embedding

I. INTRODUCTION

Every year, several natural disasters in the United States and worldwide cause structural damage, deaths, and chaos [1]. Disasters can have significant social impacts on cultural heritage. They can also destroy cultural heritage, leading to long-term financial losses and livelihood disruptions [2]. In some cases of natural disasters, the presence of social media plays a crucial role in assisting every activity within the disaster management cycle. During the pre-disaster stage, social media can serve as an early warning before a disaster occurs [3]. Therefore, numerous studies have been conducted utilizing social media data as a social network sensor, as demonstrated by [4], which can function as a primary information dissemination platform. The concept of a social

network sensor is a new concept originating from physical sensors. This concept aims to blend the idea of physical sensors into cyberspace through social media [5]. The social media ecosystem, such as online platforms like Twitter, provides an environment where various individuals, both experts and non-experts, can easily share, discuss, and engage with knowledge. Its usage is evidence of the value Twitter brings to researchers. A study in 2017 reported that 1%-5% of 187 million Twitter users are active scientists [6]. Given the widespread and global use of social media, platforms like Twitter, a prominent communication medium, especially during disasters, can be utilized as emergency communication channels [7][8].

One of the techniques that can be used to conduct research and extract information from social media is in the field of Text Mining. Text Mining is a scientific field that provides methods to analyze and process unstructured data, constituting around 95% of big data [9]. The text mining domain has a subsection known as text classification. Text Classification, also known as document classification or text categorization, encompasses topics such as sentiment analysis, emotion detection, spam analysis, and document indexing [10]. Sentiment analysis involves classifying text opinions into categories such as positive, negative, and neutral. This is often referred to as subjectivity analysis, opinion mining, and sentiment extraction [11].

To perform sentiment analysis, the data needs to be preprocessed first. Word embedding, also known as word vector representation, is one technique to transform words into vectors or arrays consisting of numerical values. Word2vec is a neural network-based model [12] that is effectively used to identify synonyms frequently appearing in similar contexts of extensive data [13]. Word2vec has proven effective in representing the meanings of words. However, hyperparameter configuration and feature selection impact the performance of Word2vec [14].

For the classification process, there is a field that adopts the capabilities of recurrent neural networks (RNNs). RNNs are an Artificial Neural Network (ANN) that allows past knowledge to be used through a recurrent architecture [15]. Representative RNNs, such as Long Short Term Memory (LSTM), have made breakthrough progress in speech and video processing, social applications, text sentiment analysis, and more [16]. The outputs for each RNN layer employ Dense Layers, and spatial contributions are captured by combining information using additional fully connected layers [17]. One classification method that adopts RNNs is LSTM and Bi-LSTM. Long Short-Term Memory (LSTM) is a gated recurrent neural network adding a mechanism to control information flow within the network based on a more complex recurrent neural network [18]. Just like in [19], numerous misclassified categories and unknown types exist. Therefore, functional and algorithmic work is needed to organize categories, requiring an LSTM-based classification model to address that issue.

Also, the study [20] showed that LSTM classification also yielded good performance in case studies of detection and diagnosis of motor electrical disorders. Due to the good performance of this classifier, we suggest that this classifier can be used by society as a benchmark for the development of new and improved motor electrical fault classification algorithms. Apart from the LSTM method used as a classification technique, another technique is a variant of RNN known as Bidirectional LSTM (Bi-LSTM). In a bidirectional LSTM network, LSTM neurons are divided into two directions: one for the forward state and the other for the backward state. One forward state or backward state are of Bi-LSTM refers to one of the two states maintained by the Long Short-Term Memory (LSTM) within the architecture of a

Recurrent Neural Network (RNN) known as Bidirectional LSTM (Bi-LSTM).

The term one forward state pertains to the internal state of an LSTM cell in the forward portion of the Bi-LSTM. The backward state refers to the states generated by the LSTM cells that run in the backward direction (from end to start of the sequence). These states reflect the model's understanding of the sequence as it processes it in the reverse direction. The forward state refers to the states generated by the LSTM cells that run in the forward direction (from start to end of the sequence). These states reflect the model's understanding of the sequence as it processes it in the forward direction. [21],[22].

Several case studies and applications of these methods have been previously implemented in research. One of them is the research [23] that conducted sentiment analysis by comparing Bag-of-Words (BoW) with TF-IDF feature extraction and Word2vec, using LR and SVM classification. The results showed comparable performance for SVM classification, while for LR classification, Word2vec outperformed with the highest accuracy of 87.4%. A study [24] also analyzed sentiments towards products, services, politics, social events, and corporate strategies. Reviews (from sources like TripAdvisor, Amazon, and IMDB) and social media posts (primarily from Twitter and Facebook) were subjected to LSTM classification, showing good performance with 85% accuracy when more training data was available.

Furthermore, a study by [25] successfully implemented Bi-LSTM for text classification with a precision of 91.54%, recall of 92.82%, and an f1-score of 92.18%. Applying Bi-LSTM made the model work more optimally as contextual information from comment text data was effectively absorbed. The best model obtained outperformed RNN, CNN, regular LSTM, and Naïve Bayes algorithms. There's also research by [26] that conducted sentiment analysis in the Indonesian language, using LSTM with Word2vec as the word embedding method for sentiment analysis. The model achieved sentiment analysis with an accuracy of up to 85.96%.

Based on research [23]-[26], various methods are presented that are considered superior in feature extraction and classification processes. However, there has not been a study that explicitly combines these superior methods in a case study. For example, research [26] only focused on LSTM classification with Word2vec extraction, and research [25] only discussed the advantages of Bi-LSTM. Thus, based on previous literature review, this study proposes a Text Classification method focused on a case study of earthquakes based on information gathered from Twitter. Several relevant methods are used in this data processing process, including Word2vec as the text data extraction process obtained from Twitter, then LSTM and Bi-LSTM as text categorization techniques.

In addition to identifying the case study, this research also aims to contribute, a. Insights into which method yields the best results by comparing the accuracy obtained from LSTM and Bi-LSTM classification with feature extraction from

Word2vec; b. Information on how well the accuracy is achieved using the LSTM and Bi-LSTM methods; c. A reference that can be utilized by future researchers interested in studying earthquake case studies and as a guide for the appropriate method selection.

II. MATERIAL AND METHODS

In general, the research process involves comparing the classification outcomes of two methods: LSTM and Bi-LSTM. To facilitate this comparison, several stages are carried out. These stages include data collection, data preprocessing, feature extraction using word2vec, data partitioning into training and testing sets, model testing, and evaluation. The proposed model can be observed in the following [FIGURE 1](#).

A. DATASET

The dataset used in this study is available in [27], consisting of Twitter data related to earthquake disasters from <https://github.com/rezafaisal/NaturalDisasterOnTwitter>. Each of the used dataset already has labels for three classes that identify the source of the tweets, namely (i) eyewitness, (ii) non-eyewitness, and (iii) don't know. The total number of data used is 3000, with an equal distribution of 1000 data points for each class. In the eyewitness category are messages about natural disasters posted by eyewitnesses at the disaster's location. Messages in the non-eyewitness category are messages about natural disasters uploaded by users who are not eyewitnesses. On the other hand, messages in the don't know category contains words related to natural disasters, but the meaning is not about natural disasters [28]. Examples of the sample dataset used can be seen in [TABLE 1](#).

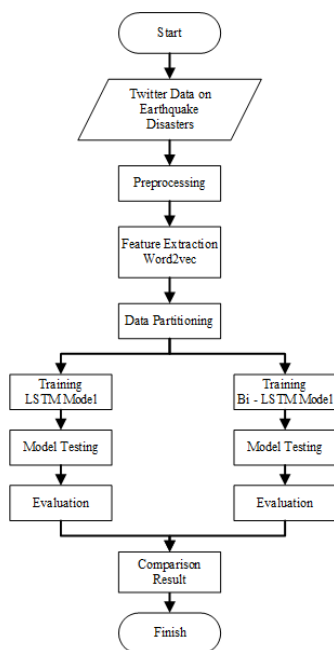


FIGURE 1. The Research Flow of LSTM and Bi-LSTM Classification Models

TABLE 1
Detail Dataset

No.	Data	Class
1.	Pantesan goyang goyang, gempa ternyata :(eyewitness
2.	Gempa btw :(
3.	Jadi tadi sekitar jam 15.20 gw ngerasain gempa:)	
4.	Beneran gempa di Sukabumi ternyata.. huuu	
5.	gempa yg kali ini kuat skali padahal beda 0.7 dari yg kemaren	
1.	Eh gempa dimana? Stay safe semua nya ??????	non-eyewitness
2.	Tadi jam 11an sukabumi gempa lagi yak 4.5 sr tapi ttp aja ak di bogor tida merasakan apa2	
3.	Gempa bermagnitudo 4,5 mengguncang Kabupaten Sukabumi, Jawa Barat, Senin (16/3/2020) pukul 10.56 WIB.	
4.	BMKG Catat Gempa M 5.0 Guncang Pacitan Jawa Timur Sore Ini	
5.	Sumpah mama bangun2 teriak gempa2 !!! aku shock kaget... Aku pribadi ga ngerasa apa2, Pas liat di bmkg masa gempa nya di ambon??? gmn si(?)	
1.	gledek kenceng banget sampe goyang kayak lagi gempa;(((don't know
2.	pusing dilevel dikit dikit ngerasa gempa	
3.	kepala gua dari tadi pusing kenapa yak..... ampe gua kira tuh gempa:'))	
4.	knp ngerasa kaya gempa mulu si, kaya org gila tbtb kaget sendiri	
5.	gledek kenceng banget sampe goyang kayak lagi gempa;(((

B. PREPROCESSING

Preprocessing is the process of enhancing the quality of raw data before using it in the subsequent stages. The preprocessing steps in this study involve the following actions: Cleansing, which involves removing characters that do not contribute to sentiment analysis, leaving only alphabet characters. This helps eliminate unnecessary noise from the text data. Case Folding, In case of folding, all words are converted to lowercase. This ensures uniformity in the text data and makes it easier to process and analyze, as capitalization differences are disregarded. Tokenization, Tokenization is the process of converting text into tokens or smaller units such as words or phrases. This is done before transforming the text into vectors, making it easier to filter out unnecessary tokens. By conducting these preprocessing steps, the text data is refined and prepared for further analysis, enhancing the quality and consistency of the dataset for sentiment analysis.

C. FEATURE EXTRACTION

Feature extraction is the process of obtaining characteristics that describe a piece of data. The feature extraction method used is word vector representation, also known as word2vec. Word2vec is a natural language processing technique. The word2vec algorithm learns word associations from an enormous corpus of text using a neural network model [29]. Word2vec is a feature extraction method applied to map words into vectors, capturing the meanings and contexts of words within documents. This method has two architectural models: Continuous Bag-of-Words (CBOW) and Skip-Gram. Both models consist of input, projection, and output layers,

although their processes for generating output differ. The input layer takes $W_n = \{W(t-2), W(t-1), \dots, W(t+1), W(t+2)\}$ as arguments, where W_n represents a word. The projection layer adapts to a multidimensional vector array and accumulates the sum of several vectors. The output layer then displays the resulting vector from the projection layer [30].

Word embedding schemes like Word2vec and others assign equal weights to each word in a sentence and compute the average embedding of each word. In both supervised and unsupervised Natural Language Processing (NLP) tasks, it has been proven that weighted word embeddings can enhance performance [31],[32].

D. LSTM

LSTM (Long Short-Term Memory) is an advancement over conventional RNNs, which only possess a single type of memory. The "A" unit structure of LSTM incorporates gate mechanisms that regulate the flow of information within the memory or cell state. LSTM introduces gate mechanisms: the input gate, forget gate, and output gate. The forget gate determines which information from the cell state is discarded. The input gate determines new information that is stored within the cell state. The input gate computes a new value to update the cell state, while the output gate determines the output value based on the cell state [33]. The structure of LSTM can be observed in [FIGURE 2](#).

In the LSTM gates, a sigmoid (σ) function is the activation function, which assigns values of 0 or 1. These binary values are used to provide clear and positive gate outputs. A value of 0 serves to discard or forget a feature, while a value of 1 signifies that the feature should be stored within the network. The equations for the input gate are represented by equation (1), the forget gate by equation (2), and the output gate by equation (3).

$$i_t = w_i [h_{t-1}, x_t] + b_i \tag{1}$$

$$f_t = w_f [h_{t-1}, x_t] + b_f \tag{2}$$

$$o_t = w_o [h_{t-1}, x_t] + b_o \tag{3}$$

In equation (1), "i" represents the input gate, "w" signifies the weight for the 'x' gate, "ht-i" stands for the output from the previous LSTM unit at time "t - 1", "xt" denotes the input for the current LSTM unit at time "t", "bx" is the bias for the 'x' gate, "ft" corresponds to the forget gate, and "ot" represents the output gate [24].

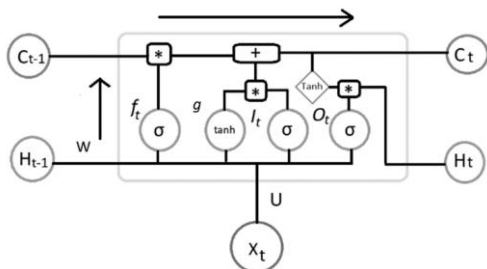


FIGURE 2. The General Architecture of LSTM

E. Bi - LSTM

Bidirectional means two directions. Bi-LSTM applies bidirectional input to the recurrent LSTM layer. The initial idea is to utilize information available from both the past and the future to optimize the utilization of the existing information. The application of Bidirectional in recurrent neural networks involves combining two independent recurrent neural networks so that the network possesses both backward and forward information about the sequence at each time step [21],[22]. Bi-LSTM is particularly useful for sequential labeling tasks when access to information before and after a given point is crucial. However, the hidden state in LSTM only captures information from the past, while information following it is not known. This issue can be addressed using Bi-LSTM [25]. Fundamentally, Bi-LSTM consists of two LSTM networks – the forward LSTM and the backward LSTM – which capture information from both directions and mitigate the vanishing gradient problem in RNN methods. Bi-LSTM has demonstrated excellent results in various Natural Language Processing tasks.

Bi-LSTM encompasses numerous parameters and hyperparameters, among which commonly used ones include epochs, batch size, dropout rate, optimizer, learning rate, word vector dimension, the number of neurons in the LSTM hidden layer, L2 regularization lambda, and loss function. The architecture of Bi-LSTM can be observed in [FIGURE 3](#).

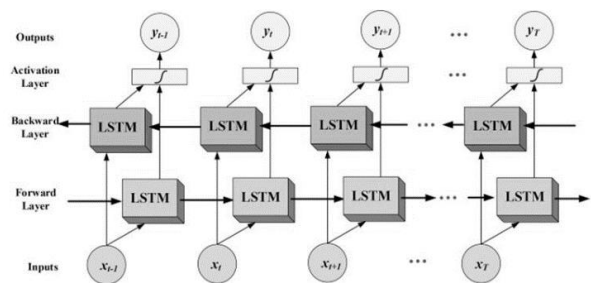


FIGURE 3. The General Architecture of Bi - LSTM

III. RESULT

This section compares the accuracy produced by LSTM and Bi-LSTM, which is utilized to determine the success rate of the chosen method. However, the used dataset needs to be preprocessed to ascertain the achieved accuracy. The first step involves data preprocessing, which encompasses cleansing, case folding, and tokenizing. The preprocessing results on the data can be observed in [TABLE 2](#).

No.	Data	Class
1.	<i>pantesan goyang goyang gemp</i> <i>ternyata</i>	eyewitness
2.	<i>gempa btw</i>	
3.	<i>jadi tadi sekitar jam 15.20 gw ngerasain</i> <i>gempa</i>	
4.	<i>beneran gempa di sukabumi ternyata</i> <i>huuu</i>	

No.	Data	Class
5.	<i>gempa yg kali ini kuat sekali padahal beda 07 dari yg kemaren</i>	
1.	<i>yang barusan kena gempa stay safe ya</i>	
2.	<i>eh gempa dimana stay safe semua nya</i>	
3.	<i>tadi jam 11an sukabumi gempa lagi yak 45 sr tapi ttp aja ak di bogor tida merasakan apa2</i>	non-eyewitness
4.	<i>gempa dengan kekuatan magnitudo 45 mengguncang kabupaten sukabumi jawa barat senin 1632020 pukul 1056 wib gempa tersebut berpusat di laut 97 km tenggara sukabumi</i>	
5.	<i>bmkg catat gempa m 50 guncang pacitan jawa timur sore ini</i>	
1.	<i>sumpah mama bangun2 teriak gempa2 aku shock kaget aku pribadi ga ngerasa apa2 pas liat di bmkg masa gempa nya di ambon gmn si</i>	don't know
2.	<i>gledek kenceng banget sampe goyang kayak lagi gempa</i>	
3.	<i>pusing dilevel dikit dikit ngerasa gempa</i>	
4.	<i>kepala gua dari tadi pusing kenapa yak ampe gua kira tuh gempa</i>	
5.	<i>knp ngerasa kaya gempa mulu si kaya org gila tbtb kaget sendiri</i>	

The outcomes of this preprocessing phase will then proceed to the feature extraction stage, where all text data will be transformed into vector form and placed into an array. This extraction process will utilize the word2vec method to map words into vectors, including meanings and contextual understanding within documents. An example of the word2vec feature extraction results can be seen in [TABLE 3](#), which displays data attributes. Based on the table, the horizontal column 0 represents per-document data that has been transformed into vector form, producing 100 columns through this extraction process. Then, vertically from column 0 downward, there are the documents used in this study, resulting in 3000 rows corresponding to the amount of data used.

TABEL 3

The Results of feature extraction using Word2Vec

	0	1	2	3	99
0	0.152108	0.158517	0.160185	-0.00109	-0.142643
1	0.271955	-0.39221	0.134015	0.459004	-0.746475
2	0.174761	-0.08083	0.256982	0.045708	-0.339191
3	0.190098	0.319282	0.06177	0.052399	-0.468803
4	0.418278	0.265517	0.037131	-0.02466	-0.332364
5	0.24303	0.221987	-0.20709	0.026885	-0.322169
6	-0.00864	0.02625	-0.01969	-0.0156	-0.390515
7	0.535739	0.008953	-0.09168	-0.07068	-0.240261
8	0.408698	0.084984	-0.04851	-0.2428	-0.697299
9	-0.04564	0.259318	0.198453	-0.33768	-0.355599
10	-0.42915	-0.62537	-0.63726	-0.00339	-0.387902
11	-0.58377	0.181611	0.30517	-0.36076	-0.595670
12	0.035886	0.08114	-0.28632	-0.25829	-0.283361
13	0.275846	0.181911	0.236801	-0.04736	-0.30703
14	0.623301	0.17498	-0.22954	-0.0221	-0.178762
....
2999	-0.01347	-0.06362	0.108993	0.1169	-0.2425

After the preprocessing and feature extraction processes are complete, the data obtained from this extraction will be

divided into two schemes: training data and testing data. Training data is the data used for training, while testing data is used for prediction based on the trained data. The consideration for dividing the dataset in such a manner is to create a training set and provide the best model estimation [34].

In this study, the data is divided using an 80% training and 20% testing split, which will be implemented for testing both the LSTM and Bi-LSTM models. However, before proceeding with the data splitting process, the extracted data is first normalized using the standard scaler normalization technique. The Standard Scaler technique standardizes attributes by subtracting the mean from each value and dividing the result by the standard deviation of the attribute, resulting in a distribution with a mean of zero and a unit variance [35].

Subsequently, the parameters for word2vec used in this testing are as follows: 100 for the vector size, 100 for the number of epochs, a window size of 3, dropout rates of (0.1, 0.2, 0.3, 0.4), and learning rates of (0.001, 0.002, 0.003, 0.004).

The first testing phase will begin with the LSTM model. After testing with LSTM, the evaluation will continue with Bi-LSTM. In the LSTM method testing, the highest accuracy obtained from the experiments is 70.67% at epoch 30. Using the same parameters, the testing with the Bi-LSTM method yields the highest accuracy of 72.17%, also achieved at epoch 30. The obtained accuracies can be compared in [TABLE 4](#) and [FIGURE 4](#).

TABLE 4
Comparison of accuracy results

Method	Accuracy (%)	Recall (%)	Precision (%)	F1 – Score (%)
LSTM	70.67	70.67	70.16	70.82
Bi - LSTM	72.17	72.17	72.32	72.22

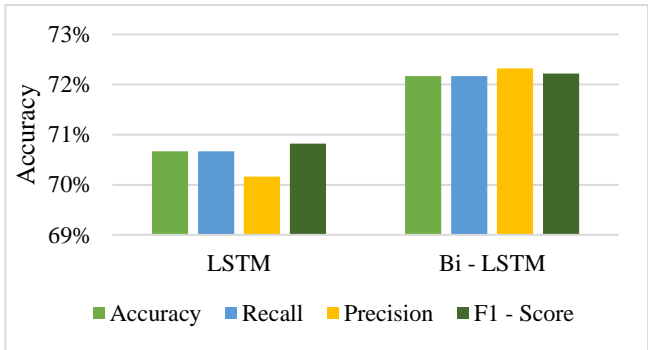


FIGURE 4. Comparison of Accuracy Results between Methods LSTM and Bi – LSTM

Based on the accuracy results obtained from [TABLE 4](#) and [FIGURE 4](#). It can be observed that the accuracy does not exhibit a significant difference, where the accuracy produced by LSTM is only 70.67%, while the accuracy yielded by Bi-LSTM is 72.17%. Based on the obtained

accuracy results, it can be concluded that the best classification ability with the word2vec feature extraction is achieved using the Bi-LSTM method.

IV. DISCUSSION

The evaluation results of the combined word2vec and LSTM model align with previous studies conducted by [23],[24],[26]. However, a fundamental difference between this study and prior research can be seen in the obtained accuracy. In the study by [26], an accuracy of 85.96% was achieved, while in this research, the accuracy achieved is only 70.67%. This difference can be attributed to variations in factors such as the dataset used and the parameter values applied to word2vec. In [26], the vector size parameters used were 100, 200, and 300, with dropout values of (0.2, 0.5, and 0.7), and learning rates of (0.0001 and 0.001). In contrast, this study employed vector dimensions up to 100, dropout rates of (0.1, 0.2, 0.3, 0.4), and learning rates of (0.001, 0.002, 0.003, 0.004). These differing parameter values are considered a potential reason for the decrease in achieved accuracy. Furthermore, the research on the combination of word2vec and Bi-LSTM aligns with studies conducted by [23], [25]. However, [25] utilized TF-IDF as the feature extraction method. The commonality between [25] and this study lies in both being text classification studies, where feature extraction is necessary prior to classification. Both studies also employ the Bi-LSTM classification method. The final results from [25] only present precision (91.54%), recall (92.82%), and F1-score (92.18%) values. In contrast, the present study yields significantly different results, including accuracy (72.17%), recall (72.17%), precision (72.32%), and F1-score (72.22%).

The underlying reasons for this notable discrepancy in accuracy are akin to the explanations provided earlier, influenced by parameters and data used. Based on the achieved accuracy results, predictive results per class for each method can also be found in [TABLE 5](#) dan [FIGURE 5](#) then, for the precision, recall, and F1-score of other classes are shown in [TABLE 6](#) and [FIGURE 6](#).

TABLE 5
Comparison of Confusion Matrix Results for Each Class

Method	Actual Class	Prediction		
		eyewitness	non-eyewitness	don't know
LSTM	eyewitness	138	39	23
	non-eyewitness	48	137	15
	don't know	24	18	158
Bi - LSTM	Eyewitness	139	46	15
	non-eyewitness	56	128	16
	don't know	27	16	157

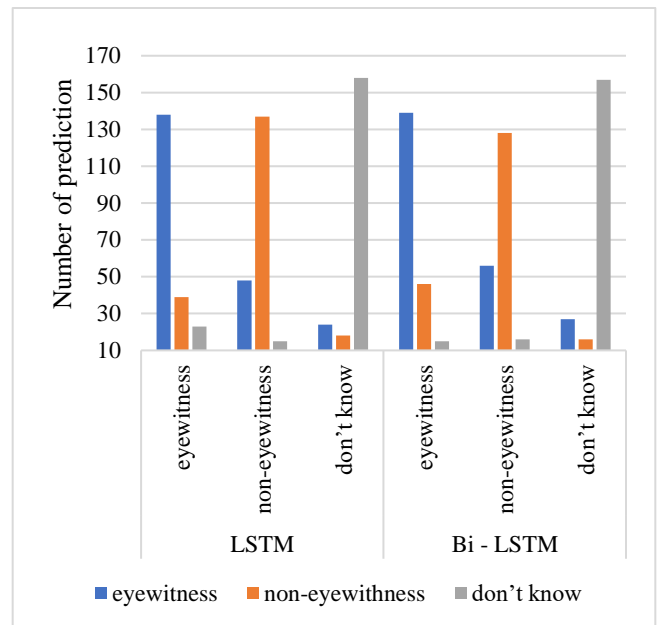


FIGURE 5. Comparison of Confusion Matrix Results for Each Class

TABEL 6
Comparison of Precision, Recall, and F1-Score Results for Each Class

Method	Result	Prediction		
		eyewitness	non-eyewitness	don't know
LSTM	Recall (%)	69	68.5	79
	Precision (%)	65.71	70.62	80.61
	F1 - Score (%)	67.32	69.54	79.8
Bi - LSTM	Recall (%)	69.5	64	78.5
	Precision (%)	62.61	67.37	83.51
	F1 - Score (%)	65.88	65.64	80.93

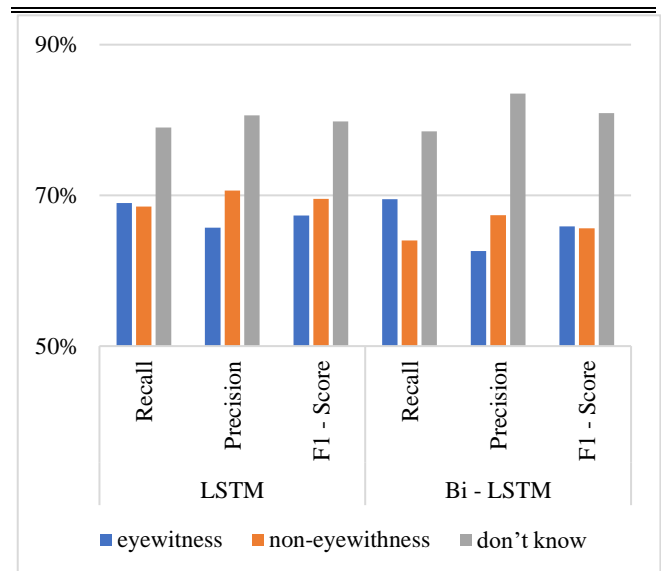


FIGURE 6. Comparison of Precision, Recall, and F1-Score Results for Each Class

The graphs and tables comparing prediction results, recall, precision, and F1-score per class show that Bi-LSTM outperforms LSTM. The accuracy for each class is a crucial factor that impacts the overall accuracy, as indicated in TABLE 4, where Bi-LSTM achieves higher accuracy than LSTM.

Various factors influence the significant difference in accuracy results. Different feature extraction methods are employed, leading to variations in the input data used for the Bi-LSTM model. Additionally, this study includes the standard scaler normalization method after the extraction process, which was not used in [25]. Similar to the LSTM model explanation, the results of the Bi-LSTM model are also influenced by the chosen parameters. The parameters used in this study are standard ones frequently used in many other research projects. Consequently, the study's weakness lies in the parameter value selection, as no optimization was performed to find the best parameters. While this study does not address this issue, it serves as a point for further investigation.

Furthermore, the findings from this study contribute to the knowledge by showcasing the results of the combination of word2vec and LSTM compared to the combination of word2vec and Bi-LSTM. The comparison between these combinations provides insight that the combination of word2vec and Bi-LSTM performs better than the combination of word2vec and LSTM. However, the study's weakness lies in the determination of parameter values. Specifically, the researchers randomly determined the parameter values for the word2vec feature extraction without prior testing to identify the optimal parameters for use. This approach can negatively impact the quality of the data results obtained and subsequently lead to inaccuracies in the classification results.

V. CONCLUSION

This study presents a discussion on the algorithms of deepening Recurrent Neural Networks (RNN), specifically the Long Short-Term Memory (LSTM) and Bidirectional LSTM (Bi-LSTM) methods. However, another technique is needed to transform the textual data into vectors using word2vec to assess the performance of these methods when implemented in text mining. The combination of word2vec and LSTM classification yielded an accuracy of 70.67% for classifying textual data related to earthquake disasters. Subsequently, using the same data, the combination of word2vec and Bi-LSTM classification resulted in an increased accuracy compared to the previous combination. This improvement amounted to 1.5%, achieving an accuracy of 72.17%.

Therefore, Bi-LSTM shows potential for further combination with word2vec, which could involve reconfiguring the parameter values. Adjusting the parameters is likely to produce different accuracy values. This presents an avenue for future researchers to explore parameter adjustments or other possible combinations of methods. The aim would be to achieve even better model performance.

VI. ACKNOWLEDGMENT

This research was supported by funding from the DRTPM Research Program of the Ministry of Education, Culture, Research, and Technology of Indonesia. Main Contract Number: 130/E5/PG.02.00.PL/2023, Derivative Contract Number: 576/UN8.2/PG/2023.

REFERENCES

- [1] M. Akter, D. Cumming, and S. Ji, "Natural disasters and market manipulation," *J. Bank. Financ.*, vol. 153, p. 106883, 2023, doi: 10.1016/j.jbankfin.2023.106883.
- [2] P. Daly, S. Ninglekhu, P. Hollenbach, J. W. McCaughey, D. Lallemand, and B. P. Horton, "Rebuilding historic urban neighborhoods after disasters: Balancing disaster risk reduction and heritage conservation after the 2015 earthquakes in Nepal," *Int. J. Disaster Risk Reduct.*, vol. 86, no. January, p. 103564, 2023, doi: 10.1016/j.ijdrr.2023.103564.
- [3] M. R. Faisal, I. Budiman, F. Abadi, M. Haikal, M. K. Delimayanti, and D. T. Nugraha, "Using Social Media Data to Monitor Natural Disaster: A Multi Dimension Convolutional Neural Network Approach with Word Embedding," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 6, pp. 1037–1046, 2022, doi: 10.29207/resti.v6i6.4525.
- [4] K. Zahra, M. Imran, and F. O. Ostermann, "Automatic identification of eyewitness messages on twitter during disasters," *Inf. Process. Manag.*, vol. 57, no. 1, p. 102107, 2020, doi: 10.1016/j.ipm.2019.102107.
- [5] K. Y. Firlia, M. R. Faisal, D. Kartini, R. A. Nugroho, and F. Abadi, "Analysis of New Features on the Performance of the Support Vector Machine Algorithm in Classification of Natural Disaster Messages," *Proc. - 2021 4th Int. Conf. Comput. Informatics Eng. IT-Based Digit. Ind. Innov. Welf. Soc. IC2IE 2021*, no. December, pp. 317–322, 2021, doi: 10.1109/IC2IE53219.2021.9649107.
- [6] M. Cormier and M. Cushman, "Innovation via social media – The importance of Twitter to science," *Res. Pract. Thromb. Haemost.*, vol. 5, no. 3, pp. 373–375, 2021, doi: 10.1002/rth2.12493.
- [7] C. J. Powers *et al.*, "Using artificial intelligence to identify emergency messages on social media during a natural disaster: A deep learning approach," *Int. J. Inf. Manag. Data Insights*, vol. 3, no. 1, p. 100164, 2023, doi: 10.1016/j.jjime.2023.100164.
- [8] M. Dou, Y. Wang, Y. Gu, S. Dong, M. Qiao, and Y. Deng, "Disaster damage assessment based on fine-grained topics in social media," *Comput. Geosci.*, vol. 156, no. March, p. 104893, 2021, doi: 10.1016/j.cageo.2021.104893.
- [9] T. Georgieva-Trifonova and M. Dechev, "Applying text mining methods to extracting information from news articles," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1031, no. 1, 2021, doi: 10.1088/1757-899X/1031/1/012054.
- [10] P. Savci and B. Das, "Comparison of pre-trained language models in terms of carbon emissions, time and accuracy in multi-label text classification using AutoML," *Heliyon*, vol. 9, no. 5, p. e15670, 2023, doi: 10.1016/j.heliyon.2023.e15670.
- [11] Nurmawiyana and K. A. Harvian, "Public sentiment towards face-to-face activities during the COVID-19 pandemic in Indonesia," *Procedia Comput. Sci.*, vol. 197, no. 2021, pp. 529–537, 2021, doi: 10.1016/j.procs.2021.12.170.
- [12] S. Kim, H. Park, and J. Lee, "Word2vec-based latent semantic analysis (W2V-LSA) for topic modeling: A study on blockchain technology trend analysis," *Expert Syst. Appl.*, vol. 152, 2020, doi: 10.1016/j.eswa.2020.113401.
- [13] F. Xue, X. Li, T. Zhang, and N. Hu, "Stock market reactions to the COVID-19 pandemic: The moderating role of corporate big data strategies based on Word2Vec," *Pacific Basin Financ. J.*, vol. 68, no. November 2020, p. 101608, 2021, doi: 10.1016/j.pacfin.2021.101608.
- [14] R. Agrawal and R. Goyal, "Developing bug severity prediction models using word2vec," *Int. J. Cogn. Comput. Eng.*, vol. 2, no. July, pp. 104–115, 2021, doi: 10.1016/j.ijcce.2021.08.001.
- [15] A. Ajitha, M. Goel, M. Assudani, S. Radhika, and S. Goel, "Design and development of Residential Sector Load Prediction model during

- COVID-19 Pandemic using LSTM based RNN,” *Electr. Power Syst. Res.*, vol. 212, no. October 2021, p. 108635, 2022, doi: 10.1016/j.epsr.2022.108635.
- [16] S. Song *et al.*, “Research on a working face gas concentration prediction model based on LASSO-RNN time series data,” *Heliyon*, vol. 9, no. 4, 2023, doi: 10.1016/j.heliyon.2023.e14864.
- [17] M. Muñoz-Organero, P. Callejo, and M. Á. Hombrados-Herrera, “A new RNN based machine learning model to forecast COVID-19 incidence, enhanced by the use of mobility data from the bike-sharing service in Madrid,” *Heliyon*, vol. 9, no. 6, p. e17625, 2023, doi: 10.1016/j.heliyon.2023.e17625.
- [18] Q. Kang, E. J. Chen, Z.-C. Li, H.-B. Luo, and Y. Liu, “Attention-based LSTM predictive model for the attitude and position of shield machine in tunneling,” *Undergr. Sp.*, vol. 13, pp. 335–350, 2023, doi: 10.1016/j.undsp.2023.05.006.
- [19] M. Khuntia and D. Gupta, “Indian News Headlines Classification using Word Embedding Techniques and LSTM Model,” *Procedia Comput. Sci.*, vol. 218, pp. 899–907, 2023, doi: 10.1016/j.procs.2023.01.070.
- [20] Z. Sun, R. Machlev, Q. Wang, J. Belikov, Y. Levron, and D. Baimel, “A public data-set for synchronous motor electrical faults diagnosis with CNN and LSTM reference classifiers,” *Energy AI*, vol. 14, no. January, p. 100274, 2023, doi: 10.1016/j.egyai.2023.100274.
- [21] U. B. Mahadevaswamy and P. Swathi, “Sentiment Analysis using Bidirectional LSTM Network,” *Procedia Comput. Sci.*, vol. 218, pp. 45–56, 2023, doi: 10.1016/j.procs.2022.12.400.
- [22] M. Arbane, R. Benlamri, Y. Brik, and A. D. Alahmar, “Social media-based COVID-19 sentiment classification model using Bi-LSTM,” *Expert Syst. Appl.*, vol. 212, no. November 2021, p. 118710, 2023, doi: 10.1016/j.eswa.2022.118710.
- [23] E. M. Alshari, A. Azman, S. Doraisamy, N. Mustapha, and M. Alksher, “Senti2vec: An effective feature extraction technique for sentiment analysis based on word2vec,” *Malaysian J. Comput. Sci.*, vol. 33, no. 3, pp. 240–251, 2020, doi: 10.22452/mjcs.vol33no3.5.
- [24] G. S. . Murthy, S. R. Allu, B. Andhavarapu, M. Bgadi, and M. Belusonti, “Text based Sentiment Analysis using Long Short Term Memory (LSTM),” *Int. J. Eng. Res. Technol.*, vol. 9, no. 05, pp. 299–303, 2020.
- [25] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, “Sentiment analysis of comment texts based on BiLSTM,” *IEEE Access*, vol. 7, pp. 51522–51532, 2019, doi: 10.1109/ACCESS.2019.2909919.
- [26] P. F. Muhammad, R. Kusumaningrum, and A. Wibowo, “Sentiment Analysis Using Word2vec and Long Short-Term Memory (LSTM) for Indonesian Hotel Reviews,” *Procedia Comput. Sci.*, vol. 179, no. 2020, pp. 728–735, 2021, doi: 10.1016/j.procs.2021.01.061.
- [27] M. R. Faisal, I. Budiman, F. Abadi, D. T. Nugrahadi, M. Haekal, and I. Sutedja, “Applying Features Based on Word Embedding Techniques to 1D CNN for Natural Disaster Messages Classification,” *2022 5th Int. Conf. Comput. Informatics Eng. IC2IE 2022*, no. December, pp. 192–197, 2022, doi: 10.1109/IC2IE56416.2022.9970188.
- [28] M. R. Faisal, I. Budiman, F. Abadi, M. Haekal, and D. T. Nugrahadi, “A comparison of word embedding-based extraction feature techniques and deep learning models of natural disaster messages classification,” *J. Comput. Sci. Inst.*, vol. 27, no. December 2022, pp. 145–153, 2023, doi: 10.35784/jcsi.3322.
- [29] M. R. Faisal, R. A. Nugroho, R. Ramadhani, F. Abadi, R. Herteno, and T. H. Saragih, “Natural disaster on twitter: Role of feature extraction method of word2vec and lexicon based for determining direct eyewitness,” *Trends Sci.*, vol. 18, no. 23, pp. 1–13, 2021, doi: 10.48048/tis.2021.680.
- [30] B. Jang, I. Kim, and J. W. Kim, “Word2vec convolutional neural networks for classification of news articles and tweets,” *PLoS One*, vol. 14, no. 8, pp. 1–20, 2019, doi: 10.1371/journal.pone.0220976.
- [31] D. Suhartono, K. Purwandari, N. H. Jeremy, S. Philip, P. Arisaputra, and I. H. Parmonangan, “Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews,” *Procedia Comput. Sci.*, vol. 216, no. 2022, pp. 664–671, 2023, doi: 10.1016/j.procs.2022.12.182.
- [32] D. Sunitha, R. K. Patra, N. V. Babu, A. Suresh, and S. C. Gupta, “Twitter sentiment analysis using ensemble based deep learning model towards COVID-19 in India and European countries,” *Pattern Recognit. Lett.*, vol. 158, pp. 164–170, 2022, doi: 10.1016/j.patrec.2022.04.027.

- [33] V. Matoušek, “Application of LSTM Neural Networks in Language Modelling,” *Univ. West Bohemia, Fac. Appl. Sci. Dep. Cybern. Univerzity 22, Plzeň, Czech rep.*, no. June 2018, 2013, doi: 10.1007/978-3-642-40585-3.
- [34] V. R. Joseph and A. Vakayil, “SPlit: An Optimal Method for Data Splitting,” *Technometrics*, vol. 64, no. 2, pp. 166–176, 2022, doi: 10.1080/00401706.2021.1921037.
- [35] L. B. V. de Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, “The choice of scaling technique matters for classification performance,” *Appl. Soft Comput.*, vol. 133, pp. 1–37, 2023, doi: 10.1016/j.asoc.2022.109924.

Bibliography



monitoring of natural disasters and early warning system.



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 Mangkurat. His research interests encompass Data Science, Software Engineering, and Bioinformatics.



as a manager of a scientific journal at his workplace as a lecturer. The research field he is involved in is Data Science.



master's degree at Monash University, Australia in 2012. And her latest education is a doctorate degree in Bioinformatics at Kanazawa University, Japan, which was completed in 2022. The research fields she focuses on are Data Science and Bioinformatics.

Rahmi Yunida originated in Jelaput, Barito Kuala, South Kalimantan. Since 2018, she has pursued her academic endeavors as a student of Computer Science Department at Universitas Lambung Mangkurat. Her current area of research lies within the realm of data science. Additionally, her final project entailed conducting research that centered around the classification of text within messages sourced from social media. The purpose of this research endeavor was to facilitate the

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 completing his bachelor's program, he gained experience as a training trainer in the field of information technology and software development. Since 2008,

Muliadi is a lecturer in the Department of Computer Science, Lambung Mangkurat University. His research interest is focused on Data Science. Before becoming a lecturer, he completed his bachelor's degree at STMIK Akakom Yogyakarta in 2004. After that, he pursued a master's degree in Computer Science at Gajah Mada University in 2009. Currently, he also serves

Fatma Indriani is a lecturer in the Department of Computer Science, Lambung Mangkurat University. Her research interest is focused on Data Science. Before becoming a lecturer, she completed her undergraduate program in the Informatics Department at Bandung Institute of Technology. In 2008, she started working as a lecturer in the Computer Science department at Lambung Mangkurat University. She then completed her



Friska Abadi finished her bachelor's degree in Computer Science from Universitas Lambung Mangkurat in 2011. Subsequently, in 2016, she obtained her master's degree from the Department of Informatics at STIMIK Amikom, Yogyakarta. Following that, she joined Universitas Lambung Mangkurat as a lecturer in Computer Science. Currently, she holds the position of head of the software engineering laboratory. Her current area

of research revolves around software engineering.



Irwan Budiman successfully finished his bachelor's degree in the informatics department at the Islamic University of Indonesia. Subsequently, he assumed the role of a lecturer in Computer Science at Universitas Lambung Mangkurat starting from 2008. Additionally, in 2010, he pursued a master's degree in Information Systems at Diponegoro University. Currently, Irwan Budiman holds the position of chair for the computer science

study program at Universitas Lambung Mangkurat. His area of research expertise lies in Data Science.



Septyan Eka Prastya is a lecturer in the Department of Information Technology, Sari Mulia University. He completed his undergraduate education in Computer Science, Lambung Mangkurat University in 2013. He also completed his master's education at the Islamic University of Indonesia in 2016. Currently, he is conducting research in the field of software engineering and

machine learning.