

Reinforcement Learning

Lecture 1. Introduction to RL

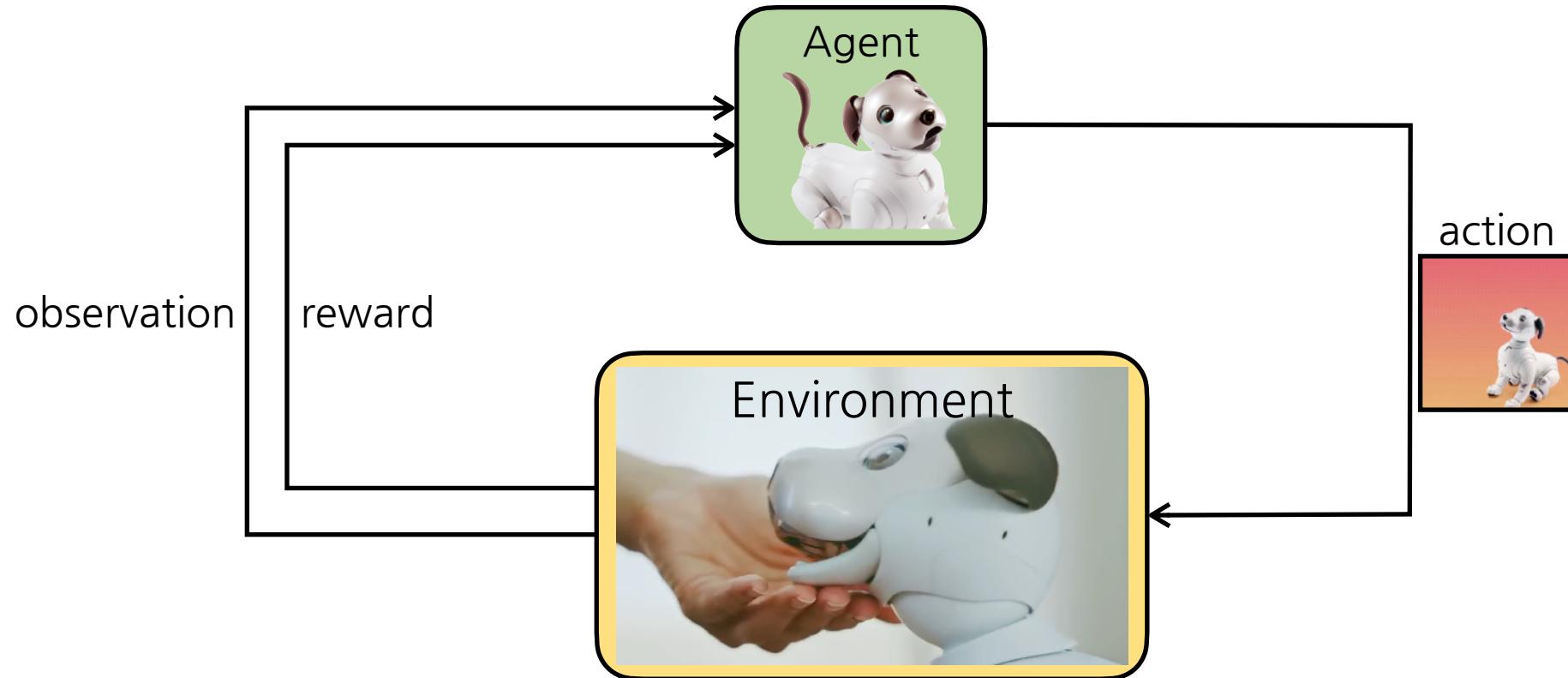
Sungjoon Choi, Korea University

Introduction

Syllabus

- Introduction to Reinforcement Learning
- Reinforcement Learning Applications
- Model-based Methods
- Model-free Methods
- Policy-based Methods
- Population-based Methods

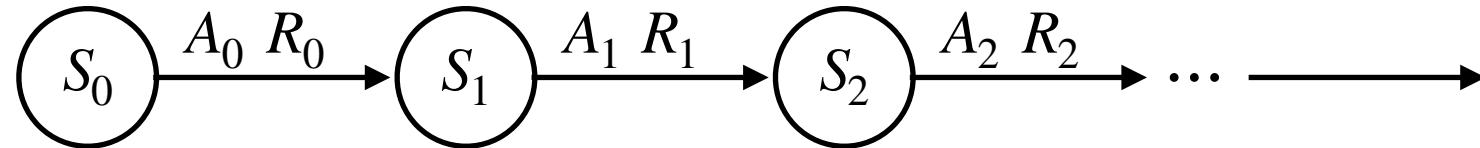
Reinforcement Learning



Key Question

What makes RL different from other ML methods (e.g., supervised or unsupervised learning)?

Markov Decision Process

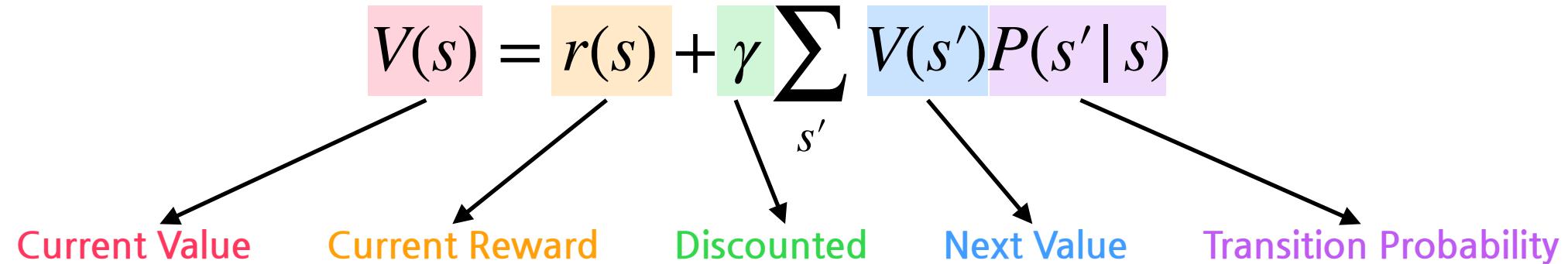


- This is the simplified world that RL considers.
- There are two very important questions:
 - What is the objective of "solving" reinforcement learning problems?
 - How can we define a state space, action space, and reward?
 - What are the limitations of using MDPs?

Bellman Equation

$$V(s) = r(s) + \gamma \sum_{s'} V(s') P(s' | s)$$

Current Value Current Reward Discounted Next Value Transition Probability

The diagram illustrates the Bellman Equation with five colored components: Current Value (red), Current Reward (orange), Discounted (green), Next Value (blue), and Transition Probability (purple). Arrows point from each component to its corresponding term in the equation.

Bellman Optimality Equation

$$Q^*(s, a) = \sum_{s'} \left[r(s, a, s') + \gamma \max_{a'} Q^*(s', a') \right] P(s' | s, a)$$

↓ ↓ ↓ ↓ ↓
 Current Value Current Reward Discounted Next Value Transition Probability

Policy Gradient Theorem

$$\eta(\pi_\theta) = \sum_s d(s)V_{\pi_\theta}(s)$$

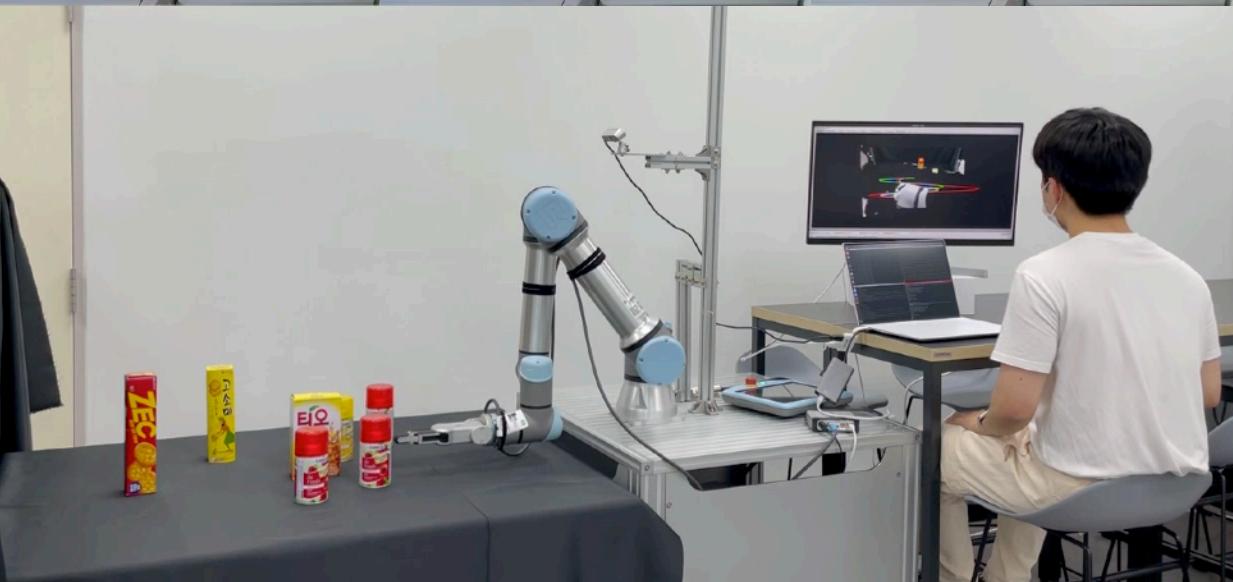
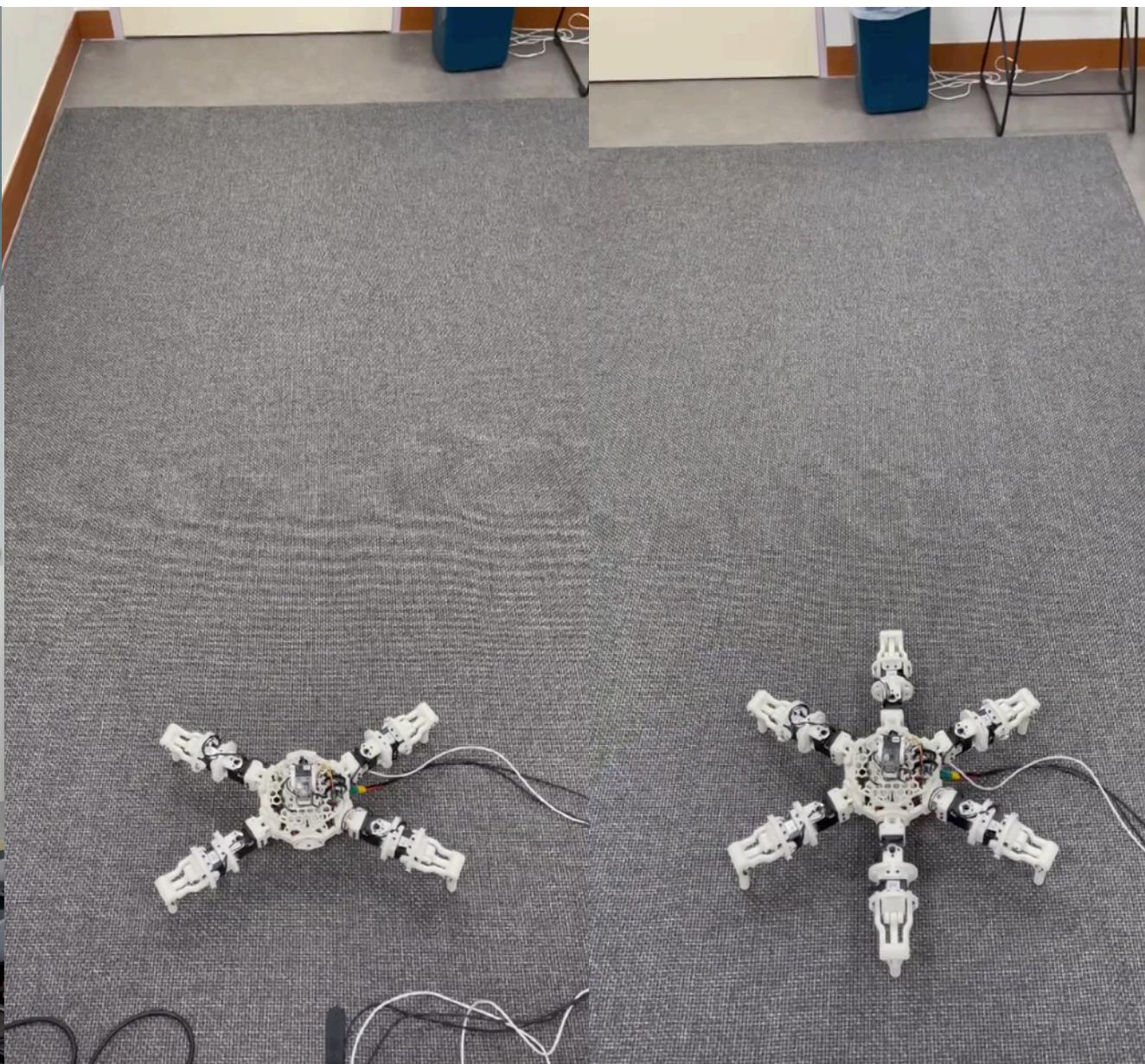
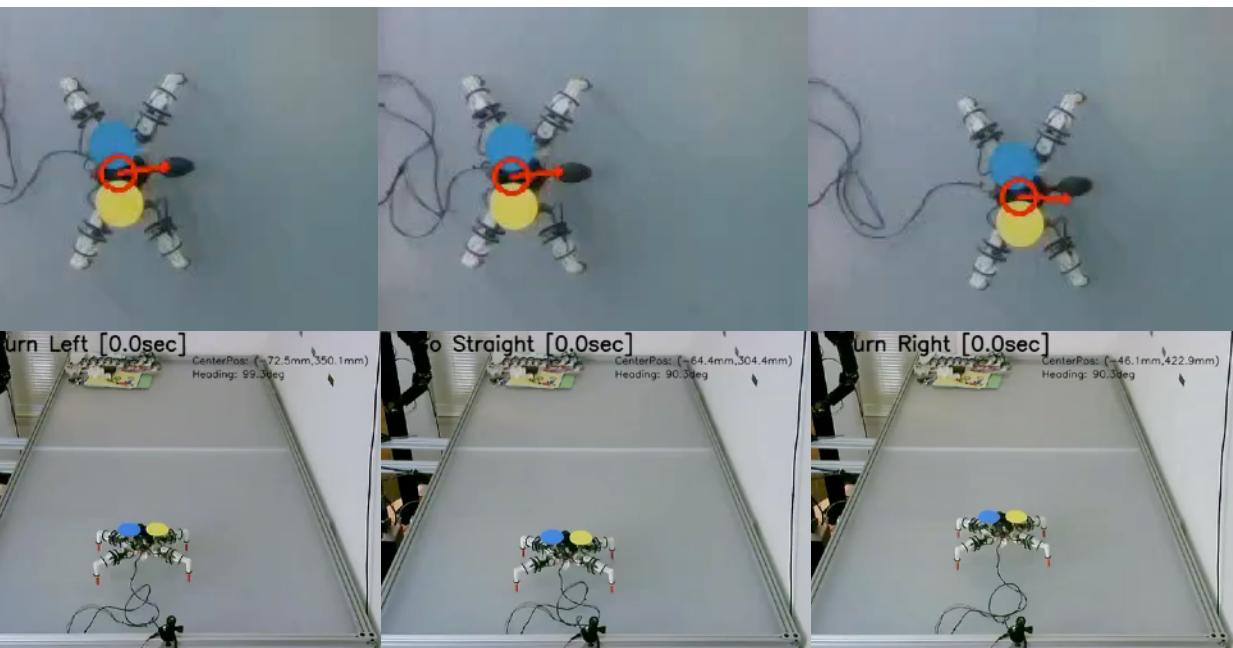
$$\nabla_\theta \eta(\pi_\theta) \approx \nabla_\theta \log \pi_\theta(a_t | s_t) Q_{\pi_\theta}(s_t, a_t)$$

↓ ↓ ↓

Return Gradient of Policy Value Function

RL Applications

RL in Our Lab



Contents

- Health Care
- Autonomous Vehicles
- Ride-sharing
- Natural Language Processing
- Robotics
- Smart Building Energy Management
- Games
- Marketing Advertising

Health Care

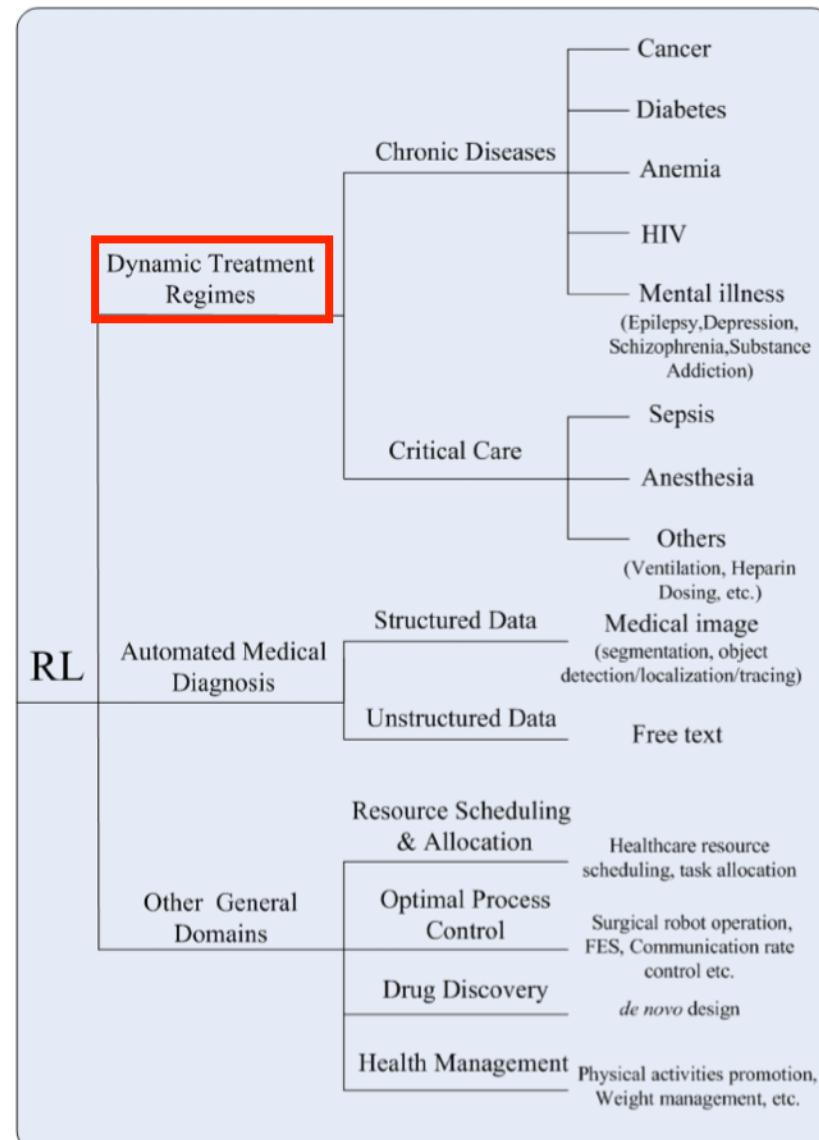
"Reinforcement Learning in Healthcare: A Survey," 2020
<https://arxiv.org/pdf/1908.08796>

RL in Health Care



b

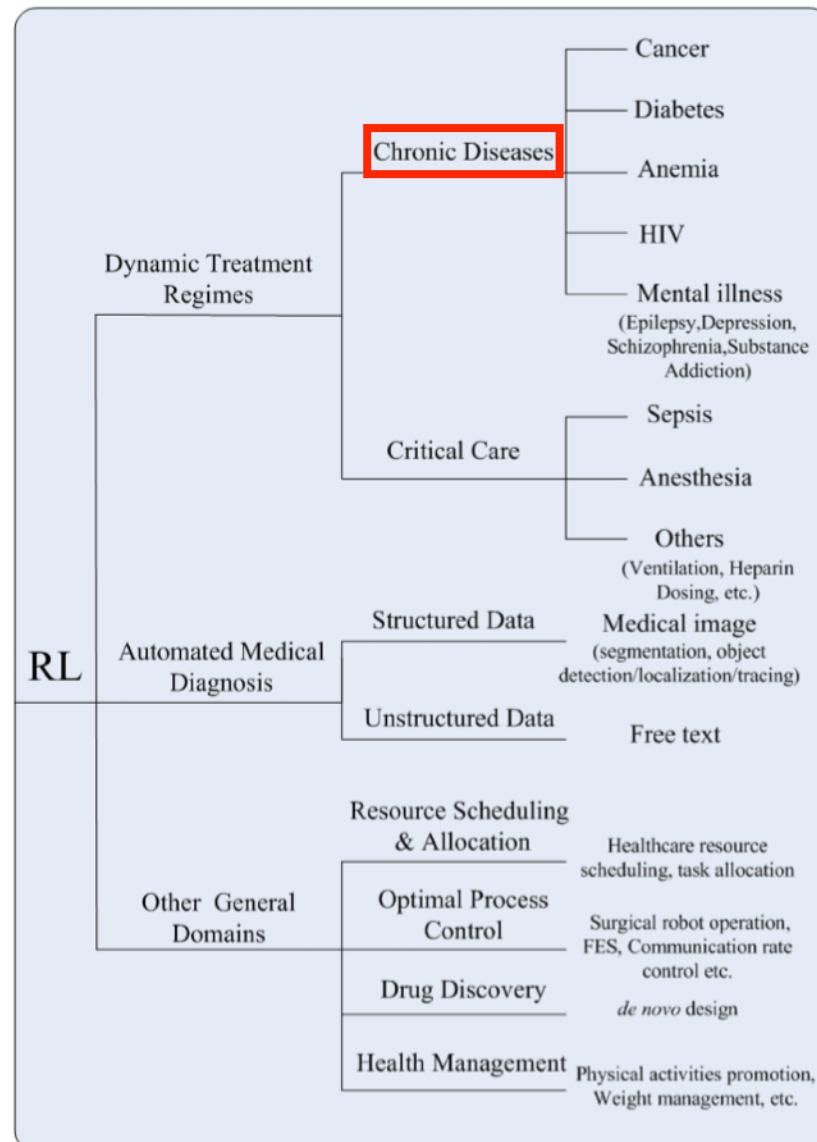
Dynamic Treatment Regimes



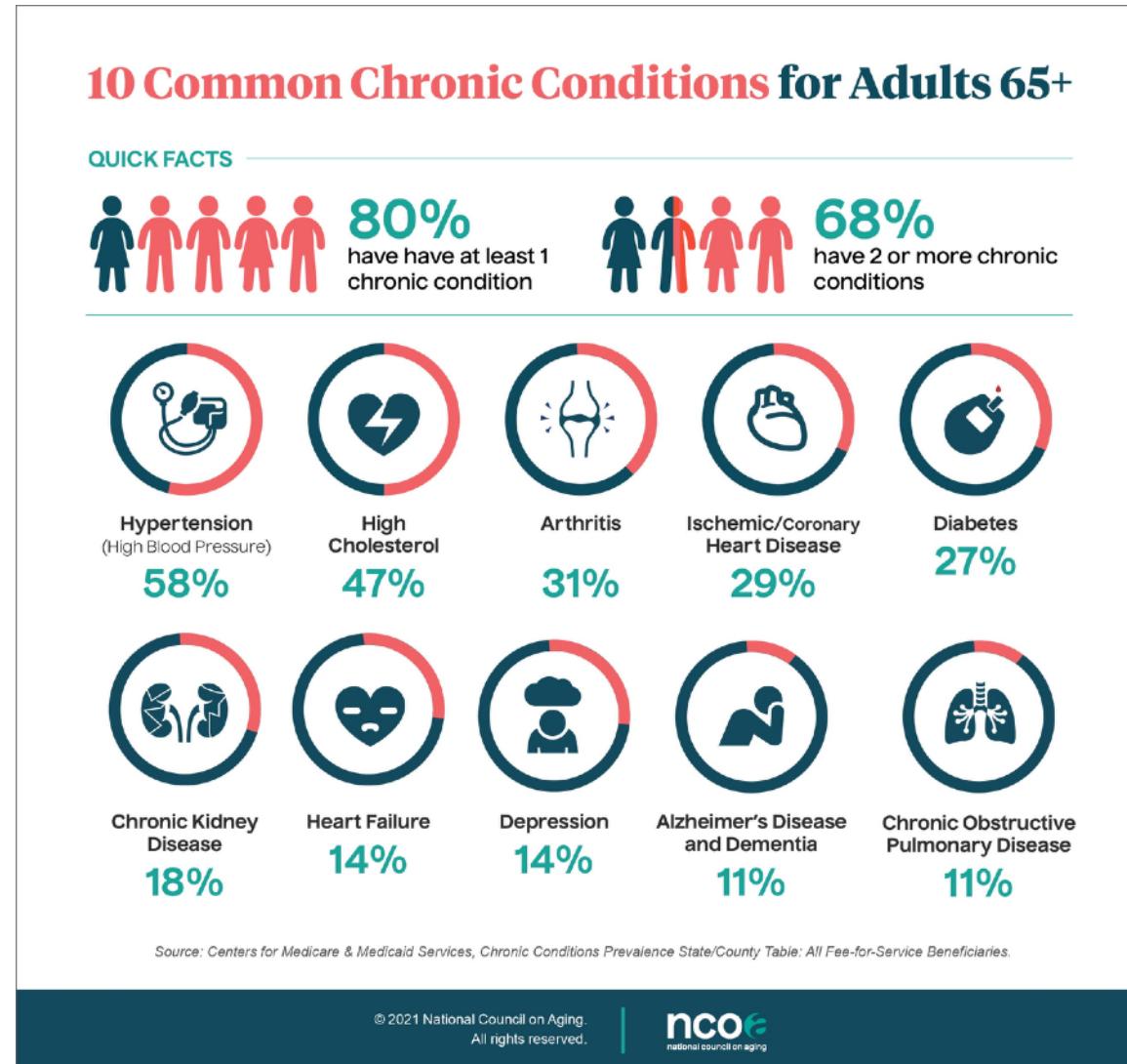
Dynamic Treatment Regimes

- Dynamic treatment regimes (**DTRs**) composed of a **sequence of decision rules** to determine the course of actions (e.g., treatment type, drug dosage, or reexamination timing) at a time point according to the current health status and prior treatment history of an individual patient.
- The design of **DTRs** can be viewed as a **sequential decision making problem** that fits into the RL framework well.

Chronic Diseases



Chronic Diseases



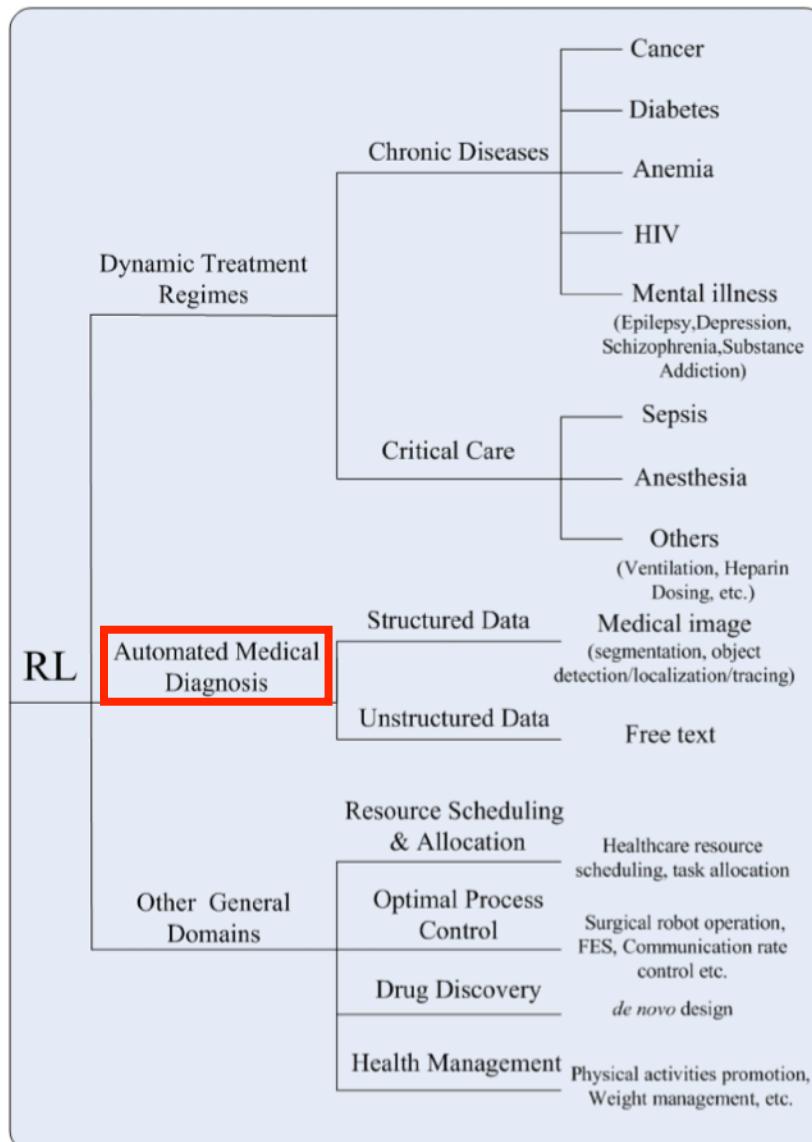
Chronic Diseases

- Chronic diseases are becoming the most pressing public health issue worldwide and they include diabetes, cancer, HIV infection, depression, and so forth.
- Long-term treatment of chronic diseases is made up of a **sequence of medical interventions** that must take into account the changing health status of a patient and adverse effects occurring from the previous treatment.
- In general, the relationship of treatment duration, dosage and type against the patient's response is **too complex to explicitly specified**.
 - RL has been utilized to automate the discovery and **generation of the optimal DTRs** in a variety of chronic diseases.

DTR in Cancer

Applications	References	Base Methods	Efficient Techniques	Representational Techniques	Data Acquisition	Highlights or Limits
Optimal chemotherapy drug dosage for cancer treatment	Zhao <i>et al.</i> [83]	Q-learning	BRL	N/A	ODE model	Using SVR or ERT to fit Q values; simplistic reward function structure with integer values to assess the tradeoff between efficacy and toxicity.
	Hassani <i>et al.</i> [84]	Q-learning	N/A	N/A	ODE model	Naive discrete formulation of states and actions.
	Ahn & Park [85]	NAC	N/A	N/A	ODE model	Discovering the strategy of performing continuous treatment from the beginning.
	Humphrey [86]	Q-learning	BRL	N/A	ODE model proposed in [83]	Using three machine learning methods to fit Q values, in high dimensional and subgroup scenarios.
	Padmanabhan [87]	Q-learning	N/A	N/A	ODE model	Using different reward functions to model different constraints in cancer treatment.
	Zhao <i>et al.</i> [88]	Q-learning	BRL (FQI-SVR)	N/A	ODE model driven by real NSCLC data	Considering censoring problem in multiple lines of treatment in advanced NSCLC; using overall survival time as the net reward.
	Fürnkranz <i>et al.</i> [52], Cheng <i>et al.</i> [89]	PI	N/A	PRL	ODE model proposed in [83]	Combining preference learning and RL for optimal therapy design in cancer treatment, but only in model-based DP settings.
Optimal fractionation scheduling of radiation therapy for cancer treatment	Akrour <i>et al.</i> [90], Busa-Fekete <i>et al.</i> [91]	PS	N/A	PRL	ODE model proposed in [83]	Using active ranking mechanism to reduce the number of needed ranking queries to the expert to yield a satisfactory policy without a generated model.
	Vincent [92]	Q-learning, SARSA(λ), TD(λ), PS	BRL (FQI-ERT)	N/A	Linear model, ODE model	Extended ODE model for radiation therapy; using hard constraints in the reward function and simple exploration strategy.
	Tseng <i>et al.</i> [93]	Q-learning	N/A	DRL (DQN)	Data from 114 NSCLC patients	Addressing limited sample size problem using GAN and approximating the transition probability using DNN.
	Jalalimanesh <i>et al.</i> [94]	Q-learning	N/A	N/A	Agent-based model	Using agent-based simulation to model the dynamics of tumor growth.
Hypothetical or generic cancer clinical trial	Jalalimanesh <i>et al.</i> [95]	Q-learning	N/A	MORL	Agent-based model	Formulated as a multi-objective problem by considering conflicting objective of minimising tumour therapy period and unavoidable side effects.
	Goldberg & Kosorok [96], Soliman [97]	Q-learning	N/A	N/A	Linear model	Addressing problems with censored data and a flexible number of stages.
	Yauney & Shah [98]	Q-learning	N/A	DRL (DDQN)	ODE model	Addressing the problem of unstructured outcome rewards using action-driven rewards.

Automated Medical Diagnosis



Automated Medical Diagnosis

- Medical diagnosis is a **mapping process** from a patient's information (e.g., treatment history, current signs, and symptoms) to an accurate clarification of a disease.
 - "It has been reported that diagnostic error accounts for as high as 10% of deaths and 17% of adverse events in hospitals."
 - It is normally formulated as a **supervised classification** problem, heavily relying on a large number of annotated samples in order to infer and predict possible diagnoses.
- To overcome the sample efficiency, researchers are interested in formulating the diagnostic inferencing problem as a **sequential decision-making process** and using RL to leverage a small amount of labeled data.

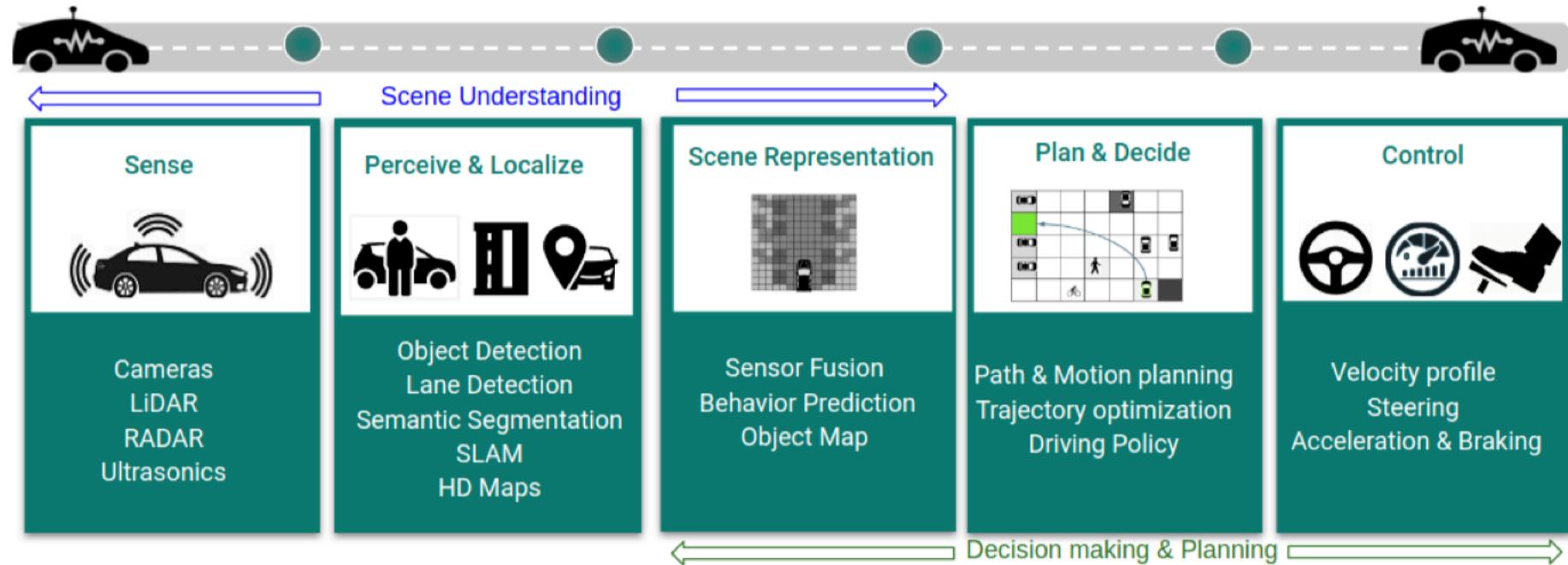
Challenges in applying RL to Health Care

- State/Action Engineering
 - How to collect and pre-process proper medical data and summarize them into **manageable state representation**?
- Reward Formulation
 - The reward may be at the **core of an RL process**, and it plays the most crucial role in the success of RL.
- Exploration Strategies
 - When dealing with patients, we **cannot bring them back to life** when they have been given fatal medical treatments.
- Credit Assignment
 - It decides which actions are **responsible** for the learning outcome after a sequence of decisions.

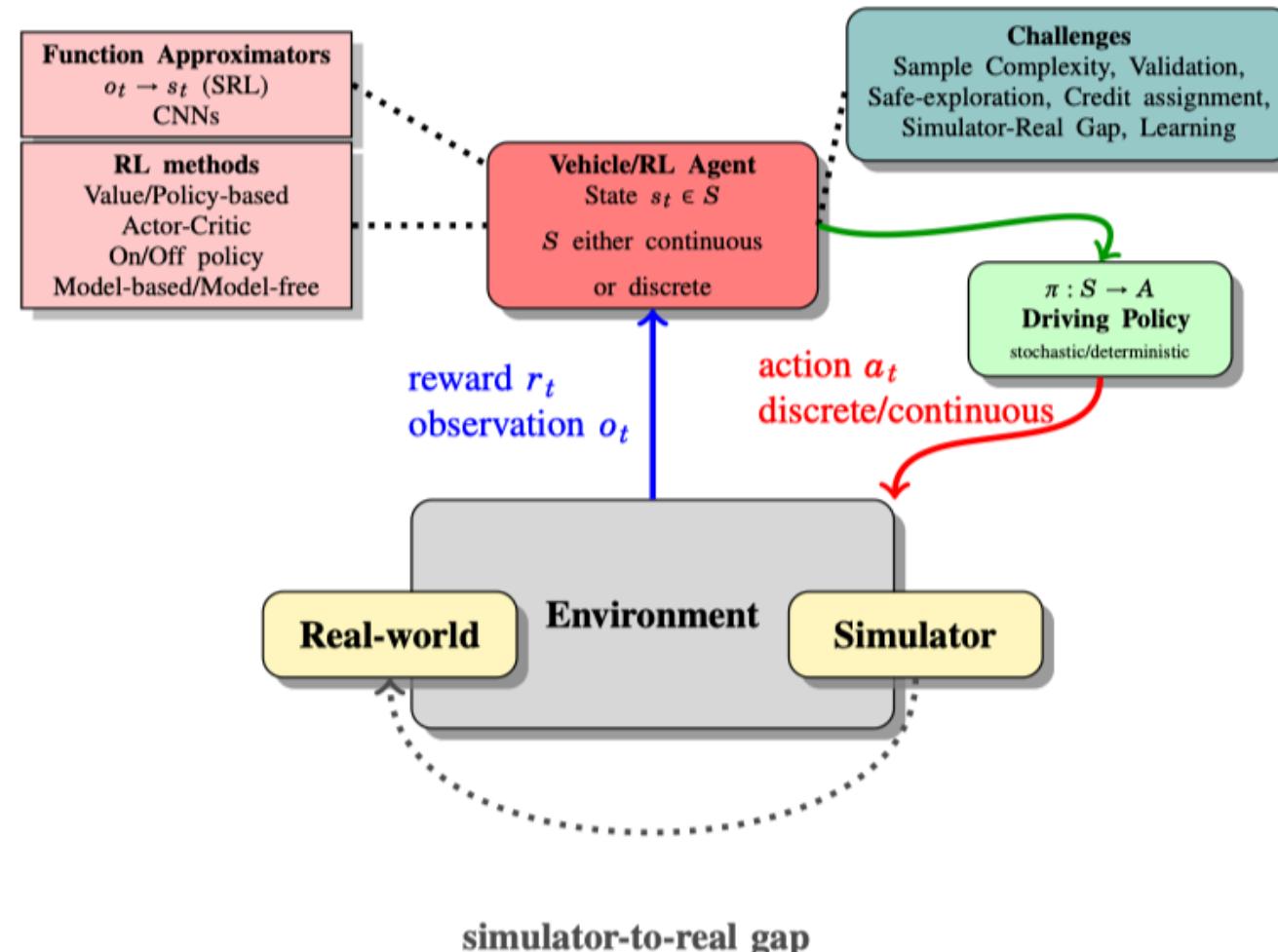
Autonomous Vehicles

Deep Reinforcement Learning for Autonomous Driving: A Survey, 2021
<https://arxiv.org/pdf/2002.00444>

Autonomous Driving System



Sim-to-Real Gap



State / Action / Reward

- **State space**

- position, heading, and velocity of ego-vehicle
- other obstacles in the sensor view
- lane information
- raw sensor data (e.g., LiDar)

- **Action space**

- steering angle and throttle

- **Reward**

- desired speed of the ego-vehicle
- lane keeping behavior
- safety

RL for Autonomous Driving

- Autonomous driving tasks where RL could be applied include:
 - controller optimization
 - path planning and trajectory optimization
 - motion planning and dynamic path planning
 - development of high-level driving policies for complex navigation tasks
 - scenario-based policy learning for highways, intersections, merges and splits
 - reward learning with inverse reinforcement learning from expert data for intent prediction for traffic actors such as pedestrian, vehicles
 - learning policies that ensures safety and perform risk estimation.

Applications

AD Task	(D)RL method & description	Improvements & Tradeoffs
Lane Keep	1. Authors [82] propose a DRL system for discrete actions (DQN) and continuous actions (DDAC) using the TORCS simulator (see Table V-C) 2. Authors [83] learn discretised and continuous policies using DQNs and Deep Deterministic Actor Critic (DDAC) to follow the lane and maximize average velocity.	1. This study concludes that using continuous actions provide smoother trajectories, though on the negative side lead to more restricted termination conditions & slower convergence time to learn. 2. Removing memory replay in DQNs help for faster convergence & better performance. The one hot encoding of action space resulted in abrupt steering control. While DDAC's continuous policy helps smooth the actions and provides better performance.
Lane Change	Authors [84] use Q-learning to learn a policy for ego-vehicle to perform no operation, lane change to left/right, accelerate/decelerate.	This approach is more robust compared to traditional approaches which consist in defining fixed way points, velocity profiles and curvature of path to be followed by the ego vehicle.
Ramp Merging	Authors [85] propose recurrent architectures namely LSTMs to model longterm dependencies for ego vehicles ramp merging into a highway.	Past history of the state information is used to perform the merge more robustly.
Overtaking	Authors [86] propose Multi-goal RL policy that is learnt by Q-Learning or Double action Q-Learning(DAQL) is employed to determine individual action decisions based on whether the other vehicle interacts with the agent for that particular goal.	Improved speed for lane keeping and overtaking with collision avoidance.
Intersections	Authors use DQN to evaluate the Q-value for state-action pairs to negotiate intersection [87],	Creep-Go actions defined by authors enables the vehicle to maneuver intersections with restricted spaces and visibility more safely
Motion Planning	Authors [88] propose an improved A^* algorithm to learn a heuristic function using deep neural networks over image-based input obstacle map	Smooth control behavior of vehicle and better performance compared to multi-step DQN

Simulators

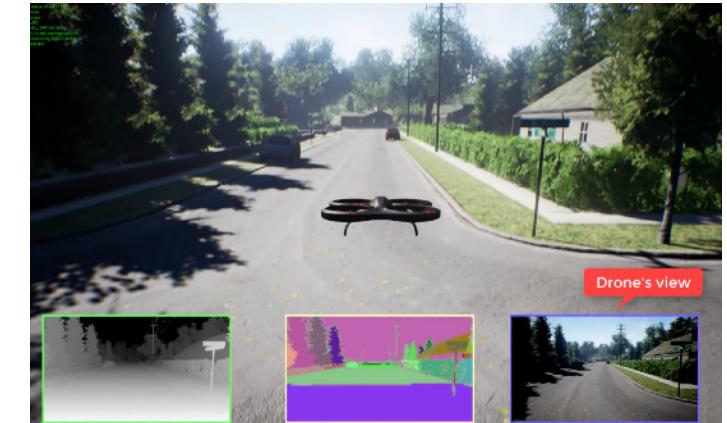
CARLA



TORCS



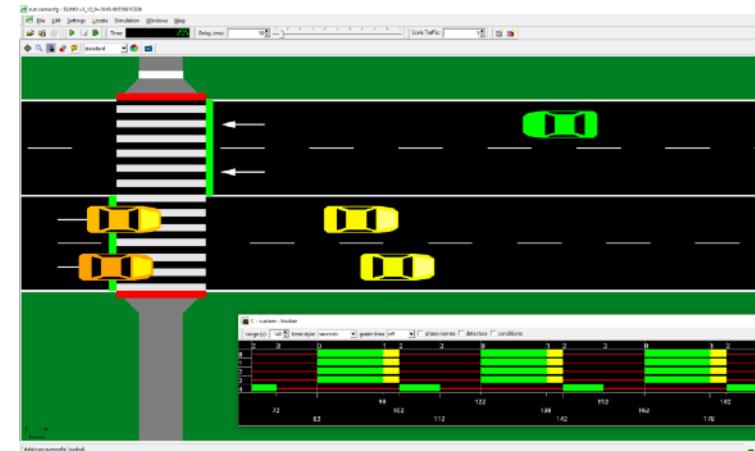
AIRSIM



GAZEBO



SUMO



DeepDrive



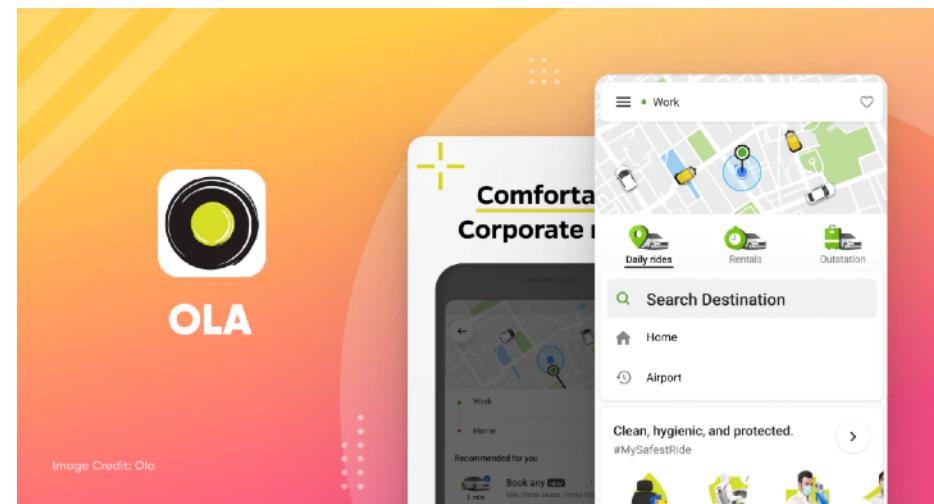
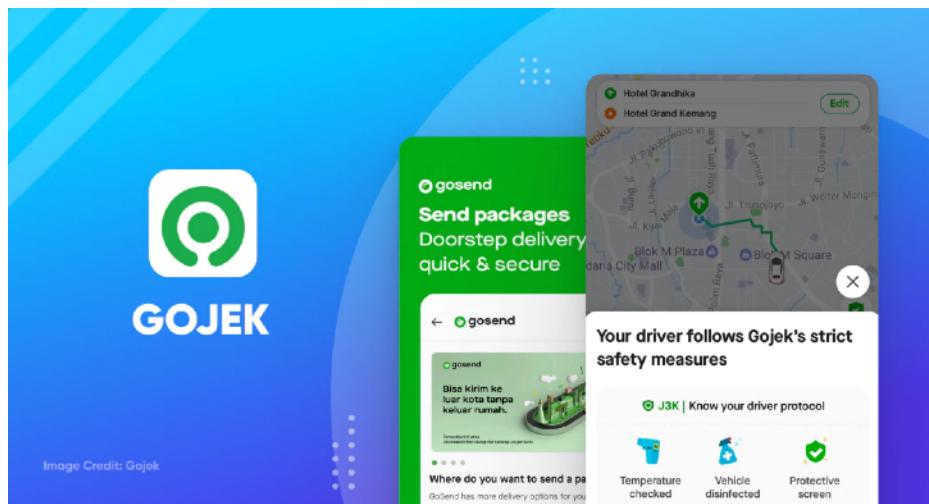
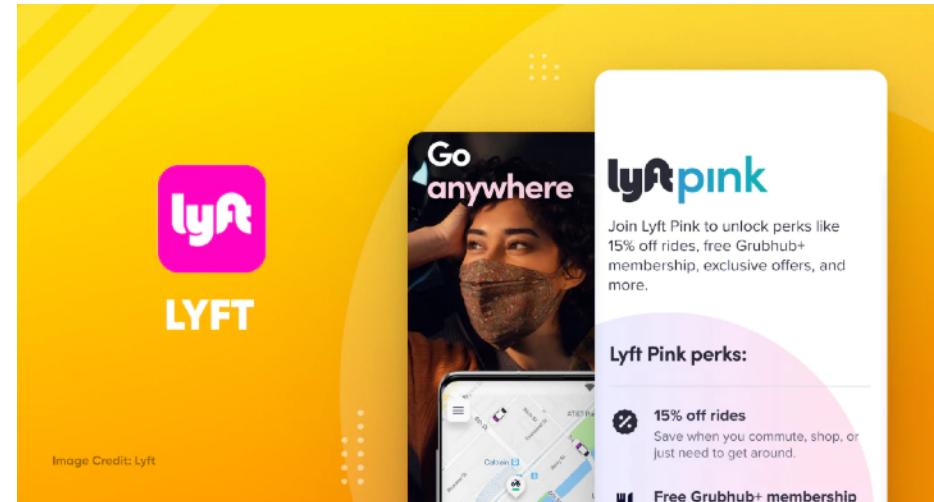
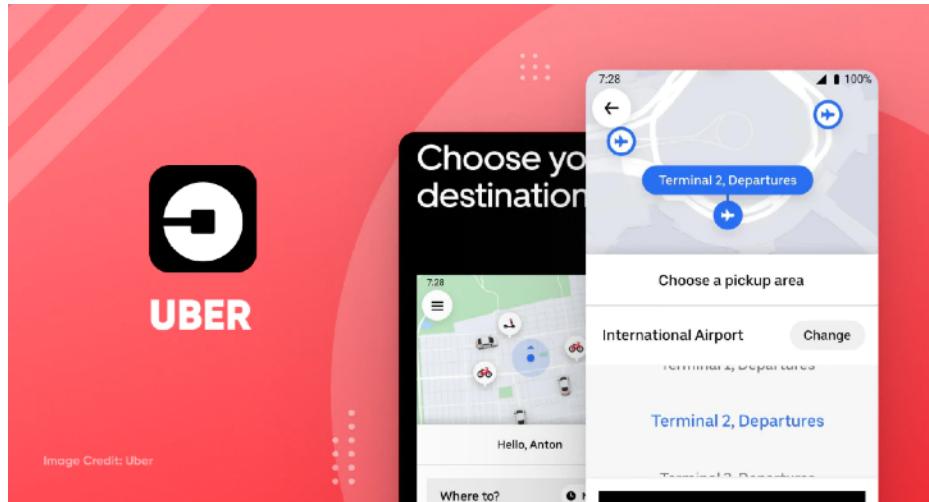
Challenges

- Validating RL systems
 - How to properly **validate** the performance of the learned policy
- Bridging the **simulation-reality gap**
 - Train in simulation, test in real world
- Sample efficiency
- Exploration issues with Imitation
- Intrinsic reward
 - Curiosity-based learning
- Incorporating safety
- Multi-agent reinforcement learning

Ride-sharing

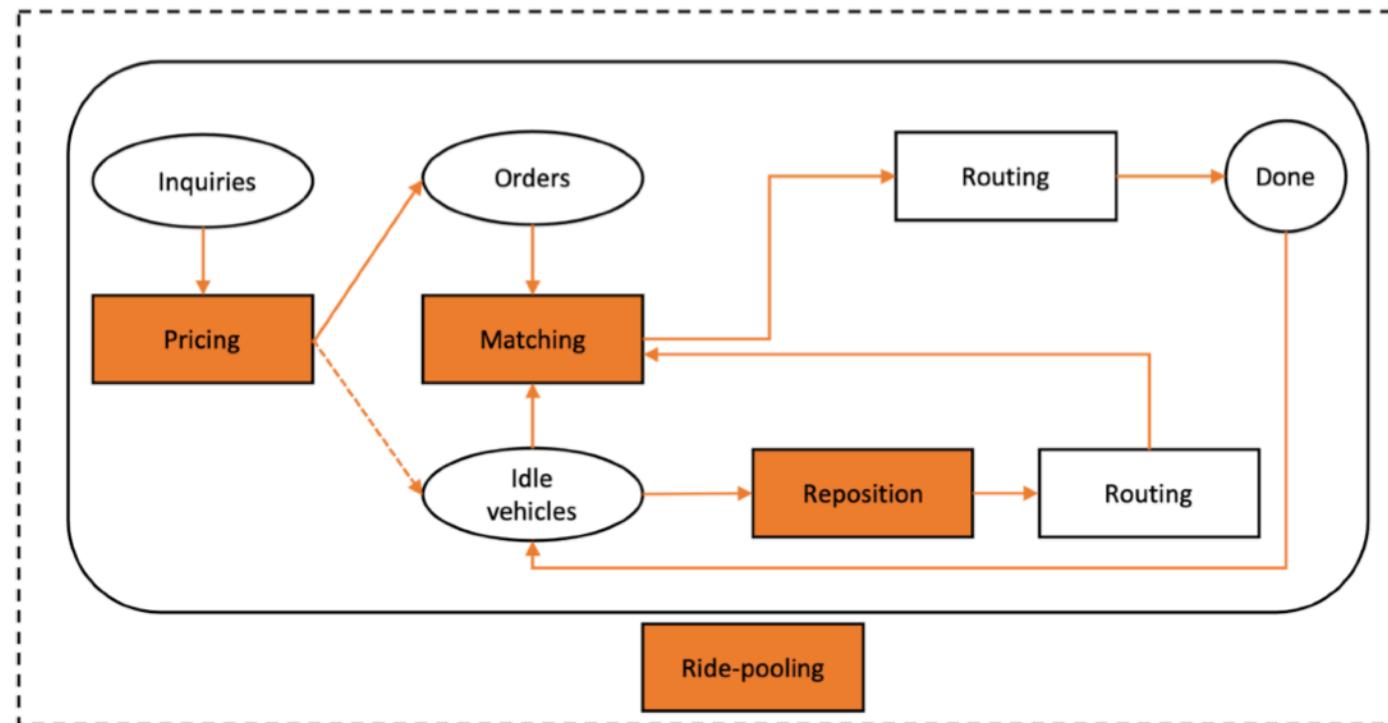
"Reinforcement Learning for Ride-sharing: A Survey," 2021
<https://arxiv.org/pdf/2105.01099>

Ride-Sharing



Ride-Sharing

- The emergence of ride-sharing, led by companies such as **DiDi**, **Uber**, and **Lyft**, has revolutionized the form of personal mobility. It is projected that the global ride-share industry will grow to a total market value of **\$218 billion** by 2025.



RL for Ride-Sharing

- Pricing
 - The ride-sharing marketplace is complex due to its **two-sided nature** (price affects drivers, and drivers affect pricing).
- Online matching
 - The RL literature for ride-sharing matching typically aims to optimize (i.e., reward) the **platform revenue** and the **service quality** (e.g., response rate and fulfillment rate).
- Vehicle repositioning
- Ride-pooling (Carpool)
 - Ride-pooling optimization concerns with matching, repositioning, and routing.
 - The typical objectives are passenger wait time, detour delay, and effective trip distance (the travel distance between the origin and the destination with and without ride-pooling).

Challenges

- Ride-pooling
 - Most of the existing RL methods assume that the action set is predetermined, hence, some make only high-level decisions (e.g., repositioning). Then, additional optimization is required to match the given high-level decision.
- Joint Optimization
 - The ride-share platform is an integrated system, so **joint optimization of multiple decision modules** leads to better solutions.
- Heterogeneous Fleet
 - In ride-sharing, a heterogeneous fleet means **multiple types of agents** with different states and action spaces (e.g., electric vehicles, autonomous vehicles, etc.).
- Sim2Real
 - Simulation is essential in solving RL problems, but none of the existing public simulators supports pricing decisions.

Natural Language Processing

"Training language models to follow instructions with human feedback," 2022
<https://arxiv.org/abs/2203.02155>

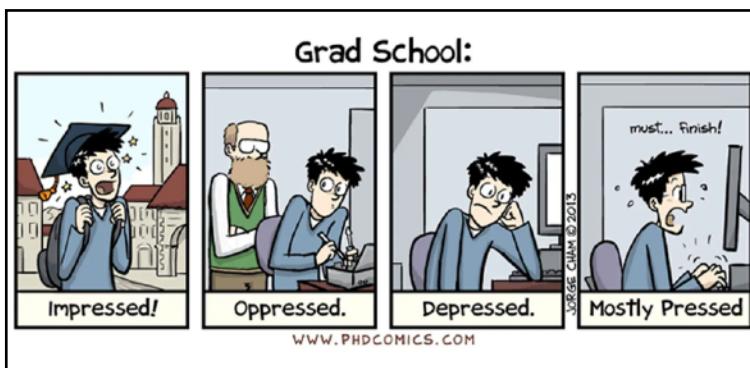
Large Language Models (LLMs)

Bohemian rhapsody

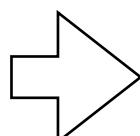


https://www.nme.com/en_au/news/film/brian-may-bohemian-rhapsody-sequel-3014651

Graduate Student



<https://www.wired.co.uk/article/jorge-cham-phd>



Bohemian rhapsody of a Graduate Student

su Could you write lyrics for Bohemian rhapsody of a graduate student version?

AI Sure, I'd be happy to create a graduate student version of Bohemian Rhapsody! The following lyrics are meant to be light-hearted and reflect some of the experiences of graduate students. I'm going to attempt to stay as close as possible to the original song's structure.

(Verse 1)

Is this a real life? Is it just academia?
Caught in a thesis, no escape from this encyclopedia
Open your books, look up to the skies and see
I'm just a poor grad, I need no sympathy
Because I'm easy come, easy go, work high, sleep low
Anyway the wind blows doesn't really matter to me, to me

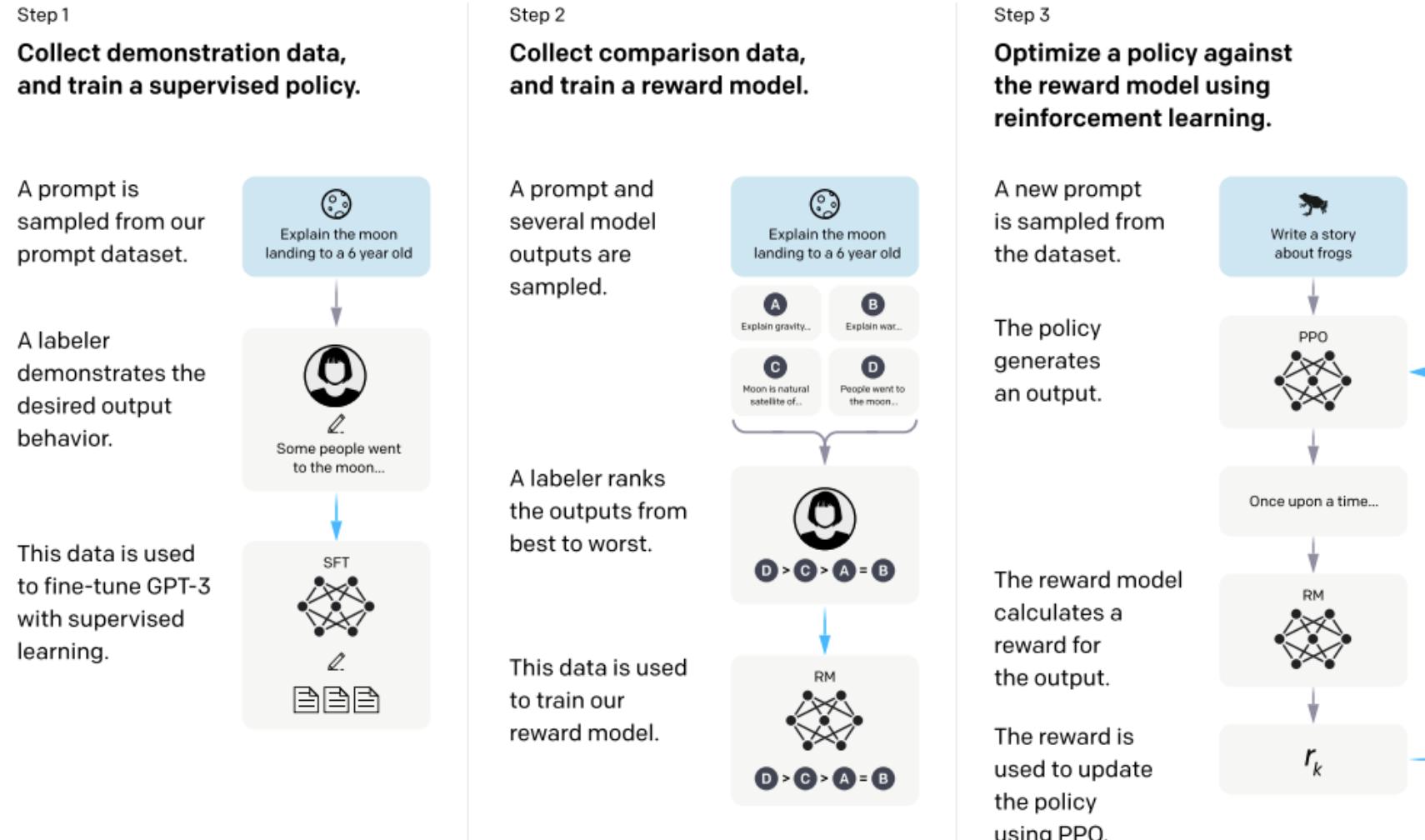
(Mama...)

Mama, just killed my sleep
Put a coffee against its head, pulled the trigger, now it's dead
Mama, life had just begun
But now I've gone and thrown it all away

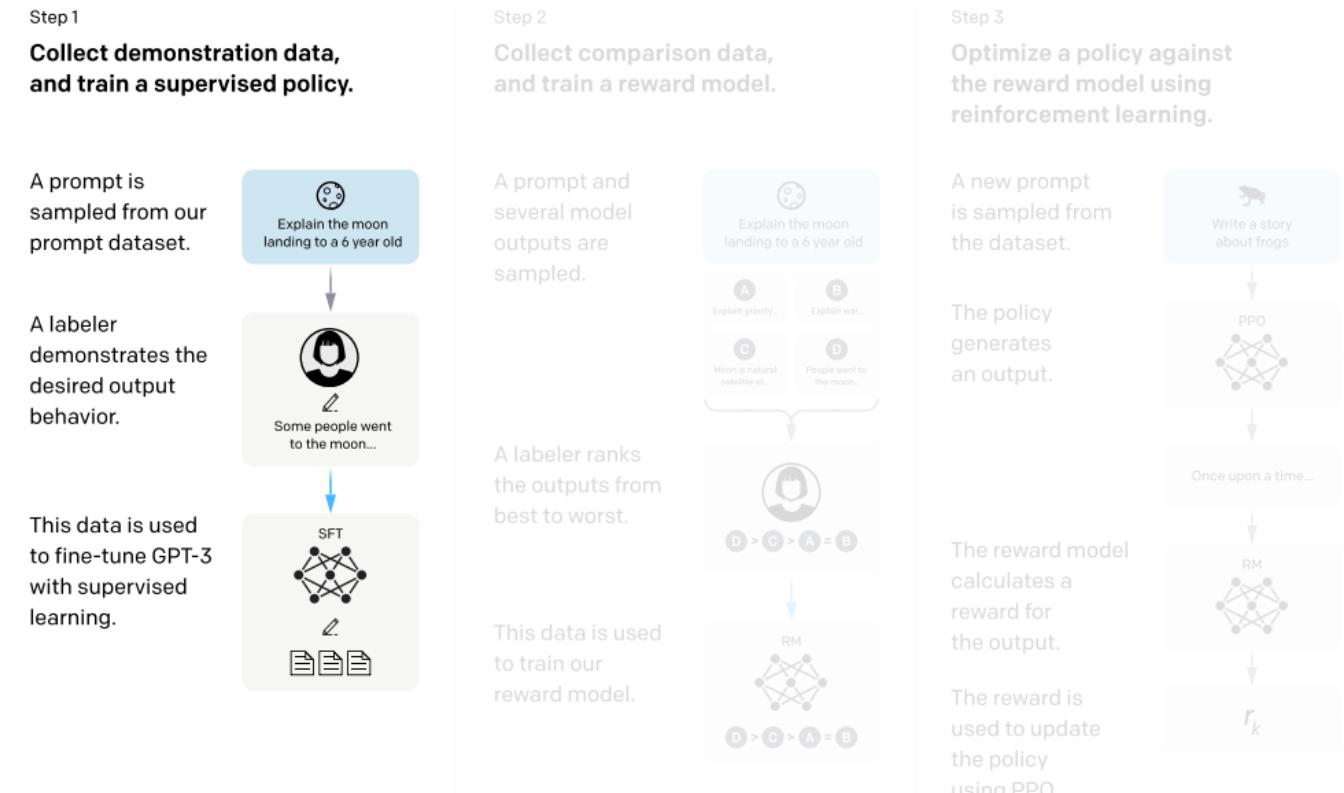
(Chorus)

Mama, ooh, didn't mean to make you cry
If I'm not done this time tomorrow
Carry on, carry on as if nothing really matters

Post-GPT series (InstructGPT, ChatGPT, GPT-4)

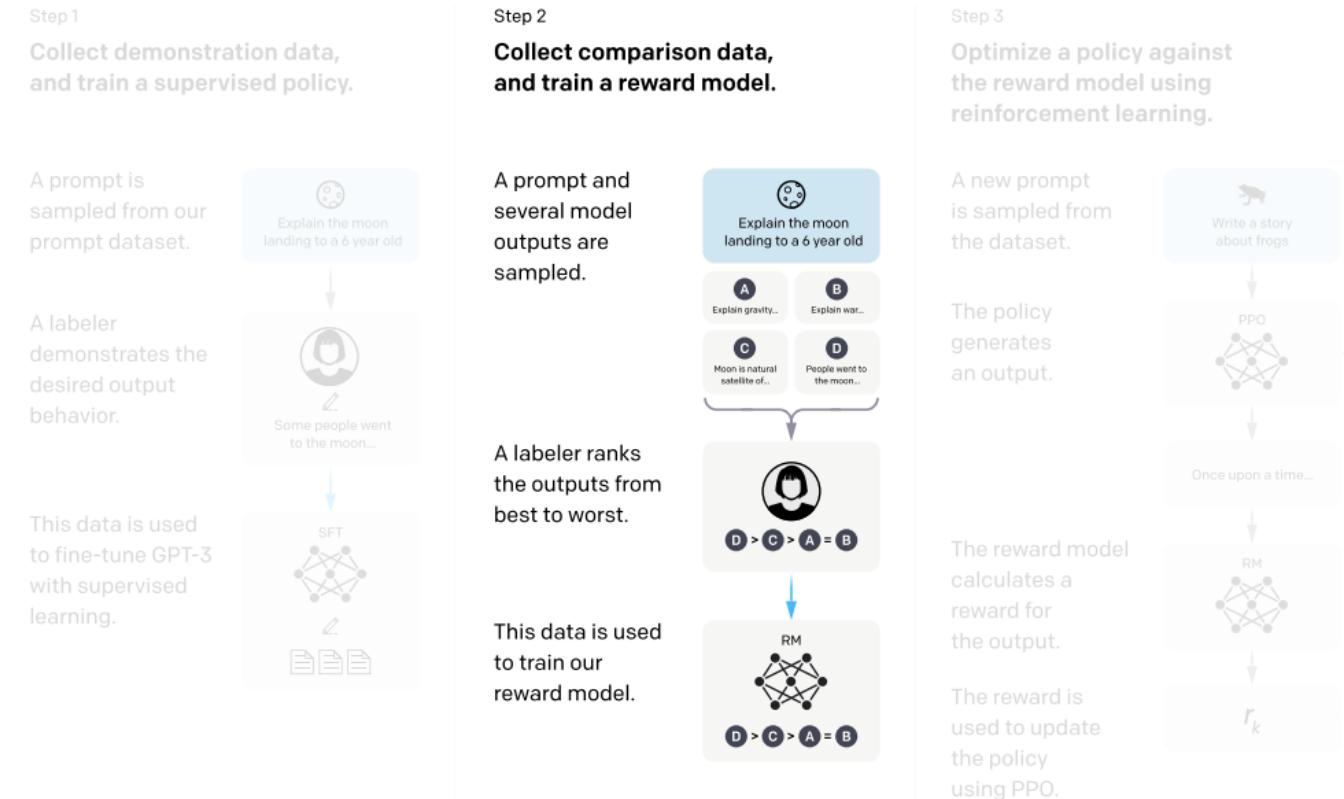


Post-GPT series (InstructGPT, ChatGPT, GPT-4)



- GPT series usually start with the GPT-3 pre-trained language model.
- Supervised fine-tuning (SFT)
 - The prompt dataset consists of text prompts submitted to the OpenAI API.
 - Human labeler demonstrations are used to fine-tune GPT-3 to get **SFT models**.

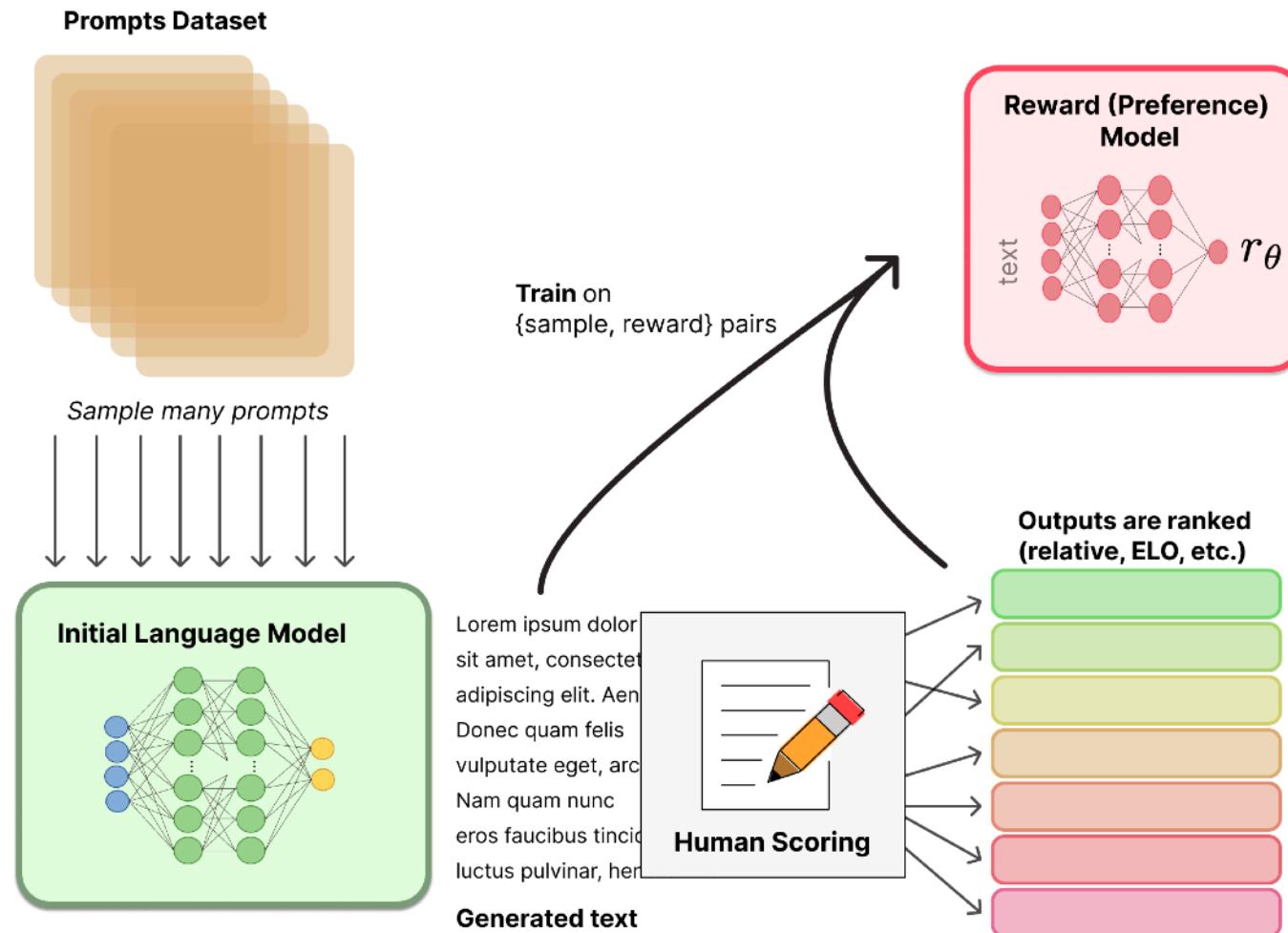
Post-GPT series (InstructGPT, ChatGPT, GPT-4)



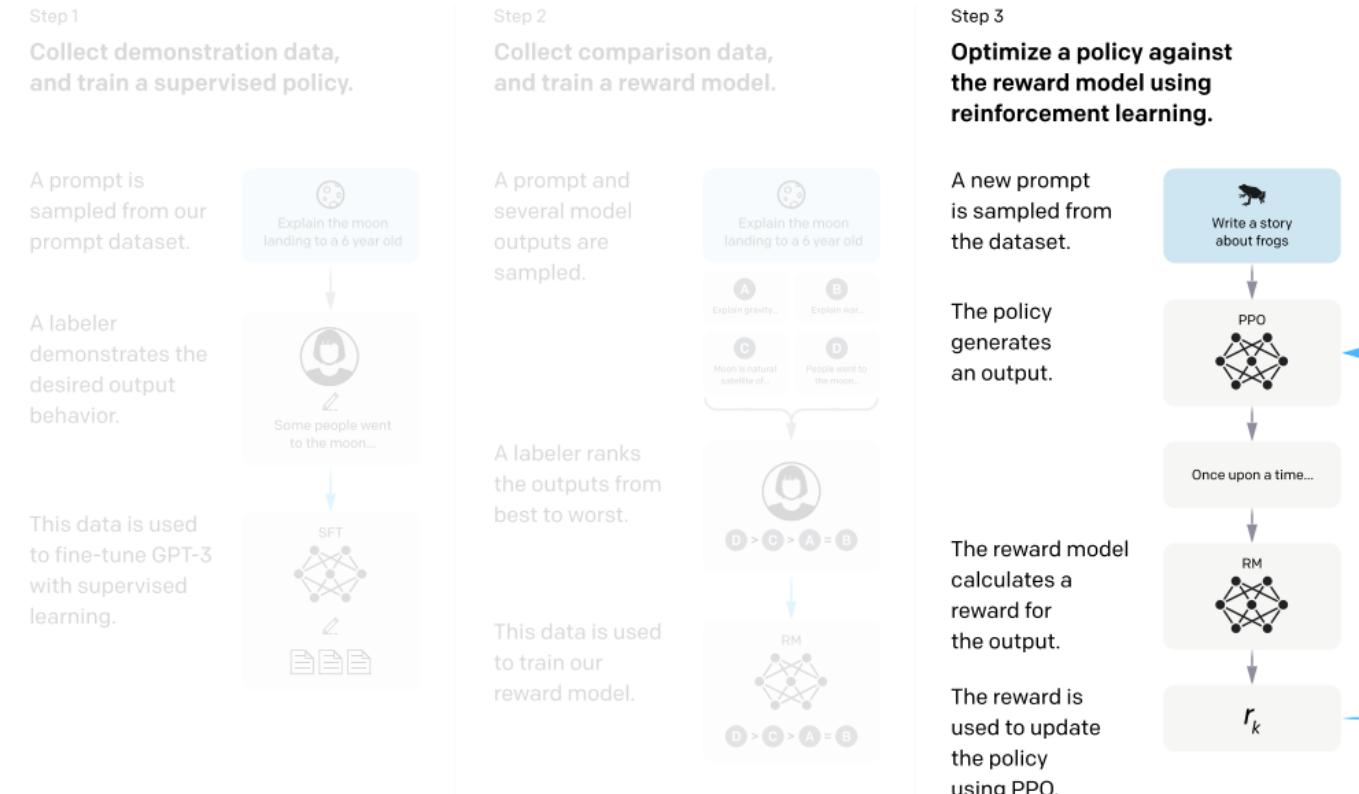
- Reward modeling (RM)

- The comparison dataset is additionally collected by comparing two different outputs of models.
- The **reward model** is trained to predict human preferences.

Reward Model Training



Post-GPT series (InstructGPT, ChatGPT, GPT-4)



- Reinforcement Learning from Human Feedback (RLHF)
 - SFT models (from the first step) are fine-tuned using the **reward function** (from the second step).
 - The model (e.g., InstructGPT) completes the given prompt, and the reward model outputs the estimated reward. The model is fine-tuned with PPO to maximize the estimated reward.

Alignment?

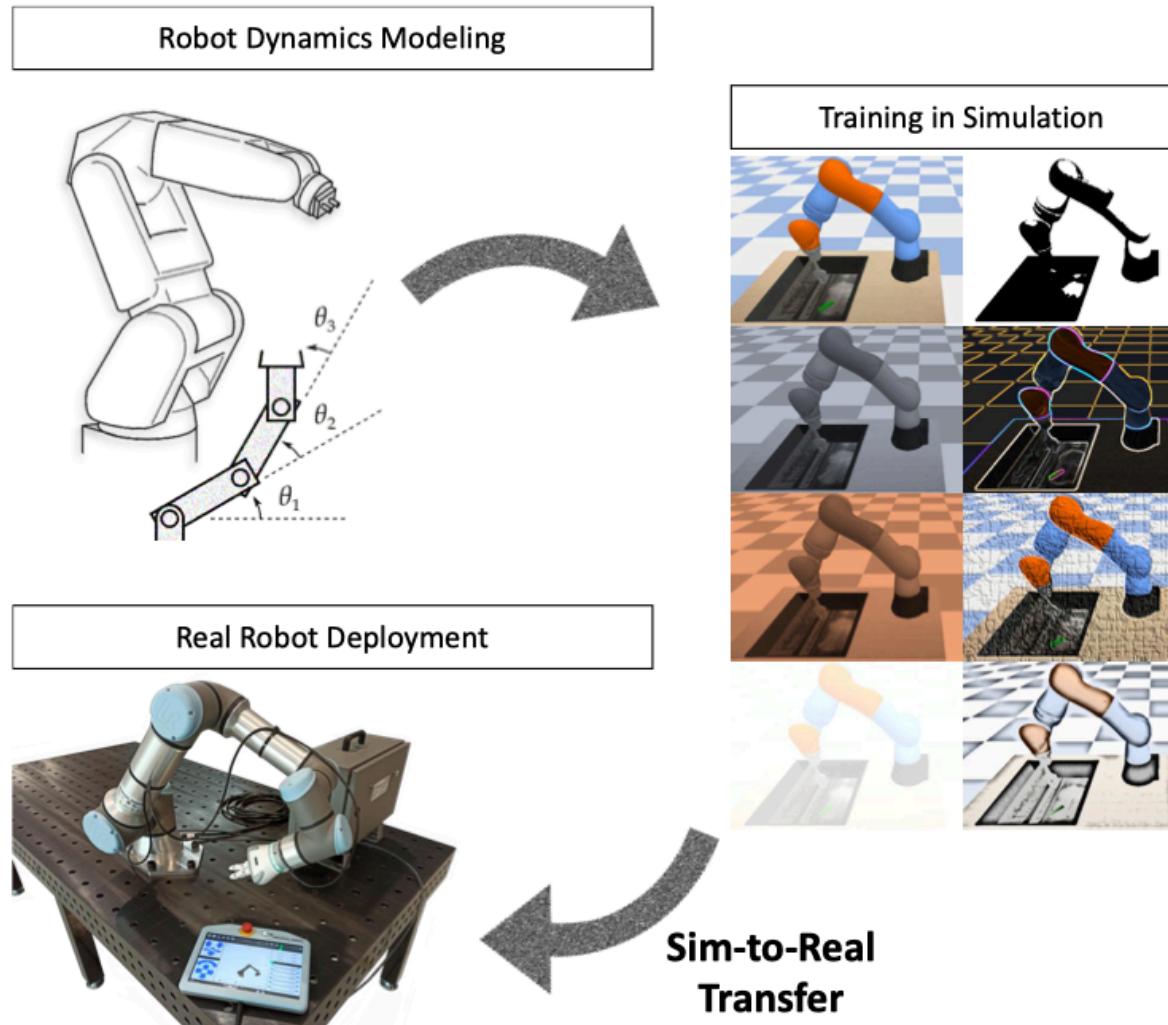


"We make progress on **aligning language models** by training them to act in accordance with the user's intention."

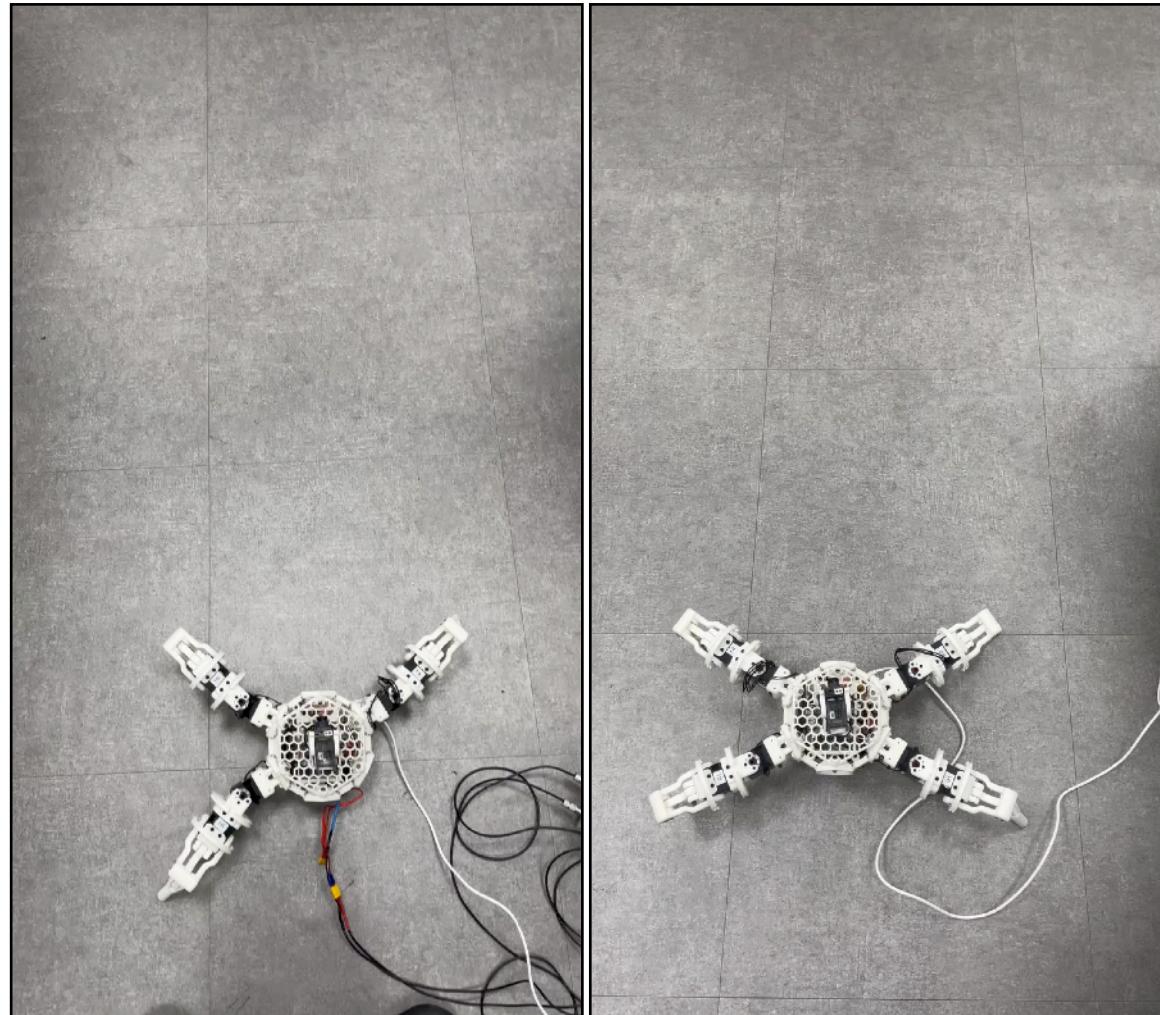
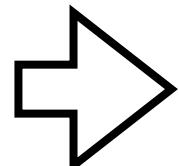
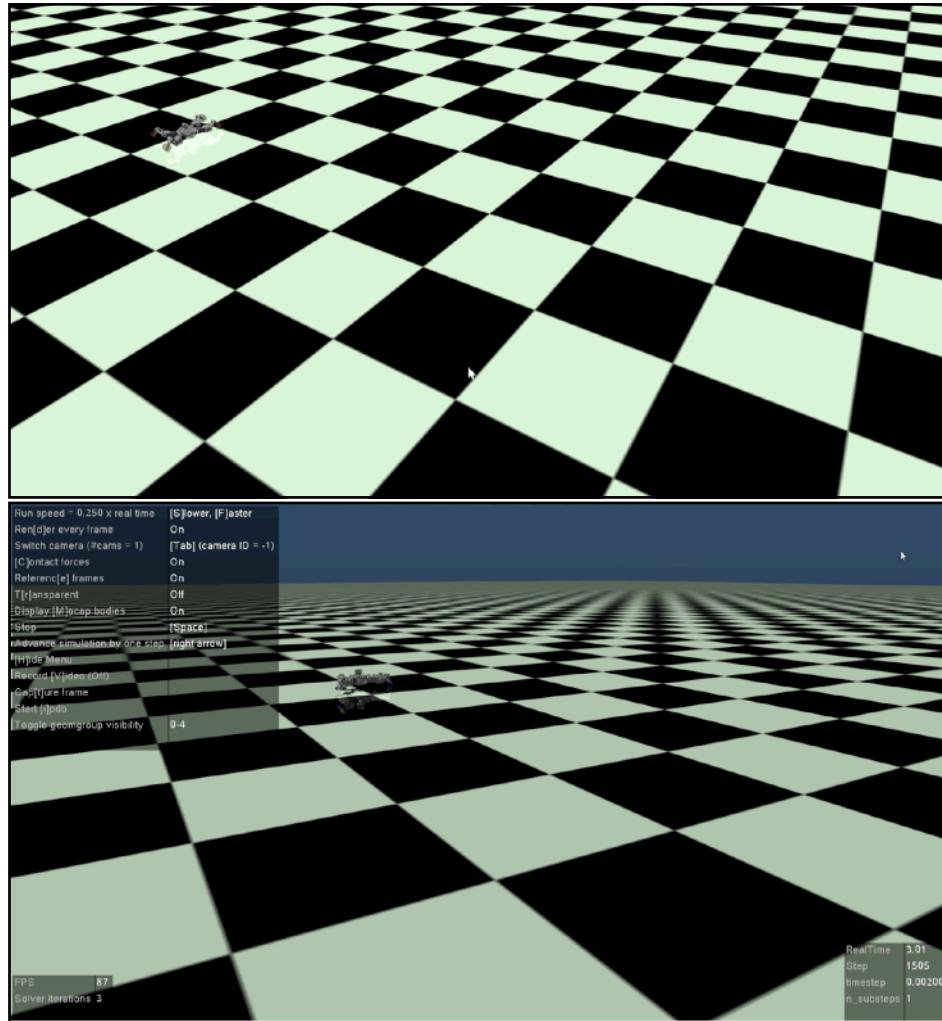
Robotics

"Sim-to-Real Transfer in Deep Reinforcement Learning for Robotics: a Survey," 2021
<https://arxiv.org/pdf/2009.13303>

Sim2Real



Sim2Real



A LOT OF PAPERS

TABLE I: Classification of the most relevant publications in Sim2Real Transfer.

	Description	Sim-to-real transfer and learning details	Multi-agent learning	Simulator / Engine	Knowledge Transfer	Learning Algorithm	Real Robot/Platform	Application
Balaji et al. [30]	DeepRacer: an educational autonomous racing platform.	Random colors and parallel domain randomization	✓(sim only) Distr. rollout	Gazebo RoboMaker	✗	PPO	DeepRacer 4WD 1:18 Car	Autonomous racing
Traore et al. [12]	Continual RL with policy distillation and sim-to-real transfer.	Continual learning with policy distillation.	✗	PyBullet	✓Multi-task Distillation	PPO2	Small mobile platform	Robotic navigation
Kaspar et al. [31]	Sim-to-real transfer for RL without Dynamics Randomization.	System identification and a high-quality robot model.	✗	PyBullet	✗	SAC	KUKA LBR iiwa +WSG50 gripper	Peg-in-Hole manipulation
Matas et al. [6]	Sim-to-real RL for deformable object manipulation.	Stochastic grasping and domain randomization.	✓(sim)	PyBullet	✗	DDPGfID	7DOF Kinova Mico Arm	Dexterous manipulation
Witman et al. [32]	Sim-to-real RL for thermal effects of an atmospheric pressure plasma jet.	Custom physics model and dynamics randomization	✗	Custom	✗	A3C	kHz-excited APPJ in He	Plasma jet control
Jeong et al. [33]	Modeling Generalized Forces with RL for Sim2Real Transfer	Modeling and learning state dependent generalized forces.	✗	MuJoCo	✗	MPO	Rethink Robotics Sawyer	Nonprehensile manipulation
Arndt et al. [11]	Meta Reinforcement Learning for Sim2Real Domain Adaptation	Domain random. and model-agnostic meta-learning.	✗	MuJoCo	✓Meta-training	PPO	Kuka LBR 4+ arm	Manipulation (hockey puck)
Breyer et al. [34]	Flexible robotic grasping with Sim2Real RL	Direct transfer. Elliptic mask to RGB-D images.	✗	PyBullet	✗	TRPO	ABB YuMi with parallel-jaw gripper	Robotic Grasping
Van Baar et al. [35]	Sim-to-real transfer with robustified policies for robot tasks.	Variation of appearance and/or physics parameters.	✓(sim)	MuJoCo +Ogre 3D	✓	A3C (sim) +Off-policy	Mitsubishi Melfa RV-6SL	Marble Maze Manipulation
Bassani et al. [36]	Sim2Real RL for robotic soccer competitions.	Domain adaptation and custom simulator for transfer.	✗	VSSS-RL	✓	DDPG /DQN	VSSS Robot	Robotic Navigation
Qin et al. [37]	Sim2Real for six-legged robots with DRL and curriculum learning.	Curriculum learning with inverse kinematics.	✗	V-Rep	✓	PPO	Six-legged robot	Navigation and obstacle avoid.
Vacaro et al. [38]	Sim-to-real in reinforcement learning for everyone	Domain randomization (light + color + textures).	✓(sim)	Unity3D	✗	IMPALA	Sainsmart robot arm	Low-cost robot arm
Chaffre et al. [39]	Sim-to-Real Transfer with Incremental Environment Complexity	SAC training using incremental environment complexity.	✗	Gazebo	✗	DDPG /SAC	Wifibot Lab V4	Mapless navigation
Kaspar et al. [40]	RL with Cartesian Commands for Peg in Hole Tasks.	Dynamics (CMA-ES) and environment randomization.	✗	PyBullet	✗	SAC	Kuka LBR iiwa	Peg-in-hole tasks
Hundt et al. [41]	Efficient RL for Multi-Step Visual Tasks via Reward Shaping.	Direct transfer with custom simulation framework.	✗	SPOT Framework	✗	SPOT-Q +PER	Universal Robot UR5	Long-term multi-step tasks
Pedersen et al. [42]	Sim-to-Real Transfer for Gripper Pose Estimation with GAN	CycleGANs for domain adaption and transfer.	✗	Unity	✗	PPO	Panda robot	Robotic Grippers
Ding et al. [43]	Sim-to-Real Transfer for Optical Tactile Sensing	Analysis of different amounts of randomization.	✗	PyBullet	✗	CNN	Sawyer robot +TacTip sensor	Tactile sensing
Muratore et al. [9]	Data-efficient Bayesian Domain Randomization for sim-to-real	Proposed bayesian randomization (BAYR).	✗	Custom/ BoTorch	✗	PPO / RF Classifier	Quanser Qube	swing-up/balancing
Zhao et al. [8]	Towards closing the sim-to-real gap in collaborative DRL with perturbances	Domain randomization (custom perturbations)	✓(sim)	Pybullet	✗	PPO	Kuka (sim-only)	Robot arm reacher
Nachum et al. [44]	Multi-agent manipulation via locomotion	Hierarchical sim-to-real, model-free, zero-shot transfer.	✓	MuJoCo	✗	Custom	D'Kitty robo (2x)	Multi-agent manipulation
Rajeswaran et al. [5]	Dexterous manipulation with DRL and demonstrators.	Imitation learning via demonstrators with VR.	✗	MuJoCo	✗	DAPG	ADROIT 24-DoF Hand	Multi-fingered robot hands

Applications

peg-in-hole



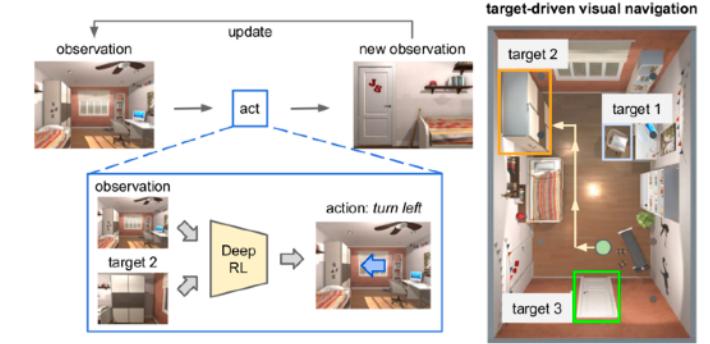
object manipulation / multi-fingered hands



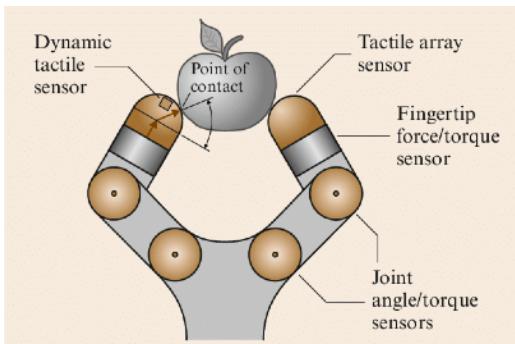
quadruped robot



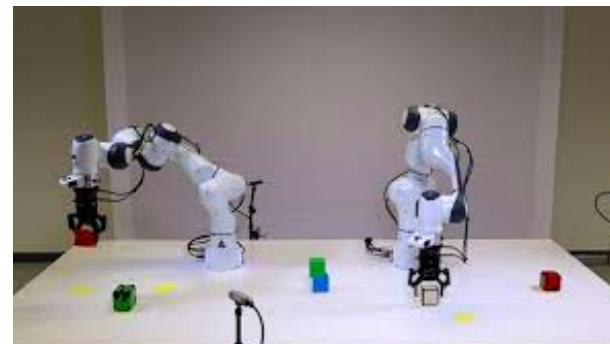
visual navigation



tactile sensing



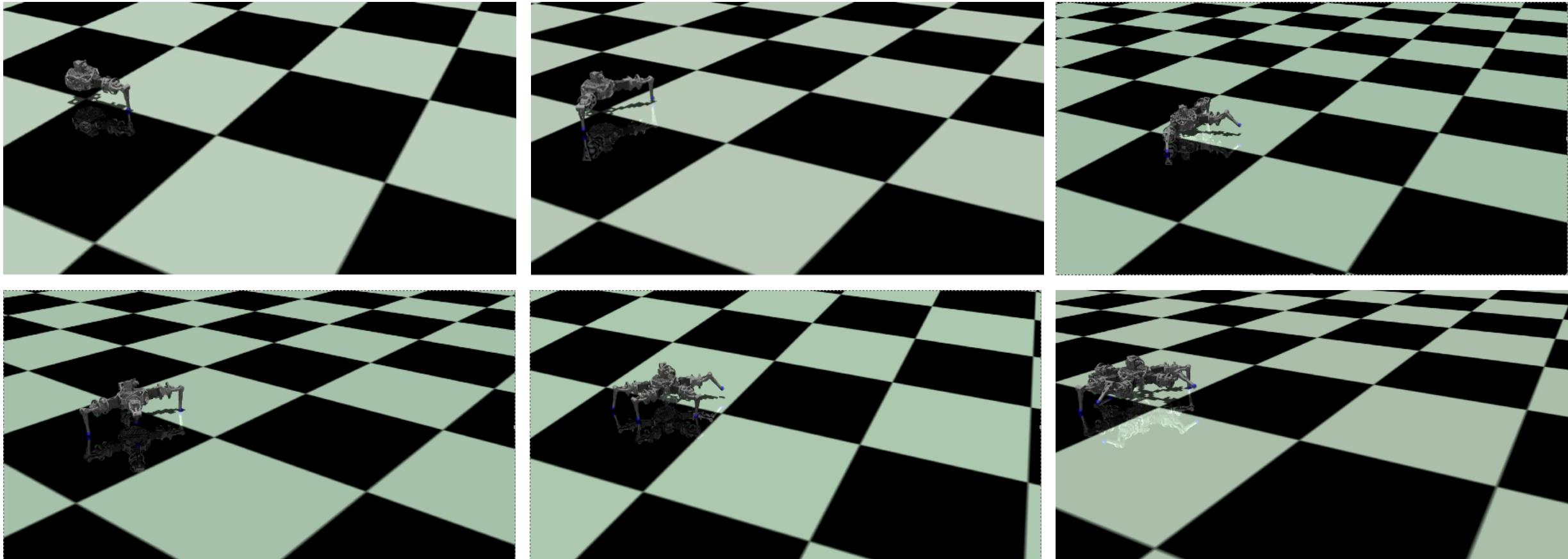
multi-agent manipulation



quad-rotor



Locomotion Tasks



Great Successes in Locomotion



Challenges

- Sample Efficiency
 - When utilizing a real robot, one cannot use excessive episodes.
- Safe Exploration
 - We do not want to destroy the robot while training.
- Reward Design
 - In often cases, it is difficult to explicitly design the proper reward function (e.g., what is the appropriate reward for a service robot?).

Smart Building Energy Management

"Deep Reinforcement Learning for Smart Building Energy Management: A Survey," 2020
<https://arxiv.org/pdf/2008.05074>

Smart Building



① Lighting

Lights control to provide the right luminosity where and when it is needed

② Control Panel

Access control panel via interactive touch screen devices or from mobile

③ Occupancy Detection

Occupancy and motion sensors for a comfortable space

④ Heating, ventilation and air-conditioning

Optimum climate, temperature and air control

⑤ Power Supply

Stable bus voltage and safe access to power network data

⑥ Management Station

Improved maintenance management and energy performance

⑦ Energy Efficiency

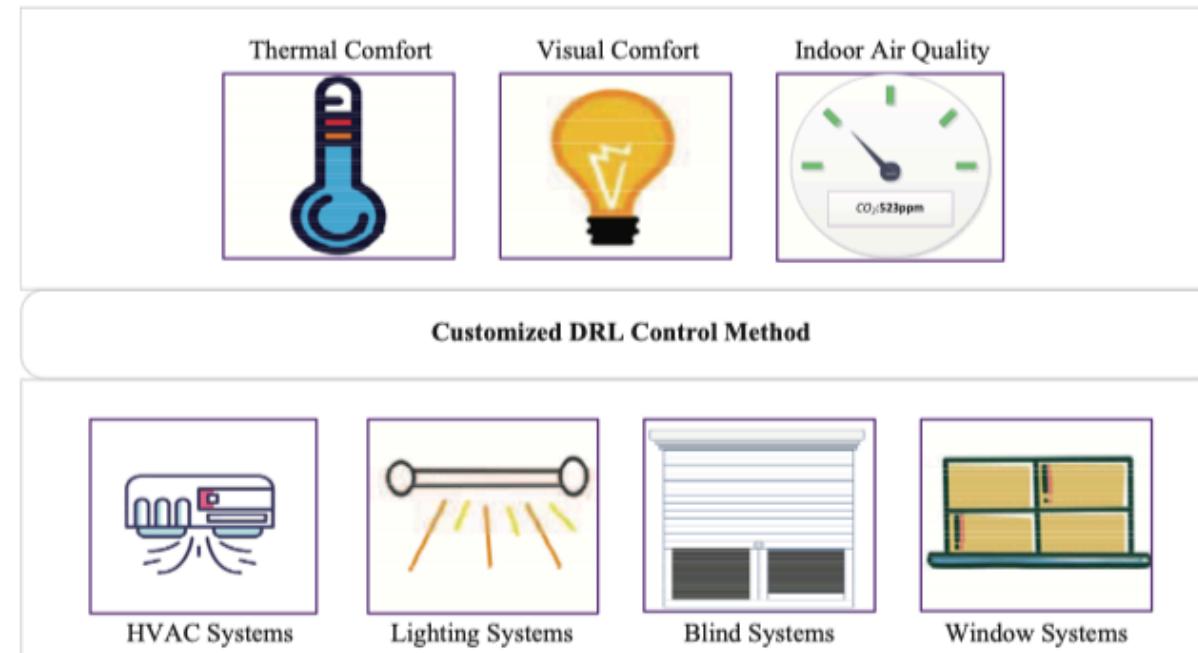
Increase energy savings and reduce building operating costs

Why Smart Building?

- Global buildings consumed 30% of total energy and generated 28% of total carbon emissions in 2018, which leads to economic and environmental concerns.
- Therefore, it is significant to reduce energy consumption, energy cost, and carbon emission of buildings while maintaining user comfort.
- Challenges:
 - It is very challenging to develop a building **thermal dynamics model** that is both accurate and efficient enough for building control.
 - There are many kinds of **uncertainties**.
 - Building energy optimization problems may have **extremely large solution spaces**, which can not be solved in real-time by traditional methods.

Energy Management

- Typical building **performance metrics** consist of energy cost, energy consumption, thermal comfort, indoor air quality, non-uniformity of radiant temperature, peak demand, consumers' satisfaction degree, lighting comfort, productivity, operating cost, the overall levelized energy cost, and carbon emission.

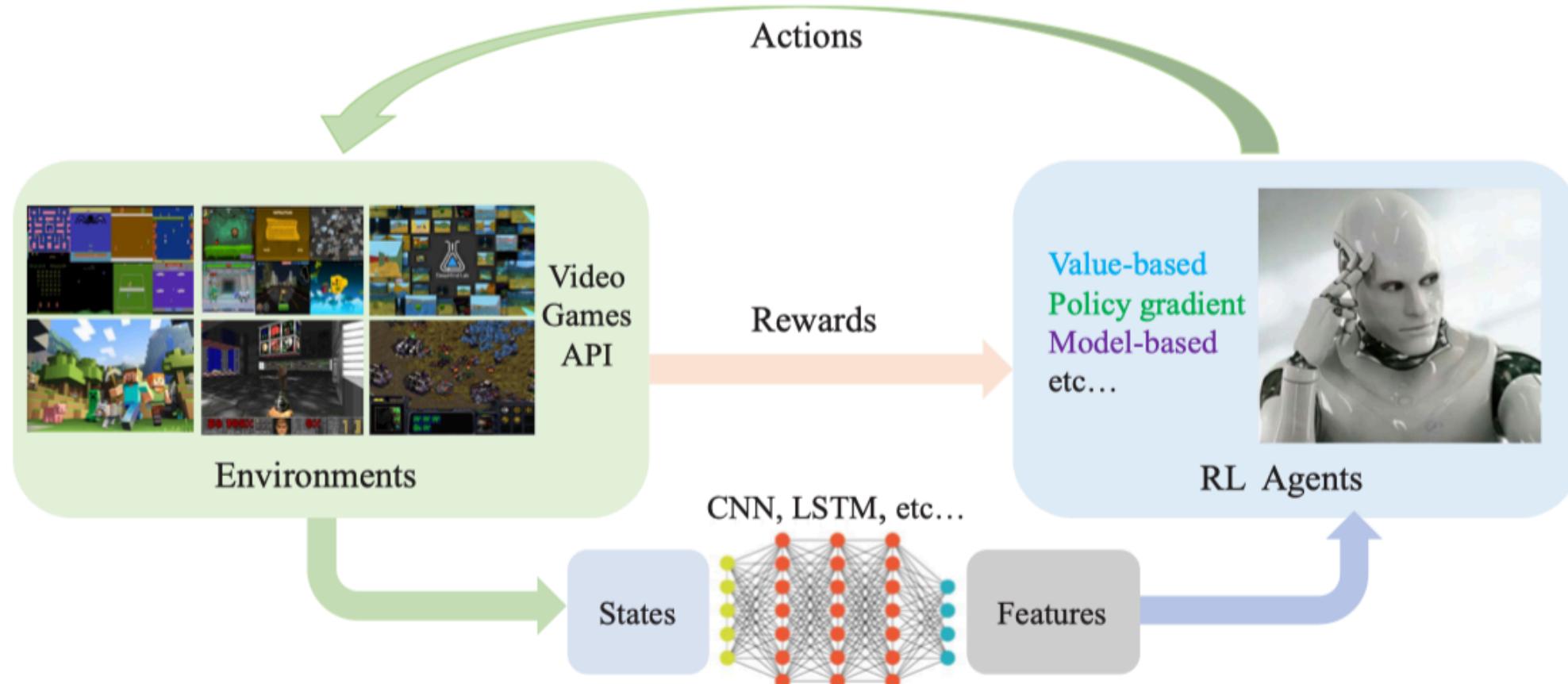


HVAC: Heating, Ventilation and Air Conditioning

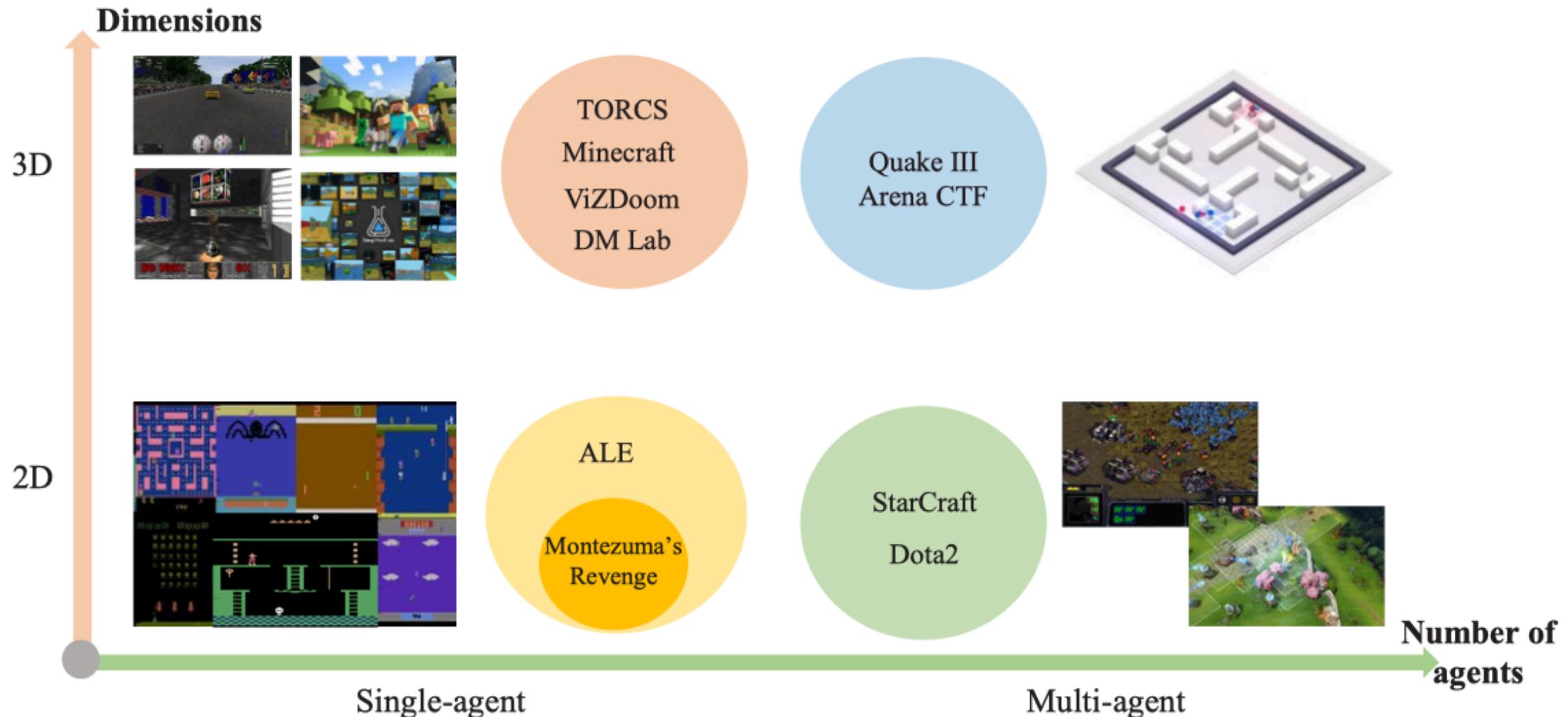
Games

"A Survey of Deep Reinforcement Learning in Video Games," 2019
<https://arxiv.org/pdf/1912.10944.pdf>

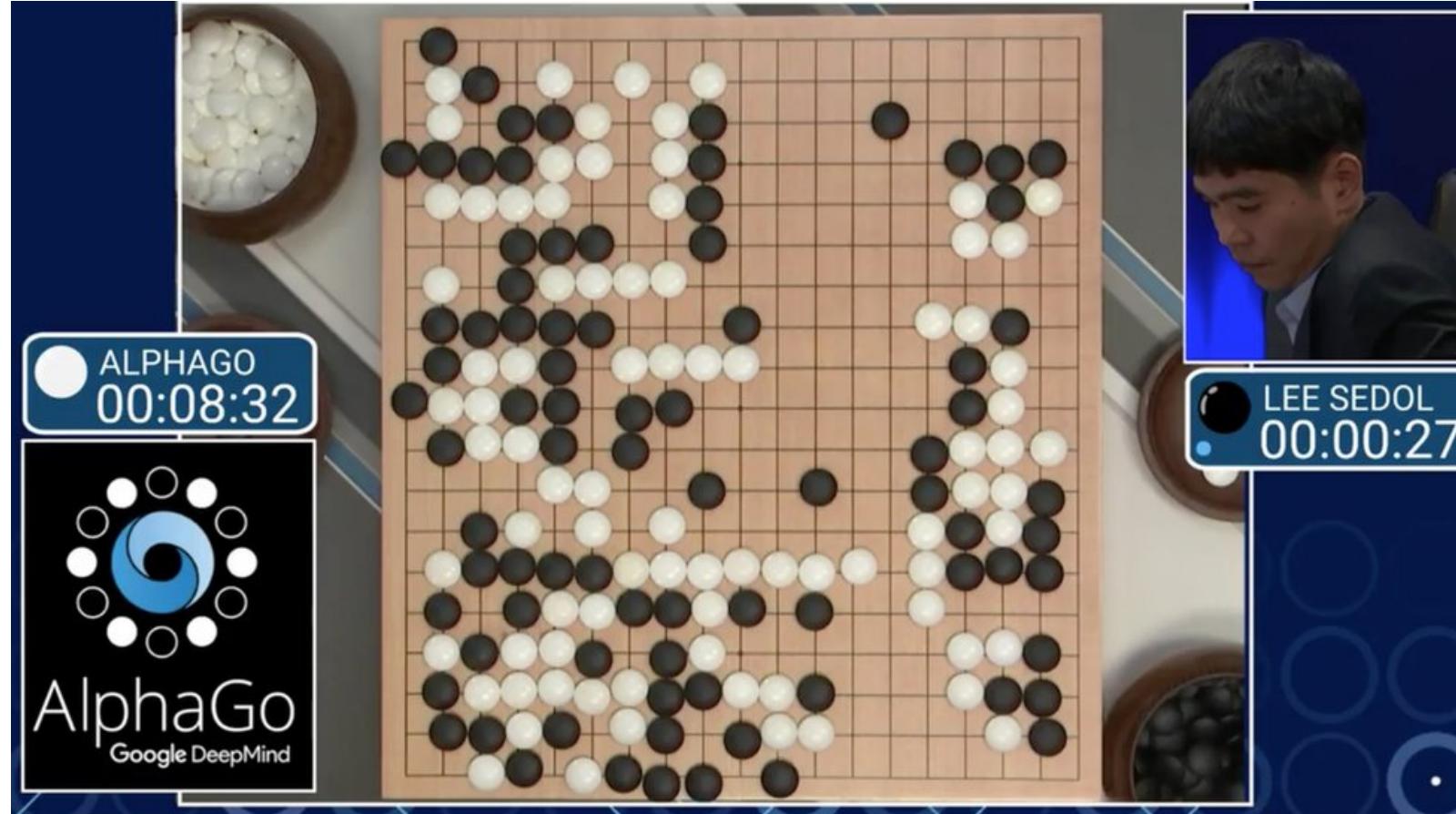
Why Games are suitable for RL?



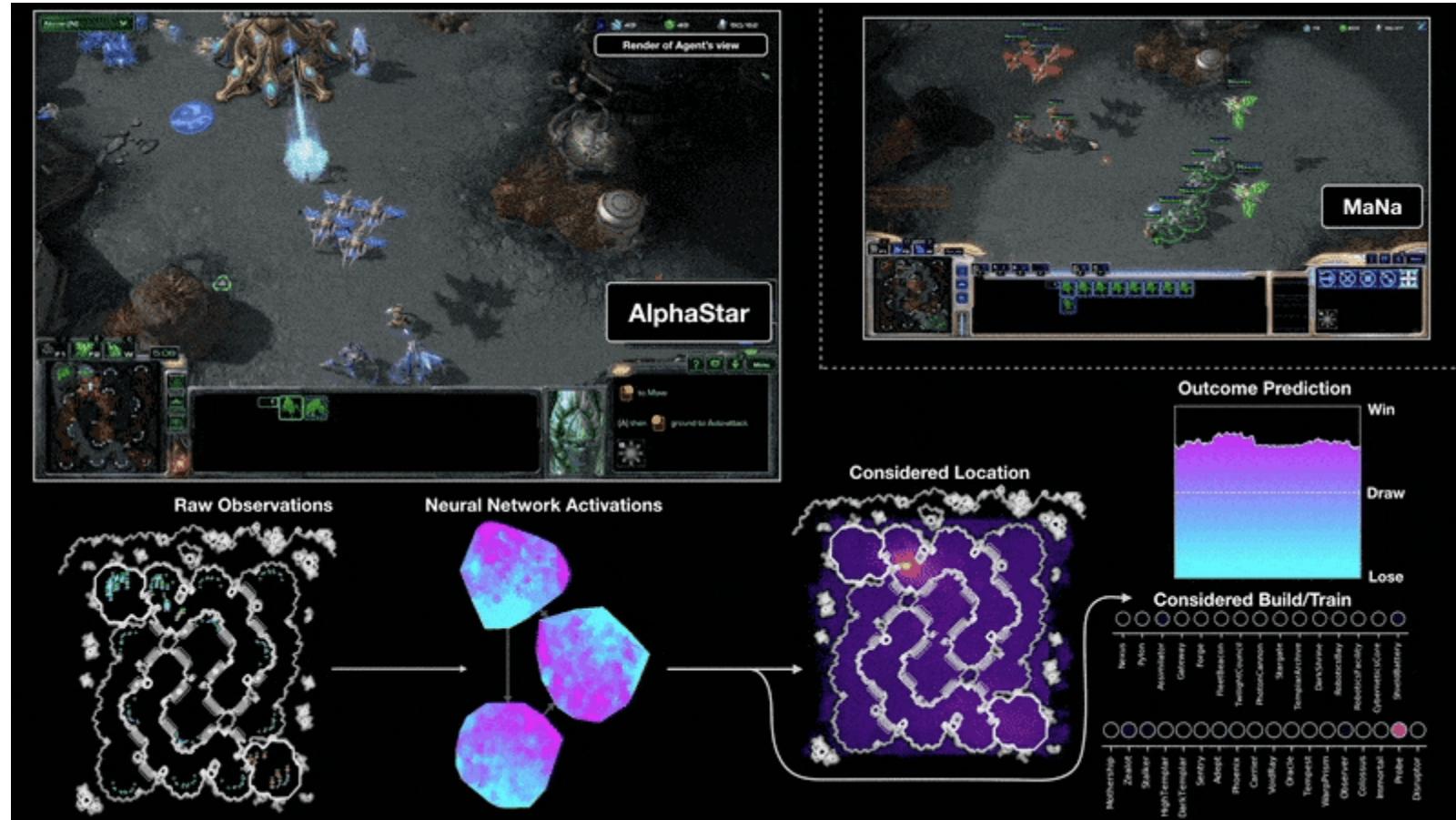
Game Environments



AlphaGo



AlphaStar



OpenAI Five

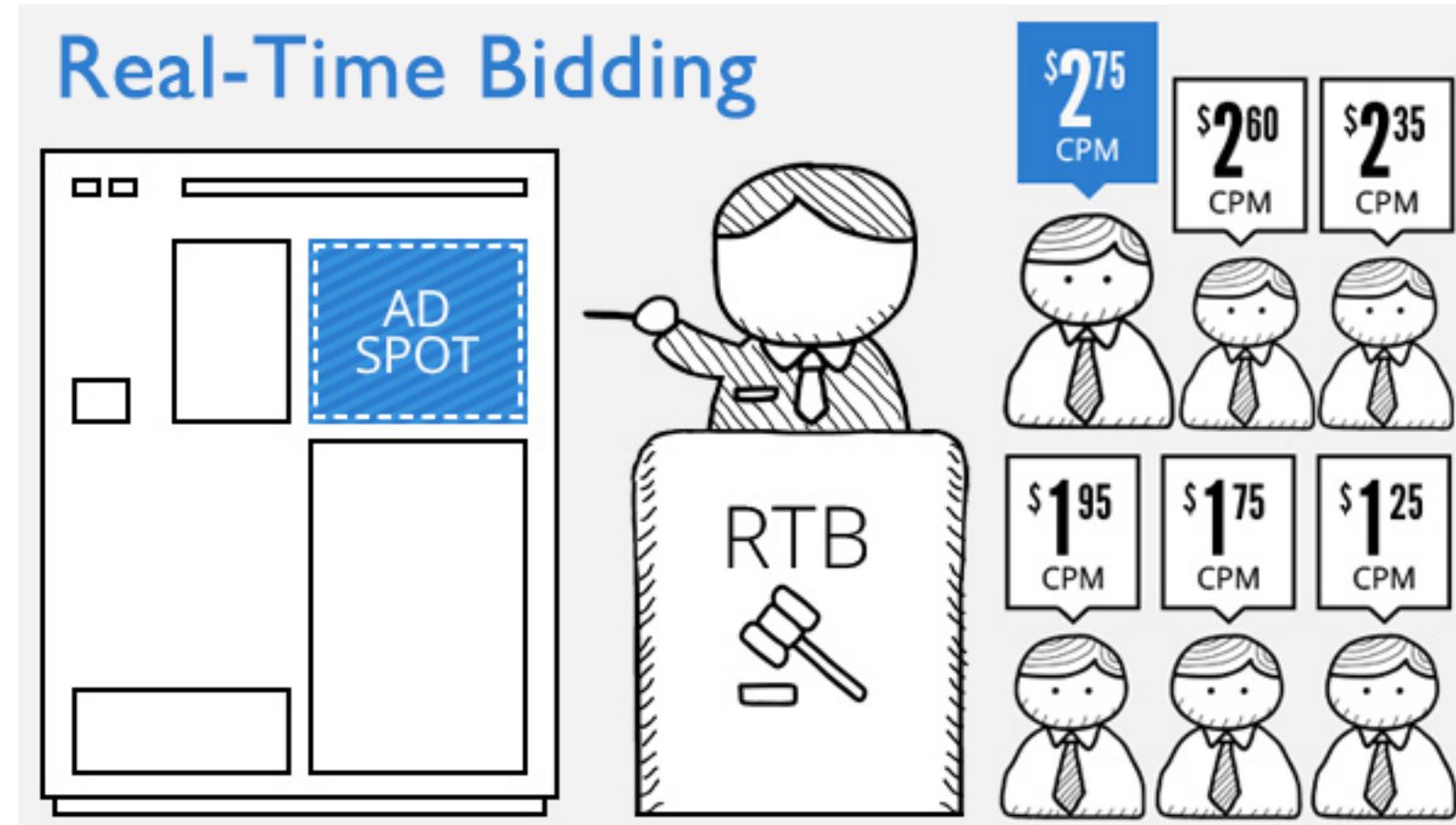


Marketing Advertising

"Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising."
2018

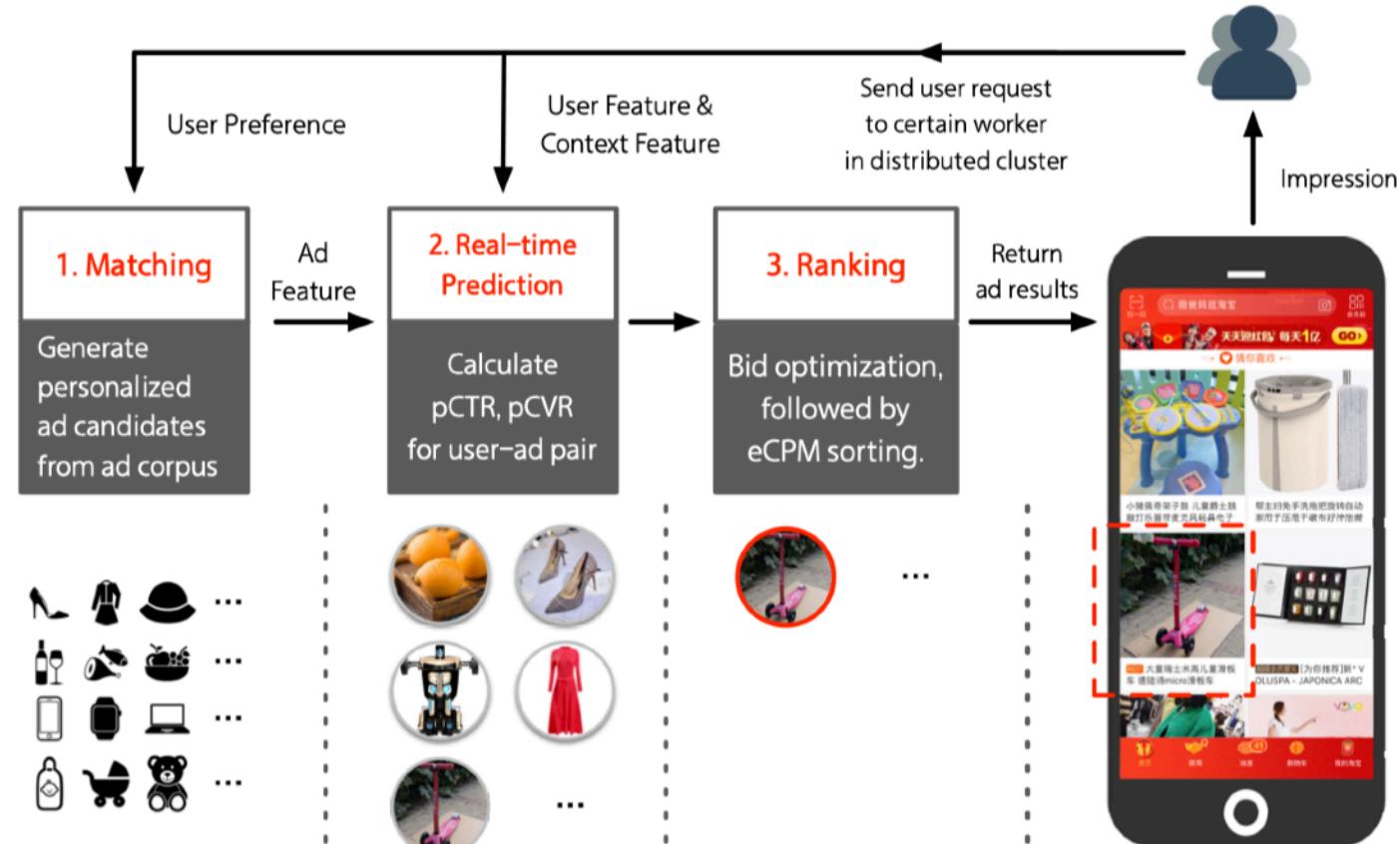
<https://arxiv.org/pdf/1802.09756.pdf>

Real-Time Bidding (RTB)



Taobao Display AD System

- This study is developed in the context of a realistic industry setting, Taobao (taobao.com), the largest e-commerce platform (Alibaba) in China.



Real-Time Bidding (RTB)

- RTB is a **multi player game** with a competition where optimizing one party's benefit may ignore and hurt other parties' benefits.
- From the ad system's viewpoint, the micro-level optimization may not fully utilize the dynamics of the ad ecosystem in order to achieve better social optimality.
- This paper addresses this issue by utilizing multi-agent reinforcement learning (MARL) named **multi-agent advertising bidding**.

RTB as a Markov Game

- **State:** the possible status of all (or **clustered**) bidding agents
 - Budget spent status (to plan for the rest auctions)
 - (cost, revenue) distribution of consumers (to distinguish quality)
 - (cost, revenue) distribution of other agents (merchants) (to evaluate the competitiveness or cooperativeness)
- **Action:** bid adjustment ratio
- **Reward:** total revenue of all bidding agents

Back to our Key Question

What makes RL different from other ML methods (e.g., supervised or unsupervised learning)?

Summary

- Reinforcement learning considers sequential decision problems.
- One must properly design (state, action, reward) for the problem of interest.
- The objective of reinforcement learning is not to maximize the instantaneous reward but to maximize the value (sum of future rewards).
- In practice, designing reward functions is very difficult.



ROBOT INTELLIGENCE LAB