强化学习应用指南

张文韬

目录

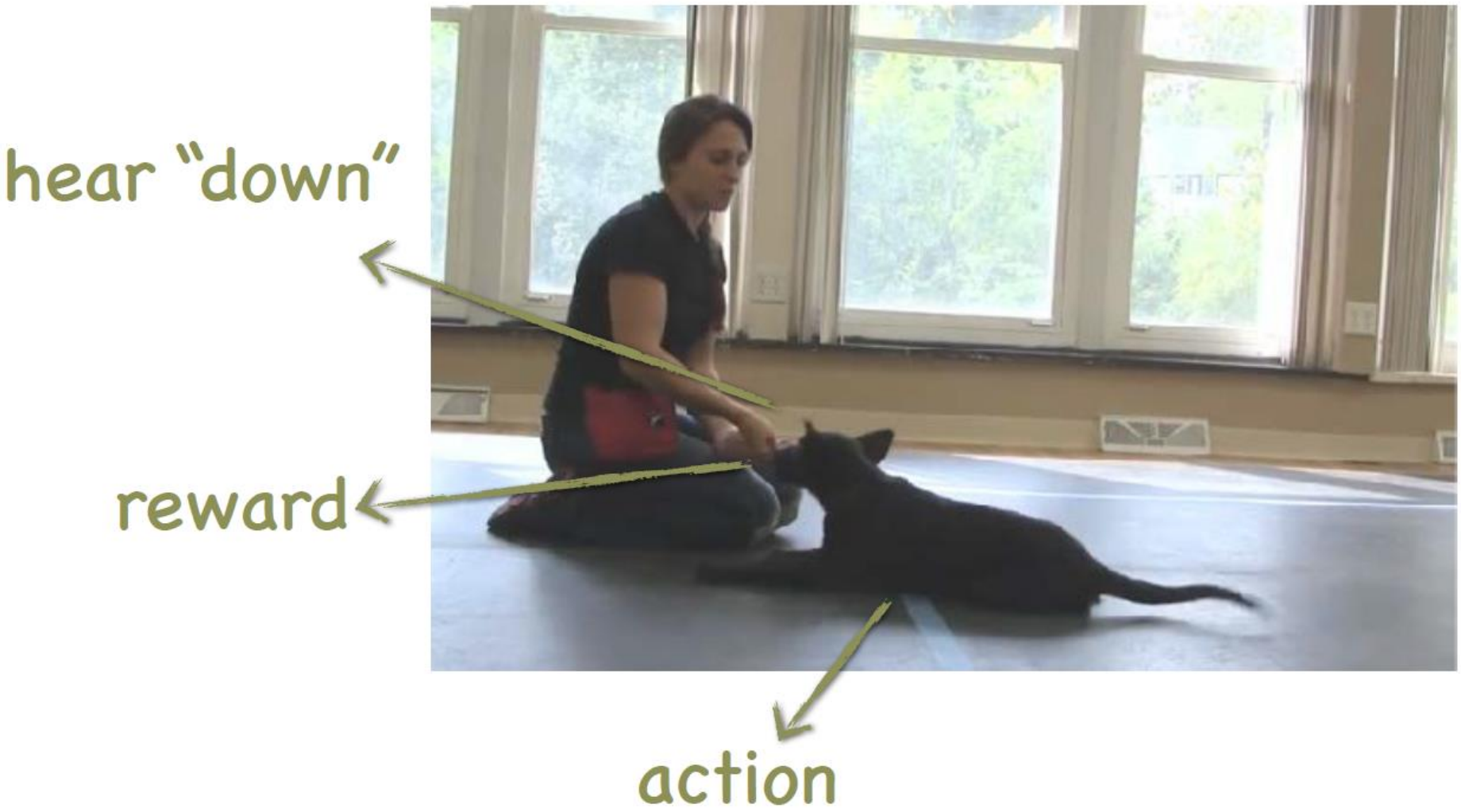
以应用方法为主，不涉及强化学习原理细节

一．强化学习简介

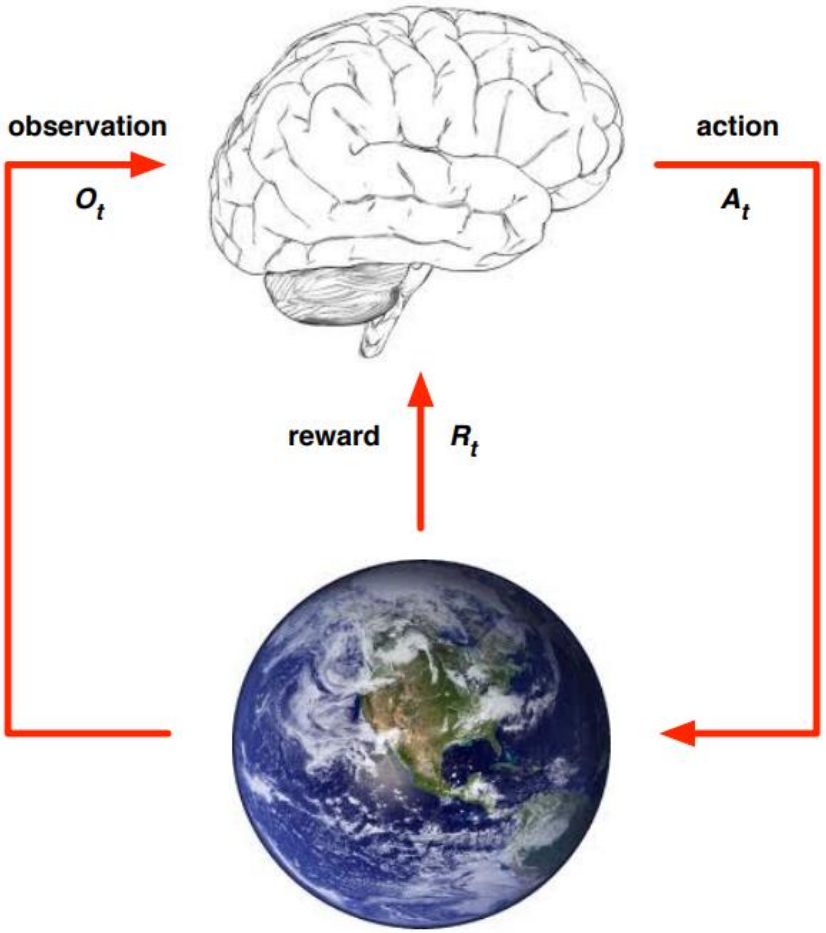
什么是强化学习？

- 强化学习，又称增强学习
- Reinforcement learning.
- Reinforcement：something that strengthens or encourages something: such as a response to someone's behavior that is intended to make that person more likely to behave that way again
- 套路，方法，范式
- 鼓励某个人或某事以更高的可能性产生同样的行为

How to train a dog?

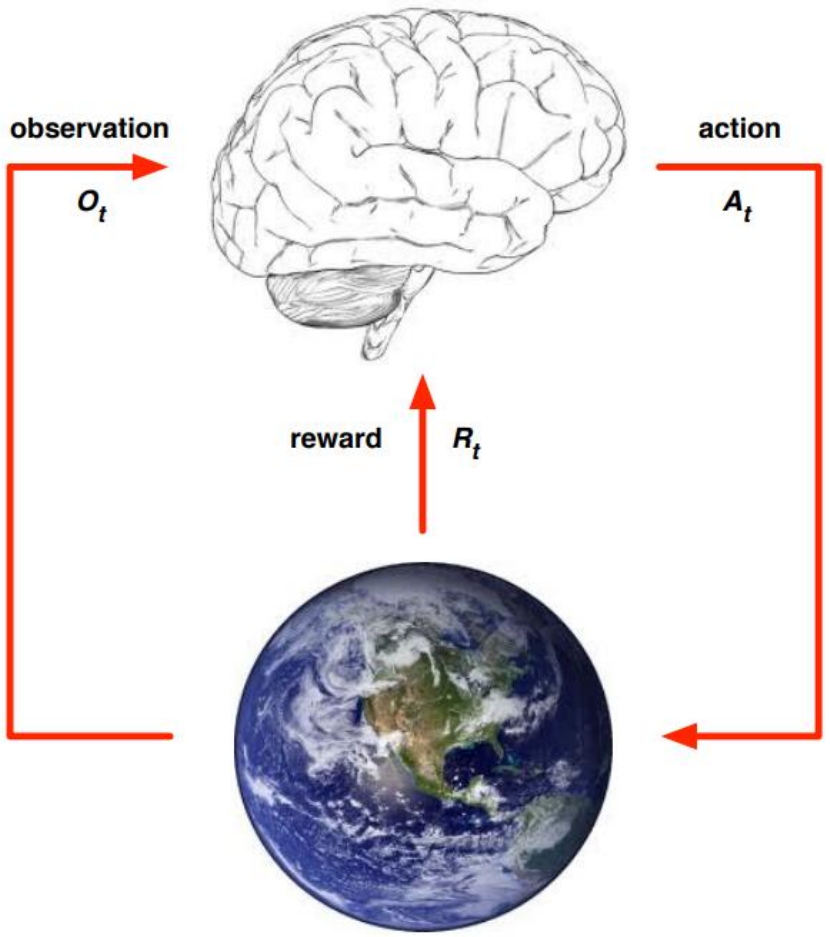


抽象成框架



why

- 学习方法
- 动态地适应环境
- 完成顺序决策任务



how

- 简单的原理介绍
- 马尔科夫决策过程：将强化学习交互过程以概率论的形式表示出来，以此推出解决强化学习问题的各种方法
- 值函数：强化学习的目标长期奖励
- 贝尔曼方程：方便地计算出值函数

二. 强化学习应用

- 计算机系统
- 工业界
- 机器人
- 自动驾驶
- 游戏
- 棋类

计算机系统

- 控制与优化
  - 参数调节
  - 云计算中的资源分配
  - 数据中心电源管理
- 安全
  - 攻击和防御算法

工业界

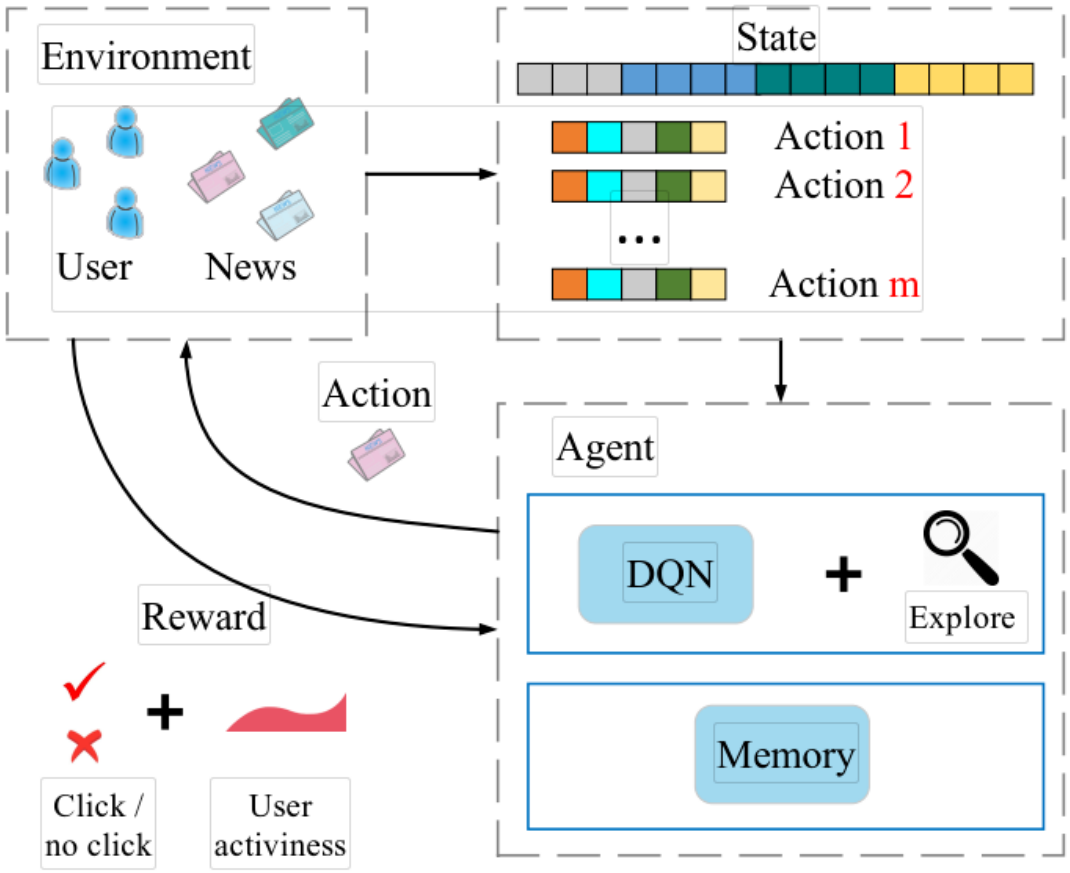
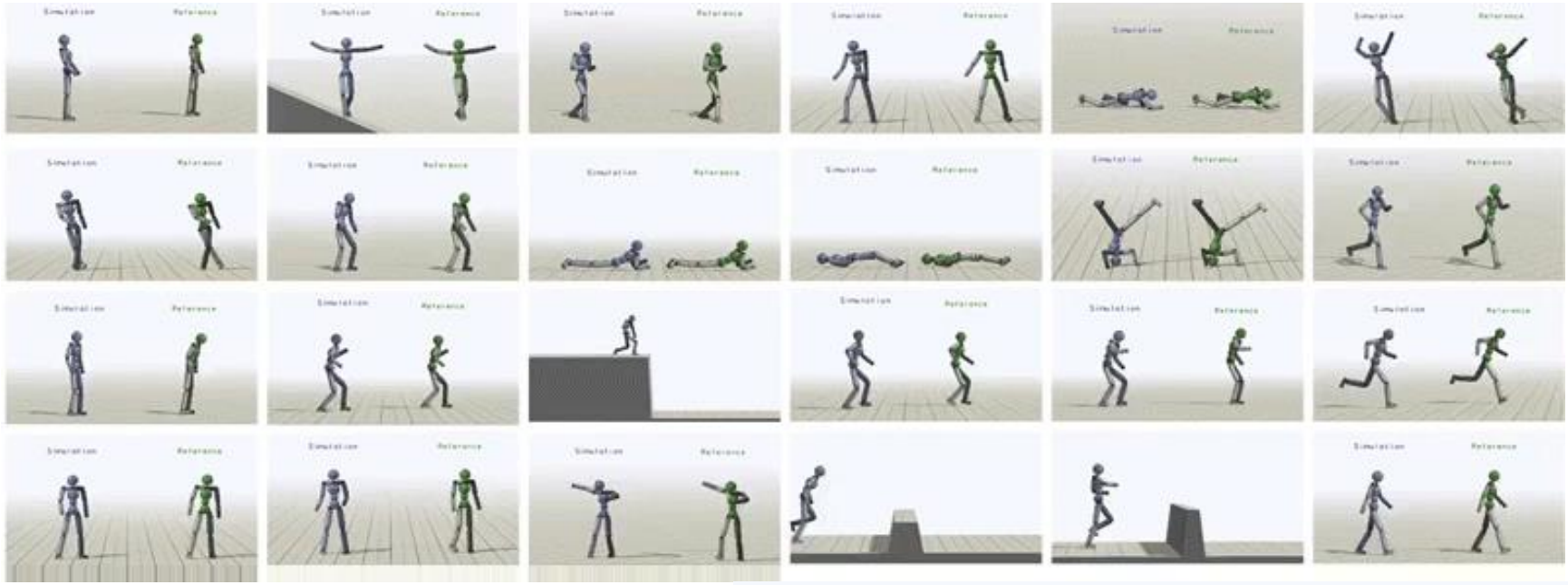
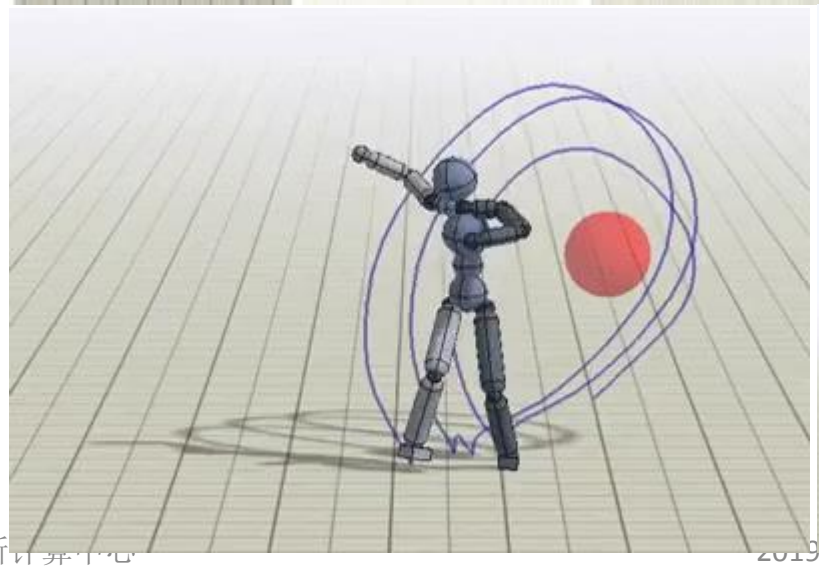


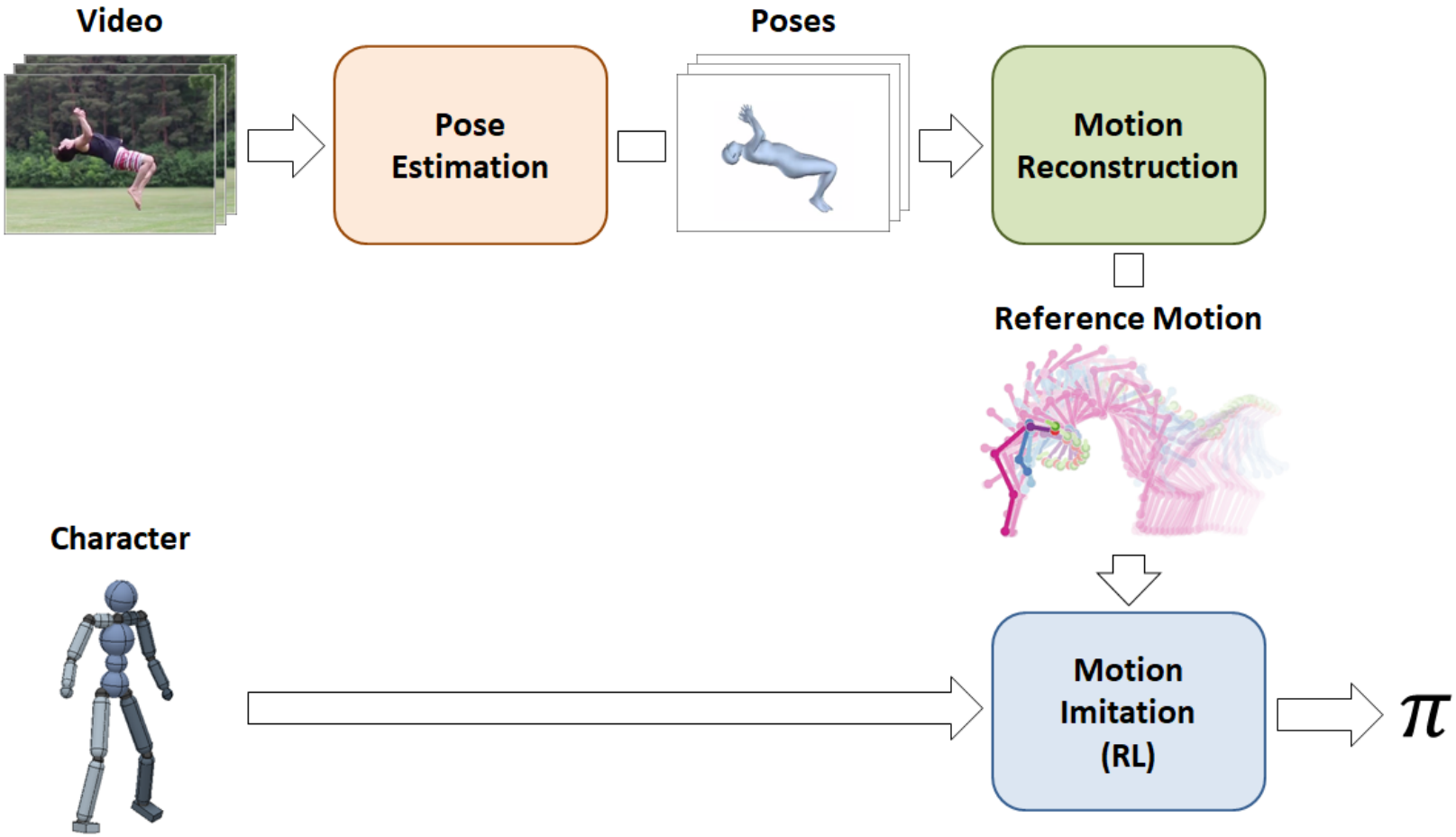
Figure 2: Deep Reinforcement Recommendation System

机器人控制之DeepMimic





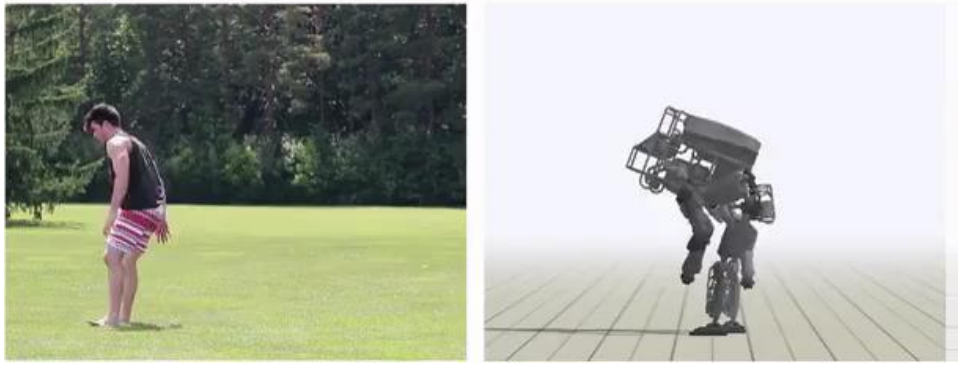
机器人控制之SFV


Video
Pose Estimation
Poses
Motion Reconstruction
Reference Motion
Character
Motion Imitation (RL)
π

机器人控制之SFV

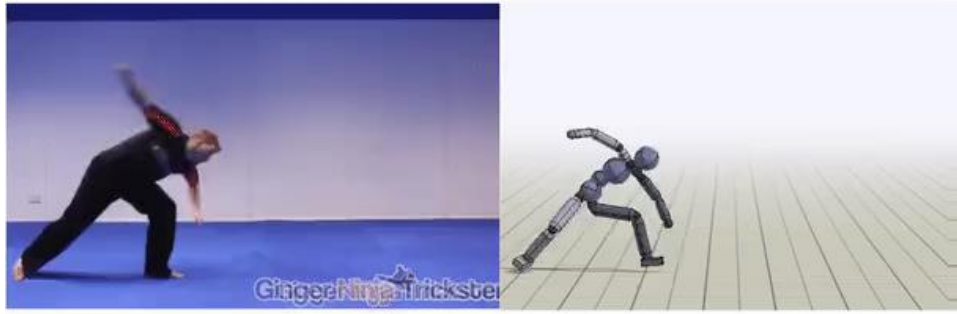
Dance

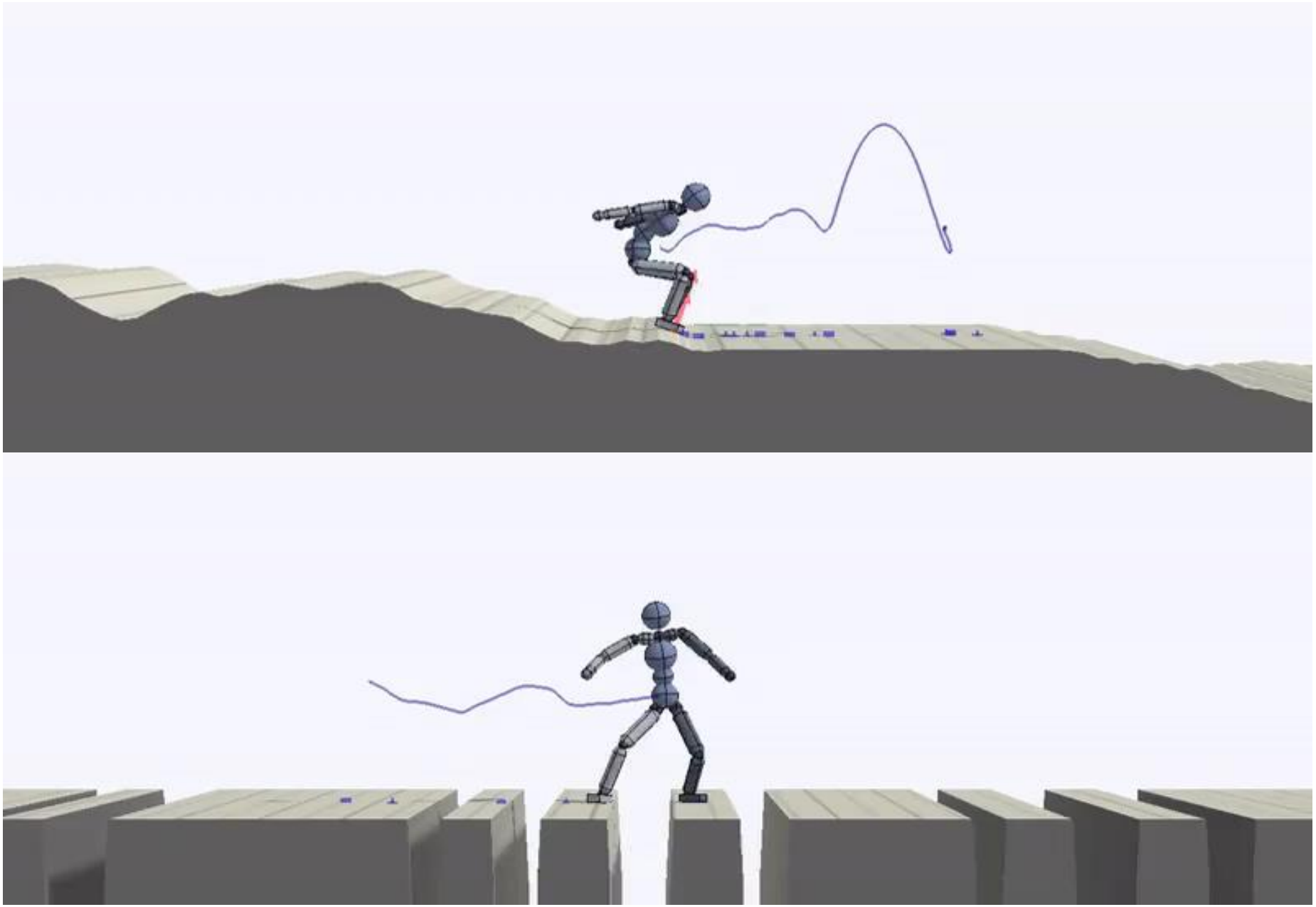
Handspring A



Kip-Up

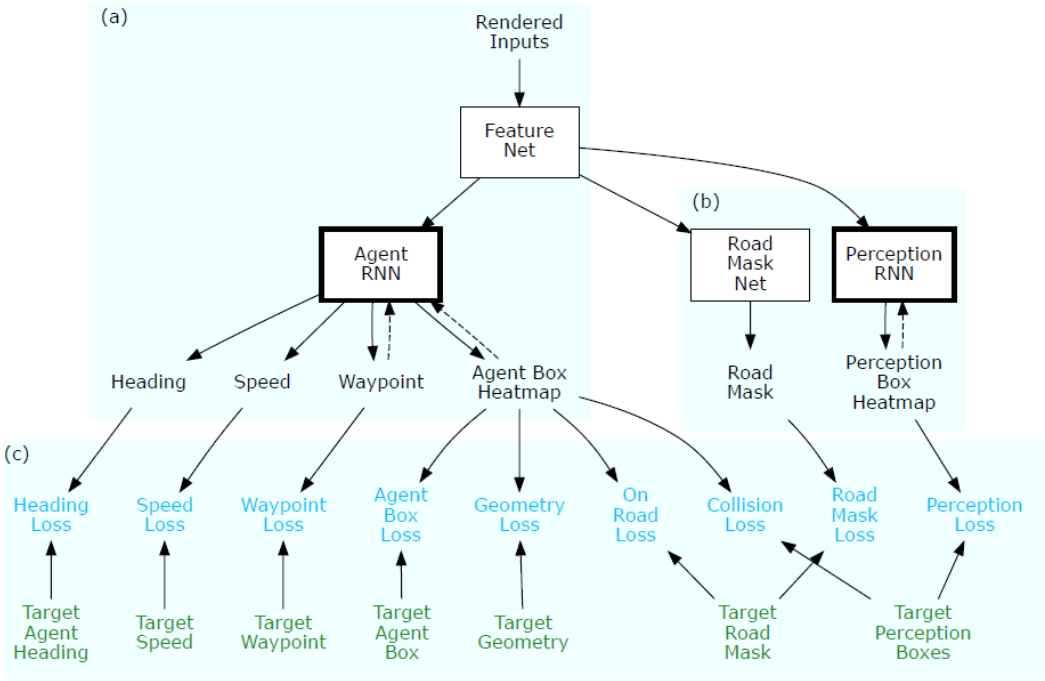
Spin





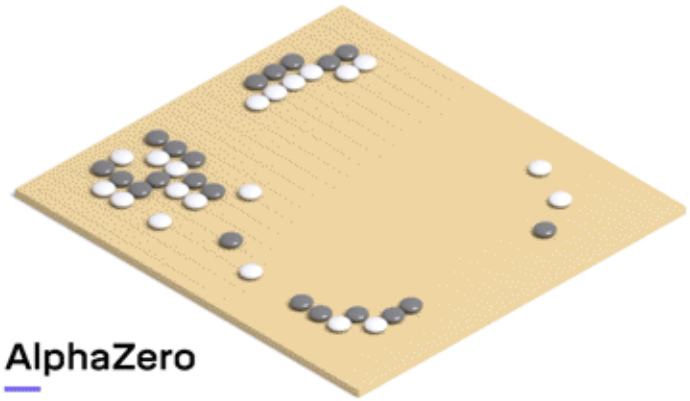
二. 强化学习应用

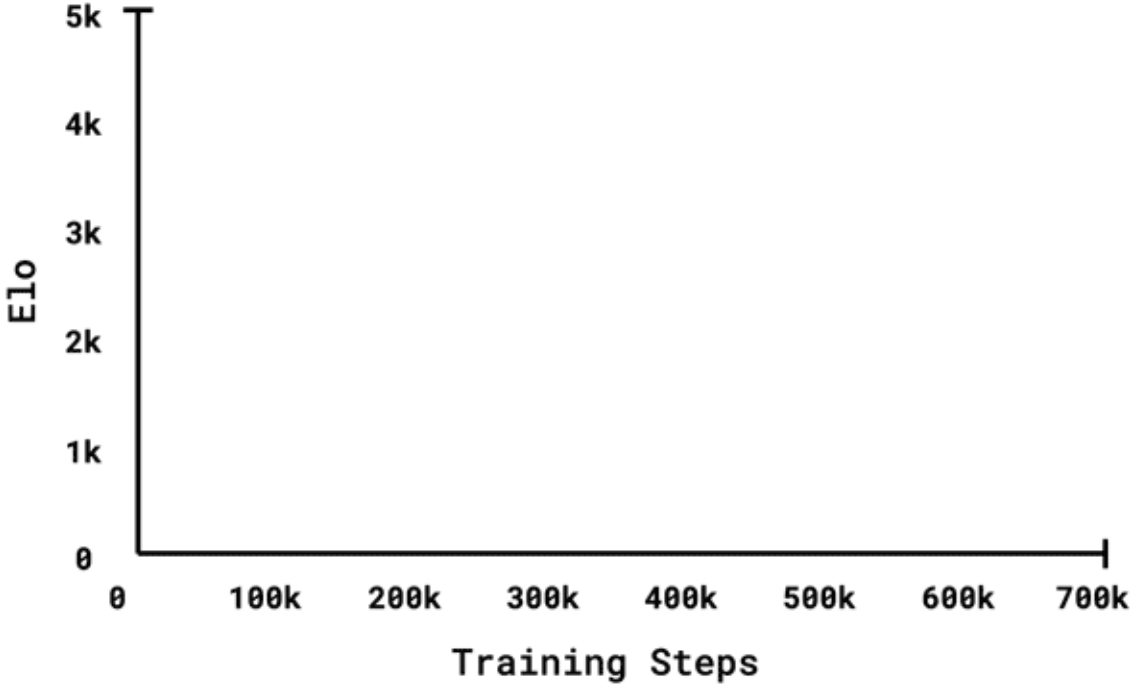
- 计算机系统
- 工业界
- 机器人
- 自动驾驶
  - waymo
- 游戏
  - Dota2: OpenAI five, StarCraft II: AlphaStar
- 棋类



AlphaZero：现阶段最高智能水平

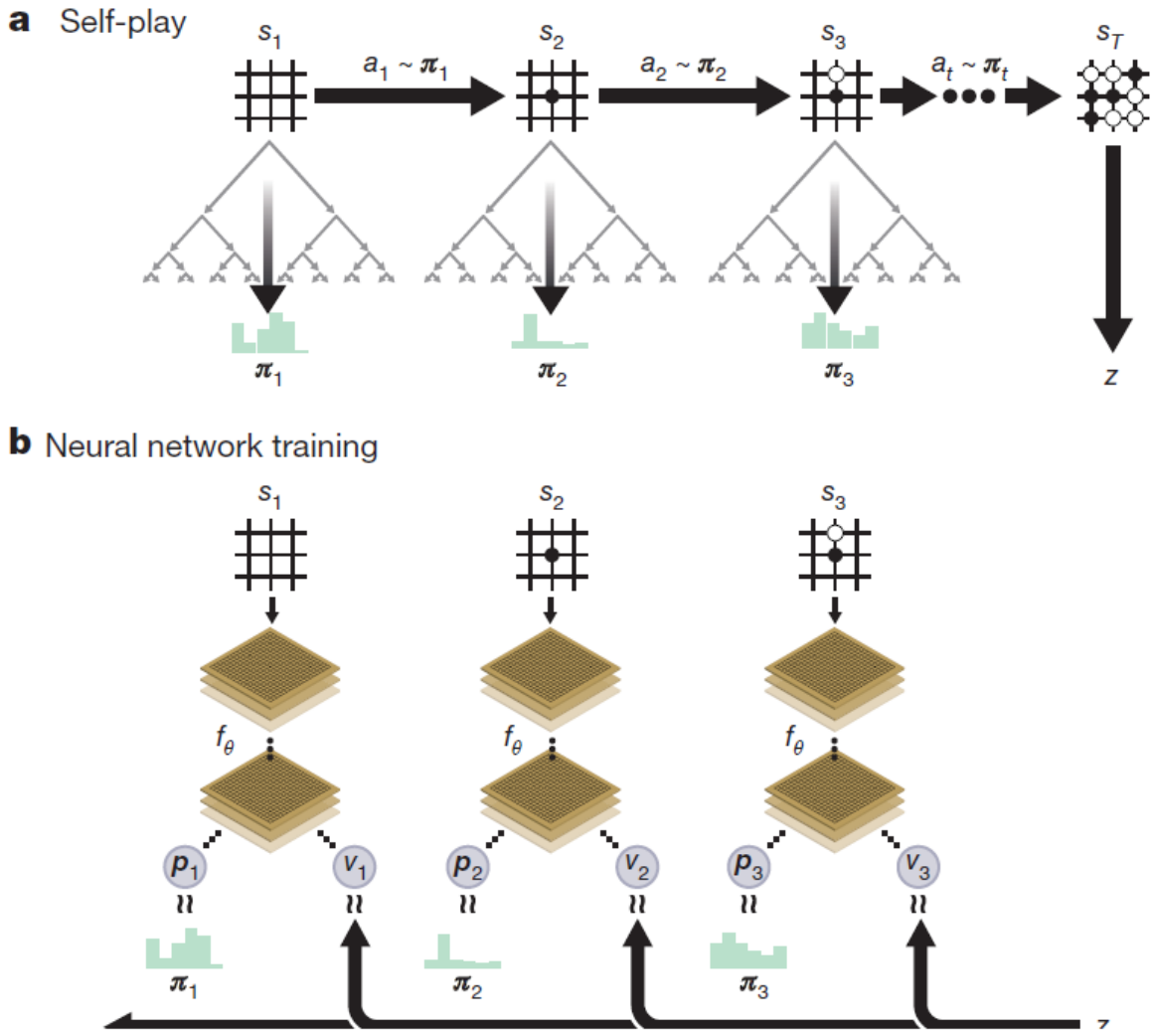
- AlphaGo，AlphaGo Zero, AlphaZero



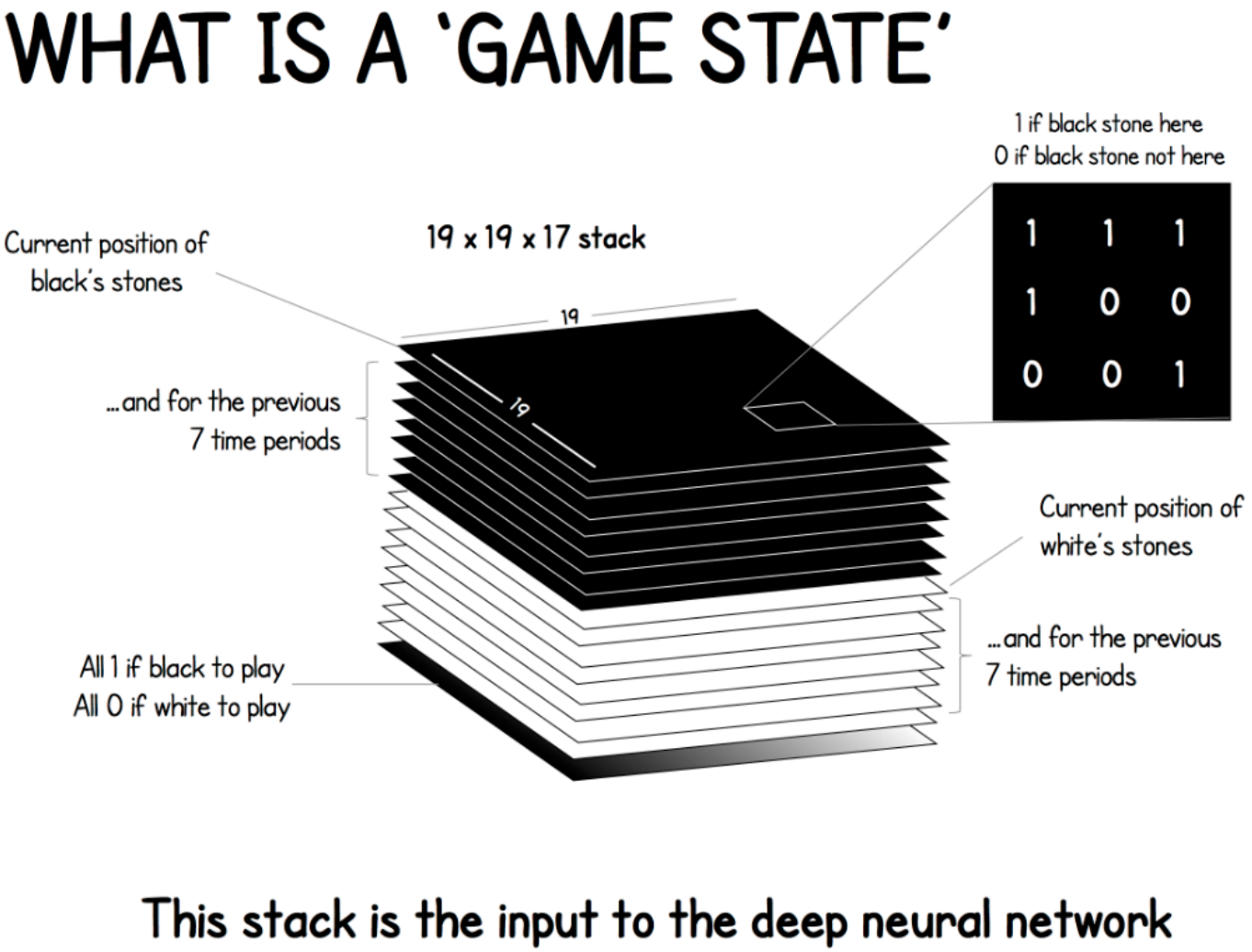


AlphaZero原理浅析

1. 自我博弈
2. 训练
3. 评估



AlphaZero原理浅析



SELF PLAY

Create a 'training set'

The best current player plays 25,000 games against itself

See MCTS section to understand how AlphaGo Zero selects each move

At each move, the following information is stored (s_t, π_t, z_t)

The game state
(see 'What is a Game State section')

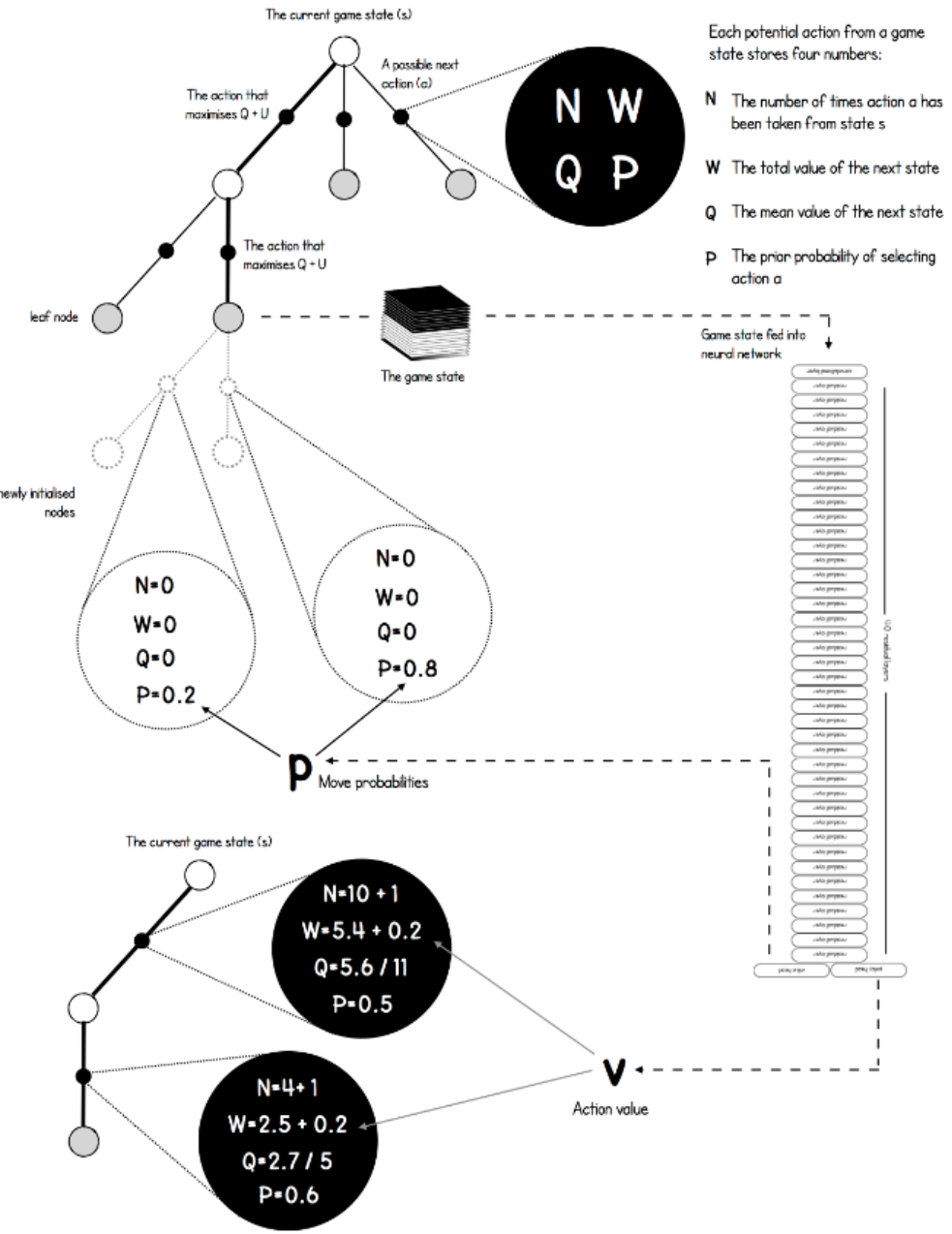
π

The search probabilities
(from the MCTS)

The winner
(+1 if this player won, -1 if this player lost - added once the game has finished)

MONTE CARLO TREE SEARCH (MCTS)

How AlphaGo Zero chooses its next move



First, run the following simulation 1,600 times...

Start at the root node of the tree (the current game state)

1. Choose the action that maximises...

Q + U

$$U(s,a) = c_{\text{puct}} P(s,a) \frac{\sqrt{\sum_b N(s,b)}}{1 + N(s,a)}$$

The mean value of the next state

A function of P and N that increases if an action hasn't been explored much, relative to the other actions, or if the prior probability of the action is high

Early on in the simulation, U dominates (more exploration), but later; Q is more important (less exploration)

2. Continue until a leaf node is reached

The game state of the leaf node is passed into the neural network, which outputs predictions about two things:

p Move probabilities

v Value of the state (for the current player)

The move probabilities p are attached to the new feasible actions from the leaf node

3. Backup previous edges

Each edge that was traversed to get to the leaf node is updated as follows:

$N \rightarrow N + 1$

$W \rightarrow W + v$

$Q = W / N$

AlphaZero原理浅析

…then select a move

After 1,600 simulations, the move can either be chosen:

Deterministically (for competitive play)

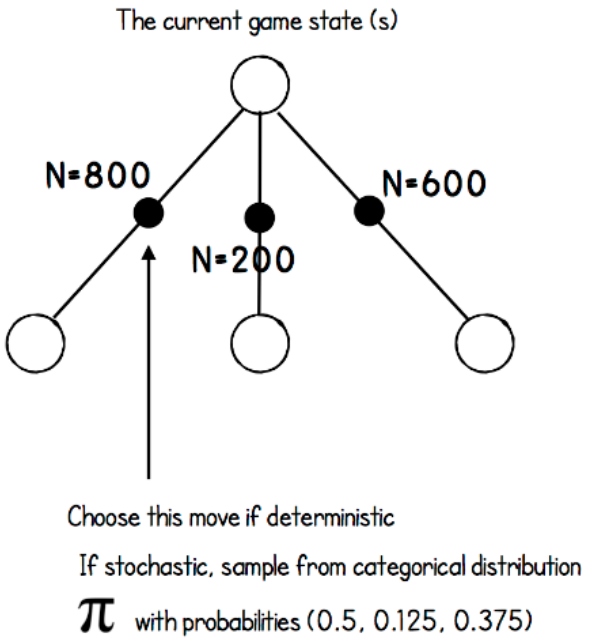
Choose the action from the current state with greatest **N**

Stochastically (for exploratory play)

Choose the action from the current state from the distribution

$$\pi \sim N^{1/\tau}$$

where τ is a temperature parameter, controlling exploration



RETRAIN NETWORK

Optimise the network weights

$$\ell = (z - v)^2 - \boldsymbol{\pi}^T \log \mathbf{p} + c||\theta||^2$$

A TRAINING LOOP

Sample a mini-batch of 2048 positions from the last 500,000 games

Retrain the current neural network on these positions

- The game states are the input (see 'Deep Neural Network Architecture')

Loss function

Compares predictions from the neural network with the search probabilities and actual winner

PREDICTIONS

p Cross-entropy π

\+

v Mean-squared error 

\+

Regularisation

ACTUAL

After every 1,000 training loops, evaluate the network

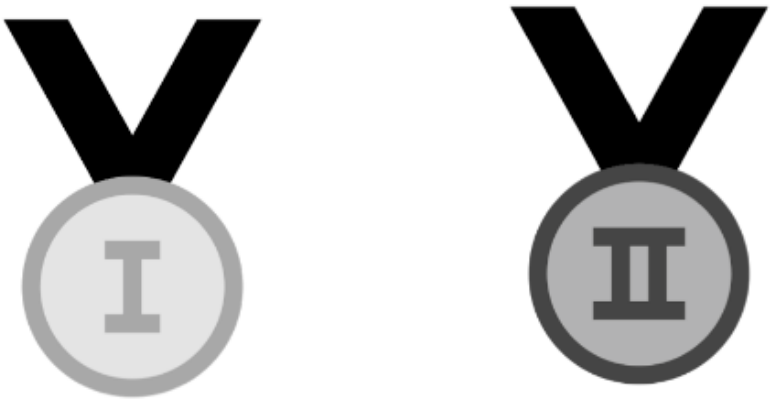
EVALUATE NETWORK

Test to see if the new network is stronger

Play 400 games between the latest neural network and the current best neural network

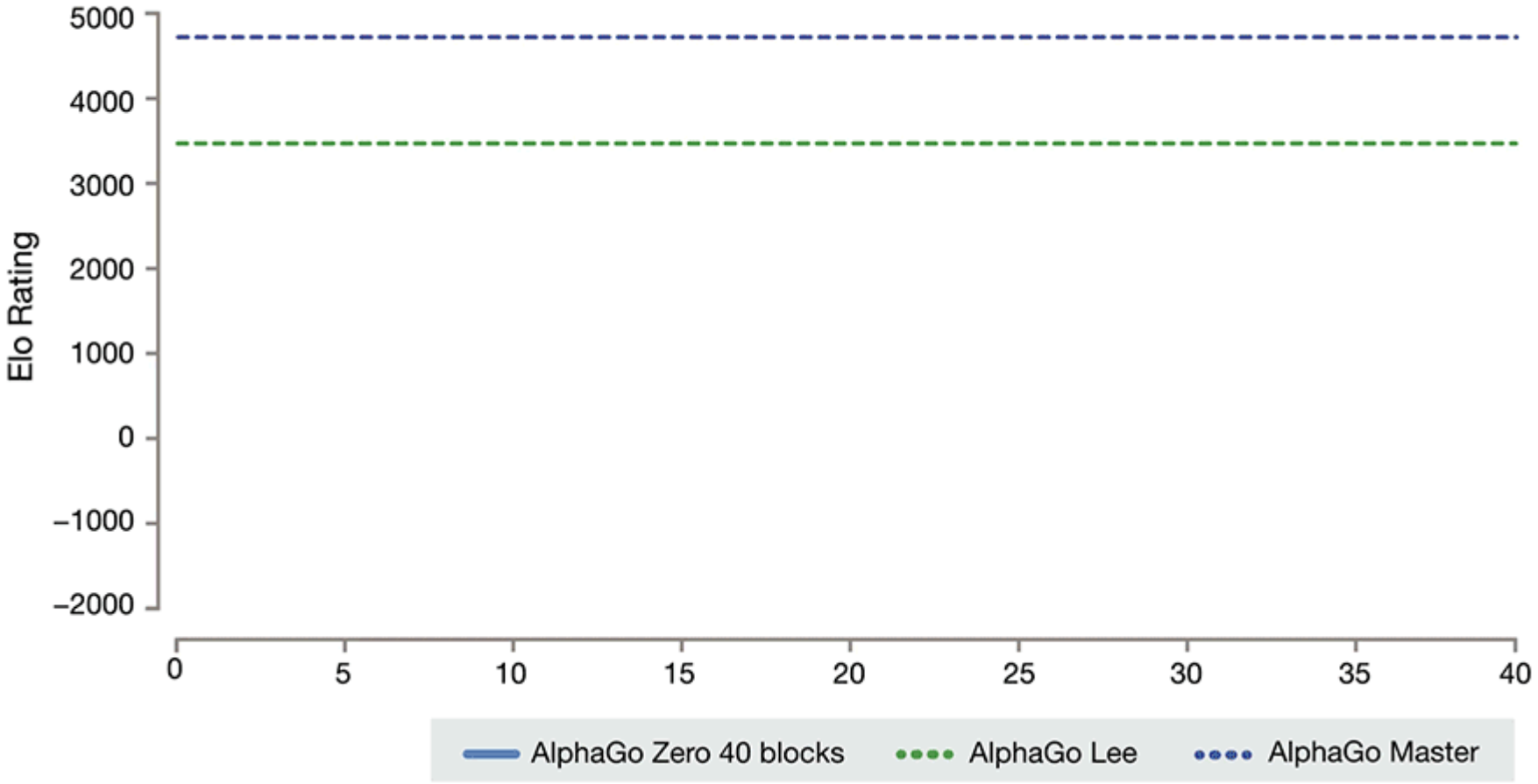
Both players use MCTS to select their moves, with their respective neural networks to evaluate leaf nodes

Latest player must win 55% of games to be declared the new best player



AlphaZero原理浅析

a single machine with 4 tensor processing units (TPUs)



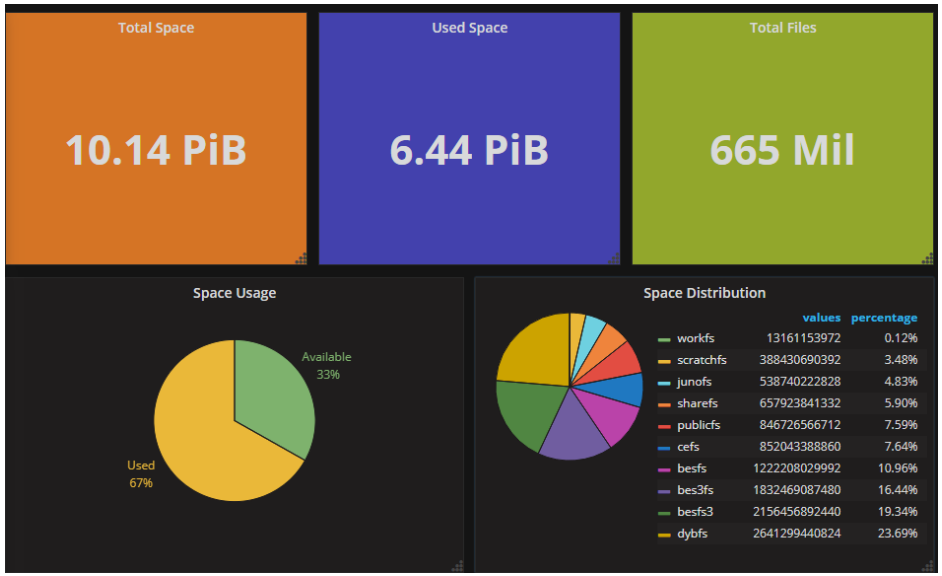
48 TPUs

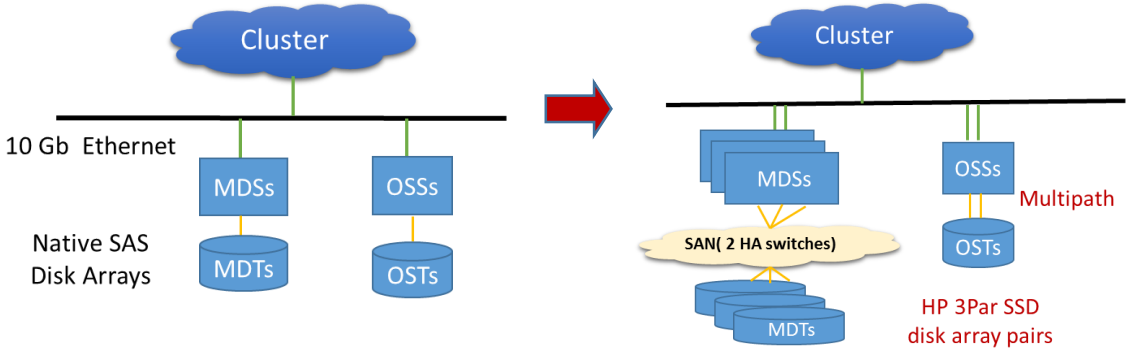
several months

三. 强化学习在计算中心的应用

Lustre在高能所的部署

- 容量
  - ~11 PB，60%+ 使用率，6亿+文件
  - 按照应用分成10个挂载点
  - ~60台存储服务器
  - >1000客户端
  - 万兆以太网互联
- 版本
  - 2.5.3
- 使用模式
  - 不分条，stripe=1





存储系统性能影响因素分析

- 硬件级影响因素
  - 网络
  - 存储连接通道
  - 磁盘阵列 IO 能力
  - 服务器处理能力
- 操作系统级影响因素
  - I/O Scheduler
  - 系统参数
- Lustre文件系统配置影响因素

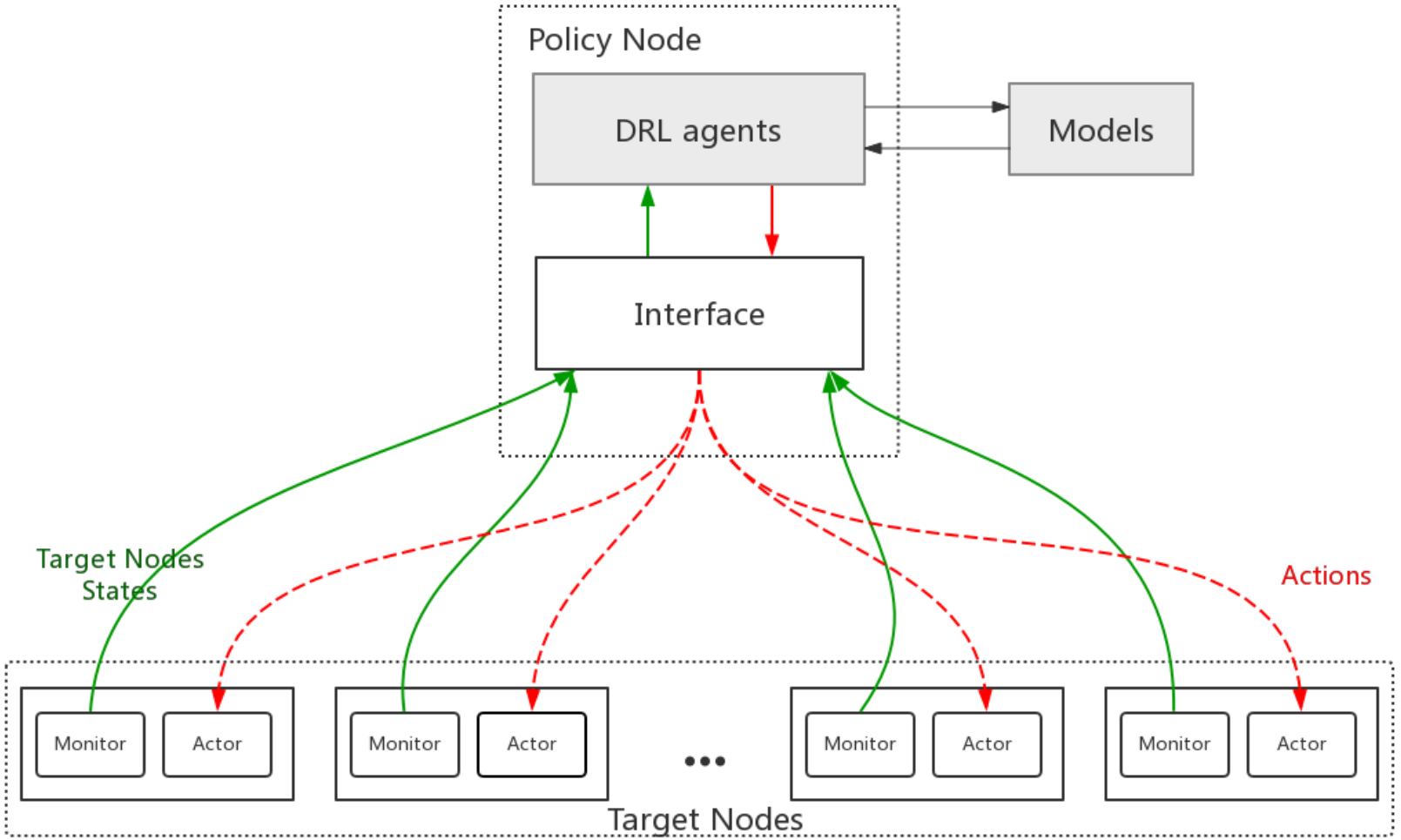
参数调优

- 根据系统的负载和当前状态，调节相应的参数值以提高系统的性能
- 人工调节存在的问题
  - 调节和反馈之间的延时导致认知偏差
  - 庞大的参数空间
  - 负载和设备的多样性
  - 人工成本
- 传统算法
  - 参数搜索
    - 参数搜索空间很大时，优化效率很低
    - 不支持动态负载变化

参数调优

- 深度强化学习
  - 强化学习：优秀的决策能力，对感知问题束手无策
  - 深度学习：具有较强的感知能力，但缺乏一定的决策能力
  - 两者结合，优势互补，能够为复杂状态下的感知决策问题提供解决思路
- 基于深度强化学习的参数调优
  - 典型的顺序决策问题，Agent：调节引擎，环境：存储系统
  - 特点
    - 支持庞大的参数空间，动态多变的负载
    - 无需先验知识
    - 在线学习
    - 24*7自动化

基于深度强化学习（DRL）的参数调优



基于深度强化学习（DRL）的参数调优

状态信息	介绍
read throughput	读吞吐率
write throughput	写吞吐率
dirty_pages_hits	脏页缓存命中的写操作数
dirty_pages_misses	脏页缓存未命中的写操作数
read IOPS	每秒读操作数
write IOPS	每秒写操作数
used_mb	使用的缓存空间
unused_mb	空闲的缓存空间
recliam_count	缓存回收次数
cur_dirty_bytes	OSC上写入和缓存的字节数
cur_grant_bytes	OSC与每个OST保留的回写缓存空间
inflight	待处理的RPC的数量
timeouts	RPC超时值
avg_waittime	请求平均等待时间
osc_cached_mb	当前总缓存空间
req_waittime	请求在服务器处理之前在队列中等待的时间量
req_active	现在正在处理的请求数

state

参数名	介绍	默认值	最小值	最大值	步长
max_dirty_mb	OSC中可以写入多少MB脏数据	32	32	4096	32
max_pages_per_rpc	在单个RPC中对OST进行I / O的最大页数	256	64	1024	64
max_rpcs_in_flight	从OSC到其OST的最大并发RPC数	8	4	256	4
max_read_ahead_mb	文件上预读的最大数据总量	40	0	160	40
max_cached_mb	客户端缓存的最大非活动数据量	128	128	32234	128
max_read_ahead_whole_mb	可以完整读取的文件的最大大小	2	0	8	2
statahead_max	statahead线程预取的最大的文件元数据	32	0	8192	32

action

IOZone测试结果

测试数据（32个进程平均吞吐率，单位：KB/S）

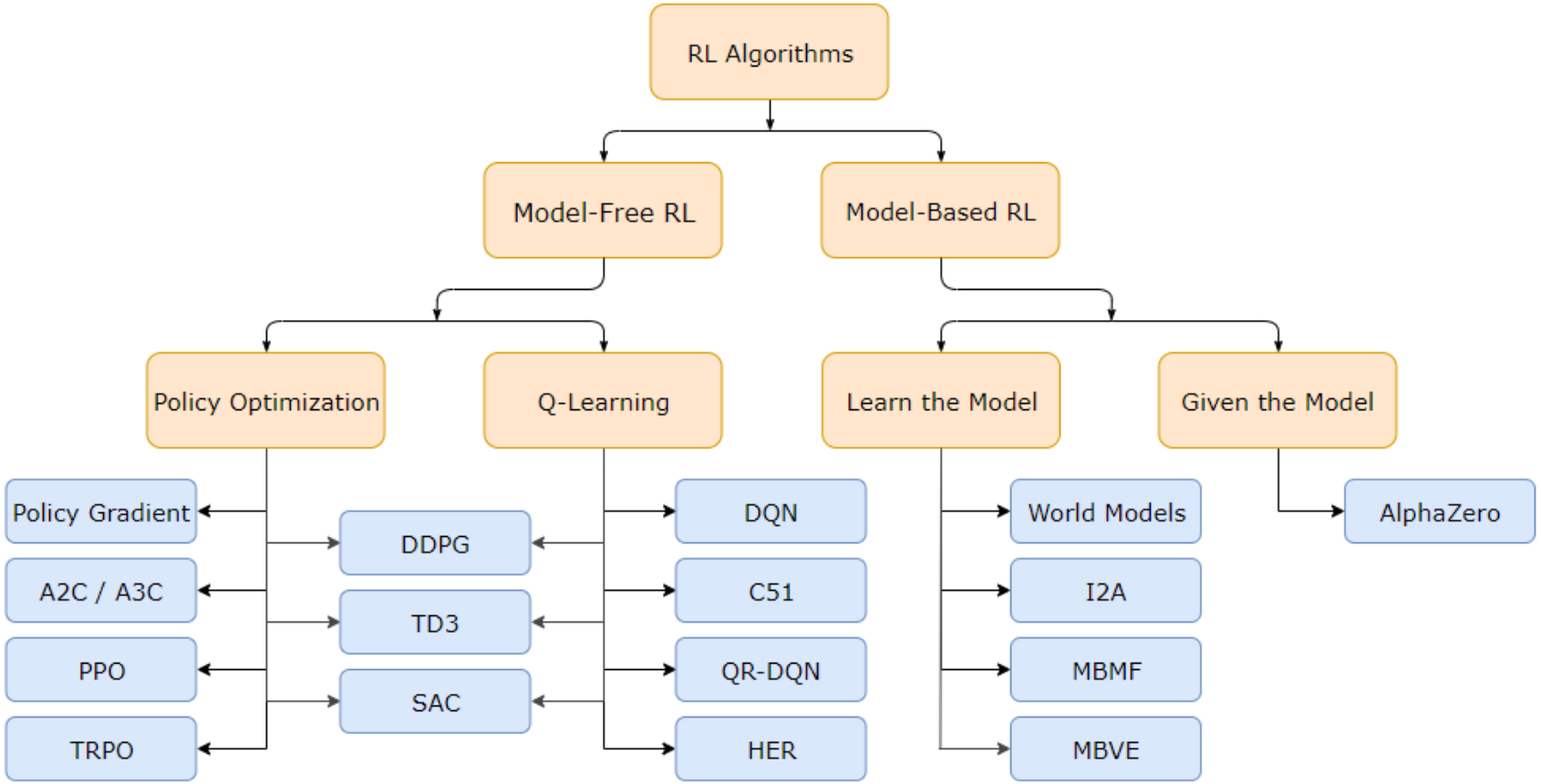
算法 / 测试项	Base	DQN	A2C	PPO
Initial write	1233	1186 (-4%)	1172 (-5%)	1215 (-1%)
Re-write	1249	1202 (-4%)	1171 (-6%)	1214 (-3%)
Read	38613	30505 (-21%)	36980 (-4%)	40924 (+6%)
Re-read	41777	30825 (-26%)	41241 (-1%)	43380 (+4%)
Stride read	3493	3570 (+2%)	4051 (+16%)	3941 (+13%)
Random read	1893	1566 (-17%)	2505 (+32%)	2407 (+27%)
Random write	560	565 (+1%)	581 (+4%)	563 (+1%)

强化学习应用指南

- 哪些问题？
  - 顺序决策问题
- 可否一战？
  - 考虑三要素：状态、动作、奖励
  - 采样效率（虚拟淘宝）
  - 写成环境

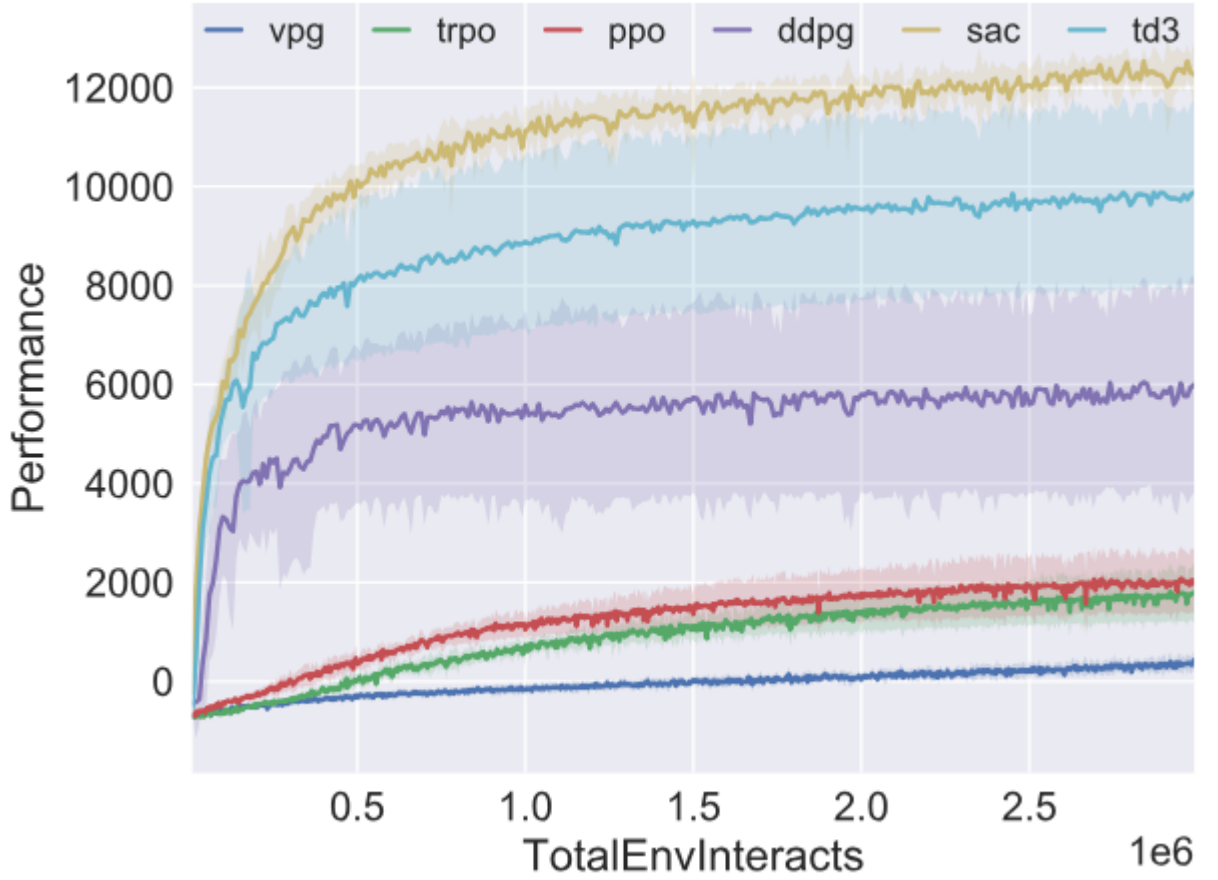
强化学习应用指南

- 算法选择



强化学习应用指南

- work ?
  - 观察 min/max/mean/stdev of episode returns
  - 观察实际性能提升



问题应对

- 采样效率问题，即学习一个简单的任务所需的经验数目相比与人类多了好几个数量级。
  - off-policy (experience replay)：即之前的经验存下来，之后反复使用
  - model-based learning: 学习或者构建模型，这样一方面能够更有方向性地探索从而减小盲目探索，另一方面在预测的时候利用模型规划使得行动的质量更高
  - prior: 从其他地方获取先验知识，把先验知识结合进来从而能够快速学习。

问题应对

- 奖励设置困难问题。现实问题中奖励需要大家自己定义，而定义一个好的奖励十分困难. 如果我们直接把目标总结成奖励，那么定义出来的奖励常常十分稀疏.
  - 去定义更好的解决方案，比如迭代更新的Gym 任务
  - 让它能自己学习到奖励，比如用imitation learning、inverse learning
  - 内在奖励，比如curiosity
  - 分层强化学习

问题应对

- 学习的最终效果不好。现实问题中奖励需要大家自己定义，而定义一个好的奖励十分困难．如果我们直接把目标总结成奖励，那么定义出来的奖励常常十分稀疏.
  - 针对特定的问题尽量建立针对该问题的模型
  - 提高模型的容量和抽象能力
  - 从专家的示范学习开始
  - 课程学习：先学习较为简单的情形，再学习更复杂的情形，即 curriculum learning

问题应对

- 对于特定环境的过拟合. 训练出来的agent基本上只能在训练的这个环境或者这个环境分布里面表现比较好，不是太能够泛化到其他的任务上。
  - 在更广的问题上学习prior，然后在特定问题上尽快学习好，比如transfer learning、meta learning
  - 逻辑推理能力
- Reality Gap，仿真中学到的策略transfer 到现实中往往效果不佳。
  - openai simulation randomization

前沿

- Model-free RL 主要目标是 Stable 和 Data Efficient
- Model-based RL 探讨 model 如何建模、建模之后如何学习或者规划
- Hierarchical RL（分层强化学习）
  - 完成一个复杂任务的模式，遇到一个复杂任务的时候，我们会把它拆解成一系列的小目标，然后逐个去实现这些小目标。
  - 解决动作空间、观察空间超复杂，并且奖励稀疏的复杂任务
- 反向强化学习, 从专家的示范中学习
- curiosity-driven learning, 不使用外在的奖励，仅使用curiosity 这种内在奖励
- WorldModels

框架推荐

人工智能现状浅谈

通过智能的机器，延伸和增强人类在改造自然、治理社会的各项任务中的能力和效率。

现状：炒作居多，落地很少，陷入困境。

- 炒作居多
  - 利益：不懂人工智能的人写报道、搞炒作宣传
  - 全面认识人工智能有困难
    - 人工智能是一个非常广泛的领域
      - 计算机视觉、自然语言处理、认知与推理、机器人学、博弈与伦理、机器学习
      - 六大领域目前较散，交叉发展
      - 横看成岭侧成峰，远近高低各不同

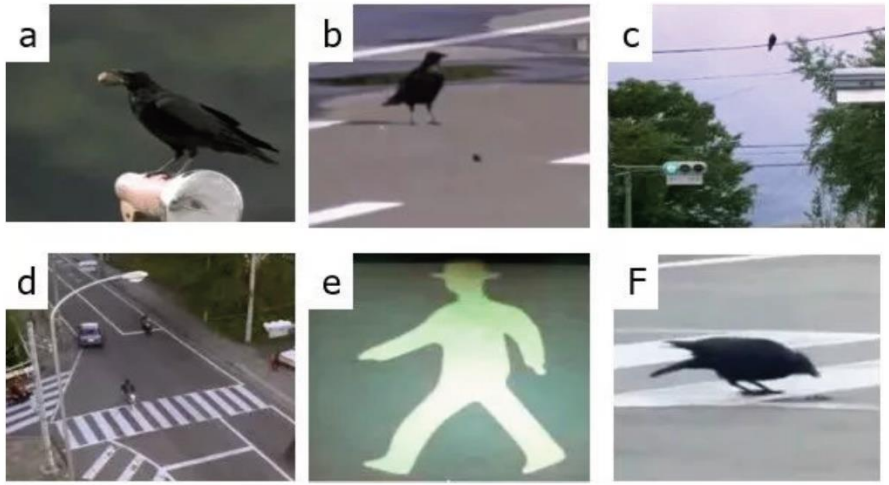
人工智能现状浅谈

炒作居多，落地很少，陷入困境。

- 炒作居多
  - 利益：不懂人工智能的人写报道、搞炒作宣传
  - 全面认识人工智能有困难
    - 人工智能是一个非常广泛的领域
    - 人工智能发展的断代现象
      - 自 1980 年代分化出以上几大学科以来，相互独立发展，基本抛弃了之前 30 年以逻辑推理与启发式搜索为主的研究方法，取而代之的是概率统计
    - 领域的分化与历史的断代客观上造成了“混乱”的局面

正视现实：落地很少，陷入困境

- 现阶段的鹦鹉式人工智能
  - 很强的模仿能力
  - 不明白说话的语境和语义，只是复杂的映射关系
  - 适用某些垂直应用
- 我们需要的乌鸦式通用人工智能
  - 感知、认知、推理、学习、和执行
- 瓶颈：研究思路问题，缺乏认知推理能力

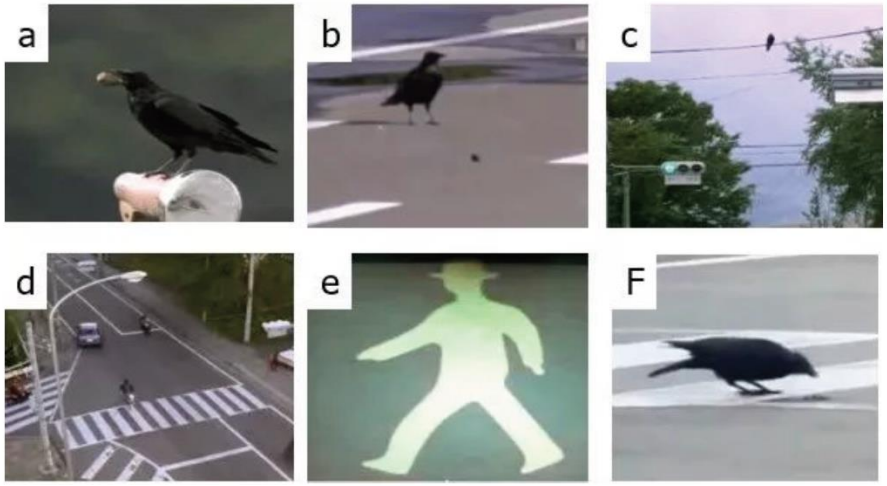


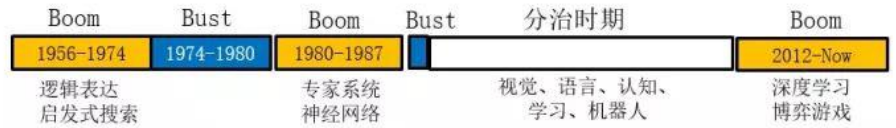
正视现实：落地很少，陷入困境

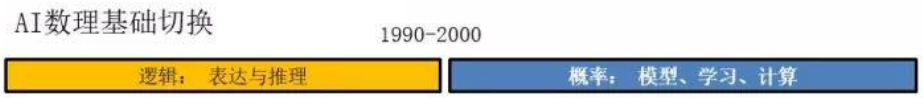
- 瓶颈：缺乏认知推理能力，缺乏物理的常识和社会的常识。

- 常识，生存的最基本的知识：
  - 使用频率最高
  - 可以举一反三推导出并且帮助获取其它知识。

- 趋势：学科之间开始兼并，出现一个理论架构把这些领域和问题统一起来







谢谢！