

Meets Specifications

Awesome ⭐⭐⭐⭐⭐

You have done great work and acquired all the concepts needed in this project. Congratulations 🎉

I have added a few suggestions you can incorporate them in your present project especially if you are sharing it on GitHub repo, and definitely keep them in mind for the future projects. All the best.

Few useful resources:

- You can use [PyTorch-summary](#) to visualize your architecture.
- The current State-Of-The-Art (SOTA) RL algorithm for discrete action space is [Rainbow](#). This is a [great post](#) explaining how to combine all of these improvements together.

Training Code

✓	The repository (or zip file) includes functional, well-documented, and organized code for training the agent.
	<p>The code is well documented. Keep the good work. 👍</p> <p>Some guidelines to make your repository standout:</p> <ul style="list-style-type: none">The Best of the Best Practices (BOBP) Guide for PythonPython Best Practices: 5 Tips For Better Code
✓	The code is written in PyTorch and Python 3.
	<p>The code is written in PyTorch and Python 3. Both PyTorch and TensorFlow are widely used in deep learning projects. You can get a good idea of the comparison between the two from these two resources:</p> <ul style="list-style-type: none">PyTorch vs TensorFlow—spotting the differenceTensorflow or PyTorch : The Force is Strong with which One? <p>There's a free Pytorch Udacity course that you may want to check out: https://in.udacity.com/course/deep-learning-pytorch--ud188</p>
✓	The submission includes the saved model weights of the successful agent.
	<p>Saved model weights are included. <code>Dueling_model.pth</code></p>

README

✓	The GitHub (or zip file) submission includes a <code>README.md</code> file in the root of the repository.
	<p><code>Readme.md</code> is included.</p>
✓	The README describes the the project environment details (i.e., the state and action spaces, and when the environment is considered solved).
	<p>The Readme file describes the project environment details.</p>
✓	The README has instructions for installing dependencies or downloading needed files.
	<p>The Readme is a place where you can let the user know about installation dependencies and downloading needed files, so it would be good to include in the readme:</p> <ul style="list-style-type: none">How to download the project environment?Any additional installations you require for the project.Also remember to add the all the modules that you have imported along with their version as requirements (dependencies) of this project, so that anyone can reproduce your results.
✓	The README describes how to run the code in the repository, to train the agent. For additional resources on creating READMEs or using Markdown, see here and here.
	<p>The instructions to train the network are included. Good Job.</p> <p>You can make the instructions more clear by including the cell numbers needed to:</p> <ul style="list-style-type: none">Train the Agent.Run the trained Agent after loading the stored weights

Report

✓	The submission includes a file in the root of the GitHub repository or zip file (one of <code>Report.md</code> , <code>Report.ipynb</code> , or <code>Report.pdf</code>) that provides a description of the implementation.
	<p><code>Report.md</code> is included</p>
✓	The report clearly describes the learning algorithm, along with the chosen hyperparameters. It also describes the model architectures for any neural networks.
	<p>Great work, you have added the description of your learning algorithm, however, you can improve your answer by adding the following points:</p> <ul style="list-style-type: none">The hyperparameters that you have used, and how you decided these values (experimentation or taken from paper?)The model architecture of Actor and Critic. (Number of layers, number of units per layer, any experimentation that you have done with different architecture)Also a brief description or pseudocode of the DQN and DDQN algorithm is better than including the code in the report.
✓	A plot of rewards per episode is included to illustrate that the agent is able to receive an average reward (over 100 episodes) of at least +13. The submission reports the number of episodes needed to solve the environment.
	<p>Great work in achieving an average reward (over 100 episodes) of at least +13. 🎉</p>
✓	The submission has concrete future ideas for improving the agent's performance.
	<p>Nice work suggesting the ideas for future improvement. Adding what part of the present algorithm you want to improve with your future ideas will be good.</p>