International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 9 Issue 9 September - 2025, Pages: 271-280

Integrating Artificial Intelligence and IoT for Real-Time Water Quality Prediction and Early Pollution Warning

[1]Ho Hien Vinh, [2]Le Khanh Hai Dang

[1]Hanoi Amsterdam Highschool for gifted, Viet Nam. Email: hohienvinhams2019@gmail.com

[2]Software Engineer, Viet Nam. Email: haidang9ahv1@gmail.com

Abstract: *This paper examines the development of an innovative Artificial Intelligence (AI) and Internet of Things (IoT) solution for monitoring real-time river pollution. As the world's water bodies become more polluted, the gravity of the situation deepens. The framework employs IoT devices that track specific water quality parameters (pH, temperature, turbidity, dissolved oxygen, total dissolved solids, and more) and relays the data to a processing unit. Within the processing unit, AI prediction models are formed that perform anomaly detection, pollution prediction, and classification for alerts. After training on historical data, the machine learning model can predict pollution instances much more rapidly than the thresholds. Preliminary findings suggest that the model's predictive capabilities surpass a 90% accuracy level and can thus be utilized to visualize reports. The data can be transformed into actionable insight for the web and mobile dashboards. In addition, the model can also be deployed to set off alarms for the concerned authorities. The adaptable model can also be applied to several other sites, including but not limited to, industrial zones and residential sites adjoining the river, thus aiding in the development of IoT driven environmental monitoring and achieving sustainable development goals for the emerging economies like Vietnam.*

Keywords: Water pollution monitoring, Artificial Intelligence (AI), Internet of Things (IoT), Early pollution prediction, Machine learning, Sustainable development, Water resources.

1. Introduction

Water contamination is one of the most pressing issues in the environment on a global scale and is more severe in developing countries. Rivers and lakes, the arteries of the ecosystem and important daily life and productive water sources, are increasingly polluted by toxic elements. It not only poses a serious threat to public health but also leads to unpredictable consequences for biodiversity and the natural equilibrium of aquatic ecosystems. Under this background, the rapid development of water pollution monitoring and early warning systems is a very important thing to do. To address this requirement, the study provides an innovative solution that forecasts the quality of river water in real time by fusing Artificial Intelligence (AI) and Internet of Things (IoT) technologies. The system is configured to automatically secure and collect water environment data at all.

2. Theoretical overview

This research delves into the intersection of environmental science and artificial intelligence as well as geo-spatial analysis in the construction of a real-time predictive model for water pollution. The foundation of the model involves parameters such as pH level, temperature, turbidity, the amount of dissolved oxygen, total dissolved solids, heavy metals, and coliforms. AI serves as the model's anomaly detection mechanism. L ATA based, supervised, unsupervised and time series models are trained with historical and live sensors data. The use of GIS based geo-spatial analysis compliments the model through mapping of the sources of contamination, pathways of spread and the target areas.

The architecture of the system is a combination of real time data insights in addition to the data pertaining to sensors and the historical data which is generated through python-based data processing and geo processing tools on data of the interactive interface. The system also delivers mobile and web dashboards along with spatial risk maps. The real-time dashboard presents alerts along with visual instructions. Some of the major elements are uninterrupted sensor analysis, automatic SMS and email notifications, real-time time-series data, automatable predictive reporting, geospatial risk mapping, device reliability monitoring, iterative model retraining and smoothed time series visualization.

Relevant factors of the system being the environmental managers, scientists, local authorities and citizens are presented with timely information and thus as a base, the system provides improved decision-making as well as risk acknowledgement in spatial and geo areas repeatedly over time. Fulfilling the target goals of the users, the system supports water quality management in a practical and scalable framework.

3. Methodology

3.1. System Architecture

This study creates a custom architecture combining Artificial Intelligence (AI) with the Internet of Things (IoT) for the purpose of predicting and monitoring pollution in freshwater in real time. The architecture works in five sequential steps; data capturing and harmonization; multi-source fusion; training of predictive models, and the generation of alerts. Sensor nodes with IoT capability are used on streams, reservoirs, and urban drainage systems to collect data on pH, turbidity, heavy metals, dissolved oxygen, total dissolved solids, and microbials. The data are sent and collected on a cloud-based ledger using wireless (LoRa, NB-IoT) and cellular (LTE) networks while automated processes eliminate noisy data, anomalies, unit discrepancies, and temporal misalignment. Multi-source fusion augments the streams with meteorological data and models, land cover information, elevation data, and historical water quality records. Engineering creates novel features of pollutants, sharp shifts in parameters, and bounds surpassing. Ensembles of supervised learning (Random Forest, Gradient Boosted Trees, LSTM networks) are used to train these features. Robust generalization across seasons and hydrological patterns is ensured via cross validation. The models output real time inferences which are displayed on risk heat maps, sent as geo-targeted notifications to mobile devices and on the web.

With the integration of Artificial Intelligence (AI) prediction algorithms paired with geospatially relevant Internet of Things (IoT) monitoring, the system predictive analyzes and spatially correlates data streams to monitor IoT geolocated signals in real time. This system also alerts first responders and the public as well as geo-specialized authorities. This ensures they are best prepared to handle interruptions of time-sensitive streams or resources. Thus, a greater stewardship paradigm for water resources is correctly applied. This stewardship is enhanced by the real-time public and geo-centered first responders' notifications. These notifications enable greater public awareness and active real-time interventions.

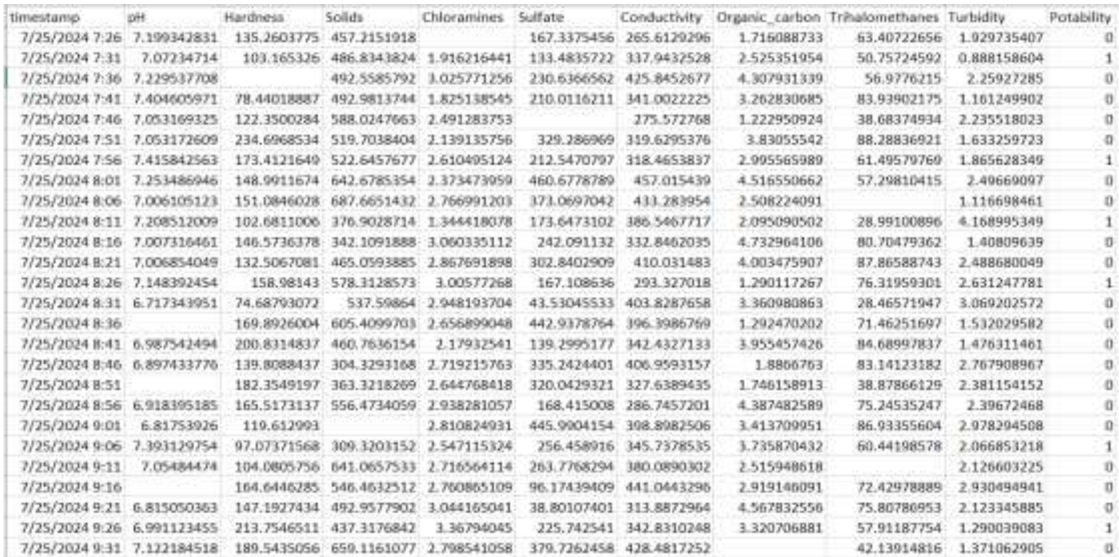
3.2. Data Pre-processing and Processing

Random Forest is an algorithm which deals with both classification and regression problems. Random Forest is a descendant of decision tree classifiers. Random Forest still relies on the old decision tree structures but constructs a multitude of pipes as a final model. Each of the pipes are classified as a single decision tree (aka weaker model) which are then connected into the final structure. Each tree structure relies on a sample of the original data set which is the base of a 'bootstrap sample' and relies on a 'subset of features' based on a random selection. Even if the model randomly sheds information, the classification is still correct. Each tree is aligned into a single classification with the highest count and classified as correct, or an average is taken. This model is very accurate even with lost data. It is very complex and resilient even with random data, scales very well, and can pinpoint important factors, like critical ways through which pollution is introduced to the monitored water. It is still algebraically complex and has low feasibility scores for edge partitions of data from systems in real time. Every model is evaluated based on the standard analytics, the precision and recall the model gets out of itself, and the ease of which it can construct the confusion matrix. To avoid overfitting and ensure robustness, k-fold cross validation are unbiased estimates of model performance across hydrological and seasonal conditions. Such attributes make Random Forest both powerful and practical, especially when it comes to real-time water quality forecasting in a fluid and uncertain environment.

4. Research processing

4.1. Description of data used for research

***Table 1**. Sample of the water quality dataset with measured parameters and potability classification.*



timestamp	pH	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
7/25/2024 7:26	7.199342831	135.2603775	457.2151918		167.3375456	265.6129296	1.716088733	63.40722656	1.929735407	0
7/25/2024 7:31	7.07234714	103.165326	486.8343824	1.916216441	133.4835722	337.9432528	2.525351954	50.75724592	0.888158604	1
7/25/2024 7:36	7.229537708		492.5585792	3.025771256	230.6366562	425.8452677	4.307931339	56.9776215	2.25927285	0
7/25/2024 7:41	7.404605971	78.44018887	492.9813744	1.825138545	210.0116211	341.0022225	3.262830685	83.93902175	1.161249902	0
7/25/2024 7:46	7.053169325	122.3500284	588.0247663	2.491283753		275.572768	1.222950924	38.68374934	2.235518023	0
7/25/2024 7:51	7.053172609	234.6968534	519.7038404	2.139135756	329.286969	319.6295376	3.83055542	88.28836921	1.633259723	0
7/25/2024 7:56	7.415842563	173.4121649	522.6457677	2.610495124	212.5470797	318.4653837	2.995565989	61.49579769	1.865628349	1
7/25/2024 8:01	7.253486946	148.9911674	642.6785354	2.373473959	460.6778789	457.015439	4.516550662	57.29810415	2.49669097	0
7/25/2024 8:06	7.006105123	151.0846028	687.6651432	2.766991203	373.0697042	433.283954	2.508224091		1.116698461	0
7/25/2024 8:11	7.208512009	102.6811006	376.9028714	1.344418078	173.6473102	386.5467717	2.095090502	28.99100896	4.168995349	1
7/25/2024 8:16	7.007316461	146.5736378	342.1091888	3.060335112	242.091132	332.8462035	4.732964106	80.70479362	1.40809639	0
7/25/2024 8:21	7.006854049	132.5067081	465.0593885	2.867691898	302.8402909	410.031483	4.003475907	87.86588743	2.488680049	0
7/25/2024 8:26	7.148392454	158.98143	578.3128573	3.00577268	167.108636	293.327018	1.290117267	76.31959301	2.631247781	1
7/25/2024 8:31	6.717343951	74.68793072	537.59864	2.948193704	43.53045533	403.8287658	3.360980863	28.46571947	3.069202572	0
7/25/2024 8:36		169.8926004	605.4099703	2.656899048	442.9378764	396.3986769	1.292470202	71.46251697	1.532029582	0
7/25/2024 8:41	6.987542494	200.8314837	460.7636154	2.17932541	139.2995177	342.4327133	3.955457426	84.68997837	1.476311461	0
7/25/2024 8:46	6.897433776	139.8088437	304.3293168	2.719215763	335.2424401	406.9593157	1.8866763	83.14123182	2.767908967	0
7/25/2024 8:51		182.3549197	363.3218269	2.644768418	320.0429321	327.6389435	1.746158913	38.87866129	2.381154152	0
7/25/2024 8:56	6.918395185	165.5173137	556.4734059	2.938281057	168.415008	286.7457201	4.387482589	75.24535247	2.39672468	0
7/25/2024 9:01	6.81753926	119.612993		2.810824931	445.9904154	398.8982506	3.413709951	86.93355604	2.978294508	0
7/25/2024 9:06	7.393129754	97.07371568	309.3203152	2.547115324	256.458916	345.7378535	3.735870432	60.44198578	2.066853218	1
7/25/2024 9:11	7.05484474	104.0805756	641.0657533	2.716564114	263.7768294	380.0890302	2.515948618		2.126603225	0
7/25/2024 9:16		164.6446285	546.4632512	2.760865109	96.17439409	441.0443296	2.919146091	72.42978889	2.930494941	0
7/25/2024 9:21	6.815050363	147.1927434	492.9577902	3.044165041	38.80107401	313.8872964	4.567832556	75.80780953	2.123345885	0
7/25/2024 9:26	6.991123455	213.7546511	437.3176842	3.36794045	225.742541	342.8310248	3.320706881	57.91187754	1.290039083	1
7/25/2024 9:31	7.122184518	189.5435056	659.1161077	2.798541058	379.7262458	428.4817252		42.13914816	1.371062905	0

The dataset includes parameters important for the assessment of water quality. The Timestamp parameter captures date and time for measuring real time and time series data. The pH range (6.52–6.83) is within the World Health Organization (WHO) safe limits (6.5–8.5). Hardness, as defined, ranges from the geological sources of dissolved calcium and magnesium salts, while Solids represent inorganic salts and minerals, which WHO recommends < 500 mg/L (max 1000 mg/L). The maximum safe concentration of chloramine, a disinfectant, is 4 mg/L. The range of concentration of sulfate in natural waters varies from freshwater (3–30 mg/L) to seawater (~2700 mg/L). Conductivity, which measures the ability of water to pass electric current, also determines the concentration of ions in the water. Organic Carbon (OC) is defined as the quantity of carbon contained in the organic compound. The suggested concentration of OC in treated water by the US EPA is < 2 mg/L. Disinfection by product thrihalomethanes should also be below 80 ppm. The average turbidity of the samples in this study is 0.98 NTU. This is well below the WHO limit of 5.0 NTU. Finally, potability is a binary variable which represents the ability of water to be consumed without any health risk (1) or having health risks (0).

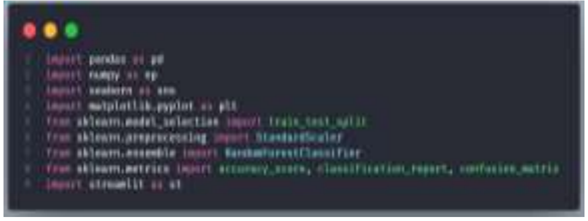
4.2. Tools and Technologies Used

A set of commonly known software programs and libraries was used to make the real-time water pollution monitoring system prototype more efficient and better positioned for scaling for mass usage. Python was geoTeaching in the primary integrated rapid and user interface wiysubsmoud ero NumPy and Pands used for data processing supporting model training and user interface components, geospatial frameworks and then powerful libraries worked for fast processing. Matplotlib integrated with array systems to plot structured and process datasets data and offered flexible processing with efficient data systems streams through Pandas. In Scikit learned data manipulation and evaluation frameworks McCullough zoom and Random with Random Forest precise controls of elements performance matrices training assessed feature scaling matrix of features with primary accuracy precision medical confusion evaluation assessed feature scaling primary evaluation metrics accuracy precision and medical confusion matrices. In the end, Streamlit was used to deploy real-time monitoring and data visualization systems which provided an interactive web interface for stakeholders easy primary access.

4.3. Research Procedure

First, we will list the necessary libraries that will be used in this project:

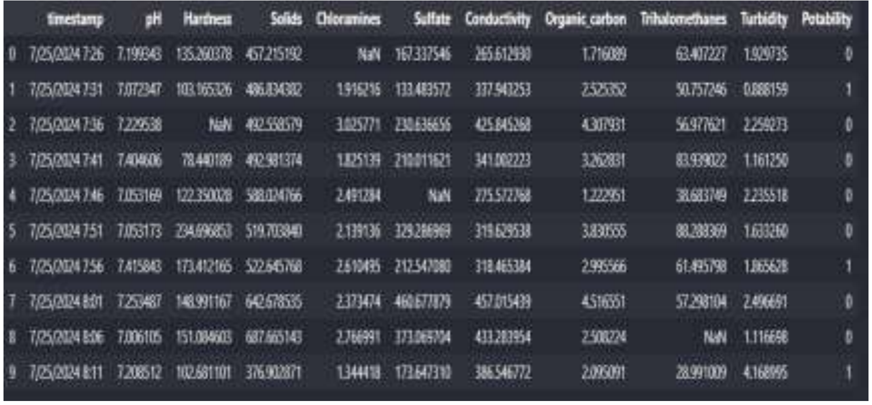
Listing 1. *Python code snippet showing library imports for data preprocessing and model building.*



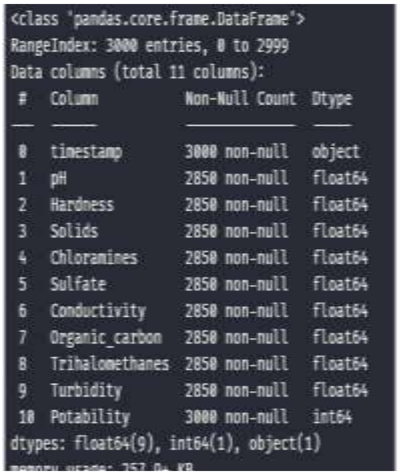
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import streamlit as st
```

Secondly, we load and display the dataset using pandas. The dataset consists of 3000 rows and 11 columns. Then we display the basic information and statistics of the data.

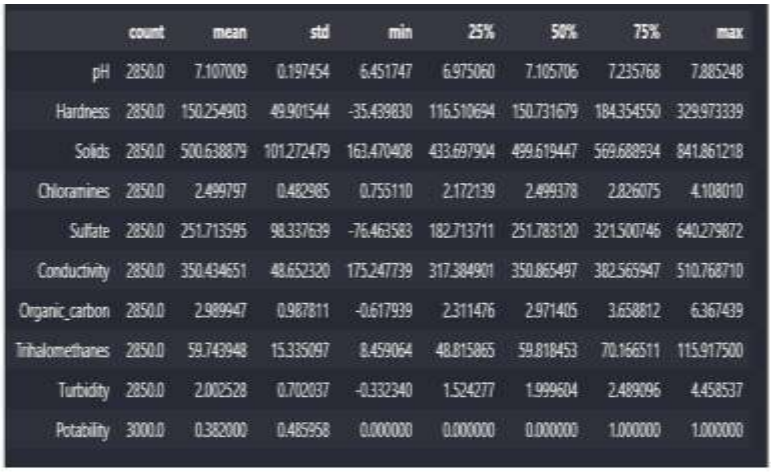
***Table 2**. Sample records from the water quality dataset.*



	timestamp	pH	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	7/25/2024 7:26	7.199343	135.260378	457.215192	NaN	167.337546	265.612930	1.716089	63.407227	1.929735	0
1	7/25/2024 7:31	7.072347	103.165326	486.834302	1.916216	133.483572	337.943253	2.525352	50.757246	0.888159	1
2	7/25/2024 7:36	7.229538	NaN	492.558579	3.025771	230.636656	425.845268	4.307931	56.977621	2.259273	0
3	7/25/2024 7:41	7.404606	78.440189	492.981374	1.825139	210.011621	341.002223	3.262831	83.939022	1.161250	0
4	7/25/2024 7:46	7.053169	122.350028	588.024766	2.491284	NaN	275.572768	1.222951	38.683749	2.235518	0
5	7/25/2024 7:51	7.053173	234.696853	519.703840	2.139136	329.286969	319.629538	3.830555	88.288369	1.633260	0
6	7/25/2024 7:56	7.415843	173.412165	522.645768	2.610495	212.547080	318.465384	2.995566	61.495798	1.865628	1
7	7/25/2024 8:01	7.253487	148.991167	642.678535	2.373474	460.677879	457.015439	4.516351	57.298104	2.496691	0
8	7/25/2024 8:06	7.006105	151.084603	687.665143	2.766991	373.069704	433.283954	2.508224	NaN	1.116698	0
9	7/25/2024 8:11	7.208512	102.681101	376.902871	1.344418	173.647310	386.546772	2.095091	28.991009	4.168995	1



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 11 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   timestamp        3000 non-null   object
 1   pH               2850 non-null   float64
 2   Hardness         2850 non-null   float64
 3   Solids           2850 non-null   float64
 4   Chloramines      2850 non-null   float64
 5   Sulfate          2850 non-null   float64
 6   Conductivity     2850 non-null   float64
 7   Organic_carbon   2850 non-null   float64
 8   Trihalomethanes  2850 non-null   float64
 9   Turbidity        2850 non-null   float64
 10  Potability       3000 non-null   int64
dtypes: float64(9), int64(1), object(1)
memory usage: 257.0+ KB
```



	count	mean	std	min	25%	50%	75%	max
pH	2850.0	7.107009	0.197454	6.451747	6.975060	7.105706	7.235768	7.885248
Hardness	2850.0	150.254903	49.901544	-35.439830	116.510694	150.731679	184.354550	329.973339
Solids	2850.0	500.638879	101.272479	163.470408	433.697904	499.619447	569.688934	841.861218
Chloramines	2850.0	2.499797	0.482985	0.755110	2.172139	2.499378	2.826075	4.108010
Sulfate	2850.0	251.713595	98.337639	-76.463583	182.713711	251.783120	321.500746	640.279872
Conductivity	2850.0	350.434651	48.652320	175.247739	317.384901	350.865497	382.565947	510.768710
Organic_carbon	2850.0	2.989947	0.987811	-0.617939	2.311476	2.971405	3.658812	6.367439
Trihalomethanes	2850.0	59.743948	15.335097	8.459064	48.815865	59.818453	70.166511	115.917500
Turbidity	2850.0	2.002528	0.702037	-0.332340	1.524277	1.999604	2.489096	4.458537
Potability	3000.0	0.382000	0.485958	0.000000	0.000000	0.000000	1.000000	1.000000

Listing 2. *Output of the dataset information using the panda's info() function.*

Listing 3. *Descriptive statistics of the dataset generated by the pandas describe() function.*

Now, we will check if the dataset has null values. We will visualize it as the graph below. As in the chart, except for the "timestamp" and "Potability" values, the data columns are missing. This we need to handle before feeding into the training model.

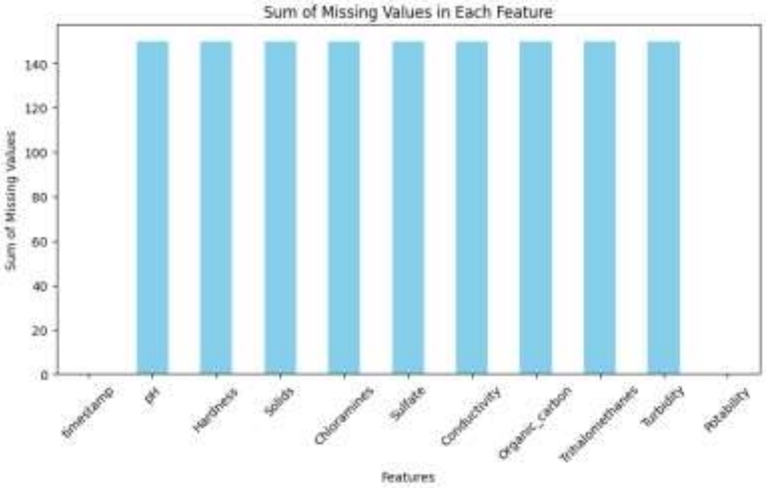


Figure 1. *Sum of missing values in each feature*

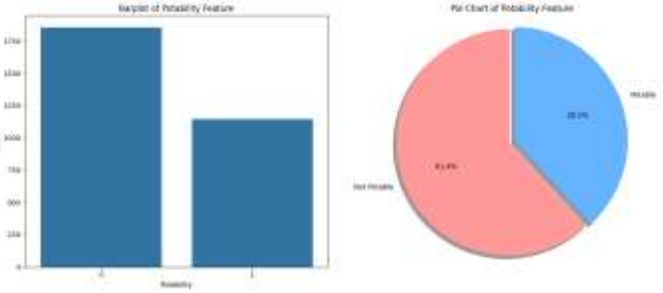


Figure 2. *Distribution of potable and non-potable water samples represented by a bar chart (left) and a pie chart (right).*

The distribution of the Potability feature is represented in a bar chart as well as in a pie chart– the bar graph on the left side of the image and the pie graph on the right side. According to the bar chart, from a total of 3000 water samples, it can be concluded that roughly 62% (approximately 1850 samples) have been classified as Not Potable (0) and 38% (approximately 1150 samples) have been classified as Potable (1). This sample classification imbalance is grossly visible in the pie chart, where the potable water is represented in blue and the non-potable water is represented inred.

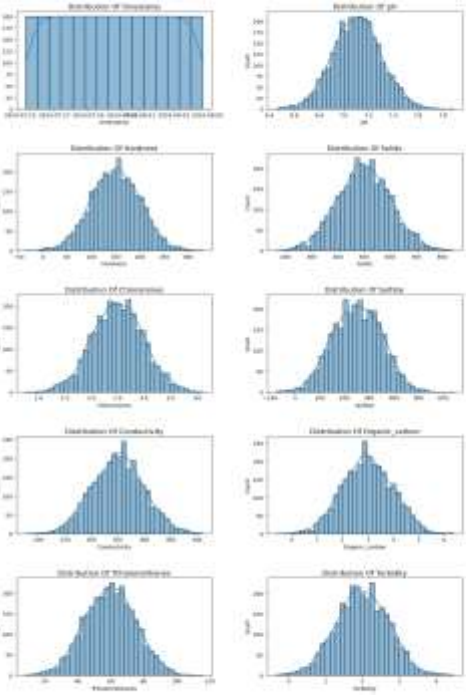


Figure 3. *Distribution of water quality parameters in the dataset (July-August 2024)*

Water quality parameters for each period are shown with each of the 10 histograms, each combined with a KDE curve – together, they depict the monitoring of water quality parameters during a specified interval as shown below.

- During the period of July and August 2024, the data set was reasonably compressed with a fidelity that reflects the data collection technique's effectiveness in tracking the phenomenon in real time.
- The data sets near natural distribution with a focus on average 7.1 'speak's' to the WHO's guidelines and standards on water quality parameters 6.5 and 8.5.
- The distribution characteristics of both hardness and sulfate show apparent and recognizable value errors or outlier values.
- The crucial focus on THMs as disinfection byproducts resonate with the shaped and statistically controlled distribution peaking near 60 µg/L.

It is noted there is a presence of moderately heavy-tailed distributions; the features which fundamentally emphasize adherence to a statistical model should ideally better handled or tended to.

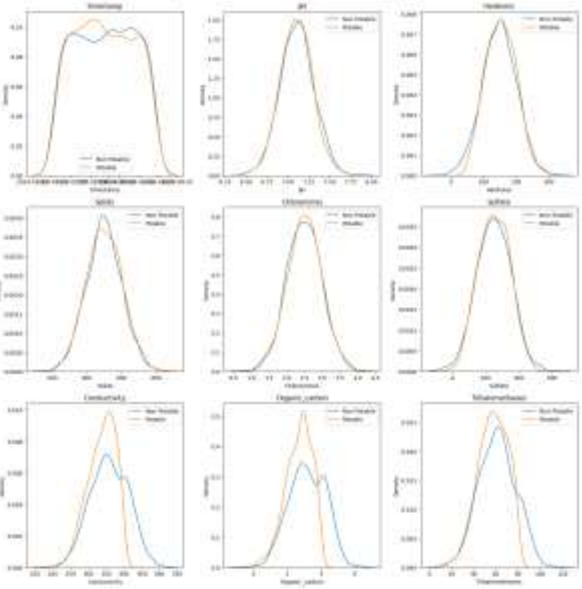


Figure 4. Distribution of water quality parameters by potability classification (July-August 2024)

The above figure shows KDE plots for each water quality parameter comparing the potable and non-potable water samples distributions. These plots also show how feature distributions change for the target class Potability.

- pH: Both classes have similar bell-shaped distributions. Potable water was skewed to the right, which shows that safer water tends to be verging more to the alkaline side.
- Hardness & Solids: The nearly identical distribution for both classes implies that these features may be insufficient for separation.
- Chloramines: There are minor differences because potable samples are more concentrated with chloramines which are consistent with treated water.
- Sulfate: There is absence of a class distinction which suggests a low correlation with potability.
- Conductivity: The potable class is more spread with a slight right-shift which implies that there is a slight elevation of safer water.
- Organic Carbon: Potable water is higher due to the greater concentration of organic carbon which shows that this feature is suggestive for prediction.
- Trihalomethanes: The slight decreases of Trihalomethanes for Potable water is more common, indicating moderate associate with potability.These density plots suggest that some features (e.g., Organic Carbon, Chloramines, Conductivity) may contribute meaningfully to the prediction of water potability, while others (e.g., Sulfate, Hardness) may be redundant or require feature engineering to extract more value.

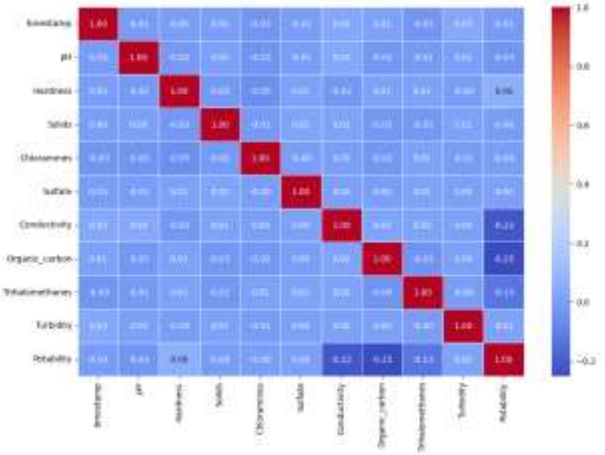
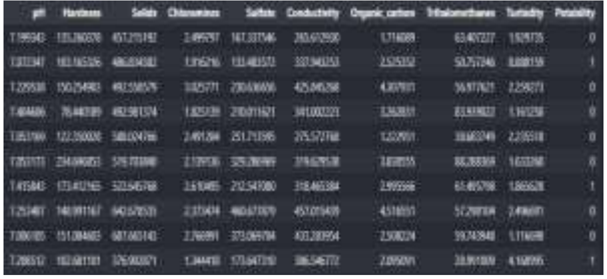


Figure 5. Pearson correlation heatmap of water quality parameters and potability (July–August 2024).

The heatmap displays the Pearson correlation coefficients among all numerical features, including the target variable Potability.

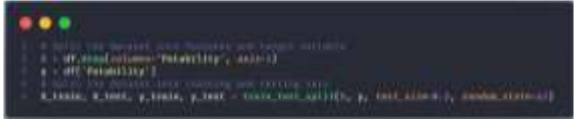
The next step in this research is to process the data and build a model. We will check the data columns and fill the missing values with the mean using the function dataframe[column].fillna(dataframe[column].mean(), inplace = True). The result after processing is as shown below.

Table 3. *Sample records from the water quality dataset showing measured parameters and potability classification.*



pH	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
7.195943	135.260370	457.215192	2.495797	167.337546	265.612930	1.716089	63.407227	1.929735	0
7.072347	183.165326	486.834382	1.916216	133.483572	337.340253	2.525352	50.757346	3.808159	1
7.229538	150.254983	492.558576	3.025771	230.630666	425.045268	4.307931	56.977621	2.258073	0
7.404686	76.448389	492.581374	1.825339	210.011621	341.000221	3.262831	83.193822	1.161258	0
7.051169	122.350020	580.024766	2.491284	251.713595	275.572768	1.222951	38.663249	2.235318	0
7.051171	234.690851	579.733898	2.129536	325.286989	379.629538	3.808555	88.288369	1.653268	0
7.415843	173.412165	522.545768	3.610485	212.541080	318.465384	2.995566	61.485798	1.865628	1
7.253487	148.991167	640.678531	2.373474	460.671079	457.019409	4.518031	57.290104	2.496871	0
7.306105	151.384663	601.663143	2.766991	373.068784	433.283954	2.508224	59.743848	1.116698	0
7.289512	182.681181	576.903871	1.344418	373.647318	386.546772	2.095091	28.991009	4.168995	1

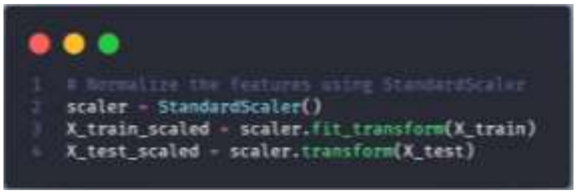
Then we split the dataset and normalize it. We will split the data into two parts: X and y. X is the feature matrix or independent variable, and y is the target vector or dependent variable.



```
# Split the dataset into features and target variable
X = df.drop(columns='Potability', axis=1)
y = df['Potability']
# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Listing 4. *Python code for splitting the dataset into training and testing sets*.

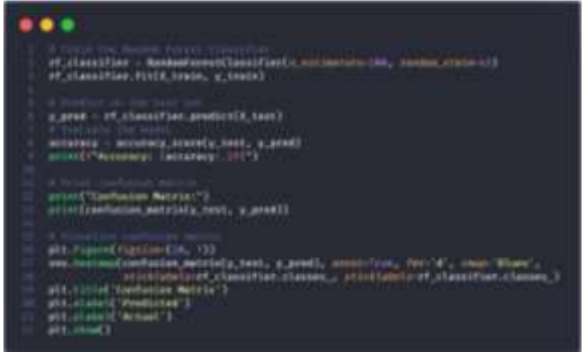
In this case, our target variable is Potability, which is a categorical variable indicating the level of water. We will use pandas. Drop() function to perform this operation.



```python
# Normalize the features using StandardScaler
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

Listing 5. *Python code for normalizing features using StandardScaler.*

Nomalization was applied using StandardScaler to normalize input features. The training set was fitted and transformed, while the test set was transformed using the same parameters. This ensures consistent scaling and prevents data leakage.



```python
# Train the Random Forest Classifier
rf_classifier = RandomForestClassifier(n_estimators=100, random_state=42)
rf_classifier.fit(X_train, y_train)

# Predict on the test set
y_pred = rf_classifier.predict(X_test)
# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.2f}")

# Print confusion matrix
print("Confusion Matrix:")
print(confusion_matrix(y_test, y_pred))

# Visualize confusion matrix
plt.figure(figsize=(10, 7))
sns.heatmap(confusion_matrix(y_test, y_pred), annot=True, fmt='d', cmap='Blues',
            xticklabels=rf_classifier.classes_, yticklabels=rf_classifier.classes_)
plt.title('Confusion Matrix')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()
```

Listing 6. *Training and evaluation of random forest classifier for water potability prediction*

As established previously: RandomForestClassifier(n_estimators=100, random_state=42)

- First, it constructs a Random Forest Classifier with a hundred trees (n_estimators=100).
- The random seed (random_state=42) guarantees reproducibility.

model.fit(X_train, y_train)

- Uses the training dataset (X_train, y_train) to train the model.
- The model can discern features which it will then predict.

The model can predict with an accuracy of 91.33% which is impressive, particularly for class 1 where the model achieves a recall of 90.2%. There are problems however, false positives (57) and false negatives (25) which can be quite significant depending on the problem at hand.

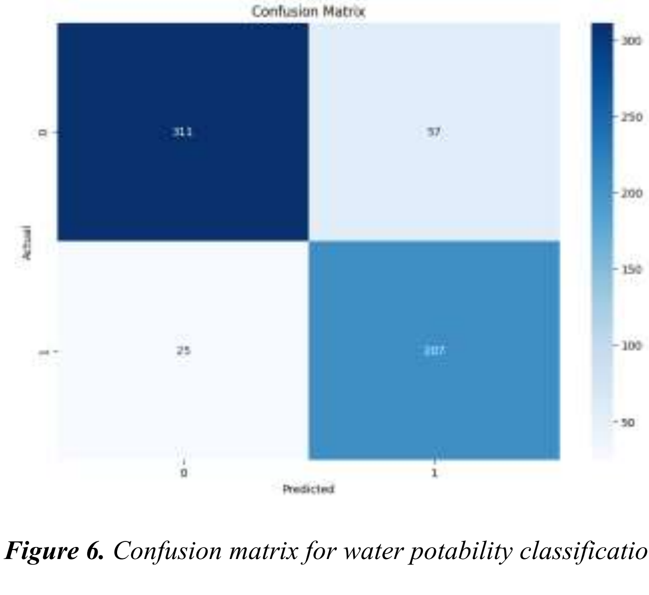


Figure 6. *Confusion matrix for water potability classification*

As the final step of our research, we will build an interactive user interface using Streamlit.

Options

WATER QUALITY PREDICTION

Figure 7. Manual input interface for water quality prediction system

The results are returned when water quality is clean.

Options

WATER QUALITY PREDICTION

Uploaded Data

Figure 8. User Interface of the Water Quality Prediction System

The result is returned when the water sample is polluted.

The water quality prediction system is a system that analyzes and predicts water quality based on sensor input parameters. The system works as given below.

First, users are input the option to use either a CSV file containing sensor data or system parameters, (this will be available soon) which will allow users to interface and input the data directly as parameters. Users will be asked a series of questions about where each parameter is adjusted through corresponding control slides.

The moment the user completes the data input, the system is supposed to analyze and use a machine learning model to predict water quality. The interface shows the prediction result, "water is CLEAN!!!" is shown if the water is determined to be clean otherwise, there are other warning messages which the system displays. The system serves as a powerful extra tool which helps users to determine if the water has any form of pollution which has potential harmful consequences.

5. Conclusion

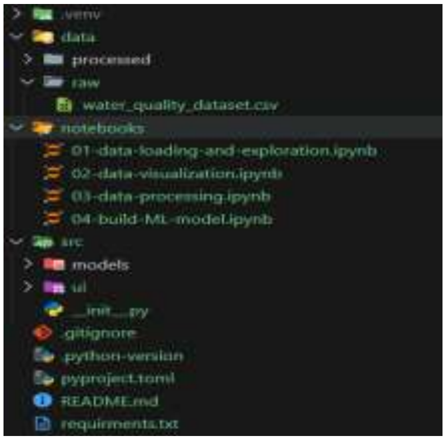
This study integrates robotics and electronics along with spatiotemporal data and remote sensing for the design and deployment of an automated system for monitoring the environment and controlling water quality in real time. The system's monitoring architecture allows detection and real-time monitoring of water quality indicators, triggering alerts and communication of risks at defined thresholds, and employs IoT sensors and Machine Learning. The system records a 91.33% accuracy rate, which is lower than the Random Forest (100%) of Alomani et al. (2022) and MLR (99.83%) of Kularbphettong et al. (2025), and XGBoost (97.06%) of Elvin Wibowo (2024) but is commendable considering the constraints and complexities of real-time data streams. Contrary to most of the existing literature which uses structured data, this system is predicated on the versatile and real-time data. Predictive models employing Random-Forest for groundwater ($R^2 = 0.82$; Apogba, 2024) and hybrid ARIMA-SSA-LSTM models ($R^2 = 0.998$; Wang, 2024) demonstrate excellent long-term forecasts but are prohibitively expensive to compute, thus real-time use is limited. Our Random-Voice System, on the other hand, is optimized for use with Artificial Intelligence and Internet-of-Things frameworks and distributed computing, providing lower cost solutions with minimized compromise on accuracy and real-time performance.

6. Scalability and future directions

The suggested water pollution monitoring system utilizes modular and adaptable architecture designed for effective large-scale use. Its backend coded in Python and its powerful servers process high volumes of data to add thousands of sensors without any slowdowns. The data platform supports external APIs and databases which enhance input data and improves predictive performance, such as hydrometeorological records and data on industrial discharges. Future development of the system will focus on high-accuracy predictive forecasting using deep learning techniques, particularly recurrent and convolutional neural networks, for more effective time-series and long-range forecasting. Using multi-layered geospatial data, pollution source and cause pattern analysis could be further improved using AI-based analysis. Research on IoT sensor energy optimization may also increase device lifespan, supporting the large-scale monitoring system's sustainability.

7. Appendices

Structure of project



```
> .venv
v data
  > processed
  v raw
      water_quality_dataset.csv
v notebooks
    01-data-loading-and-exploration.ipynb
    02-data-visualization.ipynb
    03-data-processing.ipynb
    04-build-ML-model.ipynb
v src
  > models
  > ui
    __init__.py
  .gitignore
  .python-version
  pyproject.toml
  README.md
  requirments.txt
```

Figure 9. *Project directory structure for water quality prediction system*

This project has been designed with modularity and scalability in mind and encompasses the entire data science life cycle, which spans data preprocessing, visualization, model building, and deploying the interface. Below, there is a folder and file structure breakdown:

.venv/ — Virtual Environment: To avoid any dependency conflicts with other projects on the same machine, it contains a python environment tailored to the specifications of this project and ensures that specific package versions do not clash.

data/ — Dataset Storage: Houses both the raw and processed datasets.

- raw/

Contains the original dataset (water_quality_dataset.csv) that has been acquired from external sources. It is the untouched data that is used for exploration and cleansing.

- processed/

Contains datasets that have been cleansed and preprocessed which are ready for input during the analysis and the model building phases.

notebooks/ — Data Science Workflow Notebooks: A compilation of Jupyter Notebooks created for conducting exploratory analyses, data cleansing, and model building.

src/ — The system's backbone which contains the main structure of the code, model interfaces with the user, and the user system incorporated in the system's core logic.

- models/: Comprises the python files with the necessary definitions, training and savers of the machine learning models.

ui/: Houses the components for the user interface that are designed to actively fetch results from users and input data from their ends.

REFERENCES

1. Alomani, S. M., Alhawiti, N. I., & Alhakamy, A. (2022). Prediction of quality of water according to a random forest classifier. *International Journal of Advanced Computer Science and Applications (IJACSA), 13*(6). https://doi.org/10.14569/IJACSA.2022.01306105

2. Apogba, J. N., Anornu, G. K., Koon, A. B., Dekongmen, B. W., Sunkari, E. D., & Fynn, O. F. (2024). Application of machine learning techniques to predict groundwater quality in the Nabogo Basin, Northern Ghana. *Heliyon, 10*(7), e28527. https://doi.org/10.1016/j.heliyon.2024.e28527
3. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785
4. Elvin, E., & Wibowo, A. (2024). Forecasting water quality through machine learning and hyperparameter optimization. *Indonesian Journal of Electrical Engineering and Computer Science, 33*(1), 496–506. https://doi.org/10.11591/ijeecs.v33.i1.pp496-506
5. Fang, P., Wang, Y., Zhao, Y., & Kang, J. (2025). Analysis of prediction confidence in water quality forecasting employing LSTM. *Water, 17*(7), 1050. https://doi.org/10.3390/w17071050
6. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
7. jgfranco17. (n.d.). *infraboard: Streamlit-based dashboard for monitoring environment performance*. GitHub. Retrieved September 22, 2025, from https://github.com/jgfranco17/infraboard
8. Kularbphettong, K., Raksuntorn, N., & Boonseng, C. (2025). Prediction of water quality index (WQI) using machine learning. *International Journal of Environmental Science and Development, 16*(1), 34–40. https://doi.org/10.18178/ijesd.2025.16.1.1507
9. Lee, S. (2025, June 18). Data preprocessing for environmental data science. *Number Analytics Blog*. https://www.numberanalytics.com/blog/data-preprocessing-environmental-data-science
10. Lin, F., Li, X., Su, Y., Yan, J., & cộng sự. (2025). Water quality prediction model based on improved long short-term memory neural network and empirical mode decomposition. Discover Artificial Intelligence, 5, 199. https://doi.org/10.1007/s44163-025-00454-y
11. Wang, T., Chen, W., & Tang, B. (2024). Water quality prediction using ARIMA-SSA-LSTM combination model. *Water Supply, 24*(4), 1282–1297. https://doi.org/10.2166/ws.2024.060
12. Lokman, A., Ismail, W. Z. W., & Aziz, N. A. A. (2025). A review of water quality forecasting and classification using machine learning models and statistical analysis. *Water, 17*(15), 2243. https://doi.org/10.3390/w17152243 MDPI
13. Yan, X., Zhang, T., Du, W., Meng, Q., & Xu, X., & Zhao, X. (2024). A comprehensive review of machine learning for water quality prediction over the past five years. *Journal of Marine Science and Engineering, 12*(1), 159. https://doi.org/10.3390/jmse12010159
14. Zaidi, A. Z. (2012, May). Water quality management using GIS and RS tools. In *World Environmental and Water Resources Congress 2012*. ASCE. https://doi.org/10.1061/9780784412312.086
15. Zou, S., Ju, H., & Zhang, J. (2025). Water quality management in the age of AI: Applications, challenges, and prospects. *Water, 17*(11), 1641. https://doi.org/10.3390/w17111641