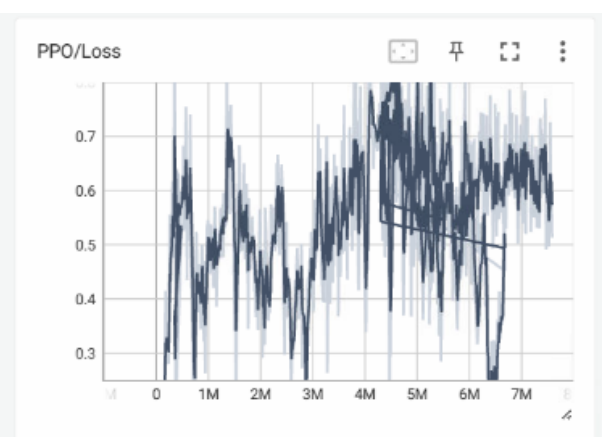
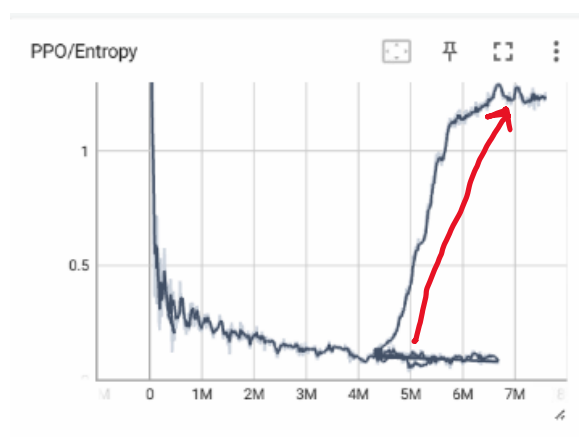
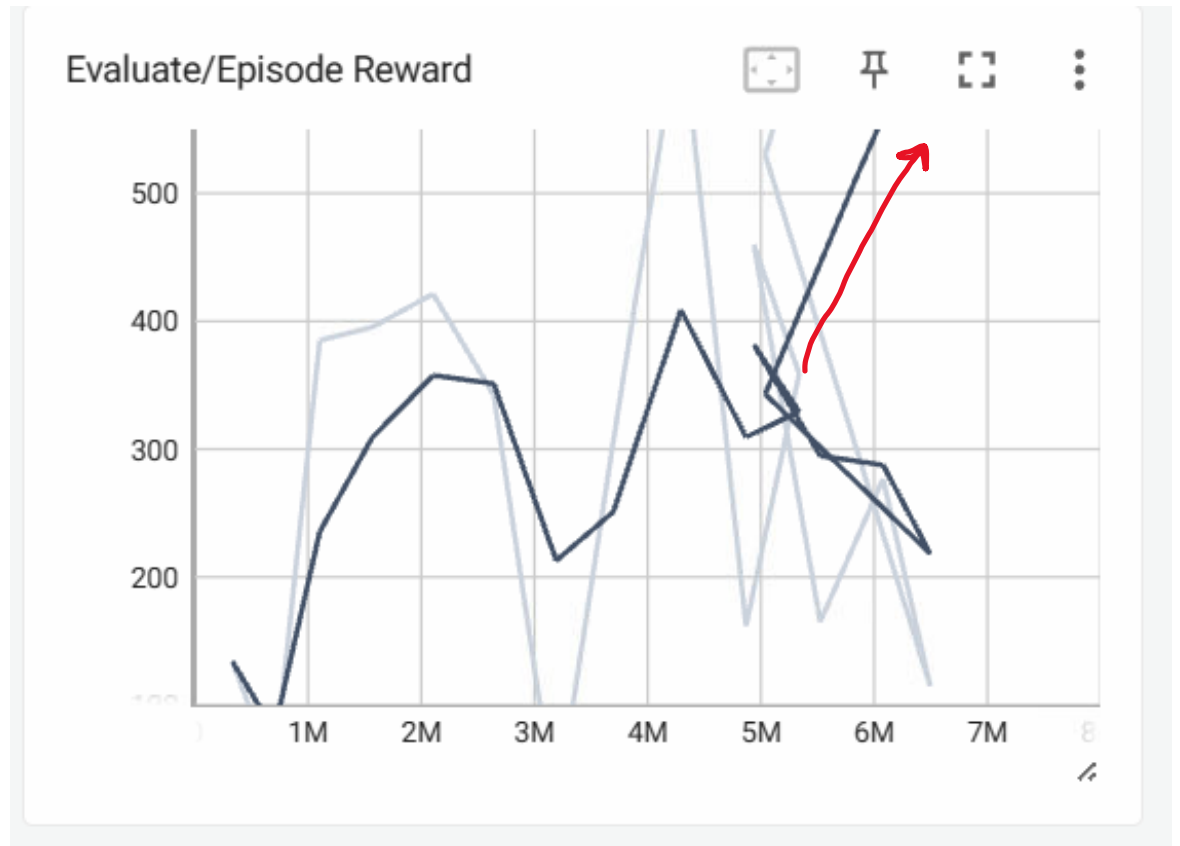
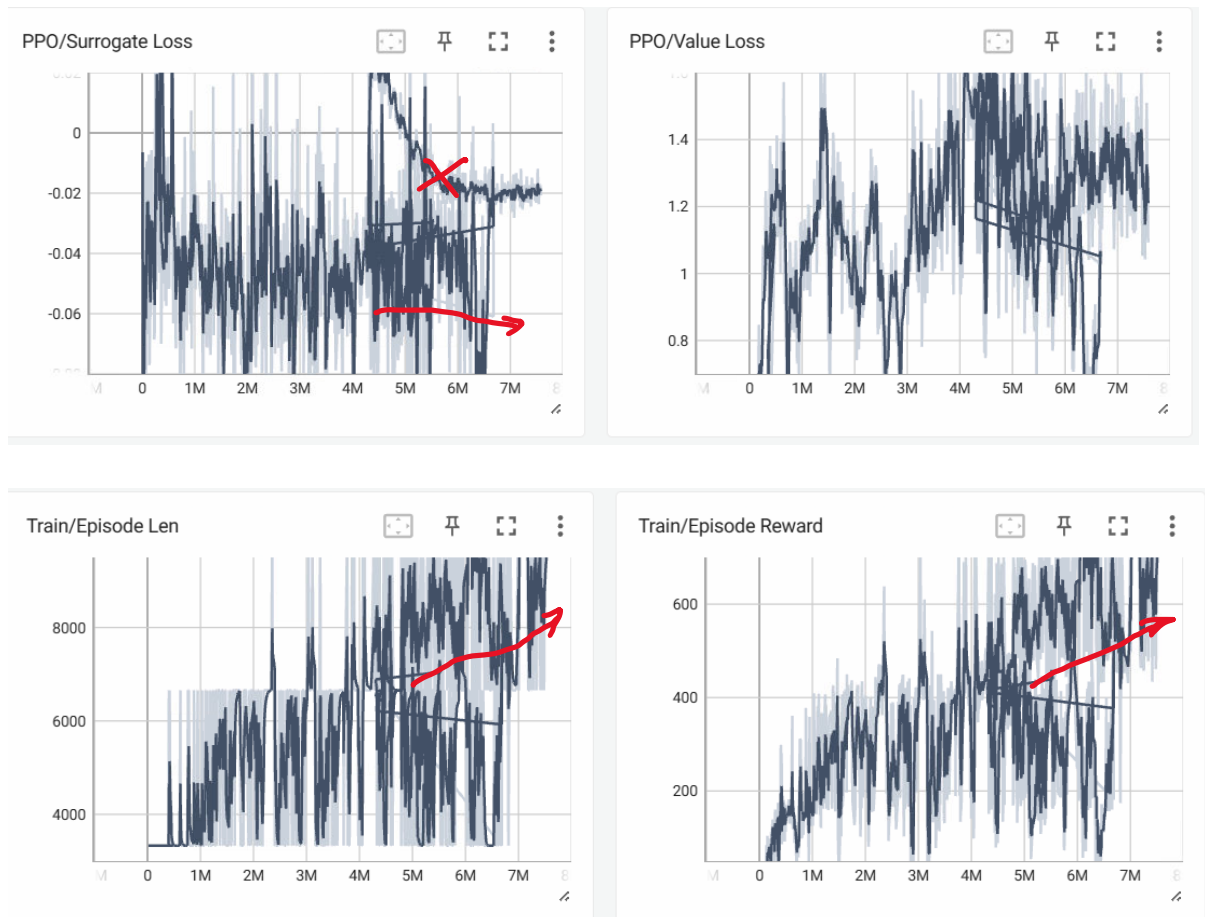


- Experimental Results (30%)

*我在前 4M step 因為 clip epsilon 設錯，導致前 4M 學習效果並不是很好，在後面 4M 重新修正結果後，效果有明顯上升

*原本跑到 7M 修正上述錯誤並從 4M 時的 model 重新開始訓練。而誤會繪圖的方式錯把初始 step 設為 4M(應該要為最後跑)，因此於 4M 出現分岔。我以紅色箭頭註記修正後的紀錄結果





```

Anaconda Prompt - python _main.py
2560/10000 [6608221/100000000] episode: 287 episode reward: 710.0 episode len: 9984
9616/10000 [6614876/100000000] episode: 288 episode reward: 468.0 episode len: 6656
avg: 0.5408911595586844 episode: 288 episode reward: 468.0 episode len: 6656
6272/10000 [6621531/100000000] episode: 289 episode reward: 497.0 episode len: 6656
avg: 0.6283123176603008 episode: 289 episode reward: 497.0 episode len: 6656
2528/10000 [6628186/100000000] episode: 290 episode reward: 494.0 episode len: 6656
avg: 0.6078444507944885 episode: 290 episode reward: 494.0 episode len: 6656
2512/10000 [6638169/100000000] episode: 291 episode reward: 767.0 episode len: 9984
9568/10000 [6644824/100000000] episode: 292 episode reward: 493.0 episode len: 6656
avg: 0.571574428630035 episode: 292 episode reward: 493.0 episode len: 6656
9552/10000 [6654807/100000000] episode: 293 episode reward: 783.0 episode len: 9984
avg: 0.5385145787520522 episode: 293 episode reward: 783.0 episode len: 9984
9328/10000 [6664790/100000000] episode: 294 episode reward: 727.0 episode len: 9984
avg: 0.6672512075728093 episode: 294 episode reward: 727.0 episode len: 9984
6192/10000 [6671445/100000000] episode: 295 episode reward: 465.0 episode len: 6656
avg: 0.596780831122637 episode: 295 episode reward: 465.0 episode len: 6656
2848/10000 [6678100/100000000] episode: 296 episode reward: 459.0 episode len: 6656
9504/10000 [6684755/100000000] episode: 297 episode reward: 483.0 episode len: 6656
avg: 0.5345443017438864 episode: 297 episode reward: 483.0 episode len: 6656
6160/10000 [6691410/100000000] episode: 298 episode reward: 499.0 episode len: 6656
avg: 0.7222742090520309 episode: 298 episode reward: 499.0 episode len: 6656
2316/10000 [6698065/100000000] episode: 299 episode reward: 495.0 episode len: 6656
6472/10000 [6704720/100000000] episode: 300 episode reward: 469.0 episode len: 6656
avg: 0.7222742090520309 episode: 300 episode reward: 469.0 episode len: 6656
evaluating...
episode 1 reward: 991.0
episode 2 reward: 1057.0
episode 3 reward: 765.0
average score: 937.6666666666666

Value Loss: 1.151011913366572 Entropy: 1.2568362879296928
Value Loss: 1.3241393847713874 Entropy: 1.2810574847289247
Value Loss: 1.2930046624141778 Entropy: 1.2837564845160634
Value Loss: 1.2135832549764032 Entropy: 1.2942197877796155
Value Loss: 1.1457986015984118 Entropy: 1.2875720302295568
Value Loss: 1.401162573491291 Entropy: 1.2924535726250284
Value Loss: 1.2642251639130084 Entropy: 1.2999460317546465
Value Loss: 1.1815378362137685 Entropy: 1.2842283830920107
Value Loss: 1.5115008846170082 Entropy: 1.295024642790718

```

- Answer the questions (bonus) (20%)
 1. PPO is an on-policy or an off-policy algorithm? Why? (5%)
 PPO 是 on-policy，因為它使用正在 train 的 model 來與環境互動
 2. Explain how PPO ensures that policy updates at each step are not too large to avoid destabilization. (5%)
 $r_t(\theta)\bar{A}_t$ 表示每次往 gradient 最大方向更新的距離，如果此值太大容易造成不穩定。PPO 使用 clip 來使得每次更新的距離(move length)介於

$$move\ length < (1 - \epsilon)\bar{A}_t, \text{ if } \bar{A}_t < 0$$

$$move\ length < (1 + \epsilon)\bar{A}_t, \text{ if } \bar{A}_t > 0$$
 如此保證不會一次太大步而不穩定

3. Why is GAE-lambda used to estimate advantages in PPO instead of just one-step advantages? How does it contribute to improving the policy learning process? (5%)

$$\widetilde{A}_t^{GAE(\gamma, \lambda)} = (\delta_t^V + (\lambda\gamma)^1 \delta_{t+1}^V + \dots + (\lambda\gamma)^k \delta_{t+k}^V + \dots)$$

$$\text{One-step advantage } \widetilde{A}_t^{GAE(\gamma, 0)} = \widetilde{A}_t^{(1)} = TD_error$$

其只考慮了下一步的結果所產生的 TD error 並以此來衡量 action 的好壞，但由於整個過程是 **stochastic** 可能只是剛好此次結果好/不好下次就是不好/好，其後續有很大的 **variance**，因此採用 **Lambda-GAE** 方式多考慮更長遠的結果來減少 **variance** 以此減少 **variance** 避免偶而遇到的極糟情況讓整個 **model** 壞掉。

4. Please explain what the lambda parameter represents in GAE-lambda, and how adjusting the lambda parameter affects the training process and performance of PPO? (5%)

Lambda 的大小用來調整要考慮非常長遠還是近期就好，當 **Lambda** 越小時，**model** 的 **bias** 越低但隨之而來的 **variance** 也越高，對 **noise** 的對抗性越低，容易在訓練過程 **model** 一遇到很糟狀況而被拖垮；反之如果太高則 **model** 的準確度越低(**bias** 大)，但對 **noise** 容忍度較高幫助 **model** 穩定