ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING WATER QUALITY CLASSIFICATION IN LAKE TOBA WITH DEEP LEARNING APPROACH

SARAH PURNAMAWATI[1], ROMI FADILLAH RAHMAT[2*], MUHAMMAD ISA DADI HASIBUAN[3], IQBAL FAKHRIZA[4], FARHAD NADI[5], SHARFINA FAZA[6], RAHMAT BUDIARTO[7]

[1,2,3,4] Department of Information Technology, Universitas Sumatera Utara, Medan, Indonesia
[5] School of Information Technology, UNITAR International University, Selangor, Malaysia
[6] Department of Computer Engineering and Informatics, Politeknik Negeri Medan, Medan, Indonesia
[7] College of Computer Science and Information Technology, Al Baha University, Saudi Arabia
*Corresponding E-mail: romi.fadillah@usu.ac.id

ABSTRACT

Lake Toba serves as a crucial water resource that is extensively utilized for various purposes. However, different water types have distinct uses, necessitating proper classification. Government Regulation No. 82 of 2001 establishes water quality standards to classify water into several categories, each indicating its appropriate use. This study proposes the use of a deep neural network (DNN) to classify the water quality of Lake Toba. The research explores different activation functions—Softmax, ReLU, and Sigmoid—along with the SGD, RMSProp, and Adam optimizers. To determine the most effective model architecture, each activation function was tested in combination with different optimizers. The findings indicate that deep neural networks (DNNs) can be effectively utilized for water quality classification, with accuracy and error rates influenced by the activation function, optimizer, number of neurons, and number of hidden layers. The dataset used in this study includes measurements of water temperature, pH level, dissolved oxygen concentration, oxidation-reduction potential, air temperature, and humidity, which are essential for monitoring the water quality of Lake Toba. The testing process consists of two approaches: (1) classification using three parameters, based on Government Regulation No. 82 of 2001, and (2) classification using six parameters. Each test is conducted using the same model architecture. The highest recorded accuracy in the experiments was 99% (0.998402), with the lowest recorded loss at 0.014616. These results were obtained from studies utilizing three parameters.

Keywords: *Lake Toba; Water quality Classification; Deep Neural Network (DNN); Machine Learning; Government Regulation No. 82 of 2001*

1. INTRODUCTION

Lake Toba, which is one of the largest lakes in Indonesia, was formed by the volcanic eruptions. It is still a vital water resource in North Sumatra Province. However, water quality has become a situation over the past decade. As one of the essentials for human life after air [1], water needs a careful quality assessment. Regular monitoring of the quality of water in Lake Toba is essential to ensure the standards concerning health and environmental regulations.

The society living around Lake Toba relies on the lake's water for many purposes, including daily needs such as drinking, cooking, and washing. Even irrigation, recreation, and fishing need it. However, different regions of the lake differ on water quality. Some areas have good water quality compared to others. This water quality problem is due to human activities and the environmental factors. Other instances, such as air pollution, also introduce harmful substances like nitrogen dioxide and sulfur dioxide into the environment [2].

The diverse topography of the lake's surroundings, which includes mountainous and hilly terrains, has an impact on these variations. Human activities generate various forms of waste, which impact the aquatic environment and overall water quality [3]. According to the World Health Organization, unsafe drinking water and inadequate sanitation are responsible for 80% of illnesses in developing countries [4]. Water is a vital resource for agricultural irrigation, industrial processes, and daily life. Its quality has a direct impact on agriculture,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

economic growth, and public health [5]. In Indonesia, water quality standards are regulated under Government Regulation No. 82 of 2001, which classifies water into different categories based on its appropriate usage.

Water quality classification is essential to assess the suitability of lake water for various activities in different regions. The water quality problem is seen as the most important social, scientific, and technological problem of our time [6]. Understanding water quality serves as a benchmark for the government in establishing regulations [3]. Traditionally, water classification involves the collection of samples from various locations, followed by laboratory analysis [7]. But this approach takes a long time and causes delays. Therefore, an effective system is required to improve the precision and efficiency of water quality classification.

In aquatic environment research, machine learning models have been employed for the construction, monitoring, simulation, evaluation, and optimization of various water treatment and management systems [8]. One strategy is to use machine learning algorithms in conjunction with sensors. Data obtained from sensors, including pH and dissolved oxygen (DO) measurements, can be classified using machine learning techniques. However, machine learning models frequently encounter performance difficulties while processing large datasets. Deep neural networks (DNNs) can be employed to mitigate these limitations. Although deep neural networks (DNNs) may occasionally exhibit performance comparable to or worse than classic machine learning techniques with small datasets, their efficacy markedly enhances with larger dataset sizes. Recent research indicates that deep neural networks (DNNs) can get accuracy rates of up to 99% in water quality classification tasks, highlighting their efficacy for environmental monitoring [9]. Furthermore, DNNs have demonstrated efficacy in the identification, prediction, and classification of data [10].

Several studies have investigated water quality classification using neural networks. The backpropagation neural network (BPNN) has been used as a classifier for water quality assessment [11], producing classification results that enable better water quality management and pollution control. Compared to traditional evaluation methods, BPNN provides more objective and reliable assessments. Additionally, this model demonstrates high flexibility and adaptability. Xue Xicheng and Chen Yan [12] explored the evaluation of rainwater quality using a radial basis function artificial neural network (RBF-ANN). Due to its superior approximation capabilities, fast training speed, ability to avoid local minima, and resilience to subjective biases, the RBF-ANN model is highly suitable for comprehensive water quality evaluation.

Wang et al. [3] conducted a study employing the Long Short-Term Memory Neural Network (LSTM NN) methodology for water quality prediction, comparing it with Backpropagation Neural Networks (BPNN) and Extreme Learning Machines (ELM). The results indicated that the LSTM neural network attained superior accuracy compared to both the BPNN and ELM. Reference [13] investigated the classification and monitoring of water quality in shrimp aquaculture utilizing electronic nose (e-nose) and electronic tongue (e-tongue) technology. The study indicated a discriminant accuracy of 86.02% for Function 1 and 8.82% for Function 2 with the e-nose, but the e-tongue attained an accuracy of 84.5% for Function 1 and 15.2% for Function 2. Furthermore, Zhu and Hao [1] evaluated water quality with a fuzzy neural network (FNN), illustrating that FNN enhances the precision of water quality assessments, rendering it a dependable and effective method.

Previous studies indicate that methods employing neural networks (NNs) frequently yield superior outcomes compared to traditional machine learning techniques. Furthermore, research comparing various types of neural networks has demonstrated that certain models are more effective or better suited for particular problems than others. Moreover, inside neural networks, the efficacy of each approach fluctuates depending on the specific problem being tackled. In this study, the method used is a DNN [1], [11]-[13].

Other studies have also employed traditional machine learning methods for water quality classification. For instance, N. Raviteja et al. utilized Support Vector Machine (SVM) and achieved an accuracy of 83% [14], while Salisu Yusuf Muhammad et al. applied the K-Star algorithm, obtaining an accuracy of 86.67% [15]. Additionally, Theyazn H. H. Aldhyani et al. implemented a Nonlinear Autoregressive Neural Network (NARNET) for water quality prediction, achieving the highest accuracy of 97.01% [16].

Another study introduced the BS-FAMLP model, a hybrid approach combining Gradient-Boosted Decision Trees (GBDT) and Multilayer Perceptron (MLP) to improve water quality classification. By using Bayesian optimization to fine-tune hyperparameters and a feature-weighting attention mechanism, the model adjusts the importance of features, making it more accurate. Tested on a groundwater dataset with 188,623 samples, it achieved 96.16% accuracy. This study highlights the effectiveness of combining machine learning techniques to improve classification performance and streamline water quality assessment [17].

Prior research indicates a distinct necessity for more effective and precise methods for categorizing water quality tailored for extensive freshwater bodies with intricate ecosystems, such as Lake Toba. Our efforts specifically concentrate on the application of DNN technology to the intricate topographical and climatic conditions of Lake Toba. This research fills an important gap between advanced computer techniques and essential environmental monitoring needs, which could change how we assess water quality in large freshwater systems affected by different human actions and environmental pressures.

Water is categorized mostly according to its degree of use rather than its degree of contamination in this study. The used data include water quality monitoring data conducted by Rahmat et al. [7]. Subsequently, the data will undergo classification using deep neural network (DNN). The presented findings consist of accuracy and error graphs, as well as Excel files with testing and classification outcomes. In this study, the authors propose a deep neural network (DNN) to classify water quality and apply it to the classification of Lake Toba water.

2. METHODOLOGY

2.1 Water Quality

Government Regulation No. 82 of 2001 concerning Environmental Management governs water quality standards in Indonesia. According to this regulation, water quality is categorized into four classes based on its intended use:

i. Class I: Suitable for drinking water or other uses requiring similar water quality;
ii. Class II: Suitable for recreational facilities, freshwater fish farming, livestock, and agriculture;
iii. Class III: Suitable for freshwater fish farming, livestock, and agriculture;
iv. Class IV: Suitable for irrigation of crops or agriculture.

Each of these water classes represents specific water quality standards deemed suitable for their respective uses. The criteria for water quality are assessed based on five groups of parameters: physical properties, inorganic chemistry, organic chemistry, microbiology, and radioactivity. One key parameter in the group of inorganic chemicals is pH. Additional parameters used to evaluate and classify water quality standards are presented in Table 1.

2.2 Proposed Method

Deep Neural Networks (DNNs) are an evolution of Neural Networks (NNs), also known as Artificial Neural Networks (ANNs), which are computational models inspired by the structure of biological neural networks [18]. Early versions of NNs were shallow, typically consisting of one input layer, one output layer, and at most one hidden layer between the two. In contrast, DNNs are characterized by having three or more layers, including the input and output layers. Essentially, a DNN is an ANN with multiple hidden layers sandwiched between the input and output layers [19].

In DNNs, each layer of nodes (neurons) trains a different set of features based on the output from the previous layer. The deeper the network, the more complex the features it can extract, as each layer combines and refines the features learned by the previous layer.

Table 1: Water Quality Standards According to Government Regulation No. 82/2001

Parameter Group	Parameter	Unit	Class			
			Class I	Class II	Class III	Class IV
Physical Properties	Water Temperature	°C	Deviation of 3 from natural temperature	Deviation of 3 from natural temperature	Deviation of 3 from natural temperature	Deviation of 5 from natural temperature
Inorganic Chemistry	pH	-	6–9	6–9	6–9	5–9
	Dissolved Oxygen (DO)	mg/L	≥6	≥4	≥3	≥0

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Sensors used for data collection

Sensor	Measurement	Unit
DS18S20 Probe Sensor	water temperature	°C
Analog pH Meter Pro	pH	-
Dissolved Oxygen Sensor Kit	Dissolved Oxygen (DO)	mg/L
DHT11 Sensor	Air Humidity	RH
DFRobot ORP Meter	Oxidation-reduction potential (ORP)	mV
DS18S20 Probe Sensor	water temperature	°C

A function called the activation is applied to input values, which results in an activation level of the neuron, which is the output value of the neuron. There are a number of functions that can be used in neurons. The activation function determines which neurons must be active and not. One of the most commonly used functions is the step function or linear threshold function.

In this approach, inputs to a neuron are summed (each multiplied by its respective weight), and the result is compared to the threshold t. If the sum exceeds the threshold t, the neuron is activated with an activation level of 1. Conversely, if the sum is below the threshold t, the neuron is inactive with an activation level of 0.

In some networks, when the input value does not exceed the threshold t, the activation level is set to 1 instead of 0. After determining the output, the network adjusts its weights to optimize performance. This adjustment process is guided by an algorithm called an optimizer. Common optimizers used are Stochastic Gradient Descent (SGD), RMSProp, and Adam. These optimizers play a crucial role in minimizing the error and improving the model's learning efficiency.

2.3 Dataset

The dataset used in this study focuses on monitoring the water quality of Lake Toba. Data collection was conducted for two days, specifically on October 25-26, 2016. The dataset includes measurements of water temperature, pH, dissolved oxygen (DO) levels, oxidation-reduction potential (ORP), air temperature, and humidity. Measurements were taken at the following locations:

i. Haranggaol Horison, Simalungun Regency;
ii. Ajibata, Toba Samosir Regency;
iii. Parapat, Simalungun Regency.

The sensors used for data collection in this research are presented in Table 2. The sensors were immersed in the water for 1-2 days, allowing the parameter values to change naturally over time.

2.4 General Architecture

The classification of Lake Toba's water quality is determined based on data collected in the research conducted by Rahmat et al. [7], which follows a systematic procedure. Generally, the procedure consists of three main steps: preprocessing, training, and testing. Prior to use, the data undergoes initial processing. Following this, the DNN model is constructed in preparation for the training and testing phases. Once the model is built, training is conducted using the processed data. The procedure then continues with the testing phase. Upon completion, a graph is generated to illustrate the accuracy and loss/error incurred during the training and testing phases. Further details about the classification process can be observed in the overall architecture presented in Figure 1.

In the preprocessing stage, the first step was data cleaning, which aims to remove rows of data with empty values, mismatched indices, or invalid data entries. Then the data is split into training and testing datasets. Each dataset is then further divided into features and labels. The subsequent step involves normalizing the features using the Least Absolute Deviations (LAD) method. Mathematically, the vector magnitude (norm) calculated using LAD can be represented as shown in Eq. (1).

$$S = \sum_{\{i=1\}}^{n} |x_i| \tag{1}$$

This calculation basically adds up all entries (x) from x_1 to x_n. After getting the norm from line n, the new value of x can be seen in Eq. (2).

$$x_{new} = x\frac{1}{s} \tag{2}$$

Once normalized, the feature data values are transformed to range between -1 and 1. The final process involves encoding the label data. This is necessary because the labels, which will serve as the output in the application, are in categorical form.

The training phase is the stage where the DNN is implemented. The normalized and encoded training data is used during this phase. The process begins with each input data (feature) being fed into the nodes of the input layer. Then, the weights and biases are initialized randomly, with their values ranging between 0 and 1. During each batch, the weights are updated based on the optimizer algorithm used. In this study, one of the optimizers applied is Stochastic Gradient Descent (SGD).

SGD updates the model parameter (θ) in the negative direction of the gradient (g) by taking a subset or mini-batch data size (m). Neural network is represented by $f\left(x^{(i)};\theta\right)$ where $x^{(i)}$ is training data and $y^{(i)}$ is a training label, gradient loss L is calculated by observing the model parameters θ. Learning rate (ϵ_k) determines the size of the steps taken by the algorithm along the gradient (in the negative direction in the case of minimization and in a positive direction in the case of maximization). The SGD mathematical notation can be seen in Eq. (3) and Eq. (4).

$$g = \frac{1}{m}\nabla_\theta \sum_i L\left(f\left(x^{(i)};\theta\right), y^{(i)}\right) \quad (3)$$

$$\theta = \theta - \epsilon_k \times g \quad (4)$$

While the calculation to get the output from node i to node j can be seen in Eq. (5).

$$X_j = \sum_{i=1}^{n} x_i \cdot w_{ij} - \theta_j \quad (5)$$

n is the number of inputs to node j; w_{ij} is the weight between nodes i and j; θ_j is the threshold value (bias) for node j, which is a random value between 0 and 1; x_i is the input value for node i; and X_j is the output of node i which is also the input for node j.

The calculation results of each node will be propagated into the activation function to get the active node. Because this study is about classification, the activation function in the output layer uses *softmax*. This activation function is intended to handle data with categorical output. The formula of the *softmax* activation function can be seen in Eq. (6).

$$\sigma(z) = \frac{e^z}{\sum_{k=1}^{K} e^z k} \quad (6)$$

z is the result of calculations in the layer, e is the error or loss value, k is the number of dimensions of the label (output).

After getting the active node, the calculation goes to the next layer and goes to the output layer that keeps repeating until the specified iteration. The last stage is testing. At this stage, the model that has undergone the learning stage is tested to see the performance of the model. Normalized data testing and encoding are used at this stage.

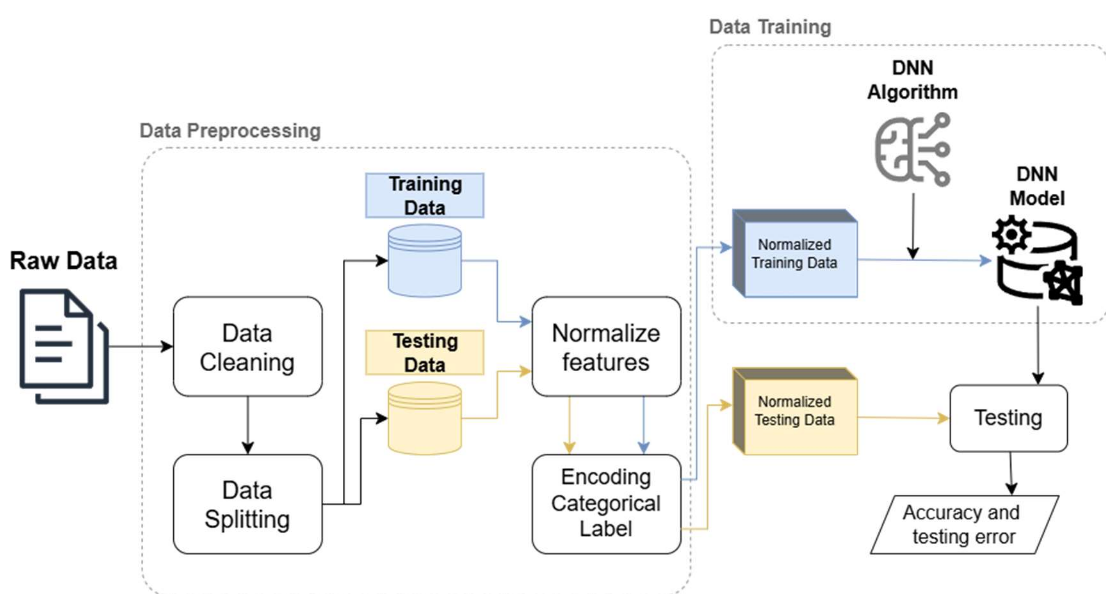


Figure 1: General Architecture

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3. RESULTS

The first test uses three parameters based on Government Regulation No. 82 of 2001, while the second test incorporates all parameters in the dataset. The preprocessing stage involves data cleaning, data splitting, separating features and labels, normalizing features, and encoding labels. The results of the data cleaning process are shown in Table 2.

Table 2 highlights the number of rows remaining after data cleaning. Many rows were removed due to the presence of invalid data lines, such as values that could not be interpreted as numeric or rows where the number of columns exceeded or fell short of the expected count. After cleaning the data, all files were merged into a single dataset. Following the data cleaning process, the dataset is split into training and testing subsets. The result of these processes is presented in Table 3.

Table 3: Number of rows in the dataset before and after data cleaning

File Name	Number of data rows	
	Before	After
Dataset.txt	15718	15642

The distribution of water quality classes in the dataset is presented in Figure 3. The majority of the dataset is classified as Class II (7,172 samples) and Class I (5,597 samples), signifying that most water samples meet relatively high-quality standards. Conversely, Class III (4 samples) and Class IV (2,869 samples) have significantly fewer data points, indicating that instances of lower water quality are less frequent in the dataset.

The classification results for each region are presented in Table 4. From the table, several key observations can be made. In Ajibata, 1,301 out of 1,400 rows were classified as Class II. In Haranggaol, 5,937 out of 6,058 rows were classified as Class I, while in Parapat, 7,318 out of 8,200 rows were classified as Class II. These results indicate that the water quality in Ajibata and Parapat falls under Class II, whereas in Haranggaol, it is classified as Class I.

The next step involves preparing the data by separating it into features and labels. After identifying which columns correspond to features and labels, the feature data is normalized, and the label data is encoded. The effectiveness of classification depends on the number of waveforms accurately recognized as belonging to a specific category [20]. To better understand the relationships between different water quality parameters, a correlation heatmap is presented. This visualization highlights the degree of correlation between each feature, helping to identify patterns and dependencies within the dataset. Figure 3 presents the correlation heatmap of the selected parameters.

Figure 2: Distribution of water quality classes after data cleaning and labeling, showing the number of observations for Class III and Class IV.

Once the preprocessing phase is complete, the training phase begins. Training is conducted using varying numbers of neurons and hidden layers. Additionally, different activation functions and optimizers are tested. The activation functions used in this research include softmax, ReLU, and sigmoid, while the optimizers tested include SGD, RMSProp, and Adam.

Initially, the experiment is conducted using three parameters: pH, DO, and water temperature. After identifying the most optimal combination of activation function, optimizer, hidden layers, and neurons, three additional parameters—air temperature, ORP, and air humidity—are introduced.

Table 4: Experiment results with the most optimal activation function and optimizer trained using six features sorted in descending order based on the highest training accuracy

Location	Class				Water Quality
	Class I	Class II	Class III	Class IV	
Ajibata	72	1280	2	28	Class II
Haranggaol	5476	1	1	526	Class I
Parapat	49	5891	1	2285	Class II

During the first phase, a series of tests is performed to determine the most suitable activation function and optimizer. The loss/error value is calculated using the categorical cross-entropy function, which is the most appropriate loss function for multi-label data. At this stage, the number of hidden layers and neurons is not yet optimized, so the initial values used are 5 hidden layers and 64 neurons. Based on the results of experiments with different activation functions and optimizers, the final outcomes are summarized in Table 5.

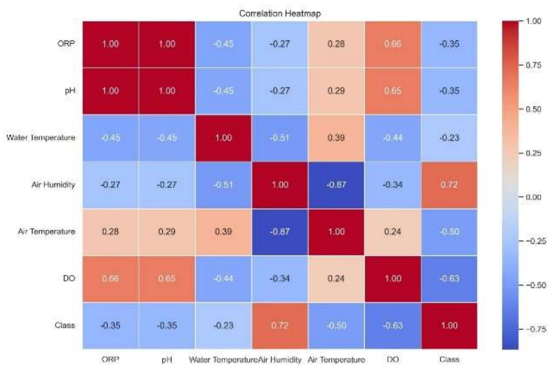


Figure 3: Correlation heatmap showing the relationships between ORP, pH, water temperature, air humidity, air temperature, and class with values ranging from -1 to 1.

Table 5 presents the experimental results using different optimizers and activation functions. Each experiment was trained for 1,000 epochs with an early stopping mechanism based on the loss value, with a patience parameter of 30. The dataset was split into 80% for training and 20% for testing. The experiments were conducted using three input parameters: pH, water temperature, and dissolved oxygen (DO). The learning rate was set to 0.001, with a batch size of 32, five hidden layers, and 64 neurons per layer. The table is sorted in descending order based on training accuracy. The highest testing accuracy of 0.998402 was achieved using the SGD optimizer with the ReLU activation function. The graph illustrating the highest accuracy levels is shown in Figure 5. The graphs illustrating the highest and lowest accuracy levels are presented in Figures 7 and 8.

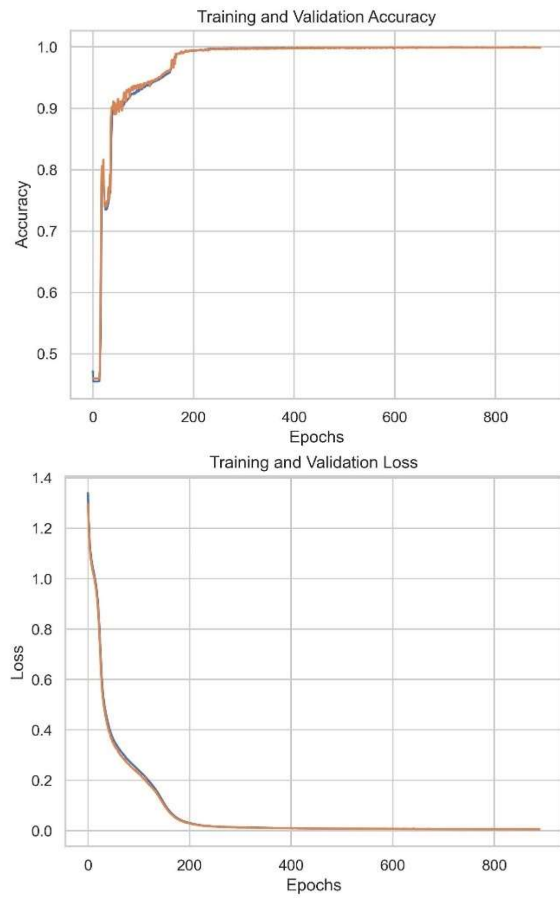


Figure 4: Training and validation accuracy and loss over epochs for the most optimal combination of optimizer and activation function, trained using three features.

Table 5: Experiment results with different activation and optimizer sorted in descending order based on the highest training accuracy

No	Optimizer	Activation	Accuracy		Loss/Error		Time to Train (s)
			Training	Testing	Training	Testing	
1	SGD	ReLu	0.999200	0.998402	0.004591	0.012583	531.231674
2	Adamax	ReLu	0.998801	0.998402	0.004512	0.006947	63.9516453
3	AdamGrad	ReLu	0.998101	0.998402	0.004512	0.006947	63.9516453
4	Adam	ReLu	0.997502	0.988494	0.009268	0.047629	26.2919938
5	RMSProp	ReLu	0.995804	0.998721	0.013465	0.004003	40.6875546
6	Adadelta	ReLu	0.990809	0.991371	0.049889	0.056193	547.378306
7	Adamax	Sigmoid	0.936963	0.937999	0.169442	0.185234	293.328241
8	Adam	Sigmoid	0.934665	0.937360	0.170226	0.178026	81.2457304
9	RMSProp	Sigmoid	0.934565	0.928731	0.169874	0.189297	202.118056

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 6: Experiment results with the most optimal activation function and optimizer trained using six features sorted in descending order based on the highest training accuracy

No	Number of Hidden Layers	Number of Neurons	Batch Size	Accuracy		Loss/Error		Time to Train (s)
				Training	Testing	Training	Testing	
1	5	96	32	0.999200	0.998402	0.004245	0.014616	647.373745
2	5	64	32	0.999001	0.998721	0.005340	0.009932	642.292580
3	3	32	16	0.998901	0.998082	0.006922	0.021817	907.060529
4	4	48	32	0.998901	0.999041	0.007405	0.011390	515.858443
5	6	64	16	0.998701	0.998721	0.004018	0.012343	956.276379
6	6	32	16	0.998401	0.998082	0.005410	0.012373	670.685301
7	5	64	64	0.998301	0.997762	0.009093	0.014748	359.039999
8	4	128	64	0.998001	0.996164	0.014139	0.025046	319.581876
9	5	80	64	0.997402	0.994886	0.013532	0.020674	443.252383
10	7	48	32	0.991208	0.986896	0.033836	0.040665	522.096319

The confusion matrix for the most optimal combination of optimizer and activation function shown in Figure 5 reveals the model's strong predictive performance across three of the four classes, demonstrating high accuracy in classifying classes I, II, and IV with 1109, 1466, and 549 correct predictions, respectively.

The minimal misclassification rates for these classes indicate robust model performance. The model's inability to correctly identify class III can be attributed to the severe class imbalance in the dataset, where class III is represented by only 4 instances. This extreme data scarcity for class III makes it statistically challenging for the model to learn meaningful patterns for this category

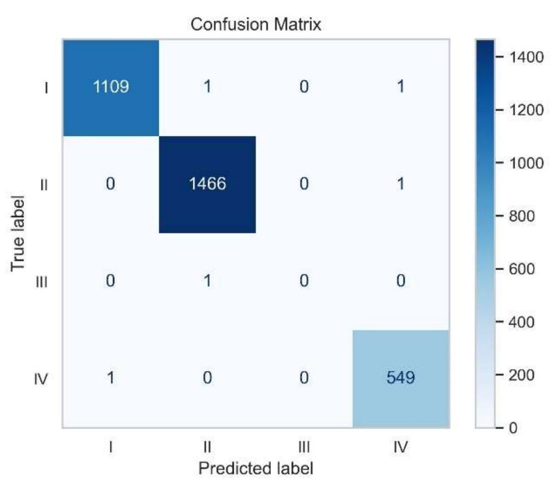


Figure 5: Confusion Matrix of the testing results for the most optimal combination of optimizer and activation function, trained using three features.

After determining the ideal model architecture, the next step involves conducting experiments using six parameters. During these experiments, we evaluate the loss and accuracy while adjusting other hyperparameters, such as the number of hidden layers, neurons, and batch size, to optimize performance. The results of these experiments are presented in Table 6.

Experiments using six parameters with the same model architecture and training settings achieved the highest training and testing accuracy of 0.999200 and 0.998402, respectively. These values are exactly the same as the results obtained using three parameters, which yielded accuracies of 0.999200 and 0.998402. The training and testing loss for the six-parameter experiment were 0.004245 and 0.014616, while the three-parameter experiment achieved lower loss values of 0.004591 and 0.012583. Although the loss values for the three-parameter experiment with the most optimal hyperparameters were slightly higher, the difference is relatively small.

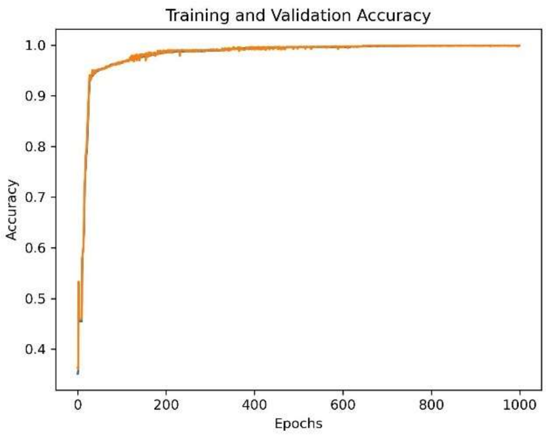
Regarding execution time, the six-parameter experiment took 647.373745 seconds, whereas the three-parameter experiment completed in 531.231674 seconds, making the latter 13.3004% faster. The longer execution time of the six-parameter model is attributed to the increased number of features, which requires additional computational resources during training. The training and validation accuracy, as well as the loss of the most optimal six-parameter model, are presented in Figure 6. The confusion matrix is also presented in Figure 7.

A previous study also utilized this dataset but classified water quality differently. It applied the Decree of the Minister for the Environment Number 115 of 2003, which categorizes water quality from

Journal of Theoretical and Applied Information Technology
15th June 2025. Vol.103. No.11

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Class A to D based on different metrics. Additionally, it used slightly different preprocessing, validation, and testing methods. Despite these variations, the study produced good results, with the lowest recorded RMSE of 1.097, further demonstrating the reliability of the dataset for water quality classification [21].



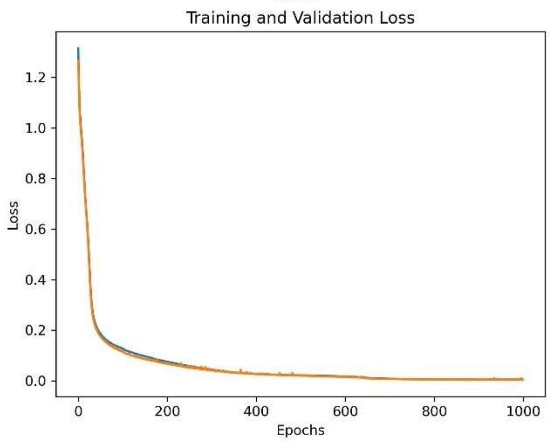


Figure 6: Training and validation accuracy in (a) and loss over epochs in (b) for the most optimal hyperparameters, trained using six features.

We also compared the results with other machine learning methods to evaluate the model's performance. In this research, we used three other traditional machine learning algorithms with various parameters: Support Vector Machine (SVM), Logistic Regression, and k-Nearest Neighbors (KNN). The results show that the deep neural network (DNN) outperforms the other traditional machine learning algorithms. The findings of this test are presented in Table 7. Among these models, DNN achieved the highest accuracy of 0.998402, followed by KNN with 0.997100 and SVM with 0.996200. Logistic regression had the lowest accuracy of 0.978300. The results indicate that while DNN outperforms the traditional machine learning models, KNN and SVM still provide competitive accuracy, making them viable alternatives for classification tasks.

Table 7: Performance Comparison of Deep Neural Network (DNN) and Traditional Machine Learning Algorithms

No	Algorithm	Testing Accuracy
1	Deep Neural Network	0.998402
2	k-Nearest Neighbors	0.997100
3	Support Vector Machine	0.996200
4	Logistic Regression	0.978300

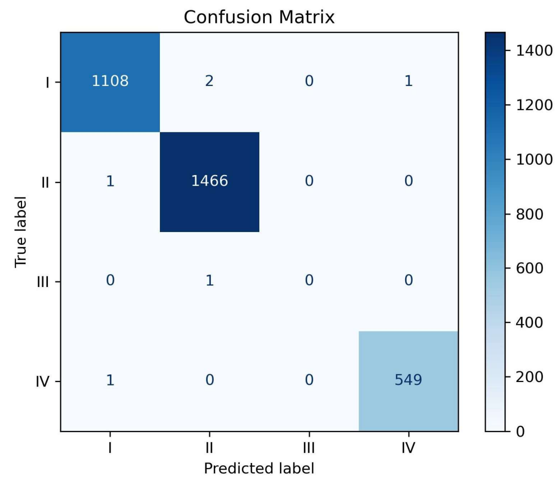


Figure 7: Confusion Matrix of the testing results for the most optimal combination of optimizer and activation function, trained using three features.

4. CONCLUSIONS

This research employs deep learning to categorize water quality in Lake Toba according to Indonesia's regulatory requirements (Government Regulation No. 82 of 2001). These guidelines define particular water quality classes for assessing the quality of Indonesian water. The study proposes an automated approach to precisely evaluate the water quality of Lake Toba utilizing deep learning methodology.

The experimental results demonstrate high accuracy, particularly during the training process. The best accuracy was achieved using a model with the ReLU activation function and the SGD optimizer, configured with five hidden layers and 96 neurons. The findings indicate that the choice of activation function and optimizer significantly impacts both the loss value and accuracy. Additionally, the number of hidden layers and neurons plays a crucial role in model performance.

Increasing the number of hidden layers and neurons can enhance accuracy; however, excessive

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

additions, especially when mismatched with the dataset size, can degrade performance. A larger number of neurons requires more extensive training, and if the dataset is insufficient, many neurons remain undertrained, leading to suboptimal performance. Therefore, careful architectural design is essential to achieve optimal accuracy.

The accuracy of deep learning models is significantly reliant on the quality, quantity and variety of the data utilized. This study's dataset has specific limitations, notably with the volume of data obtained and the imbalance in the class distribution. The model was particularly trained on data from Lake Toba, which is one of the largest lakes in Indonesia. Consequently, its performance may not be applicable to other lakes with varying environmental conditions, temperatures, or pollution sources. Moreover, seasonal climatic fluctuations may impact the dataset and the classification outcomes, thereby compromising the model's reliability over time.

In conclusion, the proposed method demonstrates strong performance in the classification task, particularly in the water quality classification problem examined. With an appropriately designed model architecture, high accuracy can be attained. However, due to the absence of sufficient data for Class III, the model's performance declines when classifying samples labeled as Class III.

ACKNOWLEDGEMENT:

This research is supported by TALENTA Research Grant No: 132/UN5.2.3.1/PPM/KP TALENTA/R/2023 from Universitas Sumatera Utara.

REFERENCES:

[1] C. Zhu and Z. Hao, "Fuzzy Neural Network Model and Its Application in Water Quality Evaluation," in 2009 International Conference on Environmental Science and Information Application Technology, IEEE, Jul. 2009, pp. 251–254. doi: https://doi.org/10.1109/ESIAT.2009.45.

[2] Madan, T., Sagar, S., Tran, T. A., Virmani, D., & Rastogi, D. (2025). Air Quality Prediction using Ensemble Classifiers and Single Decision Tree. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *52*(1), 56–67. https://doi.org/10.37934/araset.52.1.5667.

[3] Y. Wang, J. Zhou, K. Chen, Y. Wang, and L. Liu, "Water quality prediction method based on LSTM neural network," in 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), IEEE, Nov. 2017, pp. 1–5. doi: https://doi.org/10.1109/ISKE.2017.8258814.

[4] Ahmed, J., Wong, L. P., Chua, Y. P., Channa, N., Mahar, R. B., Yasmin, A., Vanderslice, J. A., & Garn, J. v. (2020). Quantitative microbial risk assessment of drinking water quality to predict the risk of waterborne diseases in primary-school children. *International Journal of Environmental Research and Public Health*, *17*(8). https://doi.org/10.3390/ijerph17082774.

[5] Tleuova, Z., Snow, D. D., Mukhamedzhanov, M., & Ermenbay, A. (2023). Relation of Hydrogeology and Contaminant Sources to Drinking Water Quality in Southern Kazakhstan. *Water (Switzerland)*, *15*(24). https://doi.org/10.3390/w15244240.

[6] Shakhman, I., & Bystriantseva, A. (2020). Water Quality Assessment of the Surface Water of the Southern Bug River Basin by Complex Indices. *Journal of Ecological Engineering*, *22*(1), 195–205. https://doi.org/10.12911/22998993/128858.

[7] R. F. Rahmat, Athmanathan, M. F. Syahputra, and M. S. Lydia, "Real time monitoring system for water pollution in Lake Toba," in 2016 International Conference on Informatics and Computing (ICIC), IEEE, 2016, pp. 383–388. doi: https://doi.org/10.1109/IAC.2016.7905749.

[8] Zhu, M., Wang, J., Yang, X., Zhang, Y., Zhang, L., Ren, H., Wu, B., & Ye, L. (2022). A review of the application of machine learning in water quality evaluation. In *Eco-Environment and Health* (Vol. 1, Issue 2, pp. 107–116). Elsevier B.V. https://doi.org/10.1016/j.eehl.2022.06.001.

[9] Almsaadi, J. R. M., Manshor, N., Affendey, L. S., Abdul Halin, A., & Mirza, M. R. (2024). HDN-Net: A Hybrid Deep Neural Network to Improve Iris Recognition in Unconstrained Environments with Eyeglasses. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 238–251. https://doi.org/10.37934/araset.54.2.238251.

[10] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[11] Xue Xicheng and Chen Yan, "Water quality evaluation of surface water based on back propagation neural network," in 2011 International Symposium on Water Resource and Environmental Protection, IEEE, May 2011,

ISSN: **1992-8645** www.jatit.org E-ISSN: **1817-3195**

pp. 524–527. doi: https://doi.org/10.1109/ISWREP.2011.5893059.

[12] L. Jianlin, Z. Guozhen, W. Fuping, W. Bigui, and Y. Hao, "Study and Application of Rainwater Quality Assessment Model Based on Radial Basis Function Artificial Neural Network," Energy Procedia, vol. 11, pp. 4328–4334, Jan. 2011, doi: https://doi.org/10.1016/j.egypro.2011.10.842.

[13] K. N. A. K. Adnan et al., "Water quality classification and monitoring using e-nose and e-tongue in aquaculture farming," in 2014 2nd International Conference on Electronic Design (ICED), IEEE, Aug. 2014, pp. 343–346. doi: https://doi.org/10.1109/ICED.2014.7015827.

[14] Raviteja, N., Saiteja, N., Sreenu, N., Senthilvelan, G., Usha, D., Kumanan, T., Year, F., Tech Of Cse, B., & Professor, A. (2023). Water Quality Classification Using Machine Learning. In *Tuijin Jishu/Journal of Propulsion Technology* (Vol. 44, Issue 6).

[15] Muhammad, S. Y., Makhtar, M., Rozaimee, A., Aziz, A. A., & Jamal, A. A. (2015). Classification model for water quality using machine learning techniques. *International Journal of Software Engineering and Its Applications*, *9*(6), 45–52. https://doi.org/10.14257/ijseia.2015.9.6.05.

[16] Aldhyani, T. H. H., Al-Yaari, M., Alkahtani, H., & Maashi, M. (2020). Water Quality Prediction Using Artificial Intelligence Algorithms. *Applied Bionics and Biomechanics*, *2020*. https://doi.org/10.1155/2020/6659314.

[17] Chen, W., Xu, D., Pan, B., Zhao, Y., & Song, Y. (2024). Machine Learning-Based Water Quality Classification Assessment. *Water (Switzerland)*, *16*(20). https://doi.org/10.3390/w16202951.

[18] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, Jan. 2015, doi: https://doi.org/10.1016/j.neunet.2014.09.003.

[19] M. Negnevitsky, Artificial intelligence: a guide to intelligent systems, Second. Harlow: Pearson Education Limited, 2005.

[20] Asmadi, M. A. D., Zainuddin, S., Mohd Nasir, H., Tengku Wook, T. M. F., Hamzah, N., Abd Rashid, N. E., Abidin, I. Z., & Raja Abdullah, R. S. A. (2024). Potholes Detection in Doppler Radar Signal's Power Spectral Density using Decision Tree. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, *55*(1), 82–93. https://doi.org/10.37934/araset.55.1.8293.

[21] Rahmat, R. F., Pangaribuan, A. B., Suwarno, E., Purnamawati, S., & Lini, T. Z. (2022). LAKE TOBA WATER QUALITY PREDICTION USING EXTREME MACHINE LEARNING. ICIC Express Letters, Part B: Applications, 13(1), 89–97. https://doi.org/10.24507/icicelb.13.01.89.