中国科学技术大学
University of Science and Technology of China

《人工智能数学原理与算法》
第 7 章：强化学习

7.1 强化学习介绍

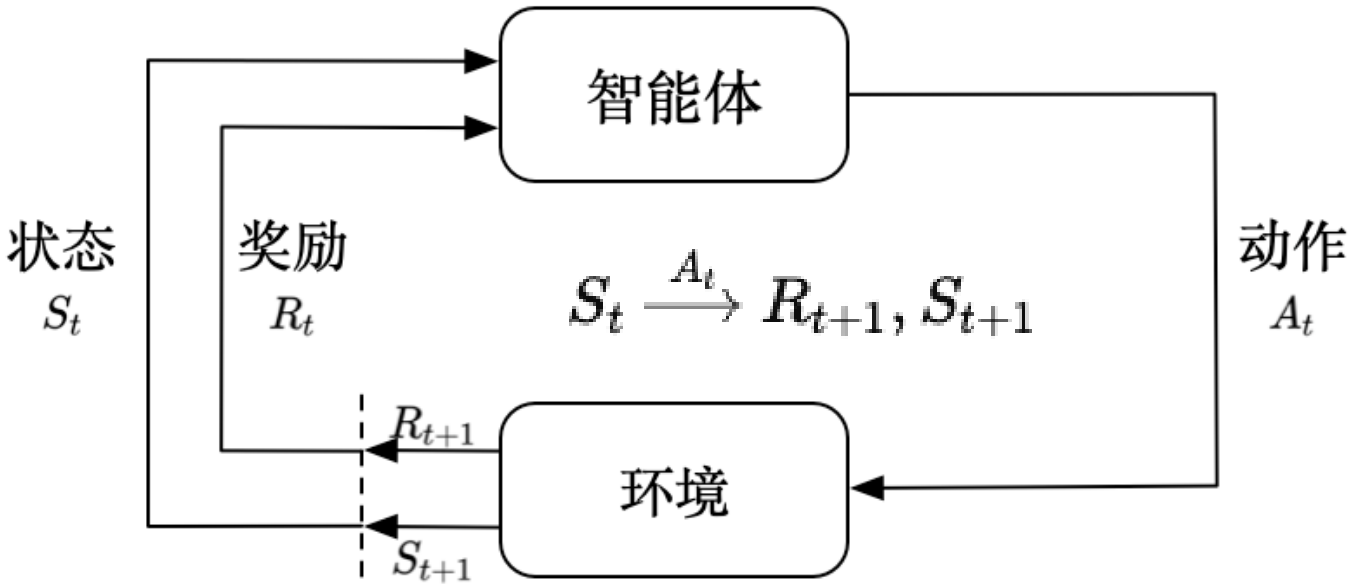
吉建民
jianmin@ustc.edu.cn

目录

目录

什么是强化学习?

- **强化学习**：智能体通过**与环境交互**，**基于奖励反馈进行策略优化**，以**最大化长期累积回报**的机器学习方法



智能体关注的不是单步奖励，而是**长期收益**：

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots$$

其中，γ（折扣因子）决定了未来奖励的权重，$\gamma \in [0,1]$。

- **环境（Environment）**：智能体所处的外部世界，决定状态如何变化并反馈奖励
- **智能体（Agent）**：决策主体，决定在特定状态下如何选择动作
- **奖励（Reward，R_t）**：环境对智能体执行动作 A_t 的即时反馈信号
- **状态（State，S_t）**：环境在时刻 t 的信息描述
- **动作（Action，A_t）**：智能体在状态 S_t 下做出的行动
- **状态转移（State Transition）**：环境根据当前状态 S_t 和动作 A_t 更新到新状态 S_{t+1}，同时给出奖励 R_{t+1}

强化学习的特点

强化学习不同于其他机器学习方法的特点：

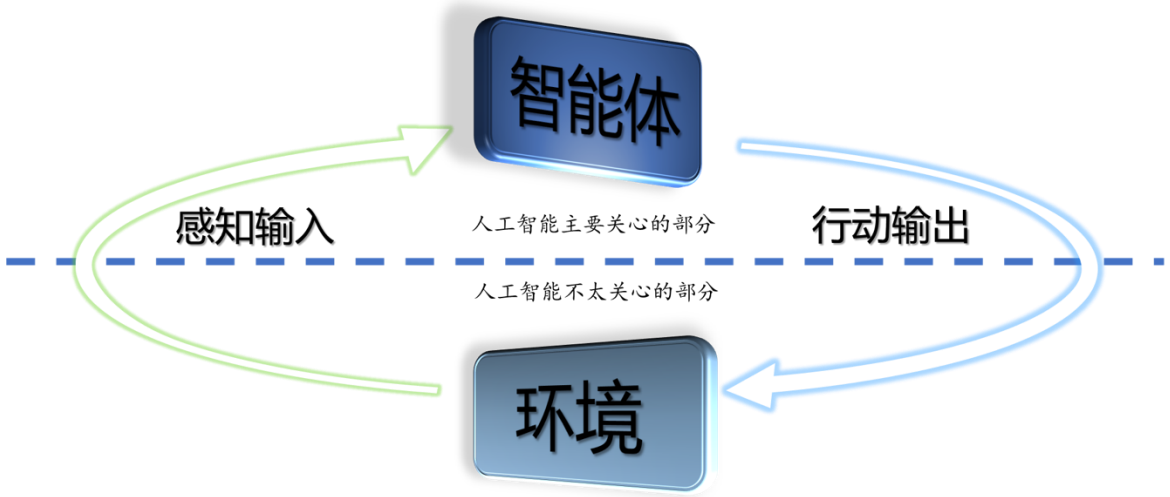
- **试错学习（Trial-and-Error Learning）**：不依赖标注数据，通过不断尝试动作、接收反馈（奖励/惩罚）逐步优化策略
  - 不同于监督学习：不依赖监督的“正确答案”，而是通过环境“奖励/惩罚”反馈自我修正
  - 例如：AlphaGo Zero 不需要人类指导，通过胜率结果优化策略，最终超越人类
- **延迟奖励（Delayed Reward）**：奖励可能滞后于动作，一个动作的好坏可能要经过多个步骤后才能体现
  - 需要智能体学会长期规划，而不是只关注短期利益
  - 例如：围棋中间某一步的价值可能要几步后才能体现
- **序列决策（Sequential Decision Making）**：智能体的决策具有时间依赖性，每个决策不仅影响当前奖励，还会影响未来的状态和奖励
  - 需考虑长期后果，优化整个策略，而非孤立优化每一步收益
  - 例如：围棋中每一步棋都影响整个棋局的发展，从而产生千变万化的局势走向

强化学习的特点

- **长期回报最大化（Maximizing Cumulative Reward）**：强化学习目标是最大化累积奖励，而非单步最优决策
  - 不同于监督学习学到“函数映射”，而是一个**策略（Policy）**，告诉智能体在不同状态下应该执行哪个动作才能长期最优
  - 例如：围棋中“弃子争先”，不能为了局部优势而放弃全局的主动权
- **环境交互（Environment Interaction）**：智能体与环境的交互是**动态且持续**的，每一步动作影响后续状态，形成动态反馈循环
  - 智能体不仅仅发现数据模式，还可以通过动作改变数据分布
  - 例如：推荐系统根据用户点击行为（动作）调整推送内容（新状态）
- **探索与利用权衡（Exploration vs. Exploitation Trade-off）**：探索，尝试未知的动作，获取更多信息；利用，基于已有经验选择当前最优动作
  - 智能体需要在探索新策略（可能更优）和利用已有策略（当前最佳）之间找到平衡
  - 例如：推荐系统如果一直推荐用户最常点击的内容，可能会错过用户的潜在兴趣

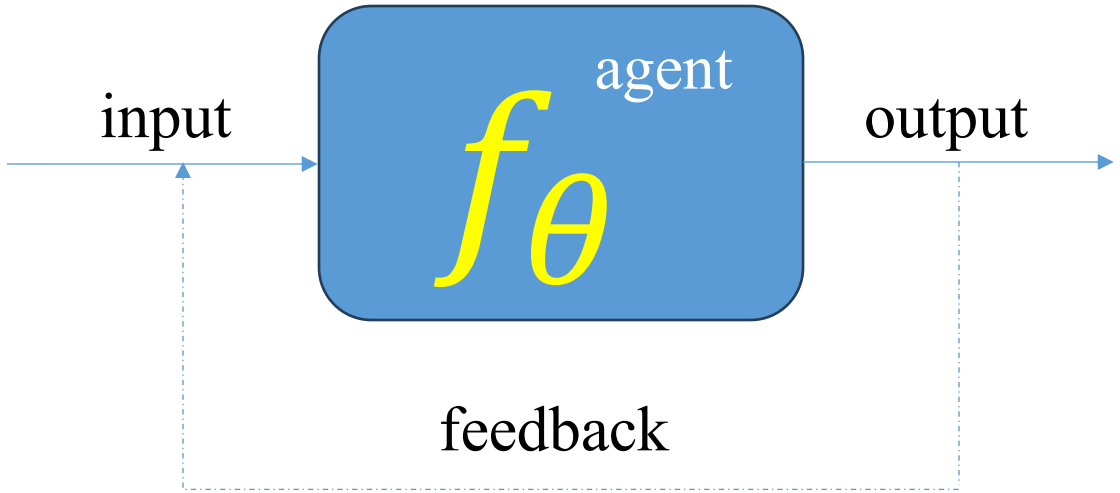
人工智能：从智能的外延到智能体（回顾）

- 每一种智能行为X都对应着一种人工X智能，行为X与环境需要进行交互



	人脸识别	对话问答	围棋象棋	机器翻译	数学证明	……
input	人脸	问题	棋盘状态	语言1句子	题目	……
output	ID	回答	下一步落子	语言2句子	答案	……
feedback	正确与否	正确与否	输赢（多步）	正确与否	正确与否（单/多步）	……

人工智能：从智能的内涵到人工智能四要素与数据形态（回顾）



- 表示：（知识/模型）长什么样？
  机器编码f_θ、input、output、feedback。
- 推理：（知识/模型）怎么用来解决问题？
  给定input，机器实现f_θ计算output。
- 学习：（知识/模型）怎么来的？
  基于数据<input, output, feedback>集，给定f，更新计算θ。

人工智能四要素（“知识”有待商榷）

1. 算法/模型：f（及部分θ）
2. 计算：f_θ/input/output/feedback转换
3. 数据：<input, output, feedback>
4. 知识：θ（及部分f）

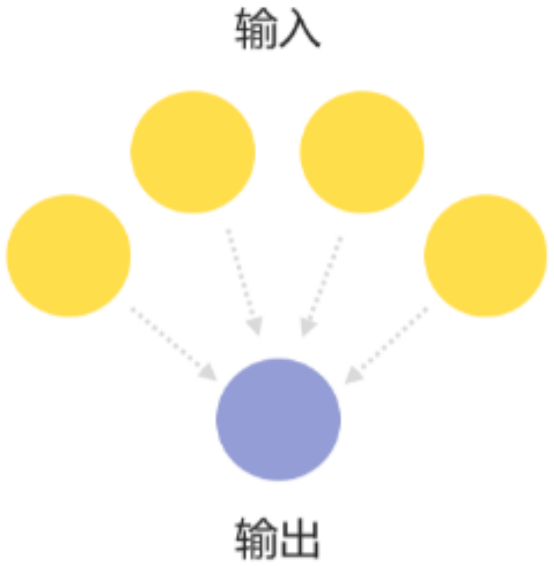
数据：<input, output, feedback>

- 有监督：<input, output, feedback>
- 无监督：<input, output, 空缺>
- 强化：<input, output, 多步>
- 自监督：<input, input*, 正/1>
- ……

强化学习与监督学习、无监督学习的区别

监督学习
(Supervised Learning)

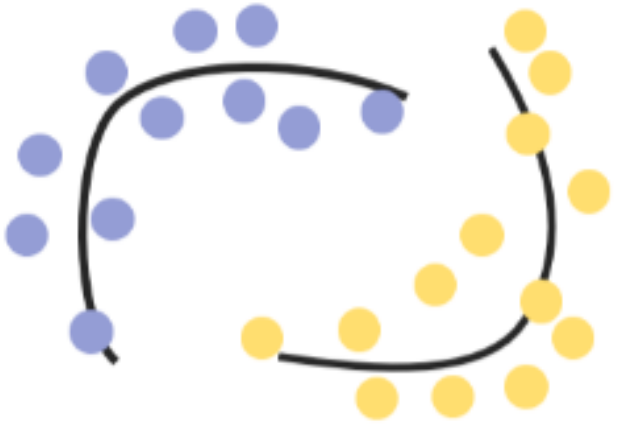
使用训练数据和数据反馈来学习给定输入与给定输出之间的关系（例如，通过价格和假期预测销售量）



- 训练数据有明确标签
- 目标是最小化误差
- 学习 **“函数映射”**

无监督学习
(Unsupervised Learning)

在不提供明确输出变量的情况下，探索输入数据的模式与规律（例如，对客户的人口分布进行分类）



- 训练数据没有标签
- 目标是找到数据的潜在模式
- 发现 **“数据结构”**

强化学习
(Reinforcement Learning)

通过最大化动作所获得的长期回报来学习执行任务（例如，最大化获得的收益以训练投资组合策略）

- 数据由智能体通过试错获取
- 目标是最大化长期收益
- 学习 **“策略”**

目录

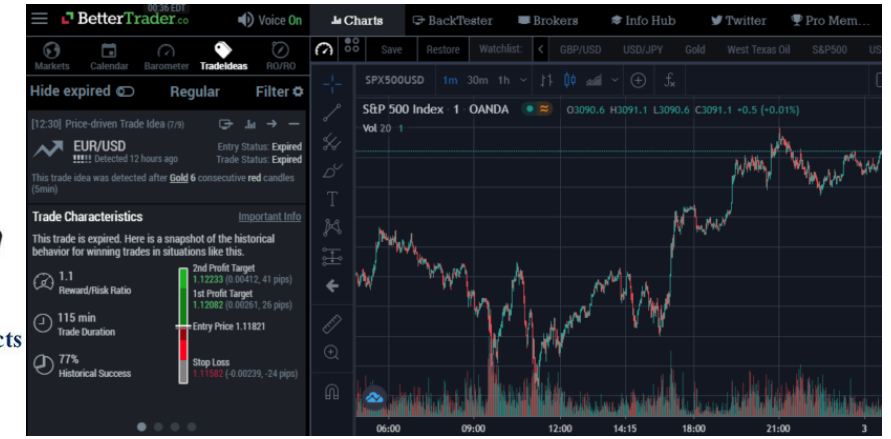
强化学习能做什么?

□ 强化学习已经与人们的生活密切相关

游戏领域

推荐系统/广告投放

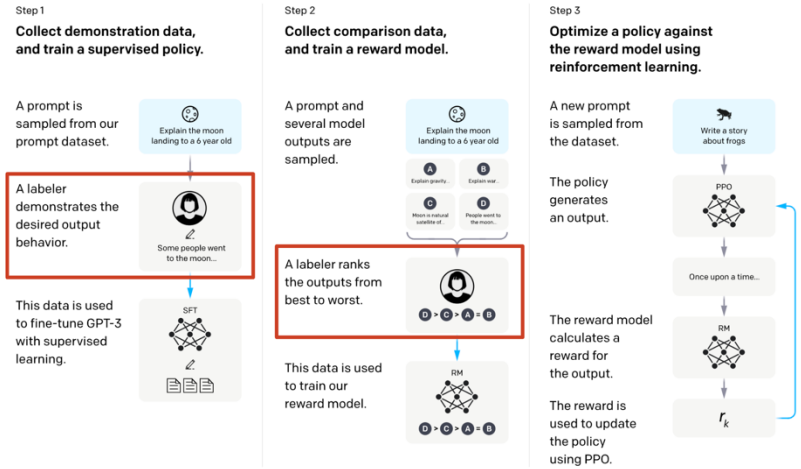
量化投资



"端到端" 自动驾驶

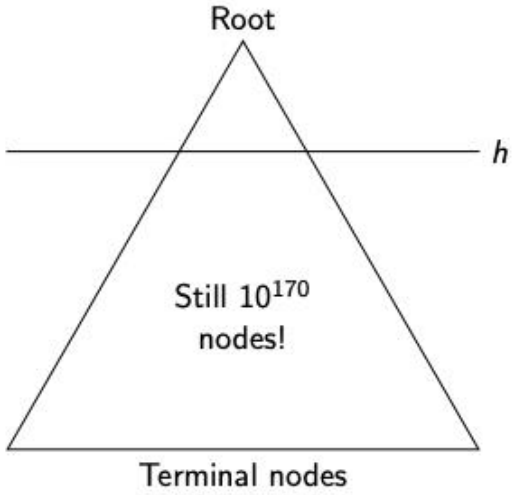
具身智能机器人

大模型训练

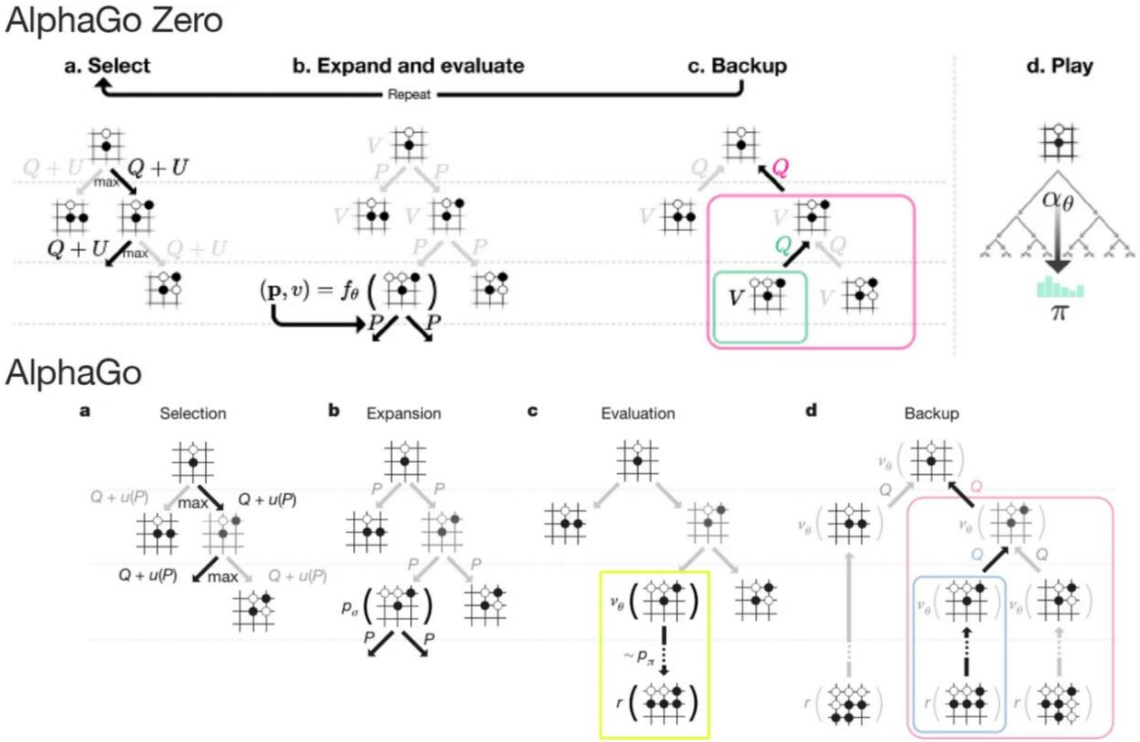


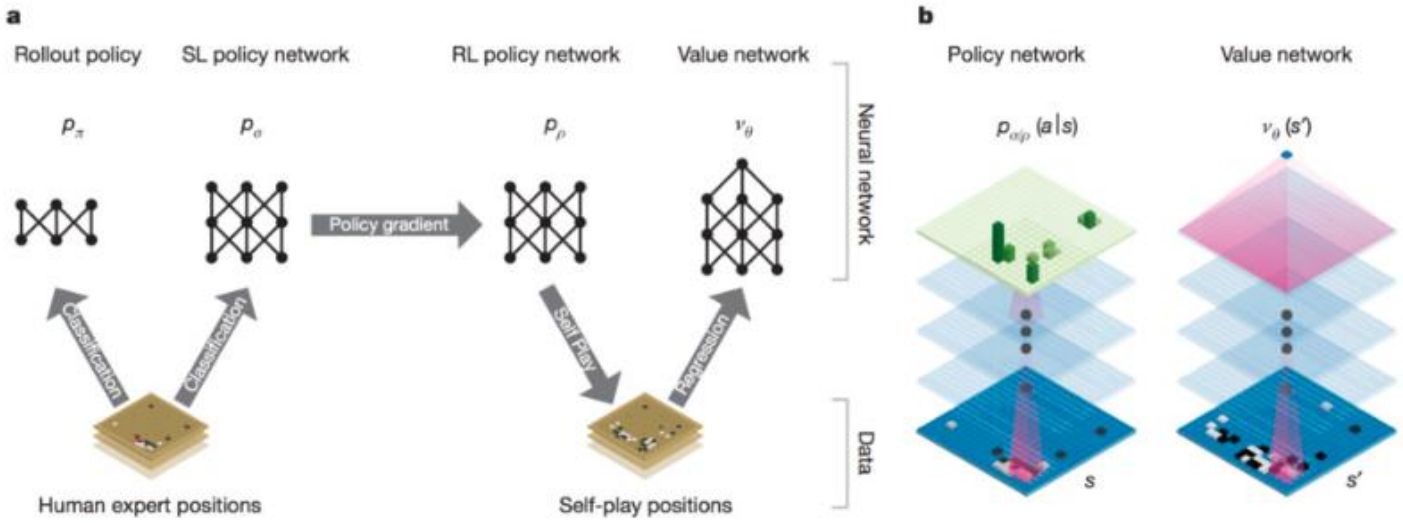
强化学习能做什么？——围棋

- **强化学习在围棋游戏中取得了超人的表现**
  - 围棋复杂度高：
    - 分支因子：平均每一步约有 200~300 个可选位置
    - 博弈树规模：大于 10^{170}
  - **AlphaGo (Zero)**
    - 强化学习方法：Actor-Critic, 策略网，价值网
    - 结合策略网络、价值网络，采用蒙特卡洛树搜索(MCTS)，选择最终落子




Who will win?

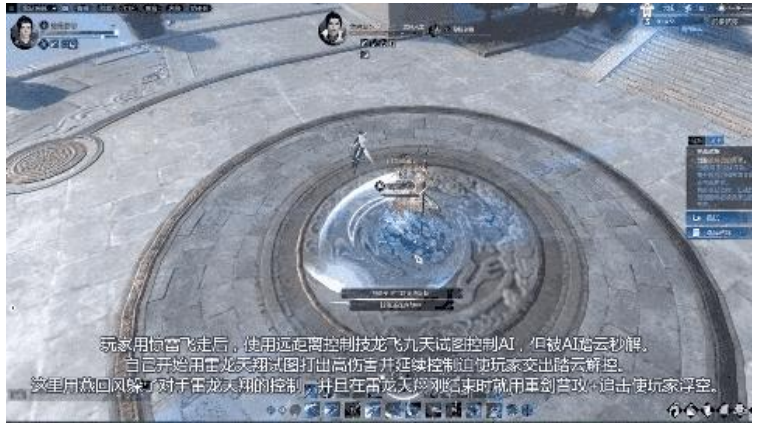




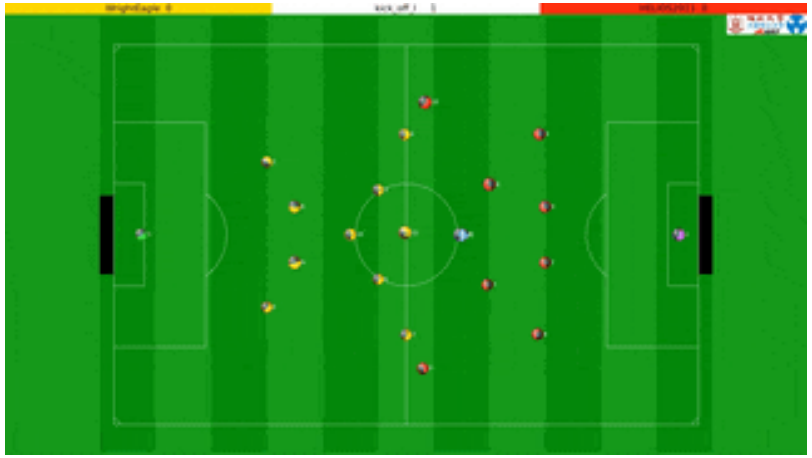
强化学习能做什么？——游戏

□ 强化学习广泛应用于各类游戏

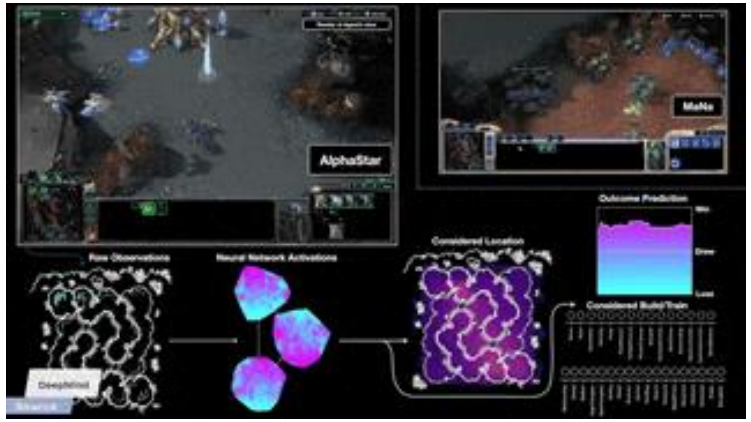
DeepMind RL 用于多款雅达利游戏



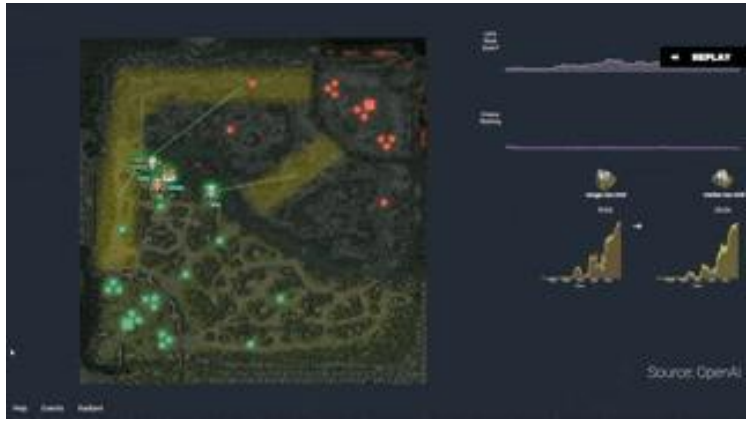
RL 用于网易《逆水寒》



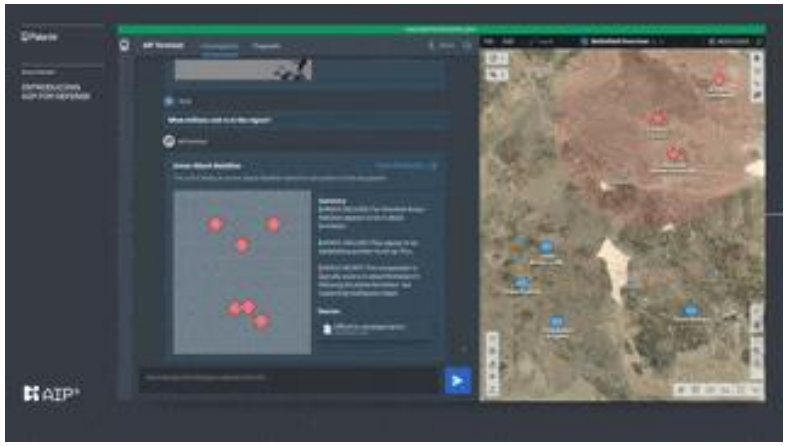
MARL 用于 RoboCup 2D



DeepMind RL 用于星际争霸2



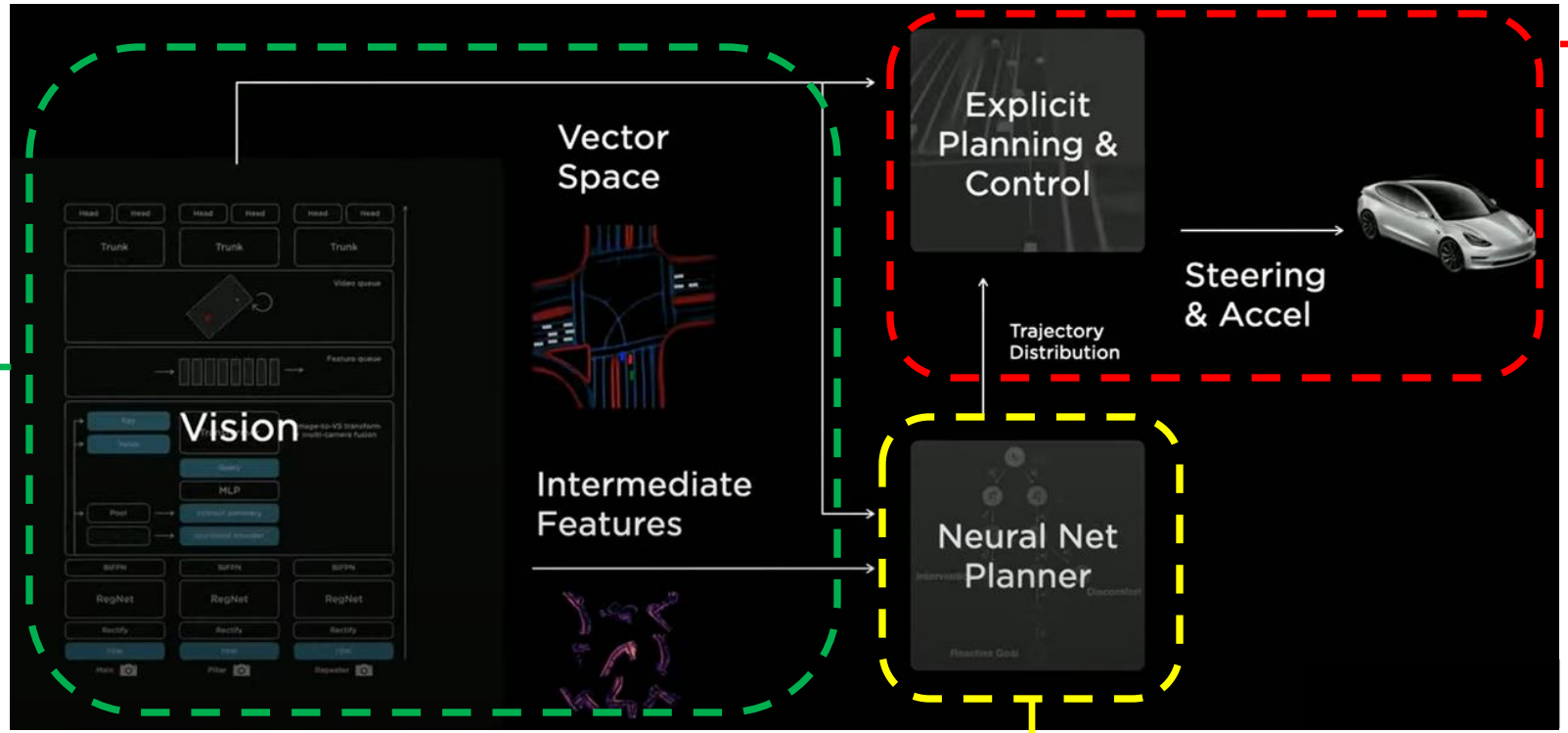
OpenAI RL 用于 Dota 2



Palantir RL 用于作战指挥系统

强化学习能做什么？——自动驾驶

□ 强化学习支撑“端到端”自动驾驶

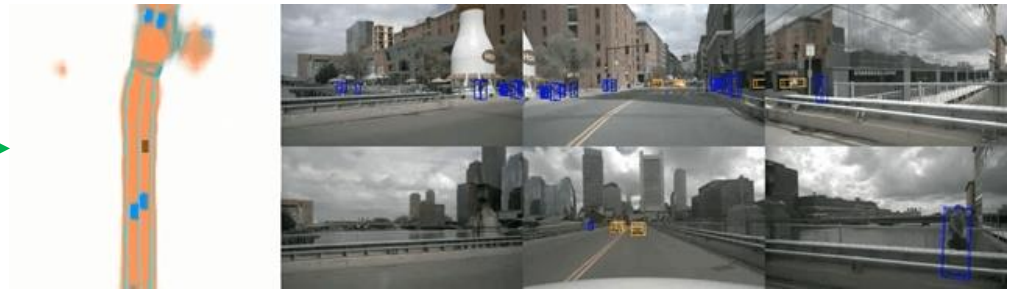


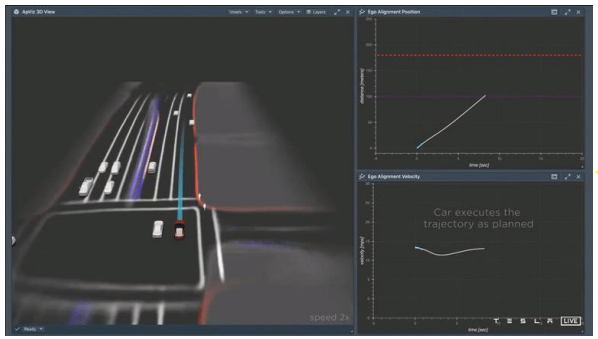
经典规划和控制

控制网：基于小样本模仿学习，以 500 Hz 频率控制底盘

端到端架构适用于各类无人系统，将感知和规划模块分别简化一张神经网络，可以**数据驱动的方式持续训练更新**

感知网：输入传感器信息，训练网络输出BEV下的数字化世界模型





规划网：输入数字化世界模型，训练网络输出可行驶轨迹分布

强化学习能做什么？——机器人

□ 强化学习广泛应用于各类具身智能机器人

宇树人形

宇树机器狗

机器人数据工厂

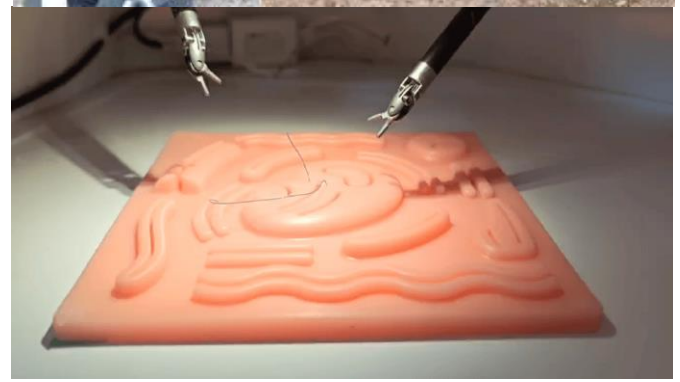
在线RL开冰箱门

小样本IL做饭

开放环境捡垃圾

IL喂饭机器人

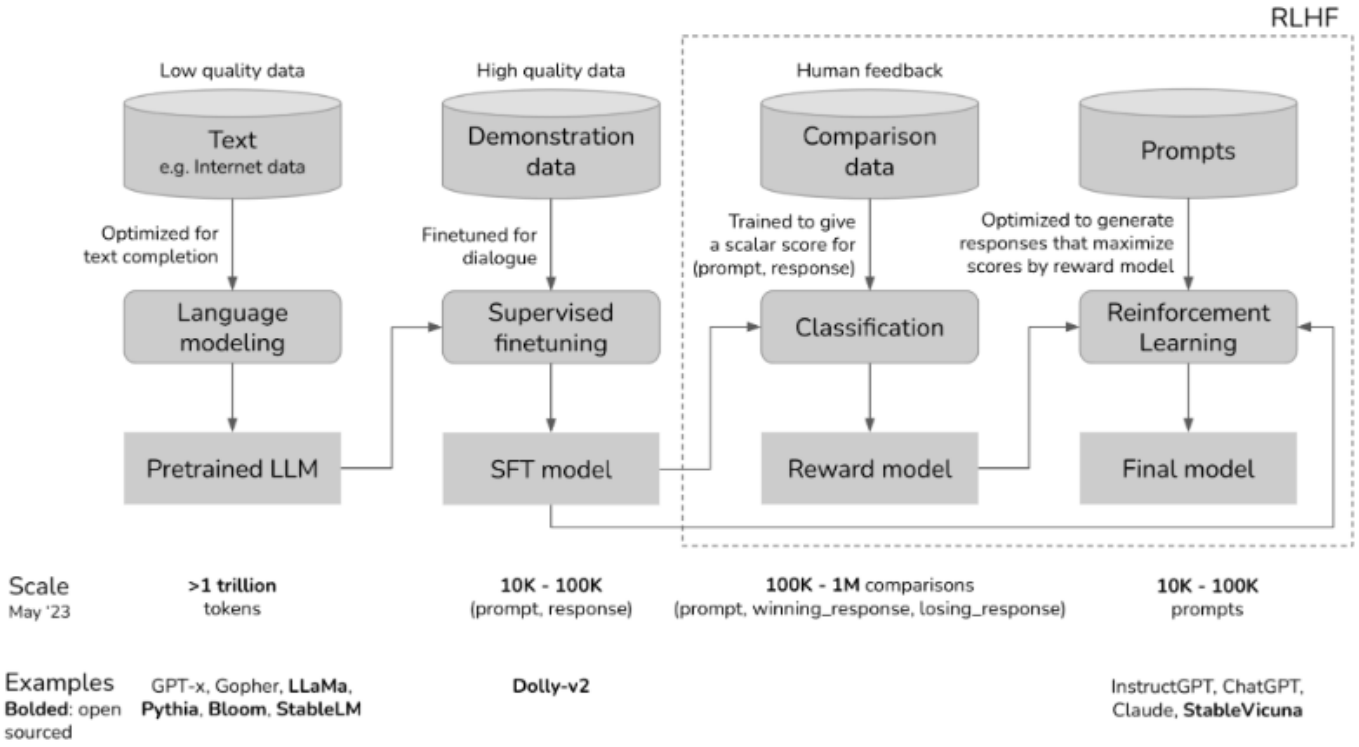
IL无人机抓取



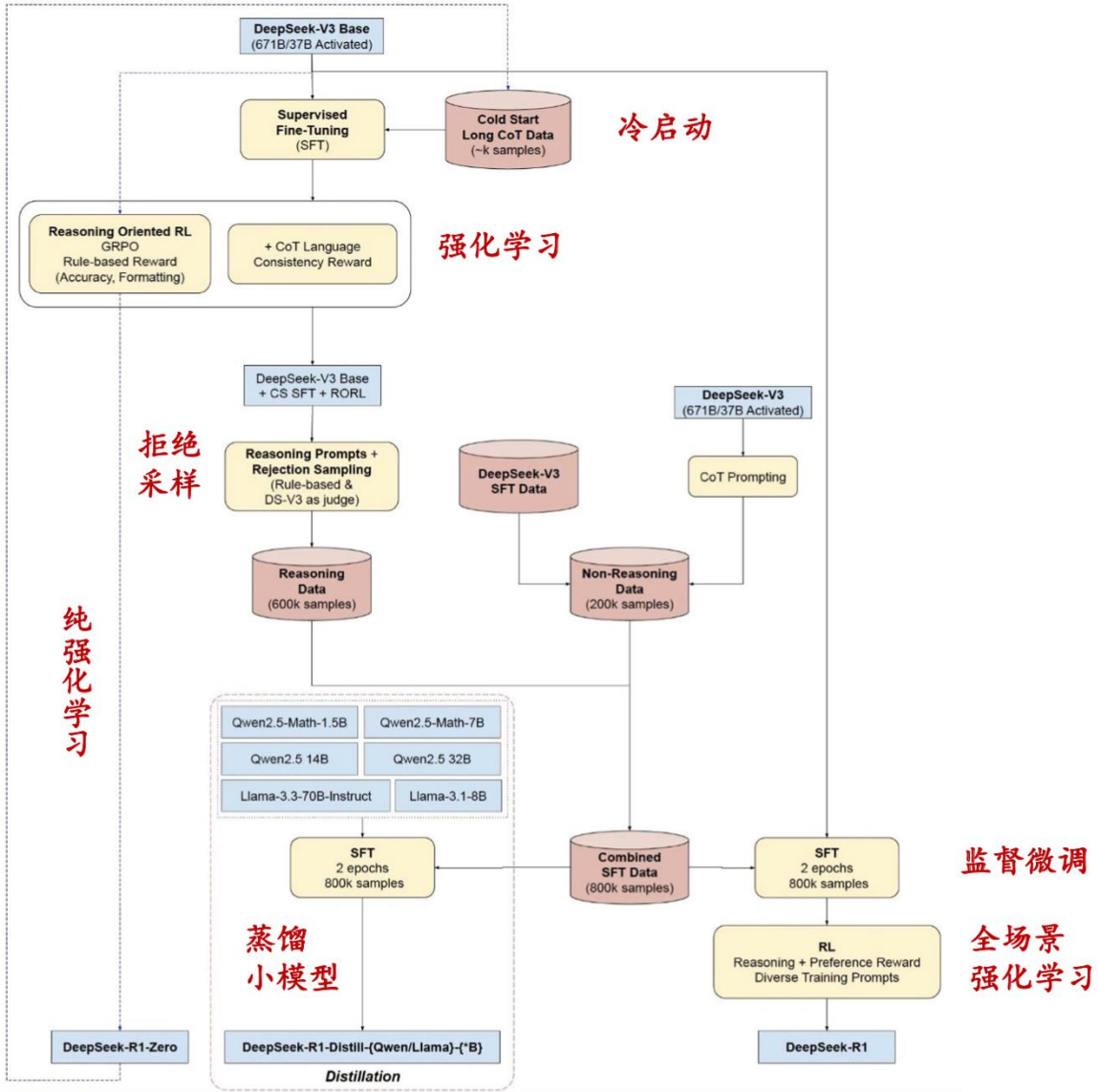
IL手术机器人打结

强化学习能做什么？——推理大模型

- 强化学习支撑大语言模型（LLM）
  - 基于人类反馈的强化学习（RLHF, Reinforcement learning with human feedback）
  - 群体相对策略优化（GRPO, Group Relative Policy Optimization）



RLHF：基于人类反馈训练 Reward Model，再进行 RL 微调

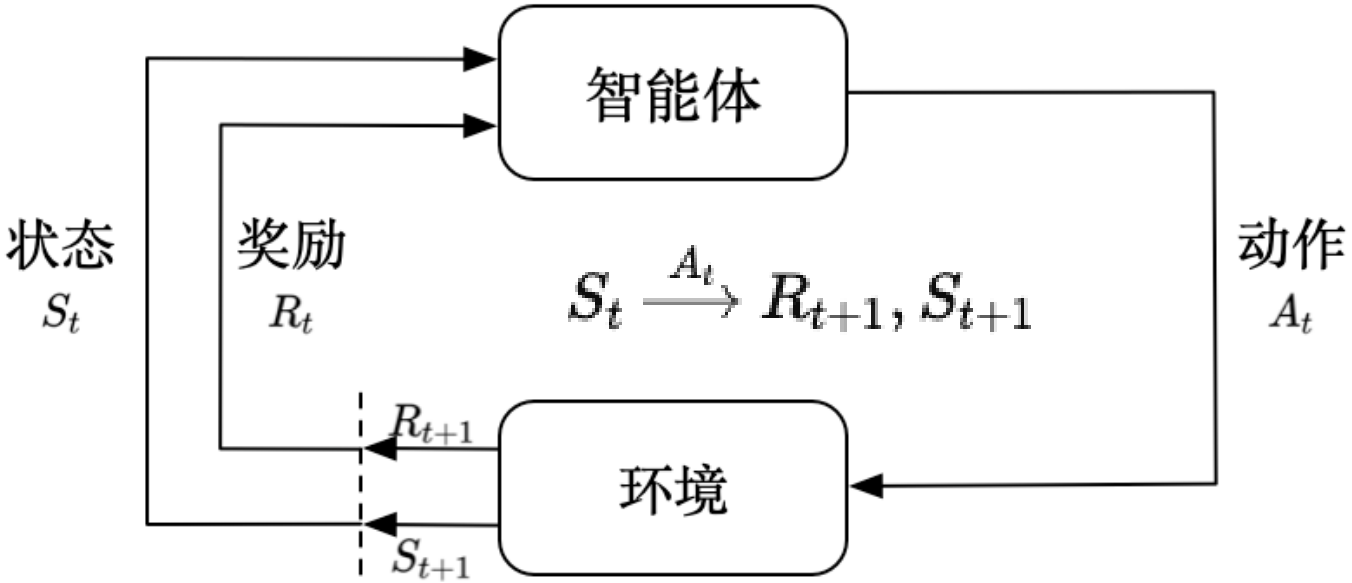


DeepSeek R1 (-Zero) 采用 GRPO，更好的生成思维链（CoT）

目录

强化学习基本概念：智能体与环境

- **智能体（Agent）**：强化学习中决策主体，观察状态并根据策略 $\boldsymbol{\pi}$ 选择动作 $\boldsymbol{A_t}$ 作用于环境
- **环境（Environment）**：外部系统，接收动作并根据状态转移概率 **P** 切换到新状态，并给予奖励 $\boldsymbol{R_t}$



- 智能体策略 $\boldsymbol{\pi}$：

$$\pi(a|s) = P[A_t = a|S_t = s]$$

- 环境状态转移概率 **P**：

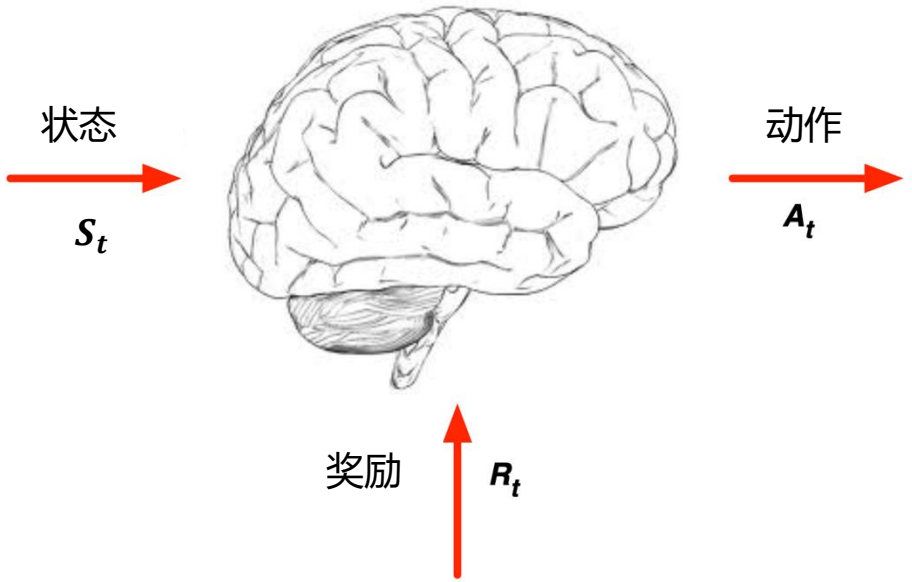
$$P[S_{t+1}|S_1, \dots, S_t]$$

- 环境给予奖励 $\boldsymbol{R_t}$：

$$\mathrm{E}[R_{t+1}|S_t = s, A_t = a]$$

强化学习基本概念：状态、动作、奖励

- **状态（State）**：环境在某时刻的情景描述 S_t
  - 例如：游戏画面、自动驾驶车辆观察到的周围环境、机器人关节角度和传感器读数
- **动作（Action）**：智能体可执行的行为 A_t
  - 例如：按键操作、车辆加减速和方向盘角度、机器人关节运动指令
- **奖励（Reward）**：环境对动作的反馈，用于度量动作的好坏 R_t
  - 例如：游戏得分、车辆是否平稳的到达目的地、机器人是否摔倒



强化学习基本概念：回报与序列决策

- **回报（Return）**：从某时刻起所有折扣奖励的总和 G_t

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

$\gamma \in [0, 1]$ 是折扣因子

  - 延迟奖励：比如下围棋，只有棋局结束才获得正/负奖励，中间过程并无明显提升

- **序列决策（Sequential Decision Making）**：通过选择一系列动作来最大化未来总回报
  - 特点：动作可能带来长期影响（long term consequences），奖励可能是延迟的（reward may be delayed），有时需要牺牲短期奖励，以获得更大的长期回报
  - 示例：
    - 给直升机加油：虽然短期花费时间和资源，但可防止数小时后因燃料不足而坠机
    - 阻挡对手招数：牺牲当下攻击机会，但可能在若干回合后增强胜率

强化学习基本概念：策略、价值函数、模型

- **策略（Policy）**：决定智能体在每个状态选择动作的规则 $\boldsymbol{\pi}$
  - 确定性策略（Deterministic Policy）：$a = \pi(s)$
  - 随机性策略（Stochastic Policy）：$\pi(a|s) = P[A_t = a|S_t = s]$
- **价值函数（Value Function）**：评估状态好坏的指标，衡量长期回报
  - 状态价值：$\mathrm{V}^{\boldsymbol{\pi}}(s) = E_\pi[G_t|S_t = s]$
  - 动作价值：$\mathrm{Q}^{\pi}(s,a) = E_\pi[G_t|S_t = s, A_t = a]$
- **模型（Model）**：预测环境的下一步变化，表示智能体对环境的估计
  - $\boldsymbol{P}$：预测下一时刻的状态

$$P^a_{ss'}(s'|s,a) = P[S_{t+1} = s'|S_t = s, A_t = a]$$

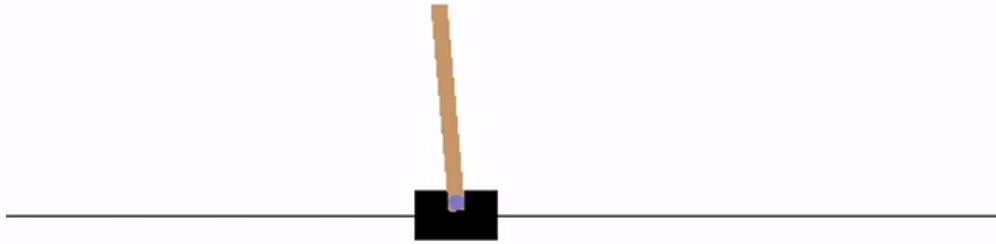
  - $\boldsymbol{R}$：预测下一步奖励

$$\mathrm{R}^a_s = E[R_{t+1}|S_t = s, A_t = a]$$

强化学习基本概念：倒立摆例子

- **倒立摆（CartPole）**：小车可以在水平轨道上左右移动，车上竖立一根杆子（“倒立摆”）；通过施加左右推力 (Action)，防止杆子倾倒，尽可能让其保持平衡

- **状态**通常由四个连续变量构成：$[x, \dot{x}, \theta, \dot{\theta}]$
  - 小车位置 x，小车速度 $\dot{x}$，杆子角度 θ，杆子角速度 $\dot{\theta}$
  - 当杆子倾斜过大，或者小车移动超出轨道边界，就视为失败状态，任务结束



- 智能体可执行的**动作**是对小车施加左右方向的推力
  - 在离散动作设定下，仅有两个动作选择：向左或向右
  - 在连续动作设定下，也可将动作定义为推力大小及方向的连续数

- **奖励**：每保持一帧（一个时间步）杆子不倒，加 1 分；如果杆子倒下或小车出界，任务立即结束，不再获得奖励
  - 智能体目标是尽可能延长杆子平衡的时间，从而获得更高累计奖励

目录

强化学习分类：基于模型 vs. 无模型

- **基于模型（Model-Based）**：智能体掌握或学习到**环境模型**：转移概率 P，奖励函数 R
  - 可以用 动态规划 或 搜索 (规划) 方法
  - 优点：可做“想象中的试错”，样本效率更高
  - 缺点：学习 / 获取模型可能困难或不精确
- **无模型（Model-Free）**：不显式建模 P 或 R，只通过交互直接学习价值或策略
  - 直接从环境交互中学习价值函数或策略
  - 优点：实现简单，适用于未知或复杂环境
  - 缺点：需要大量环境交互，学习效率可能较低
- 例子：AlphaGo 既可看做部分 Model-Based（通过 MCTS）也有 Model-Free 成分（学习价值网络、策略网络）

强化学习分类：价值式 vs. 策略式

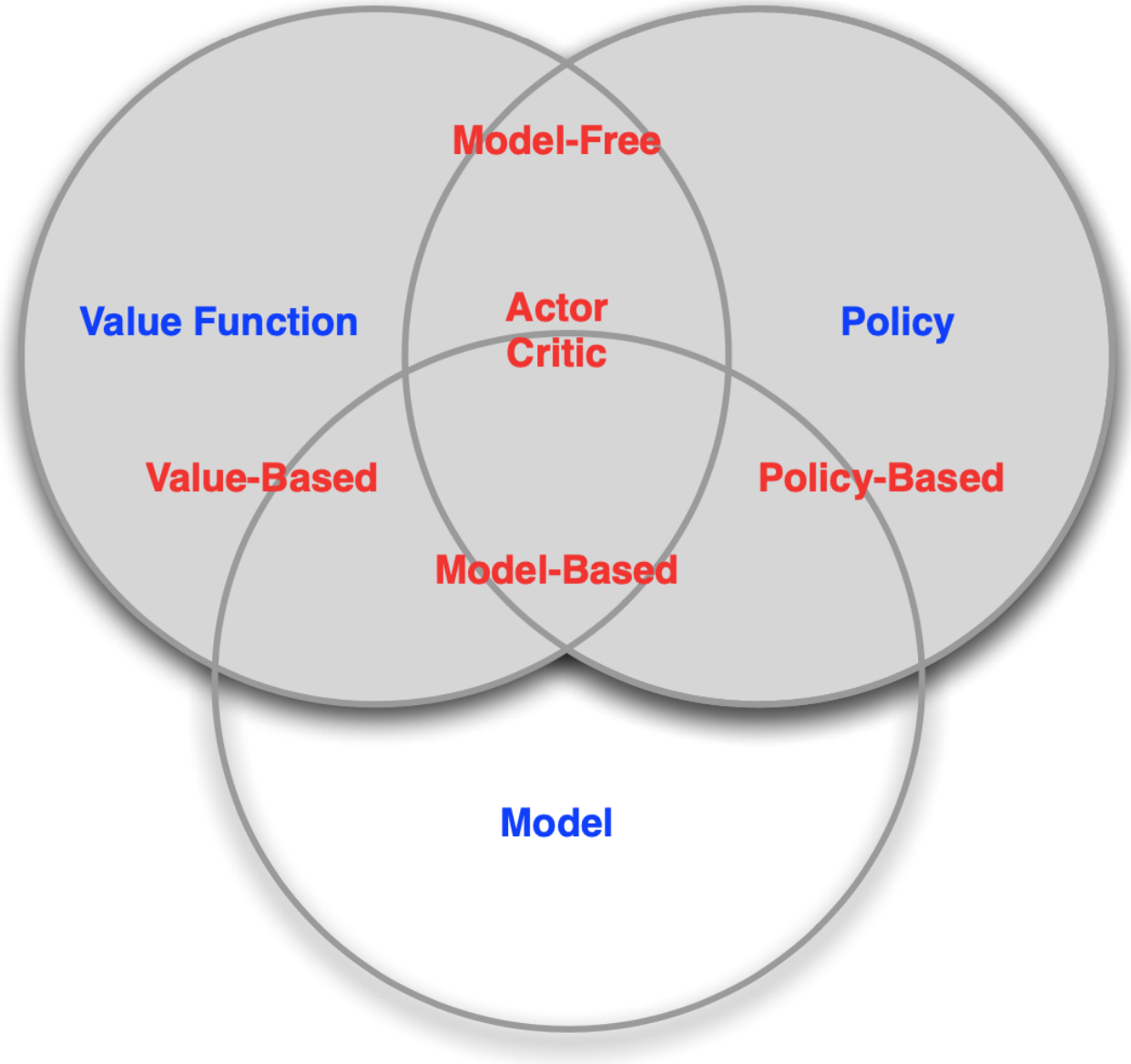
- **价值式（Value-Based）**：先学习状态 / 动作价值函数 $Q^{\pi}(s,a)$ 或 $V^{\pi}(s)$，再通过价值函数得到策略
  - 策略通常通过 $\arg max_a Q(s,a)$（贪心）或 ϵ-贪心获得
  - 优点：容易收敛到最优值，易于理解
  - 缺点：对高维连续动作空间不太友好；难以训练随机性策略
- **策略式（Policy-Based）**：直接对策略 $\pi_\theta(a|s)$ 参数化并优化
  - 不需要显式维护 Q；用梯度 (Policy Gradient) 优化策略
  - 优点：适合连续动作，高维动作场景；可以训练随机性策略
  - 缺点：容易出现高方差，需要结合基线降低方差
- **演员-评论家（Actor-Critic）**：价值式与策略式的结合
  - Actor（策略网络）直接输出动作分布，或输出动作的参数
  - Critic（价值网络）估计价值函数指导 Actor 更新

强化学习分类：同策略 vs. 异策略

- **同策略（On-policy）**：所收集的交互数据来自当前执行的策略，学习的也是这个策略本身
  - 优点：学习和执行一致，理论分析简单
  - 缺点：可能探索不足，样本利用率低
- **异策略（Off-policy）**：所收集的交互数据可能来自其他策略，能利用历史或外部数据
  - 优点：数据效率较高，可使用任意来源数据
  - 缺点：学习过程可能不稳定（行为与目标策略不一致）

强化学习分类汇总

- Model-Based vs. Model-Free
- Value-Based vs. Policy-Based
- On-policy vs. Off-policy



强化学习主要算法分类

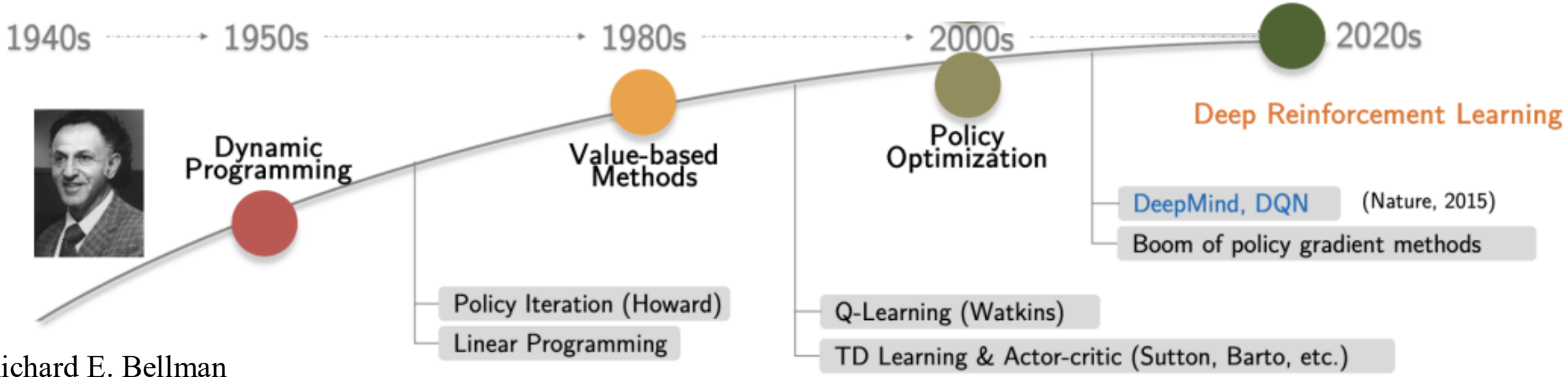
- **Q-Learning**：Model-Free + Value-Based + Off-policy
- SARSA：Model-Free + Value-Based + On-policy
- **DQN (Deep Q-Leaning Network)**：Model-Free + Value-Based + Off-policy
  - 在 Q-Learning 基础上使用深度网络逼近动作值函数
- REINFORCE：Model-Free + Policy-Based + On-policy
- A2C (Advantage Actor-Critic)：Model-Free + Actor-Critic + On-policy
- **PPO (Proximal Policy Optimization)**：Model-Free + Actor-Critic + (On-policy/Off-policy)
- SAC (Soft Actor-Critic)：Model-Free + Actor-Critic + Off-policy
- **GRPO (Group Relative Policy Optimization)**：Model-Free + Policy-Based + (On-policy/Off-policy)
- AlphaGo：部分 Model-Based (MCTS 搜索) + 部分 Model-Free + Actor-Critic + On-policy

目录

强化学习的历史脉络

强化学习的历史脉络

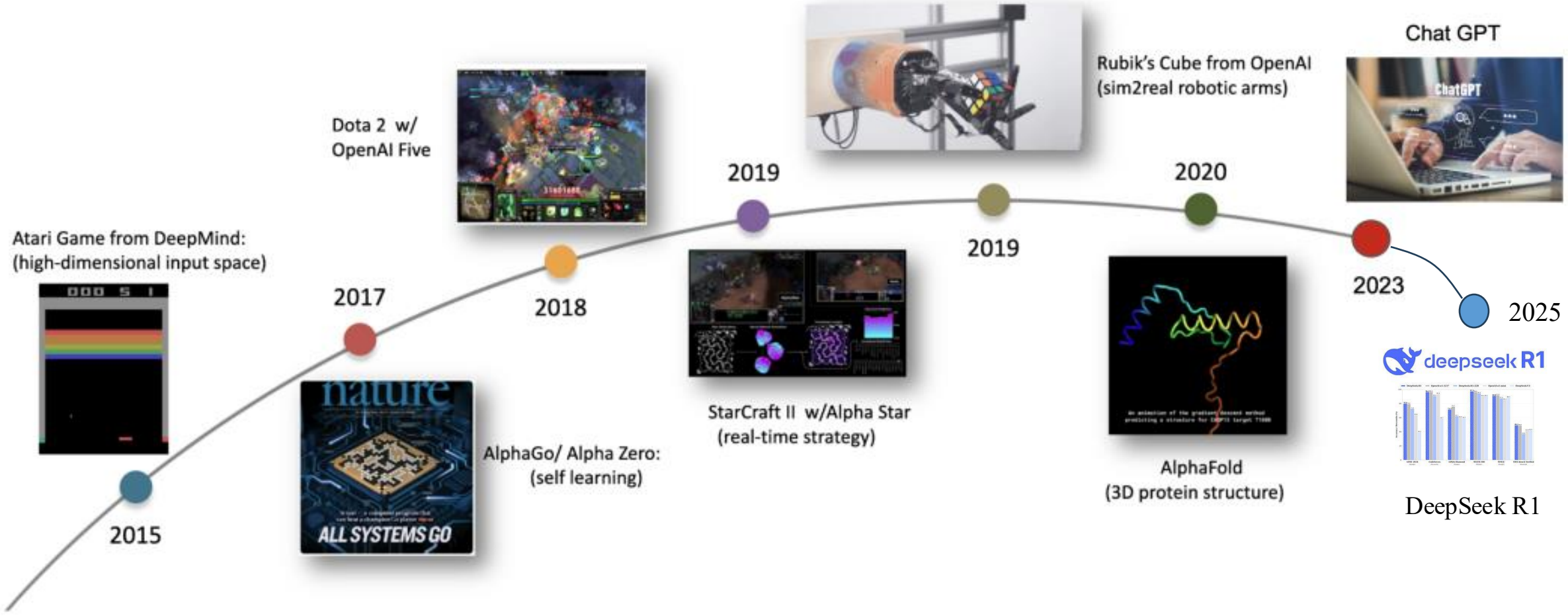
- 早期理论基础：马尔可夫决策过程（MDP）和动态规划（Dynamic Programming）
- 现代强化学习基石：时序差分（Temporal Difference，TD）学习，Q-Learning
- 与深度学习结合：DQN (Deep Q-Network)，AlphaGo
- 现代深度强化学习的百花齐放：Actor-Critic 体系，离线强化学习，多智能体强化学习



Richard E. Bellman

深度强化学习发展里程碑

□ 强化学习与深度学习结合仍在快速演变，在具身智能和大模型方面展示出巨大潜力

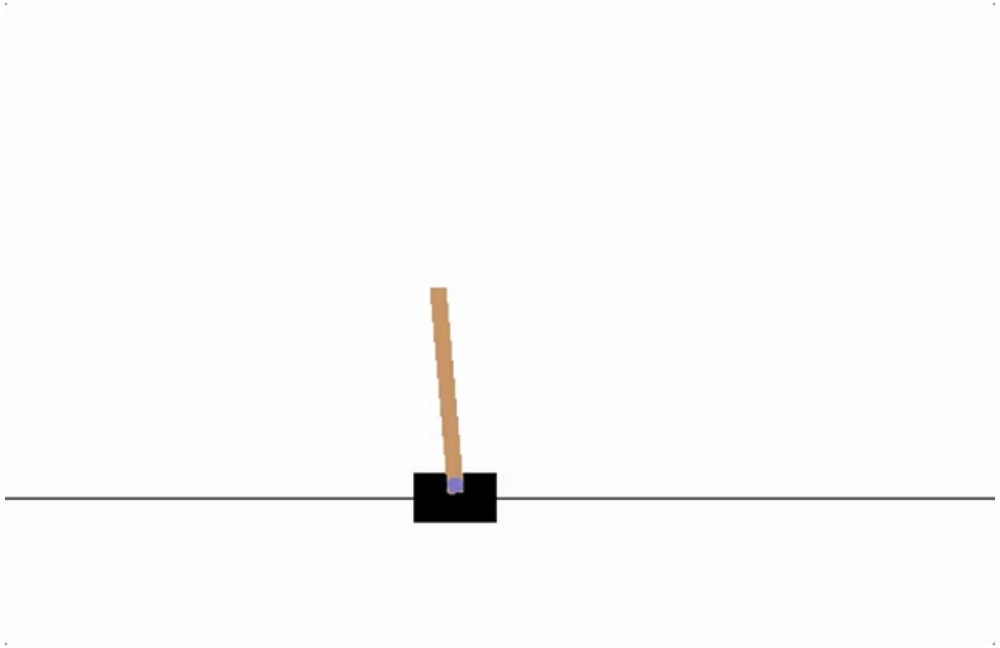


目录

强化学习示例——倒立摆

什么是倒立摆?

- 倒立摆是经典的控制和强化学习测试环境。
- 系统由一根可以绕某个轴旋转的杆（摆杆）和底座（小车）组成。
- 目标是通过施加控制力或动作，让摆杆在竖直向上的“不稳定平衡点”维持平衡，而不是倒向一侧。



强化学习示例——倒立摆

□ 如何将强化学习应用于倒立摆?

第一步：**构建环境**

- 在强化学习中，环境（Environment）是指系统与智能体（Agent）交互的场所。
- 对于倒立摆，环境需要实时模拟物理运动，即，小车受力后环境状态的变化，包括：
  - 小车的位置、速度；
  - 摆杆的角度、角速度等。
- 可以使用开源模拟环境（如 OpenAI Gym 的 CartPole）或自行编程实现。

第二步：**定义奖励函数**

- 奖励函数（Reward Function）用于衡量智能体行为的好坏：
  - 当摆杆接近竖直状态且小车位置在合理范围内时，应给予较高或正向的奖励；
  - 如果摆杆角度偏离太大甚至倒下，应给予低奖励或惩罚。

强化学习示例——倒立摆

□ 如何将强化学习应用于倒立摆?

第三步：**定义状态空间与动作空间**

- 状态空间（State Space）指环境可观测的变量集合，比如：
  - 小车位置 x，小车速度 $\dot{x}$；
  - 杆子角度 θ，杆子角速度 $\dot{\theta}$
- 动作空间（Action Space）指智能体能够做出的控制操作，比如：
  - 对小车施加的力或扭矩；
  - 在离散动作环境中，可以是“向左推”、“向右推”两种动作；
  - 在连续动作环境中，可以是任意大小的力或扭矩值。

强化学习示例——倒立摆

口 如何将强化学习应用于倒立摆?

第四步：**算法选择与训练**

- 在倒立摆问题中，常用的强化学习方法包括：
  - 价值式方法（基于值函数的方法）： Q-Learning, DQN (Deep Q-Network) 等；
  - 策略式方法（基于策略的方法）： Policy Gradient, REINFORCE, PPO (Proximal Policy Optimization) 等。
- 训练的目标是在不断试错中找到最优（或近似最优）策略，使智能体在任何状态下都能采取合适的动作来保持摆杆平衡。

强化学习示例——倒立摆

□ 如何将强化学习应用于倒立摆?

第五步：**测试与可视化**

- 将训练好的智能体在同一环境下进行测试：
  - 观察摆杆能否在多次随机初始条件下成功保持平衡；
  - 统计能坚持的时间步数，或在限制时间内是否倒下。
- 可视化有助于理解智能体在各个时刻的决策：
  - 使用图形界面或实时渲染来查看摆杆随时间的角度变化，以及小车在轨道上的移动。
  - 分析失败案例，找出模型需要改进的地方。

强化学习示例——倒立摆

□ 常用指标

- 平均回合奖励
  - 每个回合获得的总奖励取平均值；
  - 越高表示智能体在倒立摆环境中表现越好。
- 成功率
  - 若设置了限定时间步（如500步），统计智能体在该时间步内不倒下的回合数占比；
  - 越高表明智能体更稳定地维持摆杆平衡。
- 步数
  - 倒立摆在平衡状态下持续的时间步数；
  - 可以帮助衡量模型是否具有稳健的控制策略。

课后作业

1. 假设使用强化学习训练一个控制策略来玩“超级马里奥”游戏，这里的状态，动作，奖励应该怎么设计?
2. 假设使用强化学习训练一个策略来控制一台自动驾驶车辆，让其在城市道路安全且高效地行驶，这里的状态，动作，奖励应该怎么设计?

谢谢！