Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

# Flood Modelling using Long Short-Term Memory Algorithm with Synthetic Minority Oversampling Technique

Alisa Shamshul Afendi[1], Hema Varssini Segar[2], Shuhaida Ismail[1,*], Shazlyn Milleana Shaharudin[3]

[1] Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Batu Pahat, Johor, Malaysia
[2] Faculty of Applied Sciences and Technology, Universiti Tun Hussein Onn Malaysia, 84600 Muar, Johor, Malaysia
[3] Faculty of Science and Mathematics, Universiti Pendidikan Sultan Idris, Tanjung Malim, Perak, Malaysia

**ARTICLE INFO**

***Article history:***
Received
Received in revised form
Accepted
Available online

***Keywords:***
Deep learning; LSTM; flood modelling; SMOTE

**ABSTRACT**

Flood disasters have occurred quite frequently in Malaysia and has been considered one of the most dangerous natural disasters. Meteorologists and hydrologist are having difficulty in predicting flooding due to the weather changes and are opting to deep learning techniques to predict flooding. This study compares the performance of deep learning techniques for flood prediction, namely Long Short-Term Memory (LSTM) network. Furthermore, the effect of using Synthetic Minority Oversampling Technique (SMOTE) as a 'treatment' to treat imbalanced data was also investigated to ensure all LSTM models were able to predict accurate flooding. The experimental results revealed the treated dataset had a positive impact in predicting flood with higher accuracy. Additionally, to increase the accuracy of deep learning methods, future researchers could use more hidden layers as well as different hyperparameter settings which could help create a better predicting LSTM model.

## 1. Introduction

Floods, in comparison to other natural catastrophic events such as landslides, tsunamis, hurricanes, and haze, have been identified by Rahman and Dewsbury [1] as the most dangerous natural disasters in Malaysia. According to Saimi *et al.,* [2], floods are generally described as bodies of water rising, swelling, and flooding land that has not been covered by water. Heavy rainfall, snowmelt, or coastal storm surges can cause floods. Flooding can occur when water levels rise and overwhelm drainage systems, rivers, and levees. Furthermore, there are several factors that contribute to flooding, including heavy rains, melting snow, and poor drainage systems.

Devastating rainfall events, which are expected to occur in the future as a result of climate change, could result in alarming flooding levels as reviewed by Department of Irrigation and Drainage (DID) Malaysia [3]. Razi *et al.,* [4], environmental scientists have noted that historical flooding events caused by torrential rain in several states of Peninsular Malaysia serve as a stark reminder of the possibility of catastrophic weather patterns brought about by climate change. Several types of floods

* *Corresponding author.*
*E-mail address: shuhaida@uthm.edu.my*

https://doi.org/10.37934/araset.61.2.2840

are recognized by the World Meteorological Organization (WMO) [5], including flash floods, fluvial floods, seasonal floods, urban floods, snowmelt floods, among others.

A flash flood occurs when a large amount of heavy rainfall falls within a short period of time and in a confined area. According to IGI Global [6], flash floods are also called torrential floods, typically defined by roaring torrents that smash over riverbanks, metropolitan roadways, or mountain valleys, sweeping everything in their path. These floods, occurring within minutes to hours after heavy rain, are highly destructive natural disasters, impacting urban and rural areas alike [7]. Consequently, Demeritt *et al.,* [8] described a flood prediction and monitoring tool are essential for mitigating flood-induced damage.

Flood prediction plays a significant role in land-use planning, urban design, and environmental management. Having the ability to accurately predict future floods is key to developing effective and sustainable flood mitigation strategies. Nevertheless, real-time flood onset and development are difficult to predict [9]. Forecasting real-time floods requires knowledge of geography and hydrology as well as accurate weather predictions. Additionally, it requires the ability to collect and analyze data in real time, which is a difficult task given the complexity of hydrological and meteorological systems.

Due to the complexity of real-time flood prediction, scientists have developed models to predict future floods based on historical data. This approach facilitates more accurate and informed decision making about flood mitigation strategies by analyzing historical data and forecasting flood events using statistical procedures. For instance, previous studies by Goodman *et al.,* [10] on flood disaster management and flood prediction systems have been conducted and developed using machine learning or deep learning. Through machine learning, computers are able to learn independently and improve predictions, classifications, and clustering by applying the practice.

The use of deep learning models such as Artificial Neural Networks (ANN) is common in flood prediction as reviewed by Mosavi *et al.,* [11], however, Long Short-Term Memory (LSTM) has become more popular for time-series flood forecasting as reviewed by Moishin *et al.,* [12]. The LSTM is capable of learning from and predicting long-term dependencies, making it suitable for handling time series and sequence data. As a result, it is an attractive choice for flood prediction. The LSTM has been successfully applied to a number of flood-related scenarios [13].

The increase occurrences of flood raised awareness in researchers that aimed to fulfill environmental duty by creating models that could accurately predicts or forecasts floods as discussed by Danladi *et al.,* [14]. Nurumal *et al.,* [15] The comparison of LSTM models with different hyperparameter input and epoch numbers concluded that LSTM model has the best flood prediction performance. Flood occurrences were forecast for the year 2021 using the chosen accurate flood prediction model.

## 2. Methodology

In the methodology part, the methods and material used in the study will be discussed and explained in detail.

### *2.1 Data Collection*

The study utilized data spanning 15 years, commencing from 1st January 2005 to 31st December 2020, for Subang Jaya region. The meteorological information was sourced from The Malaysian Meteorological Department (MET Malaysia) and encompassed 5844 data points with eight distinct variables, including the year, month, day, daily maximum and minimum temperatures (°C), daily relative humidity (%), daily rainfall amount (mm), and daily mean sea level (MSL) pressure (hPa).

Additionally, the dataset encompassed Nino Oceanic Index and Southern Oscillation Index (SOI) data, obtained from an open-source platform. Notably, both the meteorological and index datasets covered the same 15-year timeframe as the Meteorological (MET) Malaysia data.

### *2.2 Feature Engineering*

Feature engineering involves certain procedures such as data cleansing, data normalization, data transformation and feature creation [16]. In this research, data cleansing was done where missing data, outliers and duplicate values were checked. After data cleansing, feature creation is done to develop two new variables which are 'rain intensity' as classified using parameters set by MET Malaysia displayed in Table 1 and 'flood occurrence', classified using the parameters as stated in Table 2. Later, data normalization was done to the training set. Data normalization is done to treat all variables equally since some variables generally have larger values than others. For example, the variable MSL pressure is in thousands meanwhile temperature is in tenths. This can affect how a model identifies these values and gives more importance to one variable compared to the other.

**Table 1**
Categories of rainfall intensity according to its range [17]

| Category | Minimum rainfall (mm) | Maximum rainfall (mm) |
|---|---|---|
| Slight Rain | 0 | 10 |
| Moderate Rain | 10 | 60 |
| Heavy Rain | 60 | 150 |
| Extreme Rain | 150 | ∞ |

**Table 2**
Classification of flood occurrence based on rain intensity

| Rain intensity | Flood occurrence |
|---|---|
| Slight rain | No |
| Moderate rain | No |
| Heavy rain | Yes |
| Extreme rain | Yes |

### *2.3 Data Splitting*

Training and testing data were separated into two categories. 4383 data points from observations totaling 11 years' worth of rainfall are included in the training set. The 1460 data points from the latter four years were used as a testing set. The training set included the period from 1st January 2005 to 31st December 2016, while the testing set covered the period from 1st January 2017 to 31st December 2020. This training and assessment set has a ratio of about 70:30.

### *2.4 Data Balancing*

To 'treat' the unequal distribution between the minority and majority classes, data balancing is carried out. The machine's inability to predict minority groups accurately due to the uneven distribution of data leads to a variety of categorization errors. The Synthetic Minority Oversampling Technique (SMOTE), which creates synthetic data to balance the classes, was introduced to address the uneven distribution of unbalanced data [12]. Interpolation between neighboring minority classes is done using the SMOTE technique. SMOTE finds k-closer nearby data to provide synthetic data rather than just replicating the minority class [9]. SMOTE is therefore mentioned as a "treatment" for

uneven data. "Before treatment" refers to the model's usage of SMOTE to address the dataset's uneven distribution. Meanwhile, 'after treatment' means that SMOTE is implemented to produce synthetic data.

### *2.5 Long Short-Term (LSTM) Memory*

The primary distinguishing feature of an LSTM network lies in its hidden layer, known as memory cells, which incorporate three essential gates, named: forget gate ($f_t$), input gate ($i_t$), and output gate ($O_t$). $f_t$ assumes the responsibility of determining the information that should be excluded from the cell state. On the other hand, $i_t$ is tasked with identifying the information that ought to be incorporated into the cell state. Lastly, the output gate $O_t$ dictates which information from the cell state was utilized.

An LSTM network is comprised of three primary layers: an input layer, one or more memory cell layers, and an output layer. The number of neurons in the input layer corresponds to the count of variables, which in this case is five. To manage computational complexity, this project opts for a two-layer configuration, as additional layers can substantially increase computational demands. The rationale behind this choice is to strike a balance between model complexity and computational efficiency. Figure 1 showed the LSTM architecture, with one input layer, two hidden layers and one output layer.

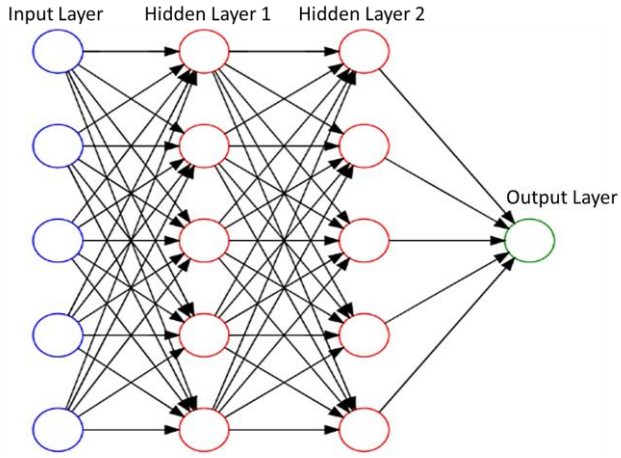



**Fig. 1.** LSTM architecture

The operation of a memory cell commences with the extraction of information from the previous cell state, denoted as $S_{t-1}$ which is determined by the LSTM layer. At a given time, step $t$, the $f_t$ activation value is computed using the present input, $x_t$, the previous memory cell output, $h_{t-1}$, and the bias terms of the forget gates, $b_f$. The sigmoid function is employed to transform activation levels to a range between 0 and 1, signifying complete forgetting and full recollection, respectively. Subsequently, the LSTM layer identifies the information to be incorporated into the network's current cell state, $S_t$. This process involves calculating candidate values, $\widetilde{S_t}$ and determining the activation values of the input gates.

Following this, new cell states, $S_t$ are computed by utilizing the insights obtained from the previous two steps. Finally, the output, $h_t$ of the memory cell unit is obtained. The standard LSTM model employs specific hyperparameters, which encompass functions set to default values. These hyperparameters are detailed in Table 3 as taken from Keras documentation [18]. The activation function facilitates nonlinear transformations within the model.

The recurrent_activation function serves the recurrent step, while the optimizer is employed to minimize overall loss and enhance accuracy. In this particular model, the chosen optimizer is 'Adam', a stochastic gradient descent method that adapts to first and second-order moments. Loss, a crucial metric, determines the quality of predictions by quantifying the disparity between predicted and actual values. Godoy [19] explained that low loss values signify accurate predictions, while high values indicate less accurate ones.

**Table 3**
Hyperparameter setting for LSTM (2) model

| Hyperparameter | Value |
|---|---|
| Number of hidden layers | 2 |
| activation | Hyperbolic tangent (tanh) |
| recurrent_activation | sigmoid |
| epoch | 100 or 200 |
| batch Size | 64 |
| dropout | 0.2 |
| optimizer | Stochastic Gradient Descent method (Adam) |
| loss | binary_crossentropy |

### *2.6 Model Performance*

The model performances were evaluated by a confusion matrix, since it provided information such as accuracy, precision, recall, F1-Score and the AUC for ROC curve [20]. Table 4 shows the framework for a confusion matrix.

**Table 4**
Confusion matrix

| | | Actual class | |
|---|---|---|---|
| | | Positive | Negative |
| Predicted class | Positive | True positive | False positive |
| | Negative | False negative | True negative |

The equations used to calculate the Precision, Recall, F1-Score, Accuracy and Area Under the Curve (AUC) values are as following:

$$\text{Accuracy: A} = \frac{TP+TN}{TP+FP+tn+FN} \quad (1)$$

$$\text{Precision: P} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{Recall: R} = \frac{TP}{TP+FN} \quad (3)$$

$$F1\text{-score: F} = \frac{(2 \cdot P \cdot R)}{(P+R)} \quad (4)$$

$$AUC = \frac{\sum TP+\sum TN}{(P+N)} \quad (5)$$

## 3. Results

### *3.1 Feature Engineering*

Several tasks are addressed under this subtopic, including managing missing data, detecting anomalies like outliers and duplicated dates, feature creation and data normalization. Notably, there were 10 instances of missing values in the daily relative humidity column. Given that these missing entries accounted for less than 5%, they are regarded as not significant. Therefore, imputation was done to rectify the missing values by substituting it with the column's average daily relative humidity, amounting to 78.13%. In relation to this dataset, apart from missing values, no duplicate dates were discovered. Figure 2 depicts a boxplot constructed to check for outliers.

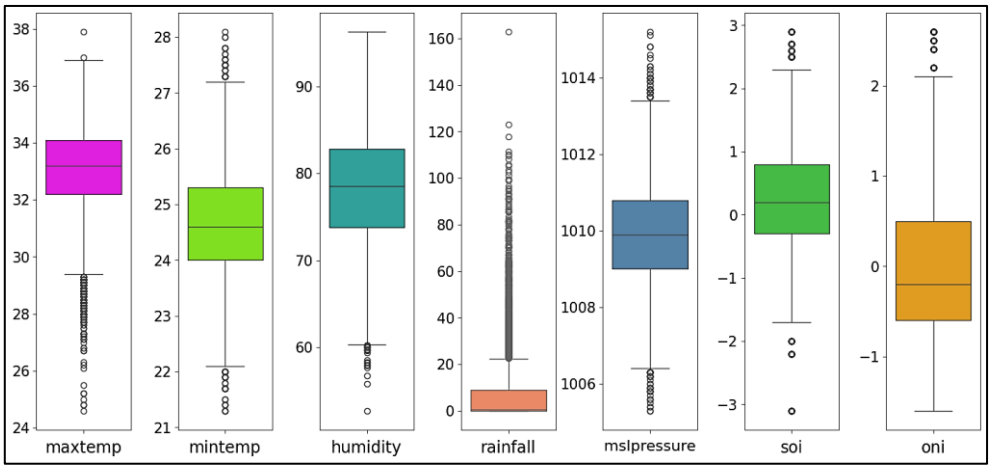



**Fig. 2.** Boxplot for each variable

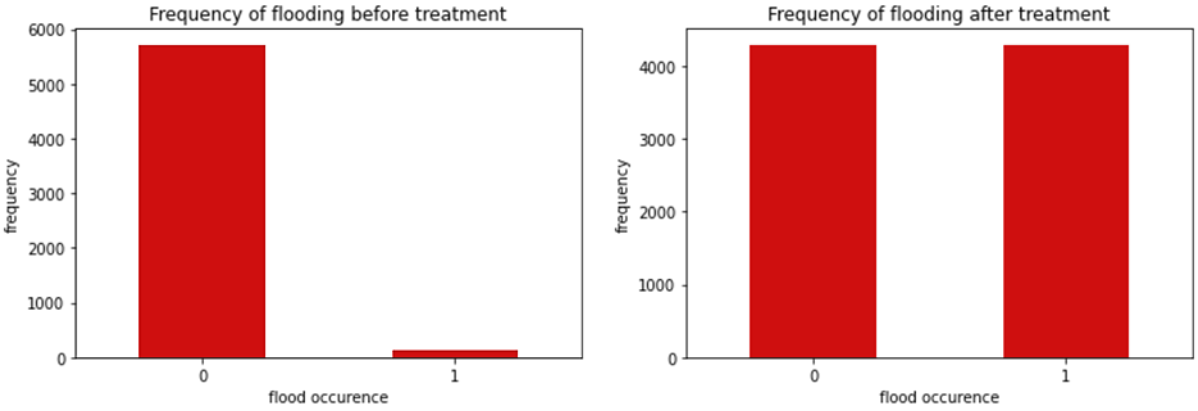
Based on the five meteorological hyperparameters that was given by MET Malaysia, 3 variables exhibit outliers value, which are maximum temperature, humidity, and rainfall. Shifting focus to skewness, the minimum temperature variable exhibits a slight negative skew. Conversely, both the minimum temperature and rainfall variables skewed positively. In contrast, humidity, and mean sea level (MSL) pressure variables adhere to normal distribution patterns. However, given the inherent nature of meteorological data, encountering extreme values is a customary occurrence that it has been determined the presence of outliers is to be anticipated and need not be removed, as they fall within the bounds of normality [21]. Additionally, similar situation is said for Southern Oscillation Index (SOI) and Nino Oceanic Index (ONI).

SOI is a standardized index based on the observed sea level pressure (SLP) differences between Tahiti and Darwin, Australia. The SOI is one measure of the large-scale fluctuations in air pressure occurring between the western and eastern tropical Pacific such as the state of the Southern Oscillation during El Niño and La Niña episodes [22]. The NINO3. 4 (ONI) index is defined as the average of SST anomalies over the region 5°N - 5°S and 170° - 120°W. The ONI is the rolling 3-month average temperature anomaly – difference from average – in the surface waters of the east-central tropical Pacific, near the International Dateline [23].

### *3.2 Data Balancing*

In the absence of any corrective measures, the accuracy and precision values may appear to be high. However, the predictions suffer from a fundamental bias in terms of class representation. This bias becomes evident when observing Figure 3(a), which illustrates the class frequencies prior to any intervention. The depicted bar chart underscores the significant imbalance among the classes,

indicating the need for corrective action to ensure unbiased predictions. In this context, the label '1' signifies 'flooding,' while '0' signifies 'no flooding.' Moving to Figure 3(b), the subsequent bar chart showcases the results post-intervention. Through the implementation of Synthetic Minority Over-sampling Technique (SMOTE), synthetic data has been generated to balance the classes. This rebalancing effort is particularly noticeable in the increased representation of '1' predictions.




(a) (b)

**Fig. 3.** Bar chart of balance of class for flood occurrences (a) Before treatment (b) After treatment

The study involves the deployment of prediction models utilizing both machine learning and deep learning techniques, both before and after the treatment process. This allows for a comparative analysis of accuracy. The observed discrepancy in accuracy levels provides insights into the efficacy of the treatment.

### *3.3 Feature Engineering*

Several tasks are addressed under this subtopic, including managing missing data, detecting anomalies like outliers and duplicated dates, feature creation and data normalization. Notably, there were 10 instances of missing values in the daily relative humidity column. Given that these missing entries accounted for less than 5%, they are regarded as not significant. Therefore, imputation was done to rectify the missing values by substituting it with the column's average daily relative humidity, amounting to 78.13%. In relation to this dataset, apart from missing values, no duplicate dates were discovered.

### *3.4 Long Short-Term Memory (LSTM) Network*

This study encompasses two LSTM (2) models that incorporate different types of hyperparameters with the same target variable. The first four model, used meteorological variables such as minimum temperature, maximum temperature, humidity, rainfall and MSL pressure while the second LSTM (2) model used indices variables SOI and ONI. Both of the models' performances were compared before and after treatment, to understand the effect of SMOTE in predicting accuracy.

Table 5 provides an overview of the results from the confusion matrix, with meteorological variables. Due to the maximum feasible AUC value, LSTM (2) with 200 epochs after treatment is the model that performs the best, as shown in Table 5.

**Table 5**
Performance results for LSTM models of meteorological variables before and after treatment

| Epoch | Treatment | Precision | Recall | F1-Score | Accuracy | AUC |
|---|---|---|---|---|---|---|
| Epoch = 100 | After | 0.6169 | 0. 9668 | 0. 6724 | 0.9349 | 0.9668 |
| | Before | 0.7640 | 0. 9571 | 0. 8329 | 0.9822 | 0.9571 |
| **Epoch = 200** | **After** | **0.6466** | **0.9266** | **0.7106** | **0.9555** | **0.9752** |
| | Before | 0.7290 | 0.9040 | 0.7905 | 0.9774 | 0.9195 |

*Bold values indicate best performing model

Table 6 provides an overview of the performance results, with SOI and ONI indices variables. Coincidentally, the highest value of AUC shows a better performing model aside the rest, that is LSTM (2) with 200 epochs before treatment.

**Table 6**
Performance result for LSTM models of SOI and ONI indices variables before and after treatment

| Epoch | Treatment | Precision | Recall | F1-Score | Accuracy | AUC |
|---|---|---|---|---|---|---|
| Epoch = 100 | After | 0.5465 | 0.5000 | 0.4091 | 0.9167 | 0.5165 |
| | Before | 0.2713 | 0.5208 | 0.3567 | 0.8958 | 0.5035 |
| **Epoch = 200** | After | 0.5465 | 0.5000 | 0.4091 | 0.8958 | 0.5165 |
| | **Before** | **0.5661** | **0.5417** | **0.4550** | **0.8958** | **0.5252** |

*Bold values indicate best performing model

In comparison of Table 5 and Table 6, overall performance of LSTM (2) that uses SOI and ONI indices variables had values of AUC around 0.5 whilst LSTM (2) that uses meteorological variables had values of AUC around 0.9. This indicates that LSTM (2) with meteorological variables is the best performing model compared to LSTM (2) with SOI and ONI indices variables. Other than this, LSTM (2) with 200 epochs performed better than 100 epochs. This indicate that LSTM (2) with more epochs allows the model more opportunities to learn from the data. Each epoch provides an additional pass through the entire training dataset, refining the model's weights and biases.

Other than that, Table 7 was formulated to facilitate a clearer examination of the percentage improvement in the performance of the top-performing LSTM (2) model before and after the treatment. The outcomes presented in Table 7 establish that applying the SMOTE treatment to the LSTM (2) model with 100 epochs has led to a reduction in its predictive capabilities. Specifically, the precision of the model has experienced a decline of 23.85%, moving from 0.7640 to 0.6169. Similarly, the accuracy has decreased from 0.9822 to 0.9349, indicating a deterioration of 5.06%. Interestingly, the AUC has demonstrated a slight increase of 1.00%, rising from 0.9571 to 0.9668. Overall, it can be deduced that the treatment has not significantly impacted the model's overall performance.

**Table 7**
Percentage of improvement LSTM (2) model using meteorological hyperparameters for 100 epochs before and after treatment

| Evaluation metrics | Before treatment | After treatment | Percentage of improvement (%) |
|---|---|---|---|
| Precision | 0.7640 | 0.6169 | -23.85 |
| Recall | 0.9571 | 0.9668 | 1.00 |
| F1-Score | 0.8329 | 0.6724 | -23.87 |
| Accuracy | 0.9822 | 0.9349 | -5.06 |
| AUC | 0.9571 | 0. 9668 | 1.00 |

Figure 4 graphs illustrate the connection between loss and accuracy for the developed LSTM (2) networks using meteorological hyperparameters both before and after treatment. In Figures 4(a) and 4(b) that displayed loss and accuracy plots before treatment reveals a consistent and fluid relationship between them, suggesting a potential inclination towards overfitting.

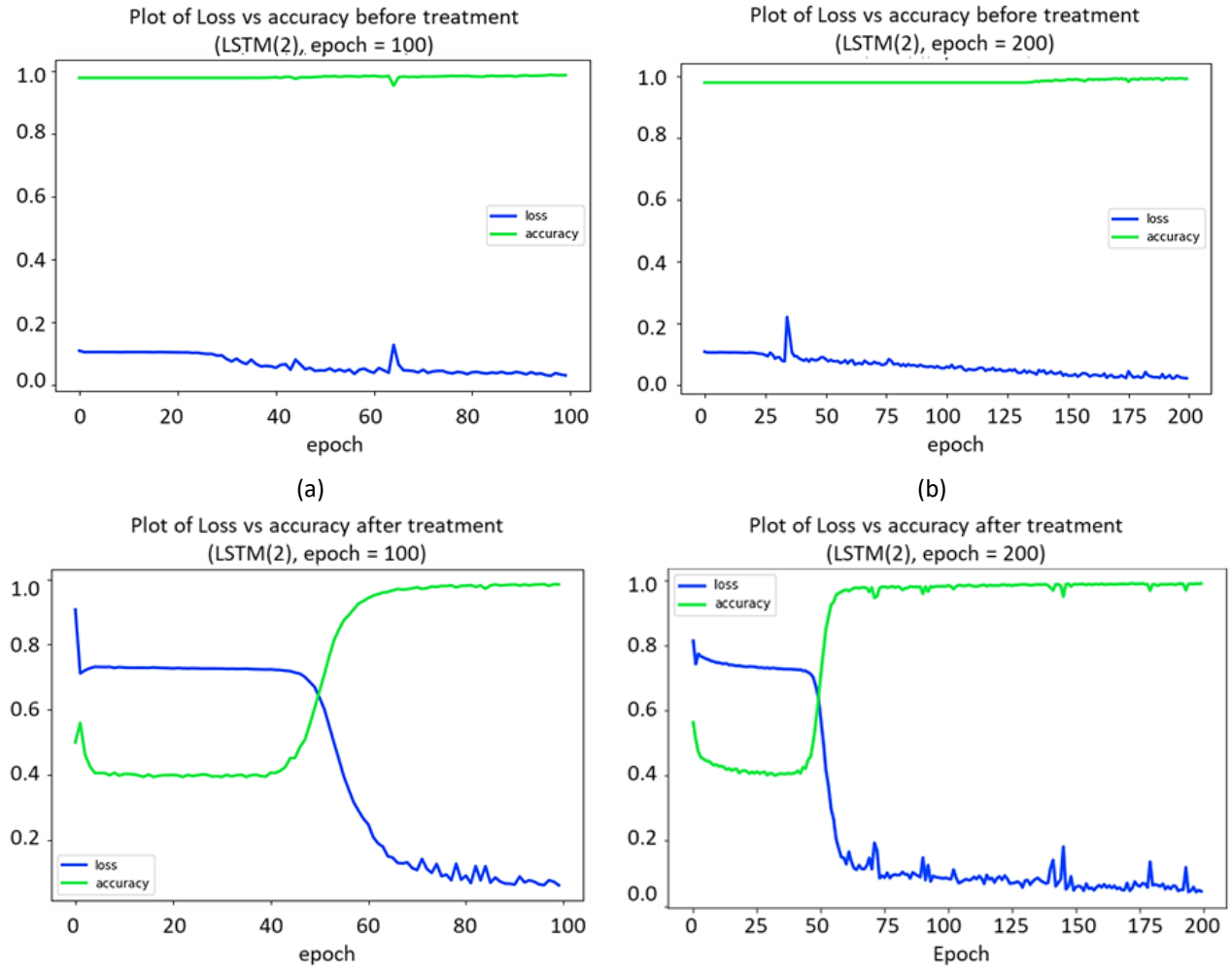



**Fig. 4.** Plot loss vs accuracy for LSTM (2) of different epochs before and after treatment using meteorological variables (a) Before = 100 (b) Before = 200 (c) After = 100 (d) After = 200

Conversely, plots for after treatment in Figures 4(c) and 4(d), the visuals indicate that the loss and accuracy metrics have converged at a specific epoch before diverging towards their maximum values. This pattern signifies successful model training on the dataset, resulting in notably high accuracy levels.

In comparison, Figure 5 graphs illustrate the connection between loss and accuracy for the developed networks using indices hyperparameters both before and after treatment. The performance of model in Figures 5(a) and 5(b), plots for before treatment showed a case of overfitting as the loss at the beginning is high and starts declining over time. Correspondingly, Figures 5(c) and 5(d) that plotted after treatment hyper-parameters indicates similar case of overfitting. All the plots described a similar pattern where it could stem from unclean data and noise in it.

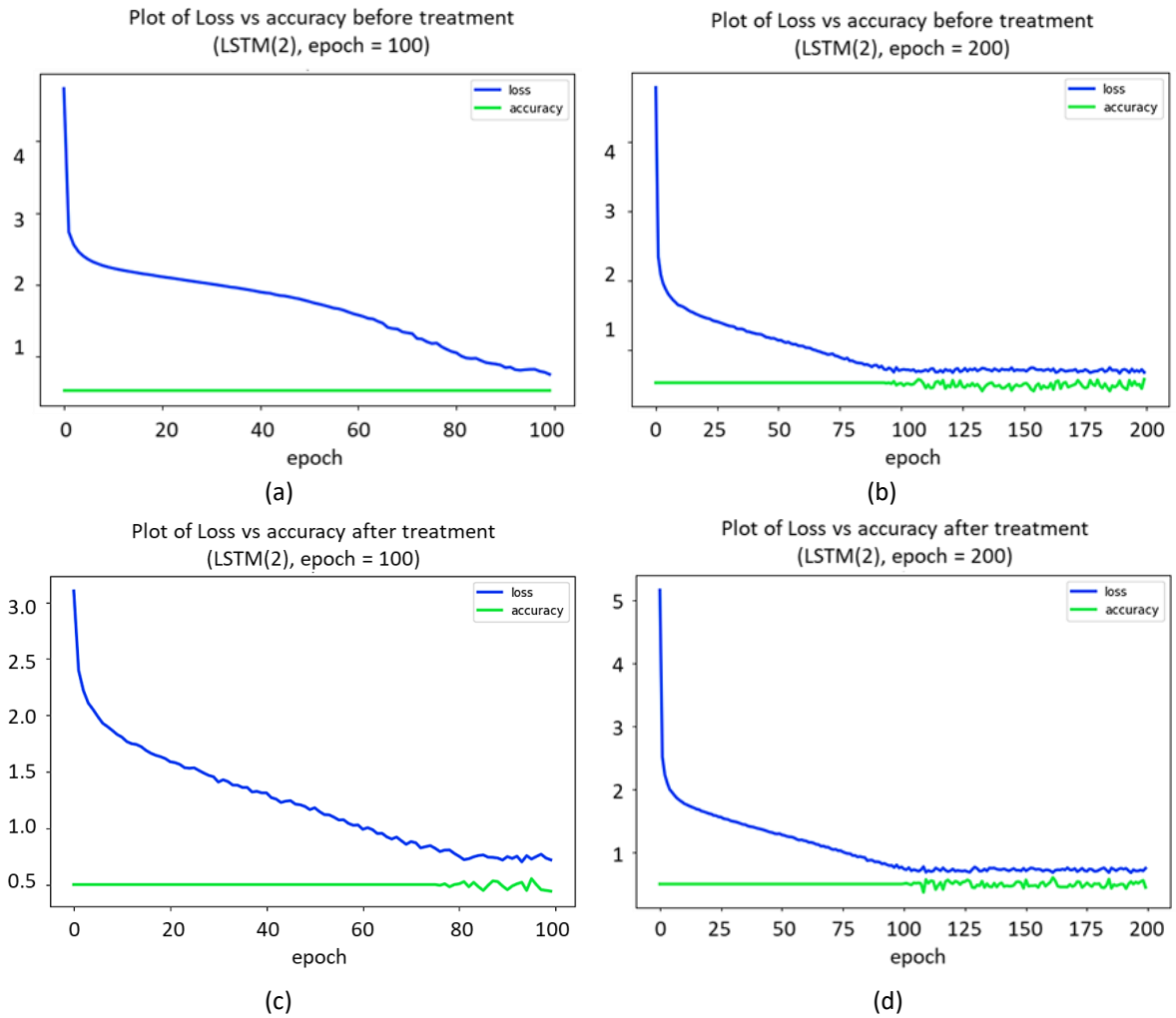



**Fig. 5.** Plot loss vs accuracy for LSTM (2) of different epochs before and after treatment using SOI and ONI indices variables (a) Before = 100 (b) Before = 200 (c) After = 100 (d) After = 200

In Figure 6, the ROC curves for LSTM (2) models with meteorological variables are depicted both before and after the treatment. Notably, all the ROC curves for different epochs of the LSTM (2) model exhibit commendable performance. Among these, the most exceptional performance is observed in the LSTM (2) model with 100 epochs after the treatment. This is evident from the steepness of its ROC curve, which nearly reaches the upper-left corner of the graph. This assertion is further substantiated by the AUC value of 0.9668, underscoring the superior performance of the LSTM (2) model with 100 epochs post-treatment.

Figure 7 shows the ROC curves for LSTM (2) models with indices hyperparameters for before and after treatment. Distinctly, all ROC curves for different epochs demonstrate unsatisfactory performance, as all the models had an AUC value encompassing 0.5. However, in comparison of these four models, the top-performing model is LSTM (2) with 200 epochs before treatment.

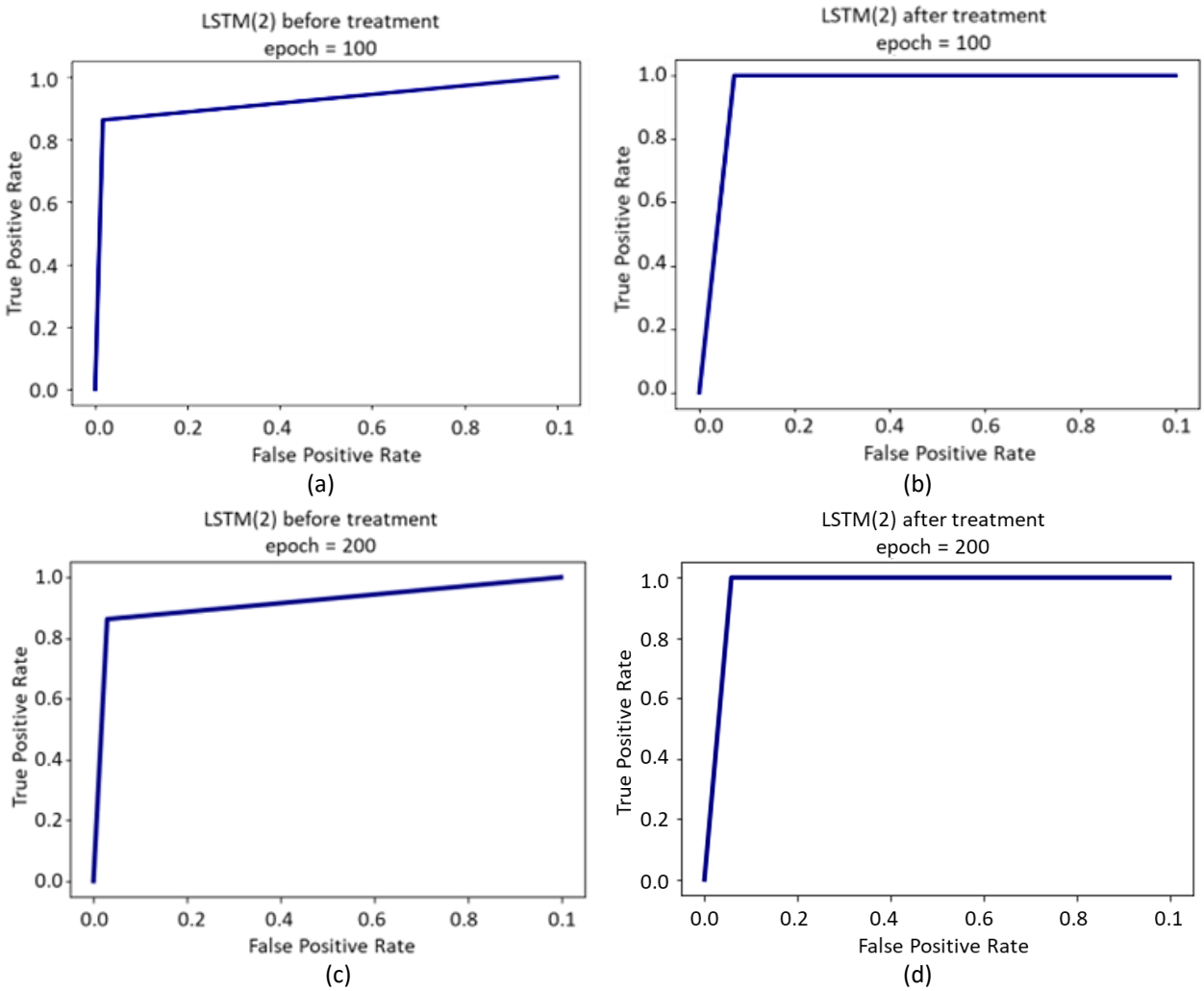
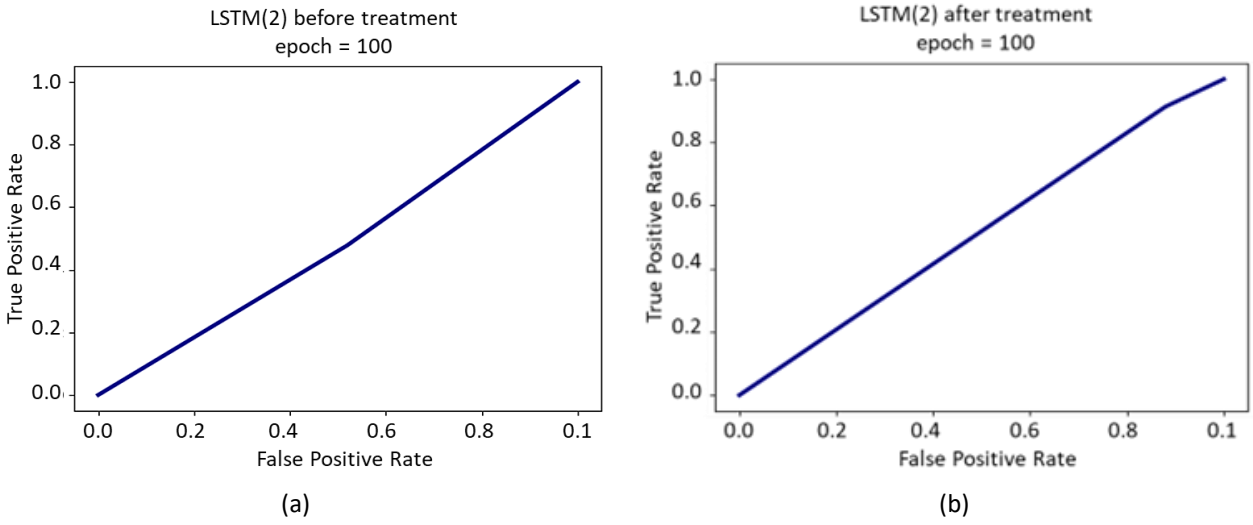
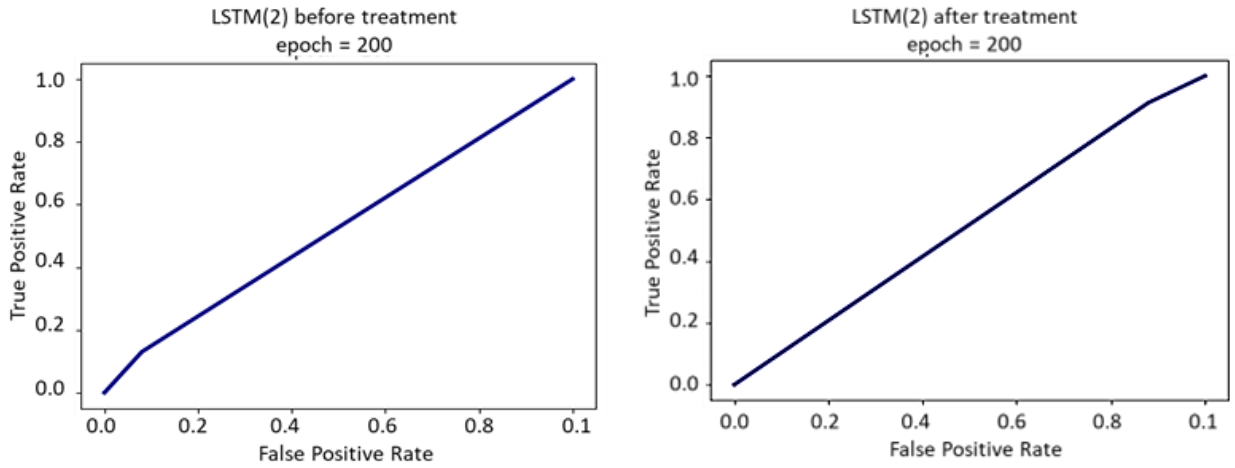



**Fig. 6.** ROC for LSTM (2) using meteorological variables of different epochs before and after treatment (a) Before = 100 (b) After = 100 (c) Before = 200 (d) After = 200






(c) (d)

**Fig. 7.** ROC for LSTM (2) using SOI and ONI indices of different epochs before and after treatment (a) Before = 100 (b) After = 100 (c) Before = 200 (d) After = 200

## 4. Conclusions

To conclude, the LSTM (2) model utilizing meteorological hyperparameters and trained for 100 epochs post-treatment is regarded as the most effective flood monitoring model among all the LSTM (2) models. It would be the best choice to construct flood predicting model for Subang Jaya area. This is because the LSTM (2) models have number of epochs that is more optimal than the other, with data that is treated for a better result. Besides, the accuracy for the LSTM (2) model is 0.9668 which is exceptionally good. Using SMOTE as a treatment for imbalanced data has also shown a significantly better accuracy in the models. The models' performances were evaluated by building the respective flood prediction models and constructing confusion matrices, ROC curve and AUC value.

In the future, researchers would be able to consider the computational time and complexity of the model when wanting to predict flooding using meteorological data for Subang Jaya area. The positive impact of using SMOTE in this research would also help researchers establish a better flood prediction model in the future without any biased results. Future researchers could also use a combination of different hyperparameters values for the models to give a better accuracy. Furthermore, researchers could experiment with number of hidden layers and nodes in the layer when using LSTM, network to produce a better prediction. Higher epochs and batch size might also help to increase the accuracy of a neural network.

## Acknowledgement

This research was supported by Ministry of Higher Education (MOHE) through Fundamental Research Grant Scheme (FRGS/1/2022/ICT06/UTHM/03/1).

## References

[1] Rahman, Ismail Abdul, and Jonathan Dewsbury. "Selection of typical weather data (test reference years) for Subang, Malaysia." *Building and Environment* 42, no. 10 (2007): 3636-3641. https://doi.org/10.1016/j.buildenv.2006.10.004

[2] Saimi, Farahani Mohd, Firdaus Mohamad Hamzah, Mohd Ekhwan Toriman, Othman Jaafar, and Hazrina Tajudin. "Trend and linearity analysis of meteorological parameters in peninsular Malaysia." *Sustainability* 12, no. 22 (2020): 9533. https://doi.org/10.3390/su12229533

[3] Sugam, Rudresh Kumar, Md Humayain Kabir, Sherin Shiny George, and Mayuri Phukan. "Integrated water resources management and flood risk management: Opportunities and challenges in developing countries." *Handbook of Flood Risk Management in Developing Countries* (2023): 249-265.

[4] Razi, Mohd Adib Mohammad, Wardah Tahir, Noratina Alias, Lokman Hakim Ismail, and Junaidah Ariffin. "Development of rainfall model using meteorological data for hydrological use." *International Journal of Integrated Engineering* 5, no. 1 (2013): 64-73.
[5] World Meteorological Organization. *Manual on flood forecasting and warning*. World Meteorological Organization, 2011.
[6] Milutinović, Slobodan, and Snežana Živković, eds. *Prevention and Management of Soil Erosion and Torrential Floods*. Engineering Science Reference, 2022.
[7] Ristić, Ratko, Stanimir Kostadinov, Biljana Abolmasov, Slavoljub Dragićević, Goran Trivan, Boris Radić, M. Trifunović, and Zoran Radosavljević. "Torrential floods and town and country planning in Serbia." *Natural Hazards and Earth System Sciences* 12, no. 1 (2012): 23-35. https://doi.org/10.5194/nhess-12-23-2012
[8] Demeritt, David, Hannah Cloke, Florian Pappenberger, Jutta Thielen, Jens Bartholmes, and Maria-Helena Ramos. "Ensemble predictions and perceptions of risk, uncertainty, and error in flood forecasting." *Environmental Hazards* 7, no. 2 (2007): 115-127. https://doi.org/10.1016/j.envhaz.2007.05.001
[9] Syeed, Miah Mohammad Asif, Maisha Farzana, Ishadie Namir, Ipshita Ishrar, Meherin Hossain Nushra, and Tanvir Rahman. "Flood prediction using machine learning models." In *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, pp. 1-6. IEEE, 2022. https://doi.org/10.1109/HORA55278.2022.9800023
[10] Bi, Qifang, Katherine E. Goodman, Joshua Kaminsky, and Justin Lessler. "What is machine learning? A primer for the epidemiologist." *American journal of epidemiology* 188, no. 12 (2019): 2222-2239. https://doi.org/10.1093/aje/kwz189
[11] Mosavi, Amir, Pinar Ozturk, and Kwok-wing Chau. "Flood prediction using machine learning models: Literature review." *Water* 10, no. 11 (2018): 1536. https://doi.org/10.3390/w10111536
[12] Moishin, Mohammed, Ravinesh C. Deo, Ramendra Prasad, Nawin Raj, and Shahab Abdulla. "Designing deep-based learning flood forecast model with ConvLSTM hybrid algorithm." *IEEE Access* 9 (2021): 50982-50993. https://doi.org/10.1109/ACCESS.2021.3065939
[13] Kratzert, Frederik, Daniel Klotz, Claire Brenner, Karsten Schulz, and Mathew Herrnegger. "Rainfall–runoff modelling using long short-term memory (LSTM) networks." *Hydrology and Earth System Sciences* 22, no. 11 (2018): 6005-6022. https://doi.org/10.5194/hess-22-6005-2018
[14] Danladi, Amos, Ho Chin Siong, and Gabriel Ling Hoh Teck. "Importance of indigenous knowledge in flood risk reduction: A review." *Journal of Advanced Research in Applied Sciences and Engineering Technology* 11, no. 1 (2018): 7-16.
[15] Nurumal, Mohd Said, Khin Thandar Aung, and Nurul Syamimi Md Yusoff. "Community experiences at the aftermath of flood disaster based on cultural context." *Journal of Advanced Research in Social and Behavioural Sciences* 8, no. 1 (2017): 89-96.
[16] Bekar, Ebru Turanoglu, Per Nyqvist, and Anders Skoogh. "An intelligent approach for data pre-processing and analysis in predictive maintenance with an industrial case study." *Advances in Mechanical Engineering* 12, no. 5 (2020): 1687814020919207. https://doi.org/10.1177/1687814020919207
[17] Jamaluddin, Ahmad Fairudz, Muhammad Ikmalnor Mustafa Kamal, Muhammad Helmi Abdullah, and Amirul Nizam Marodzi. "Comparison between Satellite-Derived Rainfall and Rain Gauge Observation over Peninsular Malaysia." *Sains Malaysiana* 51, no. 1 (2022): 67-81. http://doi.org/10.17576/jsm-2022-5101-06
[18] Gulli, Antonio, and Sujit Pal. *Deep learning with Keras*. Packt Publishing Ltd, 2017.
[19] Godoy, Daniel. "Understanding binary cross-entropy/log loss: A visual explanation." *Towards Data Science* 21 (2018).
[20] Sellami, E. M., Mehdi Maanan, and Hassan Rhinane. "Performance of machine learning algorithms for mapping and forecasting of flash flood susceptibility in Tetouan, Morocco." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 46 (2022): 305-313. https://doi.org/10.5194/isprs-archives-XLVI-4-W3-2021-305-2022
[21] Zhang, A. Z., Li, J. Z., Gao, H., Chen, Y. B., Ma, H. Z., and Bah, M. J. "CrowdOLA: Online aggregation on duplicate data powered by crowdsourcing." *Journal of Computer Science and Technology* 33, no. 2 (2017): 366-379. https://doi.org/10.1007/s11390-018-1824-5
[22] NOAA. (2023). "El Niño/Southern Oscillation (ENSO): Southern Oscillation Index (SOI). *National Centers for Environmental Information*, US, 2024.
[23] Wang, Chunzai. "Three-ocean interactions and climate variability: a review and perspective." *Climate Dynamics* 53, no. 7 (2019): 5119-5136. https://doi.org/10.1007/s00382-019-04930-x