1. Implement Advanced RL algorithm

   a. Advanced RL algorithm: D3QN

   b. D3QN is Value-based algorithm, by learning how to find the best Value Function, try to find the best policy.

   Policy Gradient is Policy-based algorithm, by optimizing the long term total reward , try to find the best policy.

   c. Construct D3QN network with 3 linear and 2 relu layers, and D3QN agent which contain replay buffer.

   In every steps, after target network compute target, we will compute loss between network output and target.

   Every120 steps, target network will copy network params.

2. a. How does the objective function of "PPO-ptx" differ from the “PPO” during RL training as used in the InstructGPT paper?

   b. Also, what is the potential advantage of using "PPO-ptx" over “PPO” in the InstructGPT paper? Please provide a detailed analysis from their respective objective functions.

   a.

$$\begin{aligned}\text{objective}\,(\phi) = & E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}\left[r_\theta(x,y) - \beta\log\left(\pi_\phi^{\mathrm{RL}}(y\mid x)/\pi^{\mathrm{SFT}}(y\mid x)\right)\right] + \\ & \gamma E_{x\sim D_{\text{pretrain}}}\left[\log(\pi_\phi^{\mathrm{RL}}(x))\right]\end{aligned}$$

   For “PPO-ptx” , γ control the strength of the KL penalty and pretraining gradients respectively, For “PPO” γ = 0.

   b.”PPO”

$$\text{objective}\,(\phi) = E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}\left[r_\theta(x,y) - \beta\log\left(\pi_\phi^{\mathrm{RL}}(y\mid x)/\pi^{\mathrm{SFT}}(y\mid x)\right)\right]$$

   “PPO-ptx”

$$\begin{aligned}\text{objective}\,(\phi) = & E_{(x,y)\sim D_{\pi_\phi^{\mathrm{RL}}}}\left[r_\theta(x,y) - \beta\log\left(\pi_\phi^{\mathrm{RL}}(y\mid x)/\pi^{\mathrm{SFT}}(y\mid x)\right)\right] + \\ & \gamma E_{x\sim D_{\text{pretrain}}}\left[\log(\pi_\phi^{\mathrm{RL}}(x))\right]\end{aligned}$$

   Both of them concern KL reward, which mitigate overoptimization of the reward model, but “PPO-ptx” concern pretraining loss, which can help fixing regressions on public NLP datasets.