Vansh Nagpal

CSCE 580 – Artificial Intelligence

Professor Biplav Srivastava

November 26th, 2023

Water Potability Regression Chatbot

Project Description:

1. **Title:** Water Potability Classification Chatbot
2. **Key Idea:** The key idea of this project is to build a system that takes up-to-date time series data regarding water quality and answer decision questions like "Is the water from Buck Creek Near Longs, SC safe to drink?"
3. **Who will care when the project is done?**
   a. Public health experts, Environmental Protection Agency (EPA), and any other agency that has policies contingent on water quality will care about this project.
4. **Data Need:** Current time-series data from water sites that includes measured parameters accessed through in-real-time Rest API calls from the past 30 days [1]. These measured parameters include:
   i. pH, dissolved oxygen (mg/L). water hardness (mg/L)
   ii. dissolved solids (ppm), chloramines (ppm), sulfate (mg/L)
   iii. specific conductance/conductivity(µS/cm), organic carbon (mg/L)
   iv. trihalomethanes (ppm), turbidity (NTU), temperature (° C)
5. **Methods/Approach:** For my chatbot interface, I used Rasa open source, which was trained on large number of examples of expected conversation. I utilized Rasa custom

actions to implement the chatbot's decision making process, data API calls, and saving the conversation after the user is finished. For my decision process, I utilized two main methods and aggregated them by taking their average. I chose to take their average because I wanted to eliminate pessimistic and optimistic decisions from both as best as possible. Additionally, there was a lack of specific measured parameters in most water sites (chloramines, sulfates, organic carbon, and trihalomethanes). These parameters were contingent for Method 2. Therefore, in most cases a default value for the measured parameter was used, leading to an optimistic decision. By combining with the generally pessimistic Method 1, I got more reasonable results.

a. Method 1: Gaussian Combination of Confidence Scores from Different Parameters

  i. I initially chose measured parameters pH, dissolved oxygen, temperature, turbidity, and specific conductance as decision parameters for water quality. I expanded the list later due to the dataset used to train the model for method 2. I researched ranges for each of these parameters that would constitute the safety of that water (i.e., safe ranges) [2, 3, 6, 8, 9]. To elaborate further, if the specific water site's measured parameter fell inside that range, it would be classified as safe. To clarify, I did not assign a value of 1 (safe) if it is within the range and 0 (unsafe) if it is outside the range.

  ii. Instead, I am assigning a smooth continuous range of confidence to the measured parameter in the form of a gaussian function. This gaussian function would be valued at 0.9 at the left and right endpoints of the safe range and 1 at the middle of the range. The rest of the values taper to 0 smoothly from the middle of the gaussian, allowing for a smooth range of confidence from 0(unsafe) to 1(safe). I then took the average of all the confidences of all the measured parameters, which were a result of each parameter's gaussian confidence function. This average is Method 1's contribution to the overall confidence (i.e. final decision).

b. Method 2: Support Vector Machine (SVM) Classifier

   i. I wanted to implementing a machine learning based approach in my project, and decided to seek out a labeled dataset with 9 measured water parameters (pH, hardness, dissolved solids, chloramines, sulfate, conductance, organic carbon, trihalomethanes, and turbidity) as features and 0 (unsafe) and 1 (safe) as labels. Because there are more parameters here than were measured in the initial exploration phase for method, I had to expand my initial range of parameters that I acquired from the USGS Rest API. For some reason, most I found one on Kaggle with 3277 samples [4].

   ii. With an 80/20 train/test partition of the dataset, I experimented with different classification models, all implemented in the python library scikit-learn, including SVM Classifier (SVC), Random Forest, K-nearest neighbors, naïve bayes, Decision Tree, and Logistic Regression. Respectively, their accuracies were 70%, 68%, 65%, 65%, 64%, and 63% after hyper parameter tuning. The code leading to these results along with preliminary data cleaning and processing is implemented in a .ipynb source file in ./code_v1/ml_trials. I then saved this trained model as a .joblib file to be loaded for inference by the chatboat system. The inference of this trained model on a water site's parameters composed Method 2's contribution to the overall confidence (i.e., final decision)

6. **Evaluation: Comparison with ChatGPT**

   a. The results of project specific queries tested on my Water Quality Chatbot and on ChatGPT can be seen in the shared drive folder.

   b. Overall, ChatGPT behaves exactly as we'd expect: never giving a definitive answer. By avoiding giving a definitive answer on topics it is not confident in, GPT reduces trust issues. When asked to make decisions on if the water from a certain site is safe, it answers by saying there are "multiple factors to consider", so it is unknown without further inspection. When asked for measured data parameters, it responds by saying it does not have access to real time data, but instead points us in the direction of certain APIs, one of which my system actually uses.

   c. In contrast, my system responds with definitive answers to decision and data requests. The downside of definitive answers is that they may not always be right, and this leads to trust issues, which are discussed in the following section. My chatbot employs a decision classifier based on data from a real time API.

7. **Trust Issues**

   a. The trust issues with this system are numerous. Primarily, a false positive (i.e., classifying unsafe water as potable) could cause the consumption of unsafe water. This could lead to health complications or a negative environmental impact.

Another trust issue with this system is the validity of the data. Of course, this has nothing to do with the methods I implemented; however, if the data from USGS or the Kaggle dataset are not ‘correct’ or are noisy, then the decision from the chatbot can be skewed the wrong way.

8. Achievement

a. I was able to create a system that can respond to user queries based on a model (method 2) that was trained with fair accuracy and a . I made a lot of assumptions in my decision process, including handling missing values in the Kaggle dataset in method and assuming the water parameters contributed to the confidence of the water safety in a smooth gaussian manner in method 1. However, if I was given a larger dataset with less missing values, I would be able to train a better classifier model. Additionally, if more water sites collected more parameters, I would have more confidence in the robustness of my system.

b. In the future, I plan to extend this project to a web application, likely using Flask or Django as my backend with React as my frontend.

9. Related Works

a. One work I researched explored the applications of machine learning for water quality in different settings [5]. This work seemed more of a review paper, rather than a new research contribution. Overall, it discussed how different deep learning models can be used to predict lost time series data like dissolved oxygen and other dissolved ions levels. This paper went a lot deeper than I did with my project, because they were trying to make datasets more robust with a water parameter predictor, while I was trying to make a water quality predictor.

b. Another work that I researched explored the application of real-time water quality sensing at a particular river-centric religious gathering in Haridwar, India [7]. This fascinating paper discussed a Weighted Arithmetic Water Quality Index (WAWQI). This approach places different weights on different measured parameters of the water. This differs from mine because I took an unweighted average, therefore giving all parameters equal importance, which is not realistic.

Works Cited

[1] "Daily Values Service Details." *Water Services Web*, United States Geological Survey, 21 Sept. 2023, waterservices.usgs.gov/docs/dv-service/daily-values-service-details/.

[2] *Dissolved Oxygen (DO) - Gov*, www.gov.nt.ca/sites/ecc/files/dissolved_oxygen.pdf. Accessed 30 Nov. 2023.

[3] Health, Deparment of. "Tracking - Ph and Conductivity and Water." *NM*, nmtracking.doh.nm.gov/environment/water/PHConductivity.html#:~:text=No%20state%20or%20national%20standard,your%20drinking%20water%20testedregularly. Accessed 30 Nov. 2023.

[4] Kadiwal, Aditya. "Water Quality." *Kaggle*, ADITYA KADIWAL, 25 Apr. 2021, www.kaggle.com/datasets/adityakadiwal/water-potability/data.

[5] Mengyuan Zhu, Jiawei Wang, Xiao Yang, Yu Zhang, Linyu Zhang, Hongqiang Ren, Bing Wu, Lin Ye, A review of the application of machine learning in water quality evaluation, Eco-Environment & Health, Volume 1, Issue 2, 2022

[6] *Ph | US EPA - U.S. Environmental Protection Agency*, www.epa.gov/caddis-vol2/ph. Accessed 30 Nov. 2023.

[7] Raychoudhury, Vaskar, et al. *Real-Time Water Quality Sensing at a Large Rivercentric Religious Gathering*.

[8] "Water Quality and Health: Review of Turbidity." *World Health Organization*, World Health Organization, www.who.int/publications/i/item/WHO-FWC-WSH-17.01. Accessed 30 Nov. 2023.

[9] "Which Water Temperature Is Best for Drinking?" *Wisewell AE*, wisewell.ae/blogs/news/which-water-temperature-is-best-for-drinking#:~:text=Throughout%20Europe%2C%20room%20temperature%20water,absorb%20the%20water%20and%20rehydrate. Accessed 30 Nov. 2023.