文章编号:1673-3819(2018)06-0068-05

强化学习研究综述

马骋乾, 谢　伟, 孙伟杰
(国防科技大学信息通信学院, 湖北 武汉　430019)

摘　要:强化学习是机器学习领域内的研究热点,主要用来实现决策优化。首先介绍了强化学习的基本原理和经典算法,包括基于值函数的强化学习算法和基于直接策略搜索的强化学习算法;然后针对强化学习目前受关注较多的 3 个方向:深度强化学习、元强化学习和逆向强化学习分别进行阐述。最后总结了强化学习目前已有的应用和未来可能发展的方向。

关键词:强化学习; 深度强化学习; 元强化学习; 逆向强化学习; 决策优化
中图分类号:TP181　　**文献标志码**:A　　**DOI**:10.3969/j.issn.1673-3819.2018.06.015

Research on Reinforcement Learning Technology: A Review

MA Cheng-qian, XIE Wei, SUN Wei-jie
(National University of Defense Technology, Wuhan 430019, China)

Abstract: Reinforcement learning is a research hotspot in the field of machine learning. It aims to solve problems of decision or optimization. This paper systematically introduces basic principles and classical reinforcement learning algorithms, including value function based reinforcement learning algorithms and direct policy search based reinforcement learning. Then three directions including deep reinforcement learning, meta reinforcement learning, inverse reinforcement learning are described. Finally, existing application and development directions of reinforcement learning are summarized.

Key words: reinforcement learning; deep reinforcement learning; meta reinforcement learning; inverse reinforcement learning; decision and optimization

根据不同的反馈方式,机器学习可以分为监督学习、非监督学习、强化学习三大类[1]。其中监督学习近年来相关研究较多且主要集中在深度学习领域,深度学习利用大量的有标签训练数据对神经网络进行训练,使得神经网络具备某些特定的能力,如分类、回归等,目前已经在计算机视觉、自然语言处理、语音识别等方面取得很好的效果[2]。但现实中很多问题无法提供大量的有标签数据,如机器人路径规划、自主驾驶、玩游戏等,这些涉及决策优化以及空间搜索的问题,深度学习并不擅长,但强化学习却可以有效地解决这些问题,因此,近年来关于强化学习的研究越来越受到重视。

本文对国内外强化学习的现状进行研究,首先解释强化学习的基本原理和主要算法;其次对目前强化学习 3 个重点研究方向:深度强化学习、元强化学习和逆向强化学习分别进行综述;最后介绍强化学习的 3 个经典应用和未来研究方向。

1　强化学习

1.1　强化学习基本原理

强化学习的基本思想是智能体(Agent)在与环境交互的过程中根据环境反馈得到的奖励不断调整自身的策略以实现最佳决策,主要用来解决决策优化类的问题。其基本要素有策略(Policy)、奖赏函数(Reward Function)、值函数(Value Function)、环境模型(Environment)[3],学习过程可以描述为如图 1 所示的马尔科夫决策过程。

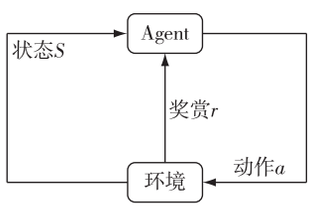

图 1　强化学习基本学习模型

首先智能体感知当前状态 S_t,从动作空间 A 中选择动作 a_t 执行;环境根据智能体做出的动作来反馈相应的奖励 r_{t+1},并转移到新的状态 S_{t+1},智能体根据得到的奖励来调整自身的策略并针对新的状态做出新的决策。强化学习的目标是找到一个最优策略 π^*,使得智能体在任意状态和任意时间步骤下,都能够获得最大的长期累积奖赏:

$$\pi^* = \mathrm{argmax}_\pi E_\pi\left\{\sum_{k=0}^{\infty}\gamma^k r_{t+k} \mid S_t = S\right\}, \forall S \in S, \forall t \geqslant 0.$$

其中 π 表示智能体的某个策略,$\gamma \in [0,1]$ 为折扣率,k 为未来时间步骤,S 为状态空间。

收稿日期: 2018-06-26　　**修回日期**: 2018-07-23

作者简介: 马骋乾(1996-),男,山西临汾人,硕士研究生,研究方向为数据链系统和强化学习。
谢　伟(1974-),男,副教授。

1.2　强化学习算法

强化学习的各类算法根据不同的特征具有多种分类方式，如根据模型是否已知可以分为模型已知(Model Based)和模型未知(Model Free)两类；根据算法更新的方式可以分为单步更新和回合制更新两类；根据动作选择方式可以分为以值为基础(Value Based)的强化学习方式和以策略为基础(Policy Based)的强化学习方式；根据学习策略和执行策略是否为同一策略可以分为同策略(On Policy)学习和异策略(Off Policy)学习；根据参数化方式的不同可以分为基于值函数的强化学习方法和基于直接策略搜索的强化学习方法。本文从参数化方式的角度来阐述基本的强化学习算法。

1.2.1　基于值函数的强化学习方法

基于值函数的强化学习方法通过评估值函数，并根据值的大小来选择相应的动作，主要包括动态规划(Dynamic Programming)、蒙特卡洛(Monte Carlo)、时间差分(Temporal Difference)、值函数逼近(Value Function Approximation)四类[4]。在强化学习模型已知的情况下，选择动态规划法，在策略迭代和值迭代的过程中利用值函数来评估和改进策略。现实中大部分问题的模型是未知的。在模型未知的情况下，我们可以通过蒙特卡洛法利用部分随机样本的期望来估计整体模型的期望，在计算值函数时，蒙特卡洛法利用经验平均来代替随机变量的期望。

蒙特卡洛法虽然解决了模型未知的问题，但更新方式是回合制，学习效率很低。Sutton 等人提出了采用时间差分法(TD)来改善这个问题[3]。时间差分法采用自举(Bootstrapping)方法，在回合学习过程中利用后继状态的值函数来估计当前值函数，使得智能体能够实现单步更新或多步更新，从而极大地提高了学习效率，目前大部分的强化学习研究都基于时间差分方法，如 Q 学习、Sarsa 等相关算法[3]。

动态规划、蒙特卡洛、时间差分三种方法应用的同一前提是状态空间和动作空间都必须离散，且状态空间和动作空间不能过大。当状态空间维数很大，或者为连续空间时，使用值函数方法会带来维数爆炸的问题。针对维数很大或连续空间的问题，可以使用函数逼近的方式来表示值函数，然后再利用策略迭代或值迭代方法来构建强化学习算法。

1.2.2　基于直接策略搜索的强化学习方法

直接策略搜索方法是将策略进行参数化，优化参数使得策略的累计回报期望最大。与值函数参数化方法相比，策略参数化更简单、具有更好的收敛性且能较好地解决连续动作选取问题，主要包括经典策略梯度[5]、置信域策略优化[6](Trust Region Policy Optimization，TRPO)、确定性策略[7]搜索三类。

经典策略梯度通过计算策略期望总奖赏关于策略参数的梯度来更新策略参数，通过多次迭代后最终收敛得到最优策略[7]。在进行策略参数化时，一般通过使用神经网络来实现，在不断试验的过程中，高回报路径的概率会逐渐增大，低回报路径的概率则会逐渐减小。策略梯度的参数更新方程式为

$$\theta_{new} = \theta_{old} + \alpha \nabla_{\theta} J$$

其中，α 为更新步长，J 为奖赏函数。

经典策略梯度最大的问题是选取合适的更新步长非常困难，而步长选取是否合适又直接影响学习的效果，不合适的步长都导致策略越学越差，最终崩溃。为了解决更新步长的选取问题，John Schulman 等人提出了 TRPO 方法[6]。TRPO 将新的策略所对应的奖励函数分解为旧策略所对应的奖励函数和其他项两个部分，只要新策略中的其他项满足大于等于零，便可以保证新策略所对应的奖励函数单调不减，策略就不会变差。

经典策略梯度和 TRPO 采用的均是随机策略，相同的状态选取的动作可能不一样，这使得算法模型要达到收敛需要相对较多的试验数据。为了提高算法效率，Silver 等人提出了确定性策略方法[7]。确定性策略利用异策略学习方式，执行策略采用随机策略来保证探索性，为了使状态对应的动作唯一，评估策略采取确定性策略，也称 AC(Actor-Critic)方法[3]。这种方式所需要的采样数据较少，且能够实现单步更新，算法性能有较大提升。

2　强化学习的主要研究方向

2.1　深度强化学习

传统强化学习在模型较为简单的场景取得了较好的效果，但现实中的问题往往都比较复杂，状态空间和动作空间维数很大，此时传统的表格型强化学习不再适用。近年来，随着深度学习的兴起，深度学习与强化学习的结合研究也受到了很多关注。谷歌 DeepMind 团队创新性地将具有强大感知能力及表征能力的深度学习与具有决策能力的强化学习相结合，形成了人工智能领域新的研究热点，深度强化学习(Deep Reinforcement Learning，DRL)。

DRL 中最具代表性的算法是由谷歌 DeepMind 团队中 Mnih 等人提出的深度 Q 网络(Deep Q Network，DQN)[8]。在 DQN 算法中，使用深度神经网络来代替 Q 表，能够适用状态空间和动作空间非常复杂的场景，将当前状态值作为神经网络的输入，输出端输出所要

采取的动作。并采用Q学习的方式对神经网络的参数进行更新,利用经验回放机制减小了数据之间的相关性,缩短了训练时间[9]。

DQN算法在进行优化时,每次都会选取下一个状态最大Q值所对应的动作,这会带来过估计的问题,Van Hasselt等人在双Q学习算法[15]的研究基础上提出了深度双Q网络(Deep Double Q-Network,DDQN)算法[10]。在DDQN中有两套不同的参数网络,分别是当前值网络和目标值网络,当前值网络用来选取动作,目标值网络用来对动作做出评估,这样使动作选择和策略评估得以分离,有效降低了Q值过估计的风险。

在DQN取得成功后,DeepMind团队中Timothy等人又将深度神经网络与确定性策略相结合提出深度确定性策略梯度(Deep Deterministic Policy Gradient,DDPG)算法[11]。DDPG使用深度神经网络来表示策略,并使用策略梯度的方式来更新策略。DDPG同时具备AC算法和深度学习的优点,在实际运用中,这种学习方式往往能够带来更加有效的学习过程。

2.2 元强化学习

元学习(Meta Learning)的目标是学会学习,从一系列学习任务中训练一个模型,这个模型在面对新的学习任务时只需要少量的样本便可以实现快速学习。元学习的学习思想是将同一系列任务的内在特征分成两类,一类是通用特征,另一类是灵敏度较高的特征。在试验过程中,首先通过求最小方差的方式得到通用特征,面对新任务时只需要少量样本就可以学习到具体任务中其他灵敏度较高的特征。

深度学习对大数据的过度依赖也在一定程度上制约了深度强化学习的发展,为了实现小样本学习,元强化学习研究开始受到关注。元强化学习(Meta Reinforcement Learning)框架最早由Schmidhuber等人提出[12]。JX等人将深度学习与元强化学习相结合提出深度元强化学习(Deep Meta Reinforcement Learning)概念[13],使用神经网络实现了对强化学习任务的快速学习。

大部分关于强化学习的研究都基于环境不变的假设,但现实世界中的环境常常是不断变化的,强化学习的策略也需要不断学习调整以适应不断变化的环境。Chelsea等人提出一种模型未知的元学习(Model-Agnostic Meta-Learning, MAML)算法[14],这种算法不针对某种特定的模型,采用梯度下降方式进行更新,在面对新任务时,模型微调之后便可以获得较好的泛化性能。Maruan等人将MAML算法和循环神经网络相结合研究,实现了强化学习模型在动态环境中的策略自适应[15]。

元强化学习也被认为是最有可能实现通用人工智能的方式,DeepMind团队在用深度学习复现大脑的导航功能后[16],又利用元强化学习框架探索研究了大脑中的多巴胺在学习过程中所发挥的作用[17],帮助解释了神经科学和心理学的一系列发现,也说明了元强化学习和人类智能存在的紧密联系。

2.3 逆向强化学习

强化学习是求累计奖赏期望最大时的策略,求解过程中的奖赏函数是人为给定的,但在很多复杂的任务中,奖赏函数往往难以直接给定,而奖赏函数的好坏又对学习结果有着非常重要的影响。吴恩达等人提出逆向强化学习来解决该问题[18],专家在完成某项任务时,其决策往往是最优或接近最优,当奖赏函数难以给定时,可以通过从专家示例中来学习奖赏函数。

经典的逆向强化学习算法包括基于学徒学习、最大边际规划、结构化分类和概率模型形式化的方法[19]。Abbeel等人提出基于学徒学习的方法[20],使用函数逼近的方法从专家示例中学习奖赏函数,使得在该奖赏函数下所得的最优策略在专家示例策略附近,主要用来解决退化解和奖赏函数歧义性问题。最大边际规划法的目标是找到使专家示例策略具有比其他策略更大累计奖赏的状态到奖赏的映射,在这个映射下,最优策略能够逼近专家示例策略[21]。Klein等人提出结构化分类方法[22],用分类的思想考虑最优的策略,并将估计的专家期望特征作为奖赏函数,这种做法避免了多次迭代计算过程。最大边际规划法和结构化分类方法往往会产生歧义,很多不同的奖赏函数会导致相同的专家策略。Ziebart等人提出基于最大熵和交叉熵两类概率模型来解决该问题[23]。

经典的逆向强化学习算法不能很好地扩展到状态空间维数很大的系统,与深度学习的结合能够发挥更大的作用。将模型中的状态和动作使用深度神经网络替代,算法可以在大型复杂的系统中取得良好的效果。如基于最大边际法的深度逆向强化学习、基于DQN的深度学徒学习[24]和基于最大熵模型的深度逆向强化学习[25]。

3 强化学习的应用

目前强化学习应用较多的领域有对弈、决策和控制等,本节从这三个方面分别选择一个典型应用进行详细介绍。

3.1 典型深度强化学习AlphaGo

AlphaGo在战胜李世石后开启了深度学习的热潮,而其能够战胜人类的关键技术就是深度强化学习。19×19的围棋状态空间复杂度近似10的172次方[8],传

统的搜索方法只能在局部进行搜索,无法有效提取整个空间的有效特征,容易进入局部最优状态,且当时计算能力较弱,因此无法有效解决围棋这类状态空间庞大的问题。

深度学习采用分层特征提取的方式将搜索的状态空间由棋局缩小到局部态势,最终到单个棋子,大大缩小了搜索空间。同时深层网络能够表示较为复杂的状态空间,这使得深度学习在避免盲目搜索的同时可以扩大搜索范围。采用强化学习中 DDPG 方法建立一个策略网络和一个价值网络,策略网络根据当前的棋局状态来选择下一步棋的位置,价值网络对当前的棋局进行评估,若评估结果为胜的概率大则增大该棋步选择的概率,否则降低选择概率。这种方式大大提高了学习效率和搜索速度,同时辅助以高速的计算能力,最终实现了 AlphaGo 的成功。

3.2　服务链网元部署

服务链是一种借助软件定义网络自动创建一个连接网络服务的虚拟服务链[28],定义了特定顺序的网络功能集合,其网元负责执行处理特定的功能,通常包括虚拟的交换机、虚拟的路由器等,网元的放置位置会影响服务提供和物力资源的使用效率。

实现网元位置的最佳决策,关键在于优化服务链的平均链路延时和服务器的负载。采用强化学习方式进行学习,通过不断与服务环境的接触和外部环境所反馈的奖励信号来调整网元的部署位置。该模型的状态集为网元的实时位置,动作集为网元位置的选择,奖励函数根据节点资源的门限值来设定。然后使用 Q 学习算法对状态集和动作集进行表示,并在不断学习迭代的过程中,增大那些使链路延时和负载变小的网元位置 Q 值,最终收敛于最优值。

3.3　自主驾驶

随着人工智能的深入研究,自主驾驶与我们的生活越来越接近。自主驾驶的实现可以分为两部分:第一部分是通过深度学习感知车辆所处的环境并进行信息收集和分析,使智能体能够获取当前的路况信息。第二部分是通过强化学习选取策略,车辆方向盘的转向等操作都是在连续动作空间内选择,因此采用 DDPG 的方法建立策略网络。起初智能体对车辆可能遇到的一些情况,如行人、车辆或偏离轨道等,采取一些随机动作,然后使用价值网络增加避让行人、车辆等安全动作的值,进而增大该动作的选择概率,降低冲撞行人、偏离轨道、超速等动作的选择概率。在不断训练的过程中智能体逐渐学会避让行人、车辆,并选择合适的速度在预定轨道内进行驾驶。

4　结束语

强化学习作为当前人工智能领域的热门研究方向之一,已经吸引了越来越多的学者对其进行不断地研究和扩展。虽然强化学习在某些领域已经取得较好的效果,但在国内相关研究目前并不是很多,在通信、军事决策,无人系统控制以及通用人工智能等很多领域仍然具有很大的潜力。相信随着强化学习研究的不断深入,强化学习将会帮助我们解决更多的问题。

参考文献:

[1] 周志华.机器学习[M].北京:清华大学出版社,2015.

[2] Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep learning[M].Cambrige:MIT Press,2016.

[3] Sutton R,Barto A.Reinforcement learning:An introduction[M]. Cambrige:MIT Press,2017.

[4] 郭宪,方勇纯.深入浅出强化学习原理入门[M].北京:电子工业出版社,2018.

[5] Sutton R S,Mcallester D A,Singh S P, et al. Policy gradient methods for reinforcement learning with function approximation [C]. In: Neural Information Processing Systems,2000,pp.1057-1063.

[6] John Schulman,Sergey Levine,Pieter Abbeel,Michel Jordan, and Philipp Moritz. Trust region policy optimization [C]. In: Proceedings of the 31st International Conference on Machine Learning,2015.

[7] Silver,David,Lever,Guy,Heess,Nicolas,Degris,Thomas, Wierstra, Daan, and Riedmiller, Martin. Deterministic policy gradient algorithms[C]. In: Proceedings of the 30st International Conference on Machine Learning,2014.

[8] Mnih,Volodymyr,et al. Playing Atari with deep reinforcement learning [EB/OL]. [2013-10-22] https://arXiv.org/abs/1312.5602.

[9] Tom Schaul, John Quan, et al. Prioritized experience replay[C]. In: International Conference on Learning, 2016.pp.713-734.

[10] Hado van Hasselt, Arthur Guez, David Silver. Deep reinforcement learning with Double Q-learning[C].In: Association for the Advancement of Artificial Intelligence, 2016,pp.453-466.

[11] Timothy P. Lillicrap, J. Hunt, Alexander et al. Continuous control with deep reinforcement learning[C]. In: International Conference on Learning,2016.pp.1120-1139.

[12] Jurgen Schmidhuber, Jieyu Zhao, and Marco Wiering. Simple principles of metalearning[C].In: Technical Report,1996.pp69-96.

[13] JX Wang,Z Kurth-Nelson, et al.Learning to reinforcement

learn [EB/OL]. [2017-06-08] https://arXiv.org/abs/1611.05763.

[14] Chelsa Finn, Pieter Abbeel, Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks [C]. In: Proceedings of the 33st International Conference on Machine Learning, 2017.

[15] Maruan AI-Shedivat, Trapit Bansal, et al. Continuous adaptation via meta-learning in nonstationary an competitive environments[C]. In: International Conference on Learning, 2018, pp1024-1043.

[16] Banino A, Barry C, Uria B, et al. Vector-based navigation using grid-like representations in artificial agents[J]. Nature, 2018.

[17] Jane X.Wang, Zeb Kurth-nelson et al. Prefrontal cortex as a meta-reinforcement learning system [J]. Nature Neuroscience, 2018.

[18] Ng A Y, Russell S J. Algorithms for inverse reinforcement learning [C]. In: Proceedings of the 16st International Conference on Machine Learning, 2000.

[19] 陈希亮,曹雷,等. 深度逆向强化学习研究综述[J].计算机工程与应用,2018,54(5):24-34.

[20] Abbell P, Ng A Y. Apprenticeship learning via inverse reinforcement learning[C]. In: Proceedings of the 20st International Conference on Machine Learning, 2004.

[21] Ratliff N D, Bagnell J A, Zinkevich M A. Maximum margin planning[C]. In: Proceedings of the 22st International Conference on Machine Learning, 2006

[22] Klein E, Geist M, Piot B, et al. Inverse reinforcement learning through structured classification[C]. In: Neural Information Processing Systems, 2012, pp.556-574.

[23] Ziebart B, Maas A, Bagnell et al. Maximum entropy inverse reinforcement learning[C]. In: Association for the Advancement of Artificial Intelligence, 2008, pp. 1005-1026.

[24] Bogdanovic M, Markovikj D, Denil M, et al. Deep Apprenticeship Learning for Playing Video Games[J]. European Judaism, 2015, 39(1):44-48.

[25] Wulfmeier M, Ondruska P, Posner I. Maximum entropy deep inverse reinforcement learning[EB/OL].[2017-05-20]https://arXiv.org/abs/1507.04888.

[26] Daniel R.Jiang, Emmanuel Ekwedike, Han Liu. Feedback-based tree search for reinforcement learning[C]. In: Proceedings of the 34st International Conference on Machine Learning, 2018.

[27] Oriol Vinyals, Timo Ewalds et al. StarCraft Ⅱ: a new challenge for reinforcement learning[EB/OL].[2017-09-10] https://arXiv.org/abs/1708.04782.

[28] 魏亮, 黄韬, 张娇,等. 基于强化学习的服务链映射算法[J]. 通信学报, 2018(1):90-100.