Usage of physical and chemical parameter for predicting water quality of Lake Toba using extreme learning machine

Romi Fadillah Rahmat, Eric Suwarno, Mohammad Fadly Syahputra, Maya Silvi Lydia

Abstract—Lake Toba serves as an important tourism attraction in Indonesia, especially in the region North Sumatera. The development of tourism in Lake Toba increases the concern of environmental issues, especially the water quality. Meanwhile, extreme learning machine (ELM) is a set of neural networks, which has the advantage in the calculation speed. In this paper, we implemented ELM to process water quality data and predict the water quality in Lake Toba. The prediction process is done through the combination of Python and MATLAB environment. Various activation functions is applied to the network to compare the root mean square error rate. The research shows that different activation functions and number of hidden neurons will affect the accuracy of the prediction process.

Index Terms—extreme learning machines (ELM), water quality, Lake Toba, artificial neural networks.

I. INTRODUCTION

HARO *et al.* [1] found that the water in Lake Toba, according to the result of the measurement process done in Haranggaol Horison district of Simalungun regency in North Sumatera province, is described as the water resource containing pollutants, ranging from low to medium level. The waste produced from households, industries, agricultural industries, and public transportations, are the main source of water pollution in Lake Toba. Moreover, the development of water hyacinth population and the waste from river streams flowing into Lake Toba, are also the source of water pollution in Lake Toba.

While the tourism industry develops in North Sumatera, mainly in Lake Toba, the possibility of environmental issues which occured in Lake Toba, mainly the water quality, is increased. The change of water quality will also affect the water ecosystem in Lake Toba. Therefore, a proper water quality assessment is required in order to control water quality in lake Toba.

Various method has been implemented to process water quality data. Artificial neural network, which has the same working mechanism with the biological brain, has been implemented in several researches, including screw insertion process [2], electric system stability monitoring [3], and wind turbine [4].

The main problem of prediction process using artificial neural network is the computation time, especially when the network receives big amount of data. Shibata & Ikeda [5] found that the number of hidden neurons used in the artificial neural network correlates with the learning speed of the network itself. Deng *et al.* [6] found that the backpropagation learning algorithm, which is introduced by Werbos [7] and Rumelhart *et al.* [8], has difficulty in processing data with the big size, which results in slower performance.

Researches have been done in improving the learning speed of the artificial neural network. Chandra & Sharma introduce parameterized multilayer perceptron [9] and parameterized deep neural network [10] to improve computational time of the artificial neural network. Hinton & Teh [11] develop the improved deep belief neural networks, resulting in faster learning speed.

Extreme learning machines (ELM) is one of the methods used to improve computational time in artificial neural network. ELM is proposed by Huang *et al.* [12] ELM is introduced to single hidden layer feedforward neural networks, by randomizing hidden layer neurons using Moore-Penrose inverse. This method results in the improvement of computational time.

Extreme learning machine has been implemented in various researches. Fu *et al.* [13] implemented extreme learning machine for liver tumor detection, by examining liver CT scan imagery. Pangaribuan & Suharjito [14] implemented extreme learning machine for diabetes mellitus diagnosis. Meanwhile, Zhai & Du [15] implemented extreme learning machine for vegetation species recognition.

In this research, the water quality data is processed using extreme learning machine. The water quality data is obtained from the research done by Rahmat *et al.* [16], and will be processed by the extreme learning machine. The result of this process will be compared according to the activation function used in the network.

II. THEORETICAL BACKGROUND

In this research, artificial neural network is applied with extreme learning machine, to increase the speed of prediction process. The theoretical background of the methodology used in this research is explained as follows:

A. Artificial neural networks

Hammerstrom [17] defines artificial neural networks as a computational structure, which is developed in line with the working mechanism of the biological brain. According to Uhrig [18], the artificial neural network consists of a set of processing elements, which are joined by the connection of input weights. The processing elements are arranged into the sequence of layers, most commonly classified as input layer,

hidden layer, and output layer. Each processing unit receives input from the connection, which will be calculated by the activation function of the unit itself. The result of the activation function will be passed to the other unit.

The two main operations performed by the artificial neural network is training and testing operation [18]. Training a neural network is needed once the architecture of the network has been constructed [19]. Meanwhile, the testing process is performed after training process is done.

Training process is the process where the neural network constructed for the application is given the random input weights. According to [20], the training process of the artificial neural network is classified to two methods, namely supervised training and unsupervised training. Supervised training is done to the artificial neural network by giving the network sample data with targeted result. Meanwhile, to perform unsupervised training on an artificial neural network, a sample data is processed by the network, without a finite final result.

B. Extreme learning machine (ELM)

According to Sun *et al.* [21], extreme learning machines (ELM) refers to a learning method applied in artificial neural networks. The architecture of neural network utilized in extreme learning machine is single hidden layer feedforward neural networks.

Extreme learning machines is proposed by Huang *et al.* [12] to increase the calculation speed of artificial neural networks, by randomizing the hidden layer. They stated that the feedforward neural network utilizes slow gradient based learning, which results in longer computational time. The randomization of the hidden layer results in faster computational speed, along with higher processing result accuracy.

A single hidden layer feedforward neural network is defined by (1):

$$f_n(x) = \sum_{i=1}^{n} G_i(x, a_i, b_i) * \beta_i, a_i \in R^d, b_i, \beta_i \in R \quad (1)$$

where $G_i(\cdot)$ refers to the activation function calculated in the ith hidden neuron, a_i refers to the input weight received by the ith hidden neuron from input neuron, b_i refers to the bias weight of the hidden neuron, and β_i refers to output weight of the hidden neuron.

For each additional nodes, G_i is defined from the additional node activation function g, as described by (2):

$$G_i(x, a_i, b_i) * \beta_i = g(a_i \times x + b_i) \quad (2)$$

Equation (3) is implemented when the hidden neuron implements RBF as the activation function.

$$G_i(x, a_i, b_i) * \beta_i = g(b_i \parallel x - a_i \parallel) \quad (3)$$

Suppose a training data set $N = \{(x_i, t_i) \mid x_i \in R_n, t_i \in R_m, i = 1, ..., L\}$, with x_i represents the training data, t_i represents the class label of the sample for each instance, and L is defined as the number of hidden nodes. When implementing the extreme learning machine for training the neural network, the steps are done as follows:

- Assign input weights w_i and biases b_i, where $i = 1, ..., L$,
- Calculate the hidden layer output matrix as H,
- Calculate the output weight, as defined in (4):

$$\beta = H^\dagger T \quad (4)$$

where $T = [t_1, ..., t_N]^T$ and $H^\dagger$ refers to the Moore-Penrose inverse of matrix H.

III. METHODOLOGY

This section provides explanation about the methodology used in the research, including the water quality data utilized in the research.

A. Utilized Data

This research utilizes water quality data obtained by Rahmat *et al.* [16]. The water quality data includes physical and chemical parameters recorded in subsequent time. The data is obtained in different locations, as shown in Fig. 1.
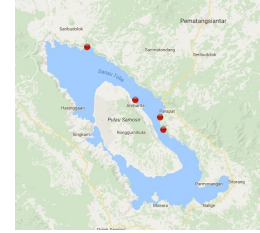

Fig. 1. Locations of Data Acquisition [16]

B. General Architecture

The general architecture of the application is shown by Fig. 2. The development of the application is done in combination of Python and MATLAB environment.
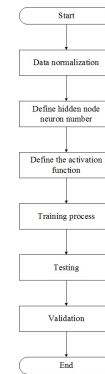

Fig. 2. Locations of Data Acquisition [16]

The general architecture of the application consists of six steps, which are explained as follows:

1) Data normalization: The water quality data is recorded by Rahmat *et al.* [16] by the format shown by Fig 3. The recorded values are separated by semicolon sign, thus needs rearrangement and normalization. According to Patro *et al.* [22] , normalization is a pre-processing stage

of the problem sets, by which the data will be scaled by the certain range, in order to enable the algorithm to work.

2) Define the number of hidden neuron: This process will be performed after data normalization process is done. According to Heaton [23] , the number of hidden neuron has to be defined by using following rules:
   - the number of hidden neuron ranges from the number of input neuron and the number of output neuron,
   - the number of hidden neuron can be defined as two thirds of the total input and output neuron, and
   - the number of hidden neuron is not higher than twice the number of input neuron

Various amount of hidden neuron will be implemented in this research for comparison of the prediction result. The number of hidden neuron varies from 1 to 20.

3) Define the activation function: Dorst [24] defines the activation function is the function implemented in the neuron to determine the state of the neuron in an artificial neural network. In this research, three activation functions will be implemented, respectively sigmoid, sine, and hardlim function, whose result will be compared.

4) Training process: The training process is a process performed in artificial neural network implementation. The training process is performed by using extreme learning machine in the following steps:
   - Initialize the hidden layer matrix H, consisting of input weights and neuron biases;
   - Calculate the hidden layer output from matrix H;
   - Calculate output weight matrix β, which is done according to (4).

5) Testing process: The testing process is performed after the training process is done. The testing process is performed to obtain the result of training process.

6) Validation process: The validation process is performed after the testing process is done. The validation process is performed to verify that the result generated in testing process conforms with the water quality index standard.

The output of the methodology in this research is a graph representing the result of prediction process by the extreme learning machine, followed by the root mean square error and the calculation time of the result.

IV. EXPERIMENT AND RESULT

The process of water quality prediction is done through combination of Python and MATLAB environment. The data normalization process is done through Python environment, while the water quality prediction process is performed through MATLAB environment.

A. Data Normalization

The data normalization process is done through Python environment. The initial data structure is not suitable to be processed with extreme learning machine, thus requiring normalization process.

Before the data is normalized, the initial dataset have to be filtered by each row, to ensure that the value of the recorded data is a valid numeric data. The result of the filtering process is a new dataset with valid numeric value.

The normalization process is performed by using min-max normalization[**?**] , thus resulting in a dataset with value ranging from -1 to 1, according to (5):

$$A' = \frac{A - A_{min}}{A_{max} - A_{min}} * (D - C) + C \tag{5}$$

where A' refers to the normalized data value of A, with the result ranging from $\mathbb{C}_D$. The result of the normalization process will be used by the extreme learning machine regression engine [25].

B. Water quality prediction

The water quality prediction process is done through MATLAB environment. The interface of the system is shown in Fig 4.

To perform the water quality prediction process, the training data and testing data have to be provided to the system. The default architecture of the artificial neural network, which will be constructed after training data is loaded, is SLFN with sigmoid function as the activation function. The number of hidden neuron is set to 1 by default, with the range of 1 to 20.

V. CONCLUSION

The conclusion goes here.

REFERENCES

[1] D. Haro, Y. Djayus and Z. Harahap, "Kondisi Kualitas Air Danau Toba di Kecamatan Haranggaol Horison Kabupaten Simalungun Sumatera Utara", *AQUACOASTMARINE*, vol. 1, no. 1, 2013.

[2] B. Lara, K. Althoefer and D. Seneviratne, "Use of artificial neural networks for the monitoring of screw insertions,"*Proceedings 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human and Environment Friendly Robots with High Intelligence and Emotional Quotients (Cat. No.99CH36289)*, Kyongju, 1999, pp. 579–584 vol.1.

[3] D. Popović, D. Kukolj, and F. Kulić, "Monitoring and assessment of voltage stability margins using artificial neural networks with a reduced input set," *IEE Proceedings - Generation, Transmission and Distribution*, vol. 145, no. 4, p. 355–362, 1998.

[4] R. Ata, "Artificial neural networks applications in wind energy systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 49, pp. 534–562, Sep. 2015.

[5] K. Shibata and Y. Ikeda, "Effect of number of hidden neurons on learning in large-scale layered neural networks," in *ICCAS-SICE*, Fukuoka: IEEE, 2009, pp. 5008–5013.

[6] C. Deng, G. Huang, J. Xu, and J. Tang, "Extreme learning machines: New trends and applications," *Science China Information Sciences*, vol. 58, no. 2, pp. 020301:1–020301:16, Jan. 2015.

[7] P. Werbos. "Beyond regression: new tools for prediction and analysis in the behavioral sciences," Harvard University, Cambridge, Massachusets, 1974.

[8] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[9] B. Chandra and R. K. Sharma, "Fast learning for big data applications using parameterized multilayer perceptron," *2014 IEEE International Conference on Big Data (Big Data)*, pp. 17–22, 2014.

[10] B. Chandra, and R. K. Sharma. "Fast learning in Deep Neural Networks." *Neurocomputing*, vol. 171, pp. 1205–1215, 2016.

[11] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[12] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489–501, Dec. 2006.

[13] F. Huixuan, W. Yuchao and Z. Hongmei, "Ship rolling motion prediction based on extreme learning machine," *2015 34th Chinese Control Conference (CCC)*, Hangzhou, 2015, pp. 3468–3472.

[14] J. J. Pangaribuan and Suharjito, "Diagnosis of diabetes mellitus using extreme learning machine," *2014 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung, 2014, pp. 33–38.

[15] C.-M. Zhai and J.-X. Du, "Applying extreme learning machine to plant species identification," *2008 International Conference on Information and Automation*, Changsha, 2008, pp. 879–884.

[16] R. F. Rahmat, Athmanathan, M. F. Syahputra dan M. S. Lydia, "Real Time Monitoring System for Water Pollution in Lake Toba," in *Proceedings of ICIC 2016*, Lombok, 2016.

[17] D. Hammerstrom, "Neural networks at work," in *IEEE Spectrum*, vol. 30, no. 6, pp. 26–32, June 1993.

[18] R. E. Uhrig, "Introduction to artificial neural networks," *Industrial Electronics, Control, and Instrumentation, 1995., Proceedings of the 1995 IEEE IECON 21st International Conference on*, Orlando, FL, 1995, pp. 33–37 vol.1.

[19] E. Reingnold and J. Nightingale, "Training an artificial neural network," in *Artificial Neural Networks Technology*, University of Toronto. [Online]. Available: http://www.psych.utoronto.ca/users/reingold/courses/ai/cache/neural3.html. Accessed: Nov. 18, 2016.

[20] M. van Heeswijk, "Advances in extreme learning machines," Aalto University, 2015.

[21] M. Gao, W. Xu, H. Fu, M. Wang and X. Liang, "A Novel Forecasting Method for Large-Scale Sales Prediction Using Extreme Learning Machine," *2014 Seventh International Joint Conference on Computational Sciences and Optimization*, Beijing, 2014, pp. 602–606.

[22] Patro, G. S. Krishna, and K. Kumar, "Title: Normalization: A Preprocessing stage," 2015. [Online]. Available: http://arxiv.org/abs/1503.06462. Accessed: Jan. 5, 2017.

[23] J. Heaton, *Introduction to neural networks for Java, 2nd ed*. New York, NY, United States: Heaton Research, United States, 2008.

[24] L. Dorst, "Neural activation functions," in *Applications of basic mathematics in Computer Science*. [Online]. Available: https://staff.science.uva.nl/l.dorst/math/sigma.pdf. Accessed: Nov. 22, 2016.

[25] Q.-Y. Zhu and G.-B. Huang, "Extreme learning machines," 2013. [Online]. Available: http://www.ntu.edu.sg/home/egbhuang/elm_random_hidden_nodes.html. Accessed: Jan. 5, 2017.