Rainfall Prediction Due to El Nino Factors Using Recurrent Neural Networks

Rizki Fadilah, Esmeralda C. Djamal, Ridwan Ilyas

Department of Informatics, Universitas of Jenderal Achmad Yani – Jawa Barat

Email: rizki.fdz@gmail.com, esmeralda.contessa@lecture.unjani.ac.id, ilyas@lecture.unjani.ac.id

Abstract. El Nino is one of the natural phenomena that have a significant influence on the weather, causing a longer dry season in several regions of Indonesia, one of which is the city of Lampung. One way to anticipate a long drought is to predict rainfall, by looking at the intensity of the rain. This paper proposes rainfall prediction using a recurrent neural network. Weather variables used to predict rainfall include air humidity, wind speed obtained from BMKG stations, and SOI index obtained from the NCDC website in the past 10 years. Weather data will be interpolated and extracted to find the maximum weather value per 4 weeks, the next step is overlapping, after which the data segmentation and normalization become 0-1 to make the data values not far adrift. The results showed the prediction of rainfall with a vulnerable 4 weeks using the Recurrent Neural Networks method produces an accuracy of 89.53%.

Keywords: BMKG, GRU, NCDC, Prediction, Rainfall, Recurrent Neural Networks.

1. Introduction

Indonesia is a country with a tropical climate. An area with a tropical climate has a fairly high rainfall intensity. Rainfall is one of the climate variables that play a role in several sectors such as paddy fields, animal feed and other food production materials [1]. However, Indonesia currently has low rainfall, so that some of its regions are hit by prolonged drought. One of the causes of this extreme climate is the El Nino phenomenon [2]. Generally related to the Sea Surface Temperature (SST) anomaly, it is intended as an additional variable from the El Nino factor namely the Southern Oscillation Index (SOI). Factors for rainfall are analyzed from the atmosphere, which is characterized by air humidity and wind speed [3].

However climatic conditions are influenced by global phenomena of world climate patterns. Decadal aspects such as ENSO (El Nino Southern Oscillation) can affect climate change in Indonesia nearly 70% of the Maritime Continent [4]. This phenomenon causes sea surface temperatures in the eastern Pacific Ocean to heat up, causing seawater in the western Pacific Ocean to cool and cause the process of cloud formation to slow, and most of Indonesia receives less rainfall. This condition causes a long drought.

A rainfall is one of the climate parameters that can anticipate the effects of drought due to El Nino factors [5]. Rainfall forecast is important to overcome the effects of prolonged drought. Therefore it is important to predict rainfall effectively because rainfall has an uncertain pattern so it is difficult to predict manually. Previous research on rainfall, when viewed in terms of time, can use short-term time predictions [6], daily [7], Long-term monthly and annual predictions [8]. Other studies also predict the intensity of rainfall that falls [9], predict rainfall to drough prevention [10], and predict rainfall to
flood prevention [11].

An increasingly advanced development makes the machine can learn data patterns, such as sequential data or time series. Some methods that have been used to predict rainfall are Support Vector Machine (SVM) with period of five years and get prediction results above 75.62% [12], rainfall prediction using Backpropagation Artificial Neural Network (ANN) with an accuracy of 95.82% for the train data [13], and rainfall prediction using Least Square Support Vector Machine (LSSVM) methods with accuracy of 85.60% [14].

Deep Learning is one branch of machine learning that works like the human brain. Previous studies predicting rainfall using the methods of Wavelet Regression [15], Backpropagation [13], Convolutional Neural Networks (CNN) [16], and Recurrent Neural Networks (RNN) [17]. Rainfall prediction using RNN has a higher level of accuracy than other learning methods [18]. Previous studies using the RNN method for rainfall prediction got accuracy 93.92% of the test data [19]. Other studies also prove that RNN is suitable for rainfall prediction [20]. The RNN method has a short-term memory caused by a Vanishing Gradient or swollen gradient value [21]. To overcome the problem of exploded gradient values, long-term memory is needed that can handle the time series data problem [22]. RNN has several variations, including Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU). LSTM and GRU are designed to capture more long-term memory so that they can become long-term memory inputs more easily. The difference between LSTM and GRU is that the structure of GRU is simpler than LSTM [23]. In this study, a model was made to predict rainfall due to El Nino factor using the RNN and GRU methods. Climate data used are air humidity, wind speed, rainfall and SOI index obtained from BMKG at Lampung Meteorological Station by taking climate data for the last ten years (2010-2019 period). Rainfall prediction due to El-nino factors is carried out on a weekly basis that is four weeks, with the output in the form of four rainfall ranges that have been determined by BMKG, namely "Light" (5mm-20mm), "Medium" (20mm-50mm), "Heavy" (50mm- 100mm) and "Very Heavy" (> 100mm).

2. Methods

This study uses a collection of weather data such as rainfall, humidity, wind speed, and SOI index. The weather data was collected by the Meteorology, Climatology and Geophysics Agency (BMKG), the Lampung Meteorological Station, and an additional SOI index variable obtained from the official National Data Climatic Center (NCDC) website. Previous studies, predictions of rainfall due to El Nino factors during the last 10 years [12]. This research predicts rainfall due to El Nino factors with three other weather parameters over the past ten years with the period 2010-2019, as shown in Table 1.

| Day - Date | Air Humidity | Wind Speed | Rainfall | SOI |
|------------|--------------|------------|----------|-----|
| 1 01/01/2010 | 90           | 1          | 0.4      | 1.1 |
| 2 02/01/2010 | 8888         | 2          | 14       | 1.1 |
| 3 03/01/2010 | 75           | 3          | 21.4     | 1.1 |
| ...         | ...          | ...        | ...      | ... |
| 3650 31/12/2019 | 77        | 2          | 9.2      | -0.8 |

Table 1 indicates that data is blank or data that is not measurable is interpolated, where it is initialized with a value of 8888 for data not measured and 9999 for data is empty. Then the preprocessing is done for each weather parameter.
Fig. 1. Rainfall Prediction Due to El Nino Factors Using Recurrent Neural Networks

Rainfall prediction model due to El Nino factor can be seen in Fig. 1, daily weather data from BMKG will be converted into four weeks data with interpolation and feature extraction stages, then segmented overlapping, and later will be normalized so that the data do not have different values, namely by the value range is 0-1.

2.1. Preprocessing Data

a) Data Loss Treatment

Weather data such as rainfall, humidity, wind speed and SOI that is used is that in Table 1 there is missing or unmeasured data. Therefore interpolation is done to repair lost data. The results of interpolation are shown in Table II.

b) Period Convert

The feature extraction process is finding the maximum value every four weeks from each climate variable such as air humidity, wind speed, rainfall and SOI. The results of interpolation can be seen in Table 2.

c) Segmentation

This study uses climate data that is grouped by a certain time which is divided into small parts to be used as training data, which in a training data set consists of four weeks which are overlapped, as in Fig. 2.

Fig. 2 shows the first data set, the data began to be filled in from parameter 1, namely Air Humidity (AH), which will be sorted from week 1 to week 4. Then it is filled with three other parameters namely Wind Speed (WS), SOI and Rainfall (RF) until the 4th week, and so on until the 128 dataset 2,205 weeks to the 2,208 weeks or for ten years, such as shown in Fig. 2 and Table II.

a) Normalization

The climatic parameters used such as rainfall, humidity, wind speed and SOI index, have different values. Therefore, normalization is needed to correct different values to the same value, which is 0-1. As shown (1) and Table 2.

\[
Z = \frac{x - \text{min}()}{\text{max}() - \text{min}()} 
\]

Where Z is normalized, x is the data to be normalized, min () is the lowest data in the column, and max () is the highest data in the column.
Table 2. Period Convert, Segmentation and Normalization of Climate Data

| Day to - | The Order of Training Data on Climate Parameters |
|---------|--------------------------------------------------|
| AH1     | AH4     | RF1     | RF4     |
| 1       | 0.63    | …       | 0.73    | …       | 0.45    | …       | 0.21    |
| 2       | 0.41    | …       | 0.78    | …       | 1       | …       | 0.66    |
| 127     | 0.73    | …       | 0.67    | …       | 0.48    | …       | 0.69    |
| 128     | 0.24    | …       | 0.43    | …       | 0.41    | …       | 0.59    |

Table II shows the results of the feature extraction calculations for four weeks, overlapping segmentation, and normalization into data with a range of 0-1.

2.2. Recurrent Neural Networks

RNN is the development of the Feedforward Neural Network by characterizing using feedback from the output as input. The output of the RNN depends not only on the initial input but also on the previous network state which acts as memory [24]. RNN has an architecture that can be used for sequence or list data as shown in Fig. 3. Recurrent neurons in RNN are similar to neurons in the Artificial Neural Network (ANN). RNN has a simple activation structure, for example $h = \tanh (Wx + b)$ where W is the output, O is the input, U is the weight, V is the weight of the matrix, and S is the bias.

$$h^{(t)} = \tanh (Wx^{(t)} + Wh^{(t-1)} + b)$$ (2)

$$o^{(t)} = \frac{1}{1 + \exp(-h^{(t)})}$$ (3)

The RNN architecture is almost the same as MLP architecture, but with more complexity for transient data processing. One of the RNN architectures is Gated Recurrent Unit (GRU). GRU is a simplification of the LSTM cell structure [25]. The function of GRU is to make each recurrent unit adaptively capture dependencies at different time scales. The information flow regulator component in the GRU is referred to as the gate and GRU has two gates, namely reset gate and gate update. Resetting the gate at GRU will determine how to merge new inputs with past information, and gate updates will determine how much past information should be kept. GRU is similar to LSTM but is simpler to calculate and implement. The architecture of the GRU is shown in Fig. 4.
A typical GRU cell consists of two gates: reset gate = r, update gate = z and has an output vector = h. Similar to LSTM cells, the hidden layer’s output at time t is calculated using the hidden layer time t-1. For the gate update calculation process can be seen in (4). After calculating the gate update function, a calculation is performed to sort out which information can be discarded or stored using the function in (5). Then the information stored by the reset gate function will be the output vector and calculated using (6).

$$r = \sigma_{gg} (W_{hr}X_{lt} + W_{hh}h_{lt-1} + b_{hh})$$  \hspace{0.5cm} (4) \\
z = \sigma_{zz} (W_{hz}X_{lt} + W_{hh}h_{lt-1} + b_{hh})$$  \hspace{0.5cm} (5) \\
h_{lt} = (1 - z_{lt}) \times h_{lt-1} + z_{lt} \times \phi_h(W_{hh}X_{lt} + W_{hh}h_{lt-1} + b_h)$$  \hspace{0.5cm} (6)

3. Result and Discussion

The training data used in this study were obtained from one weather observation station with a span of ten years from 2010-2019 using four weather parameters, namely air humidity, wind speed, rainfall, and SOI index. Climate data is divided into two parts, 70% of training data and 30% of test data. The test consists of different optimization models, namely Adaptive Moment Estimation (Adam) and Stochastic Gradient Descent (SGD) with a learning rate of 0.001. The amount of data and RNN configuration, which will later produce four rainfall ranges.

3.1. Amount of Data Testing

This research develops rainfall prediction models due to El Nino factors by using historical climate data for the past five years and ten years. Testing of training data on non-training data was carried out to determine the effect of configuration on the quality of learning produced using two different optimization models, namely Adam and SGD 500 epoch, and by using RNN and GRU configurations. The amount of training data is used as a comparison between the last five years and ten years, as shown in Table 3.

| No | History | Training Data | Testing Data |
|----|---------|---------------|--------------|
|    |         | Loss | Accuracy (%) | Loss | Accuracy (%) |
| 1. | ten years | 0.0011 | 89.53 | 0.0019 | 86.41 |
| 2. | five years | 0.0032 | 74.12 | 0.0043 | 69.77 |

Table 3 shows the accuracy of the data set for ten years is 89.53% higher than the training data, and 86.41% of the test data. Whereas with a history of data for 5 years resulted in an accuracy of 74.12% of the training data and 69.77% of the test data. The number of historical data sets that are trained will affect the accuracy that is generated.

3.2. Comparison Between LSTM and GRU

This study predicts rainfall with an additional El Nino phator using the amount of data from the past
ten years. In the RNN there are several different variations of the RNN, including GRU and LSTM. Both variations of the RNN have been tested using Adam's optimization model with 500 epochs and the results can be seen in Table 4.

| No | Variation | Testing Data | Learning Time |
|----|-----------|--------------|---------------|
| 1. | GRU       | 0.0019       | 2.21 Minute   |
| 2. | LSTM      | 0.0009       | 3.26 Minute   |

In table 5 we can see that the accuracy produced by the GRU variant has a faster learning time than LSTM. This is because GRU has a simpler architecture than LSTM because GRU is a simplification of LSTM. But despite having a faster learning time, the accuracy generated by LSTM is greater than GRU. Although it requires a longer learning time, LSTM produces greater accuracy than GRU.

3.3. Optimization Model Testing

This research was conducted by comparing two optimization models, namely Adaptive Moment Estimation (Adam) and Stochastic Gradient Descent (SGD), and use GRU variation. The two optimization models are used to find out which optimization model produces the best learning and accuracy.

| No | History | Training Data | Testing Data |
|----|---------|---------------|--------------|
| 1. | Adam    | 0.0012        | 0.0019       |
|    |         | 89.53         | 86.41        |
| 2. | SGD     | 0.0026        | 0.0038       |
|    |         | 81.34         | 59.63        |

Based on Table 5 the use of different optimization models affects the level of prediction accuracy. Accuracy and loss models resulting from Adam and SGD optimization models show that Adam's optimization model is better than SGD. This aspect is based on accuracy such as Fig. 5 and Fig. 6.

From Fig. 5 it can be seen that Adam's optimization model has greater accuracy compared to SGD. This is because when conducting training using SGD, the training data used by SGD is taken randomly so that it has inconsistent results and does not represent the whole class.
Fig. 6. Model Losses of Adam and SGD

In Fig. 6 we can see that Adam has a small loss, while SGD has a loss that tends to be large. This is because when training Adam did randomization of training data consistently, so that all training data were trained and produced a smaller and more stable loss. In contrast to Adam, SGD does randomization of training data not thoroughly, although it produces a stable loss but has a large enough value.

4. Conclusion

In this research, a computational model has been made using the RNN variant namely GRU with Adam optimization model and learning rate of 0.001 with 500 epochs. The results obtained for 89.53% for training data and 86.41% for test data with a range of historical data used 10 years. GRU has a faster learning time of 2.21 minutes, but LSTM has greater accuracy than GRU. In addition to the variants and amount of data used, the selection of optimization models also influences the accuracy. Adam's optimization works better than SGD because Adam has better performance when doing the learning process by randomizing training data consistently.

References

[1] Q. Y. Feng et al., “Climate Learn: A Machine-Learning Approach For Climate Prediction using Network Measures,” Geosci. Model Dev. Discuss., no. February, pp. 1–18, 2016.
[2] X. Yu, Z. Wang, H. Zhang, and S. Zhao, “Impacts of Different Types and Intensities of El Niño Events on Winter Aerosols Over China,” Sci. Total Environ., vol. 655, pp. 766–780, 2019.
[3] L. Wang, Y. Zeng, and T. Chen, “Backpropagation Neural Network With Adaptive Differential Evolution Algorithm For Time Series Forecasting,” Expert Syst. Appl., vol. 42, no. 2, pp. 855–863, 2015.
[4] Supari, F. Tangang, E. Salimun, E. Aldrian, A. Sopaheluwakan, and L. Juneng, “ENSO Modulation of Seasonal Rainfall and Extremes in Indonesia,” Clim. Dyn., vol. 51, no. 7–8, pp. 2559–2580, 2018.
[5] L. Mareta, R. Hidayat, R. Hidayati, and G. Alsepan, “Influence of the Positive Indian Ocean Dipole in 2012 and El Nino-Southern Oscillation (ENSO) in 2015 on the Indonesian Rainfall Variability,” IOP Conf. Ser. Earth Environ. Sci., vol. 284, no. 1, pp. 1–12, 2019.
[6] Y. Qin, D. Song, H. Cheng, W. Cheng, G. Jiang, and G. W. Cottrell, “A Dual-Stage Two-Phase Attention-based Recurrent Neural Networks For Long-Term and Multivariate Time Series Prediction,” IJCAI Int. Jt. Conf. Artif. Intell., vol. 0, no. 0, pp. 2627–2633, 2017.
[7] K. Kaba, M. Sargül, M. Avcı, and H. M. Kandırmaz, “Estimation of Daily Global Solar Radiation Using Deep Learning Model,” Energy, vol. 162, pp. 126–135, 2018.
[8] F. K. Muthoni et al., “Long-Term Spatial-Temporal Trends and Variability of Rainfall Over Eastern and Southern Africa,” Theor. Appl. Climatol., vol. 0, no. 0, pp. 1–14, 2018.
[9] S. Poornima and M. Pushpalatha, “Prediction of Rainfall Using Intensified LSTM Based Recurrent Neural Network with Weighted Linear Units,” Atmosphere (Basel.), vol. 10, no. 11, pp. 1–18, 2019.
[10] L. Ni et al., “Streamflow and Rainfall Forecasting By Two Long Short-Term Memory-based
Models,” *J. Hydrol.*, vol. 583, p. 124296, 2020.

[11] E. Khosla, D. Ramesh, R. P. Sharma, and S. Nyakotey, “RNNs-RT: Flood based Prediction of Human and Animal deaths in Bihar using Recurrent Neural Networks and Regression Techniques,” *Procedia Comput. Sci.*, vol. 132, no. Iccids, pp. 486–497, 2018, doi: 10.1016/j.procs.2018.05.001.

[12] Y. Radhika and M. Shashi, “Atmospheric Temperature Prediction using Support Vector Machines,” *Int. J. Comput. Theory Eng.*, vol. 1, no. 1, pp. 55–58, 2009.

[13] K. Abhishek, A. Kumar, R. Ranjan, and S. Kumar, “A Rainfall Prediction Model Using Artificial Neural Network,” *Proc. - 2012 IEEE Control Syst. Grad. Res. Colloquium, ICSGRC 2012*, no. Icsgrc, pp. 82–87, 2012.

[14] J. Farajzadeh and F. Alizadeh, “A Hybrid Linear–Nonlinear Approach To Predict The Monthly Rainfall Over The Urmia Lake Watershed Using Wavelet-SARIMAX-LSSVM Conjugated Model,” *J. Hydroinformatics*, vol. 20, no. 1, pp. 221–231, 2018.

[15] M. K. Goyal, “Monthly Rainfall Prediction Using Wavelet Regression and Neural Network,” *Theor. Appl. Climatol.*, vol. 118, no. 1–2, pp. 25–34, 2014.

[16] G. Swapna, K. P. Soman, and R. Vinayakumar, “Automated Detection of Diabetes Using CNN and CNN-LSTM Network and Heart Rate Signals,” *Procedia Comput. Sci.*, vol. 132, no. Iccids, pp. 1253–1262, 2018.

[17] Z. Liu and C. J. Sullivan, “Prediction of Weather Induced Background Radiation Fluctuation with Recurrent Neural Networks,” *Radiat. Phys. Chem.*, vol. 155, no. 0, pp. 275–280, 2019.

[18] R. P. Deshmukh and A. A. Ghatol, “Short Term Flood Forecasting Using Recurrent Neural Networks a Comparative Study,” *Int. J. Eng. Technol.*, vol. 2, no. 5, pp. 430–434, 2010, doi: 10.7763/ijet.2010.v2.160.

[19] W. Li, A. Kiaghadi, and C. Dawson, “High Temporal Resolution Rainfall–Runoff Modeling Using Long-Short-Term-Memory (LSTM) Networks,” *Neural Comput. Appl.*, vol. 6, no. 0, pp. 1–18, 2020.

[20] M. A. Zaytar and C. El Amrani, “Sequence to Sequence Weather Forecasting with Long Short-Term Memory Recurrent Neural Networks,” *Int. J. Comput. Appl.*, vol. 143, no. 11, pp. 975–8887, 2016.

[21] D. Zhang and M. R. Kabuka, “Combining weather condition data to predict traffic flow: A GRU- based deep learning approach,” *IET Intell. Transp. Syst.*, vol. 12, no. 7, pp. 578–585, 2018.

[22] F. A. Gers, J. Schmidhuber, and F. Cummins, “Learning to Forget: Continual Prediction with LSTM,” *Neural Comput.*, vol. 12, no. 10, pp. 2451–2471, 2000.

[23] X. Zhang, C. Li, and Y. Morimoto, “A Multi-factor Approach for Stock Price Prediction by using Recurrent Neural Networks,” *Bull. Networking, Comput. Syst. Softw.*, vol. 8, no. 1, pp. 9–13, 2019.

[24] Z. Abbas, A. Al-Shishawy, S. Girdzijauskas, and V. Vlassov, “Short-Term Traffic Prediction Using Long Short-Term Memory Neural Networks,” *Proc. - 2018 IEEE Int. Congr. Big Data, BigData Congr. 2018 - Part 2018 IEEE World Congr. Serv.*, vol. 0, no. 0, pp. 57–65, 2018.

[25] S. Kachwala, M. Jha, D. Shah, U. Shinde, and H. Namdeo Bhor, “Predicting Rainfall from Historical Data Trends,” *SSRN Electron. J.*, vol. 0, no. 0, p. 0, 2020.