

课程作业

Environment: Google Football



包括

- Full Game: 11v11
- Mini Game: 如射空门、带球过人、角球等

github.com/google-research/football



Environment: Install

按照官网教程，若在本地安装，建议ubuntu18.04，python3.6

colab安装训练较为简便，作业的计算量不大，本地安装有问题可在colab端完成，colab无图形界面
官方提供docker安装，无图形界面

In colab

Open our example [Colab](#), that will allow you to start training your model in less than 2 minutes.

This method doesn't support game rendering on screen - if you want to see the game running, please use the method below.

On your computer

Install required apt packages with:

```
sudo apt-get install git cmake build-essential libgl1-mesa-dev libsdl2-dev \
libsdl2-image-dev libsdl2-ttf-dev libsdl2-gfx-dev libboost-all-dev \
libdirectfb-dev libstl-dev mesa-utils xvfb x11vnc libsdl-sge-dev python3-pip
```

Then install the game from GitHub master:

```
git clone https://github.com/google-research/football.git
cd football
pip3 install .
```

This command can run for a couple of minutes, as it compiles the C++ environment in the background. Now, it's time to play!

```
python3 -m gfootball.play_game --action_set=full
```

Environment: Install

Setup (should take < 100 seconds)

```
[ ]  !apt-get update
      !apt-get install libsdl2-gfx-dev libsdl2-ttf-dev

      # Make sure that the Branch in git clone and in wget call matches !!
      !git clone -b v2.0.4 https://github.com/google-research/football.git
      !mkdir -p football/third_party/gfootball_engine/lib

      !wget https://storage.googleapis.com/gfootball/prebuilt_gameplayfootball_v2.0.4.so
      !cd football && GFOOTBALL_USE_PREBUILT_SO=1 pip3 install .
```

Environment: Example

强化学习基本接口：

while done:

```
    steps++  
    get observation  
    set action  
    get reward, done, other_info
```

```
import gfootball.env as football_env  
env = football_env.create_environment(env_name="academy_empty_goal", #->情景  
                                      representation='simple115', #->观察的紧密表达, 115维计算量较小, 也可采用72*96的视觉输入  
                                      number_of_left_players_agent_controls=1, #->在multi agent情况下应为agent数目,  
                                                       #->对方agent均为内置AI  
                                      stacked=False, logdir='/tmp/football',  
                                      write_goal_dumps=False,  
                                      write_full_episode_dumps=False, render=False)  
max_episodes = 10000  
i_episode = 0  
while i_episode < max_episodes:  
    i_episode += 1  
    env.reset()  
    steps = 0  
    done = False  
    while not done:  
        action = env.action_space.sample() #->由rl policy产生  
        obs, rew, done, info = env.step(action) #->在multi agent情况下, 均为array  
        steps += 1
```

Environment: Example

observation

建议simple115模式，计算量小训练更快，也可尝试视觉输入。

action

int型，代表不同动作，每个时间步选取一个，可根据需要禁用某些动作

Top	Bottom	Left	Right
Top-Left	Top-Right	Bottom-Left	Bottom-Right
Short Pass	High Pass	Long Pass	Shot
Keeper Rush	Sliding	Dribble	Stop-Dribble
Sprint	Stop-Moving	Stop-Sprint	Do-Nothing



reward

作业中只可选scoring模式，我方进球为+1，对方进球为-1，其余为0
在multi agent情境中，team内的agents共享同一个global reward
reward相对sparse，可参考针对sparse reward问题的论文加速训练

multi agent

multi agent要求所有智能体同时做出决策，实现方法参见样例代码和github上multi agent support部分



academy_empty_goal



academy_3_vs_1_with_keeper

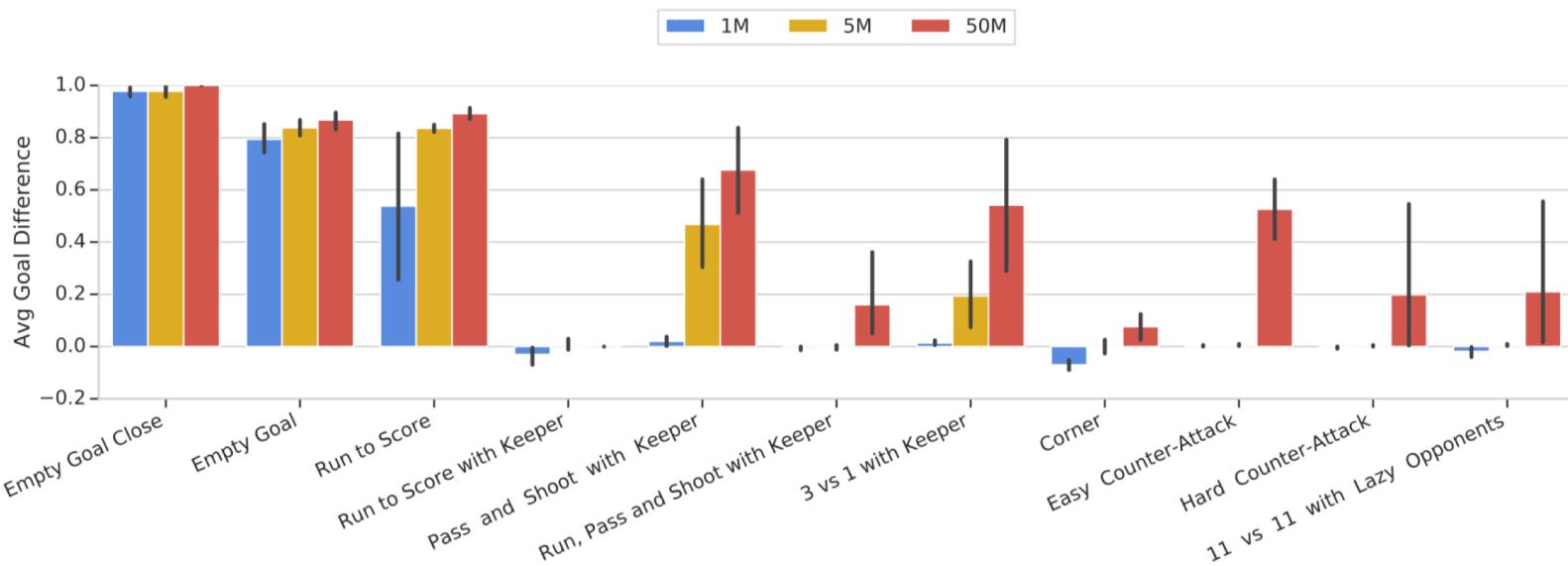


Figure 6: Average Goal Difference on *Football Academy* for IMPALA with SCORING reward.

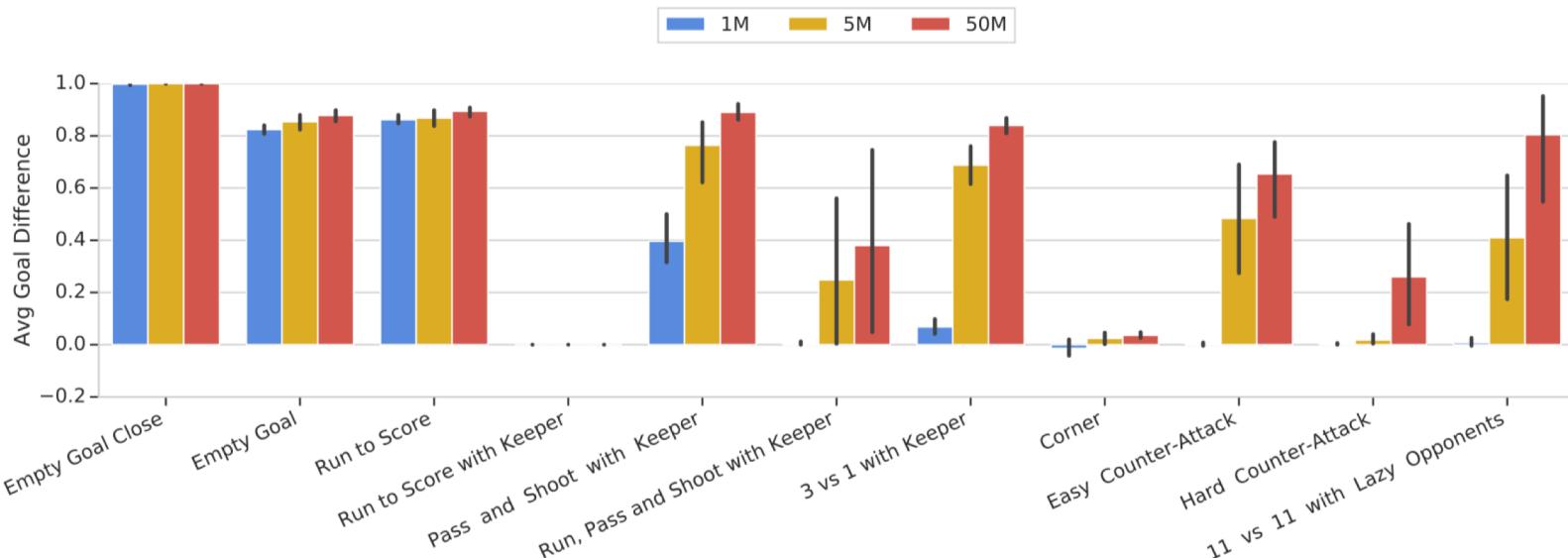


Figure 7: Average Goal Difference on *Football Academy* for PPO with SCORING reward.