WATER QUALITY PREDICTION APPLICATION

Kartik Sharma

10/02/2024

ABSTRACT

Aim: Creation and execution of a Water Quality Prediction Application for determining water suitability for drinking and agricultural use. Two subscription models are proposed: a subscription-based approach aimed at small-scale farmers and gardeners, with monthly and yearly possibilities, and a pay-per-use model for intermittent users such as farmers and governments.

Method: Machine learning models are trained on nine input features to predict water portability, which is an important driver of water safety. Three ML models(Logistic Regression, Decision Trees, K-nearest neighbour) are applied to find water quality, and to get best accuracy stack ensemble is used with stacking all the three models.

Results: After evaluating specific models such as Logistic Regression (LR), Decision Tree (DT), and K-Nearest Neighbours (KNN) classifiers, accuracies of 0.51, 0.80, and 0.66 were obtained. However, using a stack ensemble technique to combine these models resulted in a much-increased accuracy of 0.92.

Conclusion: Finally, the Water Quality Prediction Application has proved its ability to perform the essential task of reliably predicting water quality. The application has demonstrated varied levels of accuracy by integrating machine learning models such as Logistic Regression (LR), Decision Tree (DT), and K-Nearest Neighbours (KNN), with LR scoring 0.51, DT scoring 0.80, and KNN scoring 0.66. However, using a stack ensemble technique, which combines these models, resulted in a substantially higher accuracy of 0.92. As a result, the Water Quality Prediction Application is an invaluable tool in addressing the critical need for precise water quality evaluation, with excellent prospects for improving decision-making processes in both the agricultural and public health sectors.

1. PROBLEM STATEMENT

Inadequate access to reliable water quality assessment tools is a significant challenge in ensuring drinking water safety and optimizing agricultural practices. Existing methods frequently lack the precision required for precise prediction, resulting in possible health concerns and inefficient resource use. As a result, there is an urgent need for a strong and accessible solution capable of accurately predicting water quality, particularly in various contexts such as agriculture, research, and public health. Addressing this need demands the creation of an innovative Water Quality Prediction Application that relies on advanced machine learning techniques to provide accurate and timely assessments, allowing stakeholders to make informed decisions about water usage while guarding both human health and agricultural productivity.

2. ASSESSMENT

2.1 Customer Need Assessment

a) Model for those who need latest assessment regularly/monthly

- Accessing trustworthy water quality evaluations can help to ensure crop and livestock safety.
- Optimize irrigation procedures to save water and increase agricultural productivity.
- Identify and manage potential water quality issues to help prevent soil and groundwater contamination.
- Improve decision-making for fertiliser and pesticide application using water quality data.
- Improve long-term planning and sustainability by tracking changes in water quality over time.

b) Model for those who want occasional assessment and report

- Obtain immediate access to water quality assessments for specific agricultural activities or research initiatives.
- Respond quickly to emergent water quality concerns to reduce the danger to crops, livestock, and public health.

- Monitor water quality on a regular basis to ensure compliance with regulatory requirements and standards.
- Conduct study into the effects of water quality on agricultural productivity, environmental sustainability, and public health.
- Access comprehensive data and analytical tools to help you make evidence-based policy and resource allocation decisions.

2.2 Market Need Assessment

a) Importance of Water Quality

- Ensures safe drinking water for the population, reducing the risk of waterborne diseases.
- Facilitates sustainable agricultural practices by optimizing irrigation and fertilization strategies based on water quality data.
- Supports regulatory compliance efforts by monitoring and maintaining water quality standards.
- Enhances public health initiatives by providing early warnings of potential water contamination events.

b) Challenges in Water Quality Management

- There is limited access to accurate and timely water quality tests, especially in rural and distant locations.
- Reliance on antiquated and manual monitoring methods, resulting in gaps in data collection and processing.
- There is a lack of awareness about the effects of water quality on human health, agriculture, and the environment.
- Insufficient resources and facilities for extensive water quality monitoring and control.
- It is difficult to comprehend complex water quality data and translate it into meaningful information for stakeholders.

c) Role of AI in Water Quality Prediction

- AI-driven solutions can automate data collecting and analysis procedures, allowing for real-time monitoring of water quality indicators.
- Machine learning algorithms can discover patterns and trends in water quality data, allowing for early detection of pollution occurrences.
- Computer vision technology can analyse satellite imagery and remote sensing data to assess water quality across wide areas.
- Predictive analytics models can estimate changes in water quality in response to environmental conditions and human activities.
- AI-enabled drones and sensors may collect comprehensive spatial and temporal data on water quality factors, hence improving monitoring efforts.

3 TARGET SPECIFICATIONS AND CHARACTERIZATION

The target customers for our water quality prediction application can be broadly categorized into two groups based on their usage preferences:

<u>Subscription-Based Model Customers:</u>

- Small-Scale Farmers: Individuals that practise small-scale agriculture or horticulture and rely on correct water quality tests to maximise crop yield and quality.
- Gardeners: People who are interested in gardening or landscaping and want trustworthy information on water quality to ensure their plants' health and vitality.

<u>Customers who use the pay-per-use model include:</u>

- Farmers Who Need Occasional evaluations: Agricultural professionals who do not require continuous monitoring but want on-demand water quality evaluations for specific objectives like crop irrigation or animal management.
- Government Agencies and Researchers: Organisations and individuals who monitor the environment, conduct research, and formulate policies, and use water quality data to make scientific decisions.

Our model has further applications, including partnerships with manufacturers of water treatment equipment.

a) Partnership with other useful Firms:

Collaborate with water treatment equipment manufacturers to integrate our water quality prediction programme into their products, thereby offering end users with full water management solutions. Revenue can be generated through commissions on equipment sales enabled through our platform, providing added value to both manufacturers and customers.

b) Data Monetisation in Research and Education:

Utilise anonymised and aggregated data obtained from various clients to create comprehensive datasets appropriate for study. Provide researchers and educational institutions with access to this data, allowing them to conduct scientific investigations and educational efforts on water quality and the environment. Ensure that data privacy and security policies are in place, and acquire clients' explicit consent before using their data for research and educational purposes.

4 EXTERNAL SEARCH

Dataset used for the ML Product building is taken from online site.

Number of rows: 3276
Number of columns: 10

After eliminating the null rows,

Updated number of rows: 2011
Updated number of columns: 10

TABLE 1: Dataset Description

Parameter	**Description**	**WHO Recommended Limit**	**Dataset Range**
pH	Acid-base balance indicator	6.5 - 8.5	6.52 - 6.83
Hardness	Caused by calcium and magnesium salts	-	Varies
Total Dissolved Solids (TDS)	Indicates water mineralization	< 500 mg/L, Max 1000 mg/L	Varies
Chloramines	Disinfectant in water treatment	Up to 4 mg/L	Varies
Sulphates	Naturally occurring substances	-	3 - 1000 mg/L
Conductivity (EC)	Electrical conductivity indicator	≤ 400 µS/cm	Varies
Total Organic Carbon (TOC)	Measures carbon in organic compounds	< 2 mg/L for treated water, < 4 mg/L for source water	Varies
Trihalomethanes (THMs)	Chemicals in water treated with chlorine	Up to 80 ppm	Varies
Turbidity	Quantity of solid matter in suspended form	≤ 5.00 NTU	0.98 NTU
Potability	Water safety indicator	1 (Safe), 0 (Not Safe)	0 or 1

5 BENCH MARKING ALTERNATE PRODUCTS

Traditional Water Quality Test Kits:

Methodology: These kits often need manual testing of water samples with chemical reagents to determine pH, dissolved oxygen, and turbidity.
Pros: Reasonably priced, ideal for on-site testing.
Cons: Limited scope, time-consuming, and requires skill to accurately evaluate results.

Online Water Quality Monitoring Systems:

Methodology: These systems use sensors and data loggers to continually monitor water quality indicators in real time, with the results delivered to an internet platform for analysis and visualisation.
Pros: Offers real-time data insights and automatic notifications for abnormal conditions.
Cons: High initial investment, ongoing maintenance requirements, and sensor deployment.

6 BUSINESS MODEL (MONETIZATION IDEA)

- SUBSCRIPTION-BASED MODEL
  (Monthly And Annual Subscriptions)

a) Allow consumers to subscribe to the water quality prediction app on a monthly or year basis.
b) Monthly memberships offer flexibility to users who may only need access to the service on occasion.
c) Yearly memberships are cost-effective and convenient for consumers that require continuing access to water quality projections.
d) Incentivize people to subscribe for extended periods of time by offering discounts and promotions.
e) Offer substantial discounts on annual subscriptions or bundle them with other connected services or goods.

- PAY-PER-USE MODEL

a) Set a fixed pricing for basic usage, allowing consumers to receive a set amount of water quality predictions at a standard rate.
b) Define consumption thresholds based on the number of predictions or the size of the water body under consideration.
c) Implement variable pricing for any additional usage over the original allocation, charging consumers based on the volume or complexity of the water body being analysed.
d) Apply differential pricing based on water body substance, geographic location, or environmental circumstances.

- PARTNERSHIP WITH OTHER FIRMS

a) Collaborate with water treatment equipment manufacturers to incorporate the water quality prediction software into their products.
b) Revenue can be generated by commissions on equipment sales facilitated by the app, which adds value to both manufacturers and customers.
c) Offering data licencing agreements to academics, educational institutions, and government agencies allows you to monetize the aggregated and anonymized data obtained from users.
d) Form relationships with research organisations to conduct collaborative studies and initiatives that take advantage of the app's data and analytics features.
e) Investigate options for sponsored content and targeted advertising within the app, tailored to users' interests and needs in water quality management and agriculture.

Our water quality prediction app aims to generate sustainable revenue by utilizing subscription-based and pay-per-use monetization models, strategic partnerships, and additional revenue streams. It also offers valuable services to users across industries and sectors.

7 CONCEPT GENERATION

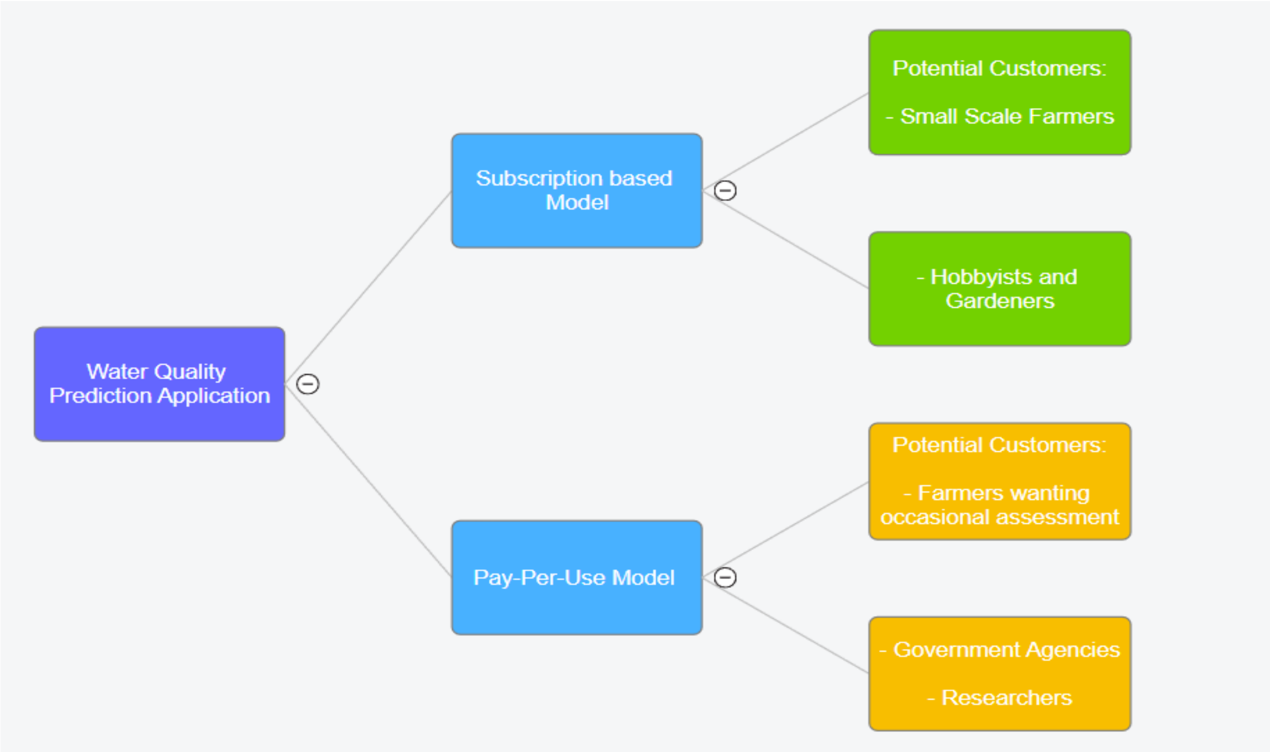
Concept generation was very easy. It has a high possibility nowadays that if u are drinking water from tap/handpump then that water is contaminated. Sometimes u realize this by drinking the water, by his taste or smell, but sometimes you can't detect due to no smell or flavour. In this type of case, it is very important to either stop the contaminated source or to make it clean and drinkable by using preferred methods.

8 CONCEPT DEVELOPMENT

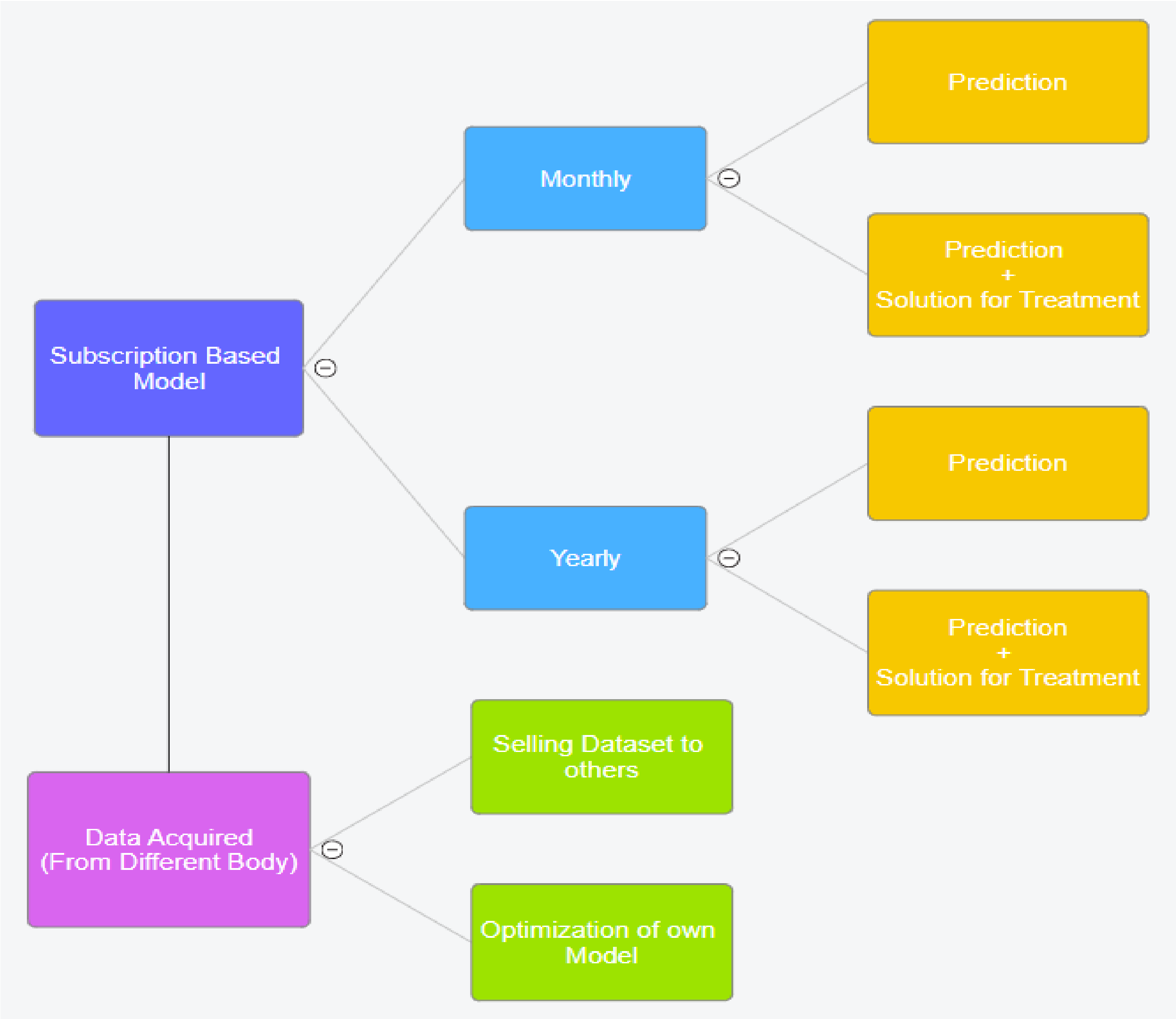
The concept for the water quality prediction application was developed through iterative market research and technical exploration. It entailed recognising the urgent need for accessible and accurate water quality assessment tools, analysing potential users' preferences and requirements, and utilising advances in machine learning and data analytics to create a scalable and user-friendly solution. Finally, schemas, strategic partnerships, and a vital tool for water management and agriculture will be used to monetize the offering.

9 FINAL PRODUCT PROTOTYPE WITH SCHEMATIC DIAGRAM

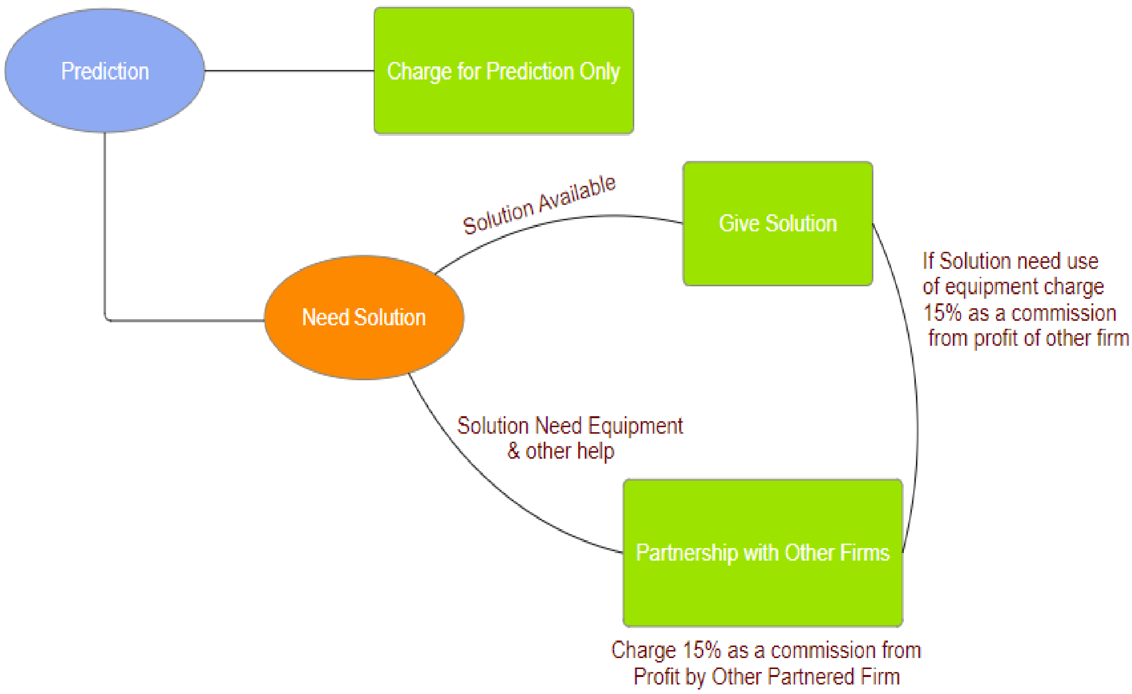
PRODUCT



MODELS


Subscription Based Model
Monthly
Yearly
Prediction
Prediction + Solution for Treatment
Prediction
Prediction + Solution for Treatment
Data Acquired (From Different Body)
Selling Dataset to others
Optimization of own Model

SOME OTHER MONETIZATION WAYS



10 PRODUCT DETAILS

How does it work?

The water quality prediction software uses advanced machine learning algorithms to analyse a variety of input data, including pH, dissolved oxygen, turbidity, and ambient variables. These algorithms use past water quality data to estimate the quality of water for a variety of applications, including agricultural and drinking. Users enter important criteria and obtain real-time forecasts and actionable insights via an intuitive interface. The programme continuously refines its forecasts in response to new data inputs, assuring long-term accuracy and reliability.

In addition to predictive capabilities, the app benefits from strong collaborations with water treatment equipment manufacturers. These collaborations provide consumers with access to a comprehensive array of solutions, including water quality monitoring and treatment equipment. This seamless integration improves the app's usefulness by offering end-to-end solutions for water quality control.

Data Source?

The data required to train the model, and predict water quality are –

ph: pH of water (0 to 14).

Hardness: Capacity of water to precipitate soap in mg/L.

Solids: Total dissolved solids in ppm.

Chloramines: Number of Chloramines in ppm.

Sulphate: Number of Sulphates dissolved in mg/L.

Conductivity: Electrical conductivity of water in μS/cm.

Organic carbon: Amount of organic carbon in ppm.

Trihalomethanes: Amount of Trihalomethanes in μg/L.

Turbidity: Measure of light emitting property of water in NTU.

Potability: Indicates if water is safe for human consumption. Potable -1 and Not potable

Algorithms Used?

There are 3 ML models used that are, K-nearest-neighbour, Decision trees, Logistic Regression. And in the last to improve the accuracy of predictions in the model, stack ensembled is used to stack all the 3 models and improve accuracy.

11 CODE IMPLEMENTATION

```
import pandas as pd
import numpy as np
```

```
[2] from google.colab import drive
    drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
[3] df = pd.read_csv("water_quality_stats.csv")
    df.head()
```

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0

```
[4] print("Number of rows:", df.shape[0])
    print("Number of columns:", df.shape[1])
```

```
Number of rows: 3276
Number of columns: 10
```

```
df.isnull().sum()
```

```
ph                 491
Hardness             0
Solids               0
Chloramines          0
Sulfate            781
Conductivity         0
Organic_carbon       0
Trihalomethanes    162
Turbidity            0
Potability           0
dtype: int64
```

```
[ ] # Since the info like ph,sulfate,trihalomethanes is very sensitive so we cant temper these values
    # So, we need to drop all these values
```

```
[6] df= df.dropna()
```

```
[7] print("Updated number of rows:", df.shape[0])
    print("Updated number of columns:", df.shape[1])
```

```
Updated number of rows: 2011
Updated number of columns: 10
```

```python
# Potability indicates if water is safe for human consumption where 1 means Potable(Drinkable) and 0 means Not potable.
df.Potability.value_counts()
```

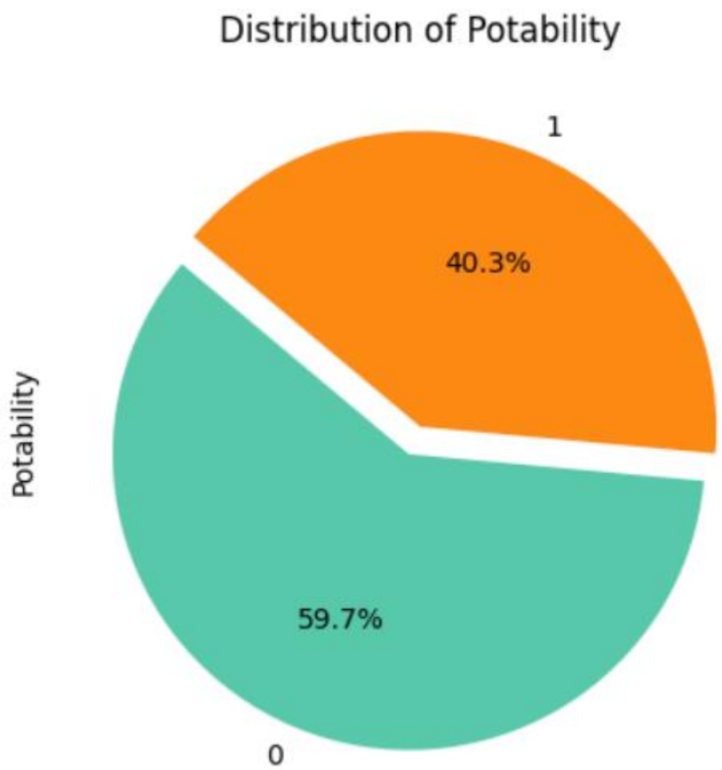
```
0    1200
1     811
Name: Potability, dtype: int64
```

[10]
```python
import matplotlib.pyplot as plt
import seaborn as sns

df['Potability'].value_counts().plot(kind='pie', autopct='%1.1f%%', colors=['#66c2a5', '#fc8d62'],
                                      explode=(0.1, 0), startangle=140).set_title('Distribution of Potability')
```

```
Text(0.5, 1.0, 'Distribution of Potability')
```

```python
plt.figure(figsize=(15, 9))
sns.heatmap(df.corr(), annot=True, cmap='inferno')
plt.title('Correlation Heatmap')
```

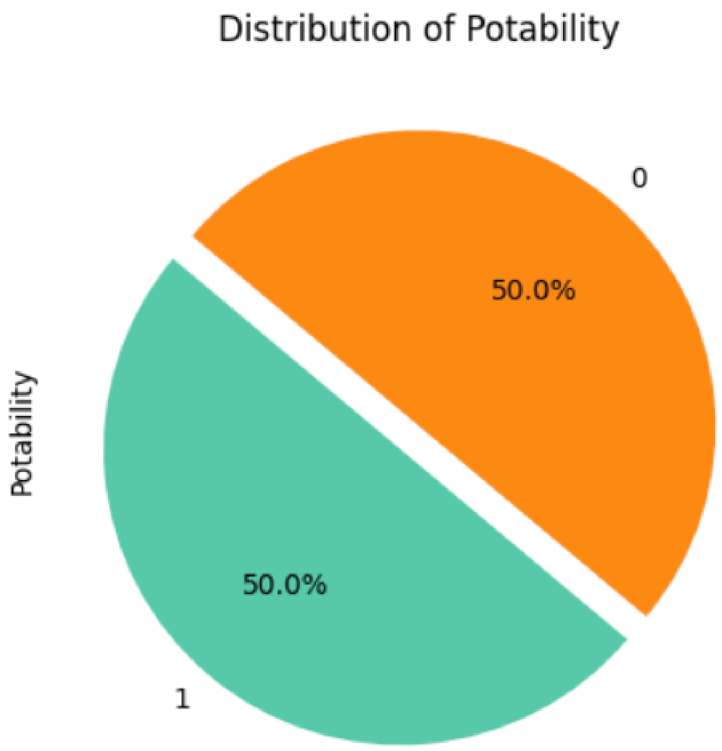
```
Text(0.5, 1.0, 'Correlation Heatmap')
```



```
[11] from sklearn.utils import resample, shuffle
     zero = df[df['Potability'] == 0]
     one = df[df['Potability'] == 1]
     one_upsampled = resample(one, replace=True, n_samples=len(zero), random_state=42)
     df = pd.concat([zero, one_upsampled])
     df = shuffle(df, random_state=42)
```

```
df['Potability'].value_counts().plot(kind='pie', autopct='%1.1f%%', colors=['#66c2a5', '#fc8d62'],
                                     explode=(0.1, 0), startangle=140).set_title('Distribution of Potability')
```

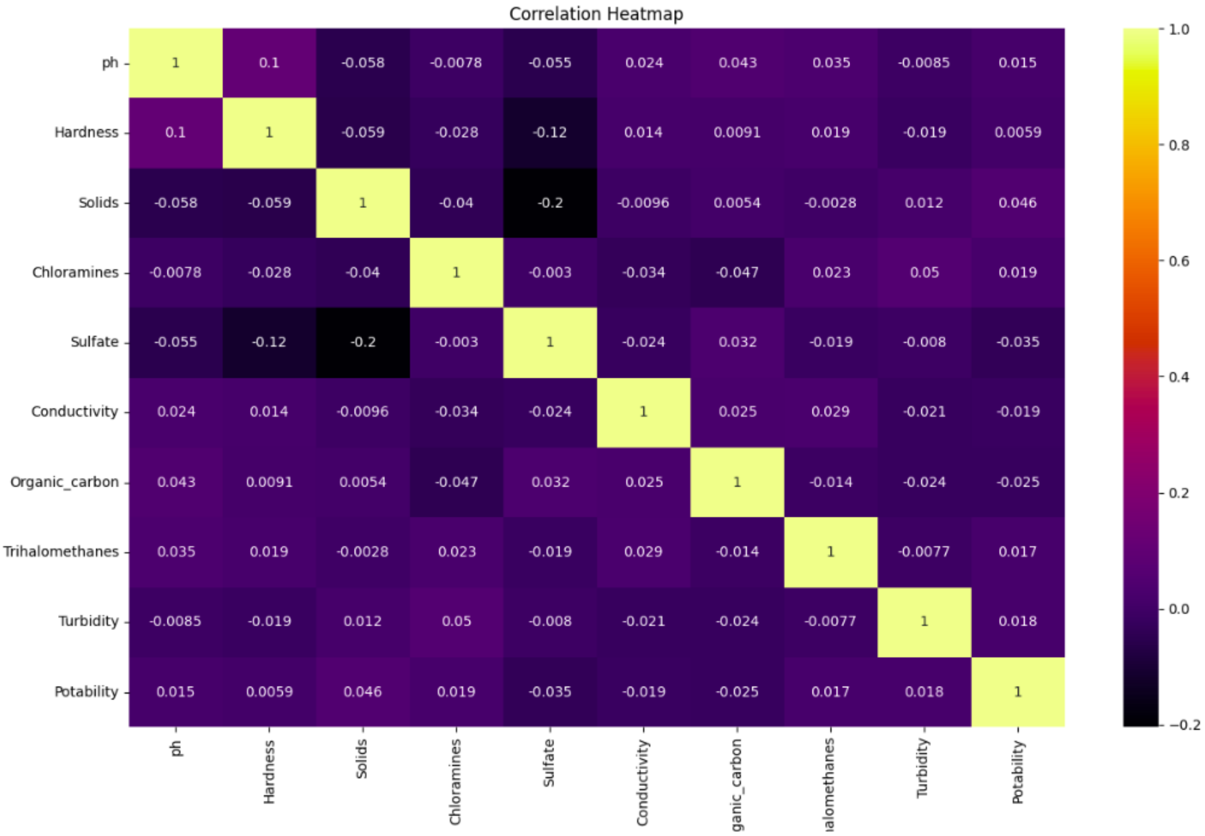
```
Text(0.5, 1.0, 'Distribution of Potability')
```



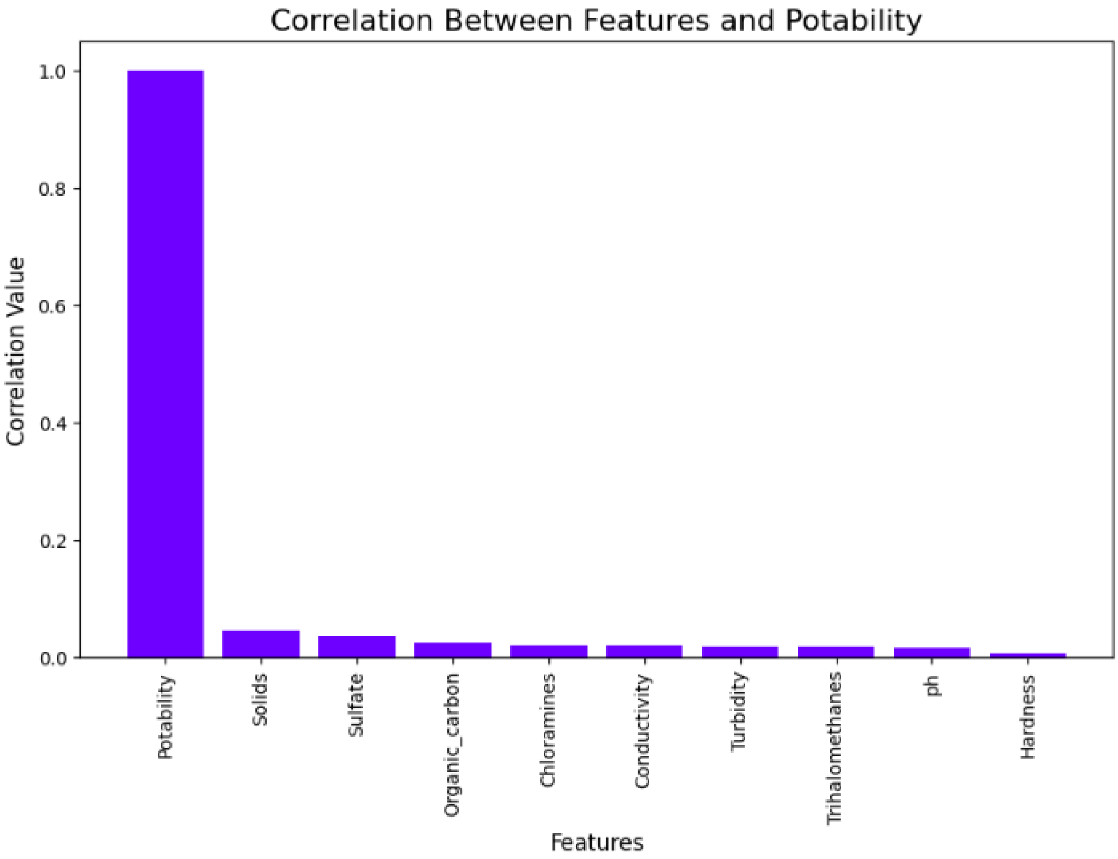

```
[13] plt.figure(figsize=(15, 9))
     sns.heatmap(df.corr(), annot=True, cmap='inferno')
     plt.title('Correlation Heatmap')
```

```
Text(0.5, 1.0, 'Correlation Heatmap')
```

```
Text(0.5, 1.0, 'Correlation Heatmap')
```




```python
import matplotlib.pyplot as plt
import pandas as pd

corr_values = df.corr().abs()['Potability'].sort_values(ascending=False)

plt.figure(figsize=(10,6))
plt.bar(corr_values.index, corr_values.values, color='blue')

plt.title('Correlation Between Features and Potability', fontsize=16)
plt.xlabel('Features', fontsize=12)
plt.ylabel('Correlation Value', fontsize=12)

plt.xticks(rotation=90)

# display plot
plt.show()
```




```
X = df.iloc[:, :-1].values
y = df.iloc[:, -1].values
```

[16]
```
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
X[:-1] = sc.fit_transform(X[:-1])
```

[17]
```
df.head()
```

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
3124	7.777665	233.761579	16780.116147	6.123297	323.538055	520.285094	17.794741	60.343891	4.683335	1
2772	8.014183	244.120098	30566.767504	7.714447	307.987458	309.930428	22.641598	61.578461	3.417076	1
2208	8.383762	156.951865	21923.874085	7.656831	379.541641	364.794883	11.773964	62.855780	3.585737	0
356	7.757270	213.048445	25259.780549	7.635153	363.684814	543.528799	15.254021	107.189584	4.165432	1
3250	7.371914	148.193698	42059.380417	7.966710	324.546262	544.848432	17.166504	62.677756	4.338957	1

[ ]
```
from sklearn.model_selection import train_test_split
```

[ ]
```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size = 0.1)
```

```python
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier

from sklearn.model_selection import RandomizedSearchCV, GridSearchCV
```

```python
lr = LogisticRegression(random_state=42)

knn = KNeighborsClassifier()

dt = DecisionTreeClassifier()

para_knn = {'n_neighbors':np.arange(1, 50)}
grid_knn = GridSearchCV(knn, param_grid=para_knn, cv=5)

para_dt = {'criterion':['gini','entropy'],'max_depth':np.arange(1, 50), 'min_samples_leaf':[1,2,4,5,10,20,30,40,80,100]}
grid_dt = GridSearchCV(dt, param_grid=para_dt, cv=5)
```

```python
classifiers = [('Logistic Regression', lr),('K Nearest Neighbours', knn),
               ('Decision Tree', dt)]
```

```python
from sklearn.metrics import accuracy_score

for classifier_name, classifier in classifiers:

    classifier.fit(X_train, y_train)
    y_pred = classifier.predict(X_test)
    accuracy = accuracy_score(y_test,y_pred)

    print(f"{classifier_name} : {accuracy:.2f}")

print("\n")
```

```
Logistic Regression : 0.50
K Nearest Neighbours : 0.64
Decision Tree : 0.74
```

```python
from sklearn.metrics import classification_report

classifiers = [('Logistic Regression', lr), ('K Nearest Neighbours', knn), ('Decision Tree', dt)]

for clf_name, clf in classifiers:
    print(f"Classification Report for {clf_name}:")
    print("\n")
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)
    print(classification_report(y_test, y_pred))
    print("\n")
```

```
Classification Report for Logistic Regression:

              precision    recall  f1-score   support

           0       0.53      0.43      0.47       127
           1       0.47      0.58      0.52       113

    accuracy                           0.50       240
   macro avg       0.50      0.50      0.50       240
weighted avg       0.51      0.50      0.50       240


Classification Report for K Nearest Neighbours:

              precision    recall  f1-score   support

           0       0.69      0.59      0.64       127
           1       0.60      0.70      0.65       113

    accuracy                           0.64       240
   macro avg       0.65      0.64      0.64       240
weighted avg       0.65      0.64      0.64       240


Classification Report for Decision Tree:

              precision    recall  f1-score   support

           0       0.85      0.66      0.74       127
           1       0.70      0.87      0.77       113

    accuracy                           0.76       240
   macro avg       0.77      0.76      0.76       240
weighted avg       0.78      0.76      0.76       240
```

```python
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.ensemble import StackingClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score

X, y = make_classification(n_samples=1000, n_features=20, n_informative=10, n_redundant=5, random_state=42)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

lr = LogisticRegression(random_state=42)
knn = KNeighborsClassifier(n_neighbors=1)
dt = DecisionTreeClassifier(criterion='gini', max_depth=27, min_samples_leaf=1)

estimators = [('lr', lr), ('knn', knn), ('dt', dt)]
stacking = StackingClassifier(estimators=estimators, final_estimator=LogisticRegression())

stacking.fit(X_train, y_train)

y_pred = stacking.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy when combining all there models using stack ensemble:", accuracy)
```

```
Accuracy when combining all there models using stack ensemble: 0.925
```

```
estimators = [('lr', lr), ('knn', knn)]
stacking = StackingClassifier(estimators=estimators, final_estimator=LogisticRegression())

stacking.fit(X_train, y_train)

y_pred = stacking.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy when combining KNN & LR:", accuracy)
```

```
Accuracy when combining KNN & LR: 0.915
```

```
[ ] estimators = [('lr', lr), ('dt', dt)]
stacking = StackingClassifier(estimators=estimators, final_estimator=LogisticRegression())

stacking.fit(X_train, y_train)

y_pred = stacking.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy when combining DT & LR", accuracy)
```

```
Accuracy when combining DT & LR 0.88
```

```
estimators = [('knn', knn), ('dt', dt)]
stacking = StackingClassifier(estimators=estimators, final_estimator=LogisticRegression())

stacking.fit(X_train, y_train)

y_pred = stacking.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy when combining KNN & DT", accuracy)
```

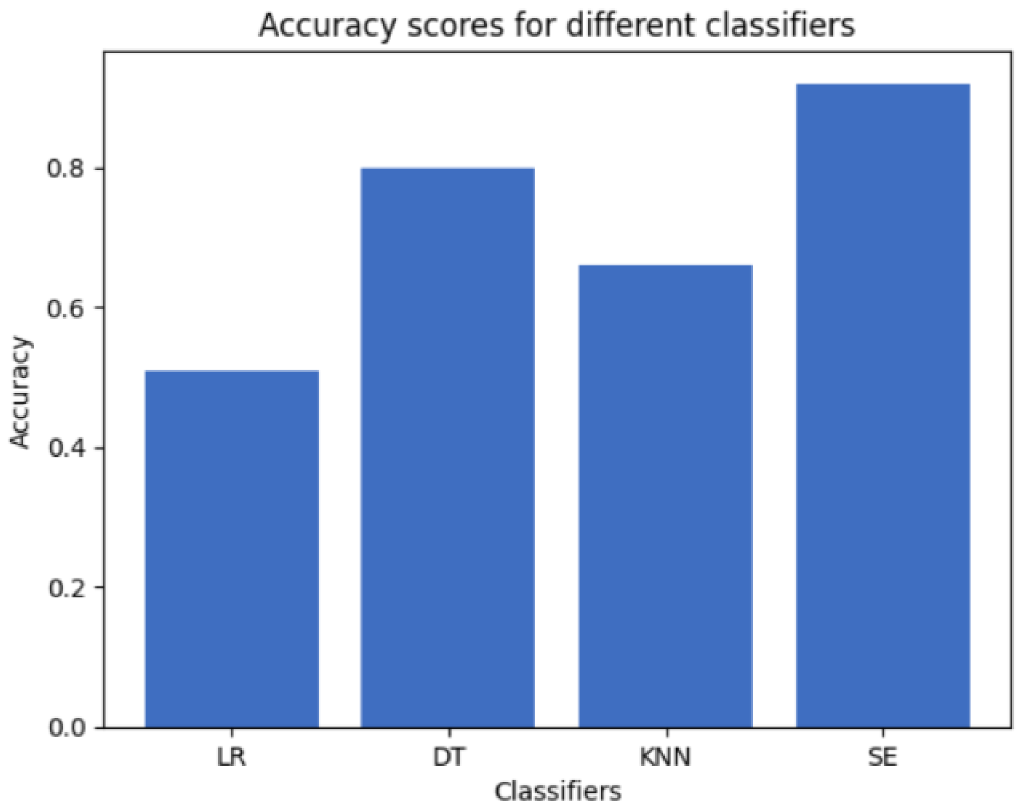
```
Accuracy when combining KNN & DT 0.915
```

Hence, we are getting best accuracy when all the three models are stacked

```python
from sklearn.metrics import classification_report

y_pred = stacking.predict(X_test)

report = classification_report(y_test, y_pred)

print("Classification report:")
print(report)
```

```
Classification report:
              precision    recall  f1-score   support

           0       0.91      0.92      0.92       102
           1       0.92      0.91      0.91        98

    accuracy                           0.92       200
   macro avg       0.92      0.91      0.91       200
weighted avg       0.92      0.92      0.91       200
```

```python
import matplotlib.pyplot as plt

classifiers = ['LR','DT', 'KNN','SE']
accuracies = [0.51, 0.80, 0.66, 0.92]

fig, ax = plt.subplots()
ax.bar(classifiers, accuracies)

ax.set_xlabel('Classifiers')
ax.set_ylabel('Accuracy')
ax.set_title('Accuracy scores for different classifiers')

plt.show()
```



GitHub link to the code implementation

https://github.com/Kartik-Sharma2205/Water-Quality-Prediction-Application

12 CONCLUSION

The creation of the water quality prediction application is a big step towards meeting the pressing demand for accessible and accurate water quality assessment tools. We developed a comprehensive solution that enables customers in a variety of industries, including agriculture, environmental management, and public health, through thorough idea development, market research, and technology innovation.

The application uses advanced machine learning algorithms to examine various input characteristics and predict water quality with great accuracy. Its user-friendly interface and adjustable subscription models enable quick access to real-time predictions and actionable insights, allowing for more educated water management decisions. In addition, strategic relationships with water treatment equipment manufacturers add to the application's value proposition by offering complete water quality management solutions. By incorporating equipment provision into our service offerings, we demonstrate our commitment to assisting users in protecting water resources and promoting sustainability.

Finally, the water quality prediction application represents our commitment to using technology to benefit society. As we refine and expand its capabilities, we remain committed to our purpose of empowering people, groups, and communities to pursue safe and sustainable water management methods.