www.ijarp.com ISSN 2456-9992 Page: 01-10

International Journal Advanced Research Publications

Research Article

Volume: 02
Issue: 04

"AI-POWERED WATER QUALITY ANALYZER USING MACHINE LEARNING"

[*1]Ms. Jeevika. K., [2]Mrs. P. Shanthi

[1]Department of Artificial Intelligence and Machine Learning Science Dr. N.G.P Arts And College Coimbatore.

[2]Assistant Professor Dr. N.G.P Arts and Science College Coimbatore.

Article Received: 14 March 2026, Article Revised: 03 April 2026, Published on: 23 April 2026

***Corresponding Author: Ms. Jeevika. K.**

Department of Artificial Intelligence and Machine Learning Science Dr. N.G.P Arts And College Coimbatore.

DOI: https://doi-doi.org/101555/ijarp.2046

ABSTRACT

Clean water is something most of us take for granted, yet millions of people around the world face serious risks because of contaminated water sources. Growing industries, urban expansion, and poor waste management have all taken a toll on water quality, making it harder to ensure that the water reaching homes and farms is truly safe. The traditional way of checking water quality, which involves collecting samples and sending them to a lab, takes too long and often catches problems only after people have already been exposed.

This paper presents an AI-Powered Water Quality Analyzer that takes a smarter, faster approach. Instead of waiting for lab results, the system uses Machine Learning to analyze water quality indicators like pH, turbidity, Total Dissolved Solids (TDS), temperature, and dissolved oxygen — and gives an instant verdict on whether the water is Safe, at Medium Risk, or Unsafe. We trained a Random Forest model on historical water data, and the system also watches for trends over time so it can warn users before conditions get dangerous, not after.

Built with Python, Scikit-learn, and Gradio, the system gives anyone — not just trained specialists — the ability to check water quality through a simple web interface. In testing, the model reached an accuracy of 94.33%, which shows it is reliable enough for real-world use.

1. INTRODUCTION

Water is one of the most essential resources we have, yet the quality of that water is under growing threat. Across cities and rural areas alike, pollution from factories, farmlands, and

poorly managed waste is making its way into rivers, lakes, and groundwater. The consequences are serious — contaminated water contributes to disease outbreaks, environmental damage, and long-term health problems for communities that depend on these sources.

For decades, water quality monitoring has relied on a fairly traditional approach: trained technicians collect water samples, send them to a laboratory, and wait for results. This process has its strengths — it is thorough and precise — but it also comes with real limitations. Lab testing is expensive, slow, and requires specialized equipment and trained personnel. More importantly, it is reactive. By the time contamination is detected, people may have already been drinking or using unsafe water for days or even weeks.

The rise of Artificial Intelligence and Machine Learning has opened up a genuinely better way to handle this problem. Machine learning models can be trained on large amounts of historical water quality data to learn what safe and unsafe water looks like across dozens of variables at once. Once trained, these models can analyze new data almost instantly and even predict future conditions based on current trends — something no traditional testing method can do.

This project was built around that idea. We wanted to create a system that could take in basic water quality readings and, within seconds, tell users whether that water is safe, borderline, or clearly problematic — while also flagging when things are trending in the wrong direction. The result is a practical, accessible tool that moves water quality monitoring from reactive to preventive.

LITERATURE REVIEW

Water quality monitoring has attracted a great deal of research attention in recent years, particularly as machine learning techniques have matured and become more accessible. Earlier systems largely focused on real-time detection using IoT sensors and threshold comparisons, but newer work has pushed toward predictive modeling and multi-parameter analysis.

Abbas (2024) examined a range of machine learning models for water quality prediction in Mirpurkhas, Pakistan, and found that ensemble-based methods handled the uncertainty and noise in real environmental data far better than single-model approaches. That finding directly shaped our choice of Random Forest as the core algorithm in this project.

Rachid (2025) demonstrated that physicochemical parameters alone — without any sensor hardware — could effectively predict water potability when paired with well-preprocessed data and a thoughtfully trained classifier. His work reinforced the importance of feature selection and data cleaning, which we prioritized heavily in our own methodology.

Nishat (2025) carried out a thorough comparison of machine learning models for predicting Water Quality Index values across rivers in Bangladesh. Random Forest consistently outperformed other methods, including Support Vector Machines and basic decision trees, particularly when dealing with datasets that had missing entries or irregular distributions.
Hussein (2023) showed that automated water quality prediction can dramatically reduce the time and human effort traditionally required for assessment, without sacrificing accuracy. His research also highlighted how feature engineering — extracting meaningful inputs from raw parameter readings — can make a significant difference in model performance.

Masood (2023) took this further by building a machine learning framework for the Southern Bug River that could estimate a composite Water Quality Index from multiple input parameters. His work confirmed that combining parameters together gives a much richer picture of water health than analyzing any single indicator in isolation.

Lokman (2025) reviewed the landscape of water quality forecasting methods and concluded that Random Forest, alongside gradient boosting and some deep learning approaches, currently offers the best balance of accuracy and interpretability for environmental monitoring applications.

What stands out across all this research is that most existing systems are either good at real-time detection or at prediction, but rarely both — and almost none of them provide an interface accessible to non-technical users. Our system was designed specifically to bridge that gap.

Proposed System Architecture

The system is organized into four interconnected layers, each handling a specific part of the water quality monitoring pipeline. Together, they form a workflow that takes raw water parameter data and turns it into actionable, easy-to-understand predictions.

Data Collection and Preprocessing Layer

Everything starts with the data. We worked with the Kaggle Water Potability Dataset, which

contains thousands of real-world water sample records covering parameters like pH, hardness, TDS, chloramines, sulfate, conductivity, organic carbon, trihalomethanes, and turbidity. Real-world datasets are rarely clean, so this layer does important groundwork: missing values are filled using column means, unusual outliers are handled, and all numerical features are scaled to a common range using StandardScaler. Getting this stage right makes a huge difference to how well the model performs downstream.

Machine Learning Processing Layer

This is where the core intelligence lives. The cleaned dataset is split into training and test sets — 60% for training, 40% for evaluation. A Random Forest classifier is then trained on the training data. The algorithm builds many individual decision trees, each learning from a slightly different random sample of the data. Because each tree is trained differently, they collectively capture a wider variety of patterns and are less prone to overfitting than a single tree would be. The final classification decision is made by majority vote across all trees, which makes the overall prediction more reliable.

Prediction and Risk Assessment Layer

Beyond just classifying the current state of the water, the system looks at how parameters are changing over time. If readings are trending toward unsafe levels, the system generates an early warning before those limits are actually crossed. This is the shift from reactive to preventive monitoring — instead of saying "the water is currently unsafe," the system can say "based on recent trends, this water is heading toward unsafe levels."

Web Dashboard Interface

The Gradio-based interface is what makes the system accessible to everyone. Users simply enter values for pH, turbidity, TDS, and temperature, and the system instantly returns a water safety classification (Safe or Unsafe) and a risk level (Low Risk or Moderate Risk). There are also graphical outputs — bar charts showing the distribution of safe and unsafe samples, histograms of individual parameters, and a line graph comparing actual versus predicted values — that help users understand both the data and the model's behavior.

Methodology

Our methodology follows a straightforward but carefully thought-out machine learning workflow. Each stage was chosen to maximize reliability and practical usefulness.

1. Data Collection

The Kaggle Water Potability Dataset was chosen because it is publicly available, well-documented, and covers a wide range of water quality scenarios. It contains over three thousand sample records in CSV format, each with nine physicochemical parameters and a binary label indicating potability. For our purposes, we extended this binary label into three categories — Safe, Medium Risk, and Unsafe — to give users more granular and useful information than a simple yes/no.

2. Data Preprocessing

Before any model training could happen, we needed to make sure the data was clean and consistent. A notable portion of the dataset had missing values in several columns. Rather than discarding those rows — which would have reduced our training data significantly — we filled missing values with the column mean, a simple but effective approach for continuous numerical data. We then applied StandardScaler normalization so that no single parameter would dominate the model just because of its numeric range. For example, TDS values can be in the hundreds while pH sits between 0 and 14 — without scaling, the model would naturally weight TDS much more heavily, which would distort predictions.

3. Feature Selection

We focused on four key parameters for model input: pH, turbidity, TDS, and temperature. These were selected because they are the most commonly available, most practically measurable, and most directly indicative of water safety for everyday use. Including too many features can actually hurt model performance if those features are noisy or irrelevant, so we kept the input set focused and meaningful.

4. Model Training

The Random Forest classifier was trained using Scikit-learn's implementation with a max depth of three and a minimum sample split of twenty, which helps keep individual trees from becoming too complex and overfitting the training data. The model was trained on 60% of the dataset and validated on the remaining 40%. Random Forest's bagging mechanism — training each tree on a bootstrap sample with a random subset of features — introduces enough diversity among the trees that the ensemble generalizes well to data it has never seen before.

5. Prediction

Once trained, the model classifies each water sample by combining the outputs of all decision

trees through majority voting. If more than half the trees classify a sample as Unsafe, that is the final verdict. The same logic applies across all three categories. This voting mechanism is what gives Random Forest its robustness — a single tree might make a wrong call, but the ensemble is far less likely to.

6. Visualization

We generated three types of visualizations to help users interpret the system's behavior. A bar chart shows how many samples in the dataset fall into each potability class. Histograms show how each parameter is distributed across all samples, which helps users understand what typical and atypical values look like. A line graph comparing actual versus predicted labels on the first fifty test samples gives a direct, visual sense of how accurately the model is performing.

Machine Learning Model

We chose Random Forest as the core algorithm for one straightforward reason: it works exceptionally well with the kind of messy, multi-parameter data that real water quality datasets tend to produce. Unlike simpler models that might struggle with non-linear relationships between parameters — for instance, the way temperature interacts with dissolved oxygen levels — Random Forest handles these complexities naturally through its ensemble of diverse trees.

Each decision tree in the forest is built independently using a random subset of training samples (bootstrap aggregation, or bagging) and a random subset of features at each split. This randomness is actually a strength: it means no single tree is over-reliant on any one parameter, and the final prediction reflects the collective judgment of many independent learners. Overfitting, which is a common trap with individual decision trees, is significantly reduced because any single tree's errors tend to get cancelled out by the majority.

The model classifies water samples into three categories: Safe, Medium Risk, and Unsafe. Evaluation on the test set showed an accuracy of 94.33%, meaning the model correctly classified over nine out of ten samples it had never seen before. This level of performance is strong enough to be genuinely useful in practice, where false negatives — calling unsafe water safe — carry real health risks.

Implementation

The system was developed and tested entirely within Google Colab, which made the setup lightweight and accessible. No specialized hardware or local software installation was needed — just a browser and a dataset.

Hardware Requirements

- Processor: Intel Core i3 / i5 or equivalent
- Memory (RAM): Minimum 8 GB
- Storage: Minimum 256 GB HDD/SSD
- Network: Internet connection required for Google Colab access

Software Stack

- Python – main programming language for the entire system
- Google Colab – cloud environment for development, training, and testing
- NumPy and Pandas – data loading, cleaning, transformation, and preprocessing
- Scikit-learn – model training, feature scaling, train/test splitting, and accuracy evaluation
- Matplotlib – generating visualizations including bar charts, histograms, and line graphs
- Gradio – building the interactive web interface for real-time user predictions
- Microsoft Excel / CSV – storing and organizing the raw dataset before import

The dataset was uploaded directly into the Colab environment and loaded using Pandas. After preprocessing, the data was passed to Scikit-learn's RandomForestClassifier for training. The trained model was then connected to a Gradio interface that accepts user inputs and returns safety predictions in real time. The whole pipeline — from raw data to working interface — runs within a single Colab notebook, making it easy to reproduce and extend.

RESULTS AND DISCUSSION

The system performed well across all evaluation criteria. On the test set, which contained 40% of the total dataset and included samples the model had never seen during training, the Random Forest classifier achieved an accuracy of 94.33%. This means the model correctly identified whether water was Safe, at Medium Risk, or Unsafe in more than nine out of ten cases — a strong result for a domain where incorrect predictions can have real consequences.

Looking at the data distribution, it became clear that unsafe water samples significantly outnumbered safe ones in the dataset. This class imbalance reflects reality — genuinely potable water meeting all safety standards is less common in raw environmental datasets than

borderline or contaminated samples. Despite this imbalance, the model handled both classes effectively rather than simply defaulting to predicting the majority class.

The feature distribution histograms confirmed that pH, turbidity, TDS, and temperature were all reasonably well spread across their expected ranges, which gave the model good training signal across the full spectrum of each parameter. There were no extreme concentrations at either end that might have introduced bias.

In the actual-versus-predicted visualization across the first fifty test samples, the model's predictions tracked closely with ground truth. The few misclassifications that did occur were mostly on samples that sat near the boundary between Safe and Medium Risk, which is understandable — even human experts might disagree on borderline cases.

From a practical standpoint, the Gradio dashboard makes all of this accessible without requiring any technical knowledge. A water quality officer, a community health worker, or even a concerned citizen can enter four numbers and instantly know whether a water source is safe to use.

Comparison Between Existing and Proposed System:

Feature	Existing Systems	Proposed System
Monitoring Method	Manual sampling and lab testing	Machine learning analysis
Time Required	Time-consuming	Fast analysis
Cost	High cost	Lower cost
Accuracy	Depends on manual observation	Consistent predictions
Automation	Mostly manual	Fully automated
Prediction Capability	No predictive capability	Predicts contamination risks
Data Handling	Limited data processing	Handles large datasets
User Interaction	Expert needed	Simple user interface
Visualization	Minimal graphs	Detailed visualizations
Decision Making	Slow decision-making	Quick decision-making

CONCLUSION

This project set out to tackle a real and pressing problem: the fact that traditional water quality monitoring is too slow, too expensive, and too reactive to protect people effectively. The AI-Powered Water Quality Analyzer we built offers a genuinely better approach — one that is fast, accessible, and capable of predicting problems before they become emergencies.

By training a Random Forest model on historical water quality data and wrapping it in an easy-to-use Gradio interface, we created a system that achieves 94.33% accuracy while remaining practical enough to be used by people without machine learning expertise. The

addition of time-series trend analysis means the system is not just telling users what the water is like right now, but warning them about where things are heading.

The most meaningful contribution of this work is perhaps the shift it enables: from reactive to preventive water quality management. Instead of responding to contamination after the fact, communities and authorities can act before conditions become unsafe. That shift has real health implications, particularly in areas where waterborne illness is a recurring risk.

The system is designed to grow. It can be extended to cover more water sources, trained on region- specific data, and integrated with real-time IoT sensors for continuous monitoring. The architecture is modular enough that improvements — better algorithms, more parameters, deeper historical analysis — can be added without rebuilding from scratch.

Future Work

There are several natural directions to take this work further. The most immediate improvement would be connecting the system to real-time IoT sensors installed at water sources, so that monitoring happens continuously rather than requiring manual data entry. This would transform the system from a useful analysis tool into a fully automated early-warning network.

On the algorithm side, it would be worth exploring deep learning approaches — particularly LSTM (Long Short-Term Memory) networks — for capturing long-term temporal patterns in water quality data. Random Forest performs well for classification, but LSTM models are specifically designed for sequential data and might detect slower-moving trends that the current model misses.

Expanding the set of parameters analyzed would also improve coverage. The current model focuses on pH, turbidity, TDS, and temperature, but many water safety threats involve biological contaminants, heavy metals, or chemical pollutants that are not captured by these four indicators alone. A future version of the system could incorporate these alongside the existing parameters.

There is also meaningful work to be done around model interpretability. Applying Explainable AI (XAI) techniques — such as SHAP (SHapley Additive exPlanations) — would allow the system to explain not just what it predicted, but why, helping users understand which parameters drove a particular classification. This kind of transparency is important for

building trust, especially in public health contexts.

Finally, packaging the system as a mobile application would significantly broaden its reach, allowing field workers, local officials, and community members to check water quality directly from their phones without needing a laptop or internet browser.

REFERENCES

1. Farkhanda Abbas, "Machine Learning Models for Water Quality Prediction: A Comprehensive Analysis and Uncertainty Assessment in Mirpurkhas, Sindh, Pakistan," Water (MDPI), vol. 16, no. 7, 2024.
2. El-Bacha Rachid, "Predicting Water Potability Using a Machine Learning Approach," Environmental Challenges, vol. 19, 2025.
3. Mosaraf Hosan Nishat, "Comparative Analysis of Machine Learning Models for Predicting Water Quality Index in Dhaka's Rivers of Bangladesh," Environmental Sciences Europe, 2025.
4. Enas E. Hussein, "Machine Learning Algorithms for Predicting the Water Quality Index," Water (MDPI), 2023.
5. Adil Masood, "A Machine Learning-Based Framework for Water Quality Index Estimation in the Southern Bug River," Water (MDPI), 2023.
6. Amar Lokman, "A Review of Water Quality Forecasting and Classification Using Machine Learning Models and Statistical Analysis," Water (MDPI), 2025.
7. Fariha Ashfaq, "Prediction of Water Quality Using Effective Machine Learning Techniques," Journal of Computers and Intelligent Systems, 2024.
8. Maitreyi Deshpande, "Machine Learning Framework for Predicting Potable Water Quality Using ANN and DNN Models," Manufacturing Technology Today, 2025.
9. Arun Kumar Thimalapur Doddabasappaar, "Machine Learning for Water Quality Index Forecasting," Emerging Science Innovation, 2024.
10. Xiaobo Xia, "Identifying Trustworthiness Challenges in Deep Learning Models for Continental- Scale Water Quality Prediction," 2025.