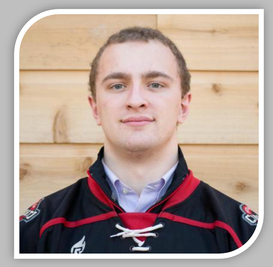
STARCRAFT 2 AI


Kyle Arens


Ryan Benner


Jon Deibel


Dr. Ali Minai
Advisor

Background

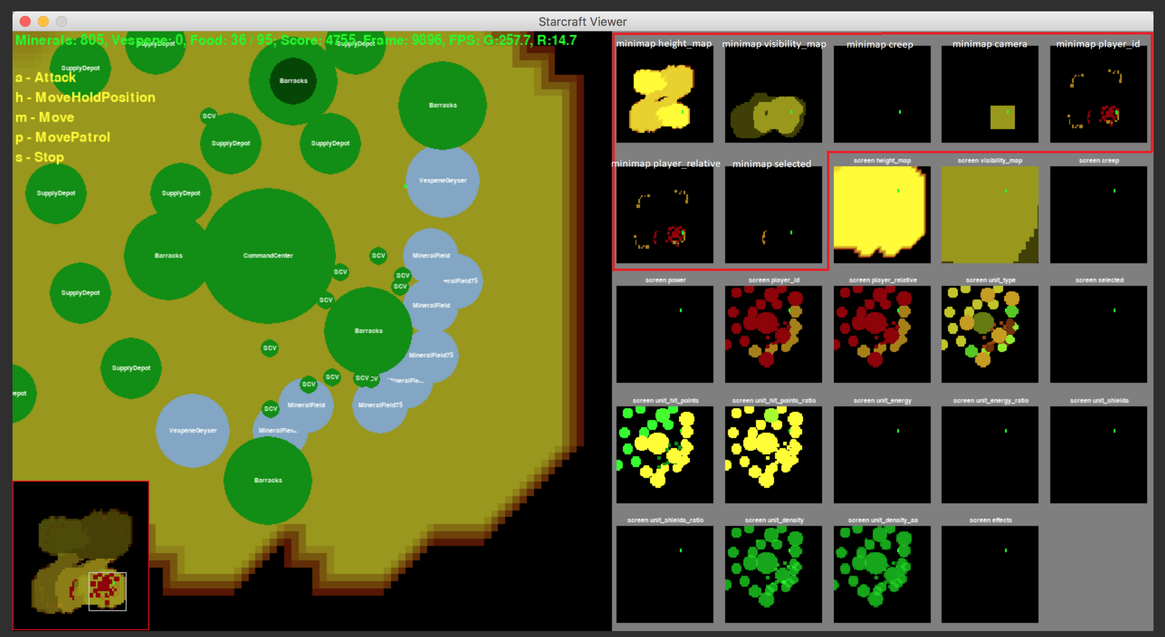
There have been many recent advances in game-playing AIs, such as the Dota 2 AI and AlphaGo. With this project, we aim to explore the use of conventional and cutting edge machine learning techniques to create a self-learning Starcraft 2 AI agent that is capable of playing against Blizzard's built-in AI.

Scope & Decisions

With 3 distinct factions to play as and an action space of ~10^8 possibilities, we limited our project in the following ways:

- Only train as Terran.
- Only train on a small 64x64 map.
- Train via reinforcement only against Blizzard AI.
- The Blizzard AI's race will change each iteration.
- Restrict our action space to simple actions, such as:
    - Build SCV
    - Build Marine
    - Build Barracks
    - Build Supply Depot
- Prevent training bot from moving camera.
- Feed the AI the current screen region and 7 distinct minimap layers.
- Utilize a Q-learning table to handle multi-dimensional tensors.

Design and Input



The seven regions boxed in red above serve as a high-level overview of the first layer of inputs that we feed into our machine learning model. These seven layers are:

1) Height Map – observe terrain differences, impacting vision.
2) Visibility Map – observe current and past exploration.
3) Creep – shows where Zerg has spread "creep". Race-specific.
4) Camera Selection – shows where the camera is currently located.
5) Player Id – shows units based on owning-player's ID.
6) Player Relative Team – shows units relative to their respective teams.
7) Selected – shows the currently selected unit.

Milestone Achievements

- 10/31 – Develop a scripted AI using capable of defeating Medium AI opponents.
- 12/05 – Implemented a simple Q-learned model capable of occasionally winning against Very Easy AI.
- 1/13 – Update to a deep-learned model with better success than Q-learned.
- 2/5 – Update reward algorithm to use player score. No noticeable change.

Future Work

- Expand action space to the full possibility of Terran's actions.
- Train against multiple maps.
- Train AI to change camera location.
- Utilize Pro-player replay data.
- Expand input space to include all mini-map layers.
- Train AI to play as Protoss and Zerg.