

Extreme Learning Machine and Particle Swarm Optimization for Inflation Forecasting

Adyan Nur Alfiyatin¹, Agung Mustika Rizki², Wayan Firdaus Mahmudy³, Candra Fajri Ananda⁴

Faculty of Computer Science, Brawijaya University, Malang, Indonesia^{1,2,3}
Faculty of Economics and Business, Brawijaya University, Malang, Indonesia⁴

Abstract—Inflation is one indicator to measure the development of a nation. If inflation is not controlled, it will have a lot of negative impacts on people in a country. There are many ways to control inflation, one of them is forecasting. Forecasting is an activity to find out future events based on past data. There are various kinds of artificial intelligence methods for forecasting, one of which is the extreme learning machine (ELM). ELM has weaknesses in determining initial weights using trial and error methods. So, the authors propose an optimization method to overcome the problem of determining initial weights. Based on the testing carried out the proposed method gets an error value of 0.020202758 with computation time of 5 seconds.

Keywords—Extreme learning machine; particle swarm optimization; inflation; prediction

I. INTRODUCTION

The economy of a country is affected by macro variables; they are Gross Domestic Brute (GDP), unemployment, and inflation. From those three components, inflation holds the most important rule [1]. Inflation is an incidence of rising prices of goods and services in general for a long time and continuously influences each other [2]. The economic stability of a country is seen based on the country's inflation rate. In Indonesia, the stability of economics is controlled by the central bank namely Bank Indonesia through monetary policy. Monetary policy is a policy illustration that serves to overcome economic problems with the aim of maintaining the stability of currency values. In this case, the stability of the prices of goods and services reflected in inflation [3]. To achieve low and stable inflation, Bank Indonesia set a framework called the Inflation Target Framework (ITF).

ITF framework has determinations of inflation targets for the next few years. An influential process for determining the inflation target is by using inflation rate forecasting. With the inflation rate forecasting, it can reduce the inflation rate up to 4 – 5 % [3]. The method used by Bank Indonesia for current inflation forecasting is by forward looking. It is better to overcome the inflation shocks that occurred during the set time. Getting information on domestic conditions about the changes occurring in the economic field will serve as one of the important information on policy formulation. In order to obtain accurate information and good influence on the policy, the set required a forecasting. Forecasting will yield accurate information if the variables used are significant and the data used is credible. A forecasting used past data, with the aim to know the pattern of future events based on past data patterns [4].

Some methods used for forecasting are machine learning [5], neural network, regression [6], fuzzy time series [7], and extreme learning machine (ELM) [8]. Based on those methods, ELM has the advantage of speeds in the learning process, the generalization is very good without overtraining, and it can determine the best results according to the input weights used. Therefore, many researchers combine ELM with optimization methods in order to get the best input weight that will be used in the hidden layer, such as hybrid ELM and PSO, ELM and GA, ELM and Regression. The purpose of the optimization method is to define the best weight of input neurons to be implemented in the ELM. So, this research will combine particle swarm optimization and ELM for inflation rate forecasting in Indonesia. The focus is to know the hybrid performance method between particle swarm optimization and ELM.

II. RELATED WORK

Recently, many researchers on forecasting used machine learning method [5] aimed for determining the performance of stock-making forecasting. Methods used are decision tree, neural network, multi-layer perceptron (MLP), support vector machine (SVM) and hybrid methods. Some of the methods tested in the study have their respective advantages in the tree decision method which has simple techniques and capabilities that can be relied upon in predicting values, both by using large and small amounts of data, Neural Networks support is able to accept or model the input relations / non-linear complex output. SVM is a learning strategy that was started and designed to solve problems, but also to solve non-linear regression problems as well. After reviewing some of the advantages and disadvantages of the method, this study proves that with a combination of the methods, the performance is better than the performance of a single method.

The next study was conducted by Semaan [6], to forecast exchange rates with statistical methods (regression) and neural networks (ANN). The purpose of this study is to know the performance of both methods and the value of the accuracy produced to find out the relationship between independent and dependent variables that affect the exchange rate. The results show that the ANN method can increase the value of accuracy for exchange rate forecasting compared with the regression method. The advantage of the ANN method is if there is a complex pattern of data with a large number, the data pattern can be extracted by ANN even though the exact mathematical equation between the relationship between the dependent variable and the independent variable is unknown. Excellence

as well as weakness of ANN can be overcome by the addition of regression methods due to the ability of regression methods that can tell the mathematical equations between the relationship of dependent and independent variables accordingly.

Other research conducted by Huarng and Yu [9] is about the combination of fuzzy time series and backpropagation for stock forecasting. This study aims to compare the performance between conventional fuzzy time series with a combination of fuzzy time series and backpropagation. Fuzzy time series is used for forecasting unidentified patterns while backpropagation is suitable for solving problems which patterns are identified. So that if these two methods are combined, the results will be better than using single method only. And it proved from this study that the combined method can work better for forecasting on known and unknown patterns, therefore this method outperforms the basic model of conventional fuzzy time series.

Research on inflation rate forecasting is also carried out by Sari [10] [11] uses time series data on consumer price index, money supply, BI rate, exchange rate and inflation rate with the backpropagation method shows that the involvement of external factors for forecasting the inflation rate is very influential on the results of the accuracy. Further research is carried out by Anggodo [12] using additional data with the same parameters gets better results because it uses a combined method between neural network and optimization.

Based on some previous researches, it shows that artificial neural network method gets higher accuracy value in comparison with statistical method, and if artificial neural network is combined with other method, then the method performance will be improved. In this research, the proposed method will be hybrid of artificial neural network, namely Extreme Learning Machine (ELM). The optimization using particle swarm optimization to define the forecasting performance is calculated using root mean square error (RMSE). This research highlights the addition of parameters are credit value and asset value to forecast the inflation rate and it is expected that combined neural network and optimization methods will get the best results compared to previous studies [10]–[12].

III. DATA SET

In this study, data were obtained from Bank Indonesia [13] and Badan Pusat Statistics (BPS). Records of data used were from January 2005 to December 2017. The parameters are using historical data with time series analysis (b-1, b-2, b-3). b-1 represents the previous month's parameter, b-2 represents the previous 2 months and b-3 represents the previous three months. This study also used several external factors that influence the inflation rate such as CPI, BI rate, money supply, exchange rate, credit value and asset value. Parameters are used as input variable for inflation rate forecasting while the output variables are the inflation forecasting rate in Indonesia.

IV. PARTICLE SWARM OPTIMIZATION (PSO)

PSO is used to optimize input weights in ELM so that the generation of weight in ELM is not randomly made. The use of Hybrid PSO-Elm aims to obtain optimal weight within minimal time. Particle Swarm Optimization proposed by Kennedy and Eberhart (1995) adopted the behaviour of a flock of birds. The solution to the PSO is called "particle". The PSO stage starts from the position initialization randomly, updates the speed, updates the Pbest value and the Gbest value, and fitness calculation. Pbest (Personal Best) is the optimal value determined by the particle itself while Gbest (Global Best) is the best value of a group from PBest [14]. For more details, it can be seen in Fig. 1 (pseudo code) below:

```
Begin
t=0
initialization position particle (xti,j), velocity (vti,j), Pbestti,j=xti,j,
calculate fitness of particles, Gbestti,j
do
    t = t+1
    update velocity vi,j (t)
    update position xi,j (t)
    calculate fitness of each particles
    update Pbesti,j (t) and Gbesti,j (t)
while (not a stop condition)
end
```

Fig. 1. Pseudo Code of Particle Swarm Optimization.

The equation used to update velocity (1) and update position (2):

$$v_{i,j}^{t+1} = w \cdot v_{i,j}^t + c_1 r_1 (Pbest_{i,j}^t - x_{i,j}^t) + c_2 r_2 (Gbest_{g,j}^t - x_{i,j}^t) \quad (1)$$

$$x_{i,j}^{t+1} = x_{i,j}^t + v_{i,j}^{t+1} \quad (2)$$

V. EXTREME LEARNING MACHINE (ELM)

This method was introduced by Huang, et al. [9]. ELM is feed forward neural network concept that Single Hidden Layer Feedforward Neural Networks (SLFNs) because ELM has only one hidden layer on its network architecture. ELM is designed to overcome the weakness of previous neural networks in the speed learning process. With the selection of parameters such as input, weight and hidden bias randomly so the performance from learning speed at the ELM is faster than another neural network method, ELM gets good generalization performance without overtraining problems. Step by step in the ELM process, there are:

A. Data Normalization

Data normalization was a process of changing the form of data into a more specific value in the limit value 0-1. The aim was to adjust the input data to the output data. Equation (3) showed the normalization of function.

$$v' = \frac{v}{\max v} \quad (3)$$

v[^] = New data

v = Actual data

max v = the maximum value of the actual data of each parameter.

B. Training Process

This process aims to conduct training using train data. With this process, an optimal weight value will be obtained. Steps in the training process are as follows [15]:

- Determine matrix W_{mn} randomly as weight input within range [0,1], in the form of array sizes m (number of hidden neurons) \times n (number of input neurons). Then it makes another random value for the bias matrix b within range [0,1] in size $1 \times$ (number of hidden neurons).
- Calculate the value of the hidden layer matrix output with equation (4). Calculation of b ($\text{ones}(i_{\text{train}},1)$) multiplies the matrix let as much as the amount of training data.

$$H = \frac{1}{1 + \exp(-(x_{\text{train}} w^T + b(\text{ones}(i_{\text{train}},1))))} \quad (4)$$

where:

- H = the hidden layer output matrix
- x_{train} = the input matrix on normalized training data
- w^T = the matrix transpose of weights
- i_{train} = the number of training data
- b = the bias matrix

- Calculate β as the output weights by using the equation (5) that H^+ or Moore-Penrose Pseudo Invers matrix can be calculated by equation (6).

$$\beta = H^+ t \quad (5)$$

$$H^+ = (H^T H)^{-1} H^T \quad (6)$$

where:

- β = the output weight matrix
- H^+ = the Moore-Penrose Pseudo Invers matrix from matrix H
- t = the target matrix
- H = the hidden layer output matrix

- Calculate the output by using the equation (7).

$$Y = H \beta \quad (7)$$

where:

- Y = predicted result
- H = hidden layer output matrix
- β = output weight matrix

C. Testing Process

After the training process, the testing process was done by using the test data. It aims to test the results of training, so it can know the accuracy of the system. Steps are shown as follows [15]:

- Determine W_{mn} , b and β value from training process.
- Calculate the value of the output matrix on the hidden layer using equation (8). The calculation of b ($\text{ones}(i_{\text{test}},1)$) is to multiply the bias matrix by the number of test data.

$$H = \frac{1}{1 + \exp(-(x_{\text{test}} w^T + b(\text{ones}(i_{\text{test}},1))))} \quad (8)$$

where:

- H = hidden layer output matrix
- x_{test} = input matrix on test data that had been normalized
- w^T = matrix transpose of weights
- i_{test} = number of test data
- b = bias matrix

- Calculate the output value by using equation (7).
- Denormalization of predicted results using equation (10).
- Calculate the evaluation value using equation (9).

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (\text{actual data}_i - \text{predicted data}_i)^2}{n}} \quad (9)$$

D. Data Denormalization

This process served to return the normalized value to the original value. Equation (10) shows data denormalization process:

$$v' = \frac{v}{\max v} \quad (10)$$

v' = New data

v = Present data

$\max v$ = the maximum value of the actual data of each parameter

VI. PROPOSED METHOD

The method used for inflation forecasting is PSO-ELM. PSO was used for weight optimization to obtain optimal input values to be used in ELM. Furthermore, the forecasting process will be carried out by the ELM method. The fitness formula that will be used in the equation (11) is as follows:

$$\text{fitness} = \frac{1}{1 + RMSE} \quad (11)$$

The initial process of the system is the initialization of particles at PSO, particles randomly generated between 0-1 in the form of real code numbers. These particles are used as weights from the input layer to the hidden layer. Then calculate the value of fitness based on equation (11). After knowing the fitness value then determined the value of PBest and Gbest, after that updates the velocity, then updates the position and calculates fitness again. After obtaining the optimal weighting value, ELM method testing is done, and the inflation forecasting results were based on the ELM method. For more details, see in Fig. 2.

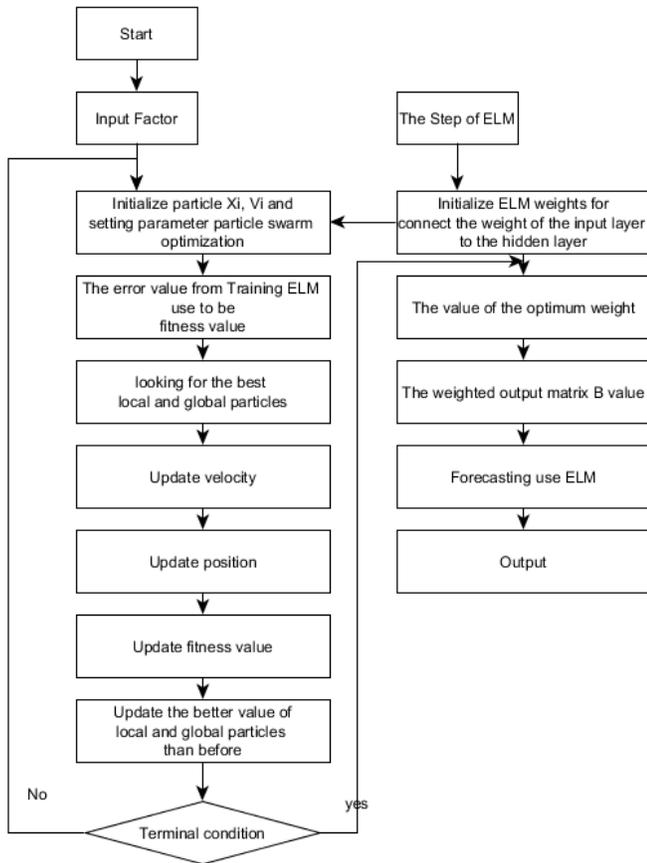


Fig. 2. Process PSO-ELM for Inflation Forecasting.

VII. EXPERIMENT AND RESULT

In forecasting the inflation rate using PSO and ELM, several tests were conducted which included particle testing, iteration testing, inertia weight, and testing the number of neurons in the hidden layer.

First tested object is particles number. It is to determine the optimal number of particles to get the maximum solution. The range of values used starts from 10-150 in multiples of 10. The range used is small because the PSO is good for searching narrow areas [16]. Other parameters use 100 iterations, inertia weight 0.6, and acceleration coefficients ($c_1 = 2$ and $c_2 = 1$).

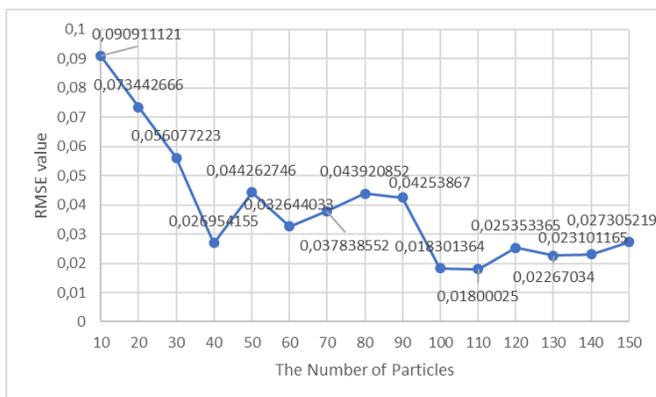


Fig. 3. Particle Chart Test.

In Fig. 3, it shows that there is a significant decrease in fitness value at the number of particles 40, then fluctuates to the number of particles 150. However, the number of particles 100 and 110 get the best fitness value with values of 0.018301364 and 0.01800025 with computation times are 9 and 10 seconds. So that in this test, the best number of particles is set at 110 with a computing value of 10 seconds. This is because the number of particles gives candidates more and varied solutions so that the search for the best solution can be done more thoroughly [16].

The next testing is the number of iterations. It is done to find out the relationship between the number of iterations and fitness values. Values are tested from 10 to 200 in multiples of 10. Other parameters use 110 on the number of particles, inertia weight 0.6, and acceleration coefficients ($c_1 = 2$ and $c_2 = 1$).

Fig. 4 shows that the best fitness value occurs in the number of iterations 110 of 0.023886. After that, between 120 and 200 the number of iterations fluctuates which form the same pattern with the greater fitness value. Therefore, this test sets the number of iterations 110 to be used in the next parameter.

Third testing is the inertia weight testing. It is used to produce the best weight to obtain optimal prediction results. This inertia weight testing starts from 0 to 1 with an addition of 0.1 and is done 6 times [14]. Fig. 5 shows the graph of the test results.

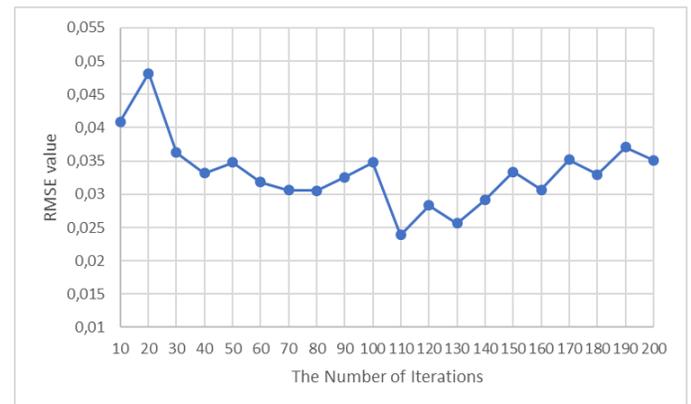


Fig. 4. Iteration Chart Test.

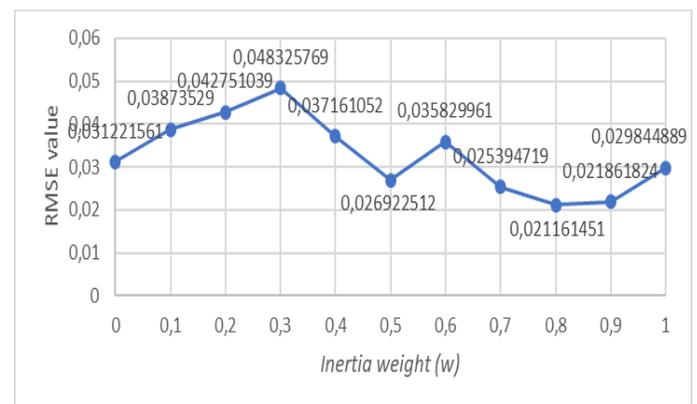


Fig. 5. Inertia Weight Chart of Testing.

Fig. 5 shows that the best fitness value of the value 0.8 with RMSE is 0.021161451. In a research conducted by Ratnavera [17], the best w value is above 0.5 and less than 1, because if w is more than 1, it will cause the particles in the PSO to be unstable due to uncontrolled speed produced. And it is proved in this study that when the value of w is 1 the RMSE value increases and it is higher than the value of 0.9.

Fourth tested object is Acceleration Coefficient. This Acceleration Coefficient test aims to determine the extent to which particles move in one iteration. This test is done by using the same combination of values and different between the range of values 1-4. Based on the book written by Cholissodin & Riyandani [18], the optimal value for the acceleration coefficient is stated in the range of values.

Fig. 6 shows that the best combination acceleration values occur in $c1$ and $c2$ worth 2 with the RMSE value of 0.020203. The values of $c1$ and $c2$ influence the motion direction of a particle whether towards local best or global best. In order to obtain optimal results, the two parameters' values between $c1$ and $c2$ are not dominant or at least close to balance, so that the particle movement route can be in accordance with the right portion to get the optimal solution [17].

The last testing is the number of neurons in the hidden layer. This test used the number of neurons, which is 3, 5 and 7. This amount is based on research [19] assume that the more number of neurons used, the more complex in determining the weight of neurons and the computing requires quite long time.

Fig. 7 shows that the error value decreases and the computing time needed is too long on adding the number of neurons. Hence, in this study, the number of neurons used was 3 neurons.

Based on all tests that have been carried out on each parameter, the authors set the architecture on PSO-ELM for forecasting the inflation rate with optimal results using 100 particles, 110 iterations, inertia weight = 0.8, acceleration coefficient = 2 and the number of neurons in the hidden layer are 3 neurons that took 9 seconds in computing time.

After the architecture is set to obtain optimal results, then the comparison between the other methods is carried out. The other methods are the backpropagation, ELM, and GA-ELM method. Table I shows the results of the comparison method.

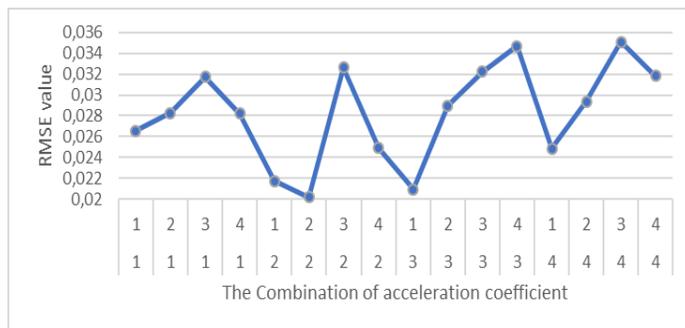


Fig. 6. Acceleration Coefficient Chart of Testing

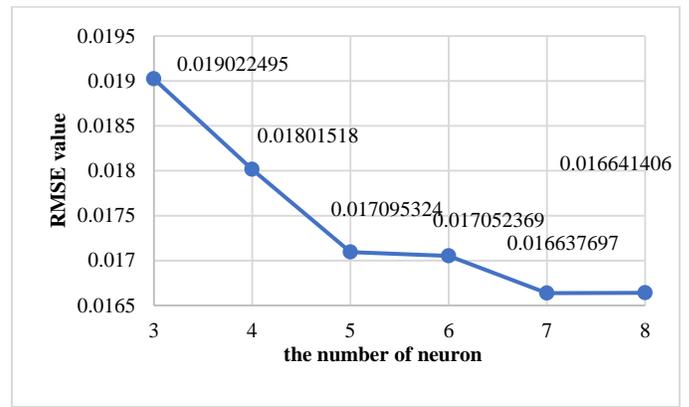


Fig. 7. The Number of Neurons Testing.

TABLE I. COMPARISON OF METHODS

Methods	Number of Hidden	Iteration	RMSE	Computation Time
Backpro	70	900	1.16035821	5 second
ELM	20	1	0.0202008	0 second
GA-ELM	7	300	0.013811	228 minutes 10 seconds
PSO-ELM	3	100	0.020202758	5 second

Table I shows that the results of four methods had very significant computation time between backpropagation, ELM, Hybrid GA-ELM and Hybrid PSO-ELM. The method that has the longest computation time is the hybrid of GA-ELM = 228 minutes 10 seconds, followed by backpropagation and hybrid PSO-ELM = 5 seconds, and the fastest is ELM = 0 seconds computation time.

Although backpropagation and PSO-ELM has the same computational time, backpropagation takes higher error value than hybrid PSO-ELM. Therefore, the PSO method can help the ELM method in decreasing the error value generated for predictions.

The case is different between ELM and PSO-ELM where ELM requires a shorter computational time than the PSO-ELM method (5 seconds). However, it is not a problem in system performance because the resulting error is relatively the same.

VIII. CONCLUSION

Based on the tests that have been executed, it is confirmed that the performance between the single ELM and Hybrid PSO-ELM methods has the same performance, proved by the difference in error values of 0.0000019. Thus, both methods are very relevant for solving forecasting problems.

The performance of the proposed method will be more optimal if the researcher uses more data to solve more complex problems. For further research, it can increase the PSO method to obtain the same time computing value with a single ELM method.

REFERENCES

- [1] N. G. Mankiw, *Macroeconomics* fifth edition, 9 edition. New York: Worth Publishers, 2010.
- [2] M. Anwar and T. Chawaa, "Post-ITF Inflation Expectation Analysis," *Work. Pap.*, vol. 9, no. Bank Indonesia, 2008.
- [3] A. Kadir, P. R. Widodo, and G. R. Suryani, *Implementation of Monetary Policy in the Inflation Targeting Framework in Indonesia*, no. 21. 2008.
- [4] F. Rachman, "Is Inflation Target Announced by Bank Indonesia the Most Accurate Inflation Forecast?," *Econ. Financ. Indones.*, vol. 62, no. 2, pp. 98–120, 2016.
- [5] S. Kamley, S. Jaloree, and R. S. Thakur, "Performance forecasting of share market using machine learning techniques: A review," *Int. J. Electr. Comput. Eng.*, vol. 6, no. 6, pp. 3196–3204, 2016.
- [6] D. Semaan, A. Harb, and A. Kassem, "Forecasting exchange rates: Artificial neural networks vs regression," *2014 3rd Int. Conf. e-Technologies Networks Dev. ICeND 2014*, vol. 2, pp. 156–161, 2014.
- [7] K. Huang and T. H. K. Yu, "The application of neural networks to forecast fuzzy time series," *Phys. A Elsevier*, vol. 363, no. 2, pp. 481–491, 2006.
- [8] L. Yu, Z. Danning, and C. Hongbing, "Prediction of length-of-day using extreme learning machine," *Geod. Geodyn.*, vol. 6, no. 2, pp. 151–159, 2015.
- [9] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feed forward neural networks," *2004 IEEE Int. Jt. Conf. Neural Networks (IEEE Cat. No.04CH37541)*, vol. 2, pp. 25–29, 2004.
- [10] N. R. Sari, W. F. Mahmudy, and A. P. Wibawa, "Backpropagation on Neural Network Method for Inflation Rate Forecasting in Indonesia," *Int. J. Adv. Soft Comput. Appl.*, vol. 8, no. 3, 2016.
- [11] N. R. Sari, W. F. Mahmudy, A. P. Wibawa, and E. Sonalitha, "Enabling External Factors for Inflation Rate Forecasting using Fuzzy Neural System," *Int. J. Electr. Comput. Eng.*, vol. 7, no. 5, pp. 2746–2756, 2017.
- [12] Y. P. Anggodo and I. Cholissodin, "Improve Interval Optimization of FLR using Auto-Speed Acceleration Algorithm," *Telecommunication, Comput. Electron. Control*, vol. 16, no. 1, pp. 1–12, 2017.
- [13] B. Indonesia, "Laporan Inflasi," 2017. [Online]. Available: <http://www.bi.go.id/id/moneter/inflasi/data/Default.aspx>.
- [14] W. Sun, C. Wang, and C. Zhang, "Factor analysis and forecasting of CO2 emissions in Hebei, using extreme learning machine based on particle swarm optimization," *J. Clean. Prod.*, vol. 162, pp. 1095–1101, 2017.
- [15] G. Bin Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.
- [16] A. N. Alfiyatin, W. F. Mahmudy, and Y. P. Anggodo, "Modelling Multi Regression with Particle Swarm Optimization Method to Food Production Forecasting," *J. Telecommun. Electron. Comput. Eng.*, vol. 10, no. 3, pp. 91–95, 2018.
- [17] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical Particle swarm optimizer with time varying acceleration coefficients," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, p. 240–255, 2004.
- [18] I. Cholissodin and E. Riyandani, *Swarm Intelligence*. Malang: Universitas Brawijaya, 2016.
- [19] M. C. C. Utomo, W. F. Mahmudy, and S. Anam, "Determining the Neuron Weights of Fuzzy Neural Networks Using Multi-Populations Particle Swarm Optimization for Rainfall Forecasting," *J. Telecommun. Electron. Comput. Eng.*, vol. 9, no. 2, pp. 37–42, 2017.