Build up Tic-tac-toe by Reinforcement Learning

Nguyen Hoang Quan
22014500

Algorithms

Q-Learning

Q-Learning is an off-policy algorithm that learns the optimal action-value function (Q-value) by maximizing the expected future reward.

SARSA

SARSA, an on-policy algorithm, learns the Q-value by following the current policy and updating it based on the actual chosen action.

Q-Value Update Rules

1

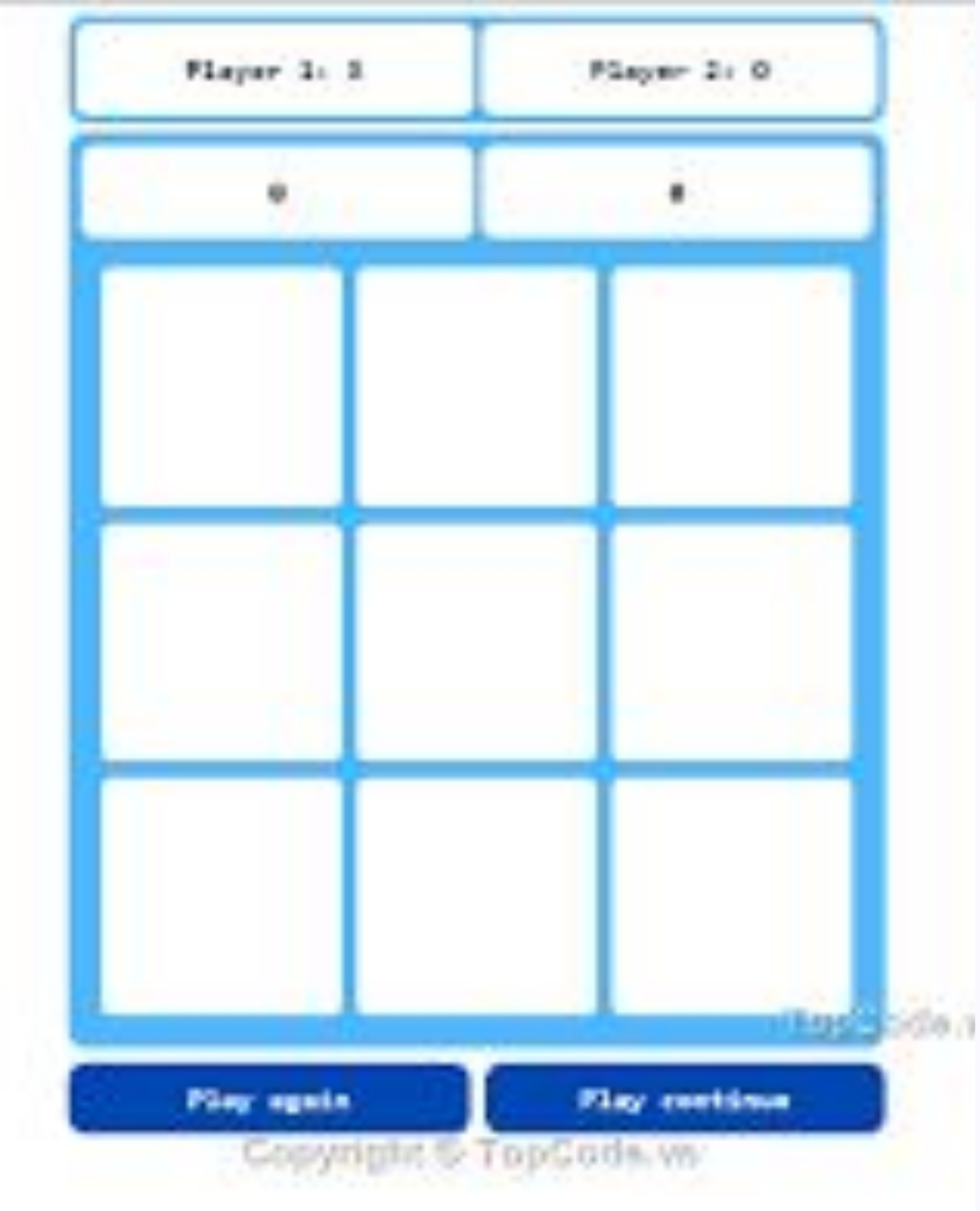
Q-Learning

$Q(s, a) = Q(s, a) + \alpha[R + \gamma * \max Q(s', a') - Q(s, a)]$

2

SARSA

$Q(s, a) = Q(s, a) + \alpha[R + \gamma * Q(s', a') - Q(s, a)]$



TopCode.vn
Copyright © TopCode.vn

Implementation Methodology

1

Environment Setup

The Tic-Tac-Toe environment was set up with a 3x3 grid, representing the game board.

2

Agent Training

Both agents were trained over 2000 episodes, with each episode comprising 10 games.

3

Reward System

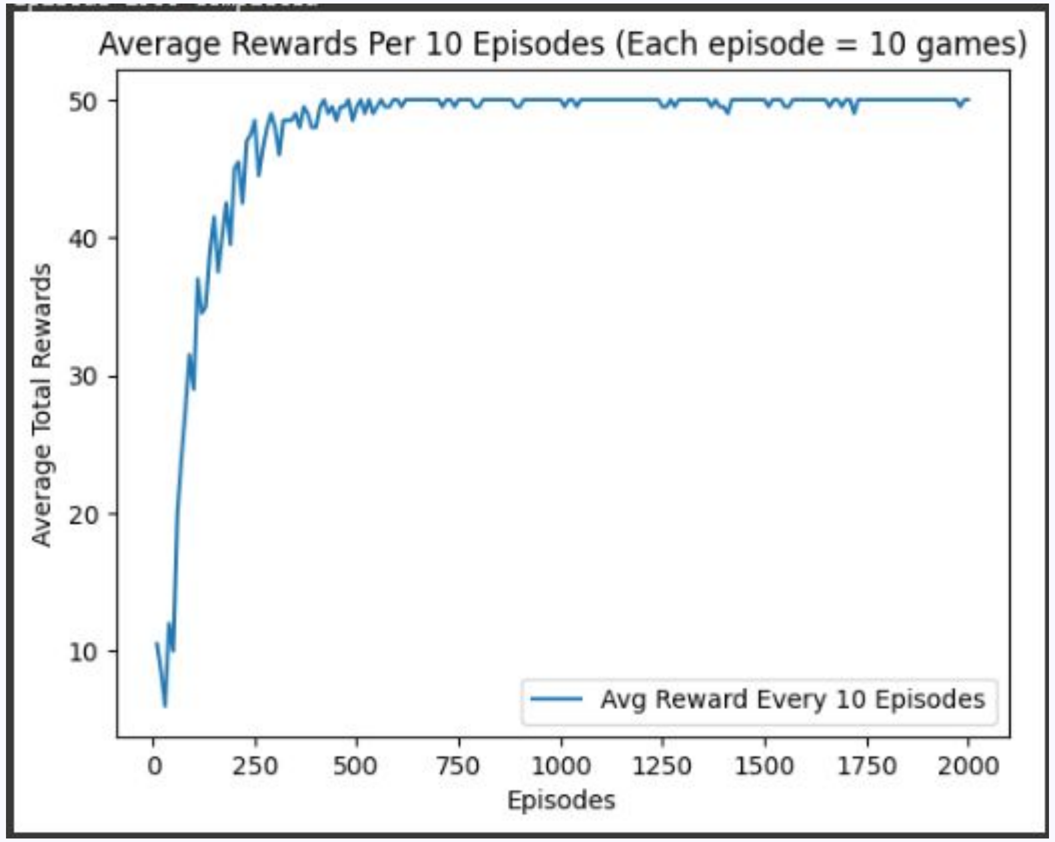
-State: list or array of 9 elements representing all the cells on the board (1:X, -1:O, 0:none).

- Action: represented by an index from 0 to 8.

- A reward system was implemented, with +5 for winning, -5 for losing, 0 for drawing, and -1 for invalid moves.

Averaging for Smoother Results

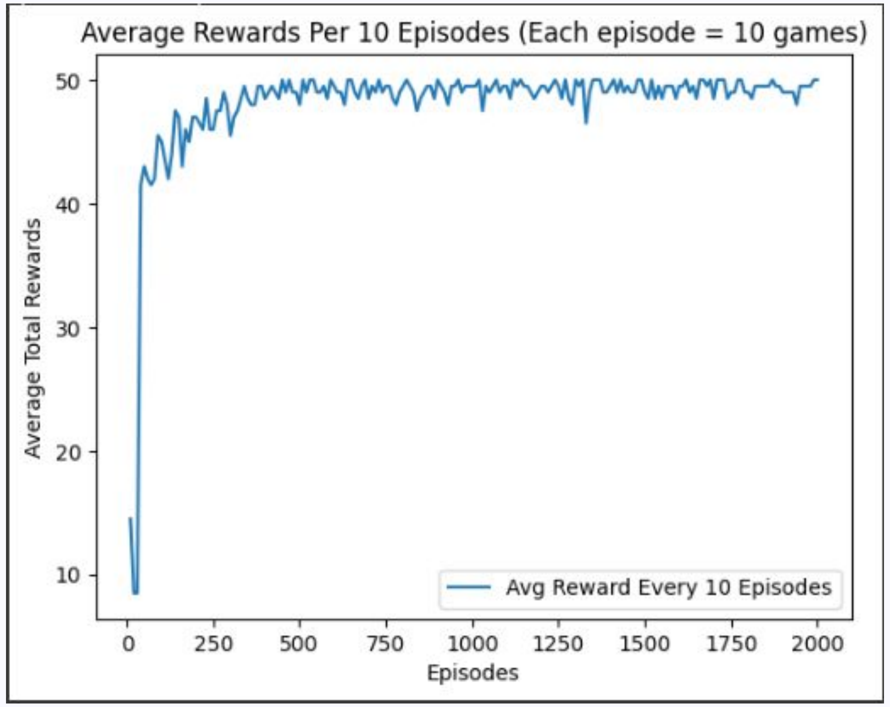
To mitigate this issue and observe a clearer trend in the learning process, we averaged the rewards over every 10 episodes.

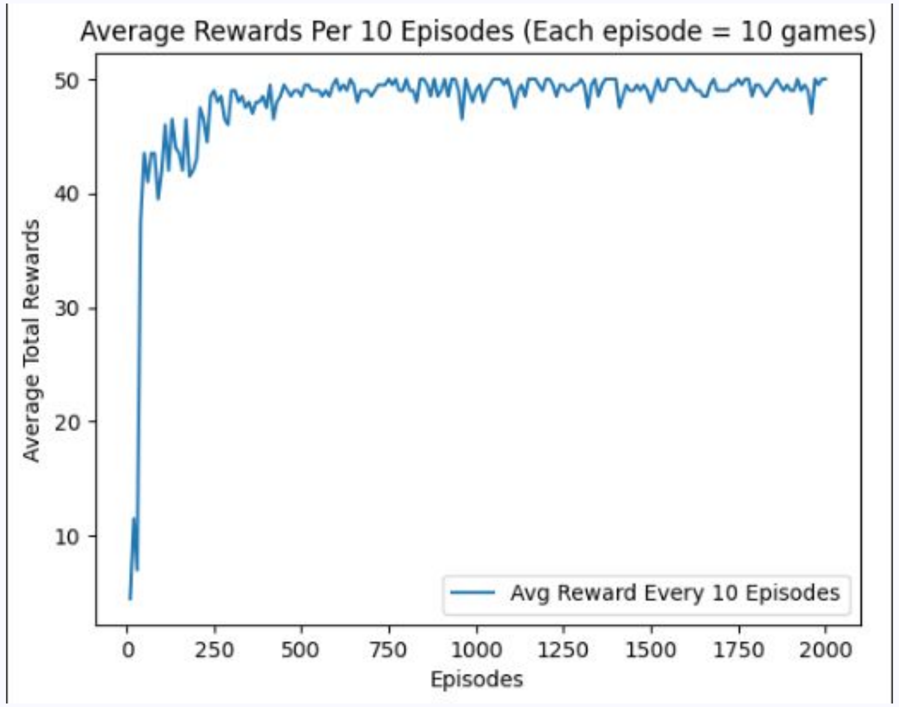


Q-Learning

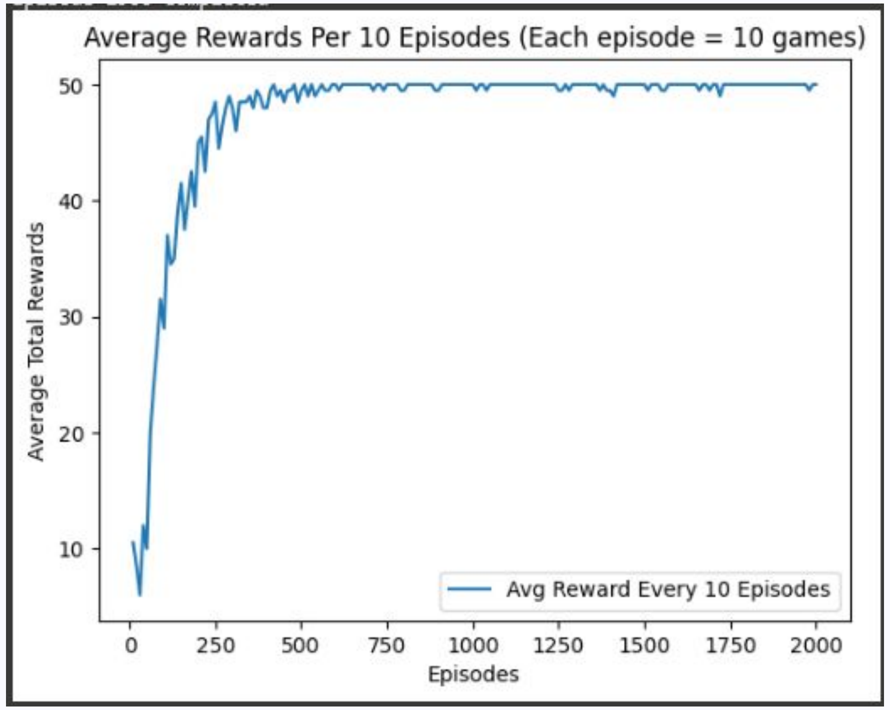
SASAR

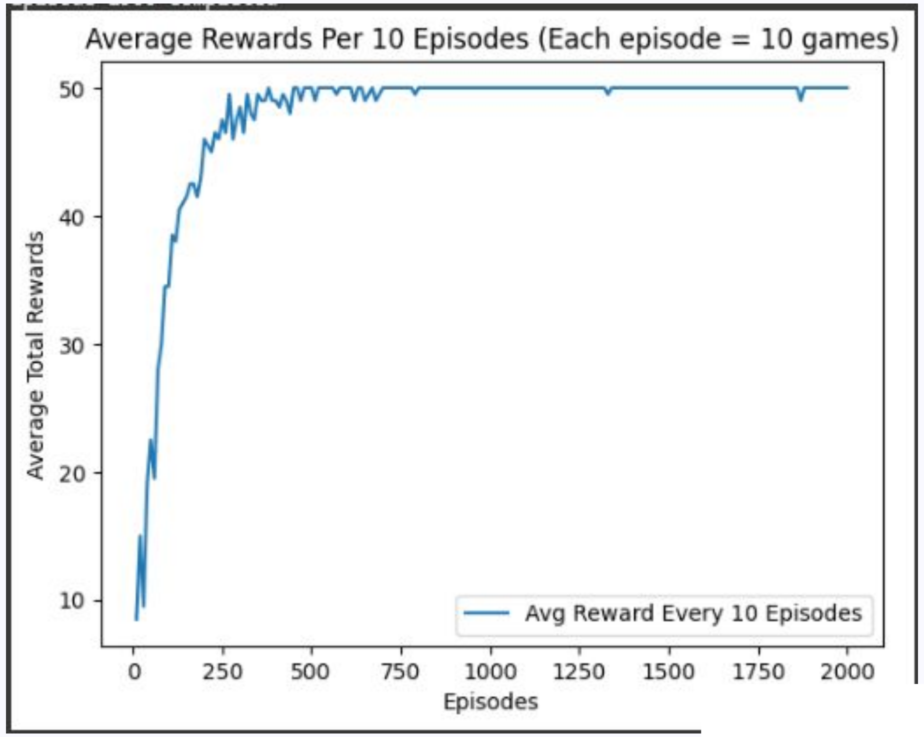
Epsilon = 0,1



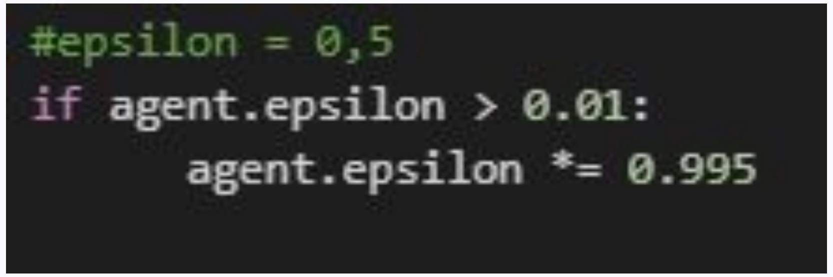


Epsilon = 0,5



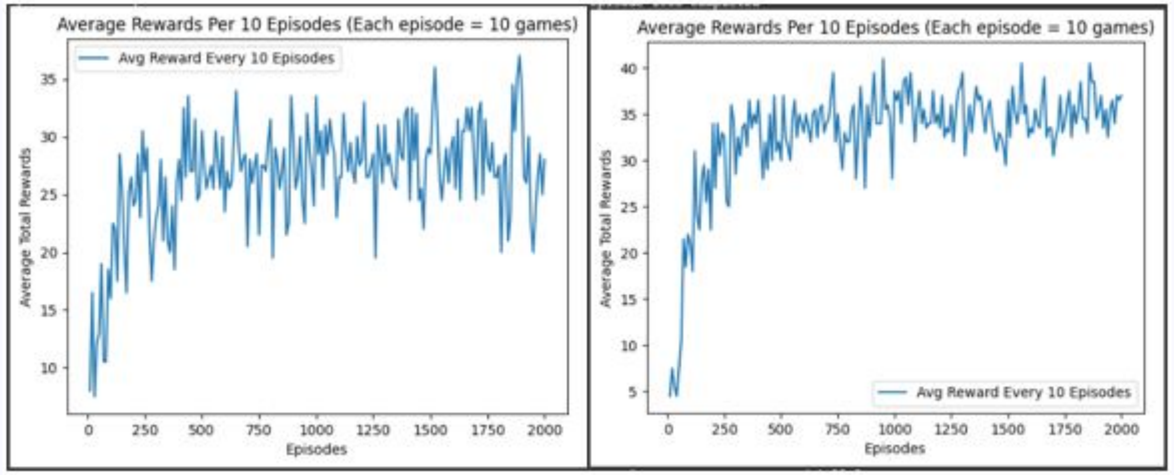


Decay Epsilon



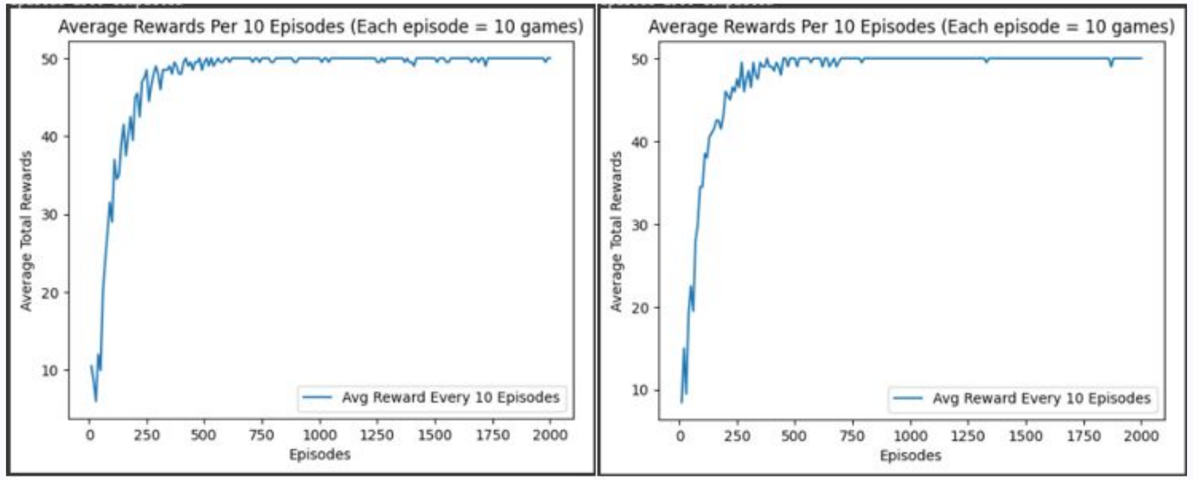
```
#epsilon = 0,5
if agent.epsilon > 0.01:
      agent.epsilon *= 0.995
```

Not decay epsilon



Q-Learning SARSA

Decay epsilon

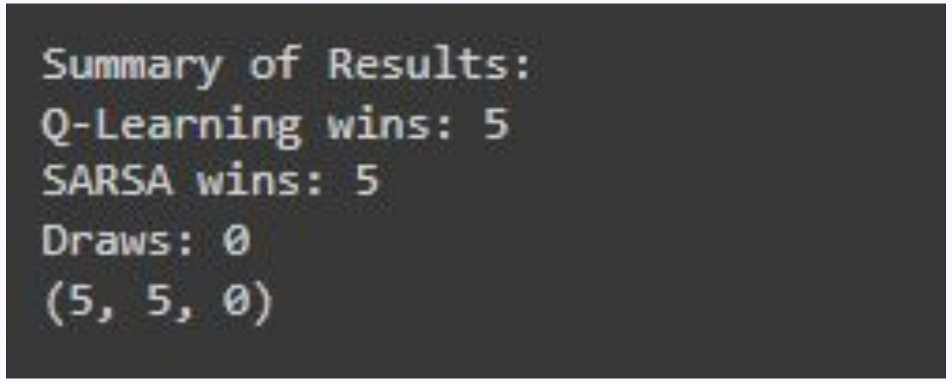


Q-Learning SARSA

Head-to-Head Comparisons (10 matchs)

Q-Learning Wins

5 Matches



```
Summary of Results:
Q-Learning wins: 5
SARSA wins: 5
Draws: 0
(5, 5, 0)
```

SARSA Wins

5 Matches

Future Recommendations

1

Complex Games

Future work could expand the project to more complex games, such as chess or Go.

2

Parameter Tuning

Tuning learning parameters like alpha, gamma, and epsilon could provide further insights into the algorithms.

THANK YOU FOR LISTENING