Sunspot Number Prediction Using Gated Recurrent Unit (GRU) Algorithm

Unix Izyah Arfianti[1], Dian Candra Rini Novitasari*[2], Nanang Widodo[3], Moh. Hafiyusholeh[4], Wika Dianita Utami [5]
[1,2,4,5]Department of Mathematics, UIN Sunan Ampel, Surabaya, Indonesia
[3]Balai Pengamatan Antariksa dan Atmosfer, Pasuruan, Indonesia
e-mail: [1]unixizyah@gmail.com, ***[2]diancrini@uinsby.ac.id,** [3]nangwidodo@gmail.com, [4]hafiyusholeh@uinsby.ac.id, [5]wikadianita@uinsby.ac.id

Abstrak

Sunspot sangat penting untuk diteliti karena bilangan sunspot menunjukkan tingkat aktivitas di matahari. Tujuan dari penelitian ini adalah untuk memprediksi bilangan sunspot menggunakan algoritma Gated Recurrent Unit (GRU) agar dapat mengetahui informasi dini mengenai bilangan sunspot pada masa yang akan datang, sehingga jika terjadi peningkatan yang signifikan bilangan sunspot dapat diinformasikan akibat fisis lain yang mungkin akan ditimbulkan. GRU merupakan modifikasi dari metode Long short-term Memory (LSTM), informasi dari memory sebelumnya diproses melalui dua gate, update gate dan reset gate, kemudian output yang dihasilkan akan menjadi input untuk proses selanjutnya. Data yang digunakan yaitu data bilangan sunspot per bulan diperoleh dari website SILSO. Penelitian ini menggunakan pembagian data dan parameter berdasarkan uji coba dan akan dibandingkan dengan LSTM. Nilai MAPE terbaik yang didapatkan adalah 7.171% dengan pembagian data 70:30, hidden layer 150, batch size 32, and learning rate drop 100 menggunakan GRU dan 9.9557% dengan pembagian data 70:30, hidden layer 150, batch size 128, dan learning rate drop 150 menggunakan LSTM. Prediksi bilngan sunspot menggunakan algoritma LSTM mendapatkan akurasi yang sangat bagus karena nilai MAPE kurang dari 10%, tetapi GRU lebih baik dari LSTM dengan selisih nilai MAPE 2.7847%.

Kata kunci— *prediksi, bilangan sunspot, time series, GRU, LSTM.*

Abstract

Sunspot is very important to be researched because sunspot numbers present the level of solar activity. This research was conducted to predict sunspot numbers using Gated Recurrent Unit (GRU) algorithm to find out the information of sunspot numbers early, so that if there is a significant increase of sunspot numbers, it can inform other physical consequences that may be caused. GRU is modification of Long short-term Memory (LSTM) method: the information from the previous memory is processed through two gates, those are update gate and reset gate, then the output generated will be input for the next process. The data used was the data of monthly sunspot numbers obtained from SILSO website. This research uses data division and parameters based on trials then will be compared by LSTM. The best MAPE value obtained was 7.171% with 70:30 data division, 150 hidden layers, 32 batch size, and 100 learning rate drop using GRU and 9.9557% with data division 70:30, 150 hidden layers, 128 batch size, and 150 learning rate drop using LSTM. Sunspot number prediction using LSTM algorithm was very good because it obtained MAPE value less than 10% but GRU is better than LSTM with difference MAPE value 2.7847%.

Keywords— *prediction, sunspot numbers, time series, GRU, LSTM.*

Received January 28th,2021; Revised April 27th, 2021; Accepted April 29th, 2021

1. INTRODUCTION

The sun is the center of the solar system which controls the solar system environment. The sun has several main activities, for example sunspot, solar flare, and Corona Mass Ejection (CME). Sunspot is very important to be researched because the bigger sunspot number, the higher level of solar activity and the smaller sunspot number, the lower level of solar activity. The impact of sunspot is not only on the space, but also on the climate and the weather on earth [1]. The phenomena of the sunspot impact can be minimized by early information obtained from the prediction results, so that if there is a significant increase in sunspot numbers, it can inform other physical consequences that may be caused. Based on the background of the problem, this research will discuss sunspot number prediction.

A previous research related to sunspot number prediction was a research that predicted sunspot number using Fuzzy Time series Markov Chain Model. The research resulted in MAPE of 9.5% [2]. Furthermore, another research used Support Vector Regression (SVR) algorithm obtained 35.32 MSE, 5.94 RMSE, and 0.12 MAAPE so it can be said that the prediction results were quite accurate [3]. Then, a research that predict sunspot number using statistical method Autoregessive Integrated Moving Average (ARIMA) obtained 96.5% correlation confection between the proposed result and ARIMA result [4].

In other problems, there are several previous researches which used GRU algorithm for predictions, including a research that predicted the number of train passengers using the GRU method. This research conducted experiments up to 15000 iterations, the smallest number of MSE was obtained at 14000 iterations with a combination of parameters of 0.01 learning rate, 100 batch size, 512 hidden layer, and 30 windows size, and resulted in the MAPE value of 4.84% [5]. Furthermore, a research that predicted cargo demand used the GRU method. The best parameters obtained were 10^{-2} learning rate, 32 hidden layer, 16 batch size, 100 epoch. The ratio of data splitting was 70% for training, 10% for validation, and 20% for testing. The RMSE result was 247.395 [6]. Then, a research compared the performance between LSTM, GRU, and ARIMA methods in predicting traffic flows. The average of MAE with GRU was reduced at about 10% than the ARIMA method and 5% than LSTM method [7].

We know that GRU is good for prediction from several previous researches which used GRU algorithm, but the architecture can be improved by finding optimal parameters and data division, and using the best optimization [8]. One of optimization algorithms is Adaptive Moment Estimation (ADAM) [9]. ADAM was proven can improve the performance of deep neural network [10]. So, in this research will use ADAM optimization to improve the performance of GRU.

Based on several phenomena due to sunspot impacts and the previous researches which prove that GRU can predict well with a good level of accuracy. This research will compare the performance of GRU and LSTM algorithm to predict sunspot numbers using ADAM optimization to improve the model. The data division, hidden layer, batch size, and learning rate drop parameters used based on trial, so that we can know the best parameter for the prediction model. It is expected that the GRU and LSTM algorithm can be implemented for predicting sunspot numbers so that it can help minimize the phenomena due to sunspot impacts and knows the best method for predicting sunspot number.

2. METHODS

2.1 Data Collection

The data used in this research was the data of monthly sunspot numbers. The data was obtained from SILSO (Sunspot Index and Long-Term Solar Observation) website with .csv file format. There were 3240 data, from January 1750 to December 2019. The data sample of sunspot numbers is shown in Table 1.

Table 1 The Data Sample of Sunspot Numbers

Year	Month	Sunspot Numbers
1750	1	148.4
1750	2	150.3
1750	3	153.9
:	:	:
2019	12	1.8

2.2 Sunspot

Sunspot is a dark area on the photosphere layer [10]. Sunspot's color is dark because the temperature of the sunspot ranges from 4000º K to 4500º K while the sun's temperature is 6000º K [3][11]. The sunspot number determines the level of solar activity [11], the greater the sunspot number, the higher solar activity and the smaller the sunspot number, the lower the solar activity. The number of sunspots has increased and decreased in approximately 11 years, known as the solar activity cycle [12]. Sunspot can be counted using formula in equation (1).

$$R = k(10g + n) \qquad (1)$$

Where R is sunspot number, k is correction factor which value is 0.65, g is the number which identifies observed sunspot group, and n is the number of spots.

2.3 Technical Research

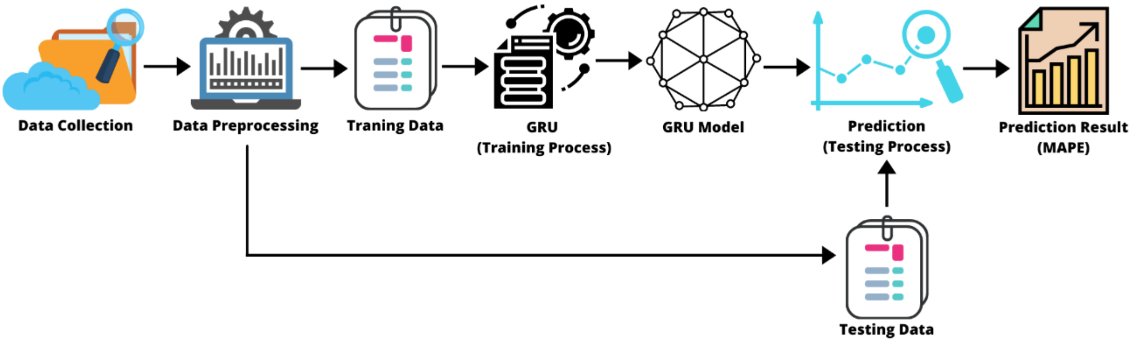


Figure 1 The Technical Research of Sunspot Numbers Prediction Using GRU Algorithm

The first step of predicting sunspot numbers is collecting sunspot number data, next the data is normalized using equation (2). After the data is normalized, the data is divided into the training and testing process. Then the data structure is formed into time series data. After that the parameters used in the training process are initialized. In this training process, ADAM optimization is used to obtain the best GRU model. The GRU model generated from the training process will be used to predict sunspot numbers in the testing process. Then the data is denormalized using equation (12). The last step is calculating the error value to measure the prediction accuracy using equation (13). The steps of predicting sunspot numbers can be seen in Figure 1.

2.3.1 Data Normalization

There are several data normalization method, one of them is min-max normalization. Min-max normalization is a method that uses linear transformations on the actual data to produce a balanced comparison of values between data before and after normalization [12]. The purpose of data normalization is to reduce the far data range because the data range affects the prediction results [13]. The data normalization formula can be seen in equation (2).

$$x = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{2}$$

Where x is the normalized data, x is the actual data, $x_{\min}$ is the minimum value of the actual data, and $x_{\max}$ is the maximum value of the actual data.

2.3.2 Gated Recurrent Unit (GRU)

GRU was first introduced by Kyunghyun Cho et al in 2014 [14]. GRU is an algorithm developed from Recurrent Neural Network (RNN) method which is similar to Long Short-Term Memory (LSTM) [15][16]. GRU has more simple architecture than LSTM [17]. The basic architecture of the RNN generates Vanishing and Exploding Gradient Descent problem [18]. This problem occurs because of continuous multiplication at Backpropagation Through Time (BPTT) result. GRU uses gates to solve this problem [19]. GRU has two gates, namely update gate and reset gate [20] which can be seen in Figure 2.

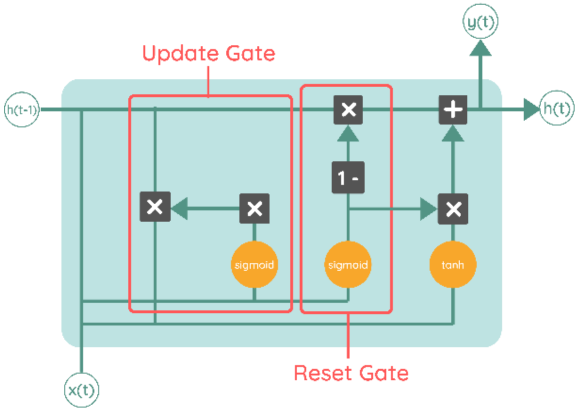


Figure 2 GRU Architecture

The first step of building a GRU model is calculating the update gate (z_t) using the formula in equation (3) which is used to determine how much previous information should be retained [21].

$$z_t = \sigma\left(w^{(z)}x_t + u^{(z)}h_{t-1} + b\right) \tag{3}$$

Where w and u are the weight, x_t is the input, h_{t-1} is the hidden state, and b is the bias.

The next step is calculating reset gate (r_t) using formula in equation (4) which is used to determine how much previous information should be removed and how to combine the new input and the previous information.

$$r_t = \sigma\left(w^{(r)}x_t + u^{(r)}h_{t-1} + b\right) \tag{4}$$

Then calculating hidden state candidate (h'_t) which will use reset gate to save relevant information from the past. Hidden state candidate can be seen in equation (5).

$$h'_t = \tanh\left(wx_t + r_t \,\square\, uh_{t-1}\right) \tag{5}$$

Where □ is the hadamard product.

The last step is calculating hidden state (h_t) using formula in equation (6). The hidden state is also output (y_t).

$$h_t = z_t \square\ h_{t-1}\left(1-z_t\right)\square\ h'_t \tag{6}$$

GRU has several parameters which can affect the prediction result including hidden layer, batch size, dan learning rate drop. Hidden layer is the number of calculations in the training process. The batch size is how often the weights will be updated. Learning rate drop is the number of iterations in determining the learning rate [22].

2.3.3 ADAM

Adam or Adaptive Moment Estimation was introduced by Diendik Kingma and Jimny Lei Ba [23]. ADAM is an optimization algorithm for gradient optimization on neural networks based on training data [5]. ADAM is a combination of the advantages of two popular methods, such as AdaGrad and RMSProp [24]. Some of the advantages of this algorithm are: it is easy and efficient, it does not require large memory, and it is suitable for problems that have a lot of data and parameters [25]. The estimation of the first moment and the second moment can be calculated using equation (7) and equation (8).

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t \tag{7}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2 \tag{8}$$

Where m_t is the estimation of the first moment and v_t is the estimation of the second moment. Both of them are initialized as vectors 0. It can affect the values are biased toward zero when the decay values are small [26]. That problem can be solved by calculating the bias correction using equation (9) and equation (10).

$$m_t = \frac{m_t}{1-\beta_2^t} \tag{9}$$

$$\hat{v}_t = \frac{v_v}{1-\beta_2^t} \tag{10}$$

Where the value of β_1 is 0.9, β_2 is 0.999

After the bias are corrected, then repairing the weight using equation (11).

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \in}} m_t \tag{11}$$

Where $\in$ is epsilon which value is 10^{-8}.

2.3.4 Data Denormalization

After obtaining the predicted value from the testing process, then proceed with data denormalization. Data denormalization is to return the data to its original range before being normalized [27]. Denormalization formula is shown in equation (12).

$$x_i = x(x_{\max} - x_{\min}) + x_{\min} \tag{12}$$

Where x_i is the denormalized data, x is the normalized data, $x_{\max}$ is the maximum value of the actual data, and $x_{\min}$ is the minimum value of the actual data.

2.3.5 Performance Measurement

Mean Absolute Percentage Error (MAPE) is a calculation used to measure the accuracy of a prediction system [28]. MAPE formula can be seen in equation (13).

$$MAPE = \left(\frac{100\%}{n}\right)\sum_{i=1}^{n}\left|\frac{y_i - y_i}{y_i}\right| \qquad (13)$$

Where y_i is the actual data, y_i is the predicted value, while n is the amount of the data. The smaller MAPE value, the better accuracy prediction [29]. The criteria of MAPE are shown in Table 2.

Table 2 The Criteria of MAPE [30]

MAPE Value	Criteria
MAPE < 10%	High accuracy prediction
10% ≤ MAPE ≤ 20 %	Good prediction
20% ≤ MAPE ≤ 50%	Reasonable prediction
50% ≥ MAPE	Inaccurate prediction

3. RESULTS AND DISCUSSION

In this research, the number of data used was 3240 data consisting of 70% of the data obtained from January 1750 to July 1938 used for the training process totaling 2263 months and 30% of the data obtained from August 1938 to December 2019 used for the testing process totaling 970 months. The data division used is based on trials which can be seen in Table 5. The training and testing data sample is shown in Table 3.

Table 3 The Normalized Data Sample for Training and Testing Process

Training			Testing		
Year	Month	Sunspot numbers	Year	Month	Sunspot numbers
1750	1	0.521	1938	8	0.622
1750	2	0.527	1938	9	0.606
1750	3	0.540	1938	10	0.603
:	:	:	:	:	:
1938	7	0.637	2019	12	0.006

Table 4 The Normalized Data Sample of Time Series

x_{t-6}	x_{t-5}	x_{t-4}	x_{t-3}	x_{t-2}	x_{t-1}	x_t	x_{t+1}
0.521	0.527	0.540	0.541	0.516	0.490	0.487	0.478
0.527	0.540	0.541	0.516	0.490	0.487	0.478	0.460
0.540	0.541	0.516	0.490	0.487	0.478	0.460	0.441
:	:	:	:	:	:	:	:
0.014	0.013	0.012	0.012	0.011	0.009	0.007	0.006

Based on Table 4, this research used seven input variables and an output variable. It was aimed to obtain the prediction results for the next one month, and it would take seven months earlier. The time series data sample is shown in Table 4. The parameters used were based on the results of the trials that resulted in the smallest MAPE value in order to obtain the optimal model [31]. Table 5 shows a comparison of MAPE values based on several parameters, those are hidden layer, batch size, and learning rate drop and data division of 70:30 and 80:20 using GRU method.

Based on Table 5, it can be seen that the more hidden layer, the lower average MAPE value. The average MAPE value of 70:30 data division was lower than that of 80:20 data division. The highest average MAPE value was obtained at 80:20 data division and 50 hidden layers, while the lowest average MAPE value was obtained at 70:30 data division and 150 hidden layers. The highest MAPE value was 21.3408% with 50 hidden layers, 256 batch size, and 50 learning rate drop while the lowest MAPE value was 7.1705% with 150 hidden layers, 32 batch size, 100 learning rate drop. Based on Table 1, it can be concluded that the GRU algorithm was suitable for long-term prediction of sunspot numbers because the MAPE value was less than 10%.

Table 5 MAPE Value Based on Different Parameters and Data Division of GRU

Parameters			Data divison			
			70:30		80:20	
Hidden Layers	Batch size	Learning rate drop	MAPE (%)	Average	MAPE (%)	Average
50	32	50	12.816	11.593	16.866	14.116
		100	13.470		18.590	
		150	10.371		9.260	
	64	50	18.484		11.143	
		100	8.626		11.895	
		150	7.584		9.936	
	128	50	11.184		18.174	
		100	12.965		13.105	
		150	7.800		13.921	
	256	50	15.075		21.341	
		100	10.922		16.384	
		150	9.816		8.778	
100	32	50	10.708	10.974	15.110	11.793
		100	9.301		10.879	
		150	7.614		9.541	
	64	50	15.389		13.476	
		100	9.185		9.791	
		150	8.917		8.375	
	128	50	14.155		15.629	
		100	11.979		13.265	
		150	8.370		9.989	
	256	50	18.294		17.215	
		100	9.495		9.152	
		150	8.274		9.096	
150	32	50	13.644	10.606	16.179	11.126
		100	7.171		11.354	
		150	8.251		8.036	
	64	50	15.339		13.991	
		100	8.912		9.981	
		150	8.346		8.566	
	128	50	15.306		13.928	
		100	10.247		9.320	
		150	7.849		8.777	
	256	50	13.658		15.408	
		100	9.313		9.678	
		150	9.233		8.289	

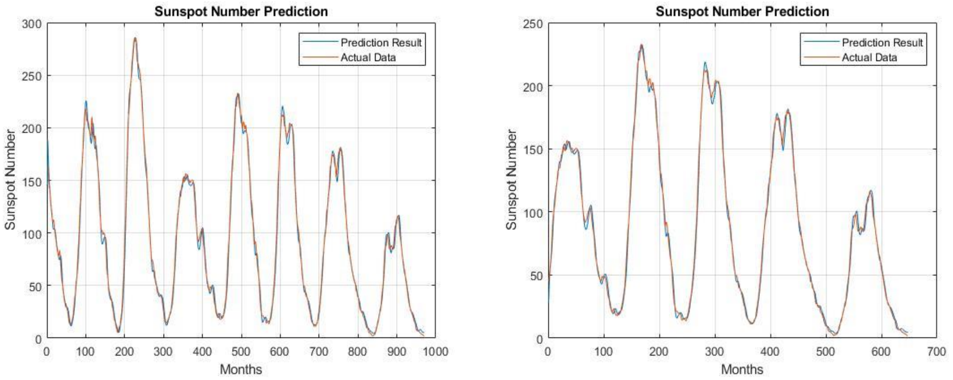
Table 6 MAPE Value Based on Different Parameters and Data Division of LSTM

Parameter			Data divison			
			70:30		80:20	
Hidden Layers	Batch size	Learning rate drop	MAPE (%)	Average	MAPE (%)	Average
50	32	50	19.3088	15.32576	21.4041	17.95173
		100	20.2017		16.3147	
		150	13.0826		17.06	
	64	50	18.5321		24.8176	
		100	15.5388		14.8375	
		150	12.2765		15.0114	
	128	50	17.6357		22.8205	
		100	12.9658		17.4439	
		150	12.3641		17.2459	
	256	50	17.484		20.5128	
		100	12.6985		13.9126	
		150	11.8205		14.0397	
100	32	50	15.7818	13.20449	19.6113	15.84138
		100	12.7523		13.3516	
		150	11.3958		12.2994	
	64	50	15.3529		20.4243	
		100	11.7269		16.8872	
		150	11.3604		13.3657	
	128	50	17.6675		18.906	
		100	11.7738		15.0475	
		150	10.1493		10.7346	
	256	50	17.1457		22.4632	
		100	11.9452		14.2591	
		150	11.4023		12.7466	
150	32	50	14.8143	12.46136	18.6868	14.27757
		100	11.2499		13.5321	
		150	10.3571		12.4605	
	64	50	14.8364		17.9773	
		100	11.4634		13.1043	
		150	11.8909		11.5085	
	128	50	16.5047		17.8684	
		100	11.5239		13.0508	
		150	9.9557		11.4351	
	256	50	15.9415		16.6958	
		100	10.9616		12.9823	
		150	10.0369		12.0289	

Table 6 shows a comparison of MAPE values based on several parameters, those are hidden layer, batch size, and learning rate drop and data division of 70:30 and 80:20 using LSTM method. It can be seen that LSTM is good for predicting sunspot number but GRU is better than LSTM. The average of MAPE values obtained by LSTM are bigger than GRU. The smallest MAPE value is 9.9557% with 150 hidden layers, 128 batch size, learning rate drop 150, and data division 70:30. Same as Table 5, the more hidden layer, the lower average MAPE value.

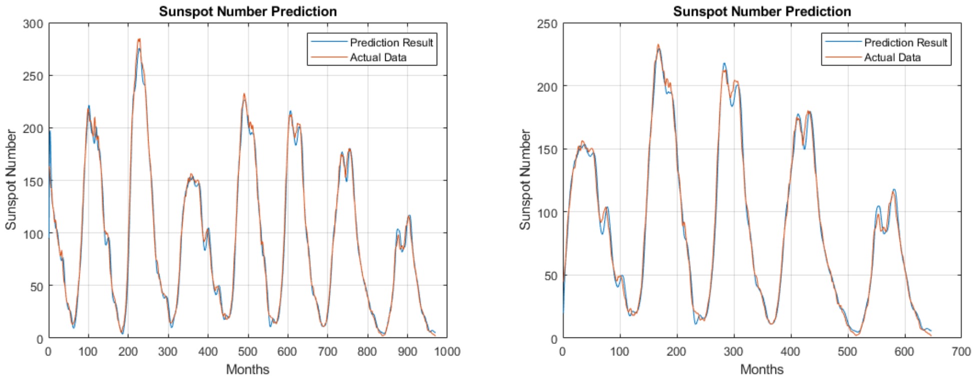
Visualization of actual data and predicted result can be seen in Figure 3 and Figure 4. Figure 3(a) and Figure 4(a) shows a graph of the predicted results and actual data from March

1939 to December 2019, while Figure 3(b) and Figure 4(b) shows a graph of the predicted results and actual data from May 1991 to December 2019. It can be seen from Figure 3 that the results of the prediction and the actual data are pretty similar than Figure 3(b), Figure 4(a), and Figure 4(b). The sunspot number in Figure 3(a) and Figure 4(a) axceeds 250. The increase of sunspot number can cause flare or explotion of CME [32].



(a) (b)

Figure 3 The best MAPE value of GRU (a) Data division 70:30 (b) Data division 80:20



(a) (b)

Figure 4 The best MAPE value of LSTM (a) Data division 70:30 (b) Data division 80:20

4. CONCLUSIONS

Based on the results of the research on the prediction of sunspot numbers using GRU and LSTM algorithm, the best MAPE value obtained was 7.171% with 70:30 data division, 150 hidden layers, 32 batch size, and 100 learning rate drop using GRU and 9.9557% with data division 70:30, 150 hidden layers, 128 batch size, and learning rate drop 150 using LSTM. Therefore, it can be said that sunspot number prediction using LSTM algorithm was very good because it obtained MAPE value less than 10% but GRU is better than LSTM with difference MAPE value 2.7847%.

5. FUTURE WORKS

This research did not not pay attention to the outliers in the time series data. These outliers can be detected by various methods [33]. One of the examples is a research which detected outliers quickly using Local Correlation Integral (LOCI) [34]. It is expected that further research will use variations of the GRU method, such as BiGRU which results in MAPE values smaller than GRU [35] or can use other deep learning methods and pay attention to the presence of outliers in time series data to improve performance in predicting sunspot numbers.

ACKNOWLEDGEMENTS

Thanks to department of Mathematics UIN Sunan Ampel Surabaya and Balai Pengamatan Antariksa dan Atmosfer Pasuruan for the support and attention for this research.

REFERENCES

[1] S. Chattopadhyay, D. Jhajharia, and G. Chattopadhyay, “Trend Estimation and Univariate Forecast of The Sunspot Numbers: Development and Comparison of ARMA, ARIMA and Autoregressive Neural Network Models,” *Comptes Rendus - Geosci.*, vol. 343, no. 7, pp. 433–442, 2011, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1631071311001520.

[2] D. C. R. Novitasari, N. Ardhiyah, and N. Widodo, “Flare Identification by Forecasting Sunspot Numbers Using Fuzzy Time Series Markov Chain Model,” *Proc. - 2019 Int. Semin. Intell. Technol. Its Appl. ISITIA 2019*, pp. 387–392, 2019, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8937242.

[3] S. Suwanto, “Prediksi Bilangan Sunspot menggunakan Support Vector Regression (SVR),” 2019, [Online]. Available: http://digilib.uinsby.ac.id/id/eprint/38114.

[4] H. I. Abdel-Rahman and B. A. Marzouk, “Statistical Method to Predict The Sunspots Number,” *NRIAG J. Astron. Geophys.*, vol. 7, no. 2, pp. 175–179, 2018, [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090997718300658.

[5] R. P. Wardana, “Penerapan Model Gated Recurrent Unit untuk Peramalan Jumlah Penumpang Kereta Api di PT. KAI (PERSERO),” pp. 45–45, 2020, [Online]. Available: http://repository.uinjkt.ac.id/dspace/handle/123456789/51047.

[6] R. A. Saputra, “Prediksi Permintaan Kargo pada Cargo Service Center Tangerang City Menggunakan Metode Gated Recurrent Unit,” 2020. http://eprints.umm.ac.id/id/eprint/63970.

[7] L. L. Fu Rui, Zhang Zuo, “Using LSTM and GRU Neural Network Methods for Traffic Flow Prediction,” *31st Youth Academic Annual Conference of Chinese Association of Automation*, 2016. https://ieeexplore.ieee.org/abstract/document/7804912.

[8] A. S. Prabowo, A. Sihabuddin, and A. SN, “Adaptive Moment Estimation On Deep Belief Network For Rupiah Currency Forecasting,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 1, p. 31, 2019, [Online]. Available: https://jurnal.ugm.ac.id/ijccs/article/view/39071.

[9] T. Dozat, “Incorporating Nesterov Momentum into Adam,” *ICLR Work.*, no. 1, pp. 2013–2016, 2016, [Online]. Available: https://openreview.net/forum?id=OM0jvwB8jIp57ZJjtNEZ.

[10] I. K. M. Jais, A. R. Ismail, and S. Q. Nisa, “Adam Optimization Algorithm for Wide and Deep Neural Network,” *Knowl. Eng. Data Sci*, vol. 2, no. 1, pp. 41–46, 2019, [Online]. Available: https://core.ac.uk/download/pdf/287322851.pdf.

[11] N. Mohamad Ansor, Z. S. Hamidi, and N. N. M. Shariff, “The Impact on Climate Change Due to the Effect of Global Electromagnetic Waves of Solar Flare and Coronal Mass Ejections (CMEs) Phenomena,” *J. Phys. Conf. Ser.*, vol. 1298, no. 1, 2019, [Online]. Available: https://iopscience.iop.org/article/10.1088/1742-6596/1298/1/012019/meta.

[12] D. A. Nasution, H. H. Khotimah, and N. Chamidah, “Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN,” *Comput. Eng. Sci. Syst. J.*, vol. 4, no. 1, p. 78, 2019, [Online]. Available:

https://jurnal.unimed.ac.id/2012/index.php/cess/article/view/11458.

[13] J. Sola and J. Sevilla, "Importance of Input Data Normalization for the Application of Neural Networks to Complex Industrial Problems," *IEEE Trans. Nucl. Sci.*, vol. 44, no. 3 PART 3, pp. 1464–1468, 1997, doi: https://ieeexplore.ieee.org/abstract/document/589532.

[14] K. Cho *et al.*, "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," *EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 1724–1734, 2014, [Online]. Available: https://arxiv.org/abs/1406.1078.

[15] B. Athiwaratkun and J. W. Stokes, "Malware Classification with LSTM and GRU Language Models and A Character-Level CNN," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2482–2486, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7952603.

[16] K. A. Althelaya, E. S. M. El-Alfy, and S. Mohammed, "Stock Market Forecast Using Multivariate Analysis with Bidirectional and Stacked (LSTM, GRU)," *21st Saudi Comput. Soc. Natl. Comput. Conf. NCC 2018*, pp. 1–7, 2018, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8593076.

[17] J. Tzeng, Y. R. Lai, M. L. Lin, Y. H. Lin, and Y. C. Shih, "Improve the LSTM and GRU model for small training data by wavelet transformation," *Proc. Int. Jt. Conf. Neural Networks*, pp. 2–7, 2020, doi: 10.1109/IJCNN48605.2020.9206840.

[18] S. Kumar, L. Hussain, S. Banarjee, and M. Reza, "Energy Load Forecasting using Deep Learning Approach-LSTM and GRU in Spark Cluster," in *Proceedings of 5th International Conference on Emerging Applications of Information Technology, EAIT 2018*, 2018, pp. 1–4, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8470406.

[19] B. Yue, J. Fu, and J. Liang, "Residual Recurrent Neural Networks for Learning Sequential Representations," *Inf.*, vol. 9, no. 3, 2018, [Online]. Available: https://www.mdpi.com/2078-2489/9/3/56.

[20] Q. Tao, F. Liu, Y. Li, and D. Sidorov, "Air Pollution Forecasting Using A Deep Learning Model Based on 1D Convnets and Bidirectional GRU," *IEEE Access*, vol. 7, pp. 76690–76698, 2019, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8732985.

[21] Y. Gao and D. Glowacka, "Deep Gate Recurrent Neural Network," in *Asian conference on machine learning*, 2016, pp. 350–365, [Online]. Available: http://proceedings.mlr.press/v63/gao30.html.

[22] D. Z. Haq *et al.*, "Long Short-Term Memory Algorithm for Rainfall Prediction Based on El-Nino and IOD Data," *ScienceDirect*, vol. 00, no. 2019, 2020.

[23] D. P. Kingma and J. L. Ba, "Adam: A method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15, [Online]. Available: https://arxiv.org/abs/1412.6980.

[24] Z. Chang, Y. Zhang, and W. Chen, "Electricity Price Prediction Based on Hybrid Model of ADAM Optimized LSTM Neural Network and Wavelet Transform," *Energy*, vol. 187, p. 115804, 2019, [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0360544219314768.

[25] Z. Chang, Y. Zhang, and W. Chen, "Effective Adam-Optimized LSTM Neural Network for Electricity Price Forecasting," *Proc. IEEE Int. Conf. Softw. Eng. Serv. Sci. ICSESS*, vol. 2018-Novem, no. Figure 1, pp. 245–248, 2019, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8663710.

[26] S. Ruder, "An Overview of Gradient Descent Optimization Algorithms," pp. 1–14, 2016, [Online]. Available: http://arxiv.org/abs/1609.04747.

[27] E. Ogasawara, L. C. Martinez, D. De Oliveira, G. Zimbrão, G. L. Pappa, and M. Mattoso, "Adaptive Normalization: A Novel Data Normalization Approach for Non-Stationary Time Series," in *Proceedings of the International Joint Conference on Neural*

Networks, 2010, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5596746.

[28] J. Tayman and D. A. Swanson, “On The Validity of MAPE as A Measure of Population Forecast Accuracy,” *Popul. Res. Policy Rev.*, vol. 18, no. 4, pp. 299–322, 1999, [Online]. Available: https://link.springer.com/article/10.1023/A:1006166418051.

[29] J. W. Koo, S. W. Wong, G. Selvachandran, H. V. Long, and L. H. Son, “Prediction of Air Pollution Index in Kuala Lumpur using fuzzy time series and statistical models,” *Air Qual. Atmos. Heal.*, vol. 13, no. 1, pp. 77–88, 2020, doi: 10.1007/s11869-019-00772-y.

[30] S. S. C. Ramasamy P and A. K. Yadav, “Wind Speed Prediction in The Mountainous Region of India Using An Artificial Neural Network Model,” *Renewable Energy*, 2015. https://www.sciencedirect.com/science/article/abs/pii/S0960148115001342.

[31] S. Bouktif, A. Fiaz, A. Ouni, and M. A. Serhani, “Optimal deep learning LSTM model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches,” *Energies*, 2018. https://www.mdpi.com/1996-1073/11/7/1636.

[32] S. Suwanto and D. C. R. Novitasari, “Forecasting Solar Activities based on Sunspot Number Using Support Vector Regression (SVR),” *JPSE (Journal Phys. Sci. Eng.*, vol. 5, no. 1, pp. 6–10, 2020, doi: 10.17977/um024v5i12020p006.

[33] S. Basu and M. Meckesheimer, “Automatic Outlier Detection for Time Series: An Application to Sensor Data,” *Knowl. Inf. Syst.*, vol. 11, no. 2, pp. 137–154, 2007, [Online]. Available: https://link.springer.com/article/10.1007/s10115-006-0026-6.

[34] S. Papadimitriou, H. Kitagawa, P. B. Gibbons, and C. Faloutsos, “LOCI: Fast Outlier Detection Using the Local Correlation Integral,” *Proc. 19th Int. Conf. data Eng. (Cat. No. 03CH37405)*, pp. 315–326, 2003, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/1260802.

[35] D. She and M. Jia, “A BiGRU Method for Remaining Useful Life Prediction of Machinery,” *Meas. J. Int. Meas. Confed.*, vol. 167, no. June 2020, p. 108277, 2021, [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0263224120308162.