






# 深度强化学习:综述

Xu Wang,  Sen Wang, Xingxing Liang,  Dawei Zhao, Jincai Huang, Xin Xu , IEEE 高级会员  
戴斌 , and Qiguang Miao , IEEE 高级会员

**摘要:**深度强化学习 (DRL)将深度学习的特征表示能力与强化学习的决策能力相结合,从而实现强大的端到端学习控制能力。

在过去的十年中,DRL在许多需要感知高维输入并做出最优或近优决策的任务上取得了实质性的进展。然而,DRL的理论和应用中仍然存在许多具有挑战性的问题,特别是在样本有限、奖励稀疏和多智能体的学习控制任务中。研究人员提出了各种解决方案和新理论来解决这些问题并推动了DRL的发展。此外,深度学习刺激了强化学习的许多子领域的进一步发展,例如分层强化学习、多智能体强化学习和模仿学习。本文全面概述了DRL的基础理论、关键算法和主要研究领域。除了基于价值和基于策略的DRL算法之外,还总结了基于最大熵的DRL的进展。并分析和讨论了DRL的未来研究主题。

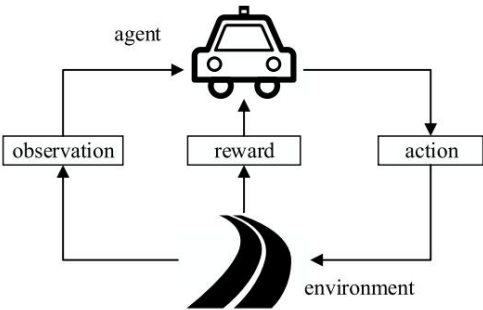


图 1. 代理与环境的交互:在每个时间步骤中,代理观察环境后,根据其策略选择一个动作,执行动作后,环境向代理发出奖励信号并转换到新状态。

**索引词** 深度学习、深度强化学习 (DRL)、模仿学习、最大熵深度强化学习 (RL)、策略梯度、价值函数。

## 一、引言

强化学习 (RL)是机器学习的一个子加强领域。目标是让代理学习如何根据环境状态采取行动,从而

稿件收到日期为 2020 年 6 月 10 日;修订日期为 2020 年 12 月 14 日、2021 年 6 月 15 日和 2021 年 10 月 17 日;接受日期为 2022 年 8 月 29 日。出版日期为 2022 年 9 月 28 日;当前版本日期为 2024 年 4 月 5 日。这项工作部分由国家重点研发计划资助,资助编号为 2018YFC0807500;部分由国家自然科学基金资助,资助编号为 61772396、61772392、61902296 和 61825305;部分由西安大数据与智能视觉重点实验室资助,资助编号为 201805053ZD4CG37;部分由陕西省国家自然科学基金 (2020JQ-330 和 2020JM-195)资助;部分由中国博士后科学基金 (2019M663640) 资助;部分由广西可信软件重点实验室 (KX202061)资助。(通讯作者:苗启光。)

王旭、王森和苗启光均就职于西安电子科技大学西安大数据与智能视觉重点实验室。中国西安 710000 (电子邮件:xuwangxw@foxmail.com;xidian\_wangsen@126.com;qgmiao@xidian.edu.cn)。

梁星星和黄金才就职于国防科技大学信息工程重点实验室,长沙 410000,湖南省 (e-mail:doublestar\_l@163.com;huangjincal@nudt.edu.cn)。

赵大伟和戴斌就职于国防科技创新研究院,北京 100071,中国 (电子邮件:zhaodawei12@nudt.edu.cn;bindai.cs@gmail.com)。

徐鑫,国防科技大学情报科学学院,长沙 410000 (电子邮件:xinxu@nudt.edu.cn)。

数字对象标识符 10.1109/TNNLS.2022.3207346

2162-237X © 2022 IEEE。允许个人使用,但转载/重新分发需要获得 IEEE 许可。  
请参阅<https://www.ieee.org/publications/rights/index.html>了解更多信息。

授权许可使用仅限于:上海交通大学。于 2024 年 10 月 27 日 09:28:42 UTC 从 IEEE Xplore 下载。有限制。

最大化预期的长期回报,其中学习问题通常可以建模为马尔可夫决策问题 (MDP)[1]。图 1 显示了代理与环境之间的交互反馈回路。早期的 RL 研究主要集中在表格和基于近似的算法 [2],[3],[4],[5],[6]。由于缺乏表示能力,传统的 RL 算法只能解决具有低维状态和动作空间的任务。然而,更复杂、更接近现实世界的任务通常具有更高维度的状态空间和连续的动作空间,从而限制了 RL 的应用 [7],[8]。

图像分类[9]、自然语言处理 (NLP)等领域的成果表明,深度学习具有强大的表示能力,可以从高维抽象输入 (如图像)中提取多层次的特征 [12]。此外,深度神经网络已被证明是一般的函数逼近器[13]、[14]、[15],可用于逼近具有高维输入的复杂任务中的值函数和策略。因此,深度强化学习 (DRL)近年来受到了广泛关注。

自从深度 Q 网络 (DQN)在游戏中成功应用 [8] 以来,越来越多的深度学习技术和算法与 RL 相结合,不仅用于解决困难的传统 RL 任务,还激发了新的研究领域 (元 DRL、迁移 DRL等)。DRL不仅在理论上取得了一些重要进展,在应用方面也取得了一些重要进展,例如机器人控制 [16]、[17]、[18]、[19]、游戏 [20]、[21]、[22]、[23]、[24]、NLP [25]、[26]、[27]、自动驾驶 [28]、[29]、[30]、推荐系统 [31]、[32] 和计算机视觉 [33]、[34]、[35]。

此外,许多研究团队已经建立了不同的开源 DRL 算法库,包括基线 [36]、

coach [37]、tf\_agents [38]、Tianshou [39]。这些库提供了性能分析工具,可以轻松集成到更复杂的DRL代理中,为DRL社区的发展做出了贡献。

先前的综述论文从各个方面总结了 DRL 的发展 [40],[41],[42],[43]。与它们相比,本文的贡献包括以下三个方面。首先,根据不同的学习目标,我们将 DRL 算法分为三类,并清楚地展示了这三类 DRL 方法之间的关系:基于价值的算法、基于策略的算法和基于最大熵的算法。其次,详细分析了在不同代码库中实现的常用 DRL 算法。最后,分析和讨论了针对解决传统 RL 和 DRL 的不同挑战的进一步研究课题。以下各节从 RL 和深度学习的背景开始。第三节和第四节分别介绍基于价值和基于策略的算法。第五节介绍了最大熵 DRL 算法。第六节讨论了最近引起关注的几个方向及其主要进展。

二、背景

本节涵盖了 RL 的基础知识,包括解决 RL 任务的基本框架和基于 MDP、动态规划、蒙特卡洛 (MC) 和时间差分 (TD) 方法的算法。本节还包括与 RL 结合使用的深度学习的一些特性。

强化学习

1) 马尔可夫决策过程: MDP [1],[44] 是解决序列决策问题的经典框架。

MDP 基于以下假设。1. 环境是马尔可夫的,这意味着下一个时间步的状态仅由当前状态决定,与先前的状态无关。2. 环境是完全可观察的。也就是说,代理可以随时观察所有环境信息。这个假设在某些情况下是不合适的,因此提出了许多 MDP 变体,例如部分可观察的 MDP [45]。

MDP 可以表示为五元组 (S, A, ρ, f, γ), 其中如下所述。

- 1) S 是所有可观测状态的集合。
- 2) A 是一组动作,动作可以是离散的,也可以是连续的。如果状态发生变化,agent 会按照一定的概率分布去采样下一个动作,这个概率分布称为策略: π: S → A。交互得到的状态和动作轨迹表示为 τ: (s<sub>0</sub>, a<sub>0</sub>, s<sub>1</sub>, a<sub>1</sub>, ..., a<sub>t-1</sub>, s<sub>t</sub>)。3) p: S × A → R 是即时标量奖励。4) f: S × A × A → [0, 1] 称为状态转换函数。f (s, a, s) = P (s | s, a) 是执行动作 a 后状态 s 转换到 s 的概率。5) γ ∈ [0, 1] 是折扣因子,其目的是

MDP 是强化学习任务的一种流行数学形式 [3]。MDP 下的强化学习任务的目标是找到能够最大化累积奖励的轨迹,累积奖励表示为从时间 t 开始的所有折扣奖励的总和

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k R_{t+K+1} \quad (1)$$

其中 R<sub>i</sub> 是时间步骤 i 的即时奖励, G<sub>t</sub> 称为回报。

2) 贝尔曼方程: 由于状态转移概率和策略的存在, 计算一个状态的回报并不容易。我们需要计算从这个状态出发的所有轨迹的回报, 并计算它们的期望。状态 s 的预期回报定义为状态值函数 V

$$\begin{aligned} V(s) &= E_{s,a} \tau \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \\ &= \sum_{a \in A} \pi(a|s) p(s_{t+1}|s, a) \sum_{a_t \in A} \tau \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \\ &= E \tau R_{t+1} + \gamma V(s_{t+1}) | s_t = s \end{aligned} \quad (2)$$

其中 τ 表示交互轨迹, π(a<sub>t</sub>|s<sub>t</sub>) 为策略, p(s<sub>t+1</sub>|s<sub>t</sub>, a<sub>t</sub>) 为状态转移概率。如 (2) 所示, 状态值函数可以表示为递归形式。因此, 寻找可以最大化累积奖励的轨迹的目标转变为寻找可以最大化 V (s<sub>t</sub>) 的策略。类似地, 在状态 s 下执行 a 的预期回报定义为状态动作值函数 Q

$$Q(s, a) = E \tau R_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) | s_t = s, a_t = a. \quad (3)$$

方程 (2) 和 (3) 称为贝尔曼方程 [4], [46]。贝尔曼方程根据所有轨迹的发生概率计算其累积奖励的加权平均值, 是解决 RL 任务的基本方程。

最优值函数是指在所有状态下具有最大值的函数: V\* = max<sub>π</sub> v<sub>π</sub> (s)。类似地, 最优动作值函数为: Q\* (s, a) = max<sub>Q</sub> Q(s, a)。对于所有 MDP, 总有一个或多个最优策略, 并且所有最优策略的值函数都是相同的。因此, 可以通过最大化最优 Q 函数来找到最优策略

$$\pi^*(a|s) = \begin{cases} 1, & \text{如果 } a = \arg\max_Q Q(s, a) \\ 0, & \text{否则} \end{cases} \quad (4)$$

3) On-Policy 与 Off-Policy 方法: On-policy 与 Off-policy 指的是两种不同的训练方式, 其主要区别 [3] 在于行为策略与目标策略是否相同。行为策略是用来与环境交互以生成训练数据的策略。目标策略是我们希望智能体学习到的策略。

在策略方法中, 目标策略直接用于生成下一轮策略优化的数据。下一轮策略优化完成后, 数据将被丢弃。因此, 行为策略与目标策略相同。离策略方法将生成的样本存储在

减少未来奖励对现在的影响。

通过在与环境交互时将行为策略放入缓冲区中来训练,在训练过程中使用缓冲区中的样本来更新目标策略。训练数据可能来自旧策略,因此行为策略与目标策略并不相同。on-policy方法的优点是可以直接优化策略。而off-policy方法具有更高的数据效率。

4)动态规划方法 :当环境模型完全已知时,可以用动态规划方法求解贝尔曼方程 [46]。动态规划是一种通过将原问题分解为相对简单的子问题来解决复杂问题的方法。基于动态规划的算法主要有策略迭代和值迭代[3]。

策略迭代的思想是迭代执行策略评估和策略改进步骤,直到策略收敛到最优。在策略评估步骤中,使用策略计算当前值。在策略改进步骤中,使用前一个策略评估步骤的值生成更优的策略。值迭代的过程是 :在每个状态下,依次执行每个动作并计算 Q值。将最优的Q值作为当前状态的值。当每个状态的最优值不再变化时,迭代结束。

策略迭代中的策略评估步骤需要价值函数收敛,而在价值迭代中,最优值和最优策略同时收敛。

因此,一旦价值函数达到最优值,策略也会收敛到其最优值,从而简化迭代步骤。

5)蒙特卡洛方法 :大多数情况下,一些环境属性很难获得,也就是说通常无法对环境进行完全建模。这种情况下,可以采用蒙特卡洛方法来评估价值。蒙特卡洛方法评估某一状态的价值的步骤如下 :首先,从该状态进行多次模拟,得到多条轨迹。然后计算每条轨迹的累积奖励。最后通过 $V(s_t) = V(s_t) + \alpha(G_t - V(s_t))$ 计算该状态的价值。由于更新估计值的奖励来自真实的交互轨迹,因此估计值是无偏的。

6)时间差分法 : TD 方法 [47] 与 MC 类似,直接从经验中学习,不需要知道环境的动态模型。不同之处在于 TD 方法只模拟当前状态下的一步,而不是达到终止状态。最简单的 TD 方法 TD(0) [48] 是

$$V(s_t) = V(s_t) + \alpha(R_{t+1} + \gamma V(s_{t+1}) - V(s_t))$$

其中 $R_{t+1} + \gamma V(s_{t+1}) - V(s_t)$ 称为 TD 误差。由上式可知,TD 方法可以从不完全序列中进行学习,从而避免情节更新,提高收敛速度。TD(0) 的更新在一定程度上基于已有的估计,与 DP 类似,因此也是一种 bootstrap 方法 [3]。

使用 TD 误差来更新价值函数的代理可以使用在线策略方法进行训练,例如 SARSA [3],[49],

或者离策略方法,比如Q 学习 [50]。Q学习是早期强化学习算法的突破 [3]。Q学习的更新方程为

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha R_{t+1} + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)$$

(6)

其中  $\alpha \in (0, 1]$  表示步长。在计算当前Q值时, Q 学习不遵循交互序列,而是在下一个时间步选择具有最大Q值的动作。这种方法会导致对Q值的估计过高,这个问题将在本文后面讨论。

7)策略梯度定理 :如上文所述,策略  $\pi: S \rightarrow A$ 是从状态到动作的映射,表示在状态s下选择动作a的概率,通常定义为所有动作的概率分布或概率密度函数

$$\pi_{\beta}(a|s) = P(a|s, \beta)$$

(7)

其中 $\beta$ 是策略的参数。

上述动态规划、MC、TD方法都需要先计算最优Q值,再利用 (4)式得到最优策略,这类方法被称为基于价值的方法。另一类能够直接优化策略的方法被称为策略梯度方法。

将预期回报定义为策略 $\pi_{\beta}$ 的性能衡量标准

$$J(\beta) = V\pi_{\beta}(s) = E\pi_{\beta}(s) \quad Q(s, a)\pi_{\beta}(a|s)$$

(8)

然后通过策略梯度定理对  $\beta$  进行微分,得到策略优化方程 [51]

$$\nabla_{\beta} J(\beta) = E\pi_{\beta}(s) \quad Q(s, a)\nabla_{\beta}\pi_{\beta}(a|s)$$

(9)

$$\beta = \beta + \alpha \nabla_{\beta} J$$

(10)

其中  $\alpha$  是步长。

强化算法 [52] 是一种传统的策略梯度强化学习算法。它使用 MC 方法从采样轨迹中获得的估计累积回报来更新策略,因为样本梯度的预期值是实际梯度的无偏估计。

其最常用的变体是带有基线的形式,其目的是减少估计梯度时产生的方差

$$\nabla_{\beta} J(\beta) = E\pi_{\beta}(s)[\nabla_{\beta} \log \pi_{\beta}(a|s)(Q(s, a) - b)]$$

(11)

其中b通常是独立于 a 的学习状态值函数。

当使用基于价值的方法解决具有连续动作空间的任务时,必须首先将动作空间离散化。然而,离散化总是受到维数灾难的影响 [53]。更糟糕的是,如果离散化的步长太大,这会导致控制结果缺乏不可接受的平滑度 [54]。策略梯度方法获得的最优策略是动作的概率分布或概率密度函数,可以是连续的,也可以是离散的,从而避免了上述缺点。

8) Actor-Critic 方法: Actor-Critic [3] 方法通常指同时学习策略和价值函数,其中价值函数用于评估策略。Actor 负责生成策略、选择动作并与环境交互。Actor 通过 (9) 和 (10) 计算出的梯度进行更新。

评论家在每个时间步骤评估参与者策略的价值函数。可以使用不同的指标来评估参与者的策略,例如动作价值函数Q(s, a)、状态价值函数V(s)或优势函数A(s, a)。

B.深度学习

随着卷积神经网络 (CNN)[9]、循环神经网络 (RNN)[57]等深度神经网络技术[55],[56]的不断发展以及计算能力的提升,深度学习的影响力越来越大。深度学习在图像识别、文本处理等领域取得的成果表明,其在处理高维数据时具有很强的拟合能力和表示能力。Krizhevsky等人[10]提出的高精度图像识别网络 AlexNet 掀起了基于深度神经网络的深度学习研究热潮。

2013 年,word2vector[58],[59]被提出,对后续基于深度学习的NLP技术产生了巨大的影响,并被广泛应用于机器翻译[60]、词语表征[61]等领域[62]。生成对抗网络 (GAN)[63]可以让神经网络生成我们想要的数

Bahdanau等人[65] 采用了类似 Attention 的机制,在机器翻译任务上同时进行翻译和对齐,这是 Attention 在 NLP 上的首次应用。2017 年,谷歌机器翻译团队大量采用了自注意力 (Self-Attention)机制来学习文本表征 [66]。

深度学习的表征能力主要依赖于以神经元[67]为基本单元的多层神经网络。感知器[68]是最早的神经网络原型,为单层神经网络 (无隐藏层),只能完成最简单的线性分类任务,无法解决异或问题[69]。由于神经元数量和层数的增多,多层感知器具有突出的非线性逼近能力。

Hornik等人[13] 证明了多层感知器可以逼近任何非线性函数。

作为深度学习与强化学习的结合,深度强化学习利用神经网络强大的表示能力来处理高维输入,并近似值或策略,以解决状态空间过大、动作空间连续的强化学习问题。以围棋为例,棋盘上每个位置都有三个状态,这创造了一个很大的状态空间,

传统的强化学习无法计算出每个状态的值,而借助深度学习,可以训练出表示棋盘状态的深度神经网络。

然后,基于状态表示,可以用强化学习来学习如何选择放置位置并获得最大的累积奖励。通常,上述两个过程可以看作是:将原始状态映射到特征,将特征映射到动作,分别由深度学习和强化学习完成。由于深度神经网络可以作为黑箱,因此 DRL 可以将这两个过程作为一个整体来考虑。DRL 更侧重于强化学习 [40],并且仍然解决决策问题,因此下一节主要介绍深度学习如何与强化学习相结合。

III.基于价值的深度强化学习方法

基于价值的方法是强化学习方法中的一个重要类别,其重点在于表示价值函数并寻找最优价值函数。上文提到的 Q 学习算法是最经典的基于价值的算法。本节将介绍基于价值的深度强化学习方法 Deep Q Network,并讨论其各种变体和改进。

A.深度 Q 学习

传统的Q学习采用TD方法更新Q值并存储在Q表中,这对于状态空间和动作空间较大的问题来说是不可行的。

Mnih等人提出了深度Q网络 [70],它将深度神经网络与 Q 学习相结合,在大多数 Atari 2600 游戏中超越了人类玩家的水平。算法 1 展示了深度Q网络的训练细节。与常规 Q 学习相比,改进有以下三个方面: 1)深度 Q 网络:在使用 Q 学习作为 RL 算法的前提下,使用一种称为深度Q网络的简单深度神经网络从 Atari 游戏的原始图像中提取低级特征,并在不需要任何其他领域的情况下近似动作值函数。DQN 的隐藏层包括三个卷积层和一个全连接层,如图 2 所示。输出层的结果是每个动作的Q值。DQN 在时间步t的近似值为:

$$y = R + \gamma \max_a Q(s_{t+1}, a; \beta)$$

(12)

其中  $\beta$  代表深度Q网络的参数,通过最小化近似 Q 值和真实Q值之间的 MSE 来更新。

为了增强探索能力,DQN 使用  $\epsilon$ -贪婪方法,以一定的概率执行随机动作,并以剩余的概

2)经验回放:交互得到的轨迹在时间域上有一定的相关性,直接用这些轨迹进行训练可能会导致估计值和期望值的差距越来越大。

提出经验回放,通过将历史转换存储到



算法 1 深度 Q 学习算法 [70]

初始化经验重缓冲区 D;用随机权重  $\beta$  初始化 Q 网络,用权重  $\beta^- = \beta$  初始化目标网络  $Q^-$ ;对于 episode=1, 2...N, 执行

使用预处理的视频游戏图像重置环境和初始状态  $\phi(s_0)$ ;对于  $t=0, 1, 2 \dots T$ ,  
使用  $\phi(st)$  作为 Q 网络的输入并获取每个动作的 Q 值;选择  $at = \arg\max_a Q(st, a)$  处的动作;  
在  $st$  下采取行动,观察奖励  $R$  和新状态  $\phi(st+1)$ ;将  $(\phi(st), at, R, \phi(st+1))$  推入 D;  
从 D 中随机取  $m$  个样本  $(\phi(sj), a_j, R_j, \phi(sj+1))$ ;计算近似值:

$$y_j = R_j + \gamma \max_{a'} Q^-(\phi(sj+1), a')$$

在哪里

$$y = R_j + \gamma \max_a Q^-(\phi(sj+1), a)$$

通过对损失函数执行梯度下降来更新 Q 网络:

$$L(\beta) = \frac{1}{m} \sum_{j=1}^m (y_j - Q(\phi(sj), a_j; \beta))^2$$

每 C 步设置  $\beta^- = \beta$ ;结束

网络负责与环境交互并获取训练样本,在每次训练步骤中,使用来自目标网络的 目标值和来自主网络的估计 Q 值来更新主网络。

每训练完成一定步数,主网络的参数就会同步到目标网络,目标网络在一段时间内保持目标值不变,从而增强 DQN 的稳定性。在目标网络的帮助下,近似值变成

$$y = R + \gamma \max_a Q^-(st+1, a; \beta^-) \tag{13}$$

其中  $Q^-$  代表具有参数  $\beta^-$  的目标网络。

B. 双 DQN

早在1993年,Thrun和Schwartz[71]就发现了Q学习算法中Q值被高估的问题,并将其视为函数逼近能力不足的影响。Hasselt[72]将其归因于环境噪声。Deepmind[73]证明了任何类型的错误都会导致Q值的高估,无论是环境噪声、函数逼近、非平稳性还是其他任何原因。该问题的解决方案是双Q学习,该算法与 DQN 相结合后被称为双深度Q网络 (DDQN)。

DQN 中引入的目标网络表示具有参数  $\beta^-$  的单值函数,负责选择动作和计算 目标值,如 (13) 所示。

DDQN 旨在将这两部分分解为两个不同的价值函数,即两个网络。主网络用于选择最优动作,目 标网络用于估计价值函数。DQN 框架中的目标网络为第二个价值函数提供了天然的候选,从而消 除了引入额外网络的需要。设主网络的参数为  $\beta$ ,求解目标值的公式为

$$y = R_{j+1} + \gamma Q^-(s_{j+1}, \arg\max_a Q(s_{j+1}, a; \beta); \beta^-) \tag{14}$$

两个网络的参数更新方法与DQN类似。

C. 优先体验重播

从经验重放中采样转换进行训练时,每个转换都以相同的概率被选中,也就是说,每个转换都以相同的频率被学习。

但实际上,由于TD误差不同,不同的transition对DQN的反向传播的影响也不同,TD误差越大,对反向传播的影响越大。

此外,缓冲区的大小有限,对学习有帮助的数据可能会在被采样之前被丢弃。Schaul等[74]提出了优先经验回放来解决这些问题。

优先经验回放以每次转换的 TD 误差作为评估优先级的标准。其形式为

图 2. [70] 中的深度 Q 网络架构。深度 Q 学习使用深度 CNN 来近似值函数,并以雅达利 2600 的游戏截图作为直接输入。最终输出是每个可执行动作的 Q 值。

缓冲区并从中随机均匀地选择样本进行训练。此外,经验回放允许 DQN 使用离 策略方式进行训练,从而显著提高数据效率。

3) 目标网络:由于参数更新,用于更新 DQN 的相邻时间步长的值由同一个网络以不同的参数 获得,可能会造成输出不稳定等问题。Mnih 等人通过引入与主网络结构相同的目标网络解决了 该问题。

在训练开始时,两个网络使用相同的参数。在整个训练过程中,主要

Schhaul等人[74]使用的 TD 误差是

$$\delta_t = R_t + \gamma \max_a Q(s_t, a) - Q(s_{t-1}, a_{t-1}). \tag{15}$$

$\delta_t$ 的绝对值越大,对应转移被选中的概率越大,对策略改进的贡献越大。在采样过程中,Schaul等[74]采用了随机优先化和重要性采样的方法。随机优先化操作不仅可以充分利用转移,还能保证多样性。重要性采样减慢了参数更新速度,保证了学习的稳定性。

D. 决斗架构

与大多数专注于改进控制和强化学习算法的研究不同,Wang等人[75] 专注于创新更好的神经网络架构,并提出了对决架构。对决架构采用了优势更新算法 [76] 的思想,其中优势函数定义为 $A(s, a) = Q(s, a) - V(s)$ ,表示在特定状态下采取每种行动的相对优势,对决架构既估计值 $V(s)$ ,也估计优势 $A(s, a)$ ,  $Q(s, a)$  是上述两个流的组合。引入优势函数可以在多个相似值行动的情况下实现更好的策略评估。

Minh等人[70] 在 2015 年提出的 DQN 中使用的深度神经网络模型有三个卷积层,后面跟着两个全连接层,输出是每个动作的Q值。与 DQN 架构不同,决斗架构将卷积层提取的特征分为两个流。一个流估计状态值函数 $V(s)$ ,另一个流估计优势函数 $A(s,a)$ 。只需将两个流的输出相加即可获得 $Q(s, a)$

$$Q(s, a; \mathbf{b}, \mathbf{a}, \mathbf{b}) = V(s; \mathbf{b}, \mathbf{b}) + A(s, a; \mathbf{b}, \mathbf{a}) \tag{16}$$

其中  $\alpha$  和  $\beta$  分别是来自A流和V流的两个全连接层的参数。给定 $Q(s, a)$ ,  $V$ 和 $A$ 有无数种可能的组合,其中只有少数是合理的,因此必须限制 $A$ 函数。最后,Wang等人采用以下方法计算 $Q(s, a)$ :

$$Q(s, a; \mathbf{b}, \mathbf{a}, \mathbf{b}) = V(s; \mathbf{b}, \mathbf{b}) + A(s, a; \mathbf{b}, \mathbf{a}) - \frac{1}{|\mathcal{A}|} \sum_{a'} A(s, a'; \mathbf{b}, \mathbf{a}). \tag{17}$$

E. 嘈杂网络

增强探索能力是强化学习和深度强化学习中共同的问题。DQN 采用的是  $\epsilon$ - 贪婪方法。本节介绍另一种方法:噪声网络 [77]。噪声网络的思想是向神经网络中添加噪声,以影响最终的值输出,从而增强策略的探索能力。噪声越大,策略与原始策略之间的差异越大,探索能力越强。以 DQN 为例,添加随机

噪声和  $\epsilon$ ,参数为  $\delta$  和  $\delta \epsilon$ ,分别加入目标网络和主网络。新的目标函数变为 $L(\beta)$

$$\begin{aligned} &= E(s_j, a_j, r_j, s_{j+1}) - D \times \\ &R_{j+1} + \gamma Q(s_{j+1}, \arg\max_a Q(s_{j+1}, a, \beta, \delta), \epsilon; \beta, \delta) - Q(s_j, a_j, \beta, \delta)^2. \end{aligned} \tag{18}$$

F. 多步骤学习

Q学习以当前即时奖励和下一时间步的值估计作为目标值。如果前一个策略较差,网络参数偏差较大,则该方法得到的目标值偏差也较大,导致学习速度较慢。为了加快收敛速度,可以进一步扩展 (3),不仅使用下一时间步的奖励,还将更多后续时间步的奖励添加到目标值中。新的方程如下:

$$Q(s_t, a_t) = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \dots + \gamma^{n-1} R_{t+n} + \gamma^n \max_a Q(s_{t+n}, a). \tag{19}$$

通过这种方式,可以在训练初期更准确地估计目标值,从而加快训练速度,上述方法称为多步学习[78]。

G. 分配方法

在传统强化学习中,价值函数的输出是每个动作的预期回报。分布式强化学习的思想是假设价值的分布比其期望更可靠,将价值视为随机变量,其目标是估计价值的分布[79]。在[79]中,  $Z$ 被定义为期望为 $Q$  的价值分布。根据贝尔曼方程的传统定义,描述当前价值分布与未来价值分布之间关系的分布贝尔曼方程定义为

$$Z(s_t, a_t) = R + \gamma Z(s_{t+1}, a_{t+1}). \tag{20}$$

确定分布类别是确定 $Z$ 具体形式的重要一步,参数分布表示一组可学习的参数控制着分布,用来对 $Z$ 进行建模,最终 $Z$ 定义如下:

$$Z(\beta, (s_t, a_t)) = \sum_i \frac{e^{\beta_i(s_t, a_t)}}{e^{\beta_j(s_t, a_t)}} \pi_i(s_t, a_t) \tag{21}$$

利用上述离散分布,Bellemare等人提出了分类算法来确保目标值分布和估计值分布相同。Wasserstein 距离 [80] 是衡量两个概率分布之间距离的更好方法。然而,当使用 Wasserstein 距离作为损失函数时,随机梯度下降技术无法进行优化,因此 Bellemare等人[79] 使用 Kullback-Leibler (KL) 散度作为距离的度量。Dabney等人[81] 使用分位数回归 [82] 解决了这个问题,以最小化 Wasserstein 损失。

H. 其他改进和变体

DRL 社区还对 DQN 进行了其他修改,以提高其在不同任务上的性能并扩大其应用范围。Hausknecht 和 Stone [83] 提出了深度循环 Q 网络 (DRQN),它使用长短期记忆 (LSTM) 与 DQN 相结合来解决部分可观测马尔可夫决策过程 (POMDP) [45] 问题。为了让智能体具有更强的探索能力,受到汤普森抽样 [84] 的启发,Osband等人[85] 提出了引导式 DQN,它可以显著减少学习时间并提高大多数 Atari 游戏的性能。Du等人 [86] 找到了一种使用差异最大化 Q 学习 (DMQ) 和线性函数近似来有效探索状态空间的方法,并实现了一种可以学习近似最优策略的算法。

Kapturowski等人[87] 研究了参数滞后导致表征漂移和复发状态陈旧的影响,并通过实证研究得出了改进的训练策略。

这些算法都可以在不同任务上提升某一方面的性能,但目前还不清楚哪些扩展或变体是互补的、可以有效组合起来。Hessel等[88] 研究了双 DQN、优先经验回放、决斗网络、噪声网络、多步学习和分布式 DQN,通过在 57 个 Atari 游戏上的实验证明了这六个组件是互补的,并提出了一种称为彩虹的组合算法。这种组合技术此前已被 Wang等 [75] 采用,他们通过实验证明决斗架构可以与带优先经验回放的双 DQN 相结合并提高其性能。由于每个组件都在一定程度上提高了性能,因此彩虹算法在提出时取得了最优性能。

IV.基于策略的DRL方法

本节介绍的方法是前面介绍的策略梯度方法和actor-critic方法的扩展,在多维状态空间和连续动作空间中更为有效。

A. 优势演员评论家方法传统的策略梯度算法通常

以在策略的方式进行更新,导致策略收敛速度慢,数据效率低。为了加快演员评论家方法的训练速度,Mnih等人[89] 提出了异步优势演员评论家 (A3C) 来异步收集数据,其中代理在 N 个线程中同时与环境交互。只要环境设置不同,从每个线程获得的交互轨迹就不会相同,从而加快样本收集速度。在收集样本后,每个线程独立完成训练,并异步更新全局模型参数。A3C 的结构如图 3 所示。此外,Mnih等人使用优势函数作为策略的度量,以更好地平衡偏差和方差。还采用了 N 步回归技术 [90],[91] 来更有效地更新价值模型。为了增强模型的探索能力,策略的熵

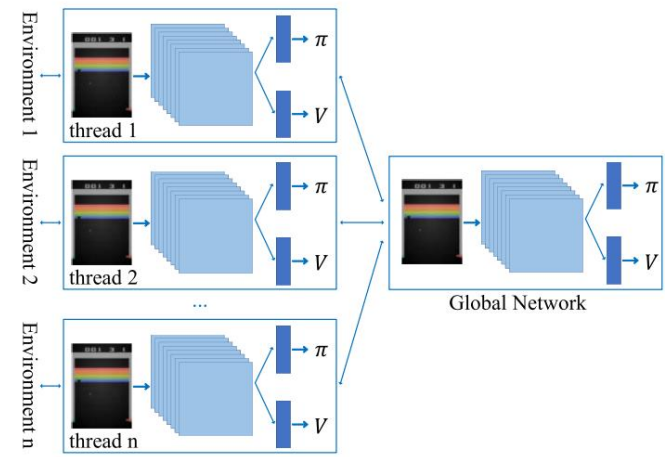


图 3. A3C 架构 [89]:全局网络采用 Actor-Critic 框架,具有 n 个工作线程,网络结构与全局网络相同。每个线程独立与自己的环境交互,并经过训练以更新全局网络参数。每隔几个时间步,这些线程的参数就会与全局网络的参数同步。

添加到目标函数中。因此,完整的策略梯度方程变为

$$J(\beta) = \mathbb{E}_{\pi(\beta)} \left[ \sum_{i=1}^n \gamma^i (r_{t+i} + v(\mathbf{s}_{t+i}) - v(\mathbf{s}_t)) + \beta \nabla \log \pi(\mathbf{s}_t; \beta) \right] \quad (22)$$

其中  $\gamma^i$  具有  $n$  步回报,  $\beta$  是  $\gamma^i$  的优势项目,  $\gamma^i$  是优势项目,  $\beta$  是优势项目,  $\gamma^i$  是优势项目。

A2C [36] 是 OpenAI 提出的 A3C 的同步版本,它可以实现与 A3C 相同或更好的性能,同时更有效地利用 GPU。具有经验回放的 Actor-critic (ACER) [92] 是一种离线策略 Actor-critic 算法,可以应用于连续和离散动作空间,可以看作 A3C 的离线策略对应物。当离线策略方法用于训练策略梯度方法时,目标策略和行为策略之间的差异可能会导致不稳定。

ACER 提出了三种方法来解决此问题。第一种方法是使用 Retrace [93] 中的 Q 值估计方法,这是一种基于离策略回报的强化学习算法,具有三个优点:1) 方差小;2) 从行为策略中收集的数据可以安全使用;3) 数据效率高。其次,ACER 采用重要性抽样来调整策略梯度的权重。因此,策略梯度

$$g_{\text{imp}} = \sum_{t=0}^{\infty} \gamma^t \nabla \log \pi(\mathbf{a}_t | \mathbf{x}_t) \quad (23)$$

其中  $\gamma^t$  表示重要性权重。然而,在计算重要性权重时,乘法操作会导致权重过大或过小。针对此问题,Wang 等人提出了一种偏差校正技巧来截断重要性权重。第三,ACER 希望更新步长的变化在策略空间中不要太大,因此采用信赖域策略优化 (TRPO)[94] 的思想来限制 KL 散度的范围。

B. 基于信赖域的算法在策略优化方程(9)、(10)中,α

控制更新的步长。设置好的步长可以使策略收敛得更快,但不合适的步长会导致策略不稳定甚至变差。从经验上讲,合适的步长应该使得新策略不比旧策略差,或者说策略应该单调递增。Schulman等人[94] 提出了 TRPO 算法,并证明了一个策略相对于另一个策略的预期优势是

$$\eta(\pi_{t+1}) - \eta(\pi_t) = E_{s_0, a_0, s_1, a_1, \dots} \sum_{t=0}^{\infty} \gamma^t A_{\pi_t}(s_t, a_t) \quad (24)$$

其中,η<sub>t</sub> 代表策略的估计值。π<sub>t</sub> 是新策略,π<sub>t-1</sub> 是旧策略。a<sub>t</sub>从 π<sub>t-1</sub>(a<sub>t</sub>|s<sub>t</sub>)中采样, s<sub>0</sub>是初始状态。因此,只要E<sub>s<sub>0</sub>,a<sub>0</sub>,s<sub>1</sub>,a<sub>1</sub>,..., [A<sub>π<sub>t</sub></sub>(s<sub>t</sub>, a<sub>t</sub>)] 保持为正,就能保证策略单调递增。Schulman等人引入了信赖域约束,以确保新旧策略在每一轮更新后不会发散太多,从而提高训练稳定性并保证单调递增。该约束由旧策略数据上新旧策略的预测分布之间的 KL 散度 [95] 定义。</sub>

Schulman等人提出了一种改进的 TRPO 算法,即近端策略优化 (PPO) [96],该算法使用截断目标函数简化了 TRPO。PPO 中的目标函数为

$$J_{\text{PPO}}(\theta) = \frac{1}{N} \sum_{i=1}^N \min(r_i(\theta) A_i, \text{clip}(r_i(\theta), 1 - \epsilon, 1 + \epsilon) A_i) \quad (25)$$

其中, r<sub>i</sub>为新旧策略的策略比, A<sub>i</sub>为优势函数。为了避免更新步长或策略比过大导致策略更新不稳定,PPO 在目标函数中加入了两个保险:第一是对r<sub>i</sub>(θ) 的限制,该比值限制在 [1 - ε, 1 + ε] 之间,保证每次更新不会出现太大的波动;第二项保险是 min 函数,即在两个结果中取较小值。

Ye等人[97] 观察到,在大规模离策略环境中,使用在策略训练方法的标准 PPO 会失败,因为当π<sub>t</sub>(a<sub>t</sub>|s<sub>t</sub>) π<sub>t-1</sub>(a<sub>t</sub>|s<sub>t</sub>)且A<sub>t</sub> < 0 时, r<sub>t</sub>(θ) 将超出边界,导致策略难以收敛。为了缓解这一问题,提出了双截断 PPO,通过进一步用下限截断r<sub>t</sub>(θ)。目标函数为

$$J_{\text{DPG}}(\theta) = E_{s_0, a_0, s_1, a_1, \dots} \sum_{t=0}^{\infty} \gamma^t \max(\min(r_t(\theta) A_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) A_t), c A_t) \quad (26)$$

其中c > 1 是常数。

还有其他方式来表达信赖域约束。使用 Kronecker 分解信赖域 (ACKTR) [98] 的 Actor Critic 采用 Kronecker 分解近似曲率 (K-FAC) [99], [100] 的信赖域公式来限制策略的更新步长。K-FAC 可以在短时间内逼近 Fisher 信息矩阵,

并且每次更新所需的时间与 TRPO 和 PPO 中使用的随机梯度下降法相似,因此可以使用自然梯度来更新模型 [101]。因此,ACKTR 实现了比 TRPO 和 PPO 更快的训练速度。

Nachum等[102] 提出了 trust-PCL,引入折扣相对熵信赖域来保证优化的稳定性,并采用离策略方法来提高样本效率。

C.确定性策略梯度

前面讨论过的以概率分布或概率密度函数建模的策略都是随机策略,因为给定任何状态,所有动作都有被采样的概率。然而,在许多任务中,策略是确定性的。在确定性策略上下文中,当代理遇到相同状态时,所选动作是相同的,表示为a = μ(s)。Silver等人[103] 推导出确定性策略梯度 (DPG) 定理如下:

$$\nabla_{\theta} J(\theta) = E_{s, a} \nabla_{\theta} Q_{\mu}(s, a) \nabla_{\theta} \mu(s) |_{a=\mu(s)} \quad (27)$$

并基于该定理可以集成到通用策略梯度框架中这一事实,提出了在线策略和离线策略的确定性演员评论家算法。

深度确定性策略梯度 (DDPG) 算法 [53] 学习确定性策略,并通过演员-评论家架构将其扩展到连续动作空间。

另外,作者将DQN、经验重放和目标网络两项技术应用于DDPG。

1)经验重放:在确定性策略问题中,价值函数只和状态相关,因此可以采用off-policy的方法。DDPG采用重放缓冲区收集样本,并随机选择其中一部分来优化价值函数,从而提高数据利用率。

2)目标网络:与 DQN 一样,为了避免高估Q值,DDPG 也使用了目标网络技术。

然而,由于DDPG中的值函数比DQN中的更复杂,因此需要一种不同于DQN的方法来同步目标网络。

该方法为软替换,通过让目标网络在每次训练迭代中“慢慢接近”主网络的方向,使训练更加稳定。同步方法转向

$$\begin{aligned} w &\leftarrow \tau w + (1 - \tau) w \\ \beta &\leftarrow \tau \beta + (1 - \tau) \beta \end{aligned} \quad (28)$$

其中,β和β<sub>-</sub>分别为主动行动者网络和目标行动者网络的参数,ω和ω<sub>-</sub>分别为主评论者网络和目标评论者网络的参数,τ<sub>1</sub>为更新系数。

除了上述两种技巧外,作者还采用了 Ornstein–Uhlenbeck [104] 噪声来增强 DDPG 的探索能力。Fujimoto等人[105] 将许多技术应用于 DDPG 以防止高估价值函数,并提出了孪生延迟深度确定性 (TD3)。这些技术如下。



- 1) CLIPPED Double Q-Learning: TD3使用两个独立的评论家,选择较小的估计值来更新目标值。
- 2)目标和策略网络的延迟更新: Td3在评论家每更新d次后更新策略,而不是同时更新,以减少多次更新中积累的错误。
- 3)目标策略平滑 (Target Policy Smoothing) : TD3对目标动作周围一小块区域的估计值进行平滑和正则化,减少价值函数估计引入的过拟合。

五、最大熵深度强化学习

A.最大熵强化学习框架

传统强化学习的目标可以表示为最大化预期奖励之和 $t=0 \mathbb{E}(r(s_t, a_t) \mid \pi)$ ,其中 $\pi$ 是由策略引起的分布 [106]。最大熵强化学习的思想是用熵项来增加奖励,这样最优策略旨在同时最大化每个状态下的熵和奖励

$$J(\pi) = \mathbb{E}_{t=0} [r(s_t, a_t) \mid \pi] + \alpha H(\pi(\cdot | s_t)) \quad (29)$$

其中  $\alpha$  是一个温度参数,可以控制优化目标,使其更加关注奖励或熵。

最大熵强化学习具有许多优点:1)由于熵项的存在,智能体可以学习最优随机策略;2)探索能力更强。熵项鼓励策略进行更广泛的探索,同时放弃无望的方法;3)策略可以在多模态奖励场景中寻找最佳模式 [106]。

许多早期作品都讨论过将强化学习与熵相结合的想法。玻尔兹曼探索 [107] 和策略梯度与 Q 学习 (PGQ) [108] 学习在每个时间步贪婪地最大化熵。Ziebart 等人 [109] 和 Boularias 等人 [110] 结合最大熵和逆强化学习 [111] 来计算专家轨迹的概率分布。Nachum 等人 [112] 在熵正则化的背景下研究了价值和基于策略的强化学习之间的联系,并提出了优于 A3C 和 DQN 基线的路径一致性学习 (PCL)。(29) 中的优化目标与 A3C 的损失函数形式相同,而 A3C 中的熵项仅用作正则项。

B.软 Q 学习和 SAC

传统强化学习给出的策略是以最优Q值为中心的分布。从该策略中采样的动作始终接近最优Q值,从而忽略了次优Q值,导致智能体无法学习完成任务的多种模式。Haarnoja等人[106] 提出了一种基于能量的模型,使用指数Q值来定义策略

$$\pi(a|s) \propto \exp Q(s, a)。$$

(30)

这样,策略就可以为每个动作分配一个特定的概率,成为随机策略。在这种情况下,找到一个好的策略的关键在于找到一个好的Q函数。

让软Q函数和V函数定义为

$$Q_{\text{soft}} = r + \mathbb{E}_{l \sim \pi} [ \sum_{l=1}^{\infty} \gamma^l (r_t + l + \alpha H(\pi(\cdot | s_t + l))) ]$$

(31)

$$V_{\text{软}} = \alpha \cdot \log \int_{\text{经验}} \frac{1}{\pi} Q_{\text{soft}}(s_t, a) da$$

(32)

Haarnoja等人证明这两个定义满足贝尔曼方程的形式,称为软贝尔曼备份。因此,可以推导出最优策略的形式为

$$\pi(\cdot | s_t) = \exp \left( \frac{1}{\alpha} (Q_{\text{soft}}(s_t, \cdot) - V_{\text{soft}}(s_t)) \right)$$

(33)

为了更好地逼近,软Q函数由以 $\beta$ 为参数的神经网络建模。重要性抽样用于将软Q迭代转化为随机优化问题,并使用随机梯度下降来更新Q网络。状态条件随机神经网络用于近似基于能量的策略,以从基于能量的模型中获得无偏动作样本。

Schulman等人[113] 证明了软Q学习和策略梯度方法是等价的,因为软Q学习的损失梯度可以表示为一个策略梯度项和一个基线误差梯度项。Wei等人[114] 在多智能体强化学习中采用了软Q学习,并且在协作任务中取得了比最先进的方法多智能体深度确定性策略梯度 (MADDPG) [115] 更好的性能 (第 VI-C 节)。

Haarnoja等人[116] 证明了在表格情况下软策略迭代可以收敛到最优策略。

为了能够在大型连续领域中应用,他们使用具有参数  $\beta$  和  $\phi$  的神经网络来表示软Q函数和策略:  $Q_{\beta}(s_t, a_t)$ 和 $\pi_{\phi}(a_t|s_t)$ ,并提出了软演员评论家 (SAC),这是第一个结合离策略、演员评论家和最大熵方法的 RL 算法。

VI.进一步的研究主题

上文介绍的大部分深度强化学习方法只适用于游戏和模拟环境,在这些环境中,智能体可以获得丰富的奖励信号并进行无限的rollout。然而,更复杂的环境或现实环境往往面临样本有限、奖励稀疏和多个智能体的问题。为了解决上述问题,强化学习研究人员开创了进一步的研究方向,如基于模型的强化学习、分层强化学习和元强化学习。在深度学习的帮助下,这些方向得到了进一步的发展。下面简要介绍其中一些。

A.基于模型的方法

以上介绍的所有算法都是无模型的,它们直接从代理与环境交互获得的数据中学习最优值函数或策略。尽管无模型方法取得了优异的

尽管在许多任务上都表现出色,但由于需要大量的训练样本,将其应用于实际问题仍然很棘手。此外,经验回放中的转换引入的误差是离策略无模型方法难以消除的负担 [117],[118],[119]。另一类 RL 算法是基于模型的方法,其目的是学习一个可以描述环境如何变化的动态模型 [120]。一旦学习到的模型接近环境,就可以直接通过一些规划方法找到最优策略。更重要的是,学习到的模型可以直接用于在策略训练 [121]。

高采样复杂度限制了无模型算法的使用范围,尤其是在面对高维函数逼近器时。顾等[122]研究了连续控制任务中的采样复杂度,提出了一种算法,主要包含两种互补的技术。首先,通过简化 Actor-Critic 架构,提出了正则化优势函数 (NAF),使Q 学习能够适应连续动作空间。

然后将连续Q学习与从离策略经验中学习到的模型相结合,以加速算法在连续动作空间中的训练过程。

基于复合误差导致长期部署不可靠的分析,Janner等人[123]提出了一种基于模型的策略优化 (MBPO) 算法,该算法从以前遇到的真实数据中学习一个模型来生成短期部署。Hafner等人[124]提出了 Dreamer 算法,该算法首先从经验中学习一个世界模型,然后使用演员-评论家算法与学习到的世界模型进行交互并学习最优策略。训练在学习到的模型中进行,因此可以获得多步累积奖励并进行长期规划。

B. 分层深度强化学习

HRL 被提出用于解决奖励稀疏且需要有针对性探索的任务。传统的 HRL 思路是将复杂的目标分解为易于解决的子目标,最终通过解决这些子目标完成原任务 [125] 或抽象出不同层次的控制层 [126]。

但由于表示能力不足,子目标的选取依赖于专家知识,而借助深度学习,HRL可以解决状态空间和动作空间较大的问题,并且可以直接处理高维输入。

此外,深度学习的表征能力为子目标的形式提供了更多的选择[125],分层深度强化学习 (HDRL)已成为DRL的一个重要研究方向。

如图 4 所示,Kulkarni等[127]提出了一种分层模型 HIRO,该模型由一个高层控制器提供粗粒度子目标,一个低层控制器完成这些子目标组成。期权框架由 Sutton等[126]首次提出,是一类传统的 HRL。Bacon等[128]推导了期权策略权重定理,并提出了一种期权批评架构,可以学习期权的内部策略和终止条件。Andrychowicz等[129]在经验回放的转换元组中加入了目标。利用目标之间的相似性,具有不同目标的转换元组可用于训练其他

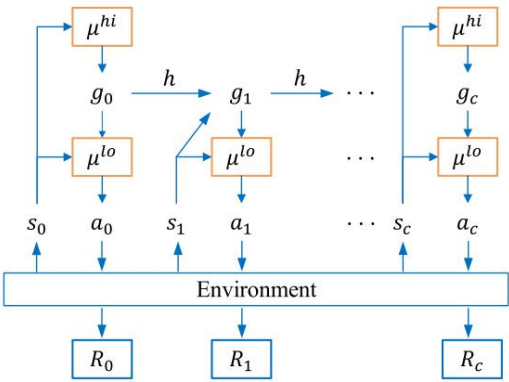


图 4. HIRO 的交互过程:环境首先产生一个初始状态 $s_0$ ,然后将其传递给 $\mu^h$ 和 $\mu^l$ 。高级控制器根据 $\mu^h$ 生成目标 $g_0$ 。随后,低级控制器将 $s_0$ 和 $g_0$ 作为输入,并根据 $\mu^l$ 选择动作 $a_0$ 。环境给出奖励 $R_0$ ,状态转换到 $s_1$ 。当前目标 $g_1$ 是通过函数  $h$  从上一个时间步的目标和当前状态生成的。在使用 $\mu^h$ 生成新目标之前,上述操作执行 $c$ 次。

目标。Eysenbach等人[130]提出了“多样性就是一切”(DIAYN)算法,该算法允许代理即使在没有办法获得真实奖励的情况下也能学习各种任务解决技能,从而帮助代理探索环境。更重要的是,这些技能的组合可用于解决具有稀疏奖励的更复杂的问题。

C. 多智能体深度强化学习

多智能体系统通常可以描述为马尔可夫博弈 [131],其中所有智能体根据当前状态同时选择并执行其动作。每个智能体的行为都会影响环境的转变。

在一些复杂的任务中,每个智能体可能只观察部分环境 [45]。在深度学习出现之前,研究人员已经使用强化学习来解决完全合作、完全竞争和混合任务 [132]。

多智能体DRL (MADRL)将DRL的思想应用于多智能体系统的学习和控制,从而克服了传统方法遇到的一些困难,例如高维输入、连续动作空间等。

Lowe等人[115]提出了 DDPG 的多智能体版本,即 MADDPG,它允许每个智能体获得评论家部分中所有其他智能体的动作信息,从而实现集中训练和分散执行。

然而,随着 MADRL 在复杂状态空间和动作空间任务中的应用越来越广泛,诸多问题也随之显现。为了激励那些对整个多智能体任务更有帮助的智能体,Foerster等[133]提出了“反事实基线”的思想,用于解决信用分配任务。价值分解网络 (VDN) [134] 将各个智能体的价值函数整合起来得到一个联合Q函数,解决了环境部分可观测性导致的虚假奖励问题和懒惰智能体问题。QMIX [135] 是 VDN 的扩展,它使用混合网络合并单个智能体的局部价值函数,并在训练过程中添加全局状态信息,以提高算法性能。

D. 从演示中学习在一些奖励稀疏或多目标的任  
务中,基于累积奖励的学习算法往往无法使代理智能地行动,因为探索空间巨大,奖励设置困难。为了确保代理更快地学习最优或次优策略,从专家的演示中学习奖励函数或策略通常是好主意,逆强化学习 [111] 和模仿学习 [136] 分别采用了这种做法。

早期基于逆强化学习的方法主要包括学徒学习 [137]、结构化分类 [138]、最大边际规划 [139] 和最大熵方法 [109], [110]。近年来,提出了一些利用深度学习技术的逆强化学习方法。

Finn等人[140] 研究了 GAN 与逆强化学习之间的联系。Fu等人[141] 提出了对抗性逆强化学习 (AIRL) ,并证明 AIRL 可以恢复对环境动态变化具有鲁棒性的奖励函数。

模仿学习的主要思想是使用监督学习方法从专家演示中学习策略[142]。传统的模仿学习方法侧重于消除行为策略与学习策略[143]和数据增强方法[144]之间的复合误差。由于模仿学习采用监督训练方式,因此很容易与深度学习相结合[145]。

Ho 和 Ermon [146] 提出了一种新的通用框架,称为生成对抗模仿学习 (GAIL) ,通过结合 GAN 和模仿学习直接从数据中提取策略,并且在大型高维环境中优于其他模仿学习算法。Song等人[147] 提出了 AdaBoost 最大熵深度逆强化学习 (AME-DIRL)算法,该算法可以在专家演示数据有限且不平衡的情况下学习非线性奖励。

E.元强化学习

元学习[148]的目的是通过从多个其他任务中学习经验,将知识推广到新任务。元强化学习[149]的主要思想是将大量强化学习任务中学习到的先验知识应用到新的强化学习任务中,提高智能体的学习速度和泛化能力。元强化学习主要解决和优化了深度强化学习的以下缺点:样本利用率低[150]、奖励函数设计困难[151]、未知任务中的探索策略[152]、缺乏泛化能力[153]。

F.离线强化学习强化学习的训练过程需要agent

与环境进行交互,经过大量的尝试和错误,最终找到最优的策略。当没有模拟,与环境交互的成本很高时,如何利用之前收集的数据进行训练是一个关键问题。

离线强化学习[154]是近年来不断发展的一种方法,它指的是利用过去收集的离线数据来训练智能体,在训练过程中无需交互。离线强化学习的一个关键问题是,模型的可靠性和性能。

离线数据集中的行为策略 $\pi_\beta$ 和离线数据集中代理策略 $\pi$ 之间的分布转变[154]。

目前,离线强化学习方法主要分为三类 [154]。第一类方法利用重要性抽样直接评估策略回报 $J(\pi)$  或通过从 $\pi_\beta$ 中采样的离线数据估计相应的策略梯度[155]。第二种方法是使用基于动态规划的强化学习算法,例如基于Q 学习的算法。此类算法在没有在线数据的情况下不能直接应用,因此提出了策略约束方法 [117] 和基于不确定性的方法 [156]。在策略约束方法中,将学习到的策略限制在行为策略附近,以消除分布偏移。基于不确定性的方法尝试利用Q 值的认知不确定性来检测分布偏移。

最后一类是前面介绍的基于模型的方法。

基于模型的方法学习一个环境的状态转换概率,并且可以使用监督方法[157],[158],因此可以更好地利用离线数据集。

G.强化学习中的迁移学习

在大多数强化学习算法中,微小的状态变化都会导致先前训练的策略失败,而从头开始训练的成本很高。迁移学习 (TL) [159] 可以通过应用从源任务中学到的经验来提高目标任务的训练效率。因此,为了提高新任务的学习速度,用于强化学习的 TL [160] 已成为一个颇具吸引力的研究方向。根据迁移的知识内容,TL 可以在以下方面为深度强化学习算法提供帮助 [161]:奖励塑造、从演示中学习、策略迁移、任务间映射和表示迁移。Lan等[162] 提出了一种新颖的迁移强化学习方法,该方法通过使用自动剪枝决策树进行元知识提取。Zhang等[163] 提出了 DQDR 方法,该方法从人类中提取领域知识并将其表示为一组规则,然后将其与 DQN 耦合,以提高强化学习算法的可迁移性。

VII.结论

DRL结合了深度学习和强化学习的优点,可以直接根据输入数据进行决策。本文介绍了当前流行的DRL库中实现的算法,包括基于价值的算法、基于策略的算法和基于最大熵的算法,并简要列出了近年来DRL的研究子领域。对DRL的研究是构建对世界有更高理解的自主系统的关键一步。

参考

[1] ML Puterman, 《马尔可夫决策过程:离散随机动态规划》。美国新泽西州霍博肯:Wiley, 2014 年。

[2] M. Minsky, “迈向人工智能”, Proc. IRE,第 49 卷,第 1 期,第 8-30 页,1961 年 1 月。

[3] RS Sutton 和 AG Barto, 《强化学习:导论》。美国马萨诸塞州剑桥:麻省理工学院出版社,2018 年。

[4] R.Bellman, “动态规划”, Science,第 153 卷,第 3731 期,第 34-37 页,1966 年 7 月。

[5] D. Bertsekas,动态规划和最优控制,第 1 卷。美国加利福尼亚州贝尔蒙特:Athena Sci.,2000 年。

[6] AL Samuel, “使用跳棋游戏进行机器学习的一些研究”, IBM J. Res. Develop.,第 3 卷,第 3 期,第 210-229 页,1959 年。

[7] R. Munos 和 A. Moore, “最优控制中的可变分辨率离散化”, Mach. Learn.,第 49 卷,第 2 期,第 291-323 页,2002 年。

[8] V. Mnih等人, “利用深度强化学习玩 Atari 游戏”,2013 年, arXiv:1312.5602。

[9] Y. LeCun,L. Bottou,Y. Bengio 和 P. Haffner, “基于梯度的学习应用于文档识别”, IEEE 会刊,第 86 卷,第 11 期,第 2278-2324 页,1998 年 11 月。

[10] A. Krizhevsky,J. Sutskever 和 GE Hinton,《使用深度卷积神经网络进行 ImageNet 分类》,《Proc. Adv. Neural Inf.》Process. Syst. (NIPS),第 25 卷,2012 年,第 1097-1105 页。

[11] K. He,X. Zhang,S. Ren 和 J. Sun, “深度残差学习用于图像识别”,载于Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016 年 6 月,第 770-778 页。

[12] L. Deng 和 D. Yu, “深度学习:方法和应用”, Found. 趋势信号过程,第 7 卷第 3-4 期,第 197-387 页,2013 年。

[13] K. Hornik,M. Stinchcombe 和 H. White,《多层前馈网络是通用近似器》,《神经网络》,第 2 卷,第 5 期,第 359-366 页,1989 年 12 月。

[14] G. Cybenko, “通过 S 形函数叠加近似”, Math. Control Signals Syst.,第 2 卷,第 4 期,第 303-314 页,1989 年 12 月。

[15] K. Hornik,M. Stinchcombe 和 H. White,《利用多层前馈网络对未知映射及其导数进行通用近似》,《神经网络》,第 3 卷,第 5 期,第 551-560 页,1990 年。

[16] L. Tai,J. Zhang,M. Liu 和 W. Burgard, “通过生成对抗模仿学习实现通过原始深度输入实现社交兼容导航”,载于IEEE 国际机器人与自动化会议论文集 (ICRA), 2018 年 5 月,第 1111-1117 页。

[17] L. Tai,G. Paolo 和 M. Liu, “虚拟到现实的深度强化学习:无地图导航的移动机器人持续控制”, IEEE/RSJ 国际智能机器人系统会议论文集 (IROS), 2017 年 9 月,第 31-36 页。

[18] O. Zhelo,J. Zhang,L. Tai,M. Liu 和 W. Burgard, “好奇心驱动的无地图导航与深度强化学习探索”,2018 年, arXiv:1804.00456。

[19] J. Hwangbo等人, “学习腿式机器人的敏捷和动态运动技能”,《Sci. Robot.》,第 4 卷,第 26 期,2019 年 1 月,文章编号 eaau5872。

[20] D. Silver等人, “利用深度神经网络和树搜索掌握围棋游戏”,《自然》,第 529 卷,第 7587 期,第 484-489 页,2016 年。

[21] D. Silver等, “无需人类知识即可掌握围棋游戏”,《自然》,第 550 卷,第 7676 期,第 354-359 页,2017 年。

[22] D. Silver等人, “一种通过自学掌握国际象棋、将棋和围棋的通用强化学习算法”,《科学》,第 362 卷,第 6419 期,第 1140-1144 页,2018 年 12 月。

[23] O. Vinyals等人, “使用多智能体强化学习实现《星际争霸 II》中的大师级水平”,《自然》,第 575 卷,第 7782 期,第 350-354 页,2019 年。

[24] C. Berner等人, “具有大规模深度强化学习的 Dota 2”,2019 年, arXiv:1912.06680。

[25] DA Hudson 和 CD Manning, “用于机器推理的合成注意力网络”,载于Proc. Int. Conf. Learn. Represent., 2018 年,第 1-20 页。

[26] X. Wang,W. Chen,J. Wu,Y.-F. Wang 和 WY Wang, “通过分层强化学习实现视频字幕”,载于Proc. IEEE/ CVF Conf. Comput. Vis. Pattern Recognit., 2018 年 6 月,第 4213-4222 页。

[27] L. Wu,F. Tian,T. Qin,J. Lai 和 T.-Y. Liu, “神经机器翻译的强化学习研究”,载于Proc. Conf. Empirical Methods Natural Lang. Process., 2018 年,第 3612-3621 页。

[28] V. Talpaert等人, “探索深度强化学习在现实世界自动驾驶系统中的应用”,第 14 届国际联合会论文集,计算机视觉、图像、计算机图形理论应用,葡萄牙塞图巴尔:SCITEPRESS,2019 年,第 564-572 页。

[29] S. Milz,G. Arbeiter,C. Witt,B. Abdallah 和 S. Yogamani, “自动驾驶的视觉 SLAM:探索深度学习的应用”,载于Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.

研讨会 (CVPRW), 2018 年 6 月,第 247-257 页。

[30] J. Li,L. Yao,X. Xu,B. Cheng 和 J. Ren, “深度强化学习用于行人防撞和人机协同驾驶”,《信息科学》,第 532 卷,第 110-124 页,2020 年 9 月。

[31] G.Zheng等人, “DRN:用于新闻推荐的深度强化学习框架”,载于Proc. World Wide Web Conf., 2018 年,第 167-176 页。

[32] M. Chen,A. Beutel,P. Covington,S. Jain,F. Belletti 和 EH Chi, “REINFORCE 推荐系统的 Top-K 离线策略校正”,载于第 12 届 ACM 国际会议网络搜索数据挖掘会议论文集, 2019 年 1 月,第 456-464 页。

[33] S. Yun,J. Choi,Y. Yoo,K. Yun 和 JY Choi, “通过深度强化学习实现视觉跟踪的动作决策网络”,载于IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) 会议论文集, 2017 年 7 月,第 2711-2720 页。

[34] SA Eslami等人, “神经场景表征与渲染”, Science, 第 360 卷,第 6394 期,第 1204-1210 页,2018 年。

[35] J. Wu,E. Lu,P. Kohli,B. Freeman 和 J. Tenenbaum, “学习通过视觉去动画看待物理学”,载于Proc. NIPS, 2017 年,第 153-164 页。

[36] P. Dhariwal等人。(2017)。OpenAI 基线。[在线的]。可用的: <https://github.com/openai/baselines>

[37] I. Caspi,G. Leibovich,S. Endrawis 和 G. Novik, “强化学习教练”,版本 0.10.0,Zenodo,2017 年 12 月, doi: [10.5281/zenodo.1134899](https://doi.org/10.5281/zenodo.1134899)。

[38] S. Guadarrama等人 (2018 年)。TF -Agents:Tensorflow 中的强化学习库。访问时间:2019 年 6 月 25 日。[在线]。网址:<https://github.com/tensorflow/agents> [39] J. Weng等人。

(2020)天守。[在线]。可访问:<https://github.com/thu-ml/tianshou> [40] K. Arulkumaran,MP Deisenroth, M. Brundage 和 AA Bharath, “深度强化学习:简要调查”, IEEE 信号过程。

杂志,卷。34。没有。第 6 页。11 月 26 日至 38 日2017 年。

[41] V. François-Lavet,P. Henderson,R. Islam,MG Bellemare 和 J.Pineau, “深度强化学习简介”, Found. Trends Mach. Learn.,第 11 卷,第 3-4 期,第 219-354 页,2018 年 12 月。

[42] Y. Li, “深度强化学习:概述”,2017 年, arXiv:1701.07274。

[43] H.-N. Wang等人,《深度强化学习:一项调查》,《Frontiers Inf. Technol. Electron. Eng.》,第 21 卷,第 1726-1744 页,2020 年 10 月。

[44] R.Bellman, “马尔可夫决策过程”,印第安纳大学数学杂志,第 6 卷,第 4 期,第 679-684 页,1957 年 4 月。

[45] LP Kaelbling,ML Littman 和 AR Cassandra, “部分可观察随机域中的规划和行动”, Artif. Intell.,第 101 卷,第 1-2 期,第 99-134 页,1998 年。

[46] RA Howard,动态规划和马尔可夫过程。美国马萨诸塞州剑桥:麻省理工学院出版社,1960 年。

[47] RS Sutton, “学习通过时间差异的方法进行预测”, Mach. Learn.,第 3 卷,第 1 期,第 9-44 页,1988 年。

[48] C. Szepesvári, “强化学习算法”, Synth. Lectures Artif. Intell. Mach. Learn.,第 4 卷,第 1 期,第 1-103 页,2010 年。

[49] GA Rummery 和 M. Niranjan, “利用连接的在线 Q 学习 tionist systems”,Citeseer,1994 年,第 37 卷。

[50] CJCH Watkins 和 P. Dayan,《Q 学习》,《Mach. Learn.》,第 8 卷,第 3-4 期,第 279-292 页,1992 年。

[51] RS Sutton,DA McAllester,SP Singh 和 Y. Mansour, “基于函数近似的强化学习的策略梯度方法”, Proc. NIPS,第 99 卷,1999 年,第 1057-1063 页。

[52] RJ Williams, “用于联结强化学习的简单统计梯度跟踪算法”, Mach. Learn.,第 8 卷,第 3-4 期,第 229-256 页,1992 年。

[53] TP Lillicrap等, “深层加强的连续控制学习”,载于Proc. ICLR, 2016 年,第 1-14 页。

[54] AE Sallab,M. Abdou,E. Perot 和 S. Yogamani, “端到端深度强化学习用于车道保持辅助”,2016 年, arXiv:1612.04340。

[55] I. Goodfellow,Y. Bengio,A. Courville 和 Y. Bengio,《深度学习》,第 1 卷。美国马萨诸塞州剑桥:麻省理工学院出版社,2016 年。

[56] Y. LeCun,Y. Bengio 和 GE Hinton,《深度学习》,《自然》,第 521 卷,第 7553 期,第 436-444 页,2015 年 12 月。

[57] S. Hochreiter 和 J. Schmidhuber,《长短期记忆》,《神经计算》,第 9 卷,第 8 期,第 1735-1780 页,1997 年。

[58] T. Mikolov,L. Sutskever,K. Chen,G. Corrado 和 J. Dean, “单词和短语的分布式表示及其组合性”,载于Proc. 26th Adv. Neural Inf. Process. Syst.,第 2 卷,2013 年 12 月,第 3111-3119 页。

[59] T. Mikolov,K. Chen,G. Corrado 和 J. Dean, “向量空间中词语表征的有效估计”,载于Proc. 1st Int. Conf. Learn. 代表。研讨会论文集,美国亚利桑那州斯科茨代尔,2013 年 5 月,第 1-12 页。

[60] K. Cho等人, “使用 RNN 编码器-解码器学习短语表征以进行统计机器翻译”, Proc. Conf. 经验方法自然语言。过程。(EMNLP), 2014 年,第 17 页。1-5。

[61] J. Pennington,R. Socher 和 C. Manning,《GloVe:用于词语表示的全局向量》,载于Proc. Conf. Empirical Methods Natural Langatrical Language. 过程。(EMNLP), 2014 年,第 1532-1543 页。

[62] Y. Zhang 和 B. Wallace, “卷积神经网络在句子分类中的敏感性分析(及实践者指南)”,2015 年, arXiv:1510.03820。



[63] IJ Goodfellow等人,“生成对抗网络”,第 27 届国际会议论文集 Conf. Neural Inf. Process. Syst. (NIPS),第 2 卷,2014 年,第 2672-2680 页。

[64] V. Mnih等人,“视觉注意力的循环模型”,载于Proc. Adv. Neural Inf. Process. Syst., 2014 年,第 2204-2212 页。

[65] D. Bahdanau,KH Cho 和 Y. Bengio,“通过联合学习对齐和翻译实现神经机器翻译”,第三届中国国际学习会议论文集。代表。(ICLR), 2015 年,第 1-15 页。

[66] A.Vaswani等人,“注意力就是你所需要的一切”,载于Proc. Adv. Neural Inf. Process. Syst., 2017,第 5998-6008 页。

[67] J. Yosinski,J. Clune,Y. Bengio 和 H. Lipson,“神经网络中的特征有多可转移?”,第 27 届国际神经信息会议论文集。Process. Syst.,第 2 卷,2014 年 12 月,第 3320-3328 页。

[68] F. Rosenblatt,《感知器:大脑中信息存储和组织的概率模型》,《心理学评论》,第 65 卷,第 6 期,第 386 页,1958 年。

[69] T. Hastie,R. Tibshirani 和 J. Friedman,《统计学习要素:数据挖掘、推理和预测》,美国纽约州纽约:Springer,2009 年。

[70] V. Mnih等人,“通过深度强化学习实现人类水平的控制”,《自然》,第 518 卷,第 7540 期,第 529-533 页,2015 年。

[71] S. Thrun 和 A. Schwartz,“在强化学习中使用函数逼近的问题”,载于第 4 届联结主义模型暑期学校论文集,美国新泽西州希尔斯特代尔,1993 年,第 255-263 页。

[72] HV Hasselt,《双 Q 学习》,《神经信息过程系统学报》。(NIPS),第 23 卷,2010 年 12 月,第 2613-2621 页。

[73] H. Van Hasselt,A. Guez 和 D. Silver,“基于双 Q 学习的深度强化学习”,载于Proc. AAAI Conf. Artif. Intell., 2016 年,第 30 卷,第 1 期,第 1-7 页。

[74] T. Schaul,J. Quan,J. Antonoglou 和 D. Silver,“优先经验 replay”,载于Proc. ICLR, 2016 年,第 1-21 页。

[75] Z. Wang,T. Schaul,M. Hessel,H. Hasselt,M. Lanctot 和 N. Freitas,“深度强化学习的对决网络架构”,载于Proc. Int. Conf. Mach. Learn., 2016 年,第 1995-2003 页。

[76] LC Baird,“优势更新”,赖特实验室,技术代表 WL-TR-93-1146,1993 年。

[77] M. Fortunato等人,“用于探索的噪声网络”,载于Proc. Int. Conf. 学习,代表,2018 年,第 1-21 页。

[78] T. Hester等人,“来自演示的深度 Q 学习”,Proc. AAAI 会议阿蒂夫.情报, 2018 年,卷。32.没有。1,页。1-8

[79] MG Bellemare,W. Dabney 和 R. Munos,“强化学习的分布视角”,载于Proc. Int. Conf. Mach. Learn., 2017 年,第 449-458 页。

[80] PJ Bickel 和 DA Freedman,“引导程序的一些渐近理论”,《统计年鉴》,第 9 卷,第 6 期,第 1196-1217 页,1981 年 11 月。

[81] W. Dabney,M. Rowland,M. Bellemare 和 R. Munos,“基于分位数回归的分布强化学习”,载于Proc. AAAI Conf. 艺术家英特尔, 2018 年,卷。32.没有。1,页。1-10。

[82] R. Koenker 和 K. Hallock,“分位数回归”,J. Econ. Perspect.,第 15 卷,第 4 期,第 143-156 页,2001 年。

[83] M. Hausknecht 和 P. Stone,“深度循环 Q 学习用于部分可观察的 MDP”,载于Proc. AAAI Fall Symp. Ser., 2015 年,第 1-9 页。

[84] WR Thompson,《从两个样本的证据看一个未知概率超过另一个未知概率的可能性》,《Biometrika》,第 25 卷,第 285-294 页,1933 年 12 月。

[85] I. Osband,C. Blundell,A. Pritzel 和 BV Roy,“通过引导式 DQN 进行深度探索”,载于第 30 届国际神经信息过程系统会议论文集, 2016 年,第 4033-4041 页。

[86] SS Du,Y. Luo,R. Wang 和 H. Zhang,“通过分布偏移误差检查 Oracle 实现可证明有效的函数近似 Q 学习”,载于Proc. Adv. Neural Inf. Process. Syst. 美国纽约州雷德胡克:Curran Associates,2019 年,第 1-11 页。

[87] S. Kapturowski,G. Ostrovski,J. Quan,R. Munos 和 W. Dabney,“分布式强化学习中的循环经验重放”,载于Proc. Int. Conf. Learn. Represent., 2018 年,第 1-19 页。

[88] M. Hessel等人,“Rainbow 结合深度强化学习的改进”,载于Proc. AAAI Conf. Artif. Intell., 2018 年,第 32 卷,第 1 期,第 1-14 页。

[89] V. Mnih等人,“深度强化学习的异步方法”,载于Proc. Int. Conf. Mach. Learn., 2016 年,第 1928-1937 页。

[90] CJCH Watkins,《从延迟奖励中学习》,伦敦国王学院,英国剑桥,1989 年。

[91] J. Peng 和 RJ Williams,《增量多步 Q 学习》,《1994 年机器学习论文集》,荷兰阿姆斯特丹:Elsevier,1994 年,第 226-232 页。

[92] Z. Wang等人,“基于经验重放的样本高效演员评论家”,2016 年, arXiv:1611.01224。

[93] R. Munos,T. Stepleton,A. Harutyunyan 和 MG Bellemare,“安全高效的离策略强化学习”,第 30 届国际会议论文集 Neural Inf. Process. Syst., 2016 年,第 1054-1062 页。

[94] J. Schulman,S. Levine,P. Abbeel,M. Jordan 和 P. Moritz,“信任区域策略优化”,载于Proc. Int. Conf. Mach. Learn., 2015 年,第 1889-1897 页。

[95] S. Kullback 和 RA Leibler,“论信息与充分性”,Ann. Math. Statist.,第 22 卷,第 1 期,第 79-86 页,1951 年。

[96] J. Schulman,F. Wolski,P. Dhariwal,A. Radford 和 O. Klimov,“近端策略优化算法”,2017 年, arXiv:1707.06347。

[97] D. Ye等人,“通过深度强化学习掌握 MOBA 游戏中的复杂控制”,载于Proc. AAAI Conf. Artif. Intell., 2020 年,第 34 卷,第 4 期,第 6672-6679 页。

[98] Y. Wu,E. Mansimov,S. Liao,R. Grosse 和 J. Ba,“使用 Kronecker 分解近似的深度强化学习的可扩展信赖区域方法”,载于第 31 届国际神经信息过程系统会议论文集, 2017 年,第 5285-5294 页。

[99] R.Grosse 和 J.Martens,“卷积层的 Kronecker 分解近似 Fisher 矩阵”,载于Proc. Int. Conf. Mach. Learn., 2016 年,第 573-582 页。

[100] J. Martens 和 R. Grosse,“利用 Kronecker 分解近似曲率优化神经网络”,载于Proc. Int. Conf. Mach. 学习, 2015 年,第 2408-2417 页。

[101] SM Kakade,“自然策略梯度”,载于Proc. Adv. Neural Inf. Process. Syst.,第 14 卷,2001 年,第 1-8 页。

[102] O. Nachum,M. Norouzi,K. Xu 和 D. Schuurmans,“Trust-PCL:一种用于连续控制的离策略信赖域方法”,Proc. Int. Conf. Learn. Represent., 2018 年,第 1-14 页。

[103] D. Silver,G. Lever,N. Heess,T. Degris,D. Wierstra 和 M. Riedmiller,“确定性策略梯度算法”,载于Proc. Int. Conf. Mach. Learn., 2014 年,第 387-395 页。

[104] GE Uhlenbeck 和 LS Ornstein,《论布朗运动理论》,《Phys. Rev.》,第 36 卷,第 823 页,1930 年 9 月。

[105] S. Fujimoto,H. Hoof 和 D. Meger,“解决演员-评论家方法中的函数逼近误差”,载于Proc. Int. Conf. Mach. Learn., 2018 年,第 1587-1596 页。

[106] T. Haarnoja,H. Tang,P. Abbeel 和 S. Levine,“基于深度能量的策略的强化学习”,载于 Proc. Int. Conf. Mach. 学习, 2017 年,第 1352-1361 页。

[107] B. Sallans 和 GE Hinton,《基于分解状态和动作的强化学习》,《J. Mach. Learn. Res.》,第 5 卷,第 8 期,第 1063-1088 页,2004 年。

[108] B. O Donoghue,R. Munos,K. Kavukcuoglu 和 V. Mnih,“结合策略梯度和 Q 学习”,2016 年, arXiv:1611.01626。

[109] BD Ziebart,AL Maas,JA Bagnell 和 AK Dey,“最大熵逆强化学习”,载于Proc. AAAI,第 8 卷,美国伊利诺伊州芝加哥,2008 年,第 1433-1438 页。

[110] A. Boularias,J. Kober 和 J. Peters,“相对熵逆强化学习”,载于第 14 届国际人工智能统计会议论文集, JMLR 研讨会和会议论文集,2011 年,第 182-189 页。

[111] AY Ng 和 S. Russell,“逆向强化学习算法”,载于Proc. ICML,第 1 卷,2000 年,第 1-2 页。

[112] O. Nachum,M. Norouzi,K. Xu 和 D. Schuurmans,“弥合基于价值和基于策略的强化学习之间的差距”,载于第 31 届国际神经信息过程系统会议论文集, 2017 年,第 2772-2782 页。

[113] J. Schulman,X. Chen 和 P. Abbeel,“策略梯度与软 Q 学习之间的等价性”,2017 年, arXiv:1704.06440。

[114] E. Wei,D. Wicke,D. Freelan 和 S. Luke,“多智能体软件 Q-learning”,载于Proc. AAAI Spring Symp. Ser., 2018 年,第 1-7 页。

[115] R. Lowe,Y. Wu,A. Tamar,J. Harb,P. Abbeel 和 I. Mordatch,“针对混合合作竞争环境的多智能体行动者评论家”,载于第 31 届国际神经信息过程系统会议论文集, 2017 年,第 6382-6393 页。

[116] T. Haarnoja,A. Zhou,P. Abbeel 和 S. Levine,“软演员-评论家:具有随机演员的离线策略最大熵深度强化学习”,载于Proc. Int. Conf. Mach. Learn., 2018 年,第 1861-1870 页。

[117] A.Kumar,J.Fu,G.Tucker 和 S.Levine,“通过引导错误减少来稳定离策略 Q 学习”,载于Proc. Adv. Neural Inf. Process. Syst., 2019,第 1-11 页。

[118] D. Silver,RS Sutton 和 M. Müller,“基于样本的学习和搜索与永久和暂时记忆”,第 25 届国际会议论文集 Mach. Learn. (ICML), 2008 年,第 968-975 页。

[119] R.I. Brafman 和 M.Tennenholtz,“R-MAX 一种用于近似最优强化学习的通用多项式时间算法”,J.Mach.Learn. Res.,第 3 卷,第 213-231 页,2002 年 10 月。



[120] AS Polydoros 和 L. Nalpantidis, “基于模型的强化学习调查:在机器人技术上的应用”, J. Intell. Robot. Syst. Theory Appl.,第 86 卷,第 2 期,第 153-173 页,2017 年 5 月。

[121] L. Kaiser 等人, “基于模型的 Atari 强化学习”,2019 年, arXiv:1903.00374。

[122] S. Gu,T. Lillicrap,J. Sutskever 和 S. Levine,“基于模型加速的持续深度 Q 学习”,载于Proc. Int. Conf. Mach. 学习, 2016 年,第 2829-2838 页。

[123] M. Janner,J. Fu,M. Zhang 和 S. Levine,“何时信任你的模型:基于模型的策略优化”,载于Proc. Adv. Neural Inf. Process. Syst.,第 32 卷。美国纽约州红钩:Curran Associates,2019 年,第 1-12 页。

[124] D. Hafner,T. Lillicrap,J. Ba 和 M. Norouzi,“控制梦境:通过潜在想象进行学习行为”,载于Proc. Int. Conf. Learn. Represent., 2019 年,第 1-20 页。

[125] AS Vezhnevets 等人, “用于分层强化学习的封建网络”,载于Proc. Int. Conf. Mach. Learn., 2017 年,第 3540-3549 页。

[126] RS Sutton,D. Precup 和 S. Singh,“MDP 与半 MDP 之间:强化学习中时间抽象的框架”,载于Proc. Int. Conf. Mach. Learn., 2015 年,第 1-12 页。

[127] TD Kulkarni,K. Narasimhan,A. Saeedi 和 J. Tenenbaum,“分层深度强化学习:整合时间抽象和内在动机”, Proc. NIPS, 2016 年,第 1-9 页。

[128] P.-L. Bacon,J. Harb 和 D. Precup,“期权-批评架构”,载于AAAI Conf. Artif. Intell. 会议录, 2017 年,第 31 卷,第 1 期,第 1-9 页。

[129] M. Andrychowicz 等人, “后见之明经验重播”,载于第 31 届国际神经信息过程系统会议论文集, 2017 年,第 5055-5065 页。

[130] B. Eysenbach,A. Gupta,J. Ibarz 和 S. Levine,“你所需要的就是多样性:学习没有奖励功能的技能”,载于Proc. Int. Conf. Mach. Learn., 2018 年,第 1-22 页。

[131] M L Littman,“马尔可夫博弈作为多智能体强化学习的框架”,《机器学习论文集》1994 年版。

[132] L. Bu, soniu,R. Babu ka 和 B. De Schutter,“多智能体强化学习:概述”,载于《多智能体系统与应用创新1》。德国柏林:Springer,2010 年,第 183-221 页。

[133] J. Foerster,G. Farquhar,T. Afouras,N. Nardelli 和 S. Whiteson,“反事实多主体策略梯度”,载于Proc. AAAI 会议阿蒂夫。英特尔, 2018 年,卷。32。没有。1,页。1-9。

[134] P. Sunehag 等人, “基于团队奖励的合作多智能体学习的价值分解网络”,第 17 届国际会议论文集 Auton. Agents MultiAgent Syst., 2018 年,第 2085-2087 页。

[135] T. Rashid,M. Samvelyan,C. Schroeder,G. Farquhar,J. Foerster 和 S. Whiteson,“QMIX:深度多智能体强化学习的单调值函数分解”,载于Proc. Int. Conf. Mach. Learn., 2018 年,第 4295-4304 页。

[136] S. Schaal,“模仿学习是通向人形机器人的途径吗?”,《认知科学趋势》,第 3 卷,第 6 期,1999 年,第 233-242 页。

[137] P. Abbeel 和 AY Ng,“通过逆向强化学习实现学徒学习”,载于第 21 届国际机器学习会议论文集 (ICML), 2004 年,第 1-8 页。

[138] E. Klein,M. Geist,B. Piot 和 O. Pietquin,“通过结构化分类实现逆向强化学习”, Proc. NIPS, 2012 年,第 1-9 页。

[139] ND Ratliff,JA Bagnell 和 MA Zinkevich,“最大裕度规划”,载于第 23 届国际机器学习会议论文集 (ICML), 2006 年,第 729-736 页。

[140] C. Finn,P. Christiano,P. Abbeel 和 S. Levine,“生成对抗网络、逆向强化学习和基于能量的模型之间的联系”,2016 年, arXiv:1611.03852。

[141] J. Fu,K. Luo 和 S. Levine,“通过对抗性逆强化学习学习稳健奖励”,载于Proc. Int. Conf. Learn. Represent., 2018 年,第 1-15 页。

[142] T. Osa,J. Pajarinen,G. Neumann,JA Bagnell,P. Abbeel 和 J. Peters,“从算法角度看模仿学习”, Found. Trends Mach. Learn.,第 7 卷,第 1-2 期,第 1-179 页,2018 年。

[143] S. Ross,G. Gordon 和 D. Bagnell,“将模仿学习和结构化预测简化为无遗憾在线学习”,载于第 14 届国际人工智能统计会议论文集, JMLR 研讨会和会议论文集,2011 年,第 627-635 页。

[144] B. Kim,A.-M. Farahmand,J. Pineau 和 D. Precup,“从有限的演示中学习”,载于Proc. Adv. Neural Inf. Process. Syst., 2013 年,第 2859-2867 页。

[145] A. Hussein,MM Gaber,E. Elyan 和 C. Jayne,“模仿学习:学习方法调查”, ACM Comput. Surv.,第 50 卷,第 2 期,第 1-35 页,2017 年。

[146] J. Ho 和 S. Ermon,“生成对抗模仿学习”,载于第 30 届国际神经信息过程系统会议论文集, 2016 年,第 4572-4580 页。

[147] L. Song,D. Li,X. Wang 和 X. Xu,“带截断梯度的 AdaBoost 最大熵深度逆强化学习”,信息科学,第 602 卷,第 328-350 页,2022 年 7 月。

[148] V. Ricardo 和 D. Youssef,“元学习的视角和调查”, Artif. Intell. Rev.,第 18 卷,第 77-95 页,2001 年 9 月。

[149] N. Schweighofer 和 D. Doya,《强化学习中的元学习》,神经网络,第 16 卷,第 1 期,第 5-9 页,2003 年 1 月。

[150] K. Rakelly,A. Zhou,C. Finn,S. Levine 和 D. Quillen,“通过概率上下文变量实现高效的离线策略元强化学习”,载于Proc. Int. Conf. Mach. Learn., 2019 年,第 5331-5340 页。

[151] JX Wang 等, “前额皮质作为元强化学习系统”,《自然神经科学》,第 21 卷,第 6 期,第 860-868 页,2018 年。

[152] A. Gupta,R. Mendonca,Y. Liu,P. Abbeel 和 S. Levine,“结构化探索策略的元强化学习”,载于第 32 届国际神经信息过程系统会议论文集, 2018 年,第 5307-5316 页。

[153] A. Nagaband 等人,“通过元强化学习学习适应动态的现实世界环境”,2018 年, arXiv:1803.11347。

[154] S. Levine,A. Kumar,G. Tucker 和 J. Fu,“离线强化学习:教程、评论和开放问题的观点”,2020 年, arXiv:2005.01643。

[155] C.-A. Cheng,X. Yan 和 B. Boots,“策略梯度方法中用于减少方差的轨迹控制变量”,载于Proc. Conf. Robot Learn., 2020 年,第 1379-1394 页。

[156] A.Kumar,A.Zhou,G.Tucker 和 S.Levine,“用于离线强化学习的保守 Q 学习”,2020 年, arXiv:2006.04779。

[157] G. Kahn,P. Abbeel 和 S. Levine,“BADGR:一种基于自主监督学习的导航系统”, IEEE Robot. Autom. Lett.,第 6 卷,第 2 期,第 1312-1319 页,2021 年 4 月。

[158] N. Rhinehart,R. McAllister 和 S. Levine,“用于灵活推理、规划和控制的深度模仿模型”,载于Proc. Int. Conf. Learn. Represent., 2019 年,第 1-19 页。

[159] SJ Pan 和 Q. Yang,“迁移学习调查”, IEEE Trans. Knowl. Data Eng.,第 22 卷,第 10 期,第 1345-1359 页,2009 年 1 月。

[160] ME Taylor 和 P. Stone,“强化学习领域的迁移学习:一项调查”, J. Mach. Learn. Res.,第 10 卷,第 7 期,第 1633-1685 页,2009 年。

[161] Z. Zhu, K. Lin, AK Jain 和 J. Zhou,“深度强化学习中的迁移学习:一项调查”,2020 年, arXiv:2009.07888。

[162] Y. Lan,X. Xu,Q. Fang,Y. Zeng,X. Liu 和 X. Zhang,“通过自动修剪策略树提取元知识进行迁移强化学习”, Knowl.-Based Syst.,第 242 卷,2022 年 4 月,文章编号 108221。

[163] Y. Zhang, J. Ren, J. Li, Q. Fang 和 X. Xu,“具有可解释和可迁移领域规则的深度 Q 学习”, Proc. Int. Conf. Intell. Comput. Cham,瑞士:Springer,2021 年,第 259-273 页。



王旭于 2015 年获得西安电子科技大学软件工程学士学位,目前正在攻读计算机科学与技术博士学位。

他的研究兴趣包括自动驾驶以及深度强化学习。



王森于 2016 年获得南京东南大学电气工程与自动化学士学位。他目前正在西安电子科技大学计算机科学与技术学院攻读硕士学位。

他目前的研究兴趣包括深度强化学习。



梁星星于 2014 年获得国防科技大学系统工程学院学士学位,2016 年获得国防科技大学信息系统工程科学与技术实验室硕士学位,目前正在系统工程学院攻读博士学位。

他的研究兴趣包括深度强化和战争游戏的多智能体系统。



徐鑫 (IEEE 高级会员)于 1996 年获得国防科技大学自动控制系统电气工程学士学位,2002 年获得国防科技大学机电一体化与自动化学院控制科学与工程博士学位。

他目前是该院的正教授  
国防科技大学智能科学与技术学院。



赵大为于2018年获得湖南长沙国防科技大学控制科学与工程博士学位。

他目前是  
国防科技国家创新研究院  
ogy,北京,中国。他的研究兴趣包括计算机视觉、机器学习和自动驾驶汽车。



戴斌于 1998 年获得国防科技大学控制科学与工程博士学位。

他目前是中国北京国防科技创新研究院的教授。他的研究兴趣包括模式识别、数据挖掘和自动驾驶汽车。



黄金才,国防科技大学教授,信息系统工程技术重点实验室研究员,主要研究方向为通用人工智能、深度强化学习、多智能体系统等。



苗启光 (IEEE高级会员)于2005年12月获得西安电子科技大学计算机应用技术博士学位。

西安电子科技大学计算机学院教授、博士生导师,在国内外重要期刊、会议上发表论文百余篇,研究方向为机器学习、智能图像处理、恶意软件行为分析与理解等。