


Drinking water potability prediction using machine learning approaches: a case study of Indian rivers

Bharati Ainapure^a, Nidhi Baheti^a, Jyot Buch^a, Bhargav Appasani ^{b,*}, Amitkumar V. Jha^c and Avireni Srinivasulu^d

^a Department of Computer Engineering, Faculty of Science and Technology, Vishwakarma University, Pune, Maharashtra 411056, India

^b KIITS University: Kalinga Institute of Industrial Technology, India

^c School of Electronics Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar 751024, India

^d School of Engineering and Technology, Mohan Babu University, Tirupati 517102, India

*Corresponding author. E-mail: bhargav.appasanifet@kiit.ac.in

 BA, 0000-0002-0878-7405

ABSTRACT

Drinking water is the most precious resource on Earth. In the past few decades, the quality of drinking water has significantly degraded due to pollution. Water quality assessment is paramount for the well-being of the people since the presence of pollutants can have serious health issues. Particularly, in developing countries such as India, water is not properly assessed for its quality. This work uses machine learning techniques to predict the water quality of Indian rivers. It focuses on finding water potability when provided with the key factors used to calculate the water quality index for the water sample. Important parameters like water temperature, pH value, electrical conductivity, dissolved oxygen, fecal coliform, total coliform counts, and biochemical oxygen demand are used to calculate the water quality index. The approaches that are explored include the use of *K*-nearest neighbor (KNN), Random Forest, and XGBoost, with and without hyperparameter tuning, and the use of a sequential artificial neural network to see which of the three models helps us to give the most accurate predictions for the potability of water. XGBoost was the most efficient model, with an accuracy of 98.93%.

Key words: KNN, machine learning, neural networks, predictive analysis, water quality index, XGBoost

HIGHLIGHTS

- The extreme gradient boosting classifier (XGBoost) model has been used for assessing the water potability of Indian rivers.
- Comparison is shown with some existing works and the results are much better.
- Case studies are applied on Indian rivers.
- Four different models are used: *K*-nearest neighbor, XGBoost, Random Forest, and artificial neural networks.

1. INTRODUCTION

One of the most critical aspects of a healthy environment is water quality (WQ). Clean water is essential for the survival of a wide range of plants and animals. Though it may appear unrelated at first, our land-based activities impact WQ. Pollutants, excess nutrients from fertilizers, and silt are commonly transferred into local lakes and rivers by runoff from cities and agricultural fields (Water Quality 2022a). Since rivers are more accessible than other water sources, they have become a better-suited option for the growth of the human society (Motagh *et al.* 2017). Due to these reasons and the fact that the ocean and groundwater come with their own set of hindrances. Ocean water is salty, is unsuitable for drinking, and difficult to transport, and groundwater in many places has a slow recharge rate. Hence, the utilization of rivers has received a lot more attention. There has been an eminent amount of research in this area, and a new field of engineering called river engineering has been proposed, which contains studies on morphological changes, WQ, sediment movement, and transmission of pollutants (Julien 1992). In earth sciences, the investigation of river WQ is a popular subject (Kashefipour 2002). Detailed quality criteria are used to assess the quality of most ambient water bodies, such as rivers, lakes, and streams. Water specifications for various applications/uses also have their standards. Irrigation water, for example, must not be overly saline or contain hazardous elements that can be passed to plants or

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (<http://creativecommons.org/licenses/by/4.0/>).

soil, harming ecosystems. WQ for industrial applications necessitates a variety of qualities depending on unique industrial activities. Natural water resources, such as ground and surface water, are some of the cheapest freshwater sources. Human/industrial activity and other natural processes can pollute such resources (Aldhyani *et al.* 2020).

According to the 2022 World Health Organization (WHO) report, over 2 billion people inhabit regions facing severe water stress, a condition expected to worsen in certain areas due to the effects of climate change and population growth. While the public expresses concerns about emerging pollutants like pharmaceuticals, pesticides, per- and polyfluoroalkyl substances, and microplastics, it is noteworthy that the most significant chemical risks in drinking water still arise from substances like arsenic, fluoride, and nitrate. The contamination of water sources by microorganisms remains a pressing issue, leading to the spread of diseases such as cholera, dysentery, typhoid, and polio, causing an estimated number of 485,000 diarrheal-related deaths annually (Drinking-water 2022).

This has a higher fatality rate than those brought on by crimes, accidents, and terrorist acts (Clean Water for a Healthy World). As a result, it is critical to provide innovative ways for analyzing and, if possible, forecasting WQ. The WQ environment has been negatively impacted by the rapid increase in population, the industrial revolution, and the widespread use of pesticides and fertilizers (Cabral Pinto *et al.* 2019). As a result, having models for predicting the WQ is extremely useful for monitoring water contamination.

This paper is dedicated to the implementation of regression algorithms and artificial neural networks (ANNs) for the purpose of detecting the water quality index (WQI). This, in turn, enables us to determine whether water is suitable for consumption. Initially, a regression method is employed to assess water potability, followed by a classification process based on this metric. This study utilizes data from the Indian government website, considering the parameters typically used to ascertain WQ. The significance of this endeavor lies in its potential to identify the potability of water based on specific parameters. This, in turn, aids in addressing and eliminating the factors responsible for rendering water unfit for drinking.

2. RELATED WORK

In 2012, Shah analyzed around 20 locations in Kerala (India). The study discovered that the water required some treatment to be suitable for marine life but was appropriate for agriculture (Al-ani 2019). Adeyemi *et al.* evaluated the chemical effects of leachate on the consistency of the surrounding water in Nigeria. According to the study, leachate-affected water samples had greater biochemical oxygen demand (BOD) and chemical oxygen demand (COD) levels in the dry season compared to the rainy season. Bacteria were more prevalent during the wet season than during the dry season. As a result, the findings of this study revealed that using leachate-contaminated water is hazardous and should be avoided (Sagar *et al.* 2015).

According to findings from the WQI (Hassan 2019), the problem of water contamination is on the rise. In response to this concern, a study conducted by Batur and Maktav (Liu *et al.* 2020) applied the principal component analysis (PCA) method to predict the WQ in Lake Gala, located in Turkey. What sets this research apart is the incorporation of satellite imagery, which offers a unique and innovative approach to assessing and monitoring WQ in the region. By utilizing PCA and satellite data, the study aimed to provide a more comprehensive understanding of the dynamics and trends in WQ, contributing to better-informed environmental management strategies.

Shailesh Jaloree & Rajput (2014) attempted to evaluate the WQ of the Narmada River using five WQ markers and used a decision tree approach. Another study advised developing a precise WQ forecasting system for smart agriculture utilizing the deep Bidirectional Stacked Simple Recurrent Unit (Bi-S-SRU) (Liu *et al.* 2020). Yan-jun & Qian (2012) used a support vector machine-based clustering model. It was discovered that this model is susceptible to vacancies.

Solanki *et al.* (2015) used a deep learning network model to assess and forecast the chemical components of water, particularly dissolved oxygen (DO) and pH value, which was shown to produce more accurate findings than supervised learning-based techniques. Khan and See used an ANN to construct a WQ model that incorporated the values of chlorophyll, DO, turbidity, and conductivity (Khan & See 2016).

Ali employed three evaluation strategies or assessment processes to measure the model's effects. The initial evaluation method was based on the segmentation of neural network connection weights, which determined the relevance of each network input parameter. On the other hand, the second and third assessment processes determined the most effective input with the greatest potential (Ahmed *et al.* 2019).

The method used by [Sagan *et al.* \(2020\)](#) combined WQ data with hyperspectral imaging, satellite data derived from laboratory analysis of grab samples, *in situ* real-time monitoring sensors installed in several water bodies across the Midwest, and grab sample analysis. The results showed that satellite-based and proximal sensors could give accurate estimates of optically active parameters, but the indirect estimation of non-optically active components is still difficult.

[Dogo *et al.* \(2019\)](#) concluded that despite the advantages of the extreme learning machine (ELM), its use is hardly ever investigated in this domain. According to the study, a hybrid DL-ELM framework was also put out as a workable choice that may be investigated further and used to find anomalies in WQ data. In a subsequent approach, [Bucak & Karlik \(2011\)](#) describe the design, implementation, and performance evaluations of an application developed for the real-time detection of drinking WQ using cerebellar model articulation controller (CMAC) ANNs. The CMAC ANN algorithm has a substantially faster learning rate than the MLP backpropagation (BP) approach.

[Patel *et al.* \(2022\)](#) used the Synthetic Minority Oversampling Technique (SMOTE) to balance the dataset as it was unbalanced. They proposed Random Forest (RF) and Gradient Boost methods achieving an accuracy of only 81%. In [Aldhyani *et al.* \(2020\)](#), long short-term memory (LSTM) and SVM techniques are used to predict the WQ, and the accuracy achieved was 96%. A 10k-fold cross-validation technique was performed, which resulted in an R^2 score of 0.99 with a possibility of ± 0.08 deviation. In the study conducted by [Al Duhayyim *et al.* \(2022\)](#), a fuzzy deep neural network has been proposed for WQ prediction along with the atom search optimization technique. They used the F-DCN model followed by hyperparameter tuning, and the achieved accuracy was 98.1%. [Rustam *et al.* \(2022\)](#) proposed two models, namely, nonlinear autoregressive neural network (NARNET) and LSTM. They were able to achieve an accuracy of 96%.

In contrast to the previous study, our research focuses on the utilization of the XGBoost classifier for assessing water potability in Indian rivers. We conducted a comprehensive comparison of various classifiers, including ANNs. Notably, our findings demonstrate that the machine learning approach with the XGBoost classifier outperforms the ANN method. This disparity in performance underscores the superior accuracy and effectiveness of XGBoost in the context of WQ assessment for Indian rivers, as opposed to the approach employed in the earlier study.

3. PROPOSED METHODOLOGY

We have proposed two approaches below. One uses basic machine learning approaches like *K*-nearest neighbor (KNN), RF, and XGBoost. The other one is based on the application of ANNs as shown in [Figure 1](#). The data were preprocessed and additional parameters were calculated, which helped us to predict water potability. Numerous machine learning techniques were tried out, and XGBoost was found to be the most efficient. In neural networks, we developed a sequential model with layers best suited for our data and tuned the parameters accordingly.

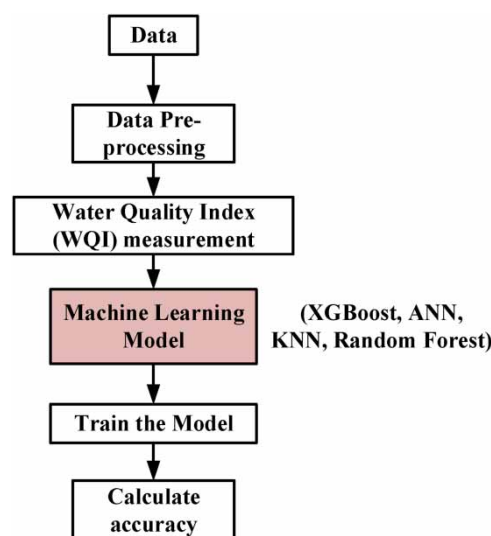


Figure 1 | Proposed approach to predict the WQ.

3.1. Dataset description

The dataset chosen contains the values of various parameters of WQ in different regions of India collected over the years 2003–2014. The sources of these data are various Indian Government websites. Twelve different parameters and their descriptions are shown in Table 1.

Table 2 shows a few rows of the dataset, including parameters used to calculate the WQI. The dataset was taken from the Indian Government website (Water Quality 2022b). Important parameters like water temperature (Temp), pH value, electrical conductivity (EC), DO, fecal coliform and total coliform counts, and BOD are used to calculate the WQI. WQ will be defined as satisfactory or unsatisfactory based on the measured ambient concentrations and corresponding criteria.

3.2. Preprocessing of the data

First, we perform some explanatory analyses on the dataset to see the values we deal. On displaying the information about our dataset, it is found that 11 out of 12 columns are of the object data type, as shown in Figure 2. As we need numeric values of the parameters to perform mathematical calculations of the WQI, we convert all the required calculations to the numeric type as shown in Figure 3.

Next, we initialize six new columns *npH*, *ndo*, *nco*, *nbdo*, *nec*, and *nnn* in our data frame, which are calculated on the basis of DO, pH, EC, BOD, nitrate, fecal coliform, and total coliform. The null values in the dataset are replaced by their standard means and modes depending on the type of value that needs to be replaced.

Table 1 | Description of the dataset

Parameters	Description
Station code	It contains the station code for all the listed places.
Location	Address of the place where the data are collected from.
State	Name of the state from which the data are collected.
Temp	Average temperature of that location over time.
DO	Contains the amount of DO present in the water over time. The optimal value for DO is 10 mg/L.
pH	It is the concentration of hydrogen ions in water over time. The optimal value for pH is 8.5.
Conductivity	It is the measure of conductivity of water over time. The optimal value for conductivity is 1,000 μ S/cm.
BOD	It is the measure of BOD of water over time. The optimal value for BOD is 5 mg/L.
Nitrate	Nitrate content of water over time. The optimal value for nitrate is 45 mg/L.
Fecal coliform	Fecal coliform bacteria are a group of bacteria that are passed through the fecal excrement of humans, livestock, and wildlife. It measures the value of fecal coliform in water. The optimal value is 100 per 100 mL.
Total coliform	It is the total measure of coliform in water. The optimal value is 100 per 100 mL.
Year	The year during which the observations were recorded.

Table 2 | WQI parameters

Sr. No	Station code	Locations	State	Temp	DO	pH	EC	BOD	N-NO3	Fecal coliform	Total coliform
1	1,393	DAMANGANGA AT D/S OF MADHUBAN	Daman and Diu	30.6	6.7	7.5	203	NAN	0.1	11	27
2	1,399	ZUARI AT D/S OF PT. WHERE KUMBARJRIA CANAL	Goa	29.8	5.7	7.2	189	2	0.2	4,953	8,391
3	1,475	ZUARI AT PANCHAWADI	Goa	29.5	6.3	6.9	179	1.7	0.1	3,243	5,330
4	3,181	RIVER ZUARI AT BORIM	Goa	29.7	5.8	6.9	64	3.8	0.5	5,382	8,443
5	3,182	RIVER ZUARI AT MARCAIM JETTY	Goa	29.5	5.8	7.3	63	1.9	0.4	3,428	5,500

#	Column	Non-Null Count	Dtype
0	STATION CODE	1991 non-null	object
1	LOCATIONS	1991 non-null	object
2	STATE	1991 non-null	object
3	Temp	1991 non-null	object
4	D.O. (mg/l)	1991 non-null	object
5	PH	1991 non-null	object
6	CONDUCTIVITY (μmhos/cm)	1991 non-null	object
7	B.O.D. (mg/l)	1991 non-null	object
8	NITRATENAN N+ NITRITENANN (mg/l)	1991 non-null	object
9	FECAL COLIFORM (MPN/100ml)	1991 non-null	object
10	TOTAL COLIFORM (MPN/100ml)Mean	1991 non-null	object
11	year	1991 non-null	int64

dtypes: int64(1), object(11)
memory usage: 186.8+ KB

Figure 2 | Object data type.

```
Out[4]: STATION CODE      object
LOCATIONS      object
STATE          object
Temp           float64
D.O. (mg/l)    float64
PH             float64
CONDUCTIVITY (μmhos/cm) float64
B.O.D. (mg/l)  float64
NITRATENAN N+ NITRITENANN (mg/l) float64
FECAL COLIFORM (MPN/100ml) float64
TOTAL COLIFORM (MPN/100ml)Mean float64
year           int64
dtype: object
```

Figure 3 | Numeric data type.

3.3. Calculation of the WQI

We will be using the 'Weighted Arithmetic Water Quality Index Method' to calculate the WQI of each water sample. The WQI is calculated using the following equation (Latha *et al.* 2007; Usha & Kumar 2013):

$$WQI = \sum_{i=1}^n w_i q_i \quad (1)$$

where w_i is the weightage factor of i th parameter and q_i is the quality rating factor of i th parameter; w_i is calculated using the following equation:

$$w_i = k/S_n \quad (2)$$

where S_n is the standard value of the i th parameter, and k is a constant, calculated as

$$k = \frac{1}{\sum_{i=1}^n \frac{1}{V_{si}}} \quad (3)$$

q_i in Equation (1) is based on the following equation:

$$q_i = \frac{V_a - V_i}{V_s - V_i} \times 100 \quad (4)$$

where V_a is the value obtained from analysis of the i th parameter. V_s is the value of the i th parameter and V_i is the ideal value.

We use our new parameters, i.e. npH , ndo , nco , $nbdo$, nec , and nna to calculate the weighted average based on the formulae given below and store the results in new columns to wph , wdo , $wbdo$, wec , wna , and wco (Rustam *et al.* 2022).

$$wph = npH \times 0.165 \quad (5)$$

$$wdo = ndo \times 0.281 \quad (6)$$

$$wbdo = nbdo \times 0.234 \quad (7)$$

$$wec = nec \times 0.009 \quad (8)$$

$$wna = nna \times 0.028 \quad (9)$$

$$wco = nco \times 0.281 \quad (10)$$

Now the WQI is calculated using the formula (Nayan *et al.* 2021):

$$WQI = wph + wdo + wbdo + wec + wna + wco \quad (11)$$

Once the WQI is calculated, we classify the water sample into potable or non-potable water based on its value, as shown in Table 3 (Wong & Rylko 2014).

Table 3 | WQI to predict potable and non-potable of water

WQI	Potability
More than 75	Potable (1)
Less than 75	Non-potable (0)

4. MODELS USED

We have proposed different machine learning models like KNN, XGBoost, and Random Forest to get the best performance. We have also used an ANN, an advanced machine learning approach to get the model performance.

We consider five scenarios, as shown in Table 4. Each scenario takes one more input variable than the previous one to observe how the presence of each input parameter affects the accuracy, precision, recall, and F1 score of the model.

Table 4 | Input scenarios

Scenario	Input variables
Scenario 1	DO
Scenario 2	DO, pH
Scenario 3	DO, pH, conductivity
Scenario 4	DO, pH, conductivity, BOD
Scenario 5	DO, pH, conductivity, BOD, and nitrate

4.1. Machine learning approach

Machine learning is a field of study in Computer Science where the machine learns from data and gives out predictions and insights. There are many machine learning algorithms formulated by mathematical and statistical concepts that help the machine predict or derive insights from the fed information. Then, these insights influence critical growth and key performance measures through decision-making within applications and companies. The

need for data scientists will increase as Big Data develops, requiring their assistance in determining the most pertinent business issues and, as a result, the data required to answer them.

Our problem statement is a classification problem that finds the portability of water. Various models can be used to solve a classification problem. Some of the approaches being used are as follows.

KNN: *K* neighbors-based categorization is a form of lazy learning because it only saves instances of the training data rather than attempting to create a general internal model. Each point is classified by a simple majority vote of its KNN. According to its name, the KNN algorithm works in such a way that a new case is compared to its closest neighbors, which already exist in the model weights, thus matching the name ‘nearest neighbors’. This algorithm is one of the simplest and most intuitive algorithms under supervised learning.

The KNN algorithm uses the previously stored data points to classify and categorize the new data points. Therefore, the algorithm is helpful in classifying and categorizing new data into previously well-defined classes. The algorithm can be used for both regression as well as classification problems, but it is more common to use it for classification purposes. Following a nonparametric approach, the KNN algorithm does not assume anything about the data.

RF: The RF classifier, a meta-estimator, uses an average to increase the model’s predictive accuracy and reduce overfitting by fitting several decision trees on various subsamples of datasets. The samples are generated via replacement; however, the subsize of the sample is always the same as the original input sample’s size. Because of their simplicity, on the one hand, and typically excellent performance, on the other, RFs, like naive Bayes and KNN algorithms, are well-liked. In contrast to the first two methods, RFs are somewhat unpredictable in terms of how the trained model will be structured. Due to the stochastic nature of tree construction, this is an inevitable result. One of the main reasons why this feature of RFs can be problematic in regulatory settings is that clinical adoption often necessitates a high level of repeatability, not only in terms of the algorithm’s ultimate performance but also in terms of the mechanics by which a particular decision is made. The decision tree is the basic building block of RF classifiers. A decision tree is a hierarchical structure created from a dataset’s characteristics (or independent variables). The decision tree is divided into nodes based on a measure connected with a subset of the features. A RF is built upon multiple decision trees. A decision tree is a machine learning algorithm where a tree is created based on attribute impact hierarchy created from the dataset’s characteristics. The nodes of the tree are divided. The RF is made up of a set of decision trees that are linked to a set of bootstrap samples created from the original dataset. The nodes are divided using the entropy of a subset of the characteristics. Subsets from the original dataset that are the same size as the original dataset are created via bootstrapping. Through the idea of overlap thinning, the bootstrapping technique makes it easier to create a RF with the appropriate number of decision trees to improve classification accuracy. After that, the best trees are chosen through a voting process and a process called bagging (bootstrap aggregate).

XGBoost: The XGBoost classifier got the greatest results after attempting several different methods to get the best performance out of our model. XGBoost is an ensemble learning technique. Relying just on a single machine learning model’s results might not always be sufficient. A technique for systematically combining the prediction skills of several learners is ensemble learning. The final model integrates the outputs of various other models into one. In boosting, the trees are constructed one after the other, with each succeeding tree attempting to minimize the errors of the one before it. Since it is a boosting method, each tree improves its accuracy by learning from the errors of the previous tree. In boosting strategy, the weak learners are combined into strong learners as each weak learner provides information which makes the next model stronger. The last strong learner lowers the bias as well as the variance. Boosting employs trees with fewer splits compared to bagging methods like RF, which uses trees that have been fully formed. Such small trees, which are not very deep, are quite simple to interpret. We can enhance the capabilities of this model by performing some hyperparameter tuning such as controlling the number of trees, adjusting the learning rate that affects the speed and efficiency with which the model reaches global minima, and controlling the depth of the tree as well so that bias in the model is not increased. There could be overfitting if there are a lot of trees present. Therefore, hyperparameter tuning helps in reducing the chances of overfitting and gives a smooth model which gives accurate results.

Using the train test split function from the sklearn package, we divided our dataset into test and train datasets before implementing the XGBoost classifier. This assists in dividing our dataset in the desired ratio so that a portion is utilized to develop our machine learning model and the remaining portion is used to test our model. By determining how closely our actual values match our predicted values, we can then determine the model’s accuracy.

4.2. ANN approach

ANNs are a set of interconnected neurons that imitate the neurons of the biological brain. Just like the animal brain where the neurons are interconnected, in ANNs, the nodes are connected to each other in a similar manner that exchange information with one another despite being on different layers. Edges are the connections that form between two neurons or nodes. The weight of the neurons and edges often changes as learning progresses. These weights control how strong a signal is at a connection. In order to develop probability-weighted correlations between the two that are stored within the network's data structure, neural networks learn (or are trained) by examining samples with known 'input' and 'output.' When training a neural network from a given sample, it is usually assessed how the network's processed output (generally a prediction) differs from a target output. The flaw is this difference. Using this error value and a learning technique, the network then adjusts its weighted associations. The output produced by the neural network will get even more similar to the target output with each adjustment. When enough of these improvements have been made, the training is adjudicated based on predetermined criteria. The architecture of a neural network primarily consists of three layers:

The input layer: This layer accepts inputs in several different formats as provided by the user.

Hidden layers: This layer, which is concealed between the input and output layers, may include one or more layers. All computations are made in order to uncover patterns and hidden features.

Output layer: This layer is finally produced using this layer after the input layer has undergone a number of changes using the hidden layer.

After trying several approaches to build a suitable model, we came up with the following model as shown in Figure 4. The model consists of two hidden layers between the input and output layers. The model architecture is sequential. There are six input dimensions for our model. The first hidden layer has 8 nodes, followed by another hidden layer having 16 nodes. The activation function used in the input and hidden layers is the ReLU activation function. A good advantage of this is that it does not activate all the neurons at the same time. For our output layer, we have used the softmax activation function, which is used when we need to obtain probability distribution for the output. We then train our model for 150 epochs and see the results obtained.

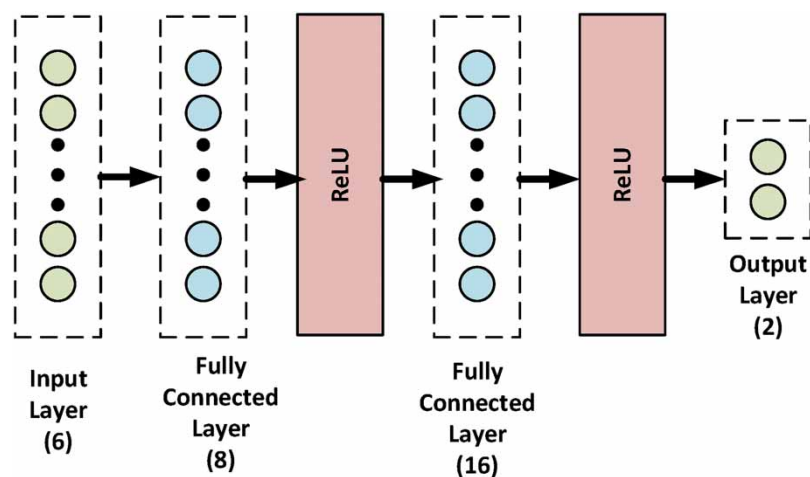


Figure 4 | Proposed ANN architecture.

5. RESULTS

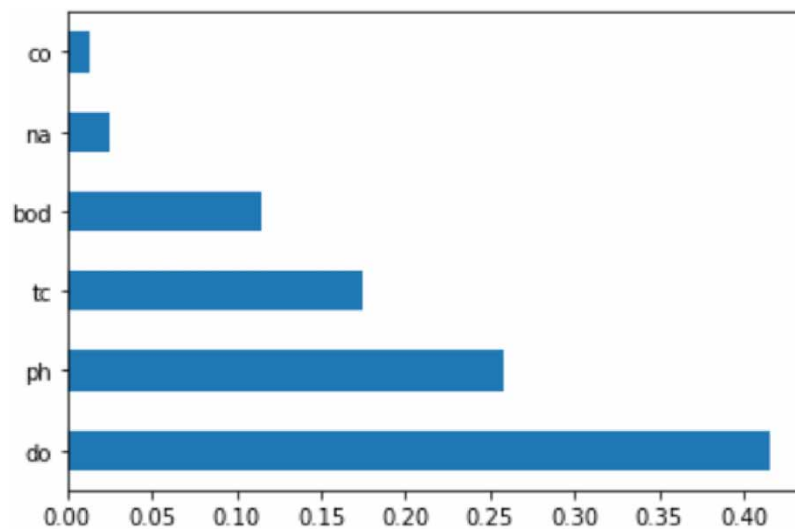
Given in Table 5 are the results of using the different scenarios as inputs. As it is seen, the accuracy is the maximum when all the input parameters are used proving that all of them are vital for determining the WQ. The graph in Figure 5 shows the importance of every feature.

5.1. Using the KNN model

On applying the KNN model to our dataset, it is found to have an accuracy of 75.67%. The confusion matrix for the same is shown in Figure 6. As the matrix depicts, there are 222 true-negative and 61 true-positive values.

Table 5 | Performance measures of the proposed work

Algorithm	Metrix	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5	All variables
XGBoost	Accuracy	0.8	0.87	0.88	0.91	0.91	0.98
	Precision	0.82	0.87	0.88	0.91	0.91	0.98
	Recall	0.8	0.87	0.88	0.91	0.91	0.98
	F1 score	0.79	0.87	0.88	0.91	0.91	0.98
	Matthews coeff.	0.56	0.7	0.72	0.79	0.79	0.96
RF	Accuracy	0.8	0.85	0.88	0.89	0.9	0.97
	Precision	0.82	0.85	0.88	0.89	0.9	0.97
	Recall	0.8	0.85	0.88	0.89	0.9	0.97
	F1 score	0.79	0.85	0.88	0.89	0.89	0.97
	Matthews coeff.	0.56	0.67	0.72	0.75	0.77	0.92
KNN	Accuracy	0.79	0.87	0.71	0.72	0.73	0.75
	Precision	0.8	0.87	0.7	0.71	0.73	0.75
	Recall	0.79	0.87	0.71	0.72	0.73	0.76
	F1 score	0.78	0.87	0.69	0.7	0.7	0.74
	Matthews coeff.	0.53	0.71	0.31	0.33	0.35	0.43

**Figure 5** | Feature importance.

As shown in Figure 7, we further evaluate this model's receiver operating characteristics (ROC) area under curve (AUC) curve. The AUC-ROC curve is a performance metric for classification problems at different threshold levels. AUC measures how separable something is, whereas ROC is a probability curve. This reveals how effectively the model can distinguish between classes. The accuracy of the model's prediction of 0 classes as 0 and 1 classes as 1 is shown by the AUC. The model's ability to reliably predict the various classes improves with increasing AUC. The AUC value for the KNN model is 0.69, which means it is not very accurate in distinguishing between the classes.

5.2. Using the RF model

On applying the RF model to our dataset, it is found to have an accuracy of 96.79%. The confusion matrix for the same is shown in Figure 8. As the matrix depicts, there are 239 true-negative and 123 true-positive values. Furthermore, it only has six false positives and six false negatives.

Now we observe the AUC-ROC curve for the RF model as shown in Figure 9. The AUC value for the RF model is 0.96, which means that it is pretty accurate in distinguishing between the classes.

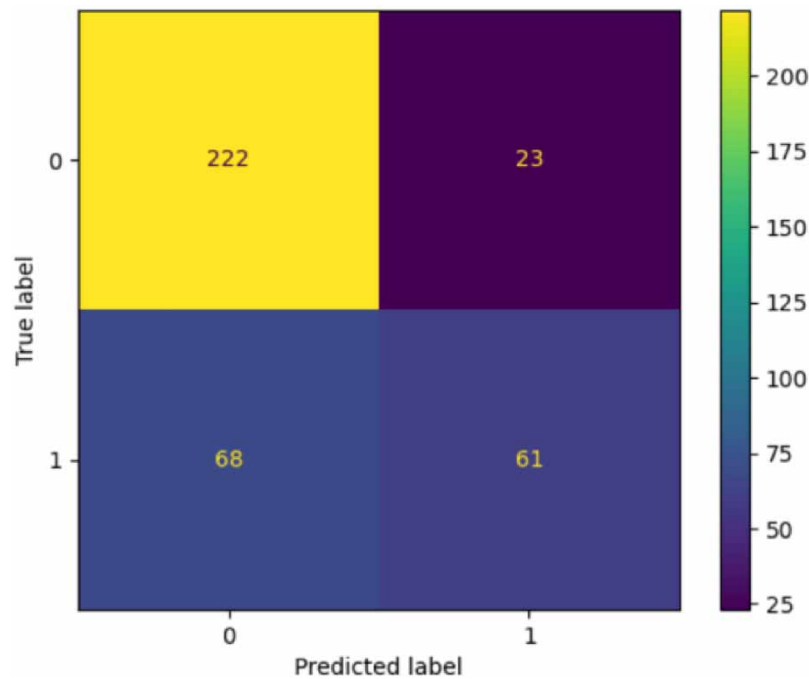


Figure 6 | Confusion matrix using the KNN model.

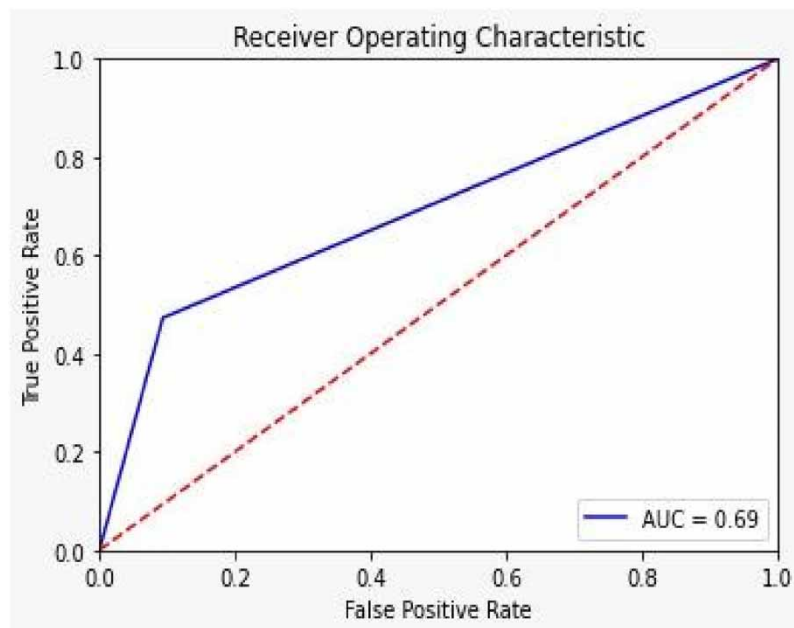


Figure 7 | AUC-ROC curve for the KNN model.

5.3. Using the XGBoost classifier

On applying the XGBoost classifier model to our dataset, it is found to have an accuracy of 98.13%. The confusion matrix for the same is shown in [Figure 10](#).

There are 123 true-positive values and 244 true-negative values, as shown in the matrix. We do hyperparameter adjustments to further increase the accuracy of our model. Hyperparameter tuning is the process of identifying the ideal model architecture from a set of parameters known as hyperparameters, which determine the model architecture. The model parameters indicate how to convert the input data into the desired output, whereas

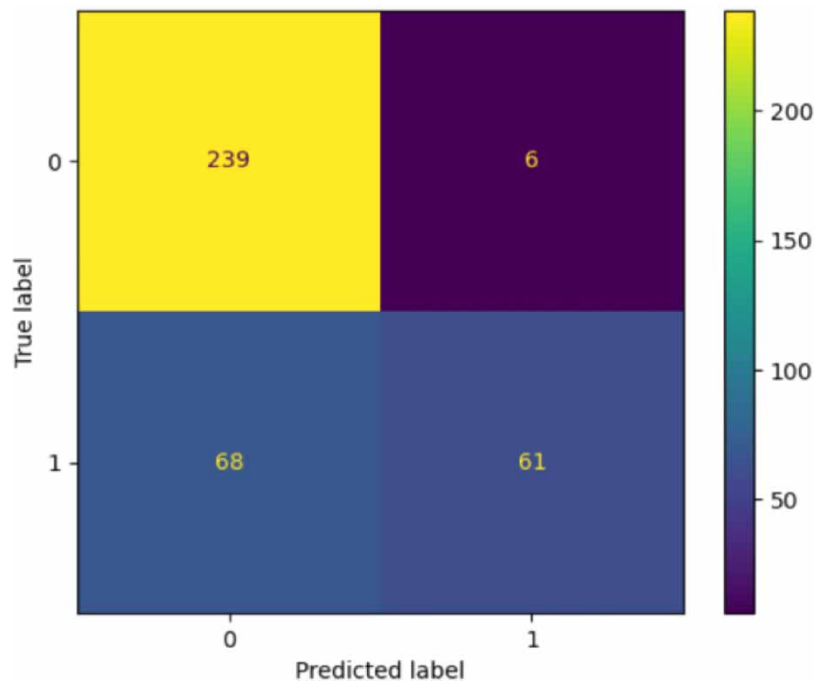


Figure 8 | Confusion matrix using the RF model.

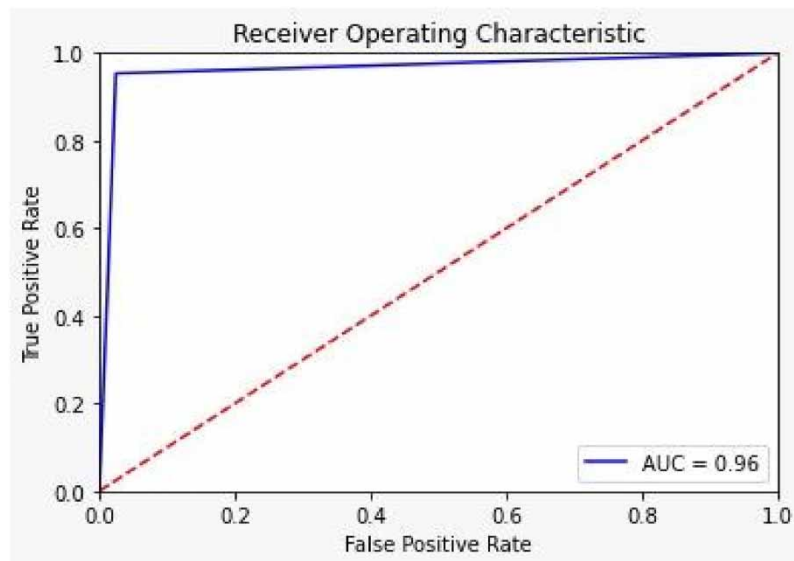


Figure 9 | AUC-ROC using the RF model.

the hyperparameters control how our model is organized. Upon using GridSearchCV to find the best suitable parameter values from our given parameter grid, we apply these values as the hyperparameter values in our model to increase its accuracy as shown in Table 6.

On performing hyperparameter tuning, the accuracy of our model is increased to 98.93% and the confusion matrix is shown in Figure 11. Compared to the untuned model, our tuned model has 245 true-negative and 125 true-positive values as shown in the matrix, depicting that the accuracy has increased. Hence, the highest accuracy achieved by our machine learning model after trying various approaches was 98.93%.

Now we observe the AUC-ROC curve for the XGBoost classifier model as shown in Figure 12. The AUC value for the XGBoost model is 0.9844, which means that it is more accurate in distinguishing between the classes compared to the previous two models.

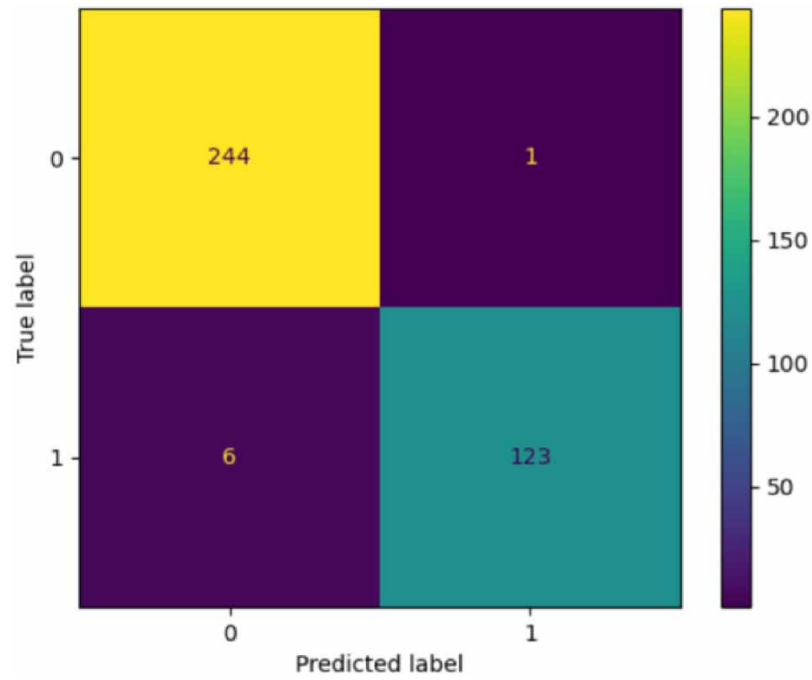


Figure 10 | Confusion matrix using the XGBoost classifier.

Table 6 | Tuned XGBoost parameters

learning_rate	max_depth	min_child_weight	n_estimators	Objective	Subsample	Bytree
0.01	6	3	150	reg:squared error	0.5	0.7

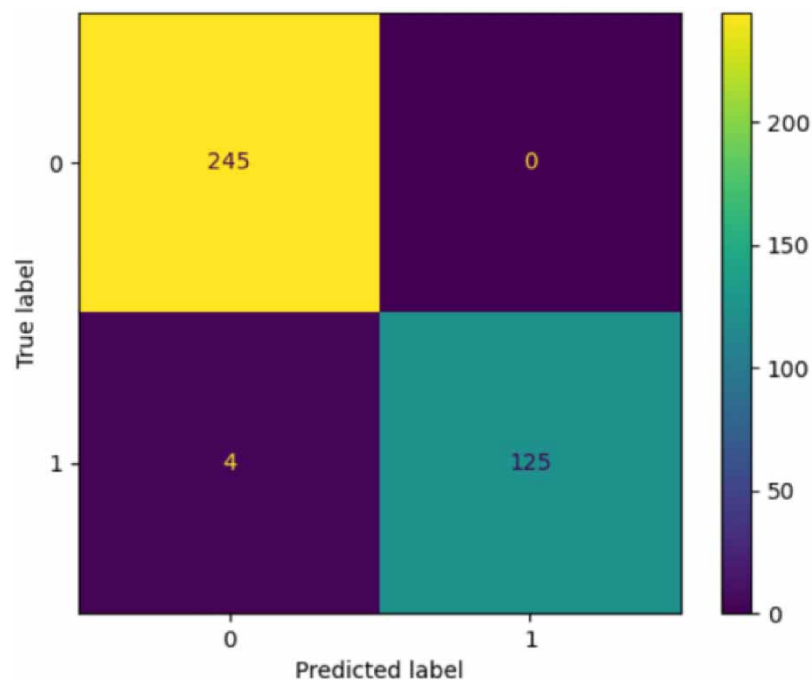


Figure 11 | Confusion matrix after hyperparameter tuning using the XGBoost classifier.

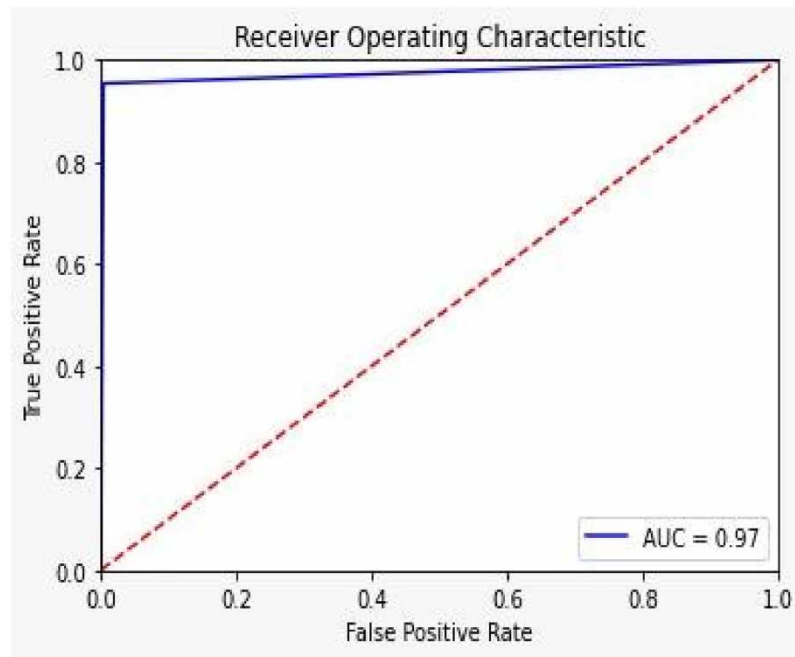


Figure 12 | AUC-ROC curve using the XGBoost classifier without hyperparameter tuning.

In our study, we conducted a comprehensive assessment of the predicted WQI values in comparison to the actual, observed values shown in Table 7. This evaluation is vital for gauging the accuracy and reliability of the models or methods employed for WQI estimation. By meticulously examining the disparities between our predictions and the ground truth, we gain insights into the effectiveness and precision of our chosen approach. This comparison not only helps us to understand the trustworthiness of our model but also holds significance for WQ management, with far-reaching implications for environmental and public health.

5.4. Using ANN

On training our ANN model for different numbers of layers, epochs, and optimizers, it was found that the most accurate ANN model for our dataset had a validation accuracy of 90.21% and a loss of 0.2824, which means there was very little overfitting in our model. The graph in Figure 13 shows the accuracy of our model with respect to the number of epochs for our test and train datasets, and as it is visible, there is very little overfitting meaning our model is accurate.

The next graph represents the loss of our model with an increasing number of epochs. Loss signifies the difference between accuracy and validation accuracy. The lesser the loss function the more accurate the model is. As

Table 7 | Actual versus predicted values for the WQI

Index	Actual	Predicted
1,728	72.86	72.89
60	89.14	89.17
189	93.28	93.34
1,626	87.66	87.65
1,172	81.42	81.17
1,486	66.44	66.29
770	88.38	88.39
455	87.66	87.76
127	66.44	66.49
1,502	82.40	82.39

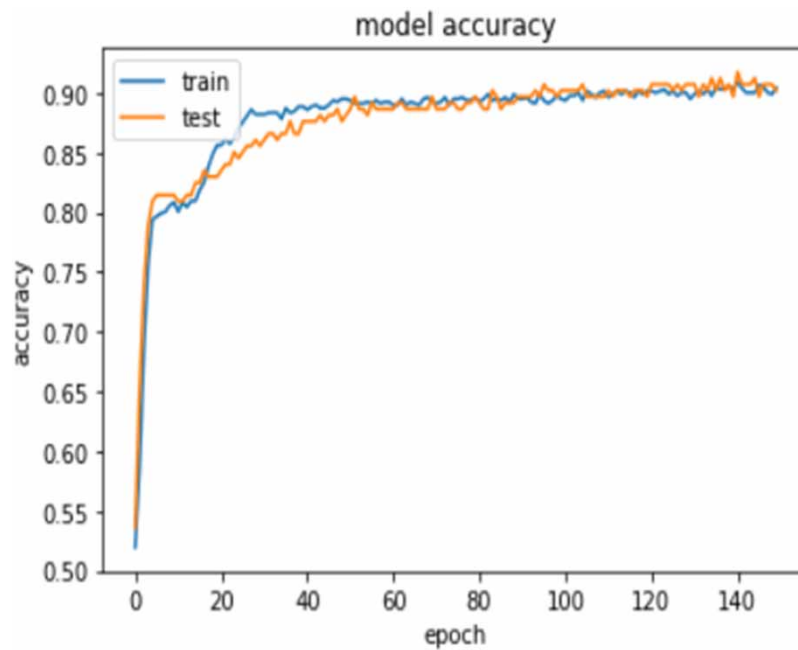


Figure 13 | Model accuracy during training and testing using the ANN.

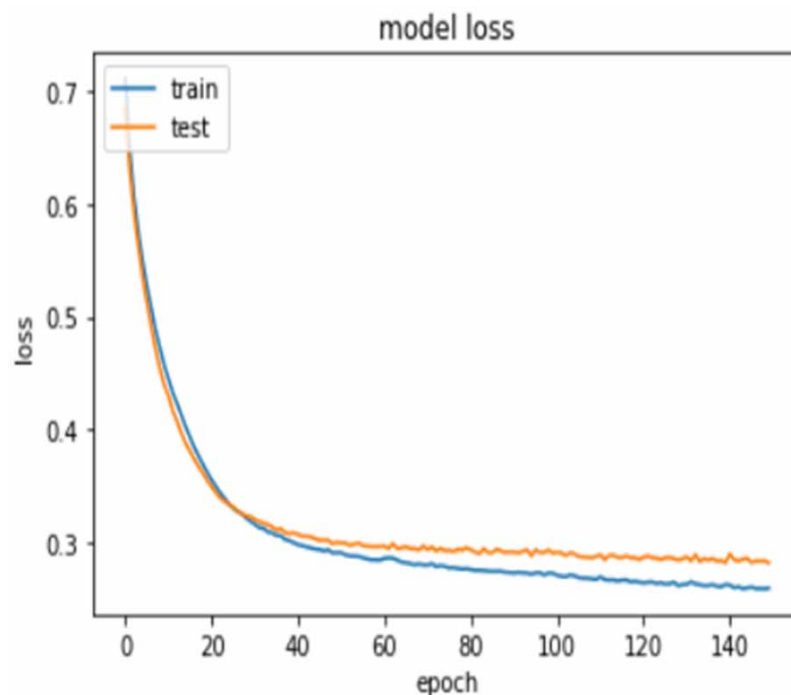


Figure 14 | Model loss using the ANN.

seen in the graph in Figure 14, the loss in the test and train datasets is continuously decreasing with more epochs and the difference between them is very less.

Table 8 contains the accuracy, precision, recall value, F1 score, and AUC-ROC, which have been calculated for both our models, and it is evident that the XGBoost model is a better approach when analyzing the potability of water.

Table 8 | Accuracy, precision, recall value, F1 score, and AUC-ROC values

Model	Accuracy	Precision	Recall	F1 score	AUC-ROC
KNN	0.7567	0.7261	0.4728	0.5727	0.6894
RF	0.9679	0.9534	0.9534	0.9534	0.9644
XGBoost classifier	0.9893	1.0	0.9689	0.9842	0.9844
ANN	0.9021	0.9560	0.8446	0.8969	0.9003

To illustrate the importance of the proposed method suggested in this study, a comparison of its performance was conducted. The results presented in [Table 9](#) clearly demonstrate that, in both scenarios, the proposed approach outperformed previous studies.

Table 9 | Performance comparison with existing approaches

References	Model	Accuracy (%)
Patel et al. (2022)	Explainable AI	80
Aldhyani et al. (2020)	LSTM	97.1
Al Duhayyim et al. (2022)	Fuzzy deep neural network	98.1
Rustam et al. (2022)	ANN	96
Azroul et al. (2022)	ANN	85.11
Proposed work	XGBoost	98.93

6. CONCLUSION

This paper focuses on the implementation of machine learning algorithms and ANNs for the detection of the WQI, which further helps us to conclude if the water is viable or not. Machine learning implies using already available data to make predictions further. Here, since we know the parameters that are used to determine the WQI, we take data from the Indian government site to predict the potability of water. It is useful as if we can determine the potability of water for a given amount of the parameters the waste or causes that make the water unfit for drinking can be eliminated. There were five different machine learning approaches, such as KNN, XGBoost with and without hyperparameter tuning, RF, and sequential ANN were tested, to see which one would be the most optimal for our cause, that is predicting the potability of water. Out of all five models, the most accuracy was obtained for hyperparameter tuning to the XGBClassifier model. The model has given an accuracy of 98.93%, an F1 score of 98.42%, a precision of 100%, and a recall of 95.60%. This model can be implemented to create a website or mobile application to help people check the WQI. It can also be integrated with environmental monitoring systems and Internet of Things (IoT) devices for automated data collection. This can be used to send alerts if the WQI of a region being monitored drops below a threshold so that preventive measures can be taken well ahead.

With the steady increase in global population and industrialization, the imperative to monitor and preserve WQ is more pressing than ever. Water, a fundamental resource for life and industry, necessitates vigilant oversight. By deploying the designed model over cloud services with an IoT device embedded with an array of essential sensors including pH, turbidity, DO, and temperature sensors, we can enhance the capabilities of both, making the whole ecosystem more efficient and helpful. These sensors are strategically positioned to monitor the WQ at its source, allowing us to observe critical parameters directly impacting purity. The real-time connectivity of these devices to predictive models will empower us to swiftly detect deviations in the WQ and anticipate potential issues. Employing data visualization techniques will help provide comprehensive insights into the state of water. This technological advancement is pivotal not only in safeguarding public health but also in promoting the sustainable management of industries dependent on water. As we confront environmental challenges and the looming impacts of climate change, this technology takes a proactive stance toward addressing WQ concerns. By bridging the worlds of IoT, data analysis, and visualization, this solution equips us to better protect and preserve one of our most indispensable resources, water.

DATA AVAILABILITY STATEMENT

All relevant data are included in the paper or its Supplementary Information.

CONFLICT OF INTEREST

The authors declare there is no conflict.

REFERENCES

- Ahmed, N. *et al.* 2019 Machine learning methods for better water quality prediction. *J. Hydrol.* **578**.
- Al-ani, I. 2019 Mathematical computation of water Al-Hilla river ecosystem. *Int. J. Civ. Eng. Technol.* **10**(01).
- Aldhyani, T. H., Al-Yaari, M., Alkahtani, H. & Maashi, M. 2020 Water quality prediction using artificial intelligence algorithms. *Appl. Bionics Biomech.* **2020**.
- Al Duhayyim, M., Mengash, H. A., Aljebreen, M., Nour, M., Salem, N., Zamani, A. S., Abdelmageed, A. A. & Eldesouki, M. I. 2022 Smart water quality prediction using atom search optimization with fuzzy deep convolutional network. *Sustainability.* **14**(24), 16465.
- Azrou, M., Mabrouki, J., Fattah, G., Guezaz, A. & Aziz, F. 2022 Machine learning algorithms for efficient water quality prediction. *Model. Earth Syst. Environ.* **8**(2), 2793–2801.
- Bucak, I. O. & Karlik, B. 2011 Detection of drinking water quality using CMAC based artificial neural networks. *Ekoloji* **20**(78), 75–81.
- Cabral Pinto, M. M. S. *et al.* 2019 An inter-disciplinary approach to evaluate human health risks due to long-term exposure to contaminated groundwater near a chemical complex. *Expo. Heal.* **12**(2), 199–214.
- Clean Water for a Healthy World.
- Dogo, E. M., Nwulu, N. I., Twala, B. & Aigbavboa, C. 2019 A survey of machine learning methods applied to anomaly detection on drinking-water quality data. *Urban Water J.* **16**(3), 235–248.
- Drinking-water. 2022 Available from: <https://www.who.int/news-room/fact-sheets/detail/drinking-water> (accessed 14 July 2022).
- Hassan, O. N. 2019 Water quality parameters. In: *Water Quality – Science, Assessments and Policy*. IntechOpen.
- Julien, P. Y. 1992 *River Mechanics*. Cambridge Univ. Press, p. 427.
- Kashefipour, S. M. 2002 *Modelling Flow, Water Quality and Sediment Transport Processes in Riverine Basins*. Cardiff University.
- Khan, Y. & See, C. S. 2016 Predicting and analyzing water quality using machine learning: A comprehensive model. In 2016 *IEEE Long Island Systems, Applications and Technology Conference, LISAT 2016*, pp. 1–6.
- Latha, P. S., Rao, K. N., Kumar, P. R. & HariKrishna, M. 2007 Water quality assessment at village level – A case study. *Indian J. Environ. Prot.* **27**(11).
- Liu, J., Yu, C. & Hu, Z. *et al.* 2020 Accurate prediction scheme of water quality in smart mariculture with deep Bi-S-SRU learning network. *IEEE Access* **8**, 2983–2989.
- Motagh, M. *et al.* 2017 Quantifying groundwater exploitation induced subsidence in the Rafsanjan plain, southeastern Iran, using InSAR time-series and in situ measurements. *Eng. Geol.* **218**, 134–151.
- Nayan, A. A., Saha, J., Mozumder, A. N., Mahmud, K. R., Al Azad, A. K. & Kibria, M. G. 2021 A machine learning approach for early detection of fish diseases by analyzing water quality. *Trends Sci.* **18**(21), 1–11.
- Patel, J., Amipara, C., Ahanger, T. A., Ladhva, K., Gupta, R. K., Alsaab, H. O. ... & Ratna, R. 2022 A machine learning-based water potability prediction model by using synthetic minority oversampling technique and explainable AI. *Comput. Intell. Neurosci.* **2022**.
- Rustam, F., Ishaq, A., Kokab, S. T., de la Torre Diez, I., Mazón, J. L. V., Rodríguez, C. L. & Ashraf, I. 2022 An artificial neural network model for water quality and water consumption prediction. *Water.* **14**(21), 3359.
- Sagar, S. K. S. S., Chavan, R. P., Patil, C. L. & Shinde, D. N. 2015 Physico-chemical parameters for testing of water-A review. *Int. J. Chem. Stud.* **3**(4), 24–28.
- Sagan, V. *et al.* 2020 Monitoring inland water quality using remote sensing: Potential and limitations of spectral indices, bio-optical simulations, machine learning, and cloud computing. *Earth Sci. Rev.* **205**, 1–68.
- Shailesh Jaloree, S. G. & Rajput, A. 2014 Decision tree approach to build a model for water quality. *Bin. J. Data Min. Netw.* **4**, 25–28.
- Solanki, A., Agrawal, H. & Khare, K. 2015 Predictive analysis of water quality parameters using deep learning. *Int. J. Comput. Appl.* **125**(9), 29–34.
- Usha, S. A. & Kumar, P. 2013 Determination of water quality index and fitness of urban water bodies in Bilari town of Moradabad (Uttar Pradesh). *J. Chem. Pharm. Res.* **5**(11), 726–731.
- Water Quality 2022a *Water Quality | Teaching Great Lakes Science*. Available from: <https://www.michiganseagrant.org/lessons/lessons/by-broad-concept/earth-science/water-quality/> (accessed 11 July 2022).
- Water Quality 2022b *Water-Quality | Open Government Data (OGD) Platform India*. Available from: <https://data.gov.in/dataset-group-name/water-quality> (accessed 28 July 2022).
- Wong, C. & Rylko, M. 2014 Health of the Salish Sea as measured using transboundary ecosystem indicators. *Aquat. Ecosyst. Heal. Manag.* **17**(4), 463–471.
- Yan-jun, L. & Qian, M. 2012 AP-LSSVM modeling for water quality prediction. In: *Proceedings of the 31st Chinese Control Conference*, pp. 6928–6932.

First received 21 February 2023; accepted in revised form 3 November 2023. Available online 16 November 2023