

手把手教你如何定制OpenAI Gym学习环境（一）

OpenAI Gym支持定制我们自己的学习环境。有时候Atari Game和gym默认的学习环境不适合验证我们的算法，需要修改学习环境或者自己做一个新的游戏，比如贪吃蛇或者打砖块。已经有一些基于gym的扩展库，比如MADDPG。

我们从定向下一步探索如何建立自己的学习环境。参考链接在文末，我综合了两篇文章结合自己的理解进行实验。OpenAI推荐的文件结构长这个样子：[gym-soccer](#)。

1.首先确定文件结构：

```
custom-gym/
|-- README.md
|-- setup.py
|-- custom_gym/
|   |-- __init__.py
|   |-- envs/
|       |-- __init__.py
|       |-- custom_env.py
|       |-- custom_env_extend.py
```

不用觉得文件多，我们只是搭个框架，里面要么是空的要么内容很少，我会挨个介绍。框架搭好了可以方便以后部署我们的环境。

一级目录为custom-gym. 你可以取任何不冲突的名字。包含setup.py(安装), README.md（说明书）和custom_gym文件夹（环境主体）。

二级目录为custom_gym.包含__init__.py（初始化）和envs文件夹（环境脚本）。

三级目录为envs.包含__init__.py（初始化）和两个测试环境脚本custom_env.py和custom_env_extend.py（当然你也可以只测试一个或者多加几个试试）。

2.手把手配置各个文件：

*custom-gym/README.md*可以空着，等你写好环境再写描述。

custom-gym/setup.py 应该包含：

```
from setuptools import setup

setup(name='custom_gym',          # Secondary directory
      version='0.1',
      install_requires=['gym']    # And any other dependencies foo needs
    )
```

有兴趣的同学可以参考下[OpenAI官方注册格式](#)。

custom-gym/custom_gym/__init__.py 应该包含：

```
from gym.envs.registration import register

register(
    id='custom_env-v0',
    entry_point='custom_gym.envs:CustomEnv',
)
register(
    id='custom_env-v0',
    entry_point='custom_gym.envs:CustomEnv',
    # Format should be xxx-v0, x
    # Explained in envs/__init__
```

```
id='custom_env_extend-v0',  
entry_point='custom_gym.envs:CustomEnvExtend',  
)
```

custom-gym/custom_gym/envs/__init__.py 应该包含:

```
from custom_gym.envs.custom_env import CustomEnv  
from custom_gym.envs.custom_env import CustomEnvExtend
```

custom-gym/custom_gym/envs/custom_env.py 应该长这个样子:

```
import gym  
  
class CustomEnv(gym.Env):  
    def __init__(self):  
        print('CustomEnv Environment initialized')  
    def step(self):  
        print('CustomEnv Step successful!')  
    def reset(self):  
        print('CustomEnv Environment reset')
```

这里省去了render, close等模块, 之后的文章会在更具具体的问题来详细设置。这里只做展示目的。

同理, *custom-gym/custom_gym/envs/custom_env_extend.py* 应该长这个样子:

```
import gym  
  
class CustomEnvExtend(gym.Env):  
    def __init__(self):  
        print('CustomEnvExtend Environment initialized')  
    def step(self):  
        print('CustomEnvExtend Step successful!')  
    def reset(self):  
        print('CustomEnvExtend Environment reset')
```

3.安装与测试:

打开Anaconda Prompt(使用Anaconda安装深度强化学习请[参照这里](#)), 切换目录到一级目录custom-gym,就是包含setup.py这个文件夹。然后运行:

```
pip install -e .
```

注意那个小点不能省。

接下来就可以愉快的测试了。

```
import gym  
import custom_gym  
  
env = gym.make('custom_env-v0')  
env.step()  
env.reset()  
  
env2 = gym.make('custom_env_extend-v0')  
env2.step()  
env2.reset()
```