Deep Q-Learning Tron Bot

Xiuzhen (Tiffany) Lei

1 Abstract

Tron-41 (CSCI1410's version of the game Tron) is a two-player game in which players move around a grid containing walls, powerups, and barriers with the goal of outlasting the other player. They leave barriers behind as they move, and the first player to collide with a barrier or wall loses. Powerups can sway the outcome of the game by adding or removing barriers, allowing players to move through barriers, or giving players multiple consecutive moves.

The goal of this capstone project is to implement a bot with machine learning to play Tron-41. The implementation uses a deep reinforcement learning approach - specifically, Deep Q-Learning. The deep Q-learning algorithm builds on the original Q-learning algorithm, differing only in its use of a deep neural net to generate Q-value approximations for all of the possible actions given the state rather than a Q-table with state-action pairs. The implementation of the deep Q-learning algorithm used in this project is mainly based on the description of the algorithm given in Mnih et al. (2015), with modifications made to the sampling of experience replay memory. The game states are defined to take into account 4 major factors: immediate blocked paths (walls, barriers, opponent within one move), space availablility (area accessible without obstruction), distance to powerups, and defense space (amount of space that can be claimed before opponent).

In order to maintain stability in final policy across training runs, additional training grids were designed to guarantee that the model saw enough of each game state, especially critical decision states. These grids confined the bot's available space and required near immediate decisions, leading to quicker termination and shorter training time.

The resulting bot was tested against four class bots on multiple grids of varying levels of difficulty. Of these, the final saved model of Deep Q bot was able to win the majority of the time, but further improvements could be made to increase the consistency of the final policy across training runs.

This paper (edit: full version, not included here) provides an overview of the general Deep Q-Learning algorithm, a description of modifications made, and specifics on its application to solving Tron-41 (including network architecture, game state representation, hyperparameters, and reward function).

2 Resources

Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. Nature 518, 529–533 (2015) doi:10.1038/nature14236