# **Deep Reinforcement Learning on OpenAI Gym Games**

|  |  |
| --- | --- |
| Yangxin Zhong1  Stanford University  Computer Science Department  yangxin@stanford.edu | Jiajie He1  Stanford University  Mechanical Engineering Department  Jiajie@stanford.edu |

Lingjie Kong1

Stanford University

Stanford Center for Professional Development

ljkong@stanford.edu

**Abstract**

*Reinforcement Learning (RL) is an area of machine learning figuring out how agents take actions in an unknown environment to maximize its rewards. Unlike classical Markov Decision Process (MDP) in which agent has full knowledge of its state, rewards, and transitional probability, reinforcement learning utilizing exploration and exploitation for the model uncertainty. Under the condition that the model usually has a large state space, a neural network (NN) is used to correlate its input state to its output actions to maximize the agent’s rewards. We will implement Deep Q-Network (DQN) to play OpenAI Gym to beat its benchmarks for our project.*

# **Introduction**

Reinforcement Learning (RL) is inspired by behaviorist psychology regarding taking the best actions to optimize agent’s reward at a specific state. It has been studies in many disciplines such as control theory, information theory, statistics, and so on.

Classical decision-making problem was formed as a Markov Decision Process (MDP) which people need to have full knowledge of the environment and carefully model its state reward, transitional reward, as well as transitional probability. Due to this limitation, reinforcement learning with Q learning was developed to let agent explore to find possible optimal solution and exploit to optimize the good solutions found up to now.

Under the condition that corollate the large input state space to agent action is not accomplished through look up table like MDP, neural network is used to capture the non-linear relationship between input and output. During the training, forward and backward propagation will be used to train the weight at each layer. With fully trained model, it will be used to inference based on the current state input, what will be the optimal action to take in order to maximize its rewards.

# **Related Work**

Figuring out how to control agent from high-dimensional inputs like vision input is one of the biggest challenge of reinforcement learning. Most successfully RL model before is based on carefully selected feature with linear combined values. Obviously, the quality of the selected features representation will largely influence the performance.

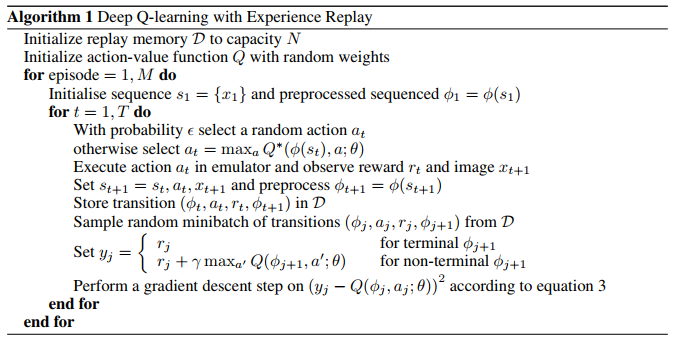
With the fast development of computer vision, it leads to some break throughs on how to extract the feature representation more efficiently by using more efficient models [1]. All these methods utilize ideas of neural network structures such as Convolutional Neural Networks, (CNN), Recurrent Neural Networks (RNN), Multilayer Perception, Boltzmann Machine Graphic Model, and so on.

Besides the challenge from the input feature representation, reinforcement learning presents other challenges. First, traditional machine learning requires large number of carefully labelled data. However, reinforcement learning algorithm have to learn from scalar rewards which is most of the time noisy and delayed from the current state. Second, unlike most supervised learning algorithm which assume the independence of samples, RL’s sample are highly correlated.

Q-Learning was algorithm [2] with stochastic gradient descent is often used to train reinforcement learning model.

# **Approach**

Deep Reinforcement Learning with experience replay [3] will be used to train the agent games in OpenAI Gym. We hope to achieve at least the same performance that was mentioned in Google’s Playing Atari with Deep Reinforcement Learning paper.



# **Experiment**

# **Conclusion**

# References

[1] A. Krizhevsky, I. Sutskever, and H. Geoffrey E., “ImageNet Classification with Deep Convolutional Neural Networks,” *Adv. Neural Inf. Process. Syst. 25*, pp. 1–9, 2012.

[2] C. J. C. H. Watkins and P. Dayan, “Q-learning,” *Mach. Learn.*, vol. 8, no. 3–4, pp. 279–292, 1992.

[3] D. Zoran *et al.*, “Playing Atari with Deep Reinforcement Learning,” *arXiv*, vol. 32, no. Ijcai, pp. 1–9, 2016.