The Scientific Temper (2023) Vol. 14 (2): 527-532
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.2.46
E-ISSN: 2231-6396, ISSN: 0976-8653
https://scientifictemper.com/

REVIEW ARTICLE

Water Quality Prediction using AI and ML Algorithms

B. Nivedetha

Abstract

Expeditious growth in industrial amelioration to support the country's expanding population and economy has contaminated our water resources like never before. Water pollution is one of the most alarming concerns for us today. Prediction of water quality has grown in popularity in the field of water environmental science. Data-driven strategies are becoming increasingly fascinating and beneficial as we extend our understanding of water means. Data mining, which can manage the complexity within the provided data, is a direct method for exploration.

Keywords: Water pollution, Quality parameters, Water Quality Index, AI & ML Algorithms.

Introduction

Water quality has been debased due to miscellaneous forms of pollution. The controlling and monitoring of drinking water, water for agriculture, agriculture and industrial wastewater is becoming increasingly captivating as a result of its influence on human existence and the ecosphere. Accurate water quality forecast is the basis of water environment management and is of great consequence for water environment protection (Wang *et al.*, 2017). Data mining is the process of examining for beneficial information in huge data sets and for patterns in it. The critical indicators of water quality and their tolerable limits are in Table 1.

Survey on AI and ML Algorithms

LSTM Algorithm

Deep learning methods are considered as a kind of machine learning to instinctively take in the features of data. In recent years, Scholars have attempted to solve time series prediction problems using deep-learning-based algorithms. The long and short memory neural network has memory due to its own special network topology. It has been effectively used in the area of temporal prediction, such as stock forecasting and traffic flow forecasting.

Data from time series on water quality indicators exhibit substantial seasonal fluctuation and are strongly impacted by the seasons. Water quality forecasting is a type of time series forecasting (Liu *et al.*, 2000). Because typical neural networks aren't well suited to process information from time series, thee research suggests an LSTMNN-based water quality prediction approach. To begin, we incorporated layers of input, hidden and output to a prediction model. Second, historical water quality indicators are used to train the algorithm. Finally, parameter preferences and the number of runs increase forecast accuracy (Yan *et al.*, 2019). LSTM NN approach is compared to two other systems for water quality prediction: One is focused on an online sequential extreme learning machine, while the other is based on a back propagation neural network. The outcome validates the efficacy of the strategy expressed.

LSTM is a current neural network-based algorithm. It enhances the RNN design. RNN adds a specific structure to the hidden layer to process information from time series. When using the back-propagation algorithm, RNN suffers a vanishing gradient issue (Jaloree, 2014; Heddam, 2016 ; Nivedetha and Vennila, 2020). Sepp Hochreite and Jurgen Schmidhuber introduced LSTM NN, a novel deep learning technique with RNN basis to solve this issue. The memory block is an arrangement of LSTM NN neurons that keeps the erroneous flow constant. As a result, dealing with time series data is preferable.

Developed system on basis of LSTM for forecasting the quality of water in order to make reliable predictions

Department of Electrical and Electronics Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India

***Corresponding Author:** B. Nivedetha, Department of Electrical and Electronics Engineering, PSG College of Technology, Coimbatore, Tamil Nadu, India, E-Mail: bna.eee@psgtech.ac.in

How to cite this article: Nivedetha, B. (2023). Water Quality Prediction using AI and ML Algorithms. The Scientific Temper, **14**(2):527-532.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.2.46

Source of support: Nil

Conflict of interest: None.

Received: 21/04/2023 **Accepted:** 20/05/2023 **Published** : 30/06/2023

Table 1: Quality criteria and WHO limits

Attributes	*Limits*
pH	6.5– 8.5
Turbidity	5NTU
Appearance	Clear
Conductance	2000 µS/cm
Chlorides	200 mg/L
Nitriteas NO2-	<1-mg/L
Fecal Coliforms	Nil Colonies/100 mL
Hardness as CaCO3	500 mg/L
Calcium	200 mg/L
Total Dissolved Solids	1000 mg/L
Alkalinity	500 mg/L

of quality indicators, as shown in Figure 1. In contrast to standard neural networks, the nodes of the hidden layer are completely linked and have a memory block structure (Gakii and Jepkoech, 2019).

BI-S-SRU Algorithm

The essential water quality characteristics are predicted using this Bi-S-SRU algorithm as shown in Figure 2. On average, it takes 12.5 milliseconds to predict data, with a prediction accuracy of 94.42%. It features a basic structure, a quick convergence rate, and strong stability. In addition, we contrast the Bi-S-SRU strategy with different approaches and examine the prediction outcomes of several water quality metrics in the same ecological setting.

The Bi-S-SRU network may include future context knowledge into present time point data prediction. The main idea behind Bi-S-SRU will overlay SRU in front and back in all teaching methodologies, with the SRUs coupled to output. This system gives the past and future contextual details for every position in the entry order of the output layer (Lu and Ma, 2020; Bouamar and Ladjal, 2008; Han *et al.*, 2011). Furthermore, no movement of info on the front and back hidden layers, ensuring an on-cyclic expansion diagram. The Figure 3 depicts: Input weight to front and back concealed levels w1 & w3, weights moved on concealed level w2 & w5 and front & back concealed level weight to output w4 & w6.

The Bi-S-SRU computation technique is split as parts: forward pass and backward pass. The front computation procedure of the concealed level of Bi-S-SRU in the forward pass is identical to one-way SRU, with the exception that the input sequences for the two hidden levels is reversed. The two hidden levels must process all of the input sequences before updating the output layer.

The back pass of Bi-S-SRU in the second stage is comparable to the RNN back propagation procedure (Kang *et al.*, 2018; Balan and Ila, 2020; Yan *et al.*, 2020). Several things are initially placed in output for computing in every time units and the objects are returned to the hidden layers in opposite ways (Balajee and Durai, 2021).

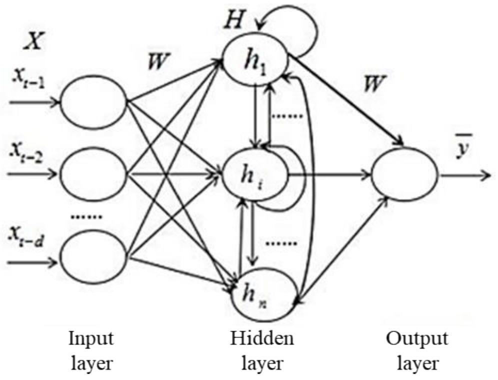

Figure 1: The LSTMNN-based quality forecast

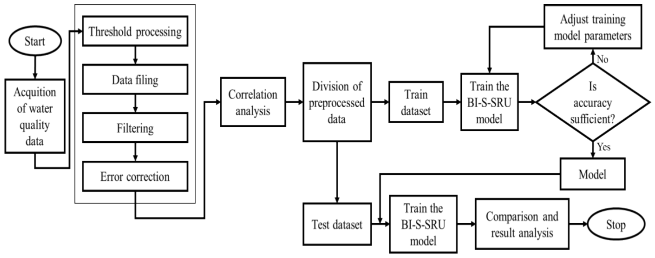

Figure 2: Bi-S-SRU method flow

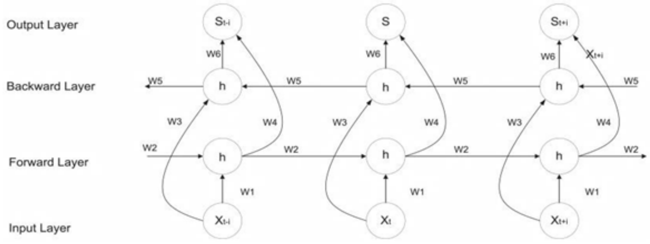

Figure 3: Six weight sets for bi-s-sru

Support Vector Machines

Support-vector machines or networks are supervised learning models used in machine learning to analyse data for classification and regression study. SVM plots tutoring samples to locations in space as much as feasible in order to expand the gap between the two categories. Following that, new instances are projected onto the same place and categorized based as to which part of the gap they lie on. SVMs are capable of both linear and nonlinear classification. The kernel approach includes transforming their data into large number of feature spaces indirectly.

Decision Tree Based Algorithms

There are numerous advantages of using a decision tree-based model:

- Capable of dealing with data and ordinary characteristics.
- Absent value insensitivity.
- The decision tree only has to be generated once, so it is extremely efficient.

In reality, there are alternatives like ANN and SVM in the realm of machine learning. In comparison, decision tree-based systems could compute more quickly and are more suited for making short-term forecasts. Furthermore, the decision tree offers a prediction benefit since data from water quality monitoring systems can include missing data resulting infault.

A decision tree can be built relatively quickly and easily when compared to the methods. A decision tree is an ML approach that divides data into mutually exclusive groups to estimate quantitative target variable or categorize observations onto one of a categorized destination variable's categories. Recursive partitioning is a frequent approach for building decision trees and ordinary algorithms include Chi-square Automatic Interaction Detection, Classification & Regression Tree and C5.0.

The accuracy of the model was examined using the Hoeffding tree, Decision Stump, Random forest, LMT and J48. Decision Stump had the lowest accuracy of 83%, while J48 decision tree achieved the highest correctness of 94%.

J48Tree

The J48 tree is a self instructed predictive system that predicts a fresh example's goal output dependingon the given data's multiple attribute values. A decision tree's internal nodes represent several characteristics. The branches between the nodes reflect the potential levels of these qualities in the samples obtained, whereas the terminal points represent the dependent variable's ultimate value.

Logistic Model Tree

LMT is a classification system coupled to a monitored tutoring process that combines logistic prediction with decision tree learning. To give a linear regression model by section, logistic model trees utilize a decision tree with models of linear regression on the leaves.

Random Forest Algorithm

A random forest is a learning approach to regression, classification and others in which a lot of decision trees are formed when the decision was made and the class, which is the classification or individual tree forecast, is output. Random forests correct the decision tree behavior which is present in the training example. Random forests provide away of minimizing variation by computing the average of numerous deep decision trees built with various portions of the same training set. This considerably improves the appearance of the final model at the cost of a slight rise in complexity and a reduction of interpretability.

As built on the bagging approach, it's an additive system; and it is one of the representatives of ensemble learning. When creating each tree, unlike bagging, prior to each node division, RF utilizes a randomly sampled predictor to potentially reduce bias. It has below traits:

- The addition of two random values ensures that RF doesn't succumb too much fitting & possesses great immune to noise.
- Handles data-set with huge features and also not with feature preferences.
- A rapid speed of training and is easily parallelized, making reasonably straightforward for implementation.

Hoeffding Tree

Hoeffding Tree is a decision tree analyse procedure that can learn from enormous data streams at any moment. The examples are supposed to remain constant. Hoeffding trees have a benefit where small sample size is typically sufficient for determining the best split attribute. The reinforcement limitation establishes the minimum number of measurements required to determine a given statistic of specified accuracy which supports this idea mathematically.

Decision Stump

Decision stump is one-level decision tree-based machine learning model. It's a decision tree of one interior node linked to the end nodes immediately. A decision stump produces theforecast on basis of a single input feature's value. They're sometimes referredtoas1-rules.

Several variants are conceivable which depends upon the kind of input characteristics. In order to represent nominal attributes, either construct a stump with leaf for every feasible feature data or a stump of two leaves, one represents a certain category while others represent all other categories. These methods have similarities in terms of binary characteristics. A missing value might be considered a separate category.

In case of continual features, a feature threshold data value is commonly chosen&the stump has two leaves, one for values lower than the threshold and one for values higher than it. In exceptional cases, multiple thresholds may be selected, resulting in a stump with three or more leaves. Table 2 shows the comparison of various decision tree-based algorithms.

Flexible Structure RBFNN

To begin, the average firing rate of a neuron is utilized to decide whether or not additional neurons should be added. The FS-firing RBFNN's rate is very similar to the pre-synaptic neuron's spiking frequency in the biological neural system. Whenever the predetermined limit of the hidden neurons' rate value is reached new neurons get added into hidden layer. Second, an information-theoretic technique is used to estimate the connectivity of hidden neurons. The mutual information (MI) in the training process is used to determine the connection between hidden neurons.

The FS-RBFNN employs a dynamic tuning method to automatically construct the topology of the RBF neural network. During the training phase, this method modifies the architecture of the neurons' average firing rate and the RBF neural network by monitoring the MI. In the RBF neural network training process, the FS-RBFNN additionally utilizes an online learning method to link the weights.

Table 2: Comparison of decision tree algorithms

Decision Tree	*Correctness (%)*	*Time to construct models (second)*
LMT	89.89	.06
Decision Stump	83.4	0.0
Hoeffding Tree	80.69	.03
Random Forest	91.7	.05
J48	93.6	0.0

The mean firing rate is used to evaluate the buried neuron's activity. Hidden neurons with a high firing rate are split up and replaced with newer neurons. The MI value is utilized to modify the network structure, i.e., the MI value is used as a measure of connectivity among the concealed and output layer; connections with a low MI value will be cut down to simplify the RBFN structure. Finally, the exactitude of the FS-RBFNN is ensured by the gradient-descent technique, which is utilized to modify the values of the parameters.

The RBFNN structure was developed using a flexible method, which included the neuron adjusting mechanism and the neuron splitting mechanism. Third, most self-organizing approaches for RBF structure creation rely on trials to determine convergence for the learning process. The FS-RBFNN training procedure must be convergent in order to be effective in practice. The FS-RBFNN was created particularly for this purpose.

Fourth, after the structure has been adjusted by growing and pruning, the FS-RBFNN is mainly concerned with reducing retraining epochs. The error required (ER) approach is used to calculate the starting values of the neurons inserted into the FS-RBFNN. An RBF structure is modified when a design approach adds (or removes) hidden neurons, and the modified structure is retrained to adjust its connection weights.

Hybrid Decision Tree

XGBoost and RF are hybrid models that integrate the CEEMDAN approach with the original decision tree-based machine learning models (Random Forest). As a result, their prediction accuracy has improved.

XG Boost

By iterating and creating many trees, XGBoost (eXtreme Gradient Boosting) may combine several feeble learning machines into a sturdy learning machine, and it has the following features:

- Important feature of XGBoost is they can repeatedly exploit the CPU's multithreading for parallelism while upgrading the algorithm to enhance precision.
- It is an automatic sparse data processing enhancing learning technique supported by the decision tree model.

Since it is challenging to learn all of the tree requirements at one time, XGBoost utilizes an additional technique that teaches one tree's parameters at a time. The stepwise forward additive model is used by XGBoost algorithm as a gradient-boosting technique. In contrast, the gradient boosting approach is a harmful gradient that trains a fragile beginner to estimate the loss function. The XGBoost approach first determines the second-order Taylor approximation of the loss function in this point. This technique then minimizes the approximation loss function, which is in-turn used to educate the fragile beginner.

Ceemdan

Data may include huge swings in the time series and has an elevated level of non-linearity. This will definitely make the prediction more challenging.

CEEMDAN (Complete Ensemble Empirical Mode Decomposition with Adaptive Noise) is employed for data decomposition and denoising, which decomposes untreated information with significant fluctuations into numerous datasets with smaller variations. The research shows that the stability of CEEMDAN-RF and CEEMDAN - XGBoost is comparatively greater than that of existing models.

Deep Belief Network

In deep learning, DBN is considered as crucial model. A Restricted Boltzmann Machine (RBM) unit sequence makes up this probabilistic generative model. Every neural unit in the RBM model's layers has no link, and each visible layer neural unit is coupled to every hidden layer neural unit. In addition, each RBM layer's output is utilized as the input for the next layer.

A multi-level RBM design is employed in the DBN model's base level. The greedy strategy is employed to prepare each layer of the trial information. The first level RBM's parameters are sent into the second-level RBM, and the parameters of all the layers are determined by analogy. Unsupervised learning involves this training process as in Figure 4.

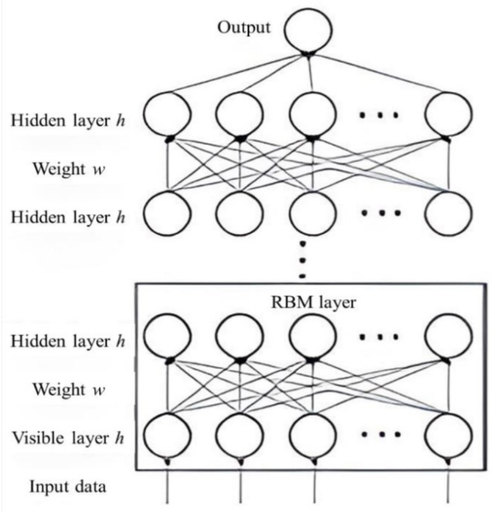

Figure 4: Deep belief network model architecture

The DBN network is utilized to obtain the key components of the cross-sectional water value. This is used to mine the significant water quality characteristics. The LSSVR layer is utilized to optimize the forecast outcome at the top level of the model, after which the forecast outcomes are sent via the LSSVR level fitting. The input to the LSSVR level is the conceptual texture supplied by the bottom model's preparation and erudition. But the LSSVR layer is also required to optimize the acquired model parameters.

Particle Swarm Optimization

This method of computing uses evolutionary algorithms. The name originates from the algorithm using a population (swarm) of feasible solutions (particles). The particle swarm technique produces an arbitrary solution before finding the best match solution repeatedly. Termination can be done based on iteration count or when the solution meets the required objective. Because of its benefits of ease of implementation, fewer setup parameters, and short convergence time, among other things, this algorithm is used in the field of optimization techniques.

The particle swarm optimization method, in its most basic version, consists of as warm of particles, each of which chooses its flight path depending on the value along with rate of an adaptive task, steadily migrating to a enhanced location, and eventually looking for the overall best result. In this process the particles are stimulated approximately in hunt of space based on a easy mathematical formula by changing both particles' position and velocity. Position of particle correlates to neural network's weight value and shows a potential solution to the problem being solved.

PSO is a metaheuristic because it makes minimal, if any, presumptions regarding the problem to be solved and may look for a vast array of potential answers. PSO also doesn't use the gradient of the problem being reduced, hence it doesn't need the reduced setback to be differentiable, and as traditional optimization methods like gradient descent and quasi-newton technique do. However, meta heuristics like PSO will not ensure the detection of an best answer always.

Least Squares Support Vector Regression Machine

The structural risk reduction criterion and statistical theory in the least squares support vector regression machine (LSSVRM), a machine learning approach. Unlike the support vector regression machine, the LSSVR transforms quadratic program tribulations into linear equations, replacing dissimilarity restrictions with similarity restrictions by loss function and error square as the main loss in the training set. This effectively increases calculation speed and accuracy while also having a good promotion performance. The LSSVR model is trained utilizing input and label data to update model parameters in a supervised way. Deep learning is used as a pre-processing system for LSSVR machine in the prediction model.

PSO Optimized DBN Network and LSSVR Model

A method based on a DBN model for forecasting water quality. DBN is used to remove attribute vectors from water quality time series data in various ranges, uses network parameters that are first optimized using the PSO approach. Then, the LSSVR machine is integrated with the PSO-DBN-LSSVR water quality forecast method as the top prediction layer.

The amount of input characteristics determines the amount of observable level neurons, while the amount of concealed level neurons, along with each layer's weights and thresholds, is determined layer by layer in the training RBM. After pre-training every level of RBM, use the amount produced from each bottom level RBM as the contribution for the upper level RBM, and then trains the upper level RBM. Feature extraction and dimension reduction will be applied before the data is transformed into a feature vector. Utilizing the PSO optimization approach to energetically optimize and update all RBM method parameters and determine the ideal starting weight, it is possible to defeat the challenge that the DBN network can easily drop into narrow optimum throughout the learning and training procedure.

The LSSVR model's parameters have been established. Output in the top layer RBM is utilised as the key of the LSSVR regression layer to train the form. When the highest amount of turns or fault is below the designated threshold, the LSSVR model training is finished, and the LSSVR forecast model is built using the best possible grouping of parameters. Every level of RBM system can check that the weights in its level are optimum for the feature vector mapping of that level after the LSSVR model has been trained, not for the whole DBN and LSSVR combined model. As a result, until the

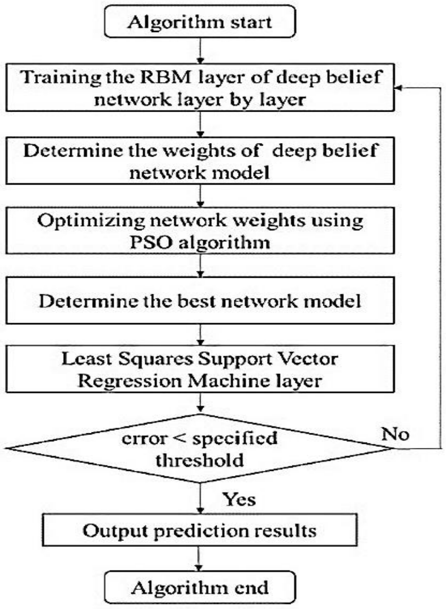
Algorithm start
Training the RBM layer of deep belief network layer by layer
Determine the weights of deep belief network model
Optimizing network weights using PSO algorithm
Determine the best network model
Least Squares Support Vector Regression Machine layer
error < specified threshold
No
Yes
Output prediction results
Algorithm end

Figure 5: Flowchart for PSO optimized DBN network & LSSVR algorithm

model converges and the training is done, the upper layer LSSVR model must propagate from top to bottom through every level of RBM, iteratively updating the fine-tuned DBN network's weights and offsets. Figure 5 shows the flow of algorithm.

Water temperature (T, C), pH, dissolved oxygen (DO, mg/L), conductivity (S/cm), turbidity (NTU), potassium permanganate index (COD, mg/L), total phosphorus (TP, mg/L), and ammonia nitrogen (NH4N, mg/L) were the nine chemical parameters that were looked at. These measures must effectively assess water quality and closely relate to water quality forecasting. The first eight chemical components were the typical key factors, and total nitrogen was the characteristic output factor.

Conclusion

Normally water quality is measured real-time, using various methods. The pollution is increasing in several folds and there is not much use in just knowing the fact that water body is affected. If we are able to predict a worst condition early, we naturally will take the necessary steps to prevent it from happening. If we picture the future with current levels of water pollution, we will try our best to prevent its occurrence. The performance of several water quality prediction algorithms is addressed. The accuracy ranges substantially depending on the number of parameters used and which are considered. With the enhancement of knowledge on data analytics, deep learning techniques and IoT infrastructures, real-time water quality is monitoring and assessment will been forced in the future world. This document reflects our recent literature survey, reviewing and comparing current work on water quality assessment based on big data technologies, deep learning techniques and machine learning models. This article has highlighted existing techniques that could be used to predict water quality.

References

Balajee, J., & Durai, M. S. (2021, July). Smart survey on recent trends in water level, drought and water quality analysis system. In Journal of Physics: Conference Series (Vol. 1964, No. 4, p. 042052). IOP Publishing.

Balan, N., & Ila, V. (2022). A Novel Biometric Key Security System with Clustering and Convolutional Neural Network for WSN. Tehnički vjesnik, 29(5), 1483-1490.

Bouamar, M., & Ladjal, M. (2008, July). A comparative study of RBF neural network and SVM classification techniques performed on real data for drinking water quality. In 2008 5th International Multi-Conference on Systems, Signals and Devices (pp. 1-5). IEEE.

Gakii, C., & Jepkoech, J. (2019). A classification model for water quality analysis using decision tree.

Han, H. G., Chen, Q. L., & Qiao, J. F. (2011). An efficient self-organizing RBF neural network for water quality prediction. Neural networks, 24(7), 717-725.

Heddam, S. (2016). Simultaneous modelling and forecasting of hourly dissolved oxygen concentration (DO) using radial basis function neural network (RBFNN) based approach: a case study from the Klamath River, Oregon, USA. Modeling Earth Systems and Environment, 2(3), 135.

Jaloree, S., Rajput, A., & Gour, S. (2014). Decision tree approach to build a model for water quality. Binary Journal of Data Mining & Networking, 4(1), 25-28.

Kang, G., Gao, J. Z., & Xie, G. (2017, April). Data-driven water quality analysis and prediction: A survey. In 2017 IEEE Third International Conference on Big Data Computing Service and Applications (BigDataService) (pp. 224-232). IEEE.

Liu, J., Yu, C., Hu, Z., Zhao, Y., Bai, Y., Xie, M., & Luo, J. (2020). Accurate prediction scheme of water quality in smart mariculture with deep Bi-S-SRU learning network. IEEE Access, 8, 24784-24798.

Lu, H., & Ma, X. (2020). Hybrid decision tree-based machine learning models for short-term water quality prediction. Chemosphere, 249, 126169.

Nivedetha, B., & Vennila, I. (2020). FFBKS: Fuzzy fingerprint biometric key based security schema for wireless sensor networks. Computer Communications, 150, 94-102.

Wang, Y., Zhou, J., Chen, K., Wang, Y., & Liu, L. (2017, November). Water quality prediction method based on LSTM neural network. In 2017 12th international conference on intelligent systems and knowledge engineering (ISKE) (pp. 1-5). IEEE.

Yan, J., Gao, Y., Yu, Y., Xu, H., & Xu, Z. (2020). A prediction model based on deep belief network and least squares SVR applied to cross-section water quality. Water, 12(7), 1929.

Yan, J., Xu, Z., Yu, Y., Xu, H., & Gao, K. (2019). Application of a hybrid optimized BP network model to estimate water quality parameters of Beihai Lake in Beijing. Applied Sciences, 9(9), 1863.