强化学习论文汇总



张楚珩 🔮

清华大学 交叉信息院博士在读

162 人赞同了该文章

归纳一下目前读到强化学习论文所涉及的话题

一、Model-free RL

主要目标是Stable和Data Efficient,另外希望能够支持High Dimensional Input、支持continuous action space、支持并行计算。

张楚珩: 【强化学习算法 1】DQN	Lecontrol through deep reinf
NICHUS CONTROL WITH DEE NR	UOUS CONTROL WITH DEEP REINFORC NG (Mangi Joseph J. Beet, Linuarie Périn, Norde Hoos, neer Jans De Wille Deep Reinford, Norde Hoos, neer Jans De Wille Deep Reinford, Norde Hoos, neer Jans De Willey, Seattle, no. 1994, 4514412, hours, no. 1994, 4514412,
张楚珩: 【强化学习算法 3】TRPO	Trust Region Policy Optimization and the second optimization of the second optimization opt
张楚珩:【强化学习算法 4】PPO	Policy Optimization Alg Volski, Prafulla Dhariwal, Alec R OpenAl 1p, prafulla, alec, oleg}c
张楚珩: 【强化学习算法 6】ACER ❷ zhuanlan.zhihu.com	PALE DEFINITION ACTION CENTRAL WITH SHOWLD SHAPE AND ACTION OF THE SHAPE AND A
张楚珩: 【强化学习算法 5】A3C	openieromo Nobindo for Desp Rainformanest Careste. Antiferio Company
张楚珩: 【强化学习算法 7】CEM ② zhuanlan.zhihu.com István Szitus, András Lör	urning Tetris Using the Noisy Cross-Entropy Method Brutin Storia ¹ Andrie Létings ¹ on a Physician Storia Chest Deserve with Chest Chest Chest Chest Chest of Chest Chest Chest Chest Chest Chest of Chest Chest Chest Chest Chest Chest (1) Chest Chest Chest Chest Chest Chest Chest (1) Chest Chest Chest Chest Chest Chest Chest (1) Chest Chest Chest Chest Chest Chest Chest Chest (1) Chest Chest Chest Chest Chest Chest Chest Chest Chest (1) Chest C
张楚珩: 【强化学习算法 8】ARS	ndom search provides a competitive to reinforcement learning sixMass Austa Our Desposis In- tense of Execute Exposition of Congress' Convey of Colleges, Books March 20, 2019
张楚珩: 【强化学习算法 9】ES	Evolution Strategies as a Alternative to Reinforcement Let Jundan Bis Nichon Sopon State Br
张楚珩: 【强化学习算法 10】SQL ② zhuanlan.zhihu.com	ent Learning with Deep Energy-Based waspi ¹¹ Brown Img ¹² Powe Abbel ¹¹¹ Supp Le
张楚珩: 【强化学习算法 11】SAC ② zhuanlan.zhihu.com	Soft Actor-Critic: y Maximum Entropy Deep Reinfo Learning with a Stochastic Actor menuga' Aurida Thosa' Peter Alberd' Sergey
张楚珩:【强化学习算法 21】TD3	

二、Model-based RL

Model-based的优势主要在Data Efficient上面,主要探讨model如何建模、建模之后如何学习或者规划。



三、Meta RL

主要讨论如何从一组任务里面学习到prior,使得拥有meta的算法能够快速在新的环境里面适应和学习。与之相关的话题有Few-shot Learning、Transfer Learning。



四、Hierarchical RL

主要是想解决动作空间、观察空间超复杂,并且奖励稀疏的复杂任务。原本任务是从北京到广州,HRL就是让一层策略发出"到北京西站-上火车-等待-下火车"的指令,下一层策略根据上一层发出"上火车"的指令,发出更为具体的"抬腿-迈腿"这样的指令。相关的问题,如何定义sub-goal?如何让上一层学习到输出合适的sub-goal?如何能是合适的reward让下一层学习到sub-goal?



Seasoning and Learning Lab, School of Computer science



另外附上我自己实现的部分RL算法,来帮助大家学习。



其优势在于

- 1. 每个算法装在一个文件里面,没有复杂的依赖,直接就能跑;
- 2. 也没有过多的wrapper, 直接是最简单的实现方法, 目的是理解算法;
- 3. 效果达不到原文的水平,但是都确保能收敛;

听说百度投资了知乎,惶恐中。贴出一个可以把整个专栏下载为 PDF 的代码。

```
import urllib.request
import shutil
import json
import time
import os
def download_articles(p_numbers, p_titles, prefix, output_dir):
    for p, t in zip(p_numbers, p_titles):
        print('processing {}-{}'.format(p, t))
       ret = os.system('wget -P {} -E -H -k -p https://zhuanlan.zhihu.com/p/{}'.form
       if ret != 0:
            raise ValueError('wget error! p={}'.format(p))
        html_file = os.path.join(prefix, 'zhuanlan.zhihu.com', 'p', '{}.html'.format(
       with open(html_file, 'r+') as f:
            html_string = f.read()
            # wkhtmltopdf ignores images wrapped by noscript - weird
            html_string = html_string.replace('<noscript>', '')
            html_string = html_string.replace('</noscript>', '')
            f.seek(0)
            f.write(html_string)
            f.truncate()
        output_file = os.path.join(output_dir, '{}.pdf'.format(p))
        ret = os.system('wkhtmltopdf {} {}'.format(html_file, output_file))
        if ret != 0:
            raise ValueError('wkhtmltopdf error! p={}'.format(p))
def get_p_numbers(zhuanlan):
   p_numbers = []
   p_titles = []
   offset = 0
   while True:
       url = 'https://zhuanlan.zhihu.com/api/columns/{}/articles?include=data&limit=
       html_string = urllib.request.urlopen(url).read()
        content = json.loads(html_string)
       p_numbers.extend([item['id'] for item in content['data']])
       p_titles.extend([item['title'] for item in content['data']])
       if len(content['data']) < 100:</pre>
            break
        else:
            offset += 100
    return p_numbers, p_titles
if __name__ == '__main__':
    zhuanlan = 'reinforcementlearning'
    prefix = 'working_dir'
   output_dir = 'output_dir'
    shutil.rmtree(prefix)
   os.makedirs(prefix, exist_ok=True)
   os.makedirs(output_dir, exist_ok=True)
    p_numbers, p_titles = get_p_numbers(zhuanlan)
    download_articles(p_numbers, p_titles, prefix, output_dir)
```



文章被以下专栏收录



进入专栏