

HW 2

Part 1: MQTT

1. Write separate python scripts, for publishing data and to receive that data using MQTT.
2. You will write 2 separate publish scripts
 - one for temperature and humidity with numbers for temperature ranging between 50 to 52 F and numbers for humidity between 60 and 80 percentage.
 - the other publish file will be for rainfall with numbers ranging between 0 and 2 inches.
3. You will also write 2 separate subscriber scripts with
 - One subscriber subscribing to all the weather data
 - This script will start getting the data with `client.loop_start()` and when 50 data points are reached, it will stop subscribing using `client.loop_stop()`
 - The other subscriber will receive the temperate and humidity data and will `loop_forever`.

Part 2: Build the DQN Model for the Lunar Lander

1. From the Farama Gymnasium library choose the version **Lunar Lander – V3**. This is the version with continuous spaces and discrete actions.
2. To complete the exercise, you will do the following:
 - Build main loop where you will take actions,
 - update Q-Values,
 - populate the replay buffer,
 - Call the Q-Network update function
3. You will write a separate function for the NN training and build a simple architecture for that.
4. Once the model achieves its goals (based on details given in the gymnasium documentation) save the model.
5. Use the saved model (the Policy) to run 5 complete episodes with flawless landings.

https://gymnasium.farama.org/environments/box2d/lunar_lander/

Due data and grading

1. HW 2 will be due by October 5th midnight.
2. Grading criteria
 - All code is implemented as per instructions and runs error free
 - Questions given below are answered for each part
 - Code is well commented
3. Part 1 Questions:
 - What was your biggest takeaway from the MQTT exercise
 - What is different between looping forever and looping with start – stop? What would happen if you had a whole application and the MQTT code with loop_forever had to receive data , store it and in the later part of the same script, analyze that data?
4. Part 2 Questions:
 - Explain how DQN works in the application that you built? Answer should be specific to what you developed.