Deep Q-Network

AAIS in PKU　陈伟杰 1901111420

April 9, 2020

1 Problem Setting

已知足球环境 google football，采用 Simple115 的向量表示状态，行动空间采取默认的 19 维行动空间。运用 DQN 方法对 academy_empty_goal 问题进行求解：

$$\mathcal{L}(\mathbf{w}) = \mathbb{E}_{s,a,r,s' \sim \mathcal{D}} \left[\left(r + \gamma \max_{a'} Q\left(s', a'; \mathbf{w}^{-}\right) - Q(s, a; \mathbf{w}) \right)^2 \right] \tag{1}$$

2 Experiment Result

实验中参数的设置可以查看 data.py，主要采用渐进的 ϵ-贪婪法，即随着训练次数增加，分阶段减少自由探索的概率。此外，q_{eval} 和 q_{target} 均采用 2 层全连接的网络结构，每隔一定训练次数进行同步。

对于 academy_empty_goal，为了减少无用操作从而减轻训练难度，因此将行动域限定为 [3,4,5,6,7,12,13,14,15,17,18]，即 top，top-right，right，bottom-right，bottom，shot，sprint，release-direction，releas-sprint，dribble，release-dribble。

数值结果如下图，其中 reward 采用滑动平均的处理方法使得图线更直观（即 $\hat{r}[i] = \frac{1}{k}\sum_{j=i}^{i+k} r[j]$）

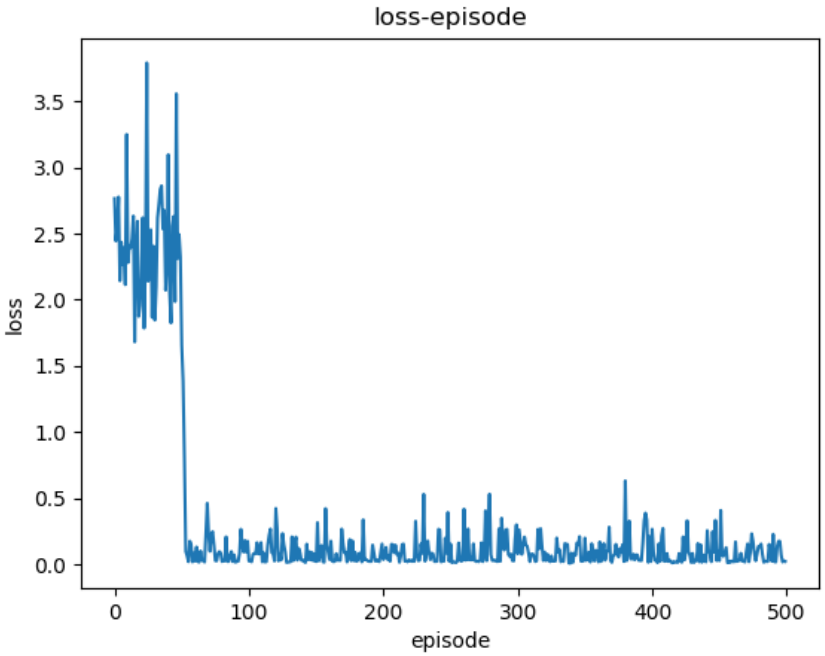

Figure 1: loss 与迭代次数的关系图

从图线可以看出，随着 episode 的增加，loss 下降，滑动平均的 reward 逐渐上升，steps 略有下降。值得注意的是，实验中对随机种子的依赖比较明显。

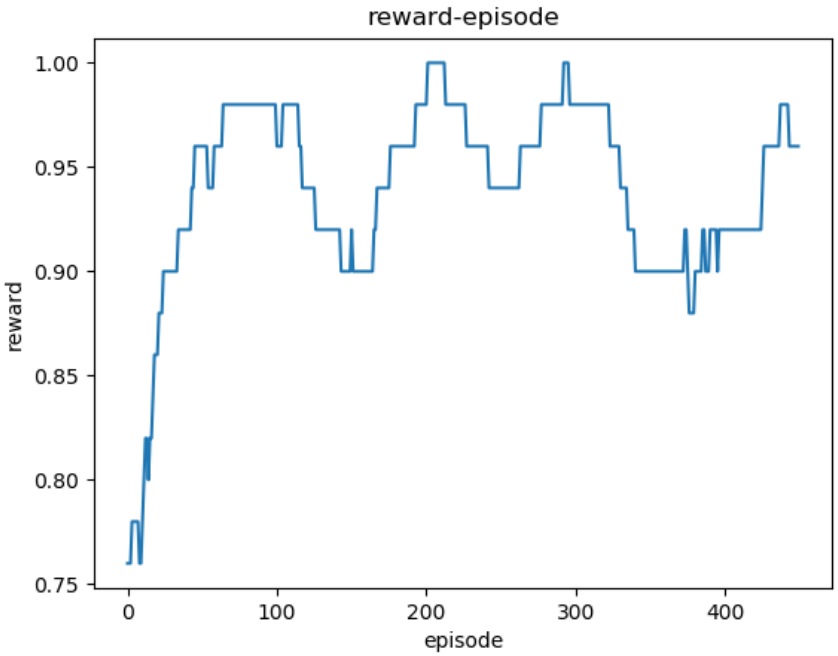

Figure 2: reward 与迭代次数的关系图

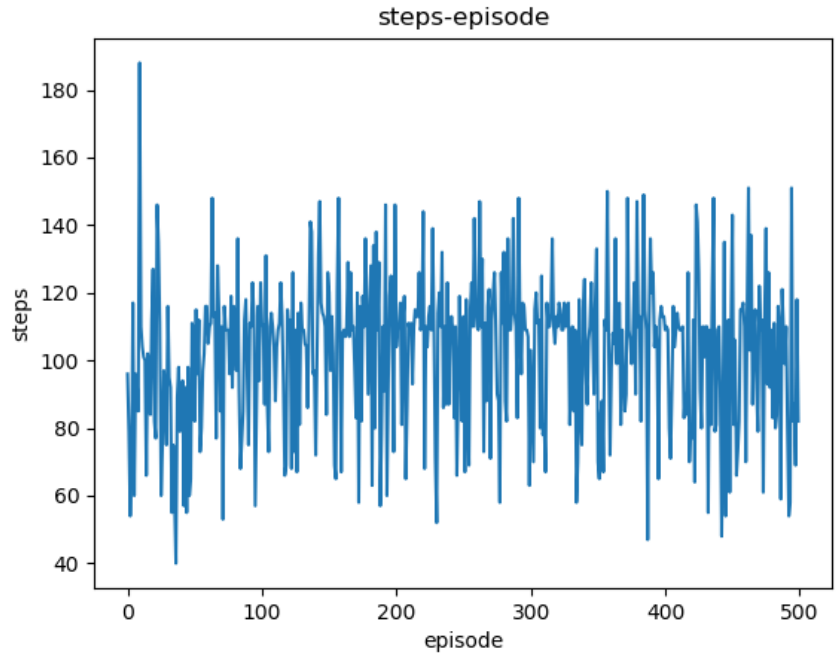

Figure 3: steps 与迭代次数的关系图

3 README

代码包括 data.py，model.py 和 train.py 三个程序，需要 google football 环境。直接运行 train.py 可以得到上述结果

- 电脑：MacBook Pro(15-inch, 2019)
- 处理器：2.6 GHz 6-Core Intel Core i7
- 内存：16 GB 2400 MHz DDR4
- 操作系统：macOS Catalina version 10.15.1
- 语言：python3.6
- 实验环境：docker+ubuntu18.04