AIML 2025 A3

Debanjan Saha, 23607

April 8, 2025

Question 1

Submit the necessary code files and report the number of steps taken by the Breadth-First Search (BFS) algorithm for your assigned maze. Confirm if a path to the goal exists.

Answer 1

My SR Number is 23607.

All necessary files for the assignment implementation, including the code for the maze environment, Q-learning pickel files, training script, and the generated Q-table and policy files, are attached.

Included files are:

- 23607_Assignment3.ipynb
- Trained Q-table pickle files:
  `23607_enabled_1.pkl`, `23607_enabled_2.pkl`, `23607_disabled_1.pkl`, `23607_disabled_2.pkl`

A path to the goal exists in the generated maze.
The Breadth-First Search (BFS) algorithm found the shortest path in **38 steps**.

Question 2

Report the number of steps taken by your trained Q-learning agent to reach the goal for the four configurations (enabled/disabled boost, 2 reward configurations for each). Compare the results for the disabled configurations with the BFS path length found in Question 1. Reference the filenames used for saving the trained agents.

Answer 2

The trained Q-learning agents for the four different configurations were saved in pickle files using the specified naming convention based on my SR Number (23607):

- Trap-Boost Enabled:
  - `23607_enabled_1.pkl`
  - `23607_enabled_2.pkl`
- Trap-Boost Disabled:
  - `23607_disabled_1.pkl`
  - `23607_disabled_2.pkl`

Taking both `23607_disabled_1.pkl` and `23607_disabled_2.pkl`, both reaches the reward in **38** steps, which is in line with the said numbers of steps found in **BFS**.

Comparison: This number of steps (38) taken by the Q-learning agent in the disabled configurations is identical to the shortest path length found by the BFS algorithm (38 steps), indicating that the agent learned an optimal path under these simpler conditions.

Question 3

Provide visualizations (plots) of the optimal policy learned by the Q-learning agent for each of the four configurations.

Answer 3

The visualizations of the optimal policies learned by the agent for each configuration are shown in Figures 1, 2, 3, and 4. The arrows indicate the preferred action (highest Q-value) for each state according to the trained agent.

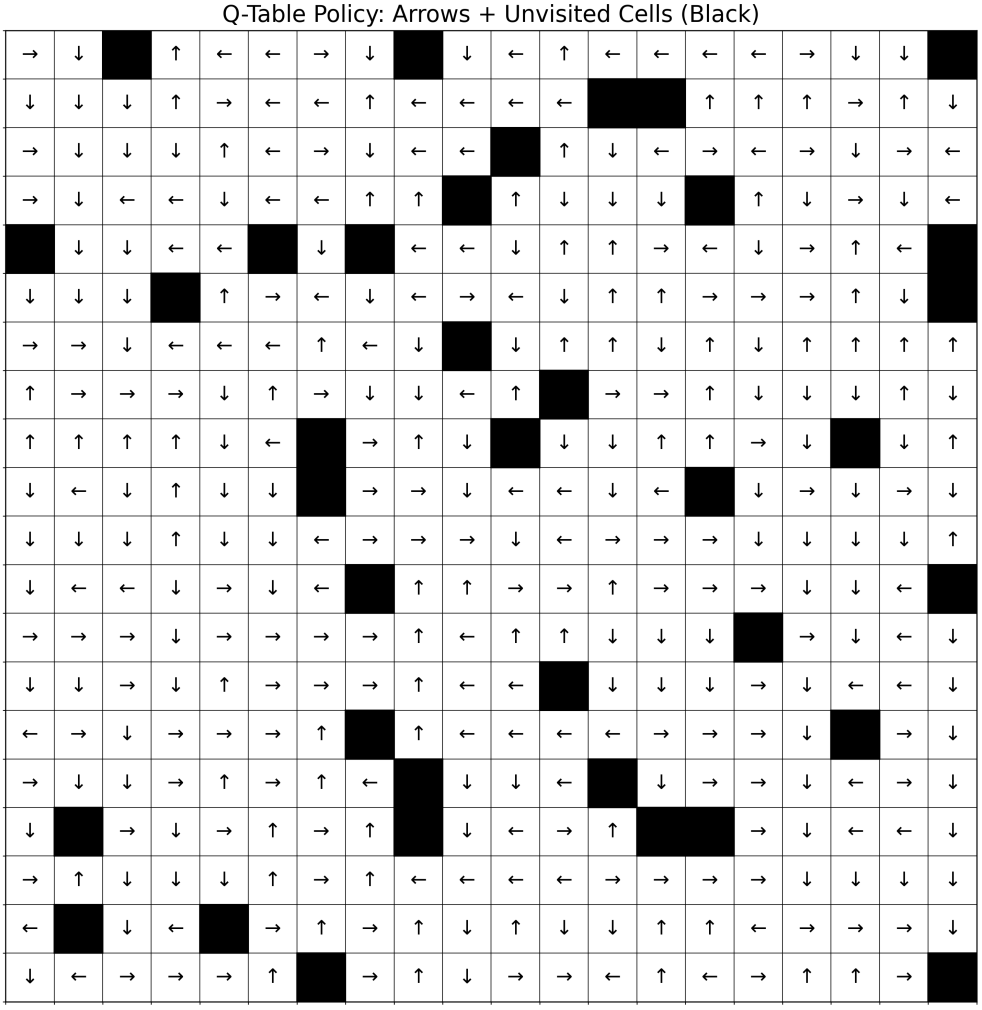

Figure 1: Optimal Policy with Trap-Boost Enabled corresponding to (`23607_enabled_1.pkl`)

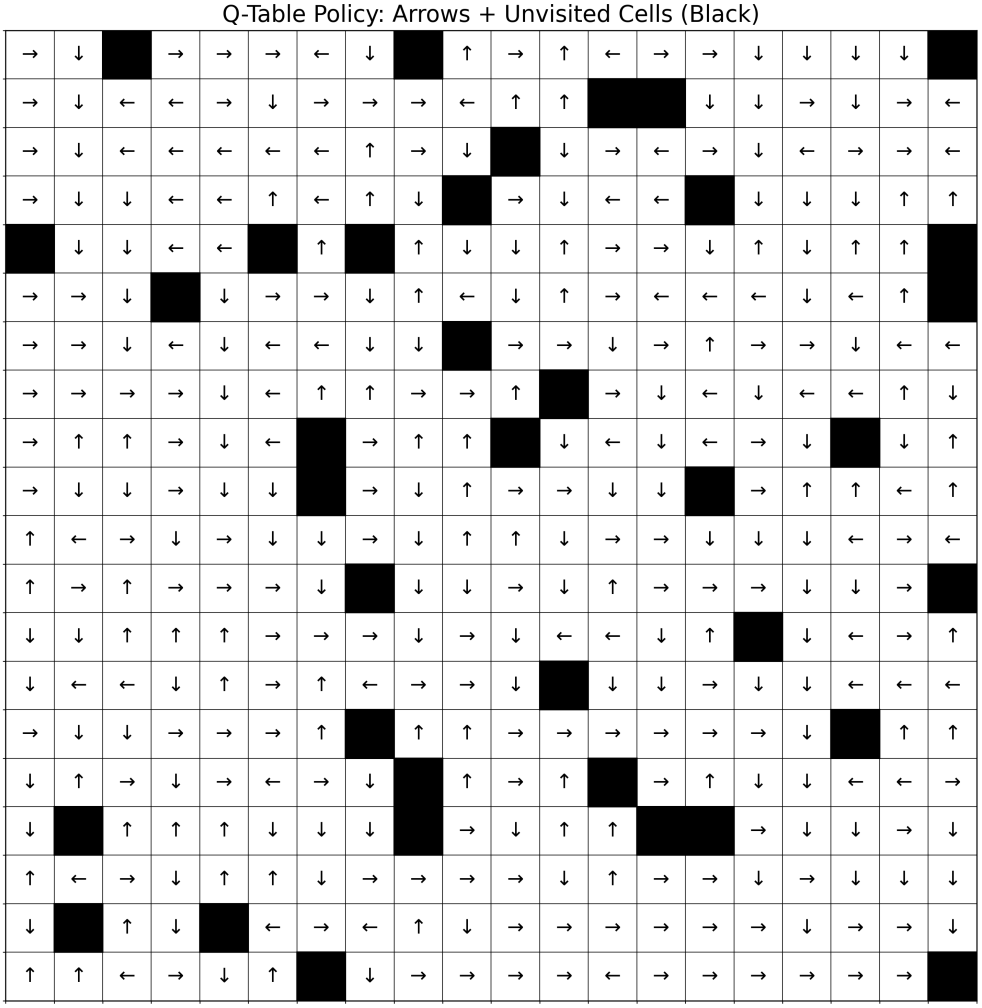

Figure 2: Optimal Policy with Trap-Boost Enabled corresponding to (`23607_enabled_2.pkl`)

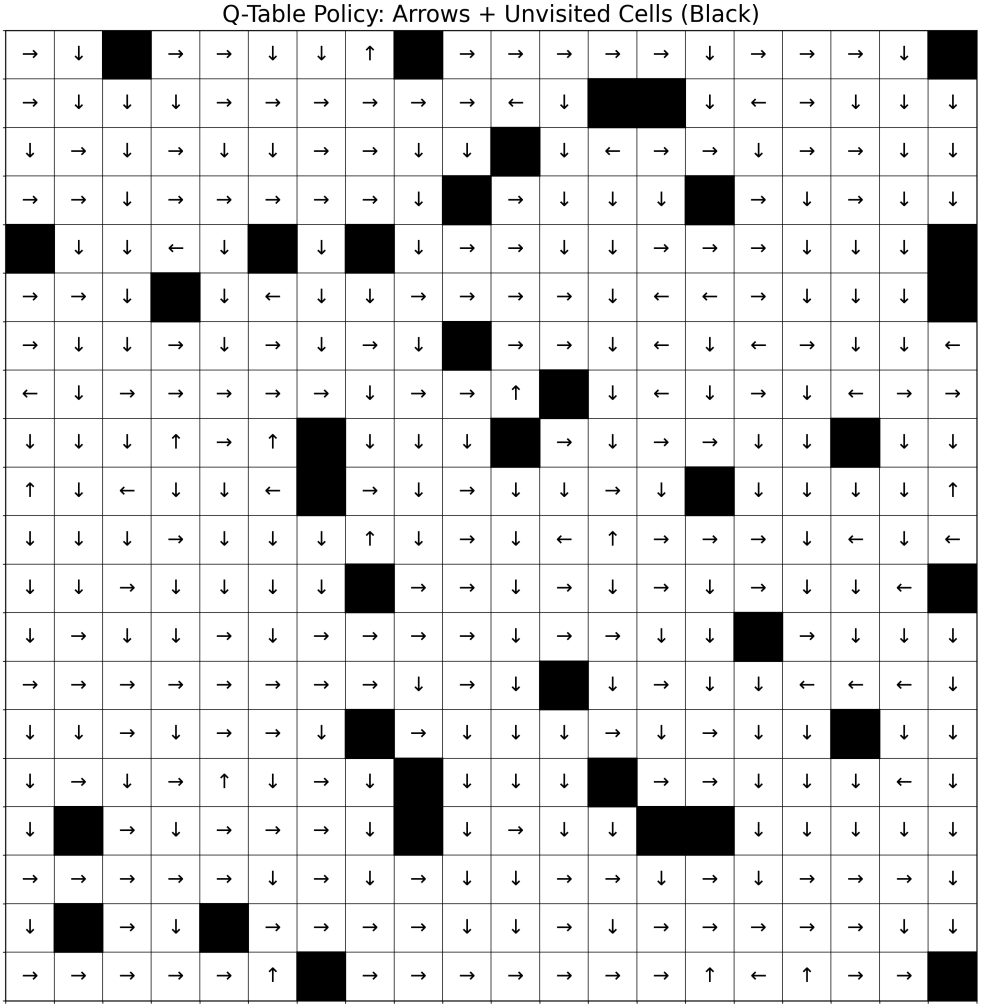

Figure 3: Optimal Policy with Trap-Boost Disabled corresponding to (`23607_disabled_1.pkl`)

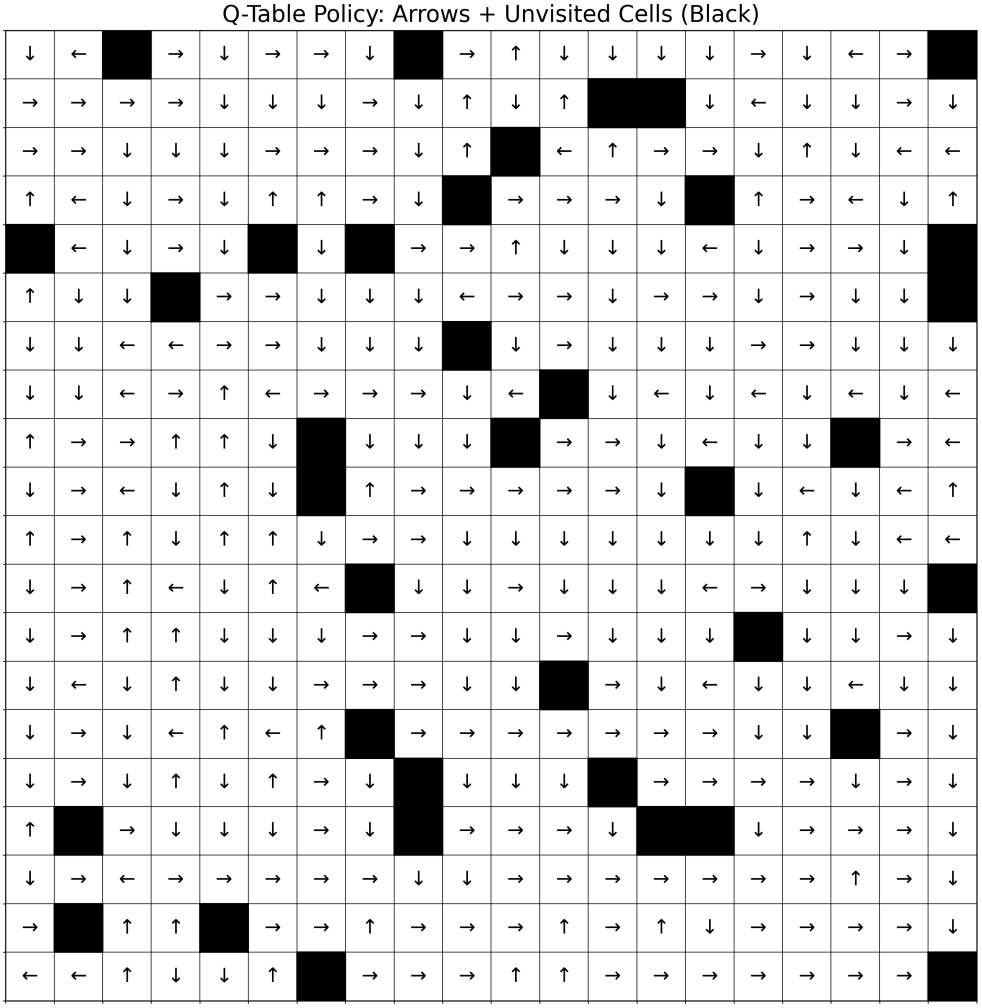

Figure 4: Optimal Policy with Trap-Boost Disabled corresponding to (`23607_disabled_2.pkl`)

Question 4

Perform manual calculations and show the Q-value updates for the first 5 steps of a new episode, starting from state (0,0), using your final trained Q-table for one of the configurations (e.g., disabled_1). Use the Q-learning parameters defined in your code (α, γ) and assume greedy actions are taken at each step for the calculation demonstration.

Answer 4

Parameters: $\alpha = 0.1$, $\gamma = 0.9$, $\epsilon = 0.7$, $R_{step} = -0.01$. Assume greedy actions are chosen. Actions: 0: Up, 1: Down, 2: Left, 3: Right.
Assumption: Greedy Selection for each new state selection, based on trained Q table values.

Step 1

Current State s_0: (0, 0), Index: $0 * 20 + 0 = 0$
Q-values $Q_0(s_0, \cdot)$: [3.1790, 3.4543, 3.0018, 4.9596]
Greedy Action a_0: argmax $Q_0(s_0, \cdot) = 3$ (Right)
Chosen Action: 3 (Right)
Next State s_1: (0, 0) + (0, 1) = (0, 1), Index: $0 * 20 + 1 = 1$
Reward r_0: -0.01
Q-values for s_1, $Q_0(s_1, \cdot)$: [2.4140, 5.0157, 3.9068, 2.1302]
Max Q-value for s_1: $\max_{a'} Q_0(s_1, a') = 5.0157$
TD Target: $r_0 + \gamma \max_{a'} Q_0(s_1, a') = -0.01 + 0.9 * 5.0157 = -0.01 + 4.5141 = 4.5041$
TD Error: TD Target - $Q_0(s_0, a_0) = 4.5041 - 4.9596 = -0.4555$
Update $Q_1(s_0, a_0)$: $Q_0(s_0, a_0) + \alpha \times$ TD Error $= 4.9596 + 0.1 * (-0.4555) = 4.9596 - 0.04555 = 4.91405$
New Q-table row for Index 0: [3.1790, 3.4543, 3.0018, 4.91405]

Step 2

Current State s_1: (0, 1), Index: 1
Q-values $Q_1(s_1, \cdot)$: [2.4140, 5.0157, 3.9068, 2.1302]
Greedy Action a_1: argmax $Q_1(s_1, \cdot) = 1$ (Down)
Chosen Action: 1 (Down)
Next State s_2: (0, 1) + (1, 0) = (1, 1), Index: $1 * 20 + 1 = 21$
Reward r_1: -0.01
Q-values for s_2, $Q_1(s_2, \cdot)$: [2.5172, 5.0658, 2.4145, 2.6029]
Max Q-value for s_2: $\max_{a'} Q_1(s_2, a') = 5.0658$
TD Target: $r_1 + \gamma \max_{a'} Q_1(s_2, a') = -0.01 + 0.9 * 5.0658 = -0.01 + 4.5592 = 4.5492$
TD Error: TD Target - $Q_1(s_1, a_1) = 4.5492 - 5.0157 = -0.4665$
Update $Q_2(s_1, a_1)$: $Q_1(s_1, a_1) + \alpha \times$ TD Error $= 5.0157 + 0.1 * (-0.4665) = 5.0157 - 0.04665 = 4.96905$
New Q-table row for Index 1: [2.4140, 4.96905, 3.9068, 2.1302]

Step 3

Current State s_2: (1, 1), Index: 21
Q-values $Q_2(s_2, \cdot)$: [2.5172, 5.0658, 2.4145, 2.6029]
Greedy Action a_2: argmax $Q_2(s_2, \cdot) = 1$ (Down)
Chosen Action: 1 (Down)
Next State s_3: (1, 1) + (1, 0) = (2, 1), Index: $2 * 20 + 1 = 41$
Reward r_2: -0.01
Q-values for s_3, $Q_2(s_3, \cdot)$: [2.60995, 2.9406, -0.6699, 5.1187]
Max Q-value for s_3: $\max_{a'} Q_2(s_3, a') = 5.1187$
TD Target: $r_2 + \gamma \max_{a'} Q_2(s_3, a') = -0.01 + 0.9 * 5.1187 = -0.01 + 4.6068 = 4.5968$
TD Error: TD Target - $Q_2(s_2, a_2) = 4.5968 - 5.0658 = -0.4690$
Update $Q_3(s_2, a_2)$: $Q_2(s_2, a_2) + \alpha \times$ TD Error $= 5.0658 + 0.1 * (-0.4690) = 5.0658 - 0.0469 = 5.0189$
New Q-table row for Index 21: [2.5172, 5.0189, 2.4145, 2.6029]

Step 4

Current State s_3: (2, 1), Index: 41
Q-values $Q_3(s_3, \cdot)$: [2.60995, 2.9406, -0.6699, 5.1187]
Greedy Action a_3: argmax $Q_3(s_3, \cdot) = 3$ (Right)
Chosen Action: 3 (Right)
Next State s_4: (2, 1) + (0, 1) = (2, 2), Index: $2 * 20 + 2 = 42$
Reward r_3: -0.01
Q-values for s_4, $Q_3(s_4, \cdot)$: [-1.6885, -1.5893, -1.5530, -1.6741]
Max Q-value for s_4: $\max_{a'} Q_3(s_4, a') = -1.5530$
TD Target: $r_3 + \gamma \max_{a'} Q_3(s_4, a') = -0.01 + 0.9 * (-1.5530) = -0.01 - 1.3977 = -1.4077$
TD Error: TD Target - $Q_3(s_3, a_3) = -1.4077 - 5.1187 = -6.5264$
Update $Q_4(s_3, a_3)$: $Q_3(s_3, a_3) + \alpha \times$ TD Error $= 5.1187 + 0.1 * (-6.5264) = 5.1187 - 0.65264 = 4.46606$
New Q-table row for Index 41: [2.60995, 2.9406, -0.6699, 4.46606]

Step 5

Current State s_4: (2, 2), Index: 42
Q-values $Q_4(s_4, \cdot)$: [-1.6885, -1.5893, -1.5530, -1.6741]
Greedy Action a_4: argmax $Q_4(s_4, \cdot) = 2$ (Left)
Chosen Action: 2 (Left)
Next State s_5: (2, 2) + (0, -1) = (2, 1), Index: $2 * 20 + 1 = 41$
Reward r_4: -0.01
Q-values for s_5, $Q_4(s_5, \cdot)$: [2.60995, 2.9406, -0.6699, 4.46606]
Max Q-value for s_5: $\max_{a'} Q_4(s_5, a') = 4.46606$
TD Target: $r_4 + \gamma \max_{a'} Q_4(s_5, a') = -0.01 + 0.9 * 4.46606 = -0.01 + 4.01945 = 4.00945$
TD Error: TD Target - $Q_4(s_4, a_4) = 4.00945 - (-1.5530) = 5.56245$
Update $Q_5(s_4, a_4)$: $Q_4(s_4, a_4) + \alpha \times$ TD Error $= -1.5530 + 0.1 * 5.56245 = -1.5530 + 0.556245 = -0.996755$
New Q-table row for Index 42: [-1.6885, -1.5893, -0.996755, -1.6741]