Games/ Dota2 And StarCraft II

Convolutional Neural Network

Kapil Gautam
kxg180032@utdallas.edu

Outline

- Motivation
- History
- Challenges for AI agent
- Objective
- Highlights - Dota2/ OpenAI
- Highlights – StarCraft II/ Deepmind
- References

Motivation

- A Real time strategy game allows players to simultaneously play the game in real-time.
  - Players needs to position structures and guide units under their control to secure areas or to destroy opponents' assets.
  - Additional structures and units can be created throughout the game limited by an upper threshold.

- Real-time strategy(RTS) games are recognized for their domain complexities.
  - Many sub-problems of RTS games, for instance, base economy, micromanagement, or build order optimization have been studied by Artificial Intelligence (AI) researchers.

- Some of the popular RTS computer games include Dota2, Starcraft2, Warcraft 3, Age of Empires 2, etc.
  - Most of these games have a either a single player v/s bot match or a team v/s team match.
  - The most interesting ones are the team v/s team matches as both sides compete for resource optimization, economy management and micro-management of their army/forces during a battle.

History

- May 1997 - IBM’s DeepBlue – an AI agent – defeated the Chess world champion, Garry Kasparov.
- March 2016 - DeepMind AlphaGo – an AI agent – defeated the Go world champion, Lee Sedol, 4-1.
- June 2018 - OpenAI five won against the world champion DOTA 2 game team OG, 3-0.
- December 2018 - Deepmind’s bot defeated a top professional player in StarCraft II game, Grzegorz "MaNa" Komincz, 5-0.
- March 2020 - Deepmind trains a superhuman AI for 57 Atari Games which exceeded at human play in each of the game with a single generalized algorithm.

Challenges for AI agent

- **Long-Term Strategy Planning**
  - A typical RTS game would take about an hour or two to complete. During that time, AI agent would need to constantly take actions to execute on an overall strategy. Early actions in a game might not take effect until much more later which would require constant long-term planning abilities.

- **Partial observable environment**
  - Unlike games like chess/Go with fully observable environment, RTS games like StarCraft II / Dota 2 never presents the complete environment configuration at any given time. The AI agent needs to be able to operate using imperfect information.

- **Balancing exploration and exploitation**
  - There is no single winning strategy in an RTS game. The AI agent needs to balance taking actions that yield immediate benefits to the need of exploring the environment in order to expand its strategic knowledge about the environment.

Challenges for AI agent

- **Real Time**
  - RTS gaming involves simultaneous actions by both players, different from a turn-by-turn game like Chess/Go. The AI agent need to evaluate thousands of options in real time and detect the best match for the long-term strategy.

- **Huge Action Space**
  - The AI agent should be able to control hundreds of units at any given time and the various combinations of actions grow proportional given the complexity of the imperfect environment

	Atari	Go	Starcraft/Dota2
Information Type	Near Perfect	Perfect	Imperfect
Players	Single Player	Multi-player	Multi-player
Action Space	17	361	~10^26
Moves per game	100's	100's	1000's
Additional aspects		Intuition	Intuition, team-work and collaboration

Objective

- Recently two of the RTS games, StarCraft2 and Dota2 have been popular in the research community to evaluate an AI agent play against a professional gamer.
    - Rules of these games are so complicated that you cannot pre-program these rules. Even if you try to pre-program it, the result would not be as good as an average player.

- The objective for researchers is to build a general AI agent which can learn by itself.
    - RTS games serve as a testbed to evaluate an AI agent which can play a complex multiplayer real-time strategy game defeating top tier game players.
    - Every hero(player character) of the game has a variety of unique actions and upgrades available.
    - To maintain an equal ground with the human player and AI agent, both teams have the same heroes.
    - In order to win over a human player an AI agent would demonstrate team-work, planning and strategy development in a partially observable environment.

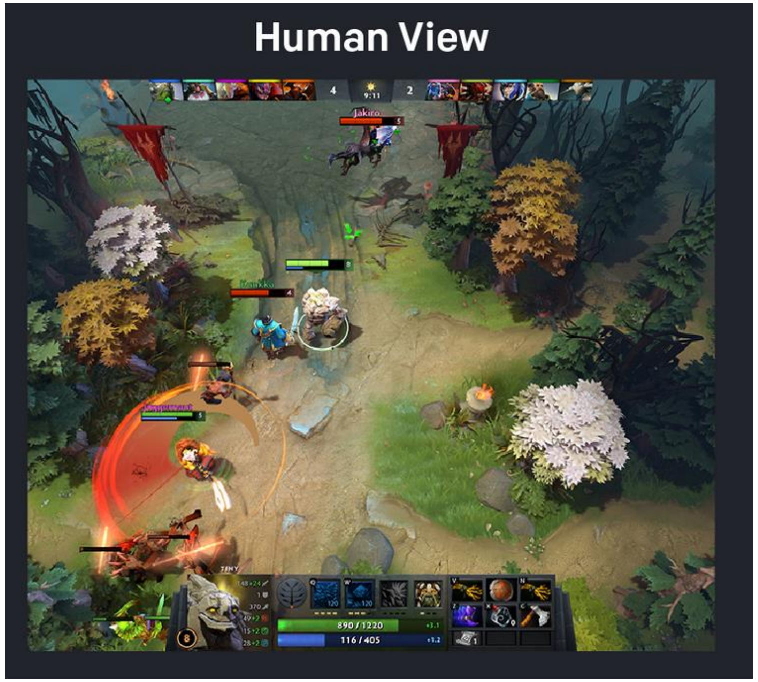
Highlights - Dota2/ OpenAI

- Metrics
  - Trained for a pool of 17 heroes playing over 10,000 years of games learning via self-play.
  - OpenAI five played 180 years worth of games against itself every day.
  - Utilized a distributed training system to do this over a period of 10 months utilizing thousands of GPUs.

- Training Phase
  - The algorithm passes through a Vision network which then passes through an imitation network.
  - The AI agent is fed with raw videos of human data, and the information is perceived as a bunch of numbers in neural network.
  - Utilizing a separate LSTM for each hero and no human data, it learns recognizable strategies.
  - Tuning of self-defined hyperparameter called – team spirit – enables the AI agent to care about its team members.

Highlights - Dota2/ OpenAI

- Training Phase
  - Each timestep, OpenAI Five receives an observation from the game engine encoding all the information a human player would see such as units' health, position, etc. OpenAI Five then returns a discrete action to the game engine, encoding a desired movement, attack, etc.
- Challenges during training
  - Training over months will have changes in the codebase, which would require to restart training.
  - Certain game mechanics were controlled by hand-scripted logic rather than the policy, like the order in which heroes purchase items and abilities, control of the unique courier unit, etc.
  - The Open AI Five only acted on every 4th frame of the game - a timestep - although the Dota 2 engine runs at 30 frames per second.



https://openai.com/content/images/2019/04/Dota-Matrix.png

AI View

3.006	-1.386	-0.4695	0.883	1	0.84
-0.3154	-0.5425	-0.5	0.866	0	0.82
3.11	-1.36	-0.9336	0.3584	1	0.78
-2.324	2.863	0.9746	0.225	0	0.86
3.037	-1.361	-0.7773	0.6294	1	0.82
-1.387	2.951	0.988	0.1565	0	0.74
3.023	-0.9395	0.05234	-0.9985	0	0.66
2.951	-0.5747	0.01746	1	0	0.72
2.963	-1.303	0.3906	0.9204	0	0.68
2.834	-3.164	0.01746	-1	0	0.68
3.127	-1.368	0.6562	0.755	1	0.55
3.088	-1.366	0.4695	0.883	0	0.55
2.984	-1.398	-0.225	0.9746	1	0.55
3.037	-1.391	0.788	0.6157	0	0.55
3.076	-1.438	0.883	0.4695	0	0.55
-2.412	2.846	0.996	0.08716	1	0.3

Highlights - Dota2/ OpenAI

- Testing Phase
  - AI agent plays unusually aggressively and use buybacks quite liberally where human players don't consider it to be a good choice. (buybacks – to resurrect a perished hero quickly costing some money).
  - AI agent waits for the right moment to win, in another it pressured the human player from the starting and never let the human player reach the end game to execute their strategies.
  - The AI agents evaluates that if the game goes on for a long time, statistically its chances to win the game dwindles, so it immediately need to go and win from the start whatever the cost.
- Key Learnings
  - The AI agent allocates resources as efficiently as possible.
  - Humans can use AI to understand why a weird move was played if it generated a better outcome than they thought and use it to improve their own game.

A sneak peak of a visualization of OpenAI Five's objectives.

https://openai.com/blog/openai-five-finals/

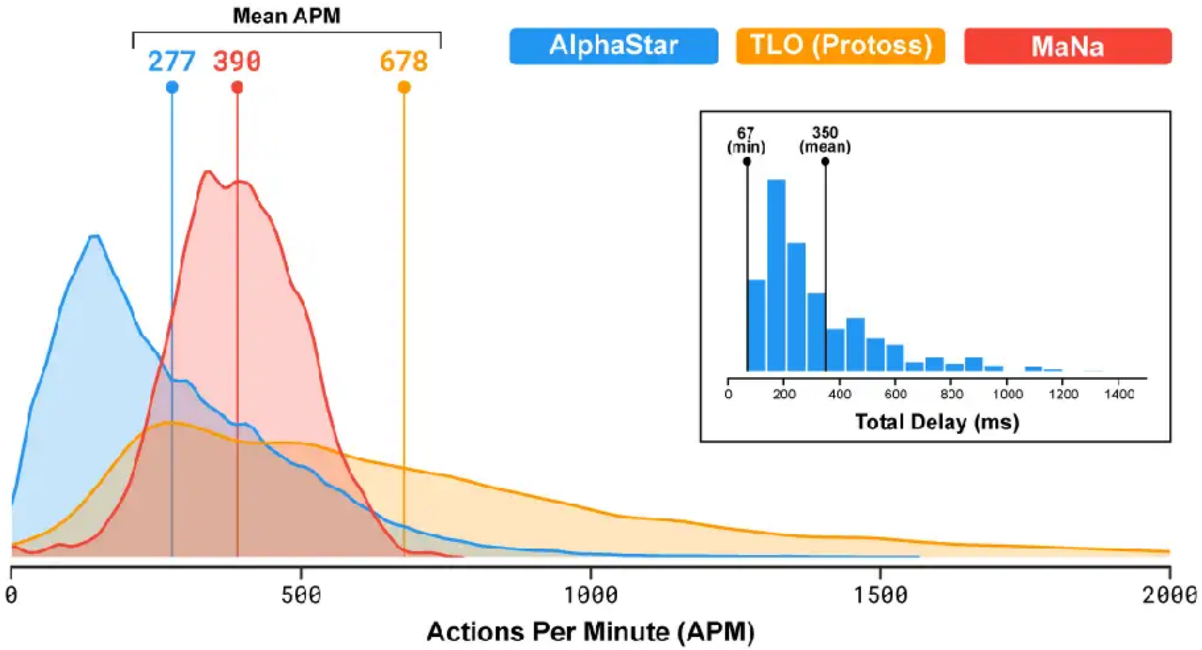
Highlights - StarCraft II/ DeepMind

- Dota 2 shared the complexities of RTS game like StarCraft for having imperfect information and large time horizons. However, unlike AlphaStar, some game rules are simplified, agents use hard-coded sub-systems for certain aspects of the game and agents do not limit their perception to a camera view.

- In StarCraft II, each player chooses one of three races—Terran, Protoss or Zerg. Each of the race has its own unique behavior, buildings and unit control.
  - Each player begins with an initial number of worker units to gather basic resources and to build more structures and to upgrade technologies.
  - To win, a player must carefully balance the macro and micro elements of the economy.
  - Macro being the big-picture management of their economy, while micro being the low-level control of their individual units.
  - Require split second decision making, it has imperfect environment, the AI only sees wherever its units or buildings are placed.

Highlights - StarCraft II/ DeepMind

- Configuration to simulate human behavior
  - The Actions per minute(APM) and the response time(350ms) was set to be of an average player of the game, to be on a fair stage with the human player.
  - This was made to ensure that the AI does not win by insanely clicking during the map, but by utilizing strategies.

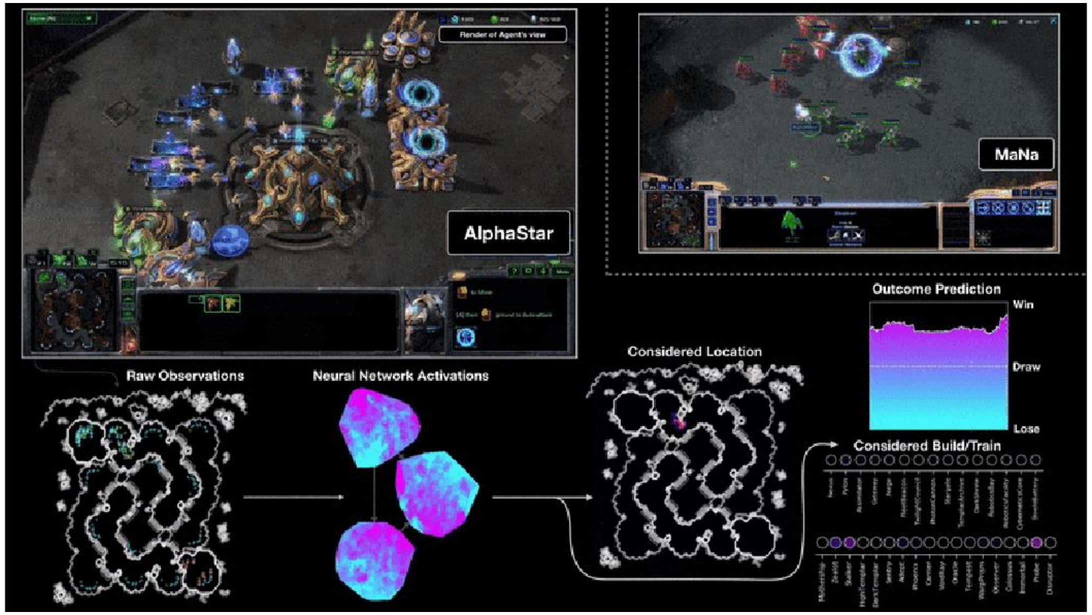


https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii

Highlights - StarCraft II/ DeepMind

- Input to AlphaStar
  - At each time-step (2 million frames per 2 seconds), AlphaStar receives an observation that includes a list of all observable units and their attributes. (these include the opponent units as seen by the player's own units only)

- Output from AlphaStar
  - outputs what action type to take out of several hundred (for example, move or build worker).
  - outputs whom to issue that action to, for any subset of the agent's units.
  - outputs where to target, among locations on the map or units within the camera view.
  - outputs when to observe and act next.



https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii

Highlights - StarCraft II/ DeepMind

- Metrics
  - For every training agent in the league, AlphaStar runs 16,000 concurrent StarCraft II matches and 16 actor tasks to perform inference.
  - The training time for the initial version was about 2 weeks.
  - Uses Reinforcement learning on a 100,000 CPU and bots learn from every game they play.
  - Supervised learning is performed on a dataset of 971,000 replays played on StarCraft II versions 4.8.2 to 4.8.6 from the top 22% of players.

- AlphaStar uses a model-free, end-to-end learning approach to playing StarCraft II that overcomes the difficulties of search-based methods which result from imperfect models and is applicable to any domain that shares some of the challenges present in StarCraft.

Highlights - StarCraft II/ DeepMind

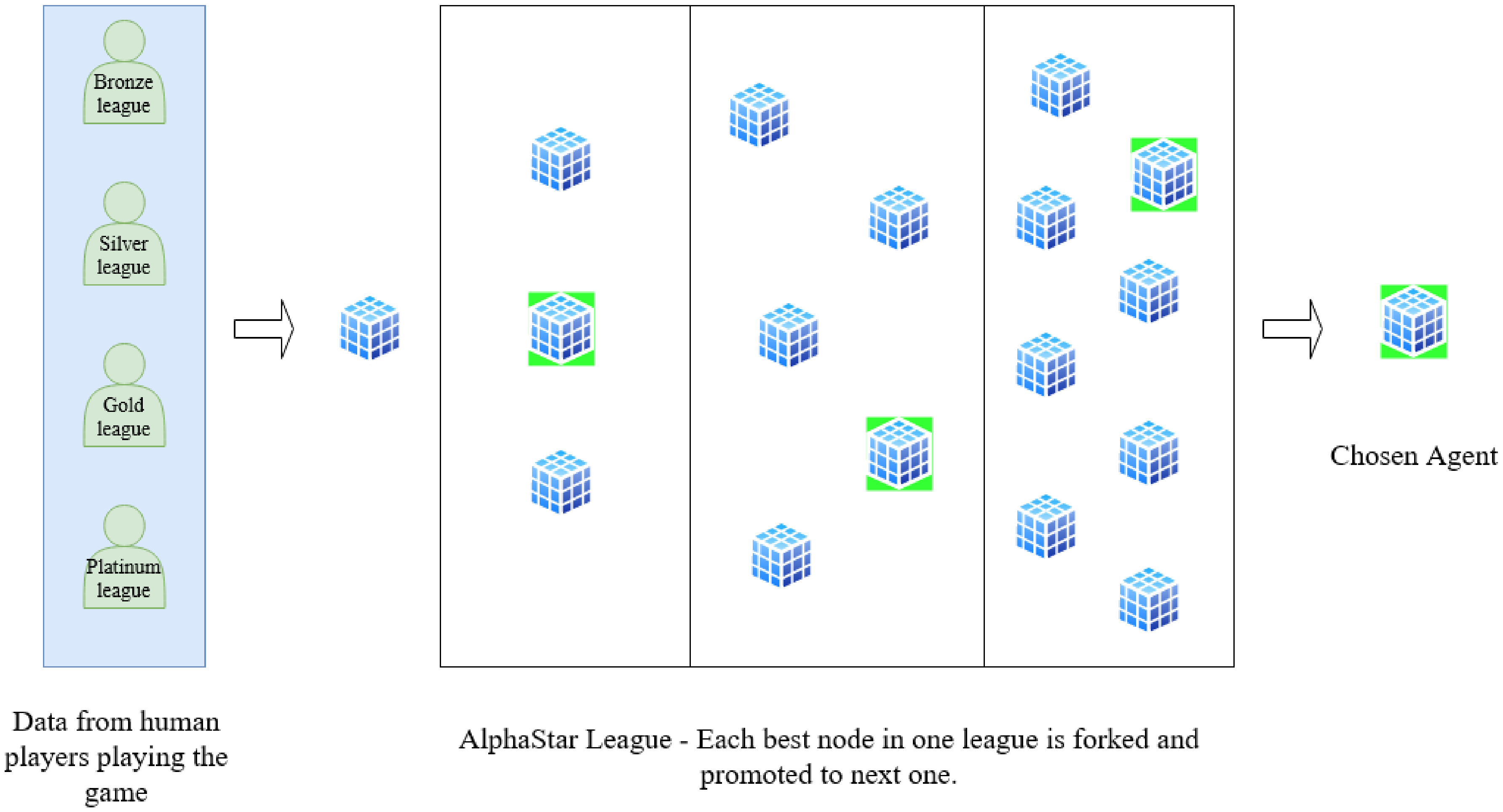
- Building blocks of Alpha Star are reasonably general and are meant to apply the similar algorithms to various tasks and learn by itself like weather prediction and climate modeling.

- Training Phase
  - AlphaStar uses a combination of new and existing general-purpose techniques for neural network architectures, imitation learning, reinforcement learning, and multi-agent learning.
  - Observations of player and opponent units are processed using a self-attention mechanism.
  - To integrate spatial and non-spatial information, scatter connections are used.
  - To deal with partial observability, the temporal sequence of observations is processed by a deep long short-term memory (LSTM) system.
  - To manage the structured, combinatorial action space, the agent uses an auto-regressive policy and recurrent pointer network.

Highlights - StarCraft II/ DeepMind

- Training Phase
  - Supervised Learning
    - Here, the parameters are updated to optimize Kullback–Leibler (KL) divergence between its output and human actions sampled from a collection of replays.
    - Updates are applied using Adam Optimizer along with L2 regularization.
  - Reinforcement learning
    - After supervised learning, the agent parameters were subsequently trained by a reinforcement learning algorithm that is designed to maximize the win-rate against a mixture of opponents. The opponent is decided by a multi-agent procedure.
    - It is based on a off-policy learning algorithm where the current policy is updated from the experience generated by a previous policy. This kind of behavior on self-learning agents work in large action spaces.
    - Updates were applied asynchronously on replayed experiences.

Highlights - StarCraft II/ DeepMind



Highlights - StarCraft II/ DeepMind

- Training Phase – League for AlphaStar
  - The league is trained using three main agents (one for each StarCraft race), three main exploiter agents (one for each race), and six league exploiter agents (two for each race).
  - These agents differ primarily in their mechanism for selecting the opponent mixture.
    - First, the main agents utilize a prioritized fictitious self-play (PFSP) mechanism where the agent selects the mixture probabilities proportionally to the win rate of each opponent against the agent in order to overcome the most problematic opponents.
    - Second, main exploiter agents play only against the current iteration of main agents. Their purpose is to identify potential exploits in the main agents; the main agents are thereby encouraged to address their weaknesses.
    - Third, league exploiter agents use a similar PFSP mechanism to the main agents but are not targeted by main exploiter agents. Their purpose is to find systemic weaknesses of the entire league

Highlights - StarCraft II/ DeepMind

- Key learnings
    - AI Agent learns when to engage and when to backout with a slight life left of the hero.
    - It can assess whether a fight is worth pursuing. Or if said other way round, if an AI is fighting you, it certainly is having a high chance of winning.
    - In an unprecedented move by human player, it kills 2 players of the opposite team before being killed by the human player team, almost like a bait to come out favorably.
    - These unprecedented moves can be studied by the players to further improve their gaming skills, aiding the human capability of thinking.
    - The learnings of the AI agent can be used to work on much bigger things like weather prediction and climate modelling.

References

- Automatic player identification in Dota 2
  - https://arxiv.org/abs/2008.12401
- Neural network surgery with sets
  - https://arxiv.org/abs/1912.06719
- Dota 2 with large scale deep reinforcement learning
  - https://arxiv.org/abs/1912.06680
- AlphaStar: Grandmaster level in StarCraft II using multi-agent reinforcement learning
  - https://deepmind.com/blog/article/AlphaStar-Grandmaster-level-in-StarCraft-II-using-multi-agent-reinforcement-learning
- AlphaStar: Mastering the Real-Time Strategy Game StarCraft II
  - https://deepmind.com/blog/article/alphastar-mastering-real-time-strategy-game-starcraft-ii
- DeepMind and Blizzard open StarCraft II as an AI research environment
  - https://deepmind.com/blog/announcements/deepmind-and-blizzard-open-starcraft-ii-ai-research-environment
- StarCraft II: A New Challenge for Reinforcement Learning
  - https://arxiv.org/abs/1708.04782