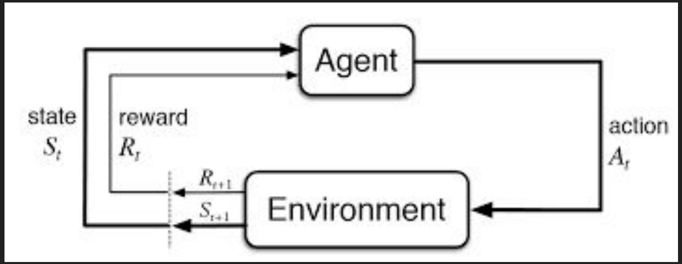
RL and Video Games 2: Transfer Learning Boogaloo

Rod Aryan, Allen Huang, Kat Inchoco, George Zhou

Introduction

- Interested in using RL in an entertaining medium
  - Atari games in Gym environment
- Previous Efforts
  - Single training effort
  - Limited to one game
- Our Project
  - Implement Transfer Learning
    - Requires fewer game-specific interactions
    - Less computational costs
  - Investigates the generalizability of learned features and policies across games
    - May provide valuable insights into transfer learning in reinforcement learning
  - Advancements in DQN may be helpful in this endeavor



Methodology

- DQN Agent
  - Deep neural network to approximate the Q-value
  - Utilizes Experience Replay
    - Stores and reuses past experiences
- Layers
  - Convolutional
    - Single layer CNN
    - No need for multiple layers
  - MLP
    - Flatten image to one dimension then feed into MLP layer
    - Did this in order to experiment on a simpler representation
- Training Protocol
  - Games: Pong and Breakout
    - Decided using Action Space similarity (see figure)
  - Settings
    - 10k training episodes, $\varepsilon = 1$ w/ .98 decay per 50 episodes
    - Replay buffer capacity set to 10k with 10 episode warmup phase

Num	Action
0	NOOP
1	FIRE
2	RIGHT
3	LEFT
4	RIGHTFIRE
5	LEFTFIRE

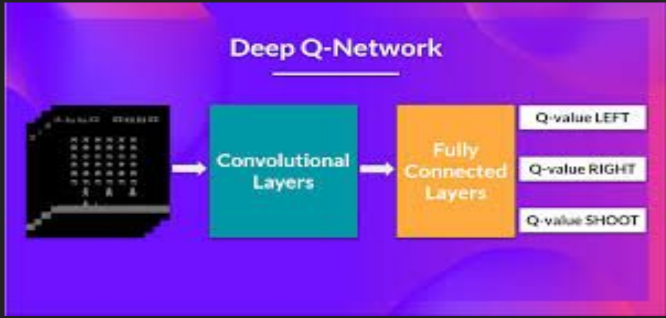
Pong/Breakout

Num	Action
0	NOOP
1	FIRE
2	RIGHT
3	LEFT

Space Invaders

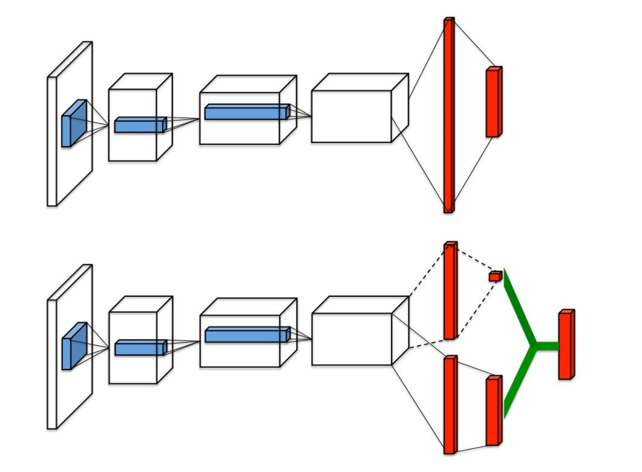
Methodology (cont.)

- Transfer Learning Application
  - Layer Freezing & Adaptation:
    - Existing layers are frozen to prevent changes during new training sessions
    - New output layer is added to tailor the network to the specific action space of the new game
  - Integration and Performance Monitoring:
    - Adapted model is integrated into a new DQN agent configured for the second game
    - Model undergoes training for a specified number of episodes
    - Continuous monitoring of both loss and reward metrics
- Evaluation Metrics
  - Overall Rewards:
    - Track total rewards accumulated across all episodes.
  - Time to Peak Reward:
    - Measure episodes/time taken to reach maximum rewards.
  - Loss Function:
    - Monitor convergence and stability through loss metrics.



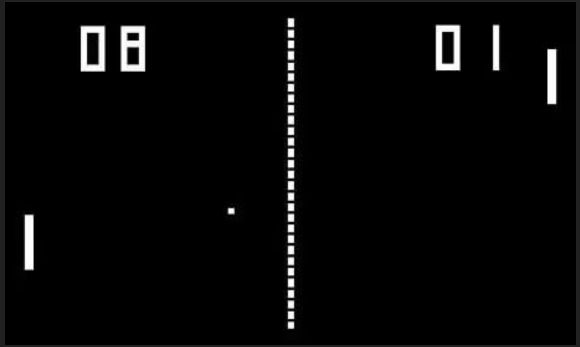
Challenges

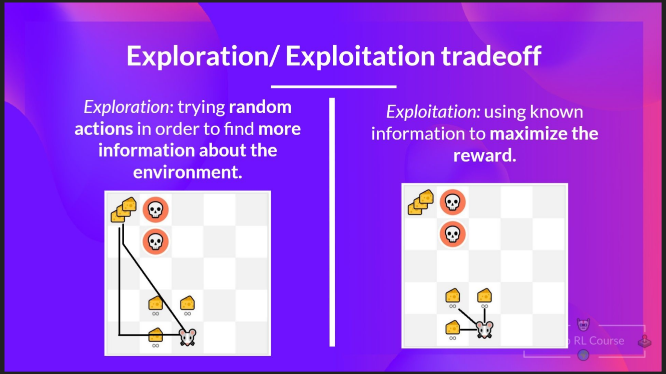
- Low GPU utilization
  - Required sampling from replay buffer, which is through CPU
    - Caused HPC to think process is not utilizing GPU, kicking us out
  - Solution
    - Move replay buffer from cpu to gpu
- Standard DQN vs Dueling Q-Networks
  - Dueling Q-Networks split the architecture into two streams
    - Estimating the state value
    - Computing the advantage for each action
  - Obscures the specific effects of transfer learning
  - Simple DQN eliminates these additional variables
- Starting w/ Pong vs Starting w/ Breakout
  - Originally thought Pong would be less intensive
    - Playing vs CPU could be reason
  - Breakout training took much less time



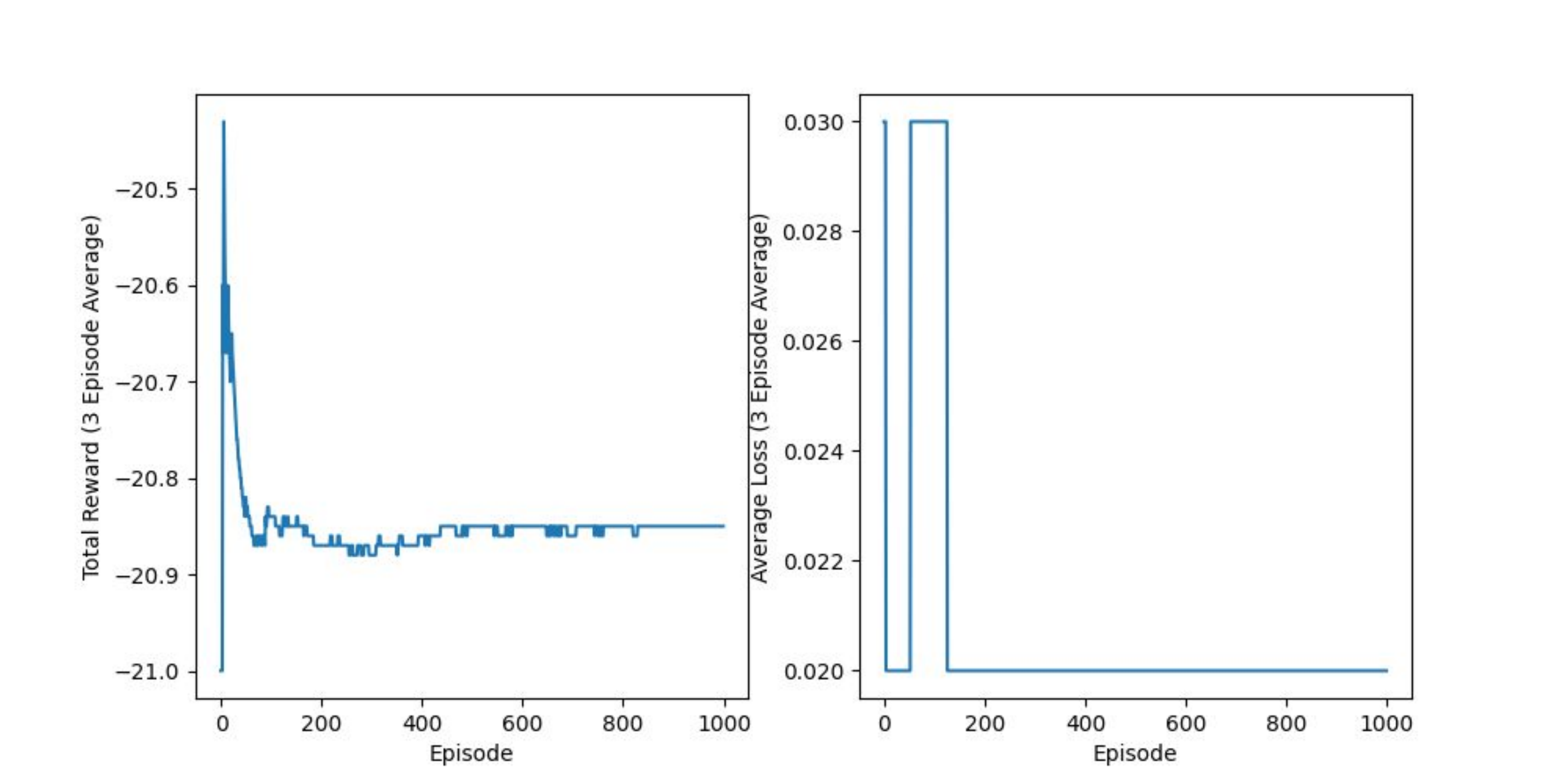
Challenges (cont.)

- Difficulty improving training
  - Computationally intensive
    - Unable to train for ~10K episodes consistently
    - Difficult obtaining resources from HPC
  - Cannot change state, rewards, actions since already defined by gym environment
  - Implemented gray-scaling to improve training performance
    - Focus on the important details
    - 3 channels to 1 channel
  - Perfecting ε decay
    - Finding the ideal epsilon decay rate was difficult
    - Affected the agent's ability to effectively balance exploration with exploitation

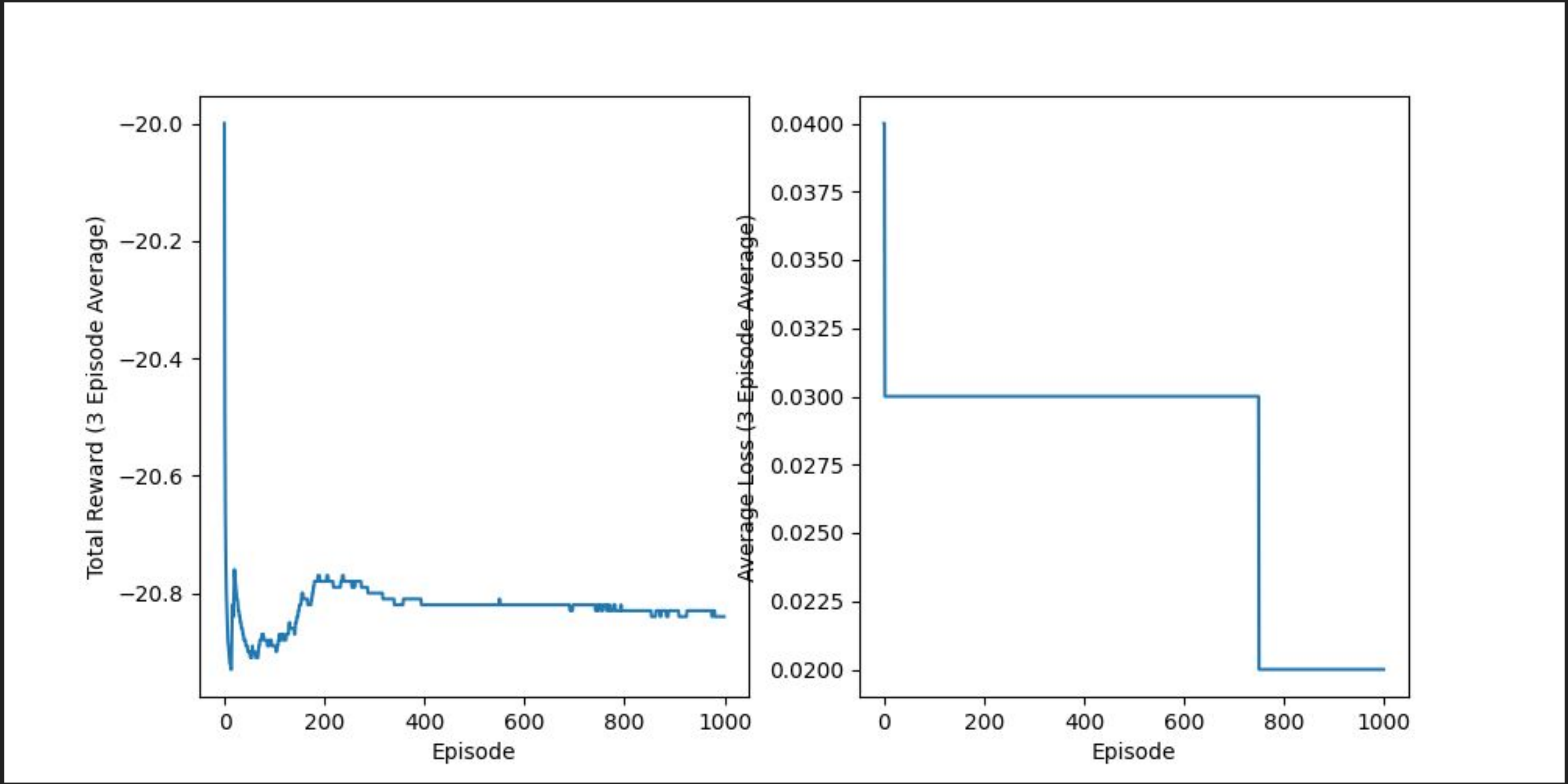




Baseline Results Pong MLP 1K Episodes



Baseline Results Pong MLP 10K Episodes



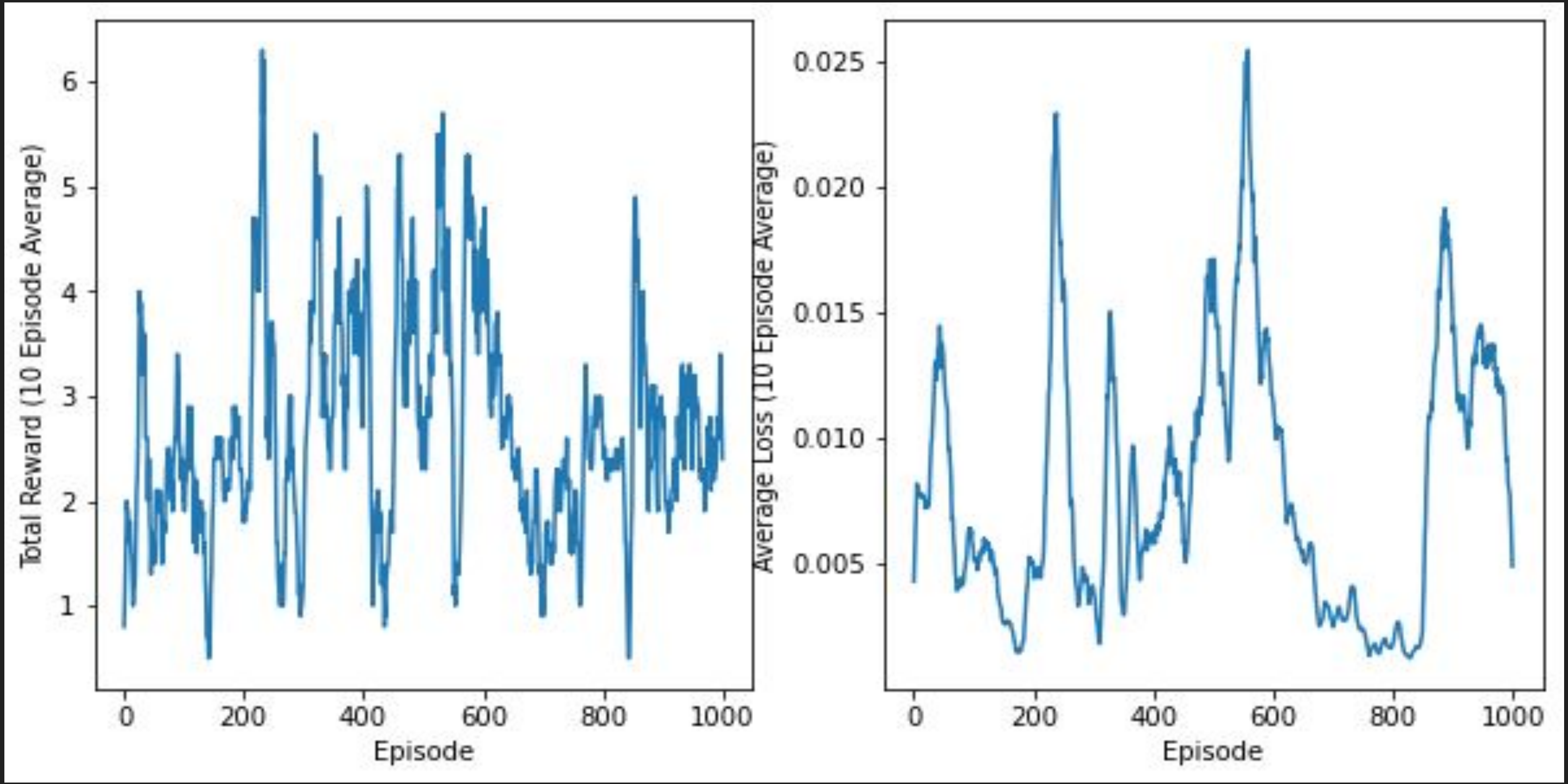
Baseline Results Pong CNN 1K Episodes

TBD

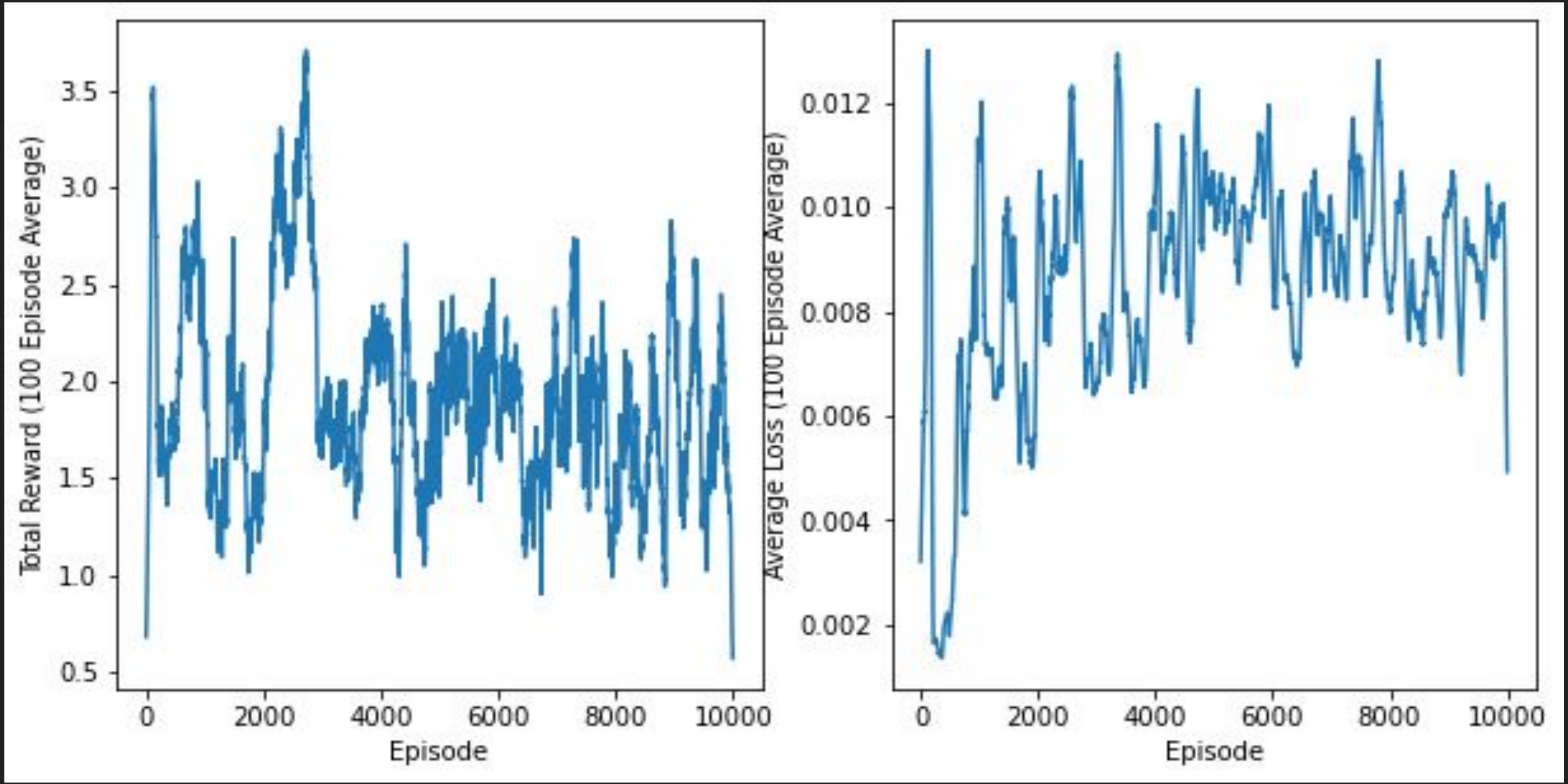
Baseline Results Pong CNN 10K Episodes

TBD

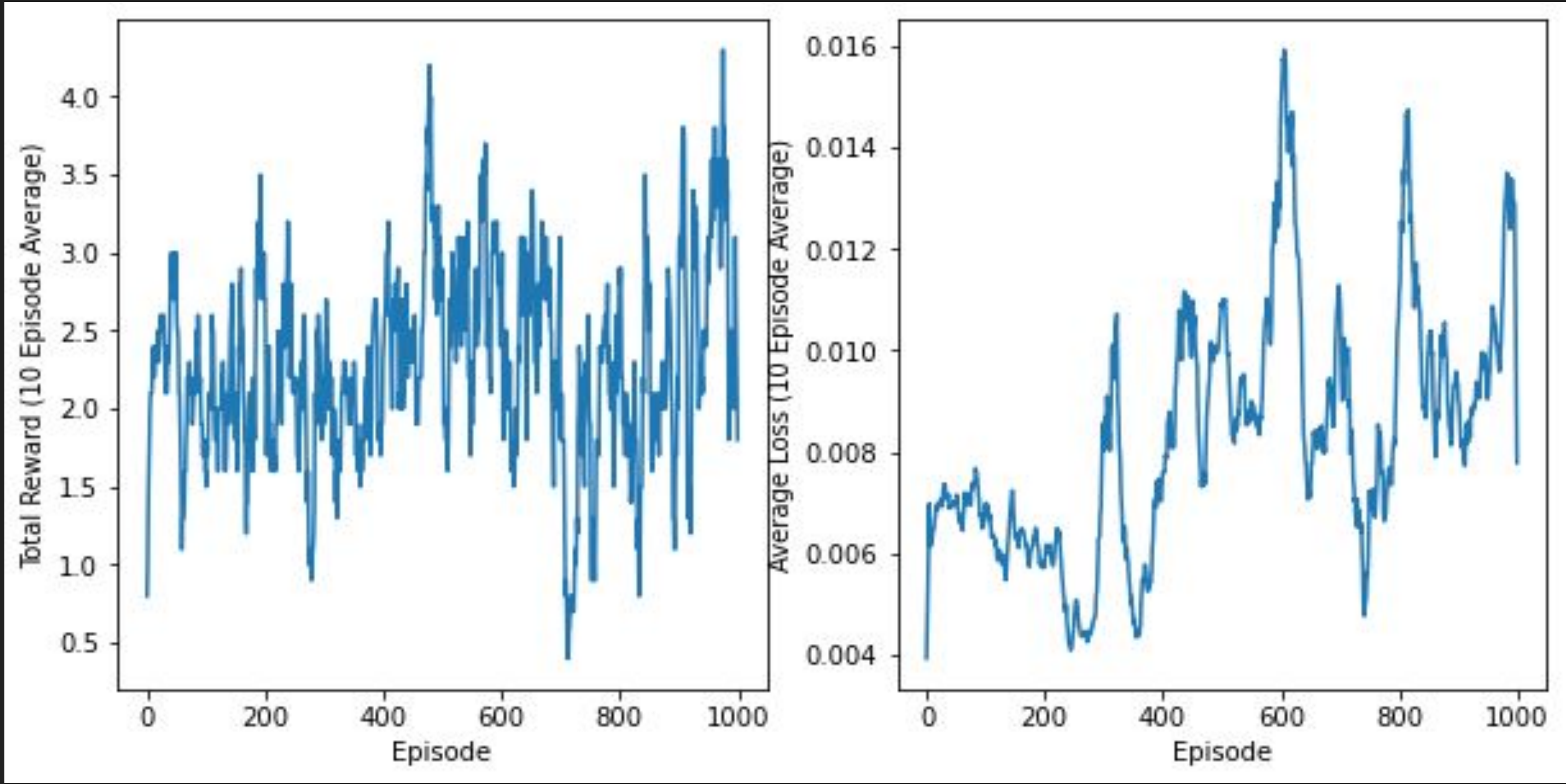
Baseline Results Breakout MLP 1K Episodes



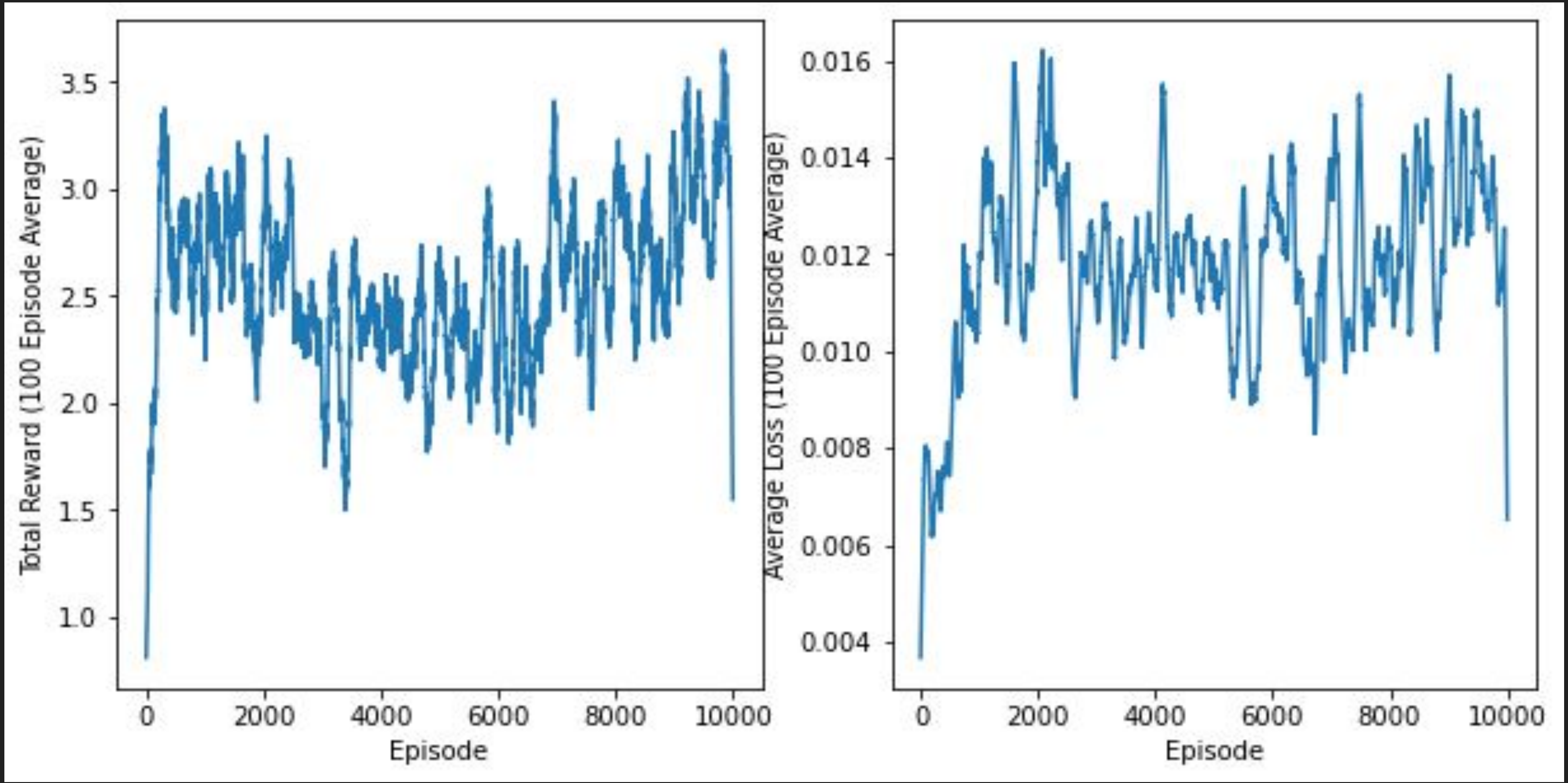
Baseline Results Breakout MLP 10K Episodes



Baseline Results Breakout CNN 1K Episode



Baseline Results Breakout CNN 10K Episodes



Transfer Learning Results

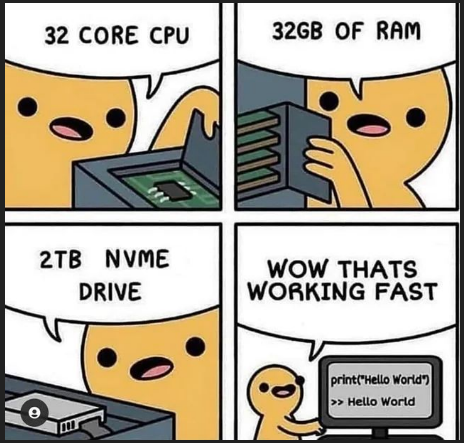
[add rendered gif]

TBD

Conclusion

- Effective Transfer Learning is difficult!
  - Project successfully implemented transfer learning between Pong and Breakout, but results were not up to our predictions
- Challenges and Limitations
  - Challenges such as optimizing epsilon decay and managing resource constraints highlighted the practical limitations within our computational environment
  - These issues underscore the importance of resource planning and hyperparameter tuning in achieving optimal training outcomes
- Insights into Generalizability
  - The adaptation of learned features and policies across different games provided valuable insights into the generalizability of transfer learning in reinforcement learning
  - We now have a better understanding of what is difficult for agents in an Atari game environment
- Future Work
  - Explore varying complexities of games
  - Dissect the impact of different network architectures on transfer learning
  - More efficient ways to handle computational resources to refine and expand our current findings

Thank you!



https://tinyurl.com/RLRebels