Reproducing DeepMind's MuZero Algorithm on Atari Games

Team Members: Madhu Sivaraj (ms2407), David Tian (dt474), Justin Chan (jlc544)

Background: MuZero is a reinforcement learning algorithm which combines a tree-based search with a learned model to outperform humans in Go, chess, shogi, and numerous Atari arcade games. It is seen as a significant advancement in model-based planning, as it is able to learn to make decisions that maximize its reward in a variety of complex domains, without any knowledge of underlying dynamics. Therefore, its creators hope that MuZero will also be applicable to a wider range of real-world applications.

Problem Statement: We aim to reproduce the results of the MuZero algorithm presented by DeepMind. The challenge here is to be able to play games such as Chess, Go, and Atari games, which includes arcade games such as Breakout, Space Invaders, and Tetris. Playing these games is an interesting task because it involves making decisions to achieve certain goals or rewards. The challenge here is to maximize a value such as the final score.

Approach Plan: We are going to solve the challenge by implementing the MuZero algorithm and reproducing its results. The MuZero algorithm combines a tree-based search with a learned model, achieving superhuman performance in a range of challenging and visually complex domains, without any knowledge of their underlying dynamics. One of the interesting things about the MuZero algorithm is that it doesn't need to know the rules of the game, which makes it simpler to extend to a wide variety of games. At a high level, it takes an image of the board or screen as input and picks from a set of actions it can take. As a result of those actions, it is given a score that can be utilized to place a value on each action at a certain state.

The evaluation plan: We can demonstrate whether the goal is achieved by comparing the final game scores that our implementation reports with the scores that the original papers attained. We can also compare the behavior of the algorithm after a certain amount of training periods. If they are similar, we can conclude that we have successfully reproduced the algorithm. If they are not, we can still look at how our results differed and examine what the causes of those differences are. We are also looking into game emulator applications to use for visualization purposes and have discovered Arcade Learning Environment, an open-source framework that allows researchers and hobbyists to develop AI agents for Atari 2600 games. If we try to expand our reproduction to other domains of games beyond Atari, we could look into Retro Learning Environment, an alternative to ALE. It can run games on the Super Nintendo Entertainment System, Sega Genesis and several other gaming consoles.

References

- Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model (2019)
- Playing Atari with Deep Reinforcement Learning (2013)