TELKOMNIKA Telecommunication Computing Electronics and Control
Vol. 22, No. 5, October 2024, pp. 1196~1208
ISSN: 1693-6930, DOI: 10.12928/TELKOMNIKA.v22i5.26091

3D word embedding vector feature extraction and hybrid CNN-LSTM for natural disaster reports identification

Mohammad Reza Faisal[1], Dodon Turianto Nugrahadi[1], Irwan Budiman[1], Muliadi[1], Mera Kartika Delimayanti[2], Septyan Eka Prastya[3], Imam Tahyudin[4]

[1]Department of Computer Science, Faculty of Mathematics and Natural Science, Lambung Mangkurat University, Banjarbaru, Indonesia
[2]Department of Computer and Informatics Engineering, Politeknik Negeri Jakarta, Depok, Indonesia
[3]Department of Information Technology, Faculty of Science and Technology, Sari Mulia University, Banjarmasin, Indonesia
[4]Magister of Computer Science Department, Faculty of Computer Science, Universitas Amikom Purwokerto, Purwokerto, Indonesia

Article Info

Article history:

Received Feb 18, 2024
Revised Jun 26, 2024
Accepted Jul 12, 2024

Keywords:

Convolutional neural network
Feature extraction
Long short-term memory
Natural disaster
Word embedding

ABSTRACT

Social media contain various information, such as natural disaster reports. Artificial intelligence is used to identify reports from eyewitnesses early for disaster warning systems. The artificial intelligence system includes a text classification model with feature extraction and classification algorithms. Word embedding-based feature extraction is widely used for 1-dimensional (1D) and 2-dimensional (2D) data, suitable for traditional or deep learning algorithms. However, applying feature extraction to 3-dimensional (3D) data for text classification is limited. Previous studies focused on word embedding for 1D, 2D, and 3D outputs with convolutional neural network (CNN). Yet, using 3D data and CNN did not perform well. Despite using CNN and 3D variants, identifying natural disaster reports remains below 80% accuracy. This research aims to improve identifying earthquakes, floods, and forest fires with 3D data and hybrid CNN long short-term memory (LSTM). The study found models with accuracies of 83.38%, 83.72%, and 89.03% for each disaster type. Hybrid CNN LSTM significantly enhanced identification compared to CNN alone, supported by statistical tests with P value less than 0.0001.

Corresponding Author:

Mohammad Reza Faisal
Department of Computer Science, Faculty of Mathematics and Natural Science
Lambung Mangkurat University
A. Yani St., KM 36, Banjarbaru, South Kalimantan, Indonesia
Email: reza.faisal@ulm.ac.id

1. INTRODUCTION

Nearly every smartphone user employs social networking platforms to communicate and exchange knowledge about daily endeavours. Consequently, social media platforms are brimming with a wealth of information, including instances when users undergo the effects of an unforeseen calamity. The disclosure of natural disasters on social media can be harnessed to ascertain the exact time and location of such incidents. The automated monitoring of natural disasters through social media channels can be facilitated using artificial intelligence. Artificial intelligence is employed to identify disaster reports, serving as an early warning and monitoring system [1], [2].

Research on natural disaster report identification is equivalent to research on text classification. This inquiry encompasses two primary phases. The initial phase entails the feature extraction to transform textual information into structured data. Subsequently, the organized data is subjected to the manipulation of classification algorithms to construct models that can automatically identify natural disaster reports. The scholars undertook research endeavours to advance both pivotal methodologies to enhance the model's performance for identifying natural disaster reports.

Research on using text classification for natural disaster identification in Indonesia via using datasets written in the Indonesian language [3]–[7]. These scholarly inquiries employed datasets of accounts of natural disasters, such as floods, earthquakes, and forest fires. The three natural disasters are recurrent incidents within numerous nations, including Indonesia.

Feature extraction methods based on the representation of vector spaces and classification algorithms based on machine learning have been extensively employed in various research on text classification in general. This technique is also utilized to identify messages related to flood disasters in the Indonesian language [3], [4]. Both of these studies employed feature extraction methods using unigram and term frequency-inverse document frequency (TF-IDF) as well as support vector machine (SVM) algorithms. The identification performance of these two studies was 77.87% [4] and 77.50% [5]. This extraction method can process messages containing varying words and produce structured data with a one-dimensional (1D) and an equal number of features. The resulting 1D data solely retains information about the frequency of occurrence of the word or term within the sentence. However, it does not retain information regarding the semantic interpretation of words and the syntactic order of words within sentences.

In recent times, the prevalent word embedding-based feature extraction technique is employed because the resultant structured data contains valuable word sense information as well as word arrangement. For Indonesian natural disaster message recognition research, three well-known word embedding techniques encompass Word2vec, FastText, and glove. In this particular investigation, 1D structured data was generated employing the 1D convolutional neural network (CNN) classification algorithm for flood, earthquake, and forest fire disasters, yielding accuracy outcomes of 74.00%, 74.33%, and 81.14% [5]. Research on the identification of natural disasters that also utilize 1D structured data relies on word2vec-based feature extraction to tackle earthquake scenarios [7]. In this study, 1D data was subjected to processing using long short-term memory network (LSTM) and bidirectional LSTM (Bi-LSTM) algorithms. The performance of the identification models derived from these techniques amounted to 70.67% and 72.17%, respectively.

These three feature extraction techniques based on word embeddings can also generate structured data in two-dimensions (2D). The detection of floods, earthquakes, and forest fires utilizing the CNN algorithm in 2D yielded an accuracy rate of 78.33%, 78.33%, and 81.97% [6]. The outcomes achieved through this investigation surpass those reported in the studies.

In research Faisal *et al*. [6], a study was also carried out on feature extraction techniques that produce three-dimensional (3D) data with two different formation methods, namely 3D type 1 and 3D type 2. The 3D Type 1 data formation method produces flood disaster identification accuracy of 72.33%, earthquakes is 77.00%, and forest fires is 81.55%. Meanwhile, 3D type 2 data produces flood, earthquake, and forest fire with identification accuracy of 71.00%, 74.33%, and 81.15%, respectively. These two types of 3D data are processed using the 2D CNN classification algorithm. In this research, no experiments have been carried out using 3D CNN. From the study of word embedding-based feature extraction techniques, it is known that it has the potential to form structured data with various dimensions.

Those methods that are explained above constructs word embedding vectors based on the number of words present in the sentence. The count of words in each message varies thus this method will also generate a distinct count of characteristics. To ensure that the resulting structured data possesses the same count of characteristics, the word padding method must be employed in conjunction with the execution of the extraction method. The word padding method utilized in [5], [6] truncates the length of a word in a message based on statistical computations of mean, median, and mode. Both investigations indicated that mean-based word padding yielded the most optimal classification performance. In both studies, word padding with maximum word length was not employed because it led to the production of sparse data in the structured data. Sparse data arises due to messages that solely consist of 2-3 words, while other messages encompass numerous words. The presence of sparse data can impact the performance of classification algorithms. Thus, studies [5], [6] refrain from its usage.

The examination [5]–[7] utilized deep learning-based classification algorithms. CNN possesses benefits in the manipulation of data with diverse dimensions and is also capable of extracting significant characteristics to enhance classification performance [8]. Conversely, LSTM and its variations are solely competent in processing 1D data, yet they exhibit proficiency in the comprehension of word sequence information [9], [10]. In text classification research, it has been demonstrated that the amalgamation of CNN and LSTM capabilities into a hybrid CNN-LSTM renders these two algorithms mutually reinforcing, leading to additional enhancements in classification performance [11]–[13]. Within these investigations, the utilization of the hybrid 1D CNN-LSTM [11] and the hybrid 2D CNN LSTM [12], [13] have been undertaken to address the issue of general text classification. Nevertheless, no comprehensive inquiry has been conducted regarding the handling of 3D type 1 and 3D type 2 structured data through the deployment of the hybrid 2D CNN-LSTM and the hybrid 3D CNN-LSTM in general text classification research, and more specifically, in the realm of identifying messages of natural disasters.

In this scholarly article, we put forth a proposition for a model designed to identify messages of natural calamities. This model is constructed by employing the 3D word embedding vector feature extraction technique, in conjunction with a hybrid CNN LSTM architecture. Furthermore, we shall demonstrate how implementing these techniques can enhance the performance of message identification. Additionally, we shall explore the formation of multi-dimensional structured data, specifically focusing on the 3D data. The primary contributions of this paper encompass the following:

- A novel framework for natural disaster identification from social media platforms, achieved by amalgamating feature extraction through the use of a 3D word embedding vector and a hybrid CNN-LSTM model.
- The confluence of three word embedding methodologies, namely Word2Vec, FastText, and glove, is employed to establish a well-structured dataset that the hybrid CNN-LSTM architecture can effectively process.
- Developing a CNN-LSTM hybrid framework that exhibits commendable performance even in sparse data scenarios.
- To the best of our knowledge, we are the first to research 3D word embedding vector feature extraction and hybrid CNN LSTM for natural disaster reports identification using Indonesian language datasets. We believe that our proposed concept will advance text classification research, especially in research of disaster report identification in Indonesia.

The rest of this paper is structured as follows. Section 2 pertains to the research that substantiates the research we engage in. Section 3 delineates the proposed method. Section 4 concerns the result and discussion. Section 5 concerns the conclusion and our future research.

2. RELATED RESEARCH

2.1. Feature extraction

In text classification research, feature extraction converts unstructured text data into structured data [14]. In this section, word embedding-based feature extraction techniques are explained. The word embedding technique is an unsupervised learning technique that processes text data. The result is that each word will be given a value as a vector. Each word vector usually contains one hundred numerical values, which are representations of the word's meaning so that words with the same meaning will have vectors whose values and directions are close together [15], [16]. Several word embedding techniques are known, namely Word2Vec, fastText, and glove.

A natural disaster report is a sentence consisting of one or several words. The word embedding-based feature extraction technique creates a sentence vector by compiling vectors from each word in a sentence. This subsection explains several ways to create sentence vectors from the research that has been carried out.

Technique 1 (T1), which teaches how to arrange word vectors from sentences, can be seen in Figure 1. Sentence vectors are formed by adding up the word vectors of each word in a sentence. The resulting sentence vector is in the form of 1D data, which contains information about the sentence's meaning. Using this method, the 1D data output will have the same number of features even though the input sentences have different words. However, this method eliminates word order information in the sentence.

An illustration of technique 2 (T2), the second method, can be seen in Figure 2. This second method shows that sentence vectors are created by arranging each word vector using concatenation [5]. The sentence vector produced in this way produces 1D data, which contains information on word meaning and word order in the sentence.

1D data produced by the first and second methods of feature extraction techniques can be processed by machine learning classification algorithms such as SVM, random forest, Naïve Bayes, and others. The 1D data can also be processed by deep learning-based algorithms such as 1D CNN and LSTM. To identify flood disaster messages, this feature extraction method has been combined with the 1D CNN classification algorithm [5]. In this research, the word embedding techniques used as the basis for feature extraction techniques are Word2Vec, fastText, and glove. This research shows that the Wor2Vec technique can provide better performance than the other two word embedding techniques. This research also combines the feature extraction results from the three word embedding techniques. The performance of this combination is better than the performance of each word embedding technique. However, the merging performance is not better than the model built using vector space representation-based feature extraction and the SVM classification algorithm [3], [4].

Technique 3 (T3), a word embedding-based feature extraction technique, can also create 2D structured data, as shown in Figure 3. This method arranges each word vector in a stacked manner so that structured data is produced as a matrix [6]. This 2D structured data can only be processed by deep learning-based classification algorithms such as CNN. Research by Faisal *et al*. [6] conducted a study on the three word embedding techniques as was carried out in research on the implementation of 1D data and 1D CNN [5]. The cases of natural disasters in this study were floods, earthquakes and forest fires. The performance of identifying flood disaster messages using 2D data and 2D CNN is better than the performance carried out in research [3]–[5].

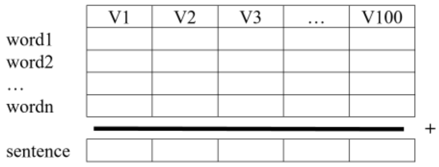

Figure 1. Sentence vector as 1D structured data from a summation of word vectors

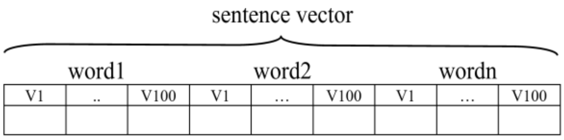

Figure 2. Sentence vector as 1D structured data from the concatenation of word vectors

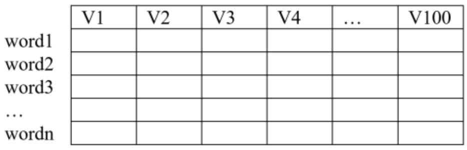

Figure 3. 2D structured data using word embedding vectors

The last way is to construct a vector statement in the configuration of 3D data. In order to generate 3D structured data, it is necessary to employ three word embedding methodologies. In antecedent investigations [6], a blend of Word2Vec, FastText, and glove methodologies was employed. This 3D data can be fashioned via two different approaches.

Technique 4 (T4), the first method of forming 3D data is shown in Figure 4. This method is inspired by feature extraction from image data, which consists of 3 layers of RGB color channels [17], [18]. In research Faisal *et al*. [6], 3D data was processed by the 2D CNN algorithm. The average performance of models built with 3D data and 2D CNN is the same as models built with 2D data and 2D CNN.

Technique 5 (T5), the second way of forming 3D data is by following the formation of structured data from image frames in video [19], [20]. This method of forming 3D data can be seen in Figure 5 The first layer of this 3D data is a word vector from the first word word1, which consists of a combination of three word vectors from three word embedding techniques. Next, create a word2 layer for the second word up to a wordn layer for the last word in the same way. The investigation [6] employed 3D data fashioned in this manner to be subjected to processing using 2D CNN. The efficacy of this framework is substandard in comparison to the efficacy of alternative models mentioned earlier.

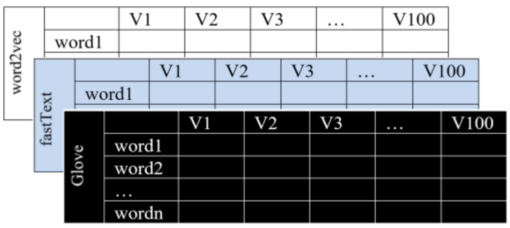

Figure 4. 3D structured data from three layers of word embedding vectors

Figure 5. 3D structured data from N layers of three word embedding vectors

2.2. Word padding

As explained above, T1 produces 1D data output with the same number of features even though given input sentences contain different numbers of words. As for T2 to T5, generate a sentence vector by constructing a word vector based on the words present in each sentence. These methodologies cause the length of features to vary depending on the number of words in the input sentence. For instance, a sentence1 sentence input consisting of three words would differ from a sentence2 sentence input comprised of five words. An illustration of the 2D data for this particular example can be observed in Figure 6. Figure 6(a) serve as an exemplification of sentance1, while Figure 6(b) exemplifies sentence2.

	V1	V2	V3	V4	…	V100
word1	0.598	0.519	0.667	0.953	0.573	0.007
word2	0.080	0.513	0.236	0.318	0.402	0.800
word3	0.818	0.716	0.229	0.735	0.514	0.373

(a)

	V1	V2	V3	V4	…	V100
word1	0.773	0.967	0.548	0.356	0.940	0.144
word2	0.152	0.530	0.213	0.642	0.810	0.909
word3	0.932	0.105	0.463	0.299	0.155	0.352
word4	0.186	0.905	0.909	0.249	0.839	0.551
word5	0.863	0.623	0.178	0.694	0.718	0.245

(b)

Figure 6. 2D data from two sentences with different numbers of words in each sentence; (a) sentence1 and (b) sentence2

Word padding is employed to ensure that the technique of feature extraction based on word embedding yields outcomes of equivalent dimensionality. This is accomplished by stipulating the desired quantity of words to be utilized. In the instance portrayed in Figure 6, if we specify that the number of words employed is 4, subsequently, the 2D data of sentance1 and sentence2 must be transformed into four rows. Consequently, the outcome can be observed as depicted in Figure 7. In Figure 7(a) sentence1 originally possessed three words along with a line of word vectors containing the numerical value of 0. The objective is to generate 2D data with four lines. Conversely, Figure 7(b) exhibits sentence2, which originally contained five truncated word vectors to yield 2D data with four lines.

	V1	V2	V3	V4	…	V100
word1	0.598	0.519	0.667	0.953	0.573	0.007
word2	0.080	0.513	0.236	0.318	0.402	0.800
word3	0.818	0.716	0.229	0.735	0.514	0.373
	0	0	0	0	0	0

(a)

	V1	V2	V3	V4	…	V100
word1	0.773	0.967	0.548	0.356	0.940	0.144
word2	0.152	0.530	0.213	0.642	0.810	0.909
word3	0.932	0.105	0.463	0.299	0.155	0.352
word4	0.186	0.905	0.909	0.249	0.839	0.551

(b)

Figure 7. Result of word padding by using number of words is four; (a) sentence1 and (b) sentence2

The determination of the quantity of words employed in the word padding technique can be precisely specified, as demonstrated in the example above. However, in the study [6], the number of words was determined based on statistical computations involving the mean, median, and mode. The utmost efficacy of the outcomes obtained from this study was observed when employing word padding with word counts predicated on the mean values.

2.3. Classification algorithms

T1 and T2 employ supplementary methods to derive 1D structured data. 1D data can be input for machine learning-based classification algorithms, such as SVM, K-nearest neighbors (KNN), Naïve Bayes, and random forest. 1D data can also be processed by deep learning-based algorithms, such as deep neural networks, LSTM, or 1D CNN. Regarding T3-T5, the techniques for extracting features result in 2D and 3D structured data. These two data types can be processed directly through CNN classification algorithms. The architecture and general workings of CNN can be seen in Figure 8. CNN is divided into two parts, namely feature learning and classification.

Feature learning is the ability of a model to extract important features from input automatically. This extraction operation begins with a convolution operation carried out by the convolution layer. This operation involves a kernel (filter) of a specific size that is shifted to cover the input data with specific steps [21], [22].

How the kernel (filter) moves depends on the dimensions of the input data and the type of convolutional layer used. In convolutional layer 1D, the kernel moves along the 1D data that is input [22]. The 2D convolution layer kernel is shifted in 2 dimensions by traversing the height and width directions in the 2D input data [22]. 2D convolution layers can also be used on 3D data. The kernel is 2D and only moves in 2 dimensions. The 3D convolution layer, kernel is shifted across the volume in the height, width, and depth directions [23].

The result of the convolutional operation is a feature map. The ReLU activation function is applied to each value in the feature maps. Next, the pooling layer reduces the spatial dimensions of the feature map. This feature map is the output of the feature learning process which contains important features [21], [22].

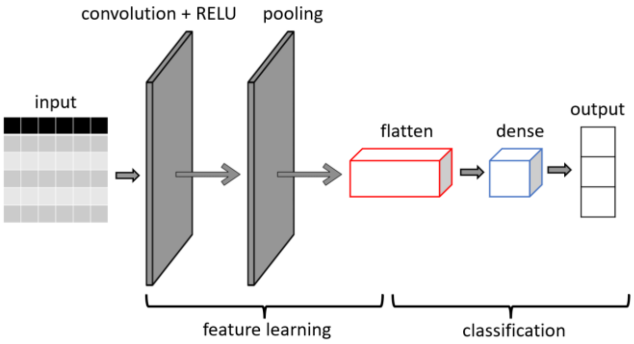

Figure 8. Common CNN architecture

The classification begins by converting the feature map into a 1D vector by a flatten layer. Then, the vector is received by the dense layer to be processed with linear and non-linear operations. The output layer produces predictions [21], [22].

CNNs have been used for various studies. 1D CNN is generally used in signal processing research cases such as electrocardiogram (ECG) classification, speech recognition, and vibration-based structural damage detection [22]. Implementing the 1D CNN algorithm in the case of text classification has been carried out to identify natural disaster messages and COVID-19 status messages [5].

The investigation's focal point was on comparing performance when utilizing various word padding and word embedding methodologies at the feature extraction phase [5]. In this particular research, the architecture of 1D CNN was implemented employing four distinct channels. The outcomes generated from this exploration offers indications that the effectiveness of identifying messages related to natural disasters reaches its peak when utilizing a convergence of all three word embedding methodologies (all) and word counts derived from mean calculations of word padding methodologies. The research conducted by the author concentrated on the impact of the count of channels on the architecture of 1D CNN and the methodologies of word embedding on the performance classification model [24]. The outcome explicitly demonstrates that the performance is superior or comparable to the count of other channels by employing solely one channel. Whereas the classification performance is at its finest when utilizing FastText.

Generally, 2D CNN and 3D CNN are commonly employed for pattern recognition on images and videos [22], [23]. In the domain of text classification, implementations of 2D CNN are also widely utilized [6], [25]. For instance, a study focused on identifying natural disasters employed a 2D CNN architecture [6]. In this particular study, the feature extraction T3 was employed. The word embedding and word padding techniques utilized in this research are identical to those employed in the investigation conducted in [5].

Furthermore, the 2D CNN employed in this study was configured to utilize 4 channels. In this research, we also conducted experiments with 3D data resulting from T4 and T5 feature extraction with the 2D CNN classification algorithm. The results show that the performance of 2D CNN with 2D data input works better than the performance of 2D CNN with 3D data input. In this research, classification performance has not been carried out using 3D input data and 3D CNN.

Currently, classification research uses a combination of CNN and LSTM algorithms [15], [26]–[28]. The studies [26], [27] use 1D input to be processed by the 1D CNN layer. Then, the output of that layer becomes the input of the LSTM layer. 1D CNN LSTM is used to solve text classification cases [26]. In this study, 1D data was created using word embedding techniques, as shown in T2. Meanwhile, in [27] 1D CNN LSTM was implemented in the case of spectral data classification to determine different physical states of pasta products.

2D CNN LSTM implementation was conducted on investigations [13], [29] for text classification. Both of these examinations formed 2D data utilizing T3. The investigation [13] examined 2D data established by diverse word embedding techniques, particularly Word2Vec, glove, and FastText. The investigation [29] only conducted a study employing a word embedding technique, Word2Vec.

Investigations [28], [30] utilized 3D CNN LSTM to recognize human motion from video facts. Establishment of 3D facts by structuring image frames as displayed in T5. We have not encountered a 3D execution of CNN LSTM on the text classification research.

The hybrid CNN LSTM architectures employed in the research [13], [26]-[30] can be delineated in general terms as displayed in Figure 9. Those studies indicated that the hybrid CNN LSTM algorithm could enhance classification efficacy in contrast to solely utilizing CNN or LSTM. These hybrid algorithms supplement one another with their respective benefits. The CNN layer handles data inputs comprising sequence information, which is efficacious in feature extraction, recognizing local patterns, and allowing dimension reduction. The outcome of the CNN layer is subsequently processed by an LSTM layer suitable for processing sequential data and comprehending the long-term context.

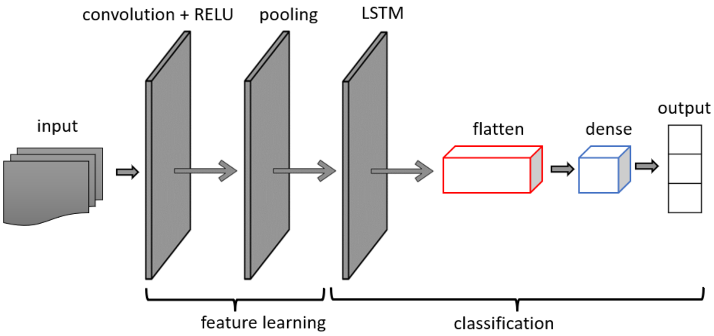

Figure 9. General hybrid CNN LSTM architecture

3. METHOD

3.1. Data and proposed method

In this segment, we delineate the phases of the proposed approach. Figure 10 illustrates the procedure executed in this research. Table 1 shows details of the dataset containing three natural disaster messages used in this research. These natural disaster messages have been used in previous studies, namely earthquakes [5], [6], floods [3], [4], and forest fires [5], [6]. All of these natural disaster messages were taken from X's social media.

Table 1. Dataset

	Class label		
Dataset	Eyewitness	Non-eyewitness	Don't know
Earthquakes	1000	1000	1000
Floods	1000	1000	1000
Forest Fires	1000	1000	1000

Natural disaster messages are divided into three class labels, namely eyewitness, non-eyewitness, and don't know. Eyewitness category messages contain natural disaster keywords posted by users who directly witnessed natural disasters. Non-eyewitness category messages report natural disasters but are not posted by eyewitnesses. Messages in the don't know category are messages that contain the keywords natural disasters, but their meaning is not about natural disasters.

The next stage is to carry out text normalization. At this stage, the text data for natural disaster messages is cleaned by removing multiple spaces, punctuation, numbers and non-alphanumeric characters. Then, the clean text data is processed by the word padding technique using the number of words based on calculating the maximum number of words (max).

This research used Indonesian Wikipedia text data to create three word embedding models (Word2Vec, fastText, and glove). These models function as a reference dictionary to convert every word in a natural disaster message into a word vector. The generation of 1D data employs the T2 technique, while the creation of 2D data utilizes the T3 technique. The establishment of 1D and 2D data in this investigation comprised two categories, specifically: i) data generated through a word embedding technique and ii) data formed from the amalgamation of three word embedding techniques. The integration of these three techniques is accomplished using performing concatenation. 3D data is produced by combining the T4 and T5 techniques with three word embedding techniques.

The structured data is devided into 80% training and 20% testing datasets. Each classification algorithm trains by utilizing a training dataset. The hybrid CNN LSTM architecture employed in this investigation is depicted in Figure 10. Figure 10(a) represents an architecture employed to construct 1D and 2D data using a solitary word embedding technique, followed by the execution of the training and testing processes. Figure 10(b) illustrates the architecture employed to generate 1D and 2D data from the outcomes of the fusion of three word embedding techniques. This particular structure is also utilized to produce 3D data using the T4 and T5 techniques. The N values in Figure 10 denote 1, 2, and 3, respectively, signifying the dimensional values of the convolution and maximum polling layers. The last stage entails assessing the performance of every model contrived in the evaluation. The quantification of model performance is achieved using accuracy.

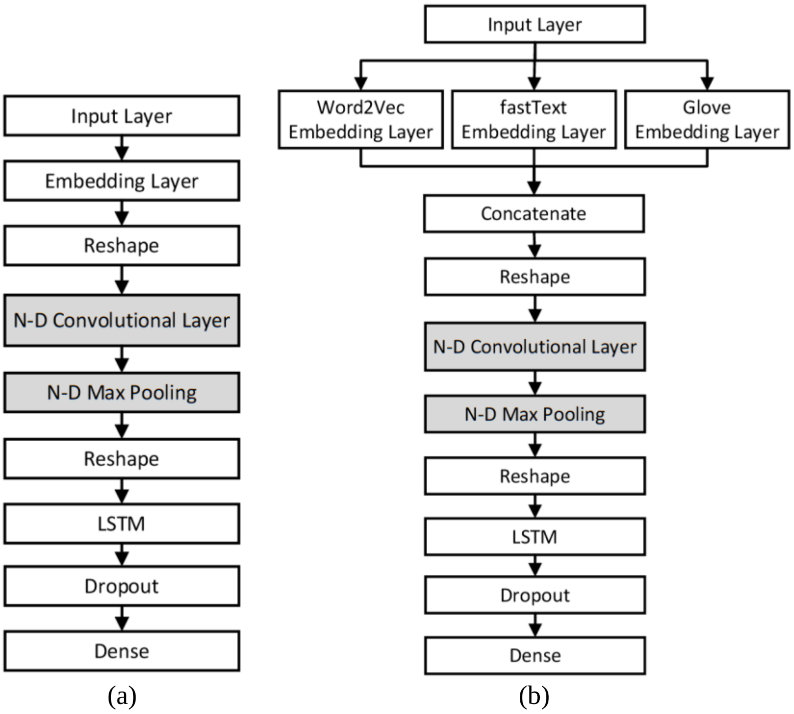

Figure 10. Proposed hybrid CNN LSTM architecture; (a) with a word embedding technique and (b) with three embedding techniques

3.2. Experiment settings

This section exhibits the configurations and parameters employed in this investigation. The concepts and architectures are realised by employing the Python language and its auxiliary libraries. To fabricate a word embedding model, the Indonesian Wikipedia text data is utilised. The text data can be download by visit this link https://dumps.wikimedia.org/idwiki/latest/. Word2Vec and FastText models are fabricated by using the Python programming language. The glove model was constructed employing the C programming language.

The outcome of this procedure is three word embedding model. Every model encompasses an index of words derived from the text data of Wikipedia. Each word is depicted by a vector comprising 100 numerical values. The method employed for word padding in this research was grounded on the utmost count of words in natural disaster messages. The maximum count for earthquakes, floods, and forest fires is 52, 47, and 57, respectively.

At the feature extraction phase, structured data is produced by employing three methods of word embedding and four variations of data dimensions. At this phase, some structured data are generated. Each structured data is divided into data train and test data. The data train derived from each structured data type is subjected to a suitable classification algorithm. Table 2 shows the progression amidst structured data and a classification algorithm in this research. The hybrid CNN LSTM architecture in this research employs the Python programming language with the TensorFlow library. The configurations and parameters employed primarily use the default values of each function.

Table 2. Experiment result

No	Dataset	Feature extraction	Dimension	Word embedding	Features	Classification algorithm	Accuracy (%)
1	Earthquakes	T2	1D	Word2Vec	5200	1D CNN LSTM	80.83
2				fastText	5200		80.27
3				Glove	5200		81.77
4				All	15600		81.11
5		T3	2D	Word2Vec	52×100	2D CNN LSTM	80.16
6				fastText	52×100		79.66
7				Glove	52×100		77.83
8				All	156×100		79.72
9		T4	3D Type 1	All	52×100×3		74.72
10		T5	3D Type 2	All	100×3×52		83.22
11		T4	3D Type 1	All	52×100×3	3D CNN LSTM	79.94
12		T5	3D Type 2	All	100×3×52		83.38
13	Floods	T2	1D	Word2Vec	4700	1D CNN LSTM	82.66
14				fastText	4700		82.83
15				Glove	4700		80.27
16				All	14100		81.72
17		T3	2D	Word2Vec	47×100	2D CNN LSTM	80.61
18				fastText	47×100		80.77
19				Glove	47×100		78.83
20				All	141×100		81.27
21		T4	3D Type 1	All	47×100×3		76.22
22		T5	3D Type 2	All	100×3×47		82.27
23		T4	3D Type 1	All	47×100×3	3D CNN LSTM	80.33
24		T5	3D Type 2	All	100×3×47		83.72
25	Forest fires	T2	1D	Word2Vec	5700	1D CNN LSTM	85.76
26				fastText	5700		86.73
27				Glove	5700		85.92
28				All	17100		88.33
29		T3	2D	Word2Vec	57×100	2D CNN LSTM	85.76
30				fastText	57×100		86.14
31				Glove	57×100		85.23
32				All	171×100		85.07
33		T4	3D Type 1	All	57×100×3		83.78
34		T5	3D Type 2	All	100×3×57		86.94
35		T4	3D Type 1	All	57×100×3	3D CNN LSTM	87.15
36		T5	3D Type 2	All	100×3×57		89.03

4. RESULTS AND DISCUSSION

This research employed a total of 12 models for each dataset, thereby yielding a total of 36 experiments. The performance of each model is observable in the accuracy column of Table 2. The models that can offer the best performance are models in the 12, 24, and 36 lines with performance that are 83.72%, 89.03%, and 83.38%. These three models possess in common: i) T5 feature extraction technique that generates 3D structured data type 2 and ii) hybrid 3D CNN LSTM classification algorithm.

To scrutinize the performance of the 12 models was accomplished by computing the mean performance of each model on all three datasets. A juxtaposition of the mean performance of each model can be observed in Figure 11. We additionally employed 1D CNN LSTM and 2D CNN LSTM classification algorithms to handle 1D and 2D data derived from the amalgamation of three word embedding methodologies (All). In the 1D CNN LSTM model, the utilization of 1D All data resulted in an improvement of the model's performance by 83.72% as compared to models solely reliant on feature extraction data predicated on a word embedding technique, albeit the disparity was not significant. Employing 2D CNN LSTM models and solely relying on 2D all data yields comparable performance to models constructed using solely Word2Vec and FastText based 2D data. This experiment showcases the potential for ameliorated performance through the utilization of structured data derived from feature extraction outcomes based on a confluence of three word embedding techniques. It is noteworthy, however, that both techniques still fail to achieve optimal efficacy.

The model with the best average performance is the T5 feature extraction technique and 3D CNN LSTM, with an average accuracy of 85.38%. The model with the second best average performance with an average accuracy of 84.48% is a 2D CNN LSTM model that processes the same structured data. These results show that 3D Type 2 data stores information on meaning and word order better because each word meaning is represented by a combination of three word embedding vectors in a 100×3 matrix. This combination has the potential to provide complete information on word meaning. Meanwhile, sequence information is represented by arranging matrix layers according to the order of words in the sentence. These results also prove that the 3D CNN LSTM algorithm works better than 2D CNN LSTM for processing 3D data. The 3D CNN layer is able to extract important features in 3D data better because the convolution operation moves in three dimensions and thus obtains better local patterns.

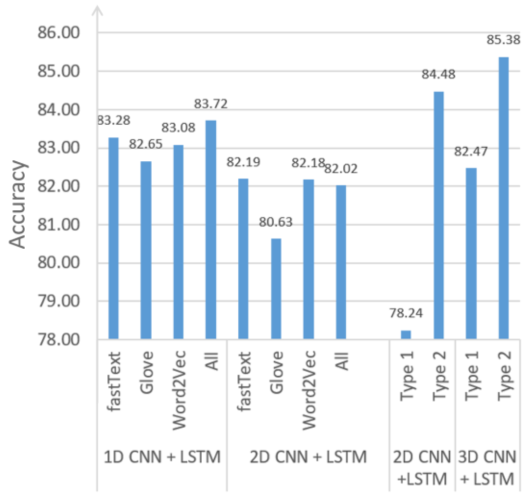

Figure 11. Average performance comparison

To ascertain the impact of amalgamating LSTM and CNN, we juxtaposed the findings of this investigation with prior research endeavours exclusively employing CNN [6]. The juxtaposition is visually presented in Figure 12. From the contrasts elucidated in Figure 12, it becomes evident that the hybrid CNN LSTM was successful in augmenting the classification efficacy of messages of natural calamities. The augmentation in performance achieved was of the utmost statistical significance, as per the outcomes of the two-tailed P-value derived from the paired t-test, which was less than 0.0001.

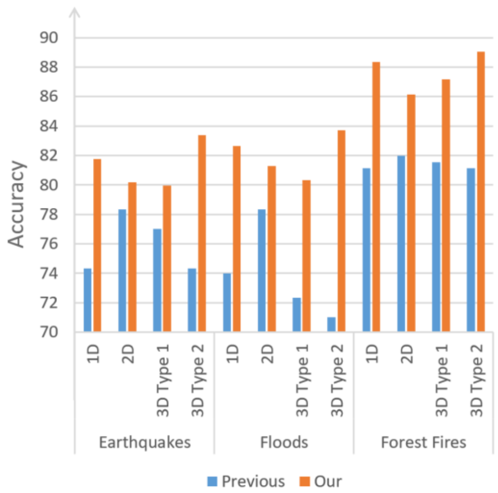

Figure 12. Comparison of our experiment result to the previous result

5. CONCLUSION

Previous investigations into identifying reports on natural disasters have been carried out employing statistical calculations based on word padding (such as mean, median, and mode), the dimensional variation of data results derived from feature extraction based on word embedding, and CNN-based classification algorithms. Models for identification have been explicitly constructed for three natural disaster reports: earthquakes, floods, and forest fires. The average performance produced was still below 80%, and the 3D data performance did not work well in this research, this may be because 2D CNN processes 3D data.

In this research, we have developed a model using the same technique for feature extraction as the earlier study, but with the incorporation of word padding maximum and the hybrid CNN LSTM as a classification algorithm. Twelve models were created for each natural disaster dataset. The most effective identification models are those constructed using feature extraction that generates 3D data type 2 and the 3D CNN LSTM algorithm. The identification accuracy of this model for earthquakes, floods, and forest fires is 83.38%, 83.72%, and 89.03%, respectively. The hybrid CNN LSTM algorithm has also demonstrated a significant improvement in the identification performance of natural disasters compared to previous studies. This is evident from the paired t-test results, where the two-tailed P-value is less than 0.0001.

The current study employs a rudimentary CNN LSTM hybrid architecture with default parameters. Although the resulting performance is improved, it still falls short of reaching 90%. Consequently, this opens up avenues for future research to develop more advanced CNN LSTM hybrid architectures that can achieve an accuracy rate surpassing 90%. Furthermore, advanced investigations can be conducted utilizing the bidirectional encoder representations from transformers (BERT) method.

ACKNOWLEDGEMENTS

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

REFERENCES

[1] D. G. S. Pranowo *et al.*, "Monitoring System of Natural Disaster from Twitter Messages Using Support Vector Machine," in *2022 9th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2022, pp. 179–185, doi: 10.23919/EECSI56542.2022.9946573.

[2] Q. Zou, "A prototype system using location-based twitter data for disaster management," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019, pp. 9514–9517, doi: 10.1109/IGARSS.2019.8898545.

[3] M. K. Delimayanti, R. Sari, M. Laya, M. R. Faisal, Pahrul, and R. F. Naryanto, "The effect of pre-processing on the classification of twitter's flood disaster messages using support vector machine algorithm," *Proceedings of ICAE 2020 - 3rd International Conference on Applied Engineering*, no. February 2021, 2020, doi: 10.1109/ICAE50557.2020.9350387.

[4] 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," in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*, pp. 317–322, 2021, doi: 10.1109/IC2IE53219.2021.9649107.

[5] 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 International Conference on Computer and Informatics Engineering, IC2IE 2022*, no. December, pp. 192–197, 2022, doi: 10.1109/IC2IE56416.2022.9970188.

[6] M. R. Faisal *et al.*, "Using Social Media Data to Monitor Natural Disaster: A Multi Dimension Convolutional Neural Network Approach with Word Embedding," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 6, pp. 1037–1046, 2022, doi: 10.29207/resti.v6i6.4525.

[7] M. R. Faisal *et al.*, "LSTM and Bi-LSTM Models For Identifying Natural Disasters Reports From Social Media," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 5, no. 4, pp. 241-249, 2023, doi: 10.35882/jeeemi.v5i4.319.

[8] U. D. Gandhi, P. M. Kumar, G. C. Babu, and G. Karthick, "Sentiment analysis on twitter data by using convolutional neural network (CNN) and long short term memory (LSTM)," *Wireless Personal Communications*, pp. 1–10, 2021, doi: 10.1007/s11277-021-08580-3.

[9] S. Bodapati, H. Bandarupally, R. N. Shaw, and A. Ghosh, "Comparison and analysis of RNN-LSTMs and CNNs for social reviews classification," in *Advances in Applications of Data-Driven Computing*, Springer, vol. 1319, pp. 49–59, 2021, doi: 10.1007/978-981-33-6919-1_4.

[10] M. A. Riza, N. Charibaldi, U. Pembangunan, and N. Veteran, "Emotion Detection in Twitter Social Media Using Long Short - Term Memory (LSTM) and Fast Text," *International Journal of Artificial Intelligence & Robotics (IJAIR)*, vol. 3, no. 1, pp. 15–26, 2021, doi: 10.25139/ijair.v3i1.3827.

[11] M. K. M. Boussougou and D.-J. Park, "Attention-Based 1D CNN-BiLSTM Hybrid Model Enhanced with FastText Word Embedding for Korean Voice Phishing Detection," *Mathematics*, vol. 11, no. 14, p. 3217, 2023, doi: 10.3390/math11143217.

[12] L. Xiaoyan, R. C. Raga, and S. Xuemei, "GloVe-CNN-BiLSTM model for sentiment analysis on text reviews," *Journal of Sensors*, vol. 2022, pp. 1–12, 2022, doi: 10.1155/2022/7212366.

[13] L. Khan, A. Amjad, K. M. Afaq, and H.-T. Chang, "Deep sentiment analysis using CNN-LSTM architecture of English and Roman Urdu text shared in social media," *Applied Sciences*, vol. 12, no. 5, p. 2694, 2022, doi: 10.3390/app12052694.

[14] I. Thalib, Widyawan, and I. Soesanti, "Indonesian question classification using feature extraction and selection approach on coarse and fine taxonomy," in *AIP Conference Proceedings*, vol. 2654, no. 1, p. 20007, 2023, doi: 10.1063/5.0114188.

[15] E. M. Dharma, F. L. Gaol, H. L. H. S. Warnars, and B. Soewito, "The Accuracy Comparison Among Word2Vec, Glove, and Fasttext Towards Convolution Neural Network (CNN) Text Classification," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 2, pp. 349–359, 2022.

[16] C. Wang, P. Nulty, and D. Lillis, "A Comparative Study on Word Embeddings in Deep Learning for Text Classification," *ACM International Conference Proceeding Series*, pp. 37–46, 2020, doi: 10.1145/3443279.3443304.

[17] C. Xu, J. Zhou, W. Cai, Y. Jiang, Y. Li, and Y. Liu, "Robust 3D hand detection from a single RGB-D image in unconstrained environments," *Sensors*, vol. 20, no. 21, p. 6360, 2020, doi: 10.3390/s20216360.

[18] K. Natarajan and K. Annappan, "Enhancing Endometrial Cancer Detection by Feature Entanglement Image Generator and Multimodal Feature Learning with Attention Mechanism," *International Journal of Intelligent Engineering & Systems*, vol. 16, no. 5, pp. 238-250, 2023, doi: 10.22266/ijies2023.1031.21.

[19] J.-H. Kim and C. S. Won, “Action recognition in videos using pre-trained 2D convolutional neural networks,” *IEEE Access*, vol. 8, pp. 60179–60188, 2020, doi: 10.1109/ACCESS.2020.2983427.
[20] S. Fadl, Q. Han, and Q. Li, “CNN spatiotemporal features and fusion for surveillance video forgery detection,” *Signal Processing: Image Communication*, vol. 90, p. 116066, 2021, doi: 10.1016/j.image.2020.116066.
[21] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into deep learning*. Cambridge University Press, 2023.
[22] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, “1D convolutional neural networks and applications: A survey,” *Mechanical systems and signal processing*, vol. 151, p. 107398, 2021.
[23] S. Mittal *et al*., “A survey of accelerator architectures for 3D convolution neural networks,” *Journal of Systems Architecture*, vol. 115, p. 102041, 2021, doi: 10.1016/j.ymssp.2020.107398.
[24] J. Gao, X. Ye, X. Lei, B. Huang, X. Wang, and L. Wang, “A multichannel-based cnn and gru method for short-term wind power prediction,” *Electronics*, vol. 12, no. 21, p. 4479, 2023, doi: 10.3390/electronics12214479.
[25] J. Wang, J. Li, and Y. Zhang, “Text3D: 3D Convolutional Neural Networks for Text Classification,” *Electronics*, vol. 12, no. 14, p. 3087, 2023, doi: 10.3390/electronics12143087.
[26] S. Alam, M. A. U. Haque, and A. Rahman, “Bengali text categorization based on deep hybrid CNN--LSTM network with word embedding,” in *2022 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, 2022, pp. 577–582, doi: 10.1109/ICISET54810.2022.9775913.
[27] P. Sun, J. Wang, and Z. Dong, “CNN--LSTM Neural Network for Identification of Pre-Cooked Pasta Products in Different Physical States Using Infrared Spectroscopy,” *Sensors*, vol. 23, no. 10, p. 4815, 2023, doi: 10.3390/s23104815.
[28] N. L. Hakim, T. K. Shih, S. P. K. Arachchi, W. Aditya, Y.-C. Chen, and C.-Y. Lin, “Dynamic hand gesture recognition using 3DCNN and LSTM with FSM context-aware model,” *Sensors*, vol. 19, no. 24, p. 5429, 2019, doi: 10.3390/s19245429.
[29] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, “A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis,” *Multimedia Tools and Applications*, vol. 78, pp. 26597–26613, 2019, doi: 10.1007/s11042-019-07788-7.
[30] E. M. Saoudi, J. Jaafari, and S. J. Andaloussi, “Advancing human action recognition: A hybrid approach using attention-based LSTM and 3D CNN,” *Scientific African*, vol. 21, p. e01796, 2023, doi: 10.1016/j.sciaf.2023.e01796.

BIOGRAPHIES OF AUTHORS

Mohammad Reza Faisal was born in Banjarmasin. Following his graduation from high school, he pursued his undergraduate studies in the Department of Informatics 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, 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.

Dodon Turianto Nugrahadi is a lecturer in Department of Computer Science, Lambung Mangkurat University. His research interest is centered on data science and computer networking. He completed his Bachelor’s degree in Informatics Engineering in the Petra, Surabaya in 2004. After that, he pursued a Master’s degree in Information Engineering at Gajah Mada University, Yogyakarta in 2009. His current area of research revolves around network, data science, internet of things (IoT), and network quality of service (QoS). He can be contacted at email: dodonturianto@ulm.ac.id.

Irwan Budiman successfully finished his Bachelor’s degree in the Department of Informatics at the Islamic University of Indonesia. Subsequently, he assumed the role of a lecturer in Computer Science at Universitas Lambung Mangkurt starting from 2008. Additionally, in 2010, he pursued a Master’s degree in Information Systems at Diponegoro University. Currently, he holds the position of chair for the computer science study program at Universitas Lambung Mangkurt. His area of research expertise lies in data science. He can be contacted at email: irwan.budiman@ulm.ac.id.

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 as a manager of a scientific journal at his workplace as a lecturer. The research field he is involved in is data science. He can be contacted at email: muliadi@ulm.ac.id.

Mera Kartika Delimayanti received the Doctoral degree in Electrical Engineering and Computer Science from Kanazawa University, Japan with specialization in Bioinformatics. Her research areas are machine learning, deep learning, bioinformatics, medical informatics, and biomedical engineering. Now, she is a lecturer at Department of Computer and Informatics Engineering, Politeknik Negeri Jakarta. She is a Head of Information and Communication Technology Unit at Politeknik Negeri Jakarta. She has supervised and co-supervised more than 80 undergraduate students and 2 masters students. She has authored or coauthored more than 30 publications: 15 proceedings and 15 journals, with 5 H-index Scopus. She has gained more than 10 intelectual property rights from Ministry of Law and Human Rights, Indonesia. She was awarded as Exemplary lecturer at 2016 in Politeknik Negeri Jakarta. She can be contacted at email: mera.kartika@tik.pnj.ac.id.

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. He can be contacted at email: septyan.e.prastya@unism.ac.id.

Imam Tahyudin received Bachelor of Science (S.Si.) from Universitas Jenderal Soedirman, Magister of Informatics (M. Kom.) from Universitas Amikom Yogyakarta, Magister of Management (M.M.) from Universitas Jenderal Soedirman, and Doctor of Engineering (Dr. Eng) from Kanazawa University, Japan. He is a member of Computer Higher Education Association (APTIKOM), a member of Indoceiss and the secretary of Central Java Chapter of Indoceiss. He has been a lecturer since 2009 and now. He is a Dean of Computer science faculty of Universitas Amikom Purwokerto. His research field in artificial intelligent, data mining, machine learning, biological monitoring, IoT, DSS, and information system. He has published more than 20 scientific articles in Scopus indexed, has 2 patents, has more than 9 copyright of software, and has published more than 15 scientific books. He can be contacted at email: imam.tahyudin@amikompurwokerto.ac.id.