**Karsten Boettcher**

**CS 5950**

**RL Paper Review**

Paper: Playing Atari with Deep Reinforcement Learning

Source: Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. & Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning. NIPS Deep Learning Workshop 2013. (<https://arxiv.org/abs/1312.5602>)

Summary: Deep reinforcement learning was the “first deep learning model to successfully learn control policies directly from high-dimensional sensory input using reinforcement learning.” Previously, the most successful RL application models have had to use their own handmade input as feature detection rather than relying on its visual and auditory data. The ability to process this data becomes increasingly difficult with larger quantities of training data. The paper shows its use of convolutional neural networks to overcome these challenges.

The goal for this paper was to use a single neural network agent that could learn as many games as possible without being given any specific handmade references to these games. Instead, the researchers wanted to mirror the references a human new to the game would get: video input, a list of possible actions, and signals to indicate success or failure within each game. The theory of learning, then, will be to made decisions with prior actions in mind creating sequences of plays that form strategies to win/optimize the game. The paper suggests this method leads to a “a large but finite Markov decision process (MDP) in which each sequence is a distinct state.” This contrasts traditional action to action decision making processes by RL methods. They propose an algorithm that is both model-free and off-policy. They also propose the use of experience replay, the ability to store experiences from previous iterations in a pool, to avoid trends that might get the agent stuck in a non-optimal local minima. The agent was able to successfully play 6 of 7 Atari games, outperforming previous models in all 6 and human experts in 3.

Paper: Dueling Network Architectures for Deep Reinforcement Learning

Source: Wang Z., Schaul T., Hessel M., van Hasselt H., Lanctot M., de Freitas N. Google DeepMind. (2016). Dueling Network Architectures for Deep Reinforcement Learning. (<https://arxiv.org/abs/1511.06581>)

Summary: The paper begins by recognizing many advancements made in RL but noting that many of these advancements rely on “conventional architectures, such as convolutional networks, LSTMs, or auto-encoders.” The researchers suggest the creation of an architecture specifically for deep RL called the dueling network. The network aims to separate the representation of state values and action advantages given a state. The model will have two streams: one for state values and one for action advantages, as well as a shared CNN for input. A special layer then combines the output of these streams and produces an estimate for the state action value Q. The idea comes from the researchers’ theory that an action doesn’t need to be chosen at every existing state, and when they are some actions may become irrelevant. The paper finds the dueling architecture could identify correct values during policy estimation faster compared to single stream architectures such as the DQN agent. The dueling network is shown to outperform single stream architectures, showing dramatic improvement across around 75% of the games played.

Paper: Robot Soccer Using Deep Q Network

Source: J. Kim, B. Kim, J. Yoon, M. Lee, S. Jung and J. y. Choi, "Robot Soccer Using Deep Q Network," 2018 International Conference on Platform Technology and Service (PlatCon), Jeju, Korea (South), 2018, pp. 1-6, doi: 10.1109/PlatCon.2018.8472776. (<https://ieeexplore-ieee-org.dist.lib.usu.edu/document/8472776>)

Summary: This paper proposes a DQN algorithm for a robot aimed to compete in the Small Size League (SSL) of RoboCup, an annual international joint project that aims to develop fully autonomous soccer-playing robots that can win against the world champion human team by 2050. The research centers on an algorithm used for “Markov Decision Process (MDP) problems, with the dynamics of the environment unknown, applied to develop scoring [strategies] under various situations.” The goal is that these scoring strategies will lead to better decisions made by the robots during a match. The researchers begin training their robots with epsilon-greedy random exploration. A training model is pitted against a target model, where the target model will perform optimal actions determined from the training model’s previous session. Their DQN method competed against three opposition scenarios: random movement, a simple rule-based algorithm, and a complex rule-based algorithm. Each game lasts 300 seconds. The research showed the DQN significantly outscored each of the opposition scenarios.

Paper: On the Potential of Rocket League for Driving Team AI Development

Source: Y. Verhoeven and M. Preuss, "On the Potential of Rocket League for Driving Team AI Development," 2020 IEEE Symposium Series on Computational Intelligence (SSCI), Canberra, ACT, Australia, 2020, pp. 2335-2342, doi: 10.1109/SSCI47803.2020.9308248. (<https://ieeexplore-ieee-org.dist.lib.usu.edu/document/9308248>)

Summary: The paper begins by acknowledging the role computer games have played in AI research and development and gives a brief on some of the notable developments related to team-based AI in games. It mentions Robocup as one of these advancements and proposes that Rocket League – a hybrid soccer game with rocket-powered cars – suits as a better environment for soccer-based team AI research than other methods (Robocup included). Some notable points being: there has already been some successful machine learning/AI applications in Rocket League for high precision tasks, the availability of human replay data, it poses an interesting test case for human/AI team coordination due to the high level of spatio-temporal data analysis needed at high levels of play.

Paper: Asynchronous Methods for Deep Reinforcement Learning

Source: Mnih V., Badia A.P., Mirza M., Graves A., Harley T., Lillicrap T.P., Silver D., & Kavukcuoglu K. Google DeepMind. (2016). Asynchronous Methods for Deep Reinforcement Learning. ICML 2016. arXiv:1602.01783. (<https://arxiv.org/abs/1602.01783>)

Summary: While deep RL algorithms based on experience replay have had unprecedented success, there lies a drawback in the amount of memory and computation power required per real interaction. Additionally, they require off-policy learning algorithms that can update from data generated by an older policy. The researchers propose rather than using these methods, they asynchronously execute multiple agents in parallel. This method decorrelates the agents’ data into a more stationary process and allows for a much larger spectrum of on-policy RL methods (as well as off-policy RL methods). To accomplish this, the researchers use actor-learners executed on separate cores of a CPU (which takes out a large communications cost incurred by using multiple machines). Running multiple actor-learners in parallel will be more likely to expose each actor to different parts of the same environment. Additionally, different explorative policies can be used for each agent to maximize results. The researchers determined the asynchronous training methods produced better results, using less resources, in less time than previous methods.