

Forecasting water quality through machine learning and hyperparameter optimization

Elvin, Antoni Wibowo

Department of Computer Science, Binus Graduate Program-Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia

Article Info

Article history:

Received Aug 18, 2023

Revised Oct 25, 2023

Accepted Nov 1, 2023

Keywords:

Classification report
Hyperparameter tuning
Machine learning
Python
Water quality

ABSTRACT

Forecasting water quality through machine learning and hyperparameter optimization is a research endeavor aimed at enhancing the water quality prediction process. The primary goal of this study is to employ various machine learning algorithms for water quality prediction and to refine existing models from previous research. The paper encompasses a comprehensive literature review of previous water quality prediction studies and introduces novel theoretical insights. The research employs a classic machine learning problem-solving approach, predominantly utilizing the extreme gradient boost (XGBoost) algorithm. Additionally, it evaluates other machine learning algorithms, including the random forest (RF) classifier, decision tree (DT) classifier, adaptive boosting (AdaBoost) classifier, support vector machine (SVM), Naïve Bayes, and extra tree classifier for comparison. The evaluation process utilizes a classification report, providing insights into the precision, recall, f1-score, and accuracy of each machine learning model. Notably, the XGBoost model exhibits superior performance, achieving an impressive 97.06% accuracy. Precision stands at 94.22%, recall at 81.5%, and F1-score at 87.4%. These results represent a significant advancement over prior water quality prediction models, emphasizing the potential of machine learning and hyperparameter optimization to enhance water quality forecasting in environmental monitoring.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Elvin

Department of Computer Science, Binus Graduate Program-Master of Computer Science

Bina Nusantara University

Jakarta, Indonesia

Email: elvin005@binus.ac.id

1. INTRODUCTION

Water is an inorganic, transparent, and colorless chemical substance that is required for the survival of most existing organisms and humans. Adequate water quality is an absolute necessity for the sustenance of all living beings. Aquatic species possess a finite tolerance for pollution, and surpassing these limits imperils their very existence. To maintain a dependable and safe water supply, constant vigilance through water quality monitoring is imperative. With the growth of our economy and the expansion of urban areas, water contamination has surged in significance. The intricate task of predicting factors that influence water quality within hydrophyte systems remains a challenging endeavor. The exploration of diverse methodologies to forecast water quality in reservoirs bears profound implications both in theory and practicality [1], [2]. Water that has poor quality will result in health and safety conditions for living things. Contaminated drinking water not only poses significant health risks but also exerts adverse effects on the environment and infrastructure, with the quality of water being compromised due to a combination of factors such as

inadequate infrastructure, lack of public awareness, and poor hygiene standards. Based on extensive research conducted by the United Nations, it has been revealed that approximately 1.5 million lives are tragically lost each year due to water-borne diseases. This distressing statistic underscores the urgent need for global efforts to ensure access to clean and safe drinking water for all, as well as the importance of sanitation and hygiene practices in preventing such devastating consequences. Addressing this issue remains a critical imperative on the global agenda, as we strive to safeguard the health and well-being of communities around the world. In developing countries, approximately 80% of health issues stem from contaminated water sources, resulting in staggering statistics such as 2.5 billion people being impacted by waterborne diseases annually and a tragic toll of five million deaths, a figure that often eclipses the focus on mortality rates attributed to crimes, accidents, and terrorist attacks [3].

In response to this pressing need, water quality monitoring has taken center stage in the quest to mitigate the consequences of contaminated and impure water sources. The researchers have undertaken the task of examining the chemical composition of these waters, analyzing elements such as aluminum, ammonia, arsenic, barium, radium, and silver. With the results of this diagnosis, researchers can determine whether the water area is fit for consumption by living things or can be harmful to living things. This effort has yielded valuable insights, although the sheer number of water sources under investigation has led to time-consuming research endeavors. Innovation has come to the rescue with the development of machine learning models designed to accelerate the process of water quality assessments. Notably, research conducted by Dalal *et al.* [4] and research conducted by Rustam *et al.* [5] has provided machine learning models with impressive predictive performance, characterized by high accuracy rates.

Motivated by this progress, this research seeks to leverage the realm of water quality and enhance existing machine-learning models. The model was built using the Python programming language and uses a machine learning algorithm as a model to predict whether the water area data entered by the user is suitable for consumption by living things. The primary objectives and contributions of the research can be summarized as follows: i) develop a machine learning model for accurate water quality predictions, ii) employ machine learning to assess water quality, providing insights into its safety for consumption, iii) selection of the most suitable machine learning algorithms for water quality prediction through hyperparameter tuning, and iv) Benchmark machine learning model against existing water quality prediction models from prior research. The selection of the extreme gradient boosting (XGBoost) algorithm as a research model is driven by its efficiency and ease of use. Notably, XGBoost's ability to produce high-performance results and maintain accuracy in training datasets is a defining feature. Moreover, it excels in addressing common issues encountered in machine learning, such as handling missing values, preventing overfitting, and maintaining training performance [6]. Apart from the XGBoost algorithm, there are also other algorithms that can be taken into consideration such as random forest (RF) classifier, decision tree (DT) classifier, adaptive boosting (AdaBoost) classifier, support vector machine (SVM), Naïve Bayes, and extra tree classifier. The algorithm was chosen based on its ability to handle large and complex data, as well as overcome overfitting problems. The combination of various machine learning algorithms can increase the accuracy of water quality predictions by exploiting the strengths of each algorithm.

2. LITERATURE REVIEW

In research on this paper. Previous researchers have developed water quality prediction using machine learning topics by using other machine learning models in several studies. Aldhyani *et al.* [7] discusses research conducted to prevent water pollution. The results show that the SVM achieves the highest accuracy compared to other machine learning algorithms in predicting the dataset model, which is 97.01%. Uddin *et al.* [8] was conducted to optimize models that had been built by previous research and to develop models with machine training (machine learning). The results of the research show that models with XGBoost, extra tree, and DT have high-performance values compared to other machine learning algorithms. Gakii and Jepkoech [9] discusses the development of a classification model in which residents can consume clean quality drinking water. The results showed that the J48 DT model had the highest accuracy of 94%.

Peterson *et al.* [10], discusses modeling the relationship between spectral reflectance and water quality parameters. The results showed that the machine learning regression model achieves good performance values with the help of feature-level fusion, the results of the R2 and root mean squared error (RMSE) models after training of 87% on the total suspended solids (TSS) attribute. Ahmed *et al.* [11] was carried out to reduce cases of people suffering from diseases caused by the consumption of water with unclean quality. The results showed that the method with the machine learning classification algorithm obtained an accuracy value of 85% and will be optimized in the future. Anand *et al.* [12], discusses the selection of the most effective machine learning algorithms for predicting water quality. The final results of the study show that the method with the convolutional neural network (CNN) algorithm achieves an accuracy of 80% and will be optimized in the future. Iyer *et al.* [13] discusses the development of water quality

prediction and modeling methods using artificial intelligence technology, especially with the machine learning approach. The final results of the study show that the model with the RF algorithm achieves a high accuracy value compared to other models. The accuracy is 68% and can be further improved by doing more data training. Vuppalapati *et al.* [14] discusses the proposal and evaluation of alternative approaches based on supervised machine learning in predicting water quality automatically real-time. The final results of the study show that the model with the RF algorithm achieves a high accuracy value compared to other models. The accuracy is 85%.

Dalal *et al.* [4] was conducted to introduce data-driven artificial intelligence techniques from a large number of water samples to develop a new ensemble model of machine learning algorithms to accurately predict water quality, including the application of AdaBoost and its comparison with existing models. The methods used are logistic regression XGBoost, multilayer perceptron, ensemble model, and chi-square automatic interaction detector (CHAID). The final results of the study show that the ensemble learning model has the highest accuracy compared to other models where the model achieves an accuracy value of 96.4%. Rustam *et al.* [5], discusses the development of a simple neural network architecture that is more efficient and accurate and can function to predict water quality and water consumption. The methods used are DT, RF, logistic regression, support vector classifier, extra tree classifier, AdaBoost, CNN, long short-term memory network, gated recurrent unit, and artificial neural network (ANN). The final results of the study show that the ANN model has the highest accuracy compared to other models where the model achieves an accuracy value of 0.96 (96%).

3. METHOD

In this section, we will discuss the research methodology employed in this study. We want to emphasize the critical importance of formulating a robust research procedure to ensure the validity of the evaluation results presented in this paper. The procedure we have developed is presented in the form of a flowchart, which is illustrated in Figure 1 flowchart. By detailing the research methodology and the construction of the flowchart, we aim to provide a clear and comprehensive understanding of the steps and processes involved in our study. This will help ensure the transparency and replicability of our research, a fundamental aspect of any scientific investigation

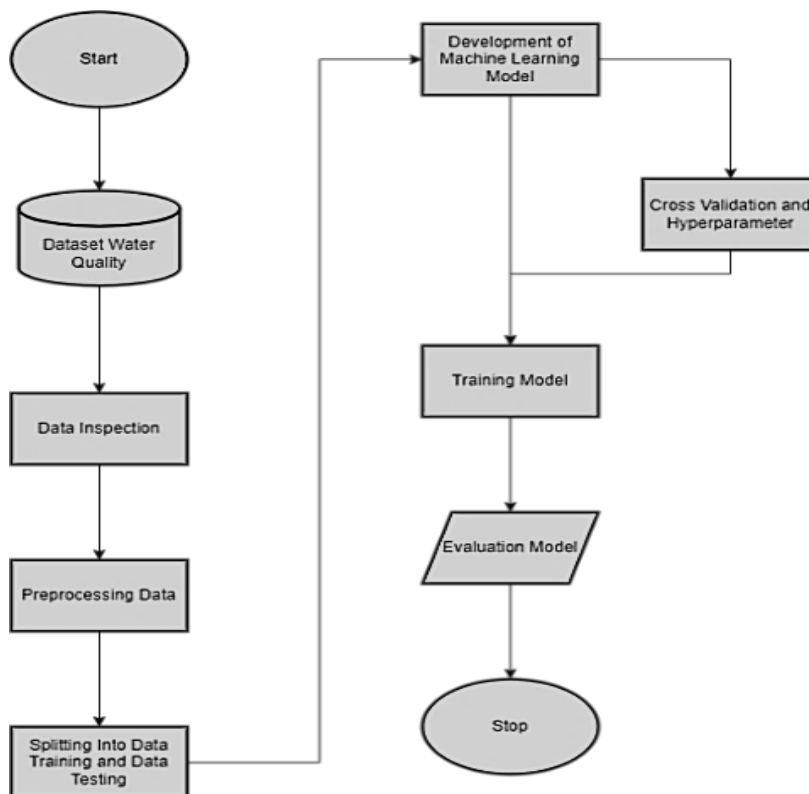


Figure 1. Flowchart

3.1. Dataset water quality

Water is a natural material needed for the life of living things using water as a medium for transporting food substances, as a source of energy for living things, and so on. With this role, water becomes a natural resource that meets the needs of living things that need to be protected so that it is always beneficial for life and the lives of living things on earth. Requirements are made to maintain or achieve water quality standards so that they can be utilized sustainably by the desired water quality level, and conservation and control efforts are made. Water as an environmental component will be influenced by other components. Water that has poor quality will result in health and safety conditions for living things. A decrease in water quality can result in a decrease in the usability, usability, productivity, carrying capacity, and capacity of natural resources. Components of natural resources that are very important must be used optimally for living things [15]. The data used is a water quality dataset obtained from a dataset site called Kaggle the data amounted to 8,000 data and 21 data attributes which were chemical substances contained in the water area. Chemical substances contained in the water area can be seen in Table 1 water quality data description.

Table 1. Data description

Attribute	Information
Aluminium	Dangerous if >2.8
Ammonia	Dangerous if >32.5
Arsenic	Dangerous if >0.01
Barium	Dangerous if >
Cadmium	Dangerous if >0.005
Chloramine	Dangerous if >4
Chromium	Dangerous if >0.1
Copper	Dangerous if >1.3
Flouride	Dangerous if >1.5
Bacteria	Dangerous if >0
Viruses	Dangerous if >0
Lead	Dangerous if >0.015
Nitrates	Dangerous if >10
Nitrites	Dangerous if >1
Mercury	Dangerous if >0.002
Perchlorate	Dangerous if >56
Radium	Dangerous if >5
Selenium	Dangerous if >0.5
Silver	Dangerous if >0.1
Uranium	Dangerous if >0.3
is_safe	Class attribute {0 - not safe, 1 - safe}

3.2. Data inspection

In this phase, an examination of the missing value and examination of the imbalanced data is carried out in the dataset used. The results of missing value examinations in the dataset can be seen by running code that can be seen in Figure 2 code for missing value dataset. With this code, it shows the result of the examination which is the total of each attribute data where there is a missing value. The results of imbalanced data examinations in the dataset can be seen by running code that can be seen in Figure 3 code for imbalanced dataset examination.

```
import pandas as pd
df = pd.read_csv("/content/waterQuality1.csv")
df = df.replace('#NUM!', pd.NA)
df.isnull().sum()
```

Figure 2. Code for missing value dataset

```
[ ] data0['is_safe'].value_counts()
0      7084
1       912
#NUM!     3
Name: is_safe, dtype: int64
```

Figure 3. Code for imbalanced dataset examination

3.3. Preprocessing data

At this stage, data preprocessing is carried out for the missing values in the dataset that has been examined. At the previous stage, it was found that there was a missing value in the ammonia column. Thus, the data preprocessing process is carried out in the “ammonia” column. the data preprocessing method used is changing the missing value to a value of 0. Here is Figure 4 code for preprocessing data.

```
ammo=[]
for i in range(len(df)):
    s=df.loc[i,'ammonia']
    if s=='#NUM!':
        ammo+= [0]
    else:
        ammo+= [float(s)]
df['ammonia']=ammo
```

Figure 4. Code for preprocessing data

3.4. Data train and data test

The next step is to divide the data into 2 types of data, namely data train and data test. Data train is used to train machine learning models while data test is used to test machine learning models that have been trained. the division of data train and data test for research is divided into 80% of the total dataset in the form of data train and 20% of the total dataset in the form of data test. In Figure 3 code for imbalanced dataset examination, it can be seen that the dataset used is imbalanced, where a data balancing process is required. One of the data balancing processes is by smote which by running code that can be seen in Figure 5 code for balancing data.

```
def oversample_data(X_train, y_train):
    smote = SMOTE(sampling_strategy='auto', random_state=42)
    X_train_resampled, y_train_resampled = smote.fit_resample(X_train, y_train)
    return X_train_resampled, y_train_resampled
```

Figure 5. Code for balancing data

3.5. Machine learning models

In this step, several models are used with each different machine learning algorithm. The machine learning algorithm used in the research is XGBoost, RF classifier, DT classifier, AdaBoost classifier, SVM, Naïve Bayes, and extra tree classifier. A brief explanation of each algorithm is as follows:

3.5.1. Decision tree

The DT algorithm is one of the most powerful and widely used algorithms in various fields, namely machine learning, image processing, and pattern identification. The DT is a sequential model that brings together an efficient and cohesive set of basic tests in which the numerical features are compared to the threshold values in each test. Conceptual rules are much easier to construct than numerical weights in connections between nodes in a neural network. DT is also a classification model (classification) that is always used in data mining. Nodes and branches consist of each tree. Each node represents a feature in the category to be classified and each subset determines the value taken by that node. Because DT analysis is simple and rigorous on various forms of data, many research implementations are using the DT algorithm [16].

3.5.2. Random forest

RF is an algorithm for supervised learning that can be used for both classification and regression. RF consists of several DTs. The more DTs you have, the stronger the RF algorithm will be. The RF algorithm uses averages to improve prediction accuracy and control overfitting. Sub-sample size is controlled with max_samples if bootstrap = true (default), otherwise the entire dataset is used to construct each tree [17].

3.5.3. Extreme gradient boosting

Gradient boosting includes DT-based supervised learning that can be used for classification and regression. The gradient boosting algorithm works sequentially adding previous predictors that don't match the predictions to the ensemble, ensuring mistakes previously made are corrected [18]. extreme gradient boosting also known as XGBoost is a machine learning library that is used for prediction and classification, the XGBoost method has the same function as other machine learning methods. XGBoost can be applied to various fields such as education, health, government, and so on. XGBoost in the process requires several parameters including the following [19]:

- `colsample_bytree` is a parameter to select the number of column samples to be used in the program.
- `eta` is a learning rate parameter that functions to prevent overfitting of the model.
- `Gamma` is a parameter to determine the pruning of the nodes in the created tree. The bigger the gamma the more conservative the model is built.
- `max_depth` is a parameter to determine the depth of the tree to be built.
- `min_child_weight` is a parameter to determine the minimum weight limit that a node has.
- `subsample` is a parameter to select the number of sample data rows to be used.
- `objective` is a parameter that serves to determine the purpose of the model built such as regression and also classification.
- `eval_metric` is a parameter to select the evaluation size used. Evaluation measures consist of mean absolute error (MAE), mean squared error (MSE), and RMSE.

3.5.4. AdaBoost classifier

Boosting is a technique in machine learning that enhances prediction accuracy by amalgamating weaker and less accurate rules, includes AdaBoost, also referred to as AdaBoost, as one among various boosting algorithm variants [20]. These boosting algorithms, including AdaBoost, find versatile application across diverse fields, owing to their strong theoretical foundation, precise predictive capabilities, and straight forward implementation involving the following sequential steps:

1. Input: s collection of research samples labeled $\{(x_i, y_i), \dots, (X_n, Y_n)\}$.
2. Initialize: the weight of a training sample $W_{i1} = 1/N$, for all $i=1, \dots, N$.
3. Do for $t=1, \dots, T$
 - a. Use the component learn algorithm to train an h_t classification component, on a training weight sample.
 - b. Calculate his training error at,

$$h_t: \varepsilon_t = \sum_{i=1}^N w_i^t, y_i \neq h_t(x_i) \quad (1)$$

- c. Set the weights for the component classifier to on,

$$h_t == a_t = \frac{1}{2} \ln\left(\frac{1-\varepsilon_t}{\varepsilon_t}\right) \quad (2)$$

- d. Update the training sample weights,

$$w_i^{t+1} = \frac{w_i^t \exp\{-a_t y_t h_t(x_i)\}}{c_t}, i = 1, \dots, N \quad (3)$$

which is a normalized output constant of,

$$f(x) = \text{sign}\left(\sum_{t=1}^T a_t h_t(x)\right) \quad (4)$$

3.5.5. Support vector machine

The SVM is a versatile prediction technique that finds applications in both classification and regression scenarios. At its core, SVM relies on the fundamental concept of a linear classifier, enabling it to separate data in a linear fashion. However, its versatility extends to tackling non-linear problems through the strategic utilization of kernel functions in high-dimensional spaces. These kernel functions play a pivotal role by transforming the initial dimensions of a dataset from lower dimensions to relatively higher dimensions, effectively creating a more complex feature space. By doing so, the SVM can identify an optimal hyperplane as a separator, a crucial step in maximizing the inter-class distance in the input space [21]. This ability to adapt to both linear and non-linear data distributions makes SVM a powerful tool in various machine learning and data analysis tasks.

3.5.6. Naïve Bayes

Naïve Bayes is a widely used machine learning algorithm, particularly in the realm of text classification. This algorithm leverages probabilistic calculations and is rooted in the foundational work of the British scientist Thomas Bayes. Bayes' theorem, the cornerstone of this algorithm, allows for the estimation of future probabilities by analyzing past experiences. In practice, the Bayes optimal classifier applies this theorem to calculate the probabilities of class membership for each attribute within a given dataset, leading to the determination of the most optimal class [22]. This approach has found extensive applications in natural language processing, spam detection, sentiment analysis, and many other areas where text classification is essential. Its simplicity and effectiveness make Naïve Bayes a valuable tool for a wide range of classification tasks.

3.5.7. Extremely randomized trees

The extra trees classifier, also known as extremely randomized trees, falls within the ensemble DT learning category and distinguishes itself by constructing a forest of unpruned DT. In this approach, the selection of attributes and splitting points during node separation is deliberately randomized, producing a set of uncorrelated DTs. While similar in concept to the RF classifier, which also forms an ensemble of DTs, extra trees' key distinctions lie in its use of the original training data for each tree and its unique feature selection process that involves choosing the best features to split based on criteria like gini index, entropy, or information gain, achieved by randomly sampling k features from the feature set for each test node [23].

3.6. Cross validation and hyperparameter

During this stage, the model development process unfolds, incorporating cross-validation (CV) and the fine-tuning of hyperparameters, which are predetermined parameters set prior to commencing the learning process. In contrast, model parameters, encompassing the algorithm-derived weights and coefficients, are derived through the training data. Each algorithm features a distinct set of hyperparameters; for example, a DT's depth parameter. CV emerges as a pivotal statistical technique utilized to gauge the accuracy of machine learning models. Given the uncertainty surrounding how well a trained model will generalize to previously unseen data, CV aids in assessing its predictive performance. By evaluating the model's performance on new and unseen data, it becomes feasible to gauge its complexity, the risk of overfitting, or its capacity for generalization. In scenarios where data is limited, CV proves invaluable in testing the effectiveness of a machine-learning model, as is the case in this research endeavor [24]. The objective of this step is to fortify machine learning models, ensuring that the accuracy achieved during testing is highly dependable. To streamline the process of configuring CV and hyperparameters, the GridSearchCV library in Python is employed. GridSearchCV plays a vital role in the hyperparameter optimization process for each model, facilitating the attainment of optimal performance by fine-tuning the parameters, as illustrated in Table 2 parameters using GridSearchCV.

Table 2. Parameters using GridSearchCV

Algorithms	Parameters
XGBoost	{'colsample_bytree': 0.8, 'gamma': 0.2, 'learning_rate': 0.1, 'max_depth': 5, 'min_child_weight': 3}
RF	{'max_depth': none, 'min_samples_split': 5, 'n_estimators': 50}
DT	{'max_depth': none, 'max_features': none, 'min_samples_leaf': 4, 'min_samples_split': 10} accuracy: 0.965
AdaBoost	{'learning_rate': 1.0, 'n_estimators': 100}
SVM	{'C': 10, 'gamma': 0.1, 'kernel': 'linear'}
Naïve Bayess	{'var_smoothing': 0.0001}
Extra tree	{'max_depth': 30, 'min_samples_leaf': 1, 'min_samples_split': 6, 'n_estimators': 300}

3.7. Evaluation models

In this stage, the process of testing the models using test data is carried out, and the results of evaluating the models trained are recorded in one of the machine learning evaluation methods, namely the classification report. A classification report is a method for evaluating machine learning classification models (machine learning). The classification report provides a summary of model performance with various metrics such as precision, recall, F1-score, and accuracy. Precision is used to calculate how much the ratio/accuracy of the predictions made by the model is correct, recall is the classifier's ability to find all positive cases, and F1-score is the harmonic average of weighted precision and recall. Score has a lower accuracy percentage than accuracy because it embeds precision and recall into its calculations, and accuracy is used to calculate the ratio of correct predictions, both positive and negative, with all existing data. Precision, recall, F1-score, and accuracy calculations can be calculated with the following formulation [25].

$$Precision = \left(\frac{TP}{TP+FP} \right) * 100\% \tag{5}$$

$$Recall = \left(\frac{TP}{TP+FN} \right) * 100\% \tag{6}$$

$$F1 - Score = 2 * \frac{Recall * Precision}{Recall + Precision} \tag{7}$$

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} * 100\% \tag{8}$$

4. RESULT AND DISCUSSION

Based on the results of the research conducted, it can be seen that the use of the XGBoost algorithm model produces a more accurate prediction of water quality. This can be seen from the developed model having a higher accuracy value compared to the other model with different algorithms. Implementation results can be measured from the results of water quality predictions generated by machine learning algorithms (machine learning). The test is carried out using test data that is separate from the training data. The predicted results are then compared with the actual values to determine how accurate the resulting predictions are. This implementation uses several machine learning algorithms so that comparisons are made to determine whether the XGBoost algorithm model outperforms other machine learning algorithm models. In this test, the evaluation results were obtained which were a classification report on the machine learning algorithm model, along with a graph of the evaluation results of the machine learning algorithm model. Plot graph evaluation of each model without hyperparameter can be seen in Figure 6 Evaluation results without hyperparameter. The evaluation results shown in the Figures 6 summarized in a table called the classification report. The evaluation results of the classification report on machine learning models used to predict water quality can be seen in Table 3 evaluation results of the classification report without hyperparameter.

The evaluation results shown in the Table 3 show that the XGBoost algorithm achieves the best evaluation in terms of accuracy, precision, recall and F1-Score. However, it has not achieved the desired results. Then a cross validation and hyperparameter process was carried out to strengthen the accuracy of the model. The following is a graph of the results of evaluating machine learning algorithm models using hyperparameters which can be seen in Figure 7. Evaluation results with hyperparameter. The evaluation results shown in the Figure 7 summarized in a table called the classification report. The evaluation results of the classification report on machine learning models used to predict water quality can be seen in Table 4 evaluation results of the classification report with hyperparameter.



Figure 6. Evaluation results without hyperparameter

Table 3. Evaluation results of the classification report without hyperparameter

Methods	Accuracy	Precision	Recall	F1-score
XGBoost	96.93%	92.65%	82%	87%
RF classifier	96%	97%	69%	80%
DT classifier	95%	83%	84%	84%
AdaBoost classifier	93%	86%	54%	66%
SVM	78%	31%	76%	44%
Naïve Bayess	88%	47%	39%	43%
Extra tree classifier	93%	88%	64%	61%

Performance Metrics for Different Models

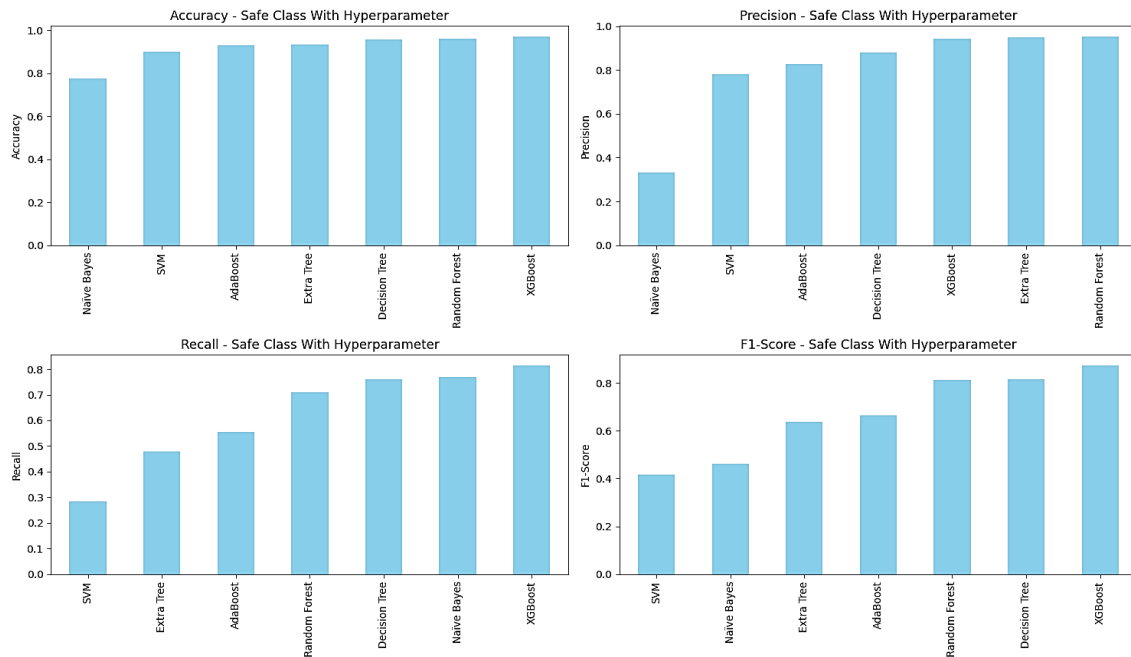


Figure 7. Evaluation results with hyperparameter

Table 4. Evaluation results of the classification report with hyperparameter

Methods	Accuracy	Precision	Recall	F1-score
XGBoost	97.06%	96.42%	81%	88%
RF classifier	95.94%	96%	70%	81%
DT classifier	95.69%	86.73%	85%	86%
AdaBoost classifier	93.00%	84%	57%	68%
SVM	90.06%	79%	34%	47%
Naive Bayess	77.63%	23%	65%	34%
Extra tree classifier	93.19%	97%	49%	66%

The evaluation results shown in the Table 4 show that the XGBoost algorithm achieves the best evaluation in terms of accuracy, precision, recall, and F1-score. One of the reasons why XGBoost achieves excellent evaluation results is because of its superiority in overfitting and the bias-variance tradeoff. XGBoost is an ensemble learning technique that combines many small DTs sequentially. By boosting, the model will focus on data that was previously poorly predicted, thereby reducing errors in more complex predictions. Even though XGBoost shows the best evaluation results in the Table 4, the performance of an algorithm is also very dependent on the characteristics and data structure used. To show the significance of the proposed model, a performance comparison was carried out in this study. In this regard, several recent studies related to the current problem were selected. The study [4] used an ensemble model for water prediction, while the study [5] used an ANN for water prediction. Similarly, models from previous studies regarding water quality prediction were implemented on the current dataset. Comparison results can be seen in Table 5 comparison machine learning model with previous research.

Table 5. Comparison machine learning model with previous research

Research	Machine learning algorithms	Accuracy
Dalal <i>et al.</i> [4]	Ensemble model	96.4%
Rustam <i>et al.</i> [5]	ANN	96%
This research	XGBoost	97.06%

5. CONCLUSION AND FUTURE WORKS

The research on hyperparameter optimization with machine learning algorithms for water quality prediction succeeded in developing a model that can be used to predict water quality and whether these waters are fit for consumption. The study has yielded promising results with an impressive accuracy rate of 97.06%. The study shows that XGBoost outperforms several other machine learning algorithms, highlighting its potential as a powerful tool for environmental monitoring and water quality management. The findings of this study indicate that XGBoost is a highly effective method for water quality prediction, surpassing the performance of other machine learning approaches. The achieved accuracy of 97.06% is satisfactory and outstanding, demonstrating XGBoost's potential in environmental monitoring and water quality management.

However, there are opportunities for further improvement and future research in this area. The following points highlight potential future works such as follows: i) to enhance the performance of the XGBoost model, researchers can explore hyperparameter tuning techniques such as grid search and Bayesian optimization. Fine-tuning the model's hyperparameters can lead to better generalization and robustness, resulting in improved predictive accuracy, ii) conducting a comparative analysis with other state-of-the-art machine learning algorithms can provide valuable insights into the strengths and weaknesses of various models. This analysis will aid in selecting the most suitable algorithm for water quality prediction tasks in different scenarios, and iii) increasing the dataset's size and diversity through data augmentation techniques can improve the model's ability to handle different water quality scenarios and enhance its predictive accuracy.

ACKNOWLEDGEMENTS

Author wants to thank his colleagues at Bina Nusantara University for their helpful feedback and support. In particular, Author would like to thank Professor Amalia Zahra for their valuable knowledges to Author research. Finally, author would like to thank his family and friends for their love and support throughout the research process. Without their encouragement and support, we would not have been able to complete this research.




REFERENCES

- [1] M. M. Hassan *et al.*, "Efficient prediction of water quality index (WQI) using machine learning algorithms," *Human-Centric Intelligent Systems*, vol. 1, no. 3–4, p. 86, 2021, doi: 10.2991/hcis.k.211203.001.
- [2] H. A. N. Silva, A. Rosato, R. Altilio, and M. Panella, "Water quality prediction based on wavelet neural networks and remote sensing," in *Proceedings of the International Joint Conference on Neural Networks*, Jul. 2018, vol. 2018-July, pp. 1–6, doi: 10.1109/IJCNN.2018.8489662.
- [3] J. P. Nair and M. S. Vijaya, "River water quality prediction and index classification using machine learning," *Journal of Physics: Conference Series*, vol. 2325, no. 1, p. 12011, Aug. 2022, doi: 10.1088/1742-6596/2325/1/012011.
- [4] S. Dalal *et al.*, "Machine learning-based forecasting of potability of drinking water through adaptive boosting model," *Open Chemistry*, vol. 20, no. 1, pp. 816–828, Jan. 2022, doi: 10.1515/chem-2022-0187.
- [5] F. Rustam *et al.*, "An artificial neural network model for water quality and water consumption prediction," *Water (Switzerland)*, vol. 14, no. 21, p. 3359, Oct. 2022, doi: 10.3390/w14213359.
- [6] Z. A. Ali, Z. H. Abduljabbar, H. A. Taher, A. B. Sallow, and S. M. Almufti, "Exploring the power of eXtreme gradient boosting algorithm in machine learning: a review," *Academic Journal of Nawroz University*, vol. 12, no. 2, pp. 320–334, 2023.
- [7] T. H. H. Aldhyani, M. Al-Yaari, H. Alkahtani, and M. Maashi, "Water quality prediction using artificial intelligence algorithms," *Applied Bionics and Biomechanics*, vol. 2020, pp. 1–12, Dec. 2020, doi: 10.1155/2020/6659314.
- [8] M. G. Uddin, S. Nash, M. T. M. Diganta, A. Rahman, and A. I. Olbert, "Robust machine learning algorithms for predicting coastal water quality index," *Journal of Environmental Management*, vol. 321, p. 115923, Nov. 2022, doi: 10.1016/j.jenvman.2022.115923.
- [9] C. Gakii and J. Jepakoch, "A classification model for water quality analysis using decision tree," *European Journal of Computer Science and Information Technology*, vol. 7, no. 3, pp. 1–8, 2019, doi: 10.37745/ejcsit.2013.
- [10] K. T. Peterson, V. Sagan, P. Sidike, E. A. Hasenmueller, J. J. Sloan, and J. H. Knouft, "Machine learning-based ensemble prediction of water-quality variables using feature-level and decision-level fusion with proximal remote sensing," *Photogrammetric Engineering and Remote Sensing*, vol. 85, no. 4, pp. 269–280, Apr. 2019, doi: 10.14358/PERS.85.4.269.
- [11] U. Ahmed, R. Mumtaz, H. Anwar, A. A. Shah, R. Irfan, and J. García-Nieto, "Efficient water quality prediction using supervised machine learning," *Water (Switzerland)*, vol. 11, no. 11, p. 2210, Oct. 2019, doi: 10.3390/w11112210.
- [12] M. V. Anand, C. Sohitha, G. N. Saraswathi, and G. V. Lavanya, "Water quality prediction using CNN," *Journal of Physics: Conference Series*, vol. 2484, no. 1, p. 12051, May 2023, doi: 10.1088/1742-6596/2484/1/012051.
- [13] S. Iyer, S. Kaushik, and P. Nandal, "Water quality prediction using machine learning," *MR International Journal of Engineering and Technology*, vol. 10, no. 1, pp. 59–62, May 2023, doi: 10.58864/mrijet.2023.10.1.8.




- [14] K. T. Vuppapapati, C. S. D. Sai, A. P. Ragav, P. Babji, and P. Padmaja, "Water quality prediction using machine learning algorithms," *Journal of Emerging Technologies and Innovative Research*, vol. 10, no. 4, pp. c711–c721, 2023.
- [15] M. Faisal and D. M. Atmaja, "Water quality at the water source in Pura Taman Desa Sanggalangit as a source of drinking water based on the Storet method," *Journal of Geography Education Undiksha*, vol. 7, no. 2, pp. 74–84, Aug. 2019, doi: 10.23887/jjg.v7i2.20691.
- [16] B. Charbuty and A. Abdulazeez, "Classification based on decision tree algorithm for machine learning," *Journal of Applied Science and Technology Trends*, vol. 2, no. 01, pp. 20–28, Mar. 2021, doi: 10.38094/jastt20165.
- [17] L. Vergni and F. Todisco, "A random forest machine learning approach for the identification and quantification of erosive events," *Water (Switzerland)*, vol. 15, no. 12, p. 2225, Jun. 2023, doi: 10.3390/w15122225.
- [18] Z. Li, S. Peng, G. Zheng, X. Chu, and Y. Tian, "Prediction of daily water consumption in residential areas based on meteorologic conditions—applying gradient boosting regression tree algorithm," *Water (Switzerland)*, vol. 15, no. 19, p. 3455, Sep. 2023, doi: 10.3390/w15193455.
- [19] O. Alshboul, A. Shehadeh, G. Almasabha, and A. S. Almuflih, "Extreme gradient boosting-based machine learning approach for green building cost prediction," *Sustainability (Switzerland)*, vol. 14, no. 11, p. 6651, May 2022, doi: 10.3390/su14116651.
- [20] Y. Ding, H. Zhu, R. Chen, and R. Li, "An efficient Adaboost algorithm with the multiple thresholds classification," *Applied Sciences (Switzerland)*, vol. 12, no. 12, p. 5872, Jun. 2022, doi: 10.3390/app12125872.
- [21] D. I. Sumantiawan, J. E. Suseno, and W. A. Syaifei, "Sentiment analysis of customer reviews using support vector machine and Smote-Tomek links for identify customer satisfaction," *Jurnal Sistem Informasi Bisnis (JSINBIS)*, vol. 13, no. 1, pp. 1–9, Jan. 2023, doi: 10.21456/vol13iss1pp1-9.
- [22] N. Normah, "Naïve bayes algorithm for sentiment analysis windows phone store application reviews," *Sinkron: jurnal dan penelitian teknik informatika*, vol. 3, no. 2, p. 13, Mar. 2019, doi: 10.33395/sinkron.v3i2.242.
- [23] T. E. Mathew, "An optimized extremely randomized tree model for breast cancer classification," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 16, pp. 5234–5246, 2022.
- [24] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for Arabic sentiment analysis," *Informatics*, vol. 8, no. 4, p. 79, Nov. 2021, doi: 10.3390/informatics8040079.
- [25] N. Al Mudawi and A. Alazeb, "A model for predicting cervical cancer using machine learning algorithms," *Sensors*, vol. 22, no. 11, p. 4132, May 2022, doi: 10.3390/s22114132.

BIOGRAPHIES OF AUTHORS



Elvin    was born in Batam, Indonesia on October 31, 2000. He has received his first degree of Information Technology with Computer Science from University Tarumanagara on 2022. He is currently pursuing a master degree in Computer Science from Bina Nusantara University since 2022. He is currently working as IT consultant backend at PT. Wahana Cipta Sinatria, Jakarta. He can be contacted at email: elvinsiau7@gmail.com.



Antoni Wibowo    is a member (M) of IAENG since 2012. He has received my first degree of Applied Mathematics in 1995 and master degree of Computer Science in 2000. In 2003, He awarded a Japanese Government Scholarship (Monbukagakusho) to attend Master and Ph.D. programs at Systems and Information Engineering in University of Tsukuba-Japan. He completed the second master degree in 2006 and Ph.D. degree in 2009, respectively. His Ph.D. research focused on machine learning, operations research, multivariate statistical analysis and mathematical programming, especially in developing nonlinear robust regressions using statistical learning theory. He has worked from 1997 to 2010 as a researcher in the Agency for the Assessment and Application of Technology-Indonesia. From April 2010-September 2014, he worked as a senior lecturer in the Department of Computer Science-Faculty of Computing, and a researcher in the Operation Business Intelligence (OBI) Research Group, Universiti Teknologi Malaysia (UTM)-Malaysia. From October 2014-October 2016, he was an associate professor at Department of Decision Sciences, School of Quantitative Sciences in Universiti Utara Malaysia (UUM). He is currently working at Binus Graduate Program (Master in Computer Science) in Bina Nusantara University-Indonesia as a specialist lecturer and continues his research activities in machine learning, optimization, operations research, multivariate data analysis, data mining, computational intelligence and artificial intelligence. He can be contacted at email: anwibowo@binus.edu.