

▲ 一文了解强化学习

3

深度学习 (<http://www.csdn.net/tag/深度学习/news>)人工智能 (<http://www.csdn.net/tag/人工智能/news>)机器学习 (<http://www.csdn.net/tag/机器学习/news>)神经网络 (<http://www.csdn.net/tag/神经网络/news>)

阅读 2183



作者：不会停的蜗牛 CSDN AI专栏作家

强化学习非常重要，原因不只在它可以用来玩游戏，更在于其在制造业、库存、电商、广告、推荐、金融、医疗等与我们生活息息相关的领域也有很好的应用。

本文结构：

1. 定义
2. 和监督式学习, 非监督式学习的区别
3. 主要算法和类别
4. 应用举例

1. 定义

强化学习是机器学习的一个重要分支，是多学科多领域交叉的一个产物，它的本质是解决 **decision making 问题**，即自动进行决策，并且可以做连续决策。

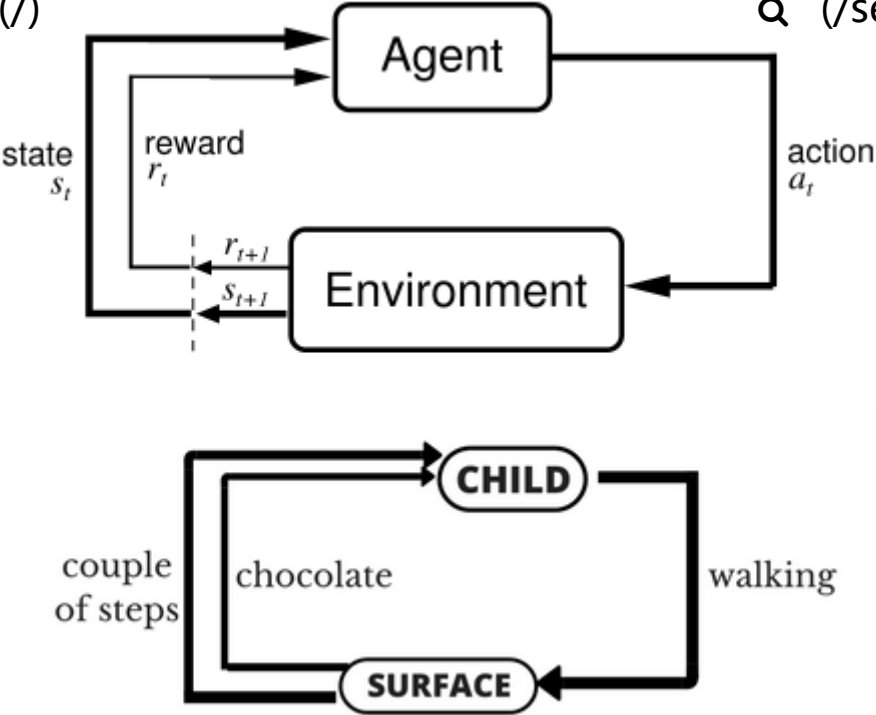
它主要包含四个元素，**agent**，**环境状态**，**行动**，**奖励**，强化学习的目标就是获得最多的累计奖励。

让我们以小孩学习走路来做个形象的例子：

小孩想要走路，但在这之前，他需要先站起来，站起来之后还要保持平衡，接下来还要先迈出一条腿，是左腿还是右腿，迈出一步后还要迈出下一步。

小孩就是 **agent**，他试图通过采取行动（即行走）来操纵环境（行走的表面），并且从一个状态转变到另一个状态（即他走的每一步），当他完成任务的子任务（即走了几步）时，孩子得到奖励（给巧克力吃），并且当他不能走路时，就不会给巧克力。





2. 强化学习与监督式、非监督式学习的区别

在机器学习中，我们比较熟知的是监督式学习，非监督学习，此外还有一个大类就是强化学习：

Types of Machine Learning

🔍 请输入标题

🔍 请输入链接地址

Machine

👤 请输入推荐理由

Task driven
(Regression /
Classification)

Data driven
(Clustering)

Algorithm learns to
react to an
environment

Analytics Vidhya

Learn Everything About Analytics

强化学习和监督式学习的区别：



监督式学习就好比你在学习的时候，有一个导师在旁边指点，他知道怎么是对的怎么是错的，但在很多实际问题中，例如 chess，go，这种有成千上万种组合方式的情况，不可能有一个导师知道所有可能的结果。

而这时，强化学习会在没有任何标签的情况下，通过先尝试做出一些行为得到一个结果，通过这个结果是对还是错的反馈，调整之前的行为，就这样不断的调整，算法能够学习到在什么样的情况下选择什么样的行为可以得到最好的结果。

就好比有一只还没有训练好的小狗，每当它把屋子弄乱后，就减少美味食物的数量（惩罚），每次表现不错时，就加倍美味食物的数量（奖励），那么小狗最终会学到一个知识，就是把客厅弄乱是不好的行为。

两种学习方式都会学习出输入到输出的一个映射，监督式学习出的是之间的关系，可以告诉算法什么样的输入对应着什么样的输出，强化学习出的是给机器的反馈 reward function，即用来判断这个行为是好是坏。

发布到

主题 ▾

发布

评论

另外强化学习的结果反馈有延时，有时候可能需要走了很多步以后才知道以前的某一步的选择是好还是坏，而监督学习做了比较坏的选择会立刻反馈给算法。

而且强化学习面对的输入总是在变化，每当算法做出一个行为，它影响下一次决策的输入，而监督学习的输入是独立同分布的。

通过探索与开发，一个 agent 可以在探索 and 开发（exploration and exploitation）之间做权衡，并且选择一个最大的回报。

请输入推荐理由

非监督式不是学习输入到输出的映射，而是模式。例如在向用户推荐新闻文章的任务中，非监督式会找到用户先前已经阅读过类似的文章并向他们推荐其一，而强化学习将通过向用户先推荐少量的新闻，并不断获得来自用户的反馈，最后构建用户可能会喜欢的文章的“知识图”。

3. 主要算法和分类

从强化学习的几个元素的角度划分的话，方法主要有下面几类：



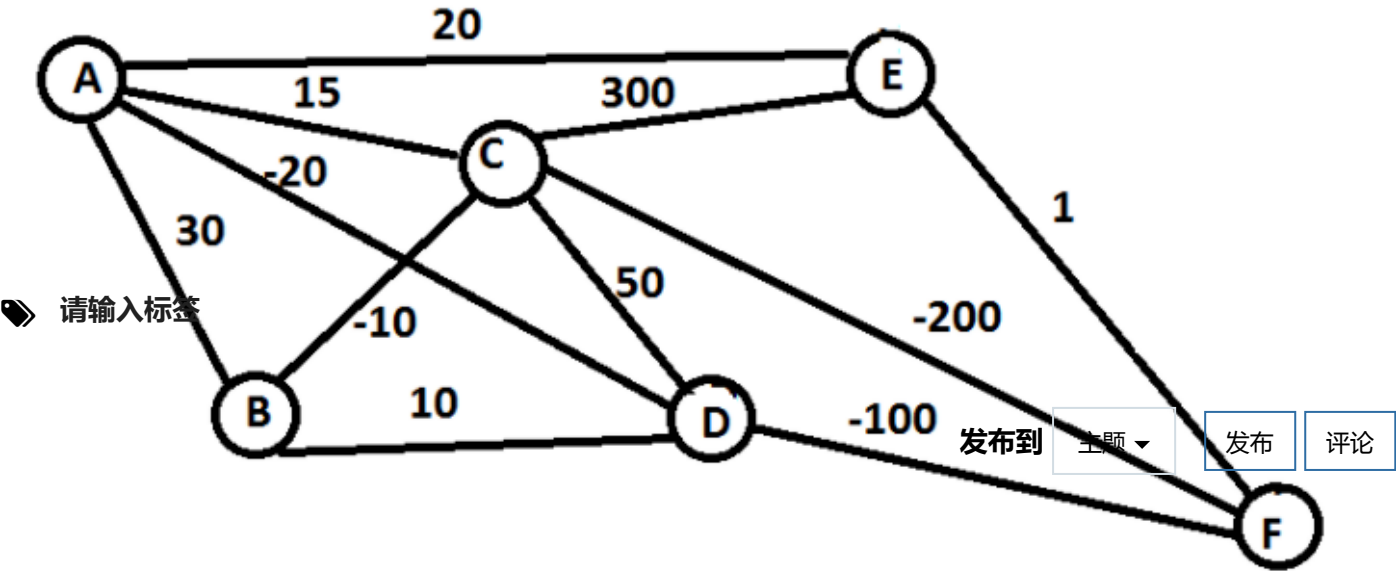
- Policy based, 关注点是找到最优策略。
- Value based, 关注点是找到最优奖励总和。

● action-based) 关注点是每一步的最优行动。

Q (/search) 8 1

我们可以用一个最熟知的旅行商例子来看，

我们要从 A 走到 F，每两点之间表示这条路的成本，我们要选择路径让成本越低越好：



那么几大元素分别是：

• 请输入标题 就是节点 {A, B, C, D, E, F}

• 请输入链接地址 action，就是从一点走到下一点 {A -> B, C -> D, etc}

• reward function 就是边上的 cost

请输入推荐理由

此外还可以从不同角度使分类更细一些：

如下图所示的四种分类方式，分别对应着相应的主要算法：



极客头条

| | Model-free | Model-based | Policy based | Value based | Monte-carlo update | Temporal-difference update | On-policy | Off-policy |
|-----------------------|------------|-------------|--------------|-------------|--------------------|----------------------------|-----------|------------|
| Qlearning | ✓ | ✓ | | ✓ | | ✓ | | ✓ |
| Sarsa | ✓ | ✓ | | ✓ | | ✓ | ✓ | |
| Policy Gradients | ✓ | ✓ | ✓ | | ✓ | | | |
| actor-critic | | | ✓ | ✓ | | | | |
| 升级版的 policy gradients | | | | | | ✓ | | |
| Monte-carlo learning | | | | | ✓ | | | |
| sarsa lambda | | | | | | | ✓ | |
| Deep-Q-Network | | | | | | | | ✓ |

- **Model-free**：不尝试去理解环境, 环境给什么就是什么，一步一步等待真实世界的反馈, 再根据反馈采取下一步行动。
- **Model-based**：先理解真实世界是怎样的, 并建立一个模型来模拟现实世界的反馈，通过想象来预判断接下来将要发生的所有情况，然后选择这些想象情况中最好的那种，并依据这种情况来采取下一步的策略。它比 Model-free 多出了一个虚拟环境，还有想象力。
- **Policy based**：通过感官分析所处的环境, 直接输出下一步要采取的各种动作的概率, 然后根据概率采取行动。
- **Value based**：输出的是所有动作的价值, 根据最高价值来选动作，这类方法不能选取连续的动作。
- **Monte-carlo update**：游戏开始后, 要等待游戏结束, 然后再总结这一回合中的所有转折点, 再更新行为准则。
- **Temporai-difference update**：在游戏进行中每一步都在更新, 不用等待游戏的结束, 这样

请输入标题

请输入链接地址

请输入推荐理由

主要算法有下面几种，今天先只是简述：

1. Sarsa



≡ 极客头条 (A, a) arbitrarily

Q (/search) 🔍 📄

Repeat (for each episode):

Initialize S

Choose A from S using policy derived from Q (e.g., ϵ -greedy)

Repeat (for each step of episode):

Take action A , observe R, S'

Choose A' from S' using policy derived from Q (e.g., ϵ -greedy)

$Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma Q(S', A') - Q(S, A)]$

$S \leftarrow S'; A \leftarrow A';$

until S is terminal

Q 为动作效用函数 (action-utility function)，用于评价在特定状态下采取某个动作的优劣，可以将之理解为智能体 (Agent) 的大脑。

SARSA 利用马尔科夫性质，只利用了下一步信息，让系统按照策略指定进行探索，在探索每一步都进行状态价值的更新，更新公式如下所示：

$$q(s, a) = q(s, a) + \alpha(r + \gamma q(s', a') - q(s, a))$$

s 为当前状态， a 是当前采取的动作， s' 为下一步状态， a' 是下一个状态采取的动作， r 是系统获得的奖励， α 是学习率， γ 是衰减因子。

🔍 请输入标题

🔍 请输入链接地址

2. Q learning

👉 请输入推荐理由

Take action A , observe R, S'

$Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a Q(S', a) - Q(S, A)]$

$S \leftarrow S';$

until S is terminal



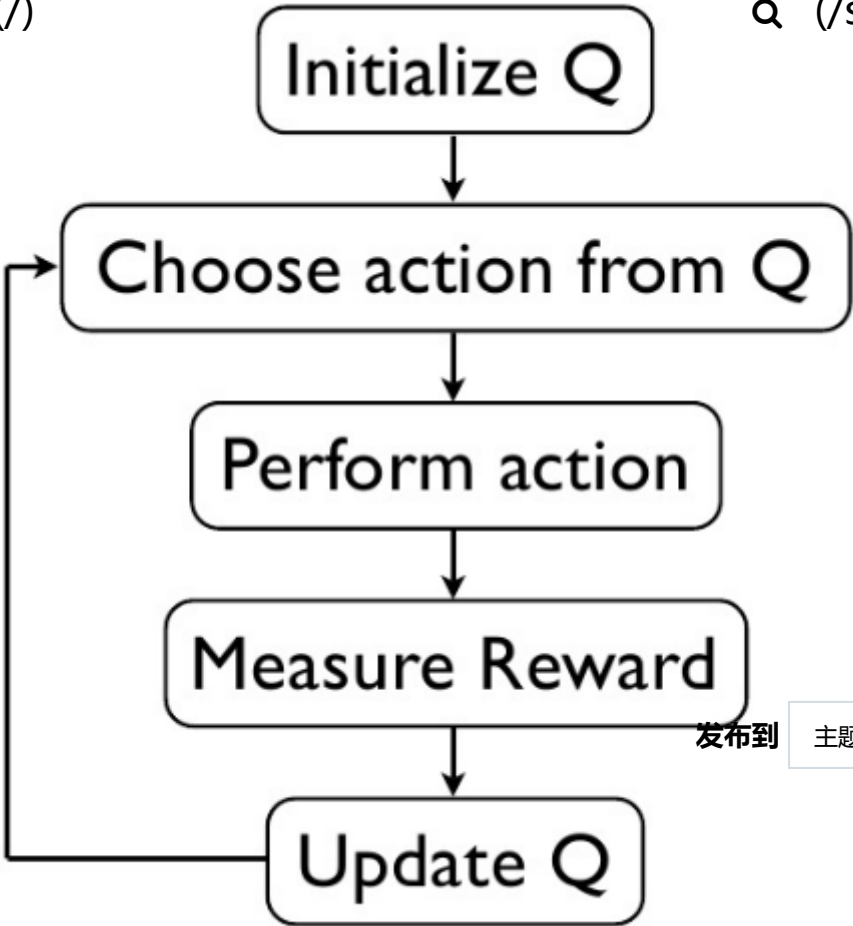
请输入标签

发布到

主题

发布

评论



Q Learning 的算法框架和 SARSA 类似, 也是让系统按照策略指引进行探索, 在探索每一步都进行状态值的更新。关键在于 Q Learning 和 SARSA 的更新公式不一样, Q Learning 的更新公式如

请输入链接地址

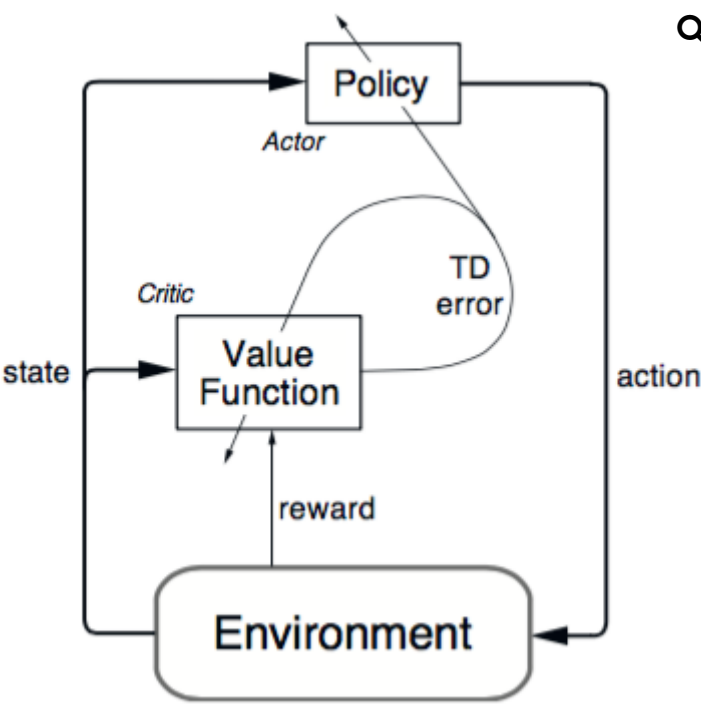
请输入推荐理由

$gt=rt+\gamma rt+1+\dots$ 等于 $q(st,a)$, 从而求解策略梯度优化问题。

4. Actor-Critic



请输入标签



发布到

主题

发布

评论

算法分为两个部分：Actor 和 Critic。Actor 更新策略，Critic 更新价值。Critic 就可以用之前介绍的 SARSA 或者 Q Learning 算法。

5. Monte-carlo learning

用当前策略探索产生一个完整的状态-动作-奖励序列:

$s_0, a_1, r_1, \dots, s_k, a_k, r_k \sim \pi$

请输入链接地址

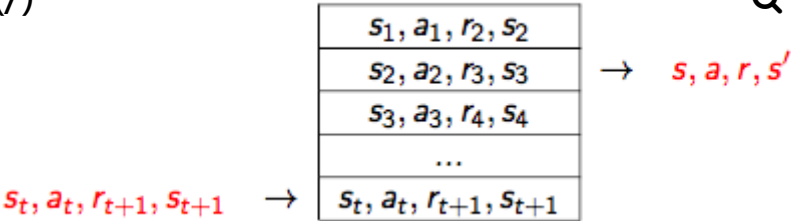
在序列第一次碰到或者每次碰到一个状态 s 时，计算其衰减奖励:

请输入推荐理由

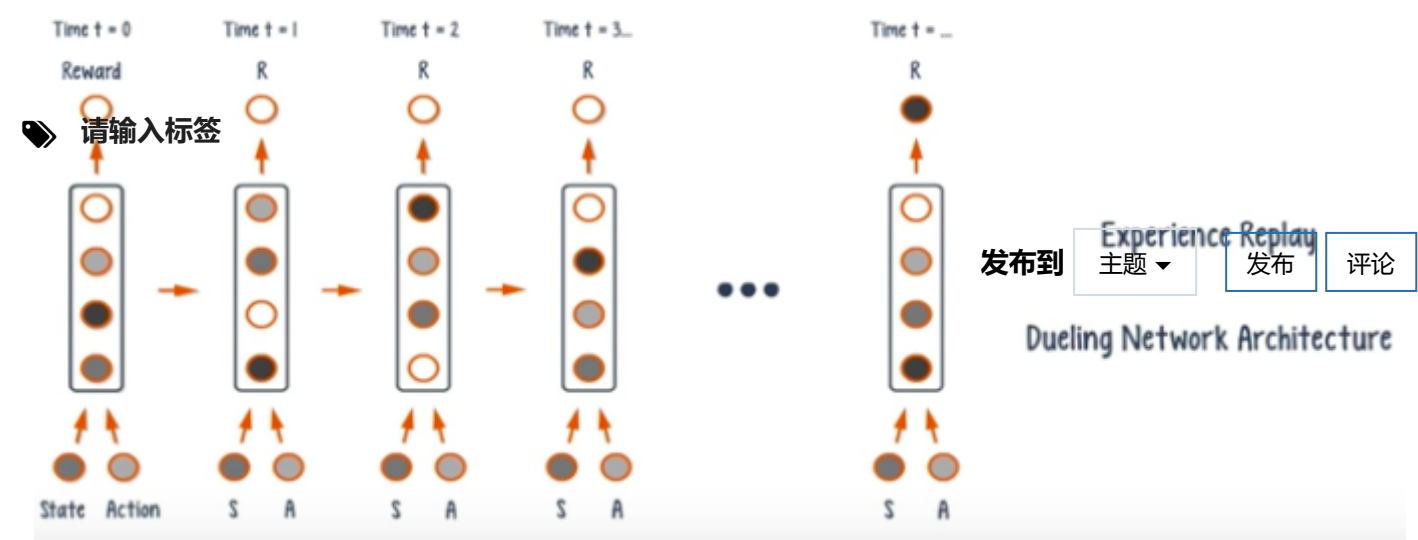
$$N(s) = N(s) + 1$$
$$v(s) = \frac{S(s)}{N(s)}$$

6. Deep-Q-Network

DQN 算法的主要做法是 Experience Replay，将系统探索环境得到的数据储存起来，然后随机采样样本更新深度神经网络的参数。它也是在每个 action 和 environment state 下达到最大回报，不同的是加了一些改进，加入了经验回放和决斗网络架构。



Deep Q Net



请输入标题

请输入内容

请输入链接地址

强化学习有很多应用，除了无人驾驶、AlphaGo、玩游戏之外，还有下面这些工程中的应用的例子。

请输入推荐理由

 请输入标签

在库存管理中，因为库存量大，库存需求波动较大，库存补货速度缓慢等阻碍使得管理是个比较难的问题。通过建立强化学习算法来减少库存周转时间，提高空间利用率。

➡ 请输入推荐理由

5. ECommerce Personalization

6. Ad Serving

例如算法 LinUCB（属于强化学习算法 bandit 的一种算法），会尝试投放更广范围的广告，尽管过去还没有被浏览很多，能够更好地估计真实的点击率。

再如双12推荐场景中，阿里巴巴使用了深度强化学习与自适应在线学习，通过持续机器学习和模型优化建立决策引擎，对海量用户行为以及百亿级商品特征进行实时分析，帮助每一个用户迅速发现宝贝，提高人和商品的配对效率。还有，利用强化学习将手机用户点击率提升了 10-20%。

7. Financial Investment Decisions

例如这家公司 Pit.ai，应用强化学习来评价交易策略，可以帮助用户建立交易策略，并帮助他们实现其投资目标。

8. Medical Industry

动态治疗方案（DTR）是医学研究的一个主题，是为了给患者找到有效的治疗方法。例如癌症这种需要长期治疗的治疗，强化学习算法可以将患者的各种临床指标作为输入来制定治疗策略。

上面简单地介绍了强化学习的概念，区别，主要算法，下面是一些学习资源，供参考：

发布

评论

1. Udacity课程1：Machine Learning: Reinforcement Learning，
课程2：Reinforcement Learning

2. 经典教科书：Sutton & Barto Textbook: Reinforcement Learning: An Introduction 被引用 2 万 多 次 http://people.inf.elte.hu/lorincz/Files/RL_2006/SuttonBook.pdf
(http://people.inf.elte.hu/lorincz/Files/RL_2006/SuttonBook.pdf)

3. Berkeley开发的经典的入门课程作业 - 编程玩“吃豆人”游戏：Berkeley Pac-Man Project (CS188 Intro to AI)

4. Stanford开发的入门课程作业 - 简化版无人驾驶：Car Tracking (CS221: AI: Principles)

请输入推荐理由

相关文章：

TensorFlow-11-策略网络：用 Tensorflow 创建一个基于策略网络的 Agent 来解决 CartPole 问题。

<http://www.jianshu.com/p/14625de78455> (<http://www.jianshu.com/p/14625de78455>)

强化学习是什么：简单图解了 DQN

<http://www.jianshu.com/p/2100cc577a46> (<http://www.jianshu.com/p/2100cc577a46>)

参考 极客头条 (/)

Q (/search) 🔍 📄

<https://www.marutitech.com/businesses-reinforcement-learning/>

(<https://www.marutitech.com/businesses-reinforcement-learning/>)

<https://www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation/> (<https://www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation/>)

<https://morvanzhou.github.io/tutorials/machine-learning/ML-intro/4-02-RL-methods/> (<https://morvanzhou.github.io/tutorials/machine-learning/ML-intro/4-02-RL-methods/>)

<https://www.zhihu.com/question/41775291> (<https://www.zhihu.com/question/41775291>)

<http://www.algorithmdog.com/reinforcement-learning-model-free-learning>

(<http://www.algorithmdog.com/reinforcement-learning-model-free-learning>)

🔖 请输入标签

2017中国人工智能大会 (CCAI 2017) | 7月22日-23日 杭州

本届CCAI由中国人工智能学会、蚂蚁金服主办，由CSDN承办，最专业的年度技术盛宴：

发布到 主题 ▾

发布

评论

- 40位以上实力讲师
- 8场权威专家主题报告
- 4场开放式专题研讨会
- 超过100家媒体报道
- 超过2000位技术精英和专业人士参会

👤 与大牛面对面，到官网报名：<http://ccai.ccai.cn/> (<http://ccai.ccai.cn/>)

🔖 请输入标题

🔗 请输入链接地址

👉 请输入推荐理由

扫码加我进群

CSDN AI 干货分享交流

♦ 名家大师、千余位业界同行 ♦ CSDN福利、资料秒送达 ♦ 线上线下活动优先报名

加群请注明：公司+职位+姓名




(http://geek.csdn.net/user/publishlist/qunnie_yi)

评论

已有2条评论

最新




meiceatcsdn (http://geek.csdn.net/user/publishlist/meiceatcsdn)

2小时前

感谢分享！我第一次了解强化学习是上周在图书馆看了一本书，是介绍围棋人机大战详细内幕的。这本书里面就重点提到了AlphaGo的学习，很重要的一部分就是深度学习。因为围棋和跳棋、象棋、国际象棋、牌类游戏的本质区别就是，可能性实在太太，大到比世界原子数量还要多的可能。所以，没发穷尽，没发输入标签。其中就提到了这个。感谢！

0

回复 投诉



CSDN无艺 (http://geek.csdn.net/user/publishlist/qunnie_yi)

1小时前

感谢您支持

发布到

主题

发布

评论

0

回复 投诉

请输入标题

请输入链接地址

请输入推荐理由