

PROJECT PROPOSAL FORM

Title of the Project: Training an agent to play Breakout using Reinforcement Learning

Student 1: Gabriel C. Ullmann

Date submitted: February 18, 2022

Abstract. This project aims to train an intelligent agent to play a version of the game [Breakout](#) using Reinforcement Learning. I will use a similar approach as proposed by [Mustafa Abusharkh](#), who created an OpenAI gym agent to play the game Snake and used the implementation of the PPO algorithm provided by the *stable_baselines3* package to train it. I will write the logic to guide the agent's behavior, as well as train it with variations of the PPO and A2C algorithms to observe how it affects the agent's actions and training time.

Objectives. 1. Determining whether Abusharkh's approach for training an agent for Snake can also be applied successfully to Breakout. 2. Determining which is the best reward strategy: rewarding the agent by breaking blocks only, by keeping the ball from dropping, or by doing both at the same time. 3. Comparing the agent's behavior by training with PPO (used by Abusharkh), A2C, and DQN (used in several academic research for training game-playing agents).

Data. While the initial inputs are random after the game starts running the agent will start collecting data from the environment to determine its next actions. The agent will be programmed to take actions according to environmental conditions (e.g: moving the paddle to a certain direction when the ball is also moving towards it). The outcome of these actions will return positive or negative rewards that are used by the agent to "learn" which actions it should take to reach success.

Measuring Success. In the context of this project, if the agent can reach Breakout's victory condition (destroying all blocks without dropping

the ball) at least 80% of the time over 20 test games, it will be considered successful.

Challenges. While Abusharkh's code will be used as a starting point, integrating the training script with a new game and testing it with different algorithms will require both Game Development and Machine Learning knowledge. I will not use the Breakout implementation given by [OpenAI gym](#) since it runs in an Atari emulator and does not give me as much flexibility to integrate the agent with the game variables. Instead, I will code a [simple version](#) of the game using the pygame library.

Literature Review. [Jeerige et al. \(2019\)](#) use the A3C and DQN algorithms to train an agent to play Breakout. A similar approach is presented in a [video](#) from the YouTube channel NamePointer. [Mnih et al. \(2015\)](#) generalize this approach to other Atari 2600 games. Similarly, [Adamski et al. \(2018\)](#) train their agent for several games, but they use BA3C. Genetic algorithms such as NEAT have also been used in this field, as presented in a [video](#) by the YouTube channel ForrestKnight.

Originality. Using Reinforcement Learning and algorithms such as DQN and A3C for playing Breakout is something that has been tried before by other academic researchers. However, I could not find comparisons between the training time and agent behavior for different algorithms. This is what I intend to do in this project.

Computational resources. I will use my personal laptop: Dell G3 15 3590, 8GB RAM, Processor Intel(R) Core(TM) i5-9300H CPU @ 2.40GHz. Running Windows 11.

Used Python packages. I will use *numpy* for array manipulation, *stable_baselines3* for running the RL algorithms, *pygame* to implement a basic version of the Breakout game, and *gym* (OpenAI Gym) to implement the agent's logic and run its training steps.