Journal of Theoretical and Applied Information Technology
31st July 2024. Vol.102. No. 14

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TSUNAMI DISASTER POTENTIAL CLASSIFICATION USING C-LSTM METHOD

INDAH AYU KARUNIA WATI[1], ANTONI WIBOWO[2]
[1, 2]Computer Science Departement, Binus Graduate Program - Master in Computer Science, Bina Nusantara University, Jakarta, 11480, Indonesia
Email: [1]indah.wati001@binus.ac.id, [2]anwibowo@binus.edu

ABSTRACT

Tsunami was one of the natural disasters caused by tectonic earthquakes, volcanic eruptions at sea, and underwater landslides. Classification or detection of tsunami disasters was very important to help detect tsunami disasters so that they could increase awareness and minimize the impact of losses. To perform classification, deep learning can be used. Therefore, this research aimed to classify the potential occurrence of tsunamis based on earthquake events that occured using the C-LSTM algorithm by configuring hyperparameter tuning implemented using the Python programming language. Hyperparameter tuning used such as filter, kernel_size, stride, optimizer, learning_rate, batch_size, num_lstm_layer. In this study a proportion of data 60%: 40%, 70%: 30%, 80%: 20%, 90%: 10% was used. Based on the results of the study, the highest accuracy was obtained at a proportion of 90%: 10% with the hyperparameters used, namely filter 64, kernel size 3, stride 1, the number of LSTM layers as many as 3 layers, using the adam optimizer with a learning rate of 0.001, and a batch size of 32 so as to obtain an accuracy of 91.46%, with a precision of 86.67%, recall of 89.66%, and f1-score of 88.14%.
Keywords: *C-LSTM, Classification, Tsunami, Hyperparameter Tuning*

1. INTRODUCTION

Tsunamis are unpredictable and infrequent natural disasters [1]. Tsunamis can be caused by earthquakes, landslides, volcanic activity, or meteorite falls [2]. Between 1900 and 2020, 149 tsunamis were recorded in Indonesia, Papua New Guinea, and the Philippines. Of those events, 85% have been attributed solely to earthquake. In addition, seven tsunamis that occurred on Indonesian territory each caused more than 1,000 fatalities, and of these seven tsunamis, three were caused by large earthquakes [3]. Tsunami waves generated from earthquake events have three stages, namely the generation, propagation and run-up stages, so it is necessary to classify or detect tsunami disasters based on earthquake events that occur in Indonesia.

Tsunami classification or detection is essential for effective disaster management and risk reduction [4]. This is important, considering that Indonesia Indonesia is an active tectonic zones country caused by the meeting of three tectonic plates, that is Indo-Australia ocenic plate, Pasific ocenic plate and Eurasia continental plate [5]. The tectonic plates continue to move actively so that they are prone to earthquakes due to the release of seismic waves in rocks within the earth's crust [6].The classification or detection of tsunami disasters aims to assist in classifying or detecting whether a tsunami disaster will occur or not, so that the classification results can increase public awareness and minimize the impact of losses experienced by victims. To classify or detect tsunami disasters, researchers utilize technological developments in the field of NLP, especially neural network technology such as deep learning.

Deep Learning is a concept in machine learning based on artificial neural networks [7]. Deep learning has proven to be able to carry out the classification process very well, this is proven by research conducted by Yuandong Luan and Shaofu Li [8] which discusses text classification by comparing CNN, LSTM, CNN-LSTM, NA-CNN-LSTM, CNN-COIF-LSTM, and NA-CNN-COIF-LSTM models. The data used is subjective and objective text data including 5000 subjective texts and 5000 objective text data. ". The results of this study show that the precision of the CNN-LSTM model has the best performance of 99.4769%. Further research conducted by Hai Zhou [9] which discusses text classification using CNN-LSTM with TF-IDF. The model will be compared with LSTM, BiLSTM, and LSTM+attention modelsThe dataset used is THUCNews and Taobao review. The results show that the CNN-LSTM model without using TF-IDF has the highest accuracy compared to other models on the THUCNews dataset of 0.999 and on the Taobao review dataset of 0.931.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

From the description above, this research will classify or detect whether a tsunami disaster will occur or not using classification techniques using the Convolutional Long Short-Term Memory (C-LSTM) algorithm. The use of the C-LSTM algorithm in this study is because C-LSTM has advantages obtained from CNN and LSTM methods such as CNN has advantages to extract the features of the input data for obataining better models [10], and LSTM has the advantage of of better capturing context information [11], so the C-LSTM combination is carried out to obtain the advantages of each of these methods . In addition, the use of C-LSTM in this study is because C-LSTM gets high accuracy results if only using LSTM alone [10], Therefore, this research has a contribution by applying a combination of the C-LSTM method and hyperparameter tuning configuration used to improve the performance of the C-LSTM model in the classification of tsunami potential from earthquake events that occur in the hope that it can help classify or detect whether a tsunami disaster will occur or not from the earthquake event that occurred. The research is also expected to provide insight to the public regarding the potential for tsunamis and can anticipate the occurrence of potential tsunamis based on earthquake events that occur.

2. LITERATURE REVIEW

There are several related studies that have been conducted previously, namely Previous research conducted by Rahman, et al, (2020) [12]. The research discusses the development of diabetes classification based on Convolutional Long Short-term Memory (Conv-LSTM). The background of the research is that the conventional diabetes detection process is tedious so that an automated system is needed to identify diabetes from clinical and physical data. The data used in the study is using the Pima Indian Diabetes Database (PIDD). And hyperparameter optimization is carried out using the Grid Search algorithm to find the optimal parameters in the applied model. The results of the study using the cross validation technique resulted in an accuracy of 97.26%. Further research has been conducted by Luan & Lin, (2019) [8]. The research discusses text classification by comparing CNN, LSTM, CNN-LSTM, NA-CNN-LSTM, CNN-COIF-LSTM, and NA-CNN-COIF-LSTM models. The dataset used is subjective and objective text data including 5000 subjective texts and 5000 objective text data. Based on the research results, it was found that in terms of precision the CNN-LSTM model got the best performance of 99.4769.

Further research was conducted by Singh & Sehgal, (2021) [13]. The research is based on the development of a reliable model for the diagnosis and classification of dental caries so that it causes effective and precise treatment time. This research proposes a CNN-LSTM model to perform detection and diagnosis of dental caries. The proposed model uses CNN for feature extraction and LSTM to perform short term and long term dependency. The CNN-LSTM model shows the best performance with 96% accuracy and helps in dental care.

Further research was conducted by Kang, et al. (2021) [14]. The research is motivated by the mental stress that many people face, making it very important to manage and monitor one's stress. The study proposed an ensemble algorithm that can determine the state of mental stress using CNN-LSTM with classification using electrocardiogram (ECG) signals. Based on the research results by evaluating the performance of the stress classification model using confusion matrix, ROC curve, and precision-recall curve, the accuracy of the model reached 98.3%. Further research was conducted by Srikantamurthy, et al, (2023) [15]. The study was motivated by the need for a lot of time and expert pathologists and physicians in the assessment of cancer hispatology slides. Therefore, automatic hispathologic classification of breast cancer subtypes is useful for clinical diagnosis and therapeutic response. The model used is a combination of CNN and LSTM to classify four subtypes of benign breast cancer and four subtypes of malignant breast cancer. Based on the results, 99% accuracy was obtained in binary classification and 92.50% for multi-class classifier in classifying benign and malignant cancer subtypes.

Based on the description of the Literature Review that has been described previously, a summary of the Literature Review is made which can be seen in Table 1.

Table 1: Literature Review

No	Author	Title	Research Methods	Research Results
1	Motiur Rahman, Dilshad Islam, Rokeya Jahan Mukti, and	A Deep Learning Approach based on Convolutional LSTM for	Convolutional Long Short-term Memory (Conv-LSTM).	This study conducted hyperparameter optimization using the Grid Search

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

No	Author	Title	Research Methods	Research Results
	Indrajit Saha *[12]*	Detecting Diabetes		algorithm to find the optimal parameters in the applied model. The results of the research using cross validation technique resulted in an accuracy of 97.26%.
2	Yuandong Luan and Shaofu Li, 2019 *[8]*	Research on text classification based on CNN and LSTM	CNN, LSTM, CNN-LSTM, NA-CNN-LSTM, CNN-COIF-LSTM, and NA-CNN-COIF-LSTM	The dataset used is subjective and objective text data including 5000 subjective texts and 5000 objective text data. Based on the research results, it was found that in terms of precision the CNN-LSTM model got the best performance of 99.4769%.
3	Prerna Singh and Priti Sehgal, 2021 *[13]*	GV Black dental caries classification and preparation technique using optimal CNN-LSTM classifier	CNN-LSTM	CNN for feature extraction and LSTM to perform short term and long term dependency. The CNN-LSTM model shows the
				best performance with 96% accuracy and helps in dental treatment.
4	Mingu Kang, Siho Shin,Jaehyo Jun, Youn Tae Kim, 2021 *[14]*	Classification of mental stress using CNN-LSTM algorithms with electrocardiogram signals	CNN-LSTM	The study proposed an ensemble algorithm that can determine the state of mental stress using CNN-LSTM with classification using electrocardiogram (ECG) signals, and resulted in model accuracy reaching 98.3%.
5	Mahati Munikoti Srikantamurthy, V. P. Subramanyam Rallabandi, Dawood Babu Dudekula, 2023 *[15]*	Classification on of benign and malignant subtypes of breast cancer histopathology imaging using hybrid CNN-LSTM based transfer learning	Combination of CNN and LSTM	This study classifies four subtypes of benign breast cancer and four subtypes of malignant breast cancer. Based on the results, 99% accuracy was obtained in binary classification and 92.50% for multi-class classifier

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

No	Author	Title	Research Methods	Research Results
				in classifying benign and malignant cancer subtypes.

Based on the literature review described in Table 1, the application of the C-LSTM method in classification can be applied well, indicated by the accuracy of the C-LSTM method getting the best performance compared to other methods [8]. This can be due to the combination process of CNN and LSTM (C-LSTM) because CNN can be used to extract high-level phrase sequence features and send them to LSTM to get a sentence representation, then the output of the LSTM layer will be forwarded to the fully connected layer.

Therefore, this research will classify the potential for a tsunami or no tsunami disaster based on the earthquake that occurred using the C-LSTM method. While the difference between this research and previous research is in the problems to be studied, where this research will classify the potential for a tsunami disaster.

3. RESEARCH METHODS

Research methods are used to achieve research objectives. In this study, a framework was created that describes the research workflow. The following in Figure 1 is the framework used in this study.

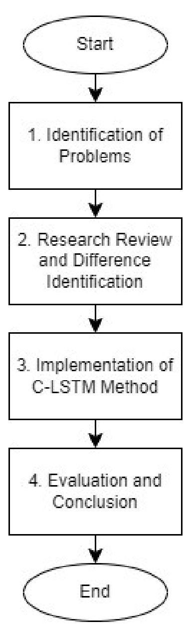

Figure 1: Framework

The following in Figure 2 are the stages of the implementation of the C-LSTM method used in this study.

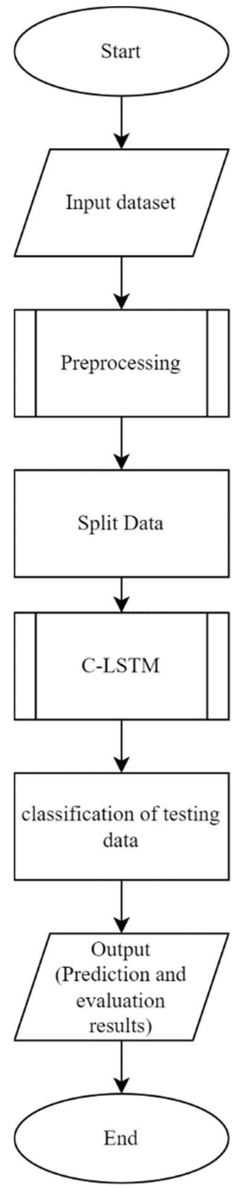

Figure 2: Classification Stages Using the C-LSTM Method

From Figure 1 above, it can be explained about the process in this study as follows:

1. Problem identification in this research focuses on the classification of tsunami disaster potential from earthquake event data. Previously, similar approaches have been applied in the health field to identify disease patterns or symptoms. However, in this context, this research faces a unique challenge as it requires simultaneous

understanding of spatial and temporal patterns in earthquake data to identify earthquakes that have the potential to create tsunamis. In this study, a Convolutional Long Short-Term Memory (C-LSTM) approach is used to address the challenge by combining the advantages of spatial analysis (CNN) and temporal analysis (LSTM). This approach is innovative as it allows researchers to apply existing models in the health field to a different context, namely tsunami disaster mitigation.

2. Conduct an in-depth review of previous studies related to the classification of tsunami disaster potential from earthquake events. Analyze methods that have been applied previously to detect earthquake patterns that have the potential to cause tsunamis. Focus on combining Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) in one model, namely Convolutional Long Short-Term Memory (C-LSTM), to improve understanding of spatial and temporal patterns in earthquake data. At this stage, the C-LSTM model will be implemented to classify potential tsunami disasters. C-LSTM is chosen for its ability to combine spatial and temporal analysis in earthquake data. This method will be applied to the collected dataset to identify earthquake patterns that have the potential to cause tsunamis.
3. Implementation of the C-LSTM model to classify potential tsunami disasters. The use of C-LSTM was chosen due to its ability to combine spatial and temporal analysis in earthquake data. The research will apply this method to the collected dataset to identify earthquake patterns that have the potential to cause tsunamis.
4. Evaluation of the C-LSTM model uses metrics such as accuracy, precision, recall, and F1-score. The evaluation results are used to draw conclusions about the model's ability to classify potential tsunami disasters from earthquake events. In addition, the implications of the results of this research are discussed to provide additional insights.

4. RESULTS AND DISCUSSION

4.1. Data Processing Results

The research results obtained are described in several stages starting with library import, data collection, data preprocessing, data splitting, classification using C-LSTM, and evaluation. An explanation of the process will be shown in the explanation below.

4.1.1. Import Library

The libraries used include pandas, numpy, seaborn, matpotlib, sklearn library with preprocessing module imported MinMaxScaler, label encoder, sklearn library with model selection module imported train test split, accuracy score, confusion matrix, precision score, recall score, f1 score, tensorflow. Library pandas is used for reading and processing tables, library numpyis used for array analysis, library library matpotlib and seaborn are used for dataset modelling, analysis, and visualization [16]. Library itertools imported product to returns the cartesian product of the input iterables [17]. Sklearn library with preprocessing module imported minmaxscaler is used to transform the dataset to scale 0 (minimum) to 1 (maximum) [18] and label encoder are used to apply label encoding to particular feature or column [19]. Sklearn library with imported model selection module train test split is used for data distribution into training data and testing data [20]. Sklearn metrics used for specify the metrics used in the evaluation of the methods [21]. As well as the tensorflow library for model building [22]. The source code library used can be seen in Figure 3.

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from itertools import product
from sklearn.preprocessing import MinMaxScaler, LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix, precision_score, recall_score, f1_score
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv1D, MaxPooling1D, LSTM, Dense
from tensorflow.keras.optimizers import Adam, RMSprop
from tensorflow.keras.models import load_model
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
```

Figure 3: Source Code Import Library

4.1.2. Data Collection

The data used in this study is data derived from the Kaggle Earthquake dataset which obtained as many as 817 earthquake data from January 1, 2001 to August 16, 2023. The data is obtained in the form of a .csv file and will be load data.

The data consists of 19 features (18 independent features and 1 dependent feature) namely title, magnitude, date_time, cdi, mmi, alert, tsunami, sig, net, nst, dmin, gap, magType, depth, latitude, longitude, location, continent, country. Feature dependent is a tsunami feature consisting of 2 classes, namely 1 (potential tsunami occurrence) and 0 (no tsunami potential).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.1.3. Preprocessing

At this stage, unnecessary features such as title, location, continent and country are removed. The removal of these features is because they are not needed / have no effect on the classification process. Preprocessing consists of feature engineering, label encoder and normalization.

4.1.4. Split Data

At this stage, data is divided into training data and testing data. Dataset division is carried out into several data proportions, namely 60% training data and 40% testing data, 70% training data and 30% testing data, 80% training data and 20% testing data, and 90% training data and 10% testing data.

4.1.5. Classification of C-LSTM Methods

1. 60% Training Data and 40% Testing Data

   The classification results of the C-LSTM method with a proportion of 60% training data and 40% testing data obtained accuracy and model loss plots.

   Figure 4 shows the plot between the accuracy results of train data and validation data at a proportion of 60%: 40%. The validation data used is taken from the test data. The best accuracy is obtained at epoch 12 with the resulting accuracy in the train data of 0.8163 and in the validation data of 0.8287.

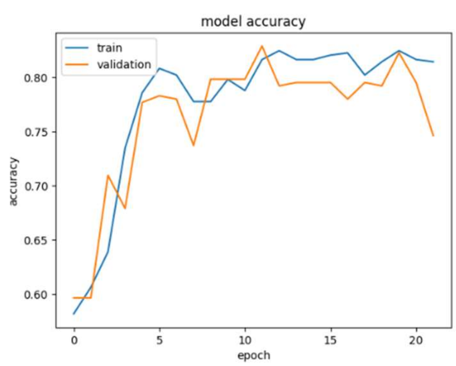

Figure 4: Results of 60%:40% Proportion Accuracy Plot

Figure 5 shows the plot between the loss results of train data and validation data at a proportion of 60%:40%. The validation data used is taken from the test data. The best loss value is obtained at epoch 12 with the resulting loss value in the train data of 0.4192 and in the validation data of 0.4352.

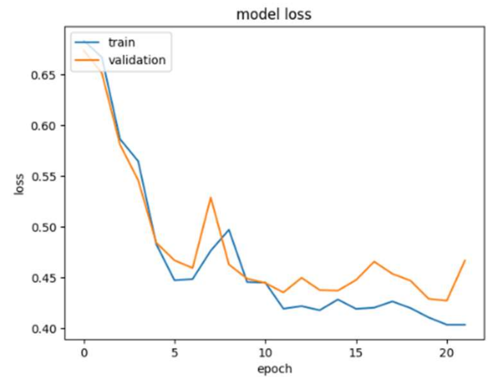

Figure 5: Results of 60%:40% Proportion Loss Value Plot

2. 70% Training Data and 30% Testing Data

   The classification results of the C-LSTM method with a proportion of 70% training data and 30% testing data obtained accuracy and model loss plots.

   Figure 6 shows the plot between the accuracy results of train data and validation data at a proportion of 70%: 30%. The validation data used is taken from the test data. The best accuracy is obtained at epoch 11 with the resulting accuracy in the train data of 0.8126 and in the validation data of 0.8333.

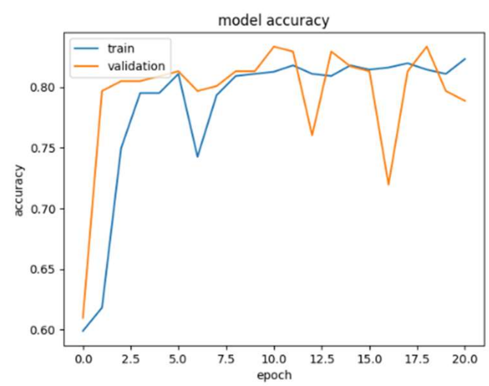

Figure 6: Results of 70%:30% Proportion Accuracy Plot

Figure 7 shows the plot between the loss results of train data and validation data at a proportion of 70%:30%. The validation data used is taken from the test data. The best loss value is obtained at epoch 11 with the resulting loss value in the train data of 0.4302 and in the validation data of 0.4237.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

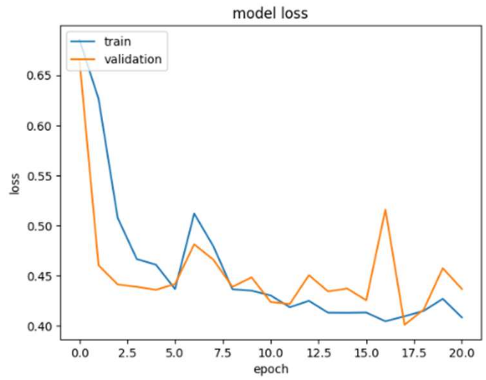

Figure 7: Results of 70%:30% Proportion Loss Value Plot

3. 80% Training Data and 20% Testing Data

The classification results of the C-LSTM method with a proportion of 80% training data and 20% testing data obtained accuracy and model loss plots.

Figure 8 shows the plot between the accuracy results of train data and validation data at a proportion of 80%: 20%. The validation data used is taken from the test data. The best accuracy is obtained at epoch 19 with the resulting accuracy in the train data of 0.8070 and in the validation data of 0.8171.

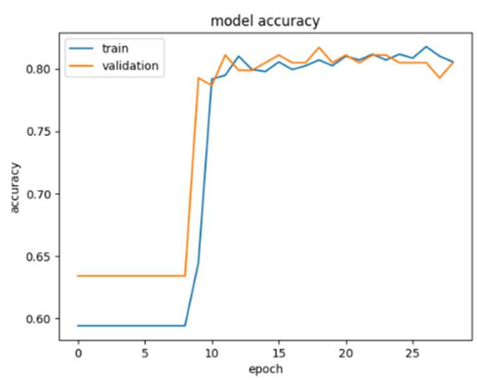

Figure 8: Results of 80%:20%Proportion Accuracy Plot

Figure 9 shows the plot between the loss results of train data and validation data at a proportion of 80%:20%. The validation data used is taken from the test data. The best loss value is obtained at epoch 19 with the resulting loss value in the train data of 0.4422 and in the validation data of 0.4191.

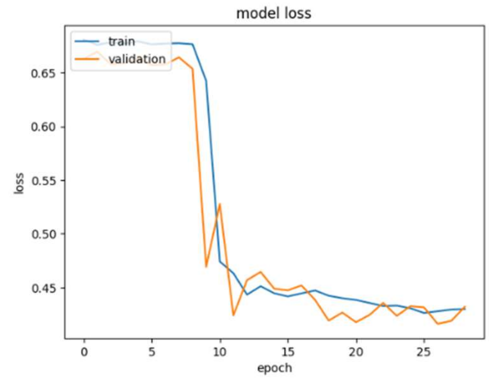

Figure 9: Results of 80%:20% Proportion Loss Value Plot

4. 90% Training Data and 10% Testing Data

The classification results of the C-LSTM method with a proportion of 90% training data and 10% testing data obtained accuracy and model loss plots.

Figure 10 shows the plot between the accuracy results of train data and validation data at a proportion of 80%: 20%. The validation data used is taken from the test data. The best accuracy is obtained at epoch 30 with the resulting accuracy in the train data of 0.8109 and in the validation data of 0.9146.

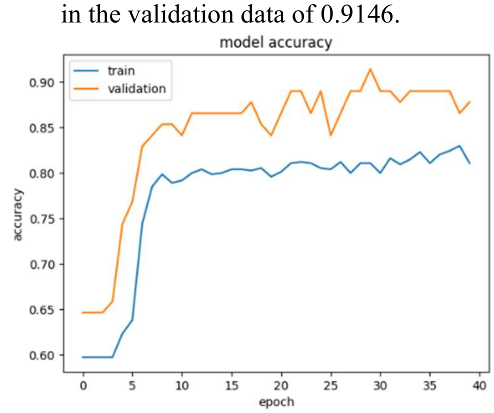

Figure 10: Results of 90%:10% Proportion Accuracy Plot

Figure 11 shows the plot between the loss results of train data and validation data at a proportion of 80%:20%. The validation data used is taken from the test data. The best loss value is obtained at epoch 30 with the resulting loss value in the train data of 0.4173 and in the validation data of 0.3055.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

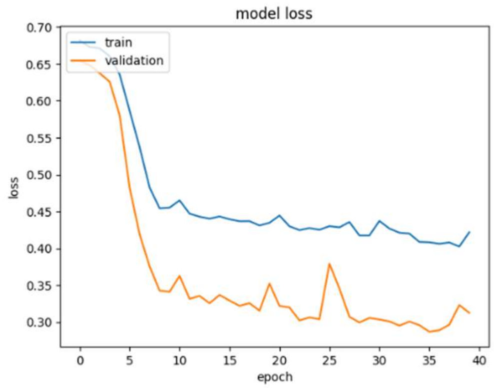

Figure 11: Results of 90%:10% Proportion Loss Value Plot

4.1.6. Evaluasi

1. 60% Training Data and 40% Testing Data

   The confusion matrix results in the proportion of 60% training data and 40% testing data can be seen in Figure 11.

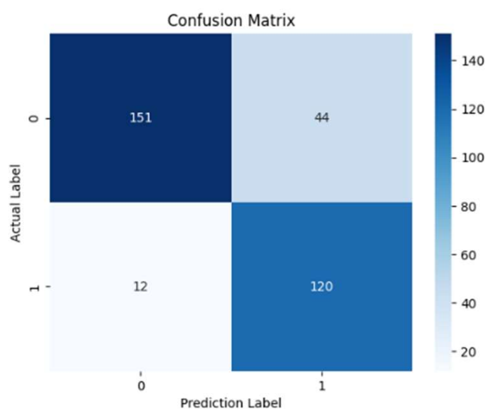

Figure 12: Confusion Matrix on 60%:40% Proportion

   Based on Figure 12, it can be explained that 151 data were correctly classified as label 0 or no tsunami, while 44 data prediction errors were included in label 1 or tsunami. At label 1 or tsunami correctly classified as much as 120 data, while the prediction error with 12 data entered into label 0 or no tsunami.

   The results of the confusion matrix evaluation at a proportion of 60%: 40% resulted in an accuracy value of 82.87%, precision of 73.17%, recall of 90.91%, and f1-score of 81.08%.

2. 70% Training Data and 30% Testing Data

   The confusion matrix results on the proportion of 70% training data and 30% testing data can be seen in Figure 12.

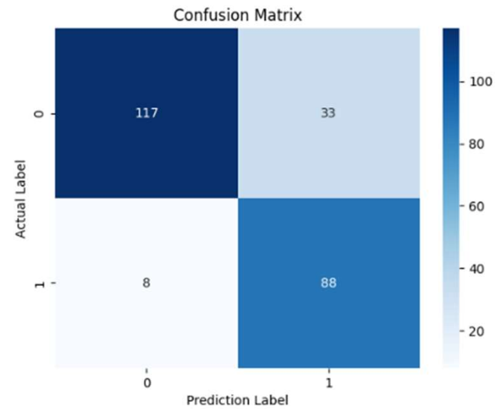

Figure 13: Confusion Matrix on 70%:30% Proportion

   Based on Figure 13, it can be explained that 117 data were correctly classified as label 0 or no tsunami, while the prediction error was 33 data that entered label 1 or tsunami. At label 1 or tsunami, 88 data were correctly classified, while the prediction error with 8 data entered the label 0 or no tsunami.

   The results of the confusion matrix evaluation at a proportion of 70%: 30% resulted in an accuracy value of 83.33%, precision of 72.73%, recall of 91.67%, and f1-score of 81.11%.

3. 80% Training Data and 20% Testing Data

   The results of the confusion matrix in the proportion of 80% training data and 20% testing data can be seen in Figure 14.

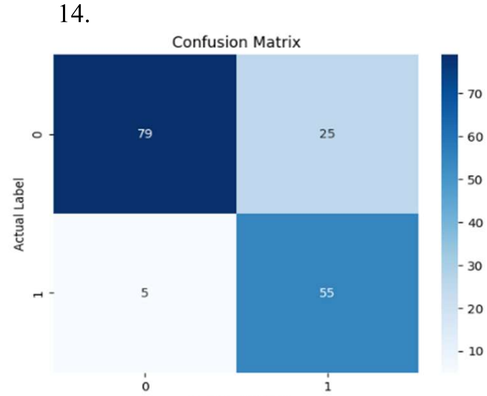

Figure 14: Confusion Matrix on 80%:20% Proportion

Journal of Theoretical and Applied Information Technology
31st July 2024. Vol.102. No. 14

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Based on Figure 14, it can be explained that 79 data were correctly classified as label 0 or no tsunami, while the prediction error was 25 data that entered label 1 or tsunami. At label 1 or tsunami correctly classified as much as 55 data, while the prediction error with 5 data that entered the label 0 or no tsunami.

The results of the confusion matrix evaluation at a proportion of 80%: 20% resulted in an accuracy value of 81.71%, precision of 68.75%, recall of 91.67%, and f1-score of 78.57%.

4. 90% Training Data and 10% Testing Data

The results of the confusion matrix in the proportion of 90% training data and 10% testing data can be seen in Figure 15.

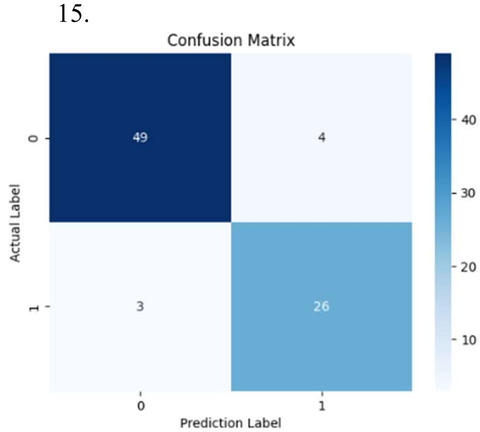

Figure 15: Confusion Matrix on 90%:10% Proportion

Based on Figure 15, it can be explained that 49 data were correctly classified as label 0 or no tsunami, while the prediction error was 4 data that entered label 1 or tsunami. At label 1 or tsunami correctly classified as much as 26 data, while the prediction error with 3 data that falls into label 0 or no tsunami.

The results of the confusion matrix evaluation at a proportion of 90%: 10% resulted in an accuracy value of 91.46%, precision of 86.67%, recall of 89.66%, and f1-score of 88.14%.

4.2 Discussion of Results

Based on the results of research that has been carried out in classifying the occurrence of tsunami disasters from earthquake events using a combination of deep learning C-LSTM methods, the results of model performance on the confusion matrix to obtain accuracy, precision, recall, and f1-score values in each proportion of data sharing can be seen in Table 2.

Table 2: Evaluation Results Using Confusion Matrix

No	Skenario	*Accuracy*	*Precision*	*Recall*	*F1-Score*
1	Train-Test (60%-40%)	82.87%	73.17%	90.91 %	81.08 %
2	Train-Test (70%-30%)	83.33%	72.73%	91.67 %	81.11 %
3	Train-Test (80%-20%)	81.71%	68.75%	91.67 %	78.57 %
4	Train-Test (90%-10%)	91.46%	86.67%	89.66 %	88.14 %

Table 2 shows the evaluation results on the overall proportion of data sharing used in this study. Based on the evaluation results using confusion matrix, the classification to determine the potential occurrence of a tsunami disaster from the earthquake event that occurred obtained the best performance at a proportion of 90%: 10% with hyperparameters used, namely filter 64, kernel size 3, stride 1, the number of LSTM layers as many as 3 layers, using adam optimizer with a learning rate of 0.001, and batch size of 32 so as to obtain accuracy of 91.46%, with precision of 86.67%, recall of 89.66%, and f1-score of 88.14%.

5. CONCLUSION

Based on the results of the research that has been carried out in the classification to determine the potential occurrence of tsunami disasters from earthquake events that occur, the following conclusions are obtained:

1. Classification to determine the potential occurrence of tsunami disasters from earthquake events that occur using a combination of deep learning C-LSTM methods has been successfully implemented using the Python programming language with several libraries used to facilitate the classification of tsunami disasters and is carried out with the proportion of data sharing,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

namely 60%: 40%, 70%: 30%, 80%: 20%, and 90%: 10%.

2. The hyperparameter tuning configuration is carried out at the C-LSTM modeling stage by using filters 32 and 64, kernel sizes 3 and 5, stride 1 and 2, the number of layers in LSTM is 1 layer, 2 layers, and 3 layers, using adam and rmsprop optimizers with learning rates 0.01 and 0.001, and using batch sizes 32 and 64 and searching for the best hyperparameter combination so as to improve the performance of the C-LSTM model. The best hyperparameter combination results in each proportion are in the proportion of 60%: 40% obtained at filter 64, kernel size 3, stride 2, the number of LSTM layers is 3 layers, using adam optimizer with learning rate 0.01, and batch size of 64. At 70%:30% proportion obtained on filter 64, kernel size 3, stride 2, the number of LSTM layers is 3 layers, using the adam optimizer with a learning rate of 0.01, and a batch size of 32. At 80%:20% proportion obtained on filter 64, kernel size 5, stride 2, the number of LSTM layers is 3 layers, using the rmsprop optimizer with a learning rate of 0. 01, and a batch size of 32. While the proportion of 90%: 10% is obtained on filter 64, kernel size 3, stride 1, the number of LSTM layers is 3 layers, using the adam optimizer with a learning rate of 0.001, and a batch size of 32.
3. The highest accuracy results in performing classification to determine the potential occurrence of tsunami disasters from earthquake events that occur are obtained in the proportion of 90%: 10% with hyperparameters used, namely filter 64, kernel size 3, stride 1, the number of LSTM layers of 3 layers, using adam optimizer with a learning rate of 0.001, and batch size of 32. The accuracy obtained is 91.46%, with a precision of 86.67%, recall of 89.66%, and f1-score of 88.14%.

The suggestions that can be given to the development of further research are that it can be developed by using a combination of other deep learning algorithms such as LSTM-GRNN which is a combination of LSTM, GRU, and RNN in performing classification so that algorithm comparisons can be made to find out the best algorithm performance.

REFERENCES

[1] J. Behrens, F. Løvholt, F. Jalayer, S. Lorito, M. A. Salgado-Gálvez, M. Sørensen, S. Abadie, I. Aguirre-Ayerbe, I. Aniel-Quiroga, A. Babeyko, M. Baiguera, R. Basili, S. Belliazzi and A. Grezio, "Probabilistic Tsunami Hazard and Risk Analysis: A Review of Research Gaps," *Frontiers in Earth Science,* vol. 9, pp. 1-28, 2021.

[2] T. Fan, Y. Wang, Z. Xu, L. Sun, P. Wang and J. Hou, "A Review of Historical Volcanic Tsunamis: A New Scheme for a Volcanic Tsunami Monitoring System," *Marine Scinece and Engineering,* pp. 1-13, 2024.

[3] J. A. Reid and W. D. Mooney, "Tsunami Occurrence 1900–2020: A Global Review, with Examples from Indonesia," *Pure and Applied Geophysics,* pp. 1549-1571, 2023.

[4] H. T. Sukmana, Y. Durachman, Amri and Supardi, "Comparative Analysis of SVM and RF Algorithms for Tsunami Prediction: A Performance Evaluation Study," *Journal of Applied Data Sciences,* vol. 5, pp. 84-99, 2024.

[5] A. N. Risqi, A. Riyanto and N. D. Hananto, "Identification of accretionary wedge at Sumatra subduction zone using 2D seismic reflection," *IOP Conf. Series: Earth and Environmental Science,* 2020.

[6] A. Jufriansah, Y. Pramudya, A. Khusnani and S. Saputra, "Analysis of Earthquake Activity in Indonesia by Clustering Method," *Journal of Physics: Theories and Applications,* vol. 5, pp. 92-103, 2021.

[7] C. Janiesch, P. Zschech and K. Heinrich, "Machine learning and deep learning," *Electronic Markets,* pp. 1-11, 2021.

[8] Y. Luan and S. Lin, "Research on Text Classification Based on CNN and LSTM," *IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA) ,* pp. 352-355, 2019.

[9] H. Zhou, "Research of Text Classification Based on TF-IDF and CNN-LSTM," *Journal of Physics: Conference Series,* pp. 1-8, 2021.

[10] H. Mou and J. Yu, "CNN-LSTM Prediction Method for Blood Pressure Based on Pulse Wave," *electronics,* pp. 1-15, 2021.

[11] X. Li and H. Ning, "Chinese Text Classification Based on Hybrid Model of CNN and LSTM," *DSIT 2020: Proceedings of the*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3rd International Conference on Data Science and Information Technology, pp. 129-134, 2020.

[12] M. Rahman, D. Islam, R. J. Mukti and I. Saha, "A deep learning approach based on convolutional LSTM for detecting diabetes," *Computational Biology and Chemistry,* vol. 88, 2020.

[13] P. Singh and P. Sehgal, "G.V Black dental caries classification and preparation technique using optimal CNN-LSTM classifier," *Multimedia Tools and Applications,* vol. 80, no. 4, pp. 1-18, 2021.

[14] M. Kang, S. Shin, J. Jung and Y. T. Kim, "Classification of Mental Stress Using CNN-LSTM Algorithms with Electrocardiogram Signals," *Journal of Healthcare Engineering,* pp. 1-11, 2021.

[15] M. M. Srikantamurthy, V. P. S. Rallabandi and D. B. Dudekula, "Classifcation of benign and malignant subtypes of breast cancer histopathology imaging using hybrid CNN-LSTM based transfer learning," *BMC Medical Imaging,* vol. 23, no. 19, pp. 1-15, 2023.

[16] P. Lemenkova, "Python libraries matplotlib, seaborn and pandas for visualization geospatial datasets generated by QGIS," *ANALELE ŞTIINTIFICE ALE UNIVERSITĂŢII „ALEXANDRU IOAN CUZA" din IAŞI,* pp. 13-32, 2020.

[17] A. D. Korb, "Evaluating approaches for more efficient UDS protocol scanning," Technische Hochschule Ingolstadt, 2021.

[18] D. Penn, V. H. Subburaj, A. S. Subburaj and M. Harral, "A Predictive Tool For Grid Data Analysis Using Machine Learning Algorithms," *IEEE,* pp. 1071-1077, 2020.

[19] V. V, R. A.C, V. K. N and A. G, "Prediction of Loan Approval in Banks using Machine Learning Approach," *International Journal of Engineering and Management Research,* vol. 13, pp. 7-19, 2023.

[20] N. Sharma, S. Mogha, J. Singh and G. Singhal, "Car Resale Price Predictor using Machine Learning," *Easy Chair,* 2023.

[21] P. Ksieniewicz and P. Zyblewski, "stream-learn — open-source Python library for difficult data stream batch analysis," *arXiv,* 2020.

[22] P. Janardhanan, "Project repositories for machine learning with TensorFlow," *Procedia Computer Science,* pp. 188-196, 2020.