Here's the Python code for a reinforcement learning environment named EarlyLanguageEnvBeg using Gymnasium, along with a Q-learning training loop. This environment simulates the process of a child learning language over three years, with different phonemes and sentence structures introduced each year. The agent's actions involve selecting phonemes to form words and sentences, aiming to match a target structure depending on the current year of learning.

The environment is designed with specific states and actions that reflect the phoneme selection and sentence formation process. The reward system is based on correct guesses, partial matches, and the responsiveness of the parent (the agent) in the simulated environment. The training loop uses Q-learning, a model-free reinforcement learning algorithm, to learn the optimal strategy for the agent.

Throughout the training process, the agent explores and exploits the environment by choosing phonemes and forming sentences. The exploration-exploitation balance is managed by the epsilon parameter. The Q-table, which stores the value of each action in each state, is updated after each action based on the reward received and the learning rate.

This implementation demonstrates how reinforcement learning can be applied to a language learning simulation. The environment and the learning algorithm can be further modified and refined to simulate more complex language learning scenarios or different reinforcement learning tasks.

Episode: 988, Total Reward: -2014561.4649999442, Epsilon: 0.3785973821726237

Episode: 989, Progress: [----------------------------------------] 0.00%0%

Episode: 989, Total Reward: -2120925.4349999386, Epsilon: 0.3782289690277246

Episode: 990, Progress: [----------------------------------------] 0.00%0%

Episode: 990, Total Reward: -2277963.419999949, Epsilon: 0.37786092411182526

Episode: 991, Progress: [----------------------------------------] 0.00%0%

Episode: 991, Total Reward: -1753729.2949999461, Epsilon: 0.37749324705688064

Episode: 992, Progress: [----------------------------------------] 0.00%0%

Episode: 992, Total Reward: -2572396.9199999603, Epsilon: 0.3771259374952137

Episode: 993, Progress: [----------------------------------------] 0.00%0%

Episode: 993, Total Reward: -2012946.399999938, Epsilon: 0.3767589950595149

Episode: 994, Progress: [----------------------------------------] 0.00%0%

Episode: 994, Total Reward: -2132622.8549999627, Epsilon: 0.3763924193828417

Episode: 995, Progress: [----------------------------------------] 0.00%0%

Episode: 995, Total Reward: -2012457.9649999617, Epsilon: 0.3760262100986184

Episode: 996, Progress: [----------------------------------------] 0.00%0%

Episode: 996, Total Reward: -1795768.6149999613, Epsilon: 0.3756603668406357

Episode: 997, Progress: [----------------------------------------] 0.00%0%

Episode: 997, Total Reward: -2455096.669999965, Epsilon: 0.3752948892430504

Episode: 998, Progress: [----------------------------------------] 0.00%0%

Episode: 998, Total Reward: -2335441.444999962, Epsilon: 0.37492977694038465

Episode: 999, Progress: [----------------------------------------] 0.00%0%

Experiment 1:

Q-learning

# Hyperparameters

learning\_rate = 0.5

discount\_rate = 0.75

epsilon = 1.0

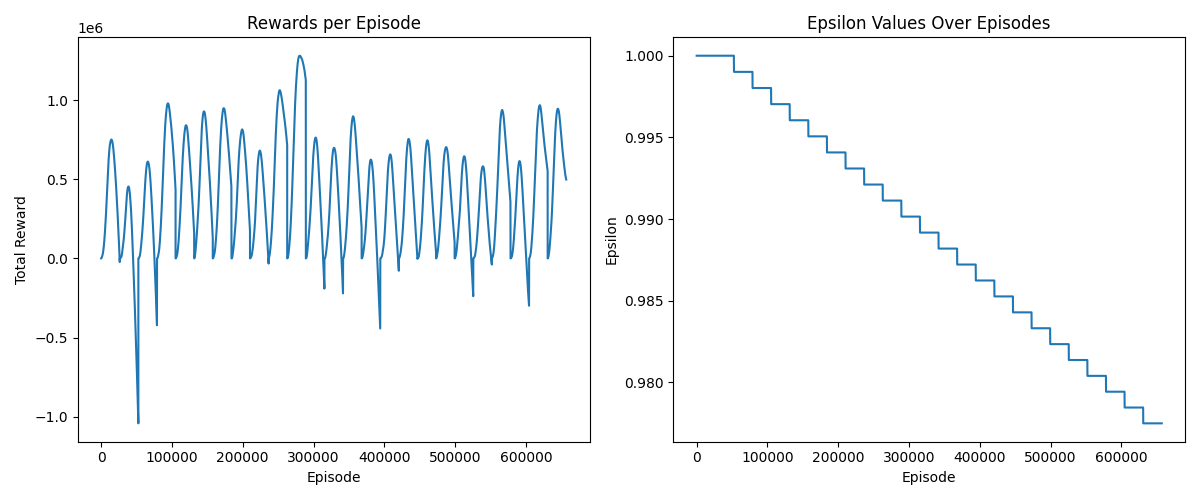
max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.001

# Reward function

self.total\_reward = 1\*self.cookie\_count + 0.15\*self.parent\_response - 0.05\* self.wrong\_guess - 0.075\* self.no\_action



Experiment 2:

# Hyperparameters

learning\_rate = 0.5

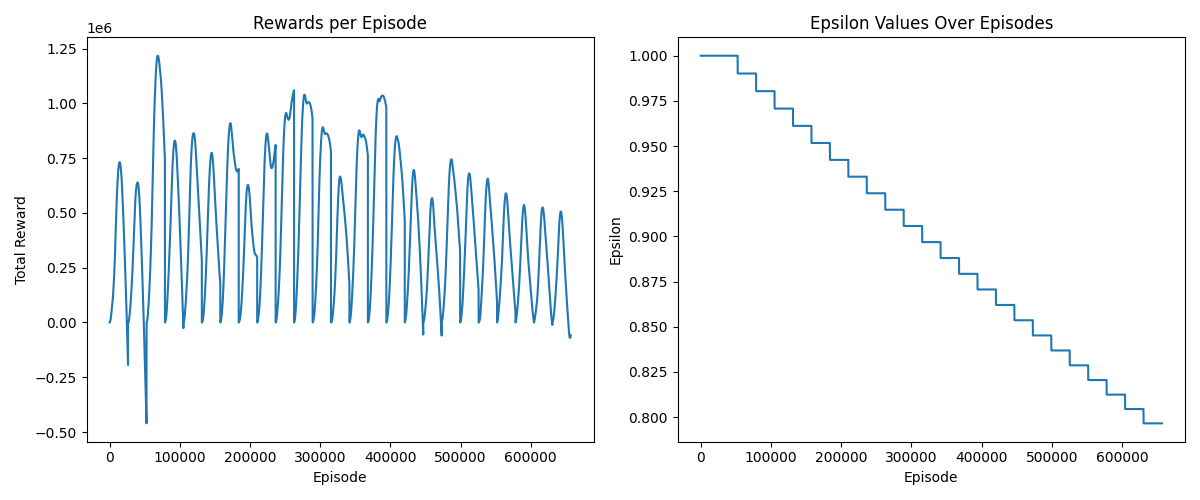
discount\_rate = 0.75

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.01



Experiment 3:

# Hyperparameters

learning\_rate = 0.5

discount\_rate = 0.75

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.1

A graph of a graph

Description automatically generated with medium confidence

Experiment 4:

# Hyperparameters

learning\_rate = 0.25

discount\_rate = 0.99

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.01

A graph and graph chart

Description automatically generated with medium confidence

Experiment 5:

# Hyperparameters

learning\_rate = 0.50

discount\_rate = 0.25 # how much a child cares about LT reward

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.1

A graph of a graph

Description automatically generated with medium confidence

Experiment 6:

# Hyperparameters

learning\_rate = 0.25

discount\_rate = 0.25 # how much a child cares about LT reward

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.01

A group of blue graphs

Description automatically generated

Experiment 7:  
  
# Hyperparameters

learning\_rate = 0.4

discount\_rate = 0.25 # how much a child cares about LT reward

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.05

A group of blue graphs

Description automatically generated

Experiment 8:

Episodes: 1000

# Hyperparameters

learning\_rate = 0.5

discount\_rate = 0.25 # how much a child cares about LT reward

epsilon = 1.0

max\_epsilon = 1.0

min\_epsilon = 0.01

decay\_rate = 0.001 # more time to explore