

Varuna: Integrating AI with Environmental Data for Water Quality Predictions

Dristi Sharma, Suhani Rai, Harshini Chandilvel

dristis2009@gmail.com, sr.suhanirai@gmail.com, harshicmk@gmail.com

McNeil High School, 5720 McNeil Drive, Austin, Texas 78729, USA

ABSTRACT

Water quality monitoring is essential for protecting human health and ecosystems. Still, traditional analysis methods are too time - consuming to process. As the new artificial intelligence revolution thrives, to address this problem, we tested the hypothesis that an AI system trained on environmental data can accurately predict water quality conditions. Publicly available water quality sample data for Austin, Texas, was combined with environmental data from an API (Application Programming Interface) over an overlapping three - months. PH, dissolved oxygen, turbidity, nitrate, phosphorus, water temperature, max temperature, total precipitation, evaporation, and daylight hours were among the key parameters that were processed and utilized to assign categories of Water Quality Index. A Random Forest classifier, a baseline model, was evaluated for accuracy and variable significance and using these results, a FLAN - T5 language model was refined to generate both water - quality predictions and natural - language explanations from numeric inputs. The refined FLAN - T5 model produced water - quality classifications along with coherent explanations aligned with these traits. An example printed result from Varuna is "Water Quality Category: Fair. Water quality is primarily influenced by low dissolved oxygen, low nitrate levels, pH within optimal range, seasonal daylight effects." Indicating that AI systems can match traditional performance while improving usability, this model is presented as a user - friendly public website for everyone to predict future forecasts with up - to - date statistics. Further research is necessary to see if the machine learning model can adapt to the constantly changing world. We will identify contamination risk levels and support early intervention strategies for protecting public health and ecosystems.

KEYWORDS: MACHINE LEARNING, WATER QUALITY, ENVIRONMENT, RANDOM FOREST, FLAN-T5

I. Introduction

I.I Background Research

Research was consulted by a senior environmental scientist from the City of Austin Watershed Protection Department, who explained that key factors such as turbidity, nitrate, and phosphorus levels are commonly used to evaluate water quality and that the process requires significant time and team effort. This motivated the authors to explore how AI could help the process of water quality prediction. The authors then reviewed research such as Chuxiao Chen and Jinghua Hao's study on neural-network-based water quality prediction at data-scarce sites, which showed how machine learning models can improve prediction accuracy even with limited environmental data. Additionally, examining research on spatial dependency learning and Kaggle water quality datasets to better understand feature selection, preprocessing methods, and AI modeling approaches used in existing water quality prediction systems.

A. Literature Review

The study by Chuxiao Chen and Jinghua Hao focuses on improving water quality prediction at monitoring sites where historical data is limited. The researchers address a major challenge in environmental modeling: machine learning systems typically require large datasets to perform well, but many real-world monitoring stations lack sufficient records. To solve this, they developed a neural-network-based framework that incorporates transfer learning, allowing knowledge from data-rich monitoring sites to be applied to data-scarce ones. By pre-training models on locations with abundant environmental data and fine-tuning them for target sites with limited records, the approach significantly improved prediction accuracy compared to models trained only on local data. The study also emphasized selecting source sites with similar water quality characteristics to enhance transfer performance.

In addition to this research, related studies on spatial dependency learning and publicly available Kaggle water quality datasets highlight the importance of proper preprocessing and feature selection in AI modeling. These works show that cleaning missing data, normalizing variables, balancing class distributions, and carefully selecting relevant water quality indicators—such as pH, turbidity, and dissolved solids—can substantially improve model performance. Together, these studies demonstrate that advanced machine learning techniques, especially when combined with thoughtful preprocessing and transfer strategies, can produce reliable water quality predictions even in environments where data availability is limited.

I.II Rationale

Water consumption is one of the most imperative necessities for human health and survival. Fresh, clean water ensures the betterment of the environment and the human population. Ensuring the purity of water makes lives easier and allows people to focus on innovation and creation in other fields so everyone is able to live without any impurities that could contaminate drinking water. Pure water prevents diseases, supports biodiversity, and provides resources for agriculture and industry.

Project Varuna aims to use explainable AI models to predict future water quality using past and present water quality index data (collecting water quality indices(WQI) with data like DO, pH, turbidity, BOD, nitrates, phosphates, and temperature, weather/hydrology, land use, upstream flow, and precipitation) to forecast future simulations. The understanding of probable issues or hazardous changes to water quality helps people identify risk areas and mitigate the possibility of impure water. Varuna: Integrating AI with Environmental Data for Water Quality Predictions will be accessible to all and will update frequently, minding climate change, and other unpredictable factors.

Especially in Austin, Texas, there is wider breadth for data as Texas is not usually prepared for drastically changing winter weather. AI can transform water quality monitoring by analyzing real time data, identifying patterns, and making predictions that enable more proactive water management. We hope to test the accuracy and usefulness of our AI system software in this project and see whether it works better than traditional data analysis which is costly and labor intensive, contains data parsing gaps, and takes too much time.

I.III Research Question

Can a machine learning-based AI decision system accurately predict and simulate changes in water quality index (WQI) values under different pollutant and environmental conditions, outperforming traditional empirical models?

I.IV Objective

The objective of Project Varuna is to create an artificial intelligence-based decision support tool that is able to make predictions about the water quality and explain these predictions in an easily understandable format. Current approaches to water quality analysis are often labor-intensive, expensive, and difficult to understand for those without a strong background in the field. Project Varuna is a combination of machine learning and explainable AI that is able to analyze physicochemical measurements of water, along with environmental factors, in order to make faster and more understandable water quality assessments. The purpose of Project Varuna is to show how AI can be used in the field of environmental science.

I.V Hypothesis

If a deep learning model is trained on multi-source environmental data, then it will predict water quality indices more accurately than traditional models because it can learn nonlinear relationships between environmental variables and handle multiple water-quality parameters in a more efficient and integrated way.

I.VI Engineering Goals

- A. To develop Project Varuna, an AI-powered decision-support tool that:
 - i. Uses a fine-tuned FLAN-T5 model to predict water-quality outcomes from environmental and chemical measurements.
 - ii. Generates natural-language explanations showing which factors most influenced each prediction.
 - iii. Demonstrates how AI can support environmental monitoring by providing fast, interpretable water-quality assessments.

I.VII Expected Outcomes

- A. A fully trained FLAN-T5 model capable of predicting WQI or potability from the UCI dataset.
 - i. Clear natural-language explanations describing why the model made each prediction.
 - ii. A comparison showing the AI model performs better than a baseline traditional method.
 - iii. A prototype interface or report demonstrating how AI could assist scientists, city planners, or environmental engineers in assessing water quality more quickly than manual methods

II. Methodology

II.I Materials

A. Hardware & Platforms

- i. Laptop – A computer was used to write code, preprocess datasets, train machine learning models, and evaluate performance metrics. The laptop served as the primary computational device for model development and experimentation.
- ii. Visual Studio Code – Visual Studio Code (VS Code) was used as the integrated development environment (IDE) for writing, debugging, and managing Python scripts. It provided extensions for Python support, version control integration, and efficient project organization.

B. Datasets & Data Sources

- i. Hugging Face Flan-T5 Model (google/flan-t5-base) – The pre-trained google/flan-t5-base language model, accessed through the Hugging Face repository, was used to generate natural language explanations of model predictions. This transformer-based model converts structured numerical outputs into clear, human-readable summaries.
- ii. Open-Meteo Weather Data – Historical weather data was obtained from the Open-Meteo API. This dataset provided environmental variables such as temperature and precipitation, which were aligned with water sampling dates to examine environmental impacts on water quality.
- iii. City of Austin Water Quality Data – Publicly available monitoring data from the City of Austin was used to obtain water chemistry measurements, including pH, turbidity, dissolved oxygen, nitrate levels, phosphorus concentration, and water temperature. These variables served as primary inputs for the predictive model.

C. Software and Libraries

- i. Python 3.9+ Environment – The project was developed using Python (version 3.9 or higher), which served as the core programming language for data processing, machine learning model development, and evaluation.
- ii. Machine Learning & Data Libraries – Python libraries were used for data manipulation, model training, and performance evaluation. These libraries supported preprocessing, feature selection, classification modeling, and metric calculation.
- iii. Hugging Face Repository – The Hugging Face platform was used to access and implement the pre-trained Flan-T5 model for explainable AI output generation.
- iv. GitHub – GitHub was used for version control, code storage, and project tracking, allowing for organized development and reproducibility of results.

II.II Variables

To ensure the validity of the projects, independent, dependent, and controlled variables were utilized in the experimental process. The definition of these variables below allows for repeatability and objective results.

- A. Independent Variables: Environmental data and water quality indicators used as model inputs.
- B. Dependent Variables: The predicted water quality category and information
- C. Controlled Variables: The location, data sources, preprocessing methods, and the machine learning model.

II.III Parameters

Acceptable parameter ranges were considered during model development, including a pH range of 6.5–8.5, turbidity levels below 5 NTU (as turbidity increases during storm events and reflects water cloudiness), and dissolved oxygen levels above 7 mg/L to support aquatic organisms. Because the dataset did not include extreme weather events, the model had limited ability to predict nitrate spikes typically associated with heavy rainfall.

II.IV Data Merging

Weather and water quality datasets were matched by date to ensure that environmental conditions corresponded to the same time periods as water chemistry measurements. After alignment, the datasets were merged into a single combined dataset containing both environmental and chemical variables. Missing values were reviewed and handled as needed to ensure data consistency before model training.

II.V Procedure

A. Data Collection

- i. Data used in this study was collected from publicly accessible water quality sampling datasets, OpenMeteo for weather and City of Austin for water quality.. The datasets included physicochemical parameters such as pH, dissolved oxygen, turbidity, nitrate, phosphorus, and water temperature. Environmental data was collected from publicly accessible weather data sources. Environmental data included daily environmental parameters such as daylight duration and temperature. No physical samples, chemicals, or biological materials were used.

B. Data Cleaning and Preprocessing

- i. The data cleaning of the water quality dataset involved the removal of invalid data, standardization of the columns, and filtering of the data based on parameters that are of interest in the assessment of water quality. The data was also transformed from a long format to wide format so that each sample of water corresponds to a single row with multiple columns.
- ii. The data cleaning of the environmental dataset involved the removal of metadata, standardizing column names, converting dates into a consistent format, and eliminating duplicate entries. When new environmental data was obtained, the old and new data was combined programmatically.

C. Dataset Integration

- i. The cleaned water quality and environmental datasets were integrated based on their common key, which was the sampling dates. This led to the creation of a single dataset that contained both water quality and environmental data. The data was made consistent by keeping only those records that had overlapping dates.

D. Water Quality Index (WQI) Labeling

- i. A Water Quality Index-based system of categorization was also implemented using scientific thresholds for important parameters like pH, dissolved oxygen, turbidity, and nitrate concentration. A sub-score was assigned to each parameter depending on the range within which it fell—optimal, moderate, and poor ranges. A water quality category was assigned by summing up the sub-scores for each parameter. This process of labeling the water samples with categories was used for machine-learning training.

E. Baseline Machine Learning Model

- i. A Random Forest classifier was used to train a model on the labeled dataset, predicting water quality categories based on numerical input data. The data was split into a training set and a test set, with accuracy, precision, recall, and feature importance used to test the model's performance. The feature importance values helped identify significant water quality drivers, such as dissolved oxygen, nutrient concentration, and pH.

F. Explainability Layer

- i. The results obtained from the feature importance of the baseline model were utilized to generate rule-based natural language explanations for each sample. These explanations described the reason for the sample's classification into a particular category based on scientifically relevant factors, such as low levels of dissolved oxygen and high levels of nitrates.

G. AI Model Fine-Tuning (FLAN-T5)

- i. The data was transformed into instruction-response form, which is applicable for large language models. In this case, the instructions were related to numeric water quality and environmental measurements, while the responses included water quality categories and explanations. A FLAN-T5 transformer-based model was used for fine-tuning to jointly classify water quality and generate human-readable explanations from numeric data.

H. Model Testing and Demonstration

- i. The trained FLAN-T5 model was subjected to testing using new or simulated realistic values for the input variables. The goal of the testing was to assess the model's ability to generalize the predictions. The AI model's predictions were compared with the baseline model predictions for the purpose of testing the interpretability of the model. A local interactive web-based demo was developed.

II.VI Prototypes/Trials

Project Varuna had multiple iterations, specifically when testing the `predict_forecast.py` prediction script. The terminal output reveals a critical misalignment between how the script was designed to receive inputs and how

the model had actually been trained. The error message indicates that the script was built to accept raw physicochemical parameters as direct command-line arguments — `--cond_max`, `--ph_max`, `--ph_min`, `--do_max`, `--temp_mean`, and several others — meaning it expected individual water quality measurements to be passed in manually before any processing occurred. However, the model pipeline at this stage had been designed around pre-computed Water Quality Index scores and categorized labels rather than raw numeric inputs fed through the command line. This created a fundamental mismatch: the interface expected one format while the model anticipated another. This kind of error is actually scientifically meaningful within the context of an iterative engineering process. It revealed that the input-output contract between the data preprocessing pipeline and the prediction interface had not yet been properly unified. In a real-world AI deployment for environmental monitoring, this type of discrepancy could cause a system to silently produce incorrect predictions or fail to run entirely, both of which are unacceptable in a tool intended to support public health decisions around water safety. Through multiple trials like this one, a diagnosis was made about where the data flow was breaking down, refining the argument structure of the prediction script, and ensuring that the model received inputs consistent with the format used during fine-tuning of the FLAN-T5 model. Each failed run served as a diagnostic checkpoint, progressively narrowing the gap between the raw environmental data coming in from the Austin weather and water quality datasets and the natural-language predictions the model was trained to generate. This iterative debugging process is a standard and necessary part of applied machine learning development, and documenting it demonstrates a rigorous, evidence-based approach to engineering refinement.

```
predict.py: error: the following arguments are required: --cond_max, --ph_max, --ph_min, --cond_min, --cond_mean, --do_max,
--do_mean, --do_min, --temp_mean, --temp_min, --temp_max
PS C:\Users\suhan\Documents\science_fair> python predict_forecast.py
usage: predict_forecast.py [-h] --cond_max COND_MAX --ph_max PH_MAX --ph_min PH_MIN --cond_min COND_MIN
--cond_mean COND_MEAN --do_max DO_MAX --do_mean DO_MEAN --do_min DO_MIN --temp_mean TEMP_MEAN
--temp_min TEMP_MIN --temp_max TEMP_MAX
predict_forecast.py: error: the following arguments are required: --cond_max, --ph_max, --ph_min, --cond_min, --cond_mean,
```

Figure 5.1- Errors from the first trial of Project Varuna

II.VII Calculations: F1 - Score

The F1 score in AI is a performance metric for classification models that calculates the harmonic mean of precision and recall. Ranging from 0 to 1 (or 0–100%), it balances the trade-off between catching all positive instances (recall) and ensuring high accuracy in positive predictions (precision). It is essential for imbalanced datasets. The F1 score is really useful. It provides a balance between precision and recall by taking their harmonic mean. The balance between precision and recall is crucial because it ensures that an increase in one metric does not disproportionately affect the overall score without a corresponding improvement in the other. In scenarios where false positives and false negatives carry different costs, the F1 score helps maintain a trade-off: improving precision typically reduces false positives, while improving recall reduces false negatives.

$$\text{F1 Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}$$

Figure 5.2 - F1 Score Formula

F1 - Score

True Positive (TP) = Predicted Fair and actually Fair $\rightarrow 6$
False Positive (FP) = predicted Fair but actually Good $\rightarrow 0$
False Negative (FN) = predicted good but actually Fair $\rightarrow 0$

Precision = $\frac{TP}{TP + FP} = \frac{6}{6 + 0} = 1.0$

Recall = $\frac{TP}{TP + FN} = \frac{6}{6 + 0} = 1.0$

F1 = $2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = 2 \cdot \frac{1 \cdot 1}{1 + 1} = 2 \cdot \frac{1}{2} = 1.0$

Figure 5.3- Calculations done to find F1 Score

II.VIII Merged Data Set

SAMPLE_ID	SAMPLE_DATE	dissolved_oxygen	nitrate	ph	phosphorus	turbidity	water_temp	temperature_2m_max	precipitation_sum	daylight_duration	et0_fao_evapotranspiration
1-LBL @ Basin BOTT	10/21/2025	3.95		7.89			26.3	30.7	0	40556.97	0
1-LBL @ Basin BOTT	12/16/2025	8.61		7.77			15.47	16.1	1.1	36719.8	0
1-LBL @ Basin SURF	10/21/2025	7.36		8.22			26.31	30.7	0	40556.97	0
1-LBL @ Basin SURF	12/16/2025	9.54		8.07			15.6	16.1	1.1	36719.8	0
1-LBL @ Basin dp1	10/21/2025	7.28		8.2			26.39	30.7	0	40556.97	0
1-LBL @ Basin dp2	10/21/2025	7.1		8.18			26.36	30.7	0	40556.97	0
1-LBL @ Basin dp3	10/21/2025	6.95		8.16			26.36	30.7	0	40556.97	0
1-LBL @ Basin dp4	10/21/2025	6.93		8.15			26.32	30.7	0	40556.97	0
1-LBL @ Basin dp5	10/21/2025	6.78		8.13			26.33	30.7	0	40556.97	0
1-LBL @ Basin dp6	10/21/2025	5.68		8.03			26.24	30.7	0	40556.97	0
1-LBL @ Basin dp7	10/21/2025	4.16		7.91			26.21	30.7	0	40556.97	0
1-LBL @ Basin-Bott	10/21/2025		0.0449		0.215			30.7	0	40556.97	0
1-LBL @ Basin-Surf	10/21/2025		0.0393		0.145			30.7	0	40556.97	0
1184-LBL ds Red Bud Trail SURF	11/21/2025	6.25		7.7			23.01	24.5	0.6	37788.25	0
1191-GIL @ West Parsons	10/14/2025	7.78	5.31	8.1	0.142	10.09	22.39			41287.28	
1252-LBL @ Powerline SURF	10/3/2025	6.12		7.72			25.83			42466.34	
1252-LBL @ Powerline SURF	10/24/2025	7.23		7.95			24.1	30.9	5.2	40248.06	0
1252-LBL @ Powerline-Surf	10/3/2025		0.0481		0.174					42466.34	
1252-LBL @ Powerline-Surf	10/24/2025		0.167		404.165			30.9	5.2	40248.06	0
142-BUL @ Spwd Sggs 7th Xing [tg]	10/14/2025	8.05	0.02	8.18	0.168	0.82	20.92			41287.28	
145-ONI southeast of SAR WWTP	10/14/2025	6.58	0.02	8.01	0.21	2.7	23.07			41287.28	
160-Backdoor Spg	12/10/2025	8.535		7.712			21.78	22.1	0	36839.94	0
1996-Redbud W of Parking lot SURF	10/3/2025	3.96		7.81			25.13			42466.34	
1996-Redbud W of Parking lot SURF	10/24/2025	5.4		7.8			24.36	30.9	5.2	40248.06	0
1996-Redbud W of Parking lot-Surf	10/3/2025		0.02		0.225					42466.34	
1996-Redbud W of Parking lot-Surf	10/24/2025		0.0317		469.08			30.9	5.2	40248.06	0
1997-Lagoon @ Festival beach SURF	10/3/2025	8.22		8.07			26.54			42466.34	
1997-Lagoon @ Festival beach SURF	10/24/2025	7.69		8.2			25.63	30.9	5.2	40248.06	0
1997-Lagoon @ Festival beach-Surf	10/3/2025		0.02		0.172					42466.34	
1997-Lagoon @ Festival beach-Surf	10/24/2025		0.0583		496.037			30.9	5.2	40248.06	0
2-LBL @ 1st BOTT	10/21/2025	6.08		7.69			25.3	30.7	0	40556.97	0
2-LBL @ 1st BOTT	12/16/2025	8.52		7.66			15.89	16.1	1.1	36719.8	0
2-LBL @ 1st SURF	10/21/2025	8.31		8.03			25.52	30.7	0	40556.97	0
2-LBL @ 1st SURF	12/16/2025	8.7		7.75			16.11	16.1	1.1	36719.8	0
2-LBL @ 1st dp1	10/21/2025	8.31		8.02			25.7	30.7	0	40556.97	0
2-LBL @ 1st dp2	10/21/2025	7.82		7.96			25.68	30.7	0	40556.97	0
2-LBL @ 1st dp3	10/21/2025	7.48		7.86			25.64	30.7	0	40556.97	0
2-LBL @ 1st-Bott	10/21/2025		0.173		0.108			30.7	0	40556.97	0
2-LBL @ 1st-Surf	10/21/2025		0.288		0.043			30.7	0	40556.97	0
233-RIN 100 ft ds Bradshaw Rd	10/14/2025	3.19	0.02	7.65	0.291	1.5	22.82			41287.28	
2360-LWA @ Quail Valley A	10/7/2025	2.63	0.901	6.99			29.13			42030.61	
2732-LBL @ 1621 SURF	11/21/2025	7.95		8.17			22.76	24.5	0.6	37788.25	0
318-TYS @ Reed Pk	10/14/2025	5.741	2.03	8.77	0.149	4.851	23.26			41287.28	
321-1118 W 7th	11/5/2025	9.61		8.27		2.65	26	28.2	0	39093.84	0
3247-LBL VIP Site 34 SURF	11/21/2025	8.271		8.275			22.89	24.5	0.6	37788.25	0
3297-Windsor Rd A	10/7/2025	3.436	4.05	6.82		1.866	25.26			42030.61	
3332-GIL @ FM 973	10/14/2025	7.58	10.9	8.1	0.465	13.47	25.04			41287.28	
3333-WLN 50 ft ds Metric Blvd	10/14/2025	5.08	0.281	7.6	0.01	1.54	20.56			41287.28	
35-Barton Spg	12/10/2025	4.28		7.31			22.95	22.1	0	36839.94	0
35-Barton Spg	10/8/2025	4.06	1.42	7.33			23.26			41922.57	
35-Barton Spg	11/18/2025	4.04	0.169	7.26			23.7	29.4	0	38002.07	0
390-11520 Lago de Garda Flood Detention SW Corner	12/8/2025	4.22		7.04			11.9	15.9	0	36901.91	0
390-11520 Lago de Garda Flood Detention West Wall	12/8/2025	7.1		7.75			17.2	15.9	0	36901.91	0
390-11520 Lago de Garda Inlet Storm Pipe	12/8/2025	7.08		7.05			15.7	15.9	0	36901.91	0
390-11520 Lago de Garda Open Channel Outlet	12/8/2025	17.4		8.6			16.7	15.9	0	36901.91	0
4040-Red Bud boat ramp SURF	11/21/2025	4.83		7.46			22.47	24.5	0.6	37788.25	0
422-Old Mill Spg	12/10/2025	2.88		7.22			19.16	22.1	0	36839.94	0
422-WQ-Old Mill Spg	11/14/2025	2.9		7.17			20.57	29.1	0	38314.11	0
428-Eliza Spg	12/10/2025	4.63		7.31			21.84	22.1	0	36839.94	0
463-WLS @ Walnut Metropolitan Park	10/14/2025	5.3	0.02	7.9	0.11	3.27	19.86			41287.28	
5-LBL @ Red Bud BOTT	10/21/2025	5.39		7.19			21.85	30.7	0	40556.97	0
5-LBL @ Red Bud BOTT	12/16/2025	9.24		8.23			16.43	16.1	1.1	36719.8	0
5-LBL @ Red Bud SURF	10/21/2025	7.02		7.92			25.16	30.7	0	40556.97	0
5-LBL @ Red Bud SURF	12/16/2025	9.01		8.22			17.02	16.1	1.1	36719.8	0
5-LBL @ Red Bud dp1	10/21/2025	6.99		7.92			25.19	30.7	0	40556.97	0
5-LBL @ Red Bud dp2	10/21/2025	5.32		7.67			24.7	30.7	0	40556.97	0
5-LBL @ Red Bud dp3	10/21/2025	6.47		7.32			22.7	30.7	0	40556.97	0
5-LBL @ Red Bud dp4	10/21/2025	6.37		7.22			22.11	30.7	0	40556.97	0
5-LBL @ Red Bud dp5	10/21/2025	5.42		7.19			21.83	30.7	0	40556.97	0
5-LBL @ Red Bud-Bott	10/21/2025		1.23		0.049			30.7	0	40556.97	0
5-LBL @ Red Bud-Surf	10/21/2025		0.6464		0.045			30.7	0	40556.97	0
51-BAR downstream of Lost Ck Blvd	10/14/2025	5.889	0.8449	7.894	0.01	1.132	22.46			41287.28	
560-LKA @ Mansfield Dam SURF	10/3/2025	1.65		7.5			21.71			42466.34	
560-LKA @ Mansfield Dam SURF	10/24/2025	6.65		7.99			26.38	30.9	5.2	40248.06	0
560-LKA @ Mansfield Dam-Surf	10/3/2025		0.02		0.095					42466.34	
560-LKA @ Mansfield Dam-Surf	10/24/2025		0.02		0.01			30.9	5.2	40248.06	0
561-LKA @ Tom Miller Dam SURF	10/3/2025	7.2		7.85			27.34			42466.34	
561-LKA @ Tom Miller Dam SURF	10/24/2025	8.165		8.14			26.59	30.9	5.2	40248.06	0
561-LKA @ Tom Miller Dam-Surf	10/3/2025		0.02		0.136					42466.34	
561-LKA @ Tom Miller Dam-Surf	10/24/2025		0.02		0.224			30.9	5.2	40248.06	0
573-LKA @ Emma Long Pk SURF	10/3/2025	6.96		8.14			26.04			42466.34	
573-LKA @ Emma Long Pk SURF	10/24/2025	7.38		8.01			26.85	30.9	5.2	40248.06	0
573-LKA @ Emma Long Pk-Surf	10/3/2025		0.02		0.051					42466.34	
573-LKA @ Emma Long Pk-Surf	10/24/2025		0.02		0.243			30.9	5.2	40248.06	0
589-WQ-Eliza Stream	11/3/2025	4.1		7.2			21.9	25.6	0	39275.96	0
624-WLR @ 23rd	10/14/2025	5.4	2.18	7.83	0.125	4.02	23.26			41287.28	
860-BSP @ Lower Dam	12/10/2025	5.84		7.51			21.85	22.1	0	36839.94	0
860-BSP @ Lower Dam	10/8/2025	5.98		7.27			23.99			41922.57	
860-BSP @ Lower Dam	11/18/2025	6.77		7.41			24.12	29.4	0	38002.07	0
9-Cold Spg	12/10/2025	6.25		7.32			21.17	22.1	0	36839.94	0
930-Spicewood Trib ds Spicewood Spg	10/14/2025	5.4	1.82	7.26	0.09	1.96	23.13			41287.28	
939-Big Boulder Spg A	10/7/2025	7.497	4.77	7.261			23.95			42030.61	

Figure 6.1-Merged Data including Environmental Data and Water Quality Data

II.IV Data from Open-Meteo

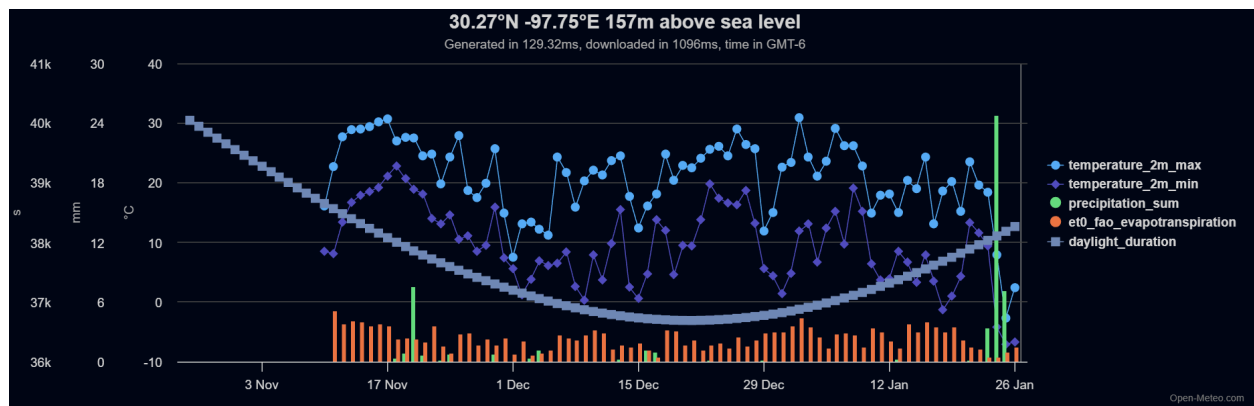


Figure 7.1- Weather Data from API: Open-Meteo

II.IV Training Model Code

```

1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from sklearn.ensemble import RandomForestClassifier
4 from sklearn.metrics import accuracy_score, classification_report
5
6
7 data_path = "data/processed/labeled_data.csv"
8 df = pd.read_csv(data_path)
9
10 feature_cols = [
11     "ph",
12     "dissolved_oxygen",
13     "turbidity",
14     "nitrate",
15     "phosphorus",
16     "water_temp",
17     "daylight_duration"
18 ]
19
20 X = df[feature_cols]
21 y = df["wqi_category"]
22
23
24 X = X.fillna(X.median())
25
26
27 X_train, X_test, y_train, y_test = train_test_split(
28     X, y,
29     test_size=0.25,
30     random_state=42,
31     stratify=y
32 )
33
34
35 model = RandomForestClassifier(
36     n_estimators=200,
37     random_state=42
38 )
39
40 model.fit(X_train, y_train)
41
42 y_pred = model.predict(X_test)
43
44 acc = accuracy_score(y_test, y_pred)
45 print(" Baseline Random Forest Results")
46 print("Accuracy:", round(acc, 3))
47 print("\nClassification Report:")
48 print(classification_report(y_test, y_pred))
49
50
51 importances = pd.Series(
52     model.feature_importances_,
53     index=feature_cols
54 ).sort_values(ascending=False)
55
56 print("\nFeature Importance:")
57 print(importances)

```

Figure 7.2- Code from Training Plan T5

III. Results and Discussion

III.I Random Forest Confusion Matrix

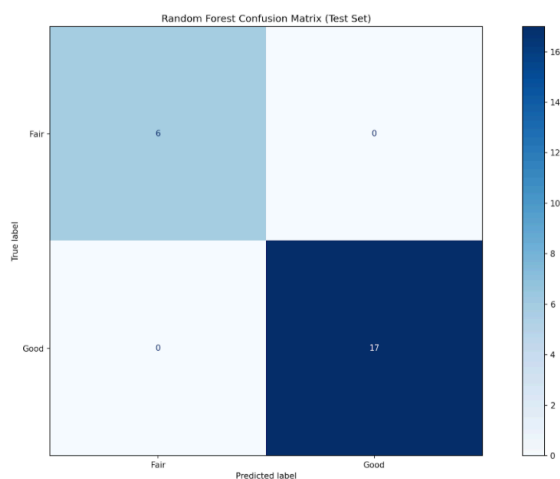


Figure 7.3 - Confusion Matrix

A confusion matrix, shows how well the Random Forest model predicted water quality categories on data it had never seen before (the test set). It compares the Actual (true) labels - left side (y-axis) by the Predicted labels - bottom (x-axis). The model made 23 correct predictions and 0 incorrect predictions. The confusion high correct predictions because the dataset is relatively small and the water quality categories were created from clear environmental thresholds. The features (DO, nitrate, pH, turbidity) strongly determine the category

III.II Data Analysis

The data analysis was carried out purely using computational techniques in the programming language Python. Once the data was preprocessed and the data from both sources was merged, summary statistics were used to obtain information about the data, including the size of the dataset and the range of the parameters. Tables were used to present the size of the dataset, the range of the parameters, and the frequency of the water quality categories. For the assessment of the predictive efficiency, a Random Forest classifier was used to develop a baseline model. The accuracy, precision, recall, and F1-score (balanced measure of precision and recall) of the model were calculated on a test set, and the results were presented in tabular form. The feature importance values obtained from the Random Forest model were presented in the form of a ranked table or chart to identify the most important features used to predict the water quality. The natural language explanations generated by the fine-tuned FLAN-T5 model were qualitatively compared with the feature importance results of the baseline model. The explanation generated by the AI model was ensured to be consistent with the physicochemical drivers, including dissolved oxygen, nutrient levels, and pH, among others. The example outputs were shown as text tables to demonstrate the relationship between the numeric inputs and the predicted categories and explanations. The classification consistency was compared between the baseline model and the AI model. By utilizing graphs, tables, and performance metrics, it was determined if an accurate water quality assessment was being made by the AI system.

III.III Project Interface

Figure 8.1- Home Page of AI Demo

talk about the different prediction outputs

Figure 8.2- Outputs of Different Data Inputs on AI Demo

Compared to traditional rule-based approaches, the AI model analyzed several variables at once and identified patterns that may not be obvious through traditional methods, allowing for more adaptable and accurate assessments. By combining predictive accuracy with explainable results, AI-based systems have the potential to support faster decision-making and improve how communities monitor and manage water resources.

III.IV Observations

- A. Water samples with higher dissolved oxygen and balanced pH were consistently classified as Good water quality.
- B. Samples with lower dissolved oxygen or higher nutrient levels were more often classified as Fair.
- C. The Random Forest model showed that dissolved oxygen, nitrate, and pH had the strongest influence on predictions.
- D. The fine-tuned FLAN-T5 model generated water-quality predictions along with explanations that referenced the same key factors identified by the baseline model.
- E. Both models produced consistent classifications when given similar input conditions.

IV. Conclusions

The results of this study show that computational methods can be used to predict water quality using categories of physicochemical and environmental data. Traditional water quality assessment methods often rely on fixed threshold rules. While these rule-based systems are simple and interpretable, they cannot capture complex interactions between multiple environmental variables. In contrast, our AI model learns patterns from data and considers how features such as pH, dissolved oxygen, turbidity, nitrate, temperature, and daylight duration interact. The fine-tuned FLAN-T5 model produced water quality categorization that is consistent with the baseline model and also being able to generate natural language explanations to showcase what the water quality predicted. This indicates that explainable AI systems can maintain precision while improving clarity of the results. Overall, the results suggest that AI-based approaches can support comprehensible and consistent water quality assessment, providing a foundation for future systems to improve and assist environmental monitoring.

IV.I Limitations

One limitation of this study is the limited geographic scope, as the model was initially trained using water quality data primarily from the Austin area. Because environmental conditions, pollution sources, and seasonal patterns vary across regions, the model's performance may change when applied to new locations without additional training data. In addition, the water quality categories were derived from predefined thresholds, which may simplify complex environmental interactions. Because the dataset currently spans only four months and does not include severe storm conditions, the model's ability to predict nitrate fluctuations related to heavy rainfall is limited. Ongoing data collection will strengthen predictive accuracy and generalizability over time.

IV.II Future Projects

In the future, we hope to expand this project beyond the Austin area by first implementing the system in additional regions across Texas, such as Dallas and Houston. After validating the model's performance across multiple cities, we plan to further expand the system to other states in order to evaluate its scalability and adaptability to different environmental conditions.

V. Acknowledgement

We would like to thank expert Christina Bryant for her insights and knowledge that she provided in our initial research on water quality metrics. Additionally, we would like to thank our teacher, Cecil Shive, for his support during our writing process.

VI. References

Checking your browser - reCAPTCHA. (2024). Kaggle.com.
<https://www.kaggle.com/code/imakash3011/water-quality-prediction-7-model/input>

Chen, C., & Hao, J. (2025). A Water Quality Prediction Model Based on Neural Network at Data-Scarce Sites. Water-Energy Nexus. <https://doi.org/10.1016/j.wen.2025.05.001>

google/flan-t5-base · Hugging Face. (n.d.). Huggingface.co.

<https://huggingface.co/google/flan-t5-base>

kudzu_admin. (2024, December 2). The Role of AI in Water Quality Monitoring. SAJDI.
<https://sajdi.com/the-role-of-ai-in-water-quality-monitoring/>

UCI Machine Learning Repository. (2022). Uci.edu.
<https://archive.ics.uci.edu/dataset/733/water+quality+prediction-1>

Zhao, L., Gkountouna, O., & Pfoser, D. (2019). Spatial Auto-regressive Dependency Interpretable Learning Based on Spatial Topological Constraints. ACM Transactions on Spatial Algorithms and Systems, 5(3), 1–28.
<https://doi.org/10.1145/3339823>

“Data.gov.” Data.gov, 2025, catalog.data.gov/dataset/water-quality-sampling-data.

“Docs | Open-Meteo.com.” Open-Meteo.com, 2026,
open-meteo.com/en/docs?latitude=30.2672&longitude=-97.7431&timezone=America%2FChicago&past_days=92&hourly=&models=best_match&daily=temperature_2m_max.

“Water Quality Sampling Data | Open Data | City of Austin Texas.” Data.austintexas.gov,
data.austintexas.gov/Environment/Water-Quality-Sampling-Data/5tye-7ray/about_data.

VII. Author

Suhani Rai is a high school senior in Texas. She plans to major in Computer Science.

Harshini Chandilvel is a high school senior in Texas. She plans to major in Mechanical Engineering.

Dristi Sharma is a high school senior in Texas. She plans to major in Electrical Engineering.