Deep Q-Learning and Hierachical Reinforcement Learning in Exploratory Games

Emmett Tomai, Roberto Rivas, Lauryn Brough and Ivan Amaya Vazquez
Department of Computer Science, University of Texas Rio Grande Valley

Previous Research & Background

In years past, machine learning has been applied to games such as Go (Silver et al., 2016) and classic Atari games (Mnih et al., 2015), and has been shown to outperform human opponents. However, these networks are trained to be rigid in their application and therefore it is hard to reapply learned skills.

The nature of exploratory games makes them ideal for testing the transfer of knowledge or skills. The cycle of gathering materials in order to craft a tool, for instance, needs to follow a "recipe" of materials for a specific tool; nonetheless, the basic skills needed for "gathering" and "crafting" any item should be the same, and thus transferable.

Goals

In many exploratory game environments, the only driving force behind a player's actions is to survive as long as possible, which can encompass many subtasks such as alleviating hunger, crafting weapons to defend itself from enemies, or simply finding shelter. Due to the repetitive nature of most of these tasks, the skills needed to accomplish one of them could be reused when trying to perform another one.

For this reason, we aim to develop an agent that can leverage previous training to learn a new but similar task faster and with a shallower learning curve.

Methods

We begin by employing the options framework (Sutton, Precup, and Singh 1999), modeled in Figure 1, which allows for an abstract hierarchical representation of an action to be used. For example, the action of gathering from a tree could be decomposed into more specific actions or skills, such as walking up to a tree or gathering resources from it.

Using trained options, the network might be able to determine the necessary subtasks to accomplish its overall goal and use previously learned skills to speed up learning, as shown in Figure 2.

We applied the options framework to a Double Dueling DQN as our learning model. To train the model, we increasingly punished repeated failed actions so that the network can learn faster how to achieve the global goal.

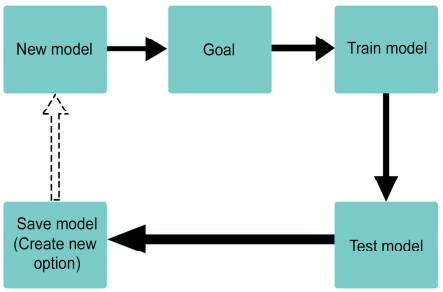


Figure 1. Basic Cycle of working with the Options Model

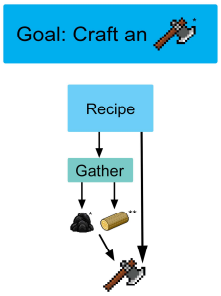


Figure 2. Generation of Subgoals Example

Double Dueling DQN

- Uses two identical networks: the first one for taking actions; the second one for generating Q-values
- Separates the action value function into two: Value and Advantage functions.
- Value function: How good it is to be in a given state.
- Advantage function: Best action to take, compared to the other possible ones.

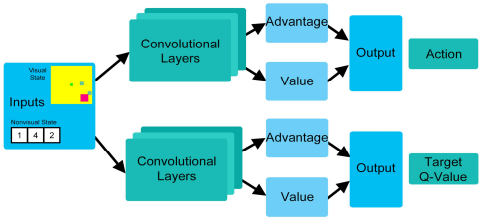
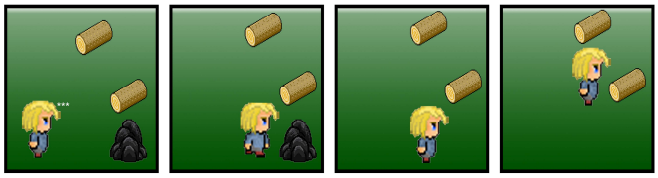


Figure 3. Visual Breakdown of Double Dueling DQN



Figures 4-7. Steps as the network identifies and moves toward a gatherable object, gathers it, and moves onto its next objective

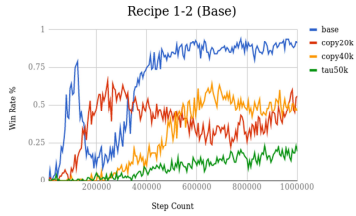


Figure 8. Graph showing reward growth over steps for different settings of the base model

Figure 9. Graph showing reward growth over steps for different settings of the options model

Results & Future Work

Tests with the Double Dueling DQN prove that the options model provides a steady improvement in the ability of the network to learn. As shown in Figures 8-9, we ran tests by fully copying to our target network after so many steps (copy), and copying gradually over so many steps (tau), for both the base and options models. As results show, most of these situations experience a sudden "drop off"; however, by periodically copying instead of doing it all at once (tau vs. copy) we don't experience this "drop off". We also see that the application of the options model (opt) generally enables us to learn faster.

We hope to expand this project beyond the basic gather-craft cycle to include map exploration, where the agent is only able to see only a part of the complete map at a given time, to be applied to searching objectives such as "find the treasure".

References

Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." nature 529.7587 (2016): 484.

Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." Nature 518.7540 (2015): 529.

Sutton, Richard S., Doina Precup, and Satinder Singh. "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning." Artificial intelligence 112.1-2 (1999): 181-211.

*StockUnlimited. "You Don't Have To Be A Designer To Get Awesome Visuals." StockUnlimited, www.stockunlimited.com

**"Learn How to Draw Hand-Crafted Pixel Art in Photoshop." Design & Illustration Envato Tuts+, design.tutsplus.com/articles/learn-how-to-draw-hand-crafted-pixel-art-in-photoshop--psd-5284.

***"Pixel Art Characters." DeviantArt, www.deviantart.com/jakus333/art/Pixel-Art-Character

UTRGV