ROBOT INTELLIGENCE LAB

Introduction to Reinforcement Learning

Lecture 1. Reinforcement Learning Applications

Sungjoon Choi, Korea University

kakaobrain

Contents

- Healthcare
- Autonomous Vehicles
- Ridesharing
- Natural Language Processing
- Robotics
- Audio-Based Applications
- Data Caching in Edge Network
- Smart Building Energy Management
- News Recommendation
- Video Games
- Cyber Security
- Marketing Advertising

ROBOT INTELLIGENCE LAB

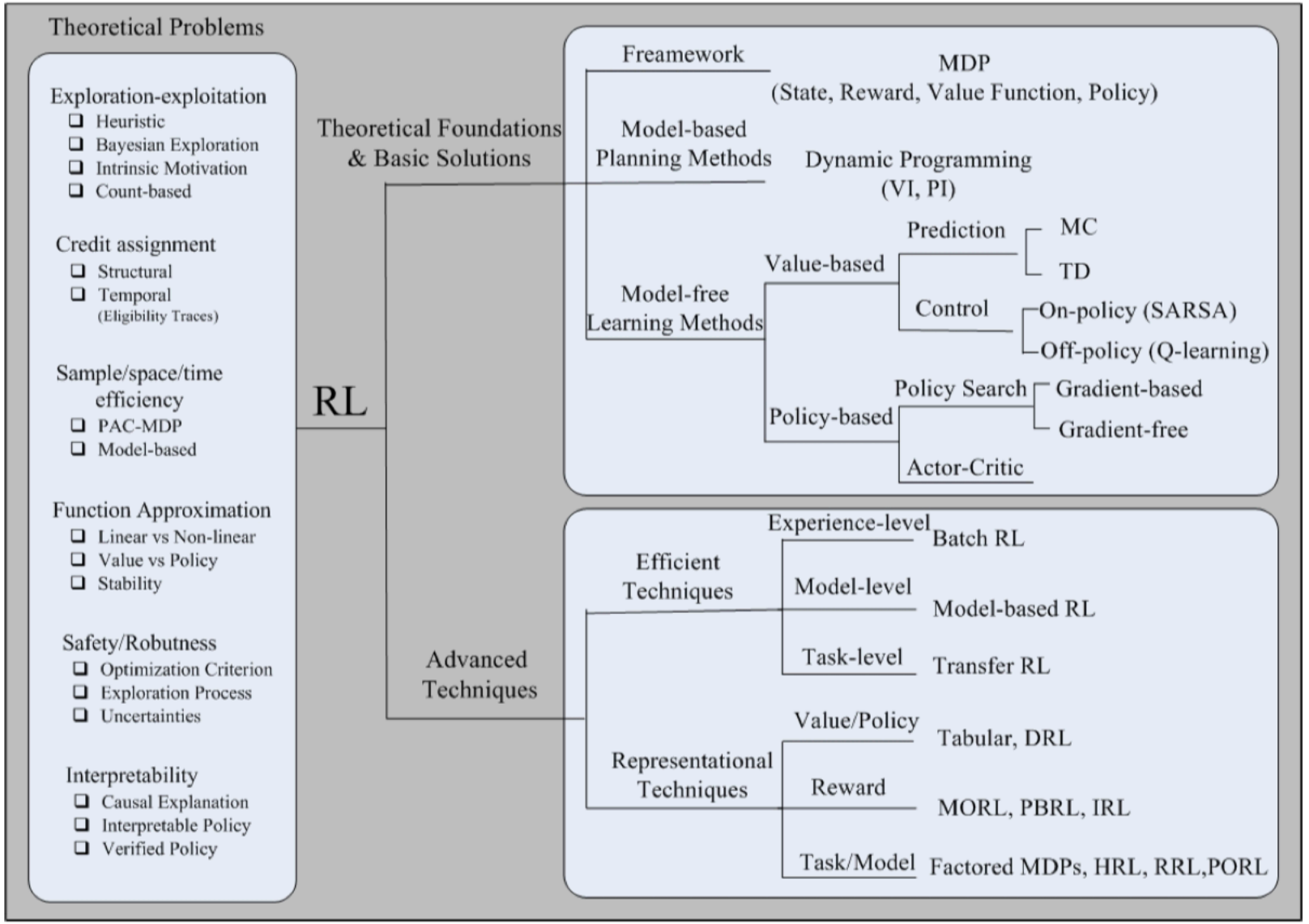
Healtcare

"Reinforcement Learning in Healthcare: A Survey," 2020

https://arxiv.org/pdf/1908.08796.pdf

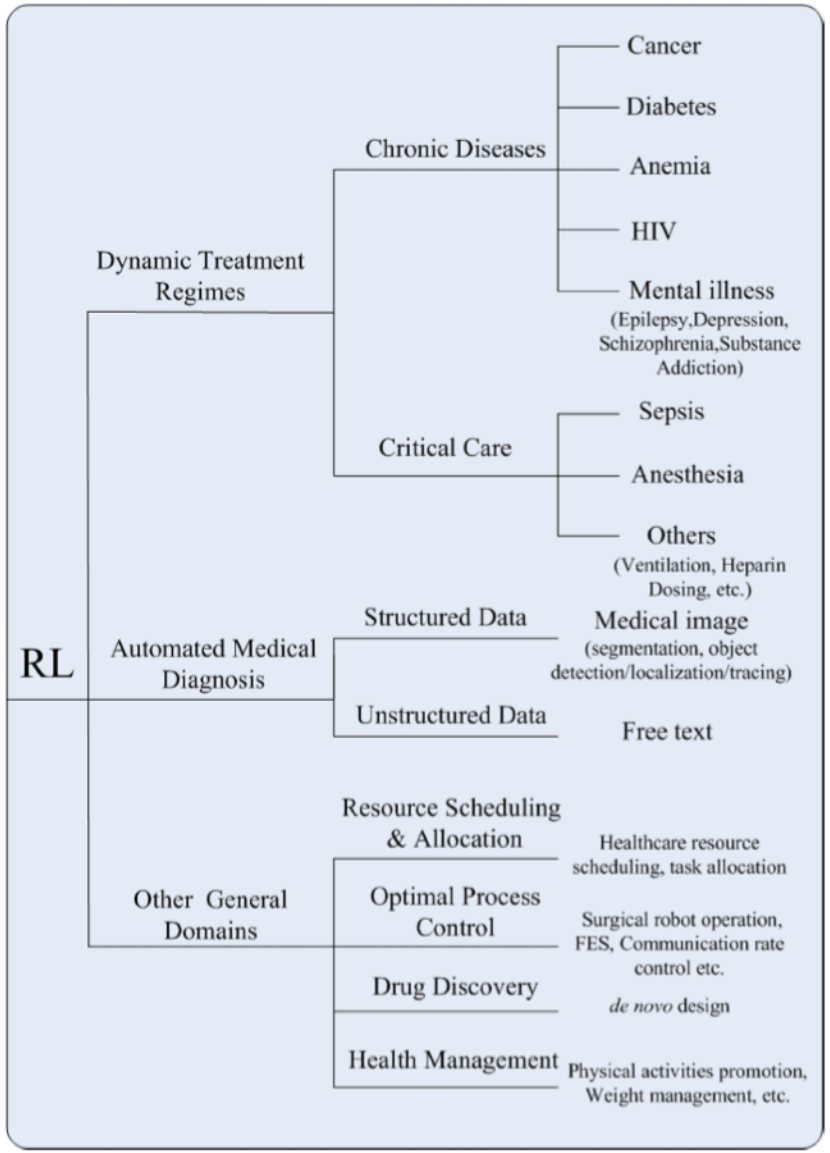
Topics in Reinforcement Learning

ROBOT INTELLIGENCE LAB

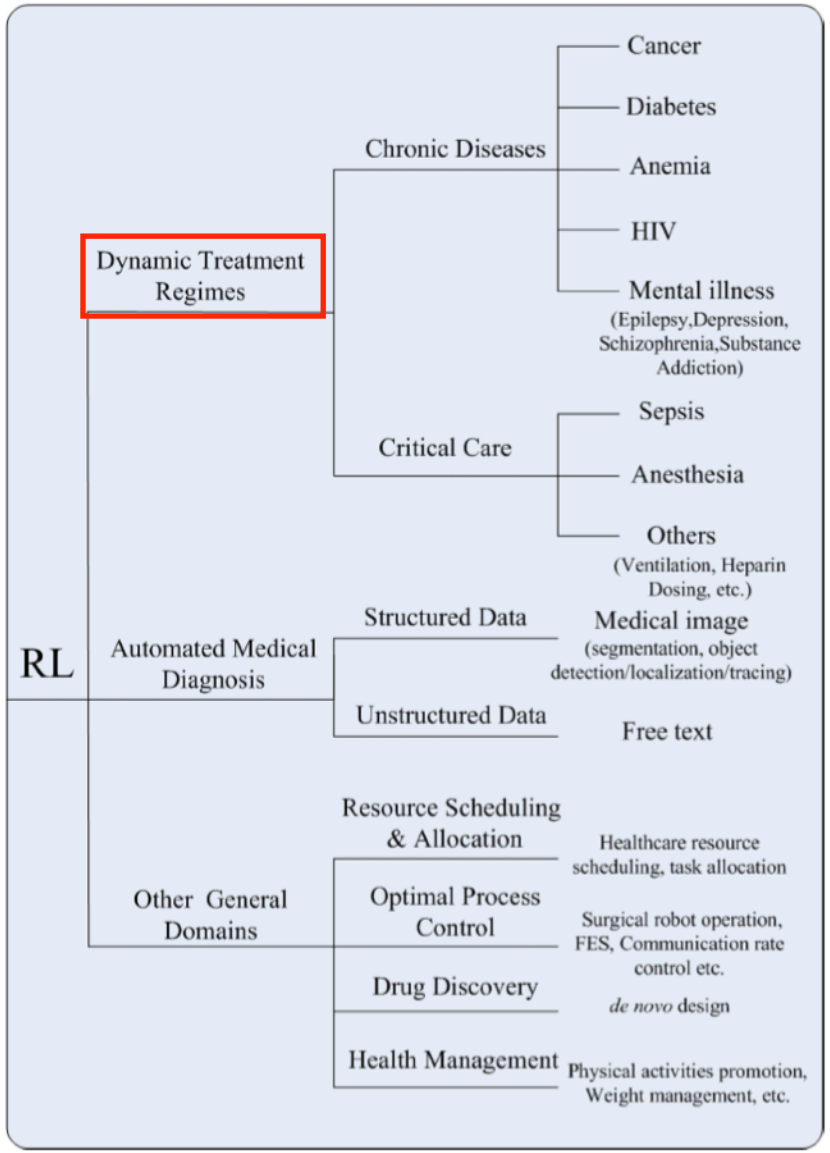


RL in Healthcare

ROBOT INTELLIGENCE LAB

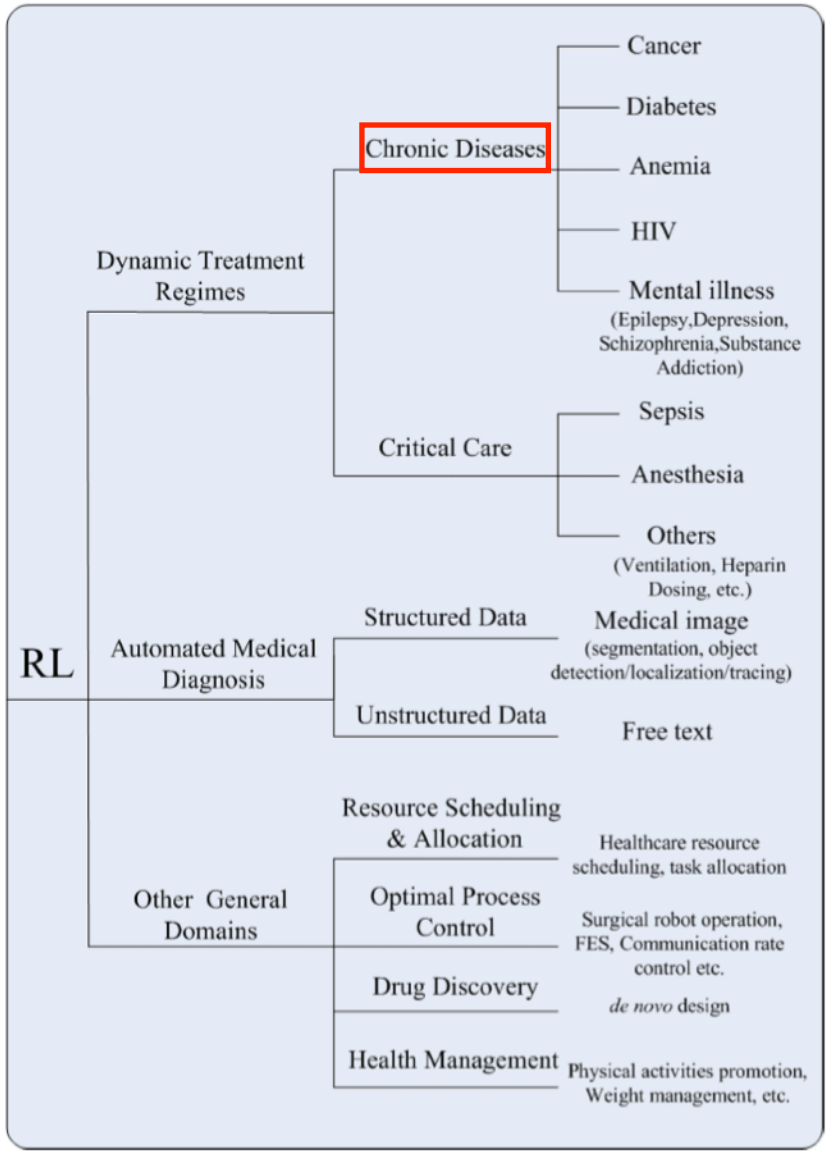


Dynamic Treatment Regimes



- One goal of healthcare decision-making is to develop effective treatment regimes that can **dynamically adapt** to the varying clinical states and improve the long-term benefits of patients.
- Dynamic treatment regimes (DTRs) provide a new paradigm to automate the process of developing new effective treatment regimes for individual patients with long-term care.
- A DTR is 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.

Dynamic Treatment Regimes



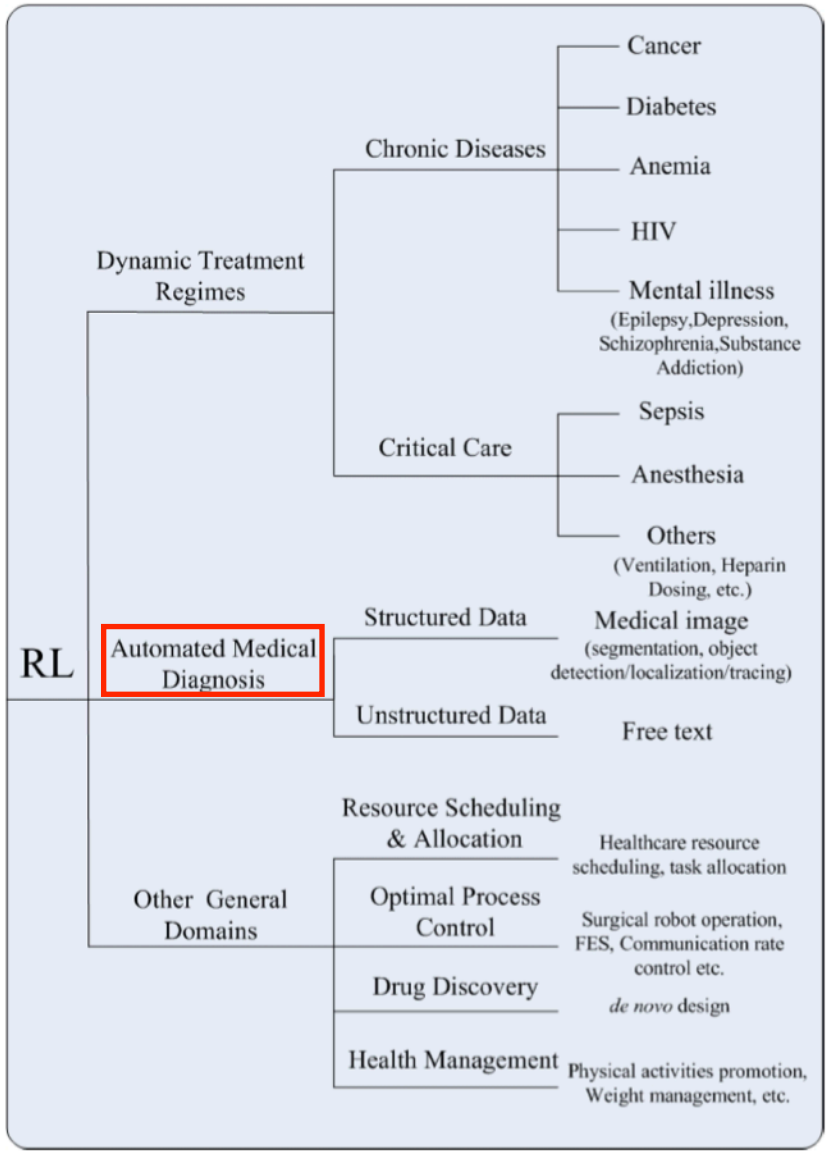
- **Chronic diseases** are becoming the most pressing public health issue worldwide. They include diabetes, cancer, HIV infection, depression, and so forth.
- Long-term treatment is made up of a **sequence of medical interventions** that must take into account the changing health status of a patient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

DTR in Cancer

ROBOT INTELLIGENCE LAB

Applications	References	Base Methods	Efficient Techniques	Representational Techniques	Data Acquisition	Highlights or Limits
Optimal chemotherapy drug dosage for cancer treatment	Zhao *et al.* [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 *et al.* [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 *et al.* [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 *et al.* [52], Cheng et al. [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.
	Akrour *et al.* [90], Busa-Fekete et al. [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.
Optimal fractionation scheduling of radiation therapy for cancer treatment	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 *et al.* [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 *et al.* [94]	Q-learning	N/A	N/A	Agent-based model	Using agent-based simulation to model the dynamics of tumor growth.
	Jalalimanesh *et al.* [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.
Hypothetical or generic cancer clinical trial	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



- **Medical diagnosis** is a mapping process from a patient's information such as 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 rely on a large number of annotated samples in order to infer and predict the 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 (e.g., feaure extracion, segmentation, etc)

Challenges

ROBOT INTELLIGENCE LAB

- State/Action Engineering
  - The first step in applying RL to a healthcare problem is determining **how to collect** and **pre-process** proper medical data, and **summarize** such data into some manageable state representations in a way that **sufficient information can be retained** for the task at hand.
- Reward Fomulation
  - Among all the basic components, the **reward** may be at the **core of an RL process**. Since it encodes the goal information of a learning task, a proper formulation of reward functions plays the most crucial role in the success of RL.
- Policy Evaluation
  - **Off-policy evaluation** is critical in healthcare domains because it is usually infeasible to estimate policy value by running the policy directly on the patients.
- Exploration Strategies
  - The consequences of wrong actions can result in unrecoverable effects (e.g., dealing with patients in healthcare domains as we cannot bring back to life when a patience has been given a fatal medical treatment).
- Credit Assignment
  - The credit assignment problem decides when an action or which actions is **responsible** for the learning outcome after a sequence of decisions.

Future Perspectives

ROBOT INTELLIGENCE LAB

- **Interpretable** Strategy Learning
  - The inability to provide explanation greatly hinders the successful adaptation of RL policies for safety-critical applications.
- Integration of **Prior** Knowledge
  - There is a wealth of prior knowledge in healthcare domains that can be used for learning performance improvement.
- Learning from **Small** Data
  - Broadly, there are two different ways of dealing with a small sample learning problem, data augmentation or domain adaptation.

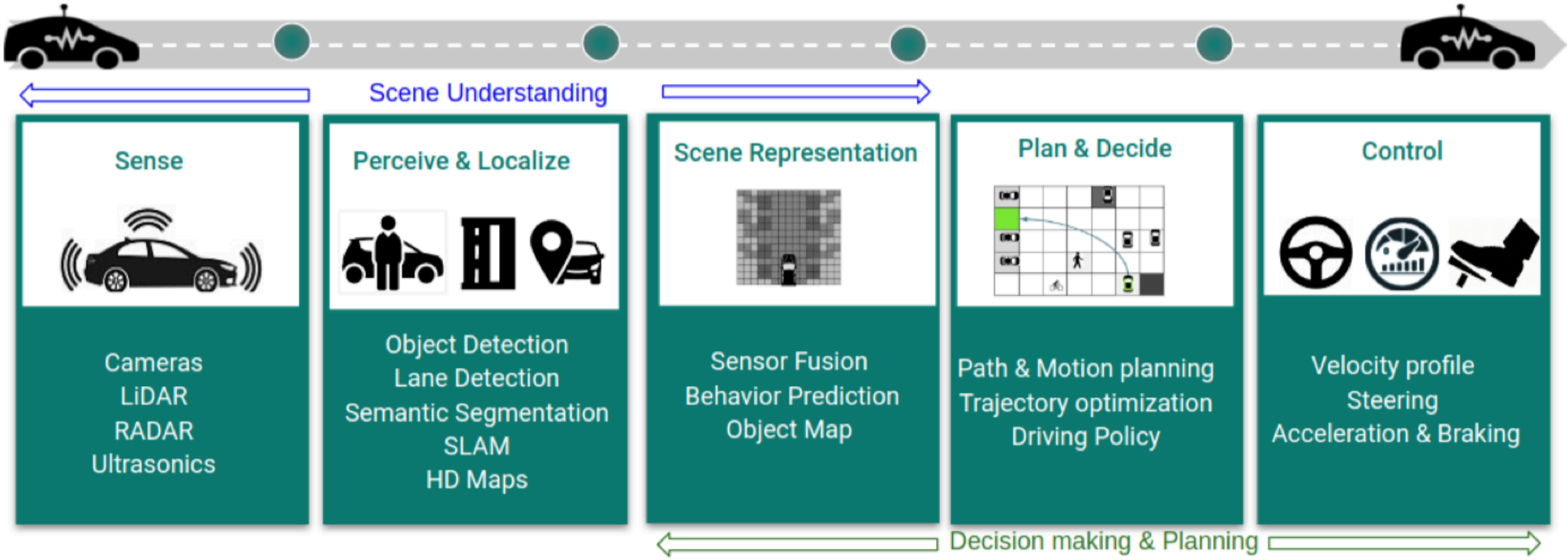
ROBOT INTELLIGENCE LAB

Autonomous Vehicles

Deep Reinforcement Learning for Autonomous Driving: A Survey, 2021

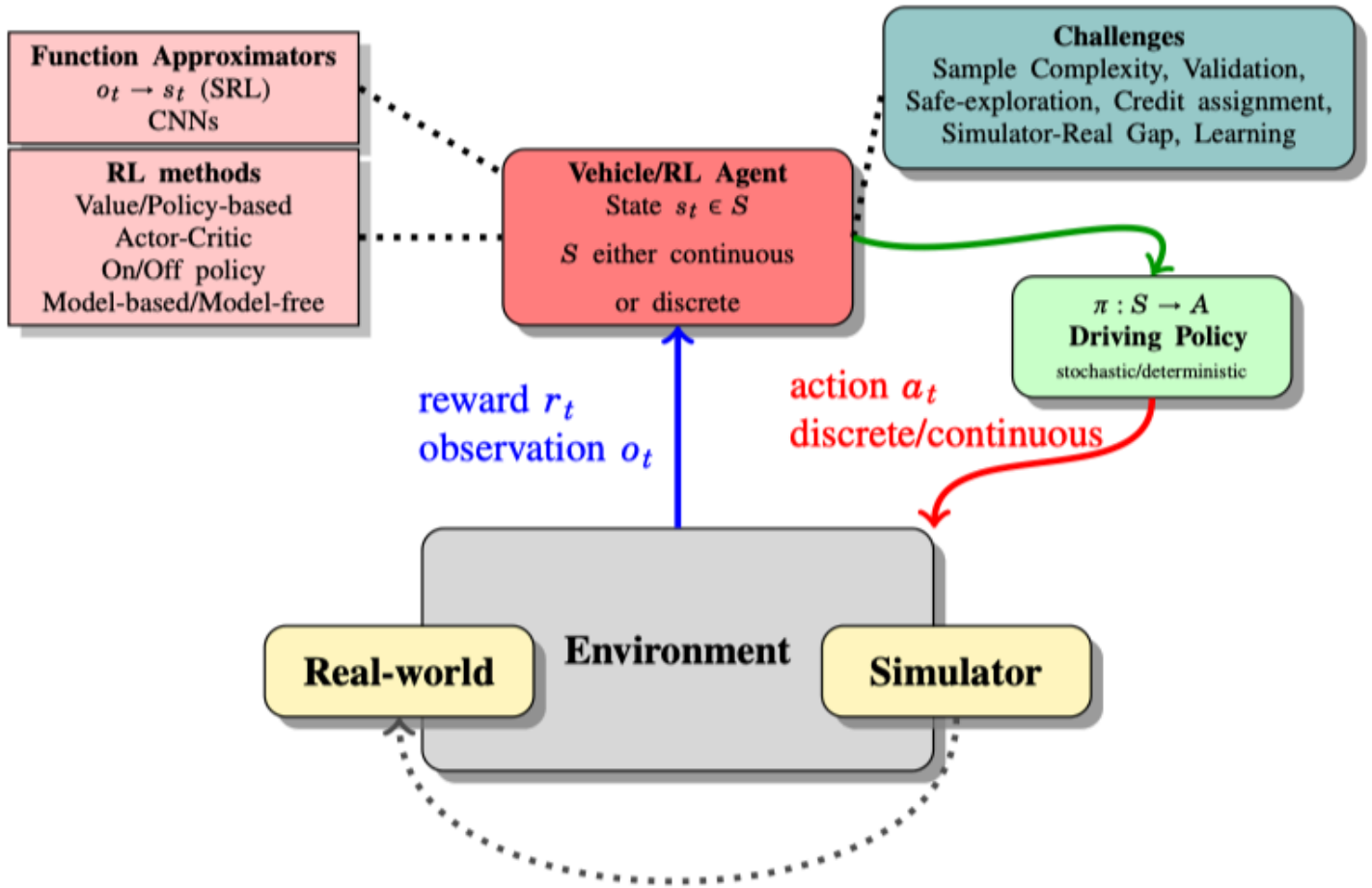
https://arxiv.org/pdf/2002.00444.pdf

Autonomous Driving System



Reinforcement Learning

ROBOT INTELLIGENCE LAB



RL for Autonomous Driving

ROBOT INTELLIGENCE LAB

- 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 of policies that ensures safety and perform risk estimation.

RL for Autonomous Driving

ROBOT INTELLIGENCE LAB

- 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

ROBOT INTELLIGENCE LAB

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 evalute 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 netowrks over image-based input obstacle map	Smooth control behavior of vehicle and better peformance compared to multi-step DQN

Simulators

ROBOT INTELLIGENCE LAB

Simulator	Description
CARLA [78]	Urban simulator, Camera & LIDAR streams, with depth & semantic segmentation, Location information
TORCS [96]	Racing Simulator, Camera stream, agent positions, testing control policies for vehicles
AIRSIM [97]	Camera stream with depth and semantic segmentation, support for drones
GAZEBO (ROS) [98]	Multi-robot physics simulator employed for path planning & vehicle control in complex 2D & 3D maps
SUMO [99]	Macro-scale modelling of traffic in cities motion planning simulators are used
DeepDrive [100]	Driving simulator based on unreal, providing multi-camera (eight) stream with depth
Constellation [101]	NVIDIA DRIVE Constellation^{TM} simulates camera, LIDAR & radar for AD (Proprietary)
MADRaS [102]	Multi-Agent Autonomous Driving Simulator built on top of TORCS
Flow [103]	Multi-Agent Traffic Control Simulator built on top of SUMO
Highway-env [104]	A gym-based environment that provides a simulator for highway based road topologies
Carcraft	Waymo's simulation environment (Proprietary)

TABLE II

SIMULATORS FOR RL APPLICATIONS IN ADVANCED DRIVING ASSISTANCE SYSTEMS (ADAS) AND AUTONOMOUS DRIVING.

Challenges

ROBOT INTELLIGENCE LAB

- 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

ROBOT INTELLIGENCE LAB

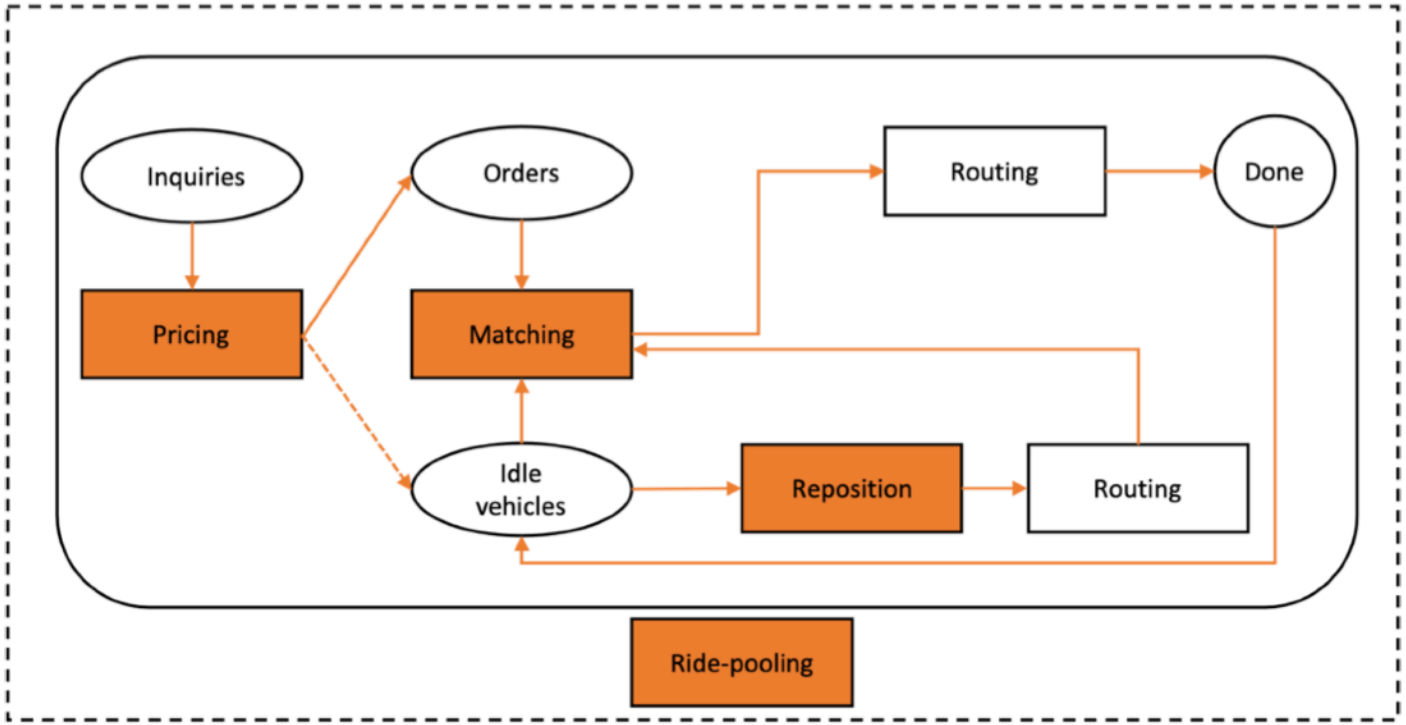
Ridesharing

"Reinforcement Learning for Ridesharing: A Survey," 2021

https://arxiv.org/pdf/2105.01099.pdf

Ridesharing

ROBOT INTELLIGENCE LAB



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

Ridesharing

ROBOT INTELLIGENCE LAB

- Three major characteristic modules: **pricing**, **matching**, and **repositioning**.
  - When a potential passenger submits a trip request, the **pricing module** offers a quote, which the passenger either accepts or rejects.
  - Upon acceptance, the **matching module** attempts to assign the request to an available driver.
  - Depending on driver pool availability, the request may have to wait in the system until a successful match.
  - The **repositioning module** guides idle vehicles to specific locations in anticipation of fulfilling more requests in the future
- Why reinforcement learning?
  - The operational decisions in ridesharing are sequential in nature and have strong spatiotemporal dependency, offering excellent applications of RL.

RL for Ridesharing

ROBOT INTELLIGENCE LAB

- Pricing
  - The ridesharing marketplace is complex due to its two-sided (price affects drivers, and drivers affect pricing) nature and spatiotemporal dimensions.
- Online matching
  - The RL literature for ridesharing 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

ROBOT INTELLIGENCE LAB

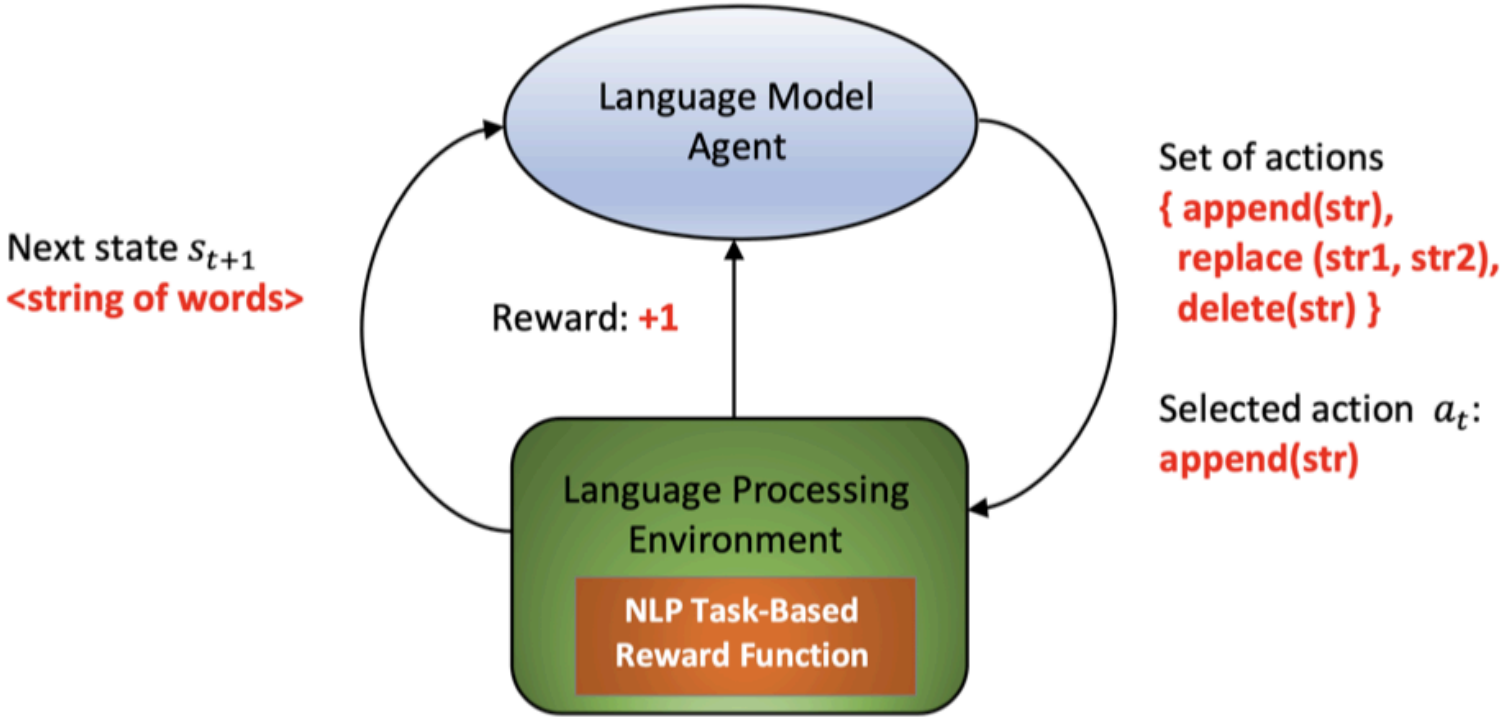
- Ride-pooling
  - Most of the existing RL methods assume that the action set is predetermined, hence, some make only high-level decisions (e.g., repositioning). Then, additional optimization is required to match the given high-level decision.
- Joint Optimizatin
  - The rideshare platform is an integrated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

ROBOT INTELLIGENCE LAB

Natural Language Processing

"Survey on reinforcement learning for language processing," 2021
https://arxiv.org/pdf/2104.05565.pdf

NLP and RL

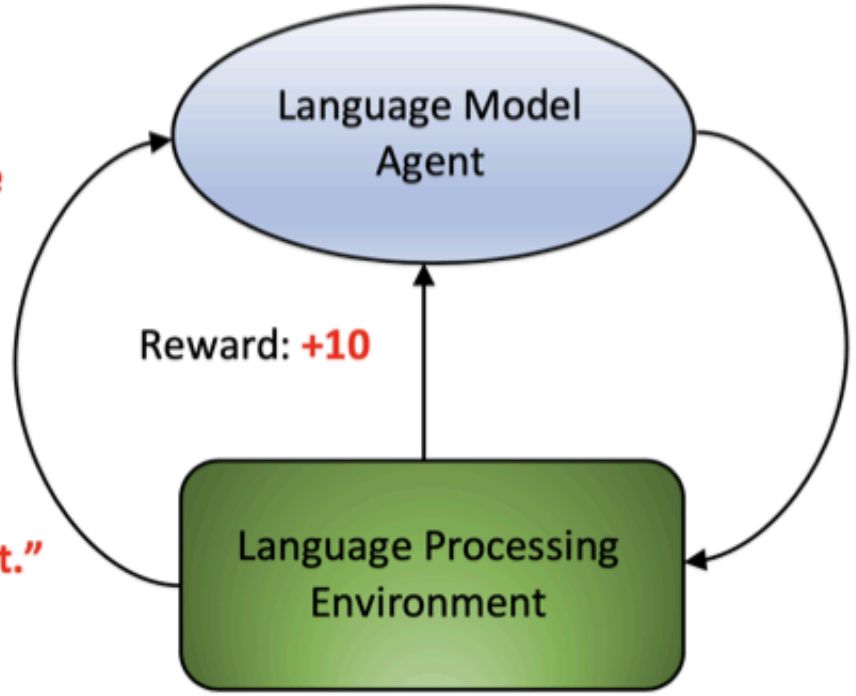


- In natural language processing, one of the main goals is the development of computer programs capable of communicating with humans through the use of natural language.

Natural Language Understanding

ROBOT INTELLIGENCE LAB

Current state (user's utterance):
"User 02: Does this product have a discount?"

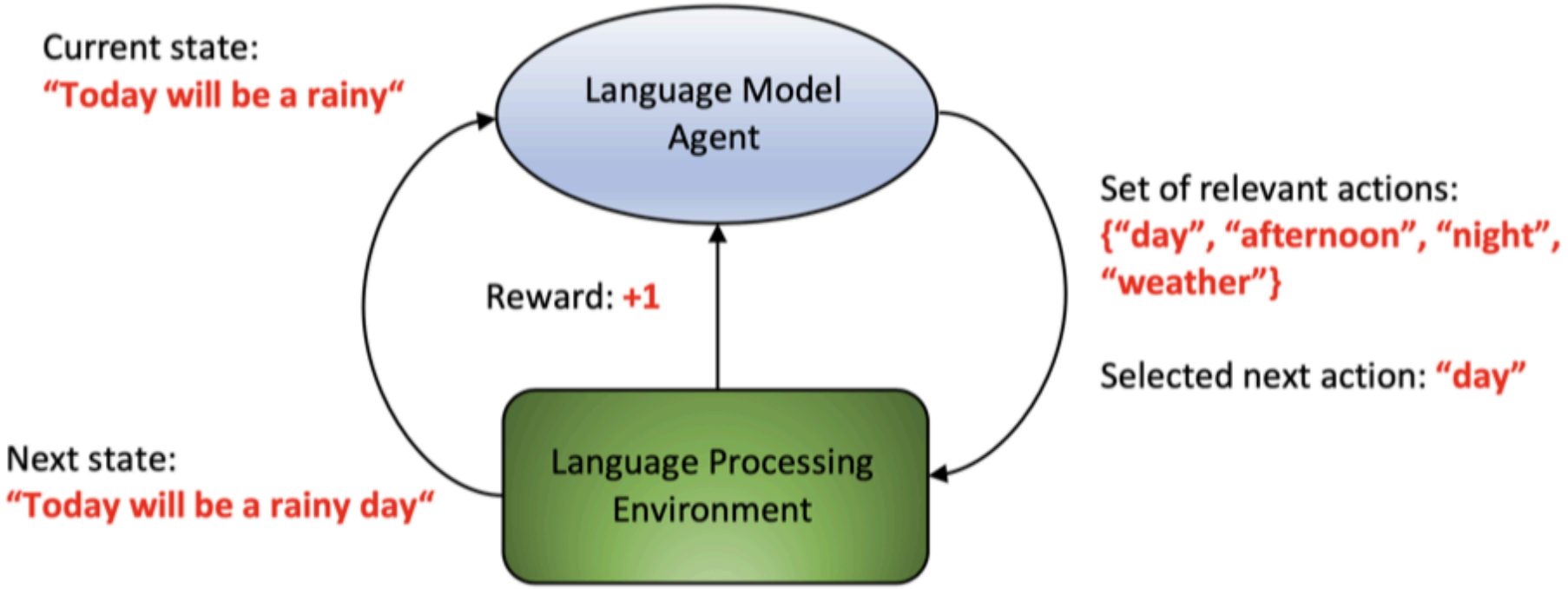


Selected action:
[who="user02", intention="wants", content="discount"]

Next state (Infobot's utterances):
"Infobot: yes, it has a 20% discount."

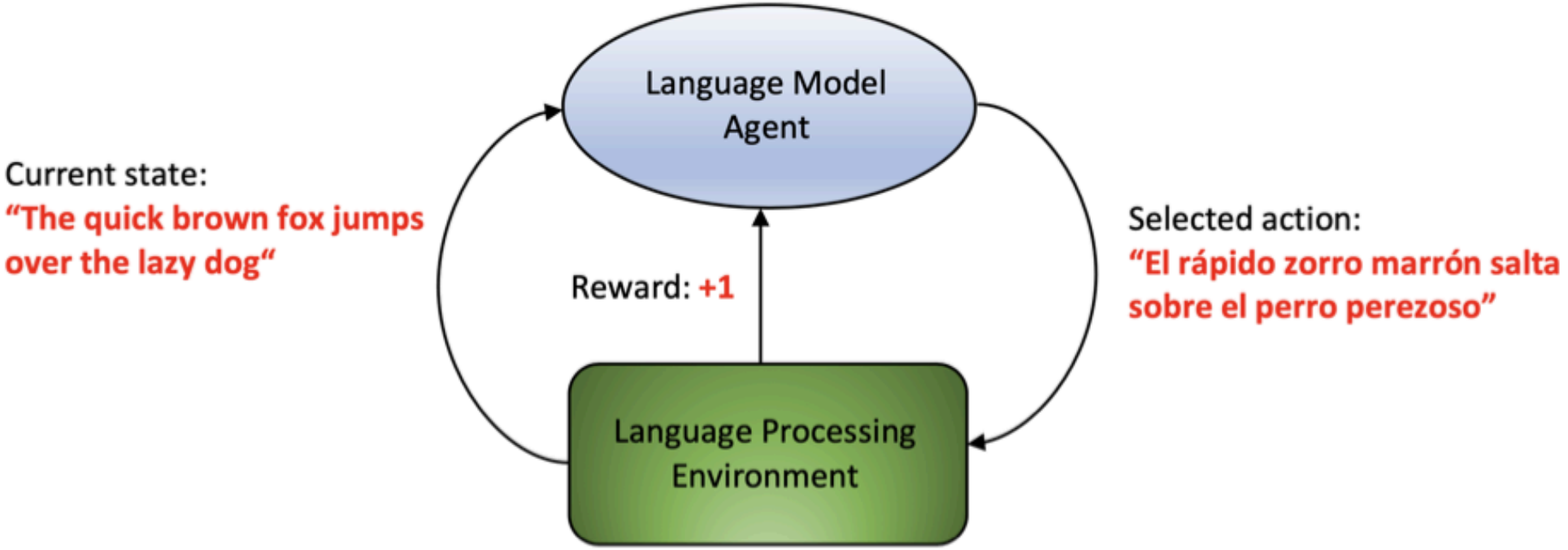
Natural Language Generation

ROBOT INTELLIGENCE LAB

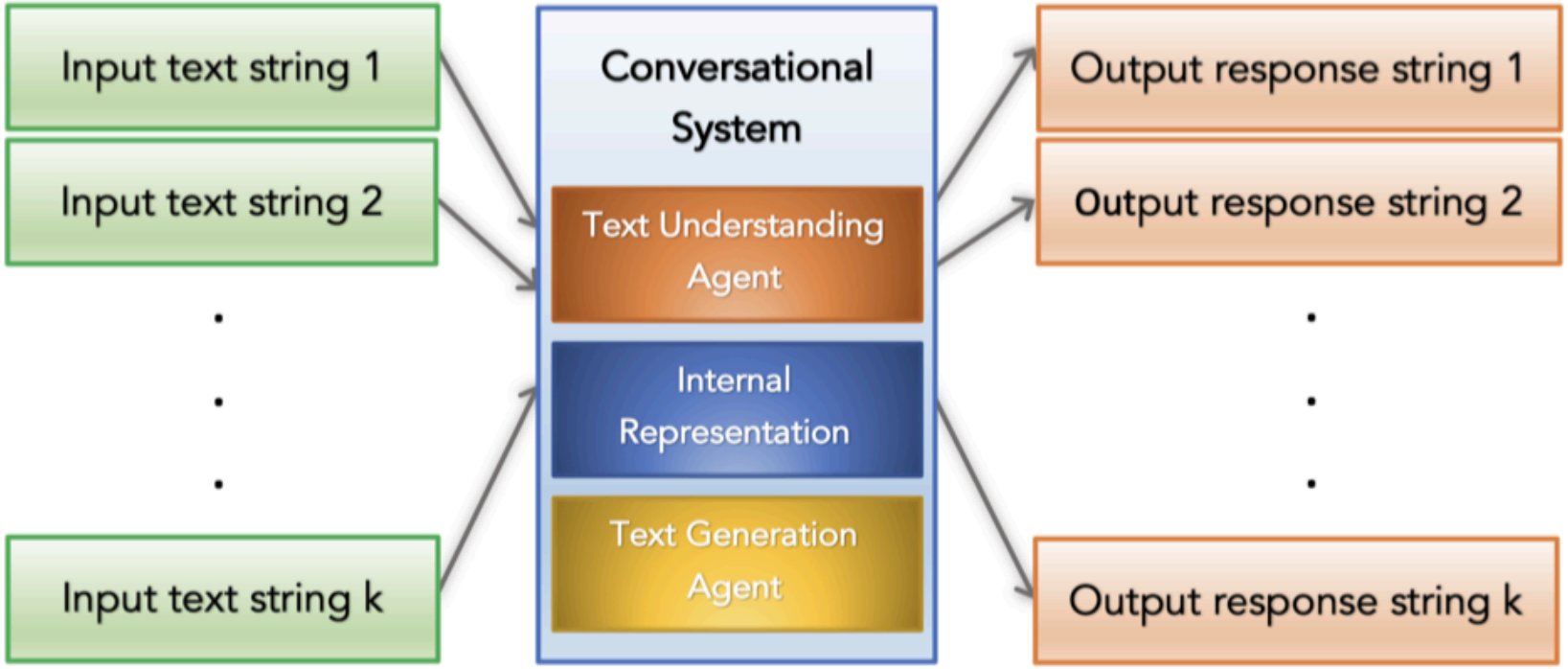


Machine Translation

ROBOT INTELLIGENCE LAB



Conversational System

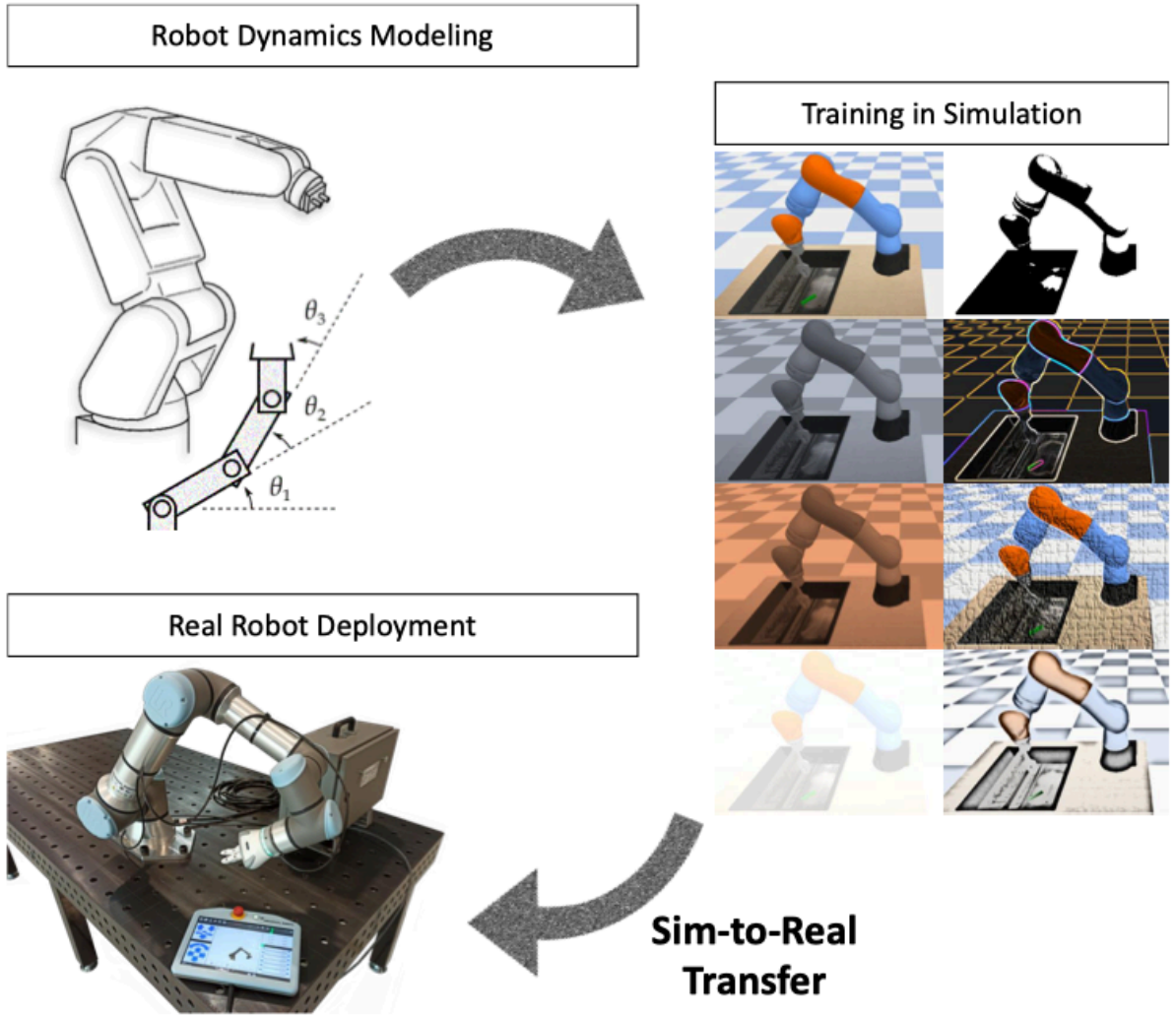


Robotics

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

Sim2Real

ROBOT INTELLIGENCE LAB



Sim2Real

ROBOT INTELLIGENCE LAB

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 radomization.	✓(sim)	PyBullet	✗	DDPGfD	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	Hierarchichal 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

ROBOT INTELLIGENCE LAB

- Dexterous Robotic Manipulation
  - peg-in-hole
  - deformable object manipulation
  - multi-fingered hands
  - force control policies
- Robotic Navigation (+Locomotion)
  - quadruped robot locomotion
  - visual navigation
- Other Applications
  - tactile sensing
  - multi agent manipulation

ROBOT INTELLIGENCE LAB

Audio-Based Applications

"A Survey on Deep Reinforcement Learning for Audio-Based Applications," 2021

https://arxiv.org/pdf/2101.00240.pdf

RL for Audio

- Automatic speech recognition
- Spoken dialogue systems
- emotions modeling
- audio enhancement
- music listening and generation
- robotics, control and interaction

Appl. Area	State Representations $\mathcal{S}$, Actions $\mathcal{A}$, and Reward Functions $\mathcal{R}$	Popular Datasets
Automatic Speech Recognition	$\mathcal{S}$: States are learnt representations from input speech features (e.g. fMLLR or MFCC vectors [93]). $\mathcal{A}$: Actions include phonemes, graphemes, commands, or candidates from the ASR N-best list. $\mathcal{R}$: They have included binary rewards (positive for selecting the correct choice, 0 otherwise), and non-binary sentence/token rewards based on the Levenshtein distance algorithm.	• LibriSpeech [94] • TED-LIUM [95] • Wall Street Journal [96] • SWITCHBOARD [97] • TIMIT [98]
Spoken Dialogue Systems	$\mathcal{S}$: They encode the uttered words by the system and recognised user words into a dialogue history and some additional information from classifiers such as user goals, user intents, speech recognition confidence scores, and visual information (in the case of multimodal systems), among others. $\mathcal{A}$: While actions in task-oriented systems include slot requests/confirmations/apologies, slot-value selection, ask question, data retrieval, information presentation, among others; actions in open-ended systems include either all possible sentences (infinite) or clusters of sentences (finite). $\mathcal{R}$: They vary depending on the company/project requirements and tend to include sparse and non-sparse numerical rewards such as dialogue length, task success, dialogue similarity, dialogue coherence, dialogue repetitiveness, game scores (in the case of game-based systems), among others.	• SGD [99] • DSTC [100] • Frames [101] • MultiWOZ [102] • SubTle Corpus [103] • Simulations [104] • Other datasets [105]
Speech Emotion Recognition	$\mathcal{S}$: Speech features (e.g., MFCC) are considered as input features. $\mathcal{A}$: Actions include speech emotion labels (e.g. unhappy, neutral, happy), sentiment detection (e.g. negative, neutral, positive), and termination from utterance listening. $\mathcal{R}$: Binary reward functions have been used (positive for choosing the correct choice, 0 otherwise).	• EMODB [106] • IEMOCAP [107] • MSP-IMPROV [108] • SEMAINE [109] • MELD [110]
Audio Enhancement	$\mathcal{S}$: States are learnt from clean and noisy acoustic features. $\mathcal{A}$: Finding closest cluster and its index, time-frequency mask estimation, and increasing or decreasing the parameter values of the speech-enhancement algorithm. $\mathcal{R}$: Positive rewards for correct choice, negative otherwise.	• DEMAND [111] • CHiME-3 [112] • WHAMR [113]
Music Generation	$\mathcal{S}$: State representations are learned from Musical notes. $\mathcal{A}$: Musical generation and next note selection are considered as actions. $\mathcal{R}$: Binary reward functions based on hard-coded musical theory rules, including the likelihood of actions.	• Classical piano MIDI database [114] • MusicNet dataset [115] • JSB Chorales dataset [116]
Robotics, Control and Interaction	$\mathcal{S}$: They encode visual and verbal representations derived from image embeddings, speech features, and word or sentence embeddings. Additional information include user intents, speech recognition scores, human activities, postures, emotions, and body joint angles, among others. $\mathcal{A}$: They include motor commands (e.g. gestures, locomotion, navigation, manipulation, gaze) and verbalizations such as dialogue acts and backchannels (e.g. laughs, smiles, noddings, head-shakes). $\mathcal{R}$: They are based on task success (positive rewards for achieving the goal, negative rewards for failing the task, and zero/shaped rewards otherwise) and user engagement.	• AVDIAR [117] • NLI Corpus [118] • VEIL dataset [119] • Simulations [120] • Real-world interactions [121]

Automatic Speech Recognition (ASR)

- Automatic speech recognition (ASR) is the process of converting a speech signal into its corresponding text by algorithms.
- Contemporary ASR technology has reached great levels of performance due to advancements in DL models. The performance of ASR systems, however, relies heavily on supervised training of deep models with large amounts of transcribed data.
- To broaden the scope of ASR, different studies have attempted RL based models with the ability to learn from **feedback**. This form of learning aims to reduce transcription costs and time by humans providing positive or negative rewards instead of detailed transcriptions.

Spoken Dialogue Systems (SDSs)

- Spoken dialogue systems are gaining interest due to many applications in customer services and goal-oriented humancomputer-interaction.
- The task of a dialogue manager in SDSs is to select actions based on observed events.
- Researchers have shown that the action selection process can be effectively optimised using RL to model the dynamics of spoken dialogue as a fully or **partially observable Markov Decision Process**.

Emotions Modelling

ROBOT INTELLIGENCE LAB

- Emotions are essential in vocal human communication, and they have recently received growing interest by the research community.
- This line of research is categorised into two areas: **emotion recognition** in conversations, and affective **dialogue generation**.
- Speech emotion recognition (SER) can be used as a **reward** for RL based dialogue systems.

Audio Enhancement

- The performance of audio-based intelligent systems is critically **vulnerable** to noisy conditions and degrades according to the noise levels in the environment.
- In DL-based systems, the audio enhancement module is generally optimised separately from the main task such as minimisation of word error rate.
- Besides the speech enhancement module, there are different other units in speech-based systems which increase their complexity and make them **non-differentiable**.
- In such situations, **DRL** can achieve complex goals in an iterative manner, which makes it suitable for such applications.

ROBOT INTELLIGENCE LAB

Data Caching in Edge Network

"A SURVEY OF DEEP LEARNING FOR DATA CACHING IN EDGE NETWORK,", 2020

https://arxiv.org/pdf/2008.07235.pdf

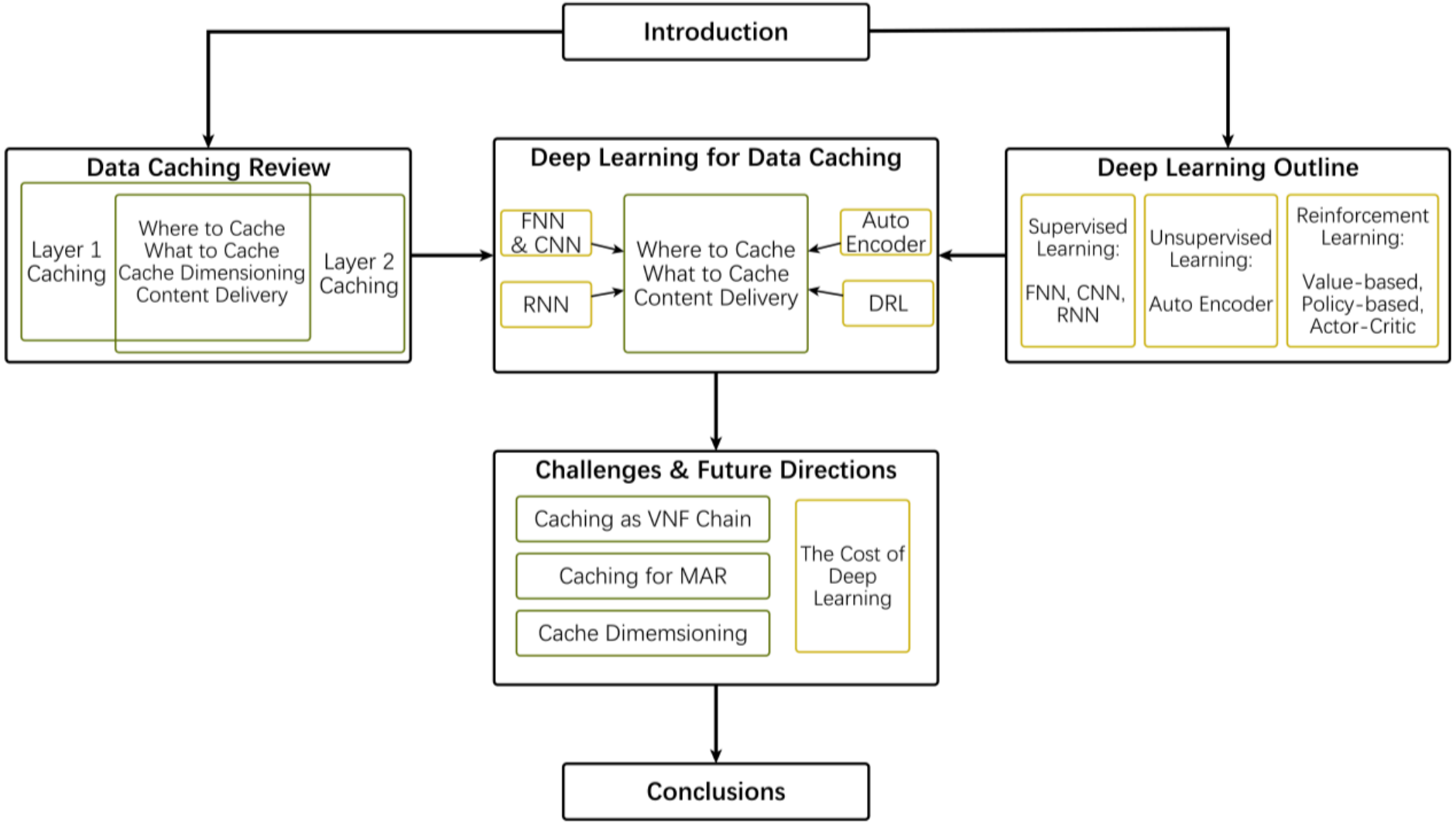
Introduction

ROBOT INTELLIGENCE LAB

- Undoubtedly, future 5G and beyond mobile communication networks will have to address stringent requirements of delivering popular content at **ultra high speeds** and **low latency** due to the proliferation of advanced mobile devices and data rich applications.
- In that ecosystem, **edge-caching** has received significant research attention as an efficient technique to reduce **delivery latency** and network congestion especially during peak-traffic times or during unexpected network congestion episodes by bringing popular data closer to the end users.

Data Caching

ROBOT INTELLIGENCE LAB



ROBOT INTELLIGENCE LAB

Smart Building Energy Management

"Deep Reinforcement Learning for Smart Building Energy Management: A Survey," 2020

https://arxiv.org/pdf/2008.05074.pdf

Introduction

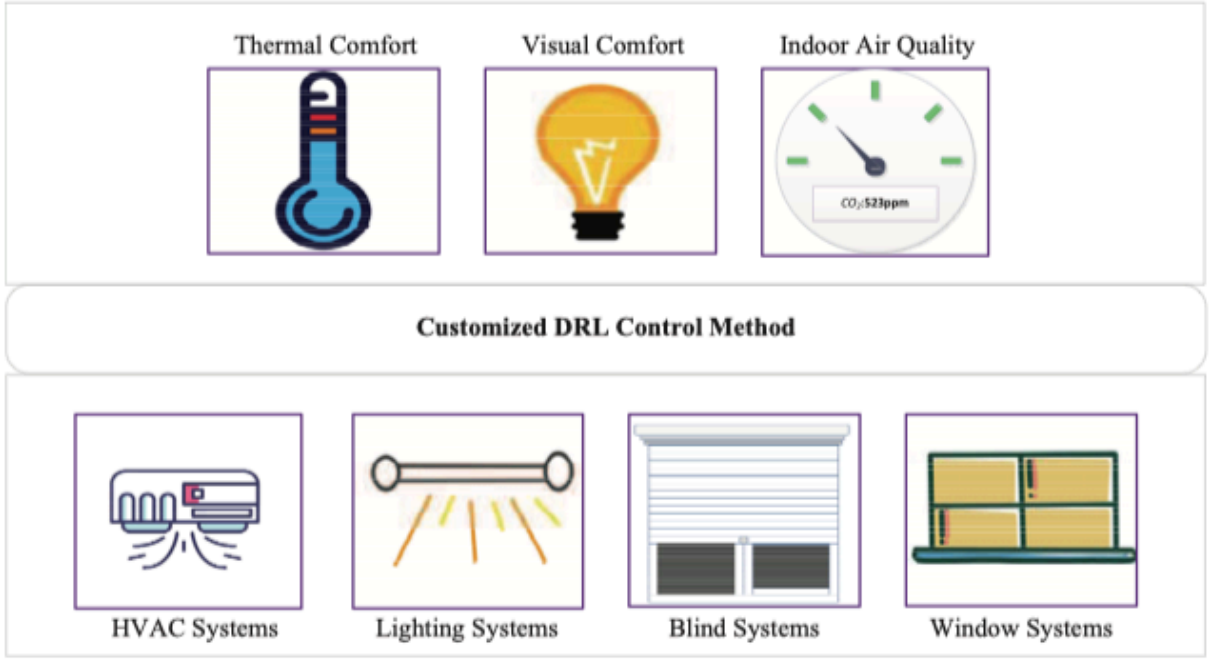
ROBOT INTELLIGENCE LAB

- Global buildings consumed 30% of total energy and generated 28% of total carbon emission in 2018, which leads to economic and environmental concerns.
- Therefore, it is of great significance 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**.
  - There are many spatially and temporally operational constraints.
  - Building energy optimization problems may have **extremely large solution spaces**, which can not be solved in real-time by traditional methods.

RL for Smart Building Energy Management

ROBOT INTELLIGENCE LAB

- What do we optimize?
  - 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

ROBOT INTELLIGENCE LAB

News Recommendation

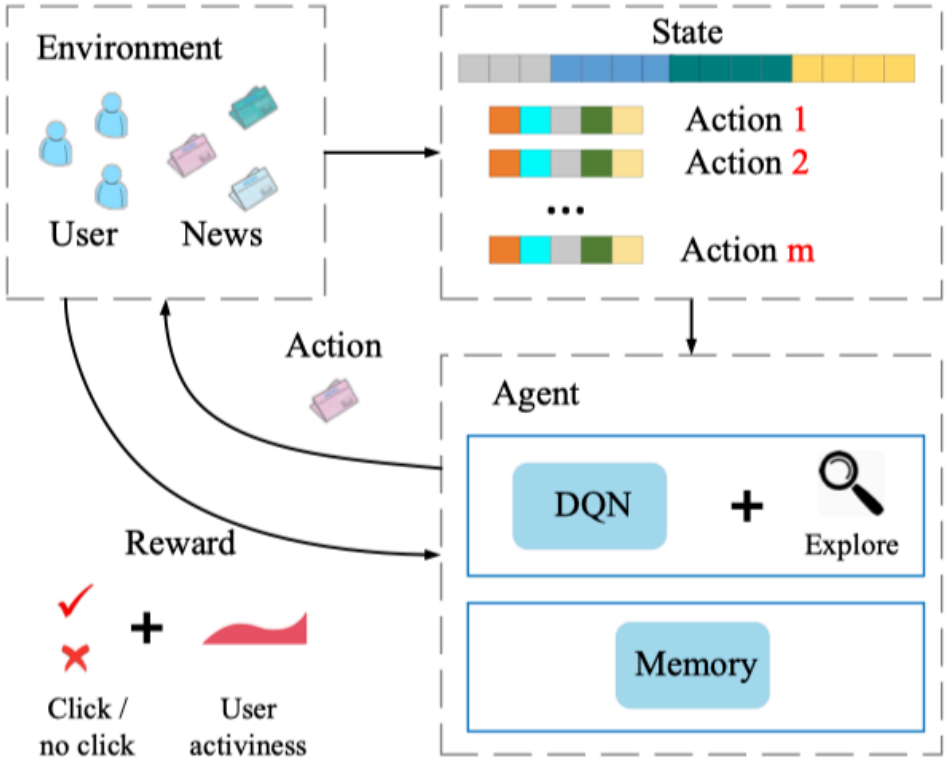
"DRN: A Deep Reinforcement Learning Framework for News Recommendation," 2018

http://www.personal.psu.edu/~gjz5038/paper/www2018_reinforceRec/www2018_reinforceRec.pdf

News Recommendation

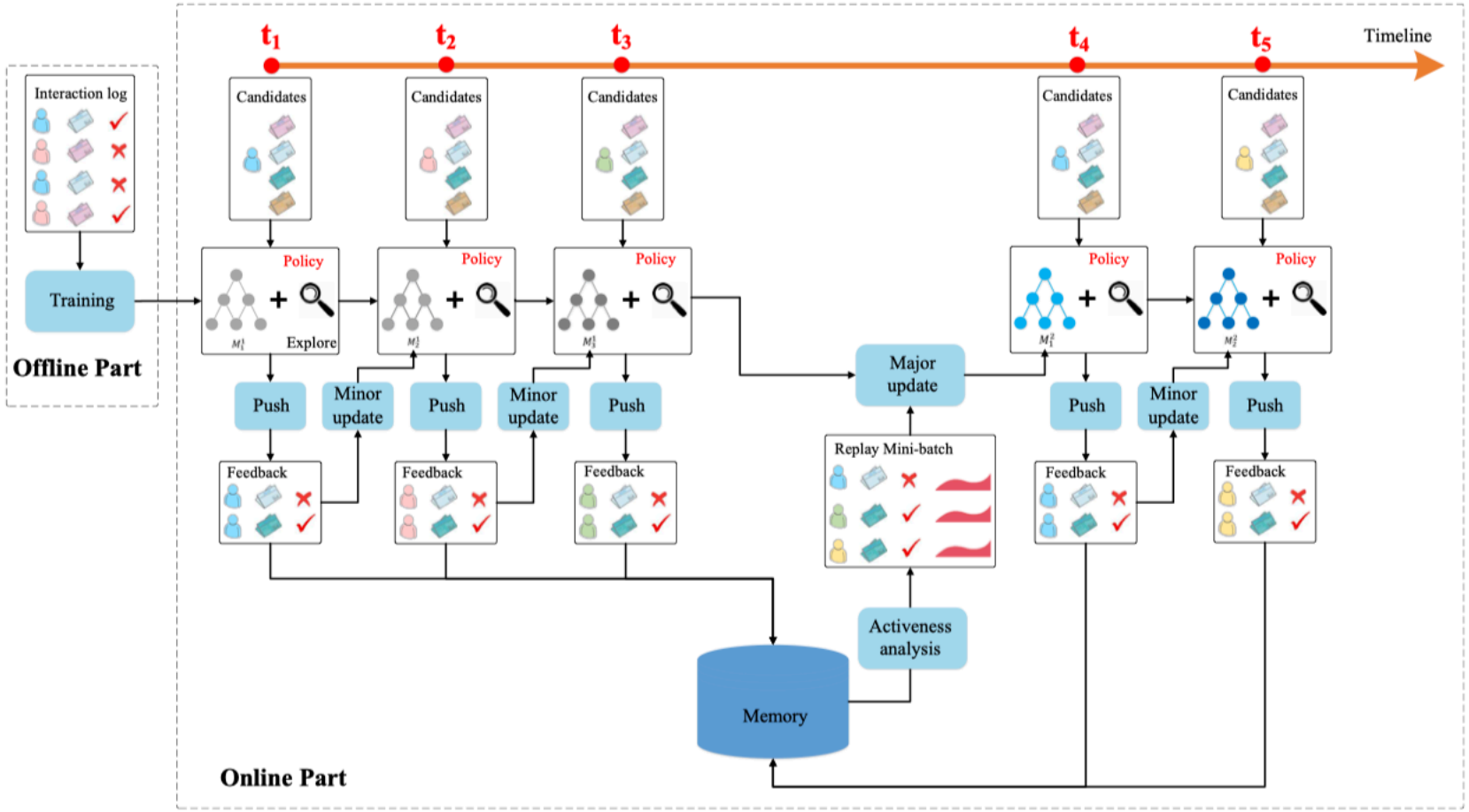
ROBOT INTELLIGENCE LAB

- **Online personalized news recommendation** is a highly challenging problem due to the dynamic nature of news features and user preferences.
- **Deep Q-Learning** based recommendatin framework is presented in this paper as it can **model future reward** explicitly.



Model

ROBOT INTELLIGENCE LAB



ROBOT INTELLIGENCE LAB

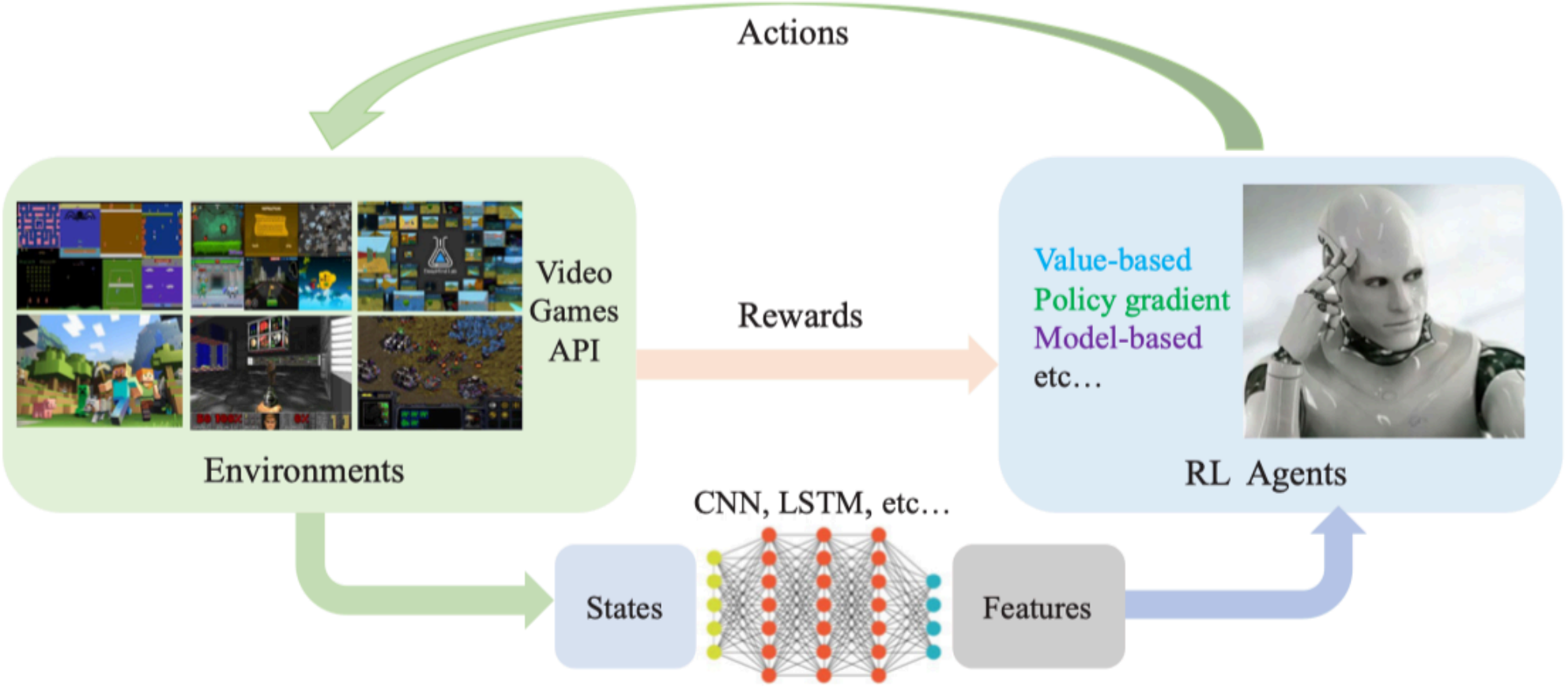
Video Games

"A Survey of Deep Reinforcement Learning in Video Games," 2019

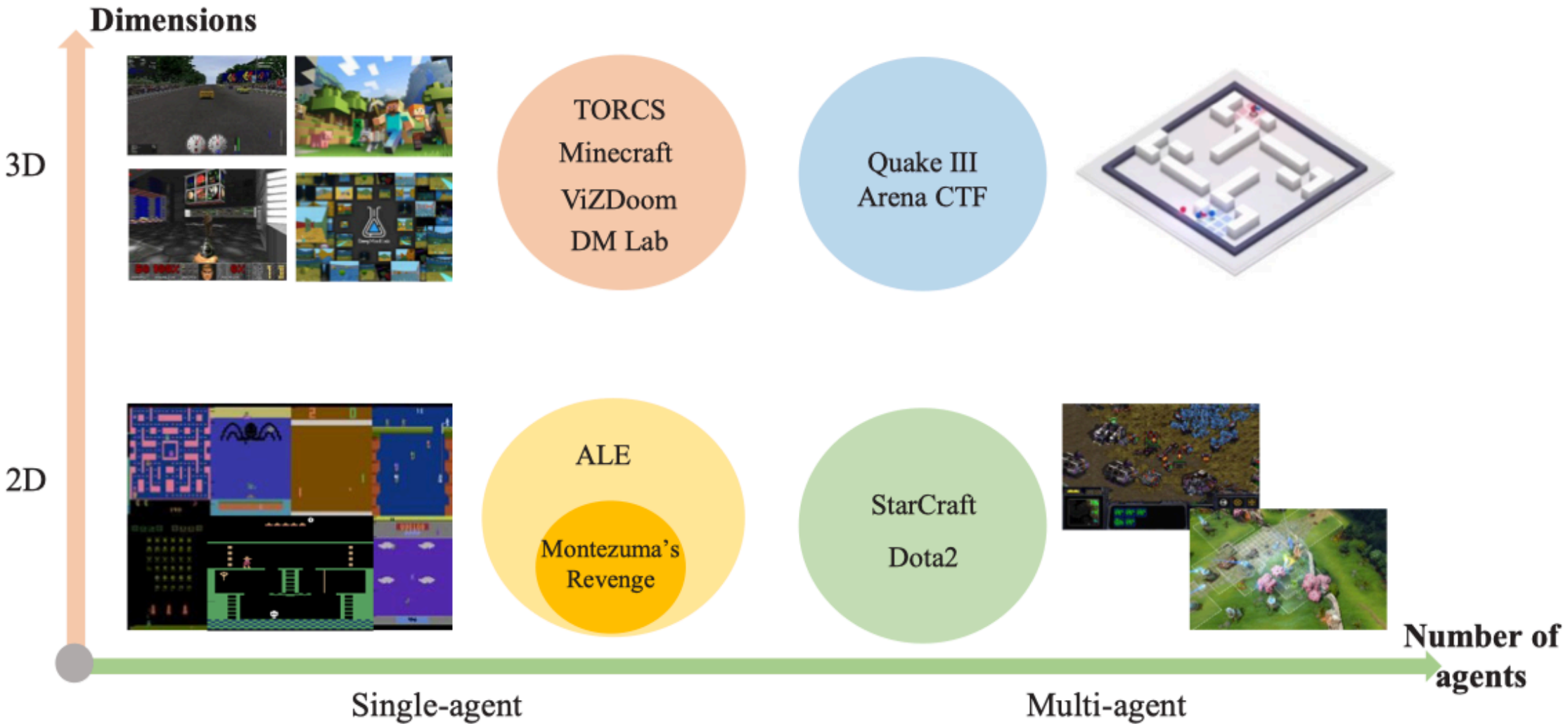
https://arxiv.org/pdf/1912.10944.pdf

DRL for Video Games

ROBOT INTELLIGENCE LAB



ROBOT INTELLIGENCE LAB



ROBOT INTELLIGENCE LAB

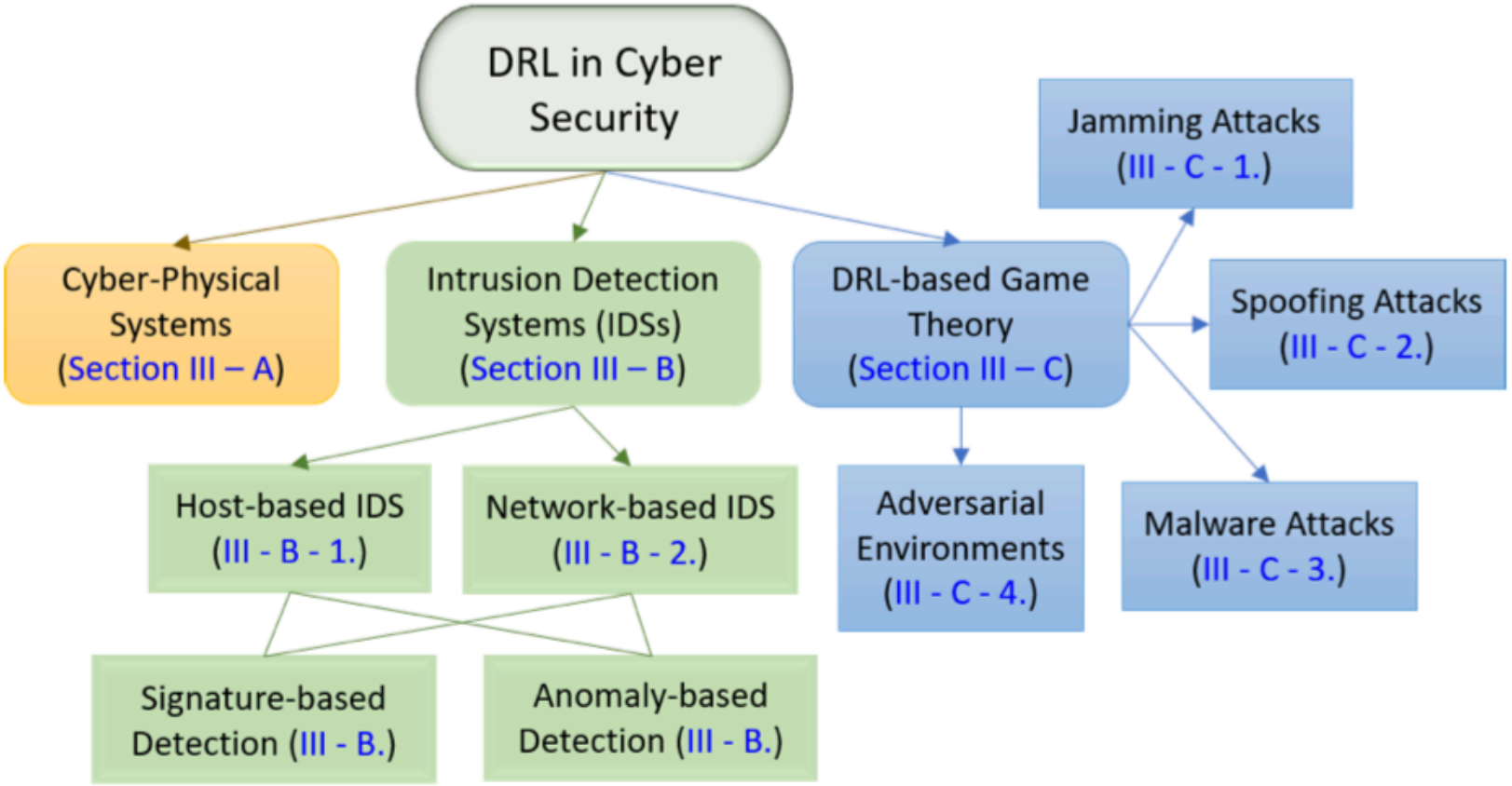
Cyber Security

"Deep Reinforcement Learning for Cyber Security," 2020

https://arxiv.org/pdf/1906.05799.pdf

RL in Cyber Security

ROBOT INTELLIGENCE LAB



Defense for Cyber-Physical Systems

- Investigations of defense methods for **cyber-physical systems** (CPS) against cyber attacks have received considerable attention and interests from the cyber security research community.
  - **CPS** is a mechanism controlled by computer-based algorithms facilitated by **internet integration** which provides efficient management of distributed **physical systems**.
  - It has been used in manufacturing, healt monitoring, smart grid, and transportation.
- The CPS defense problem is modeled as a **two-player zero-sum game** by which utilities of players are summed up to zero at each time step.
- The defender is represented by an actor-critic DRL algorithm where it can learn an optimal strategy to timely and accurately defend the CPS from unknown cyber attacks.

Intrusion Detection

- To detect **intrusions**, security experts conventionally need to observe and examine audit data, e.g., application traces, network traffic fow, and user command data, to **differentiate** between normal and abnormal behaviors.
- As the volume of audit data and complexity of intrusion behaviors increase, adaptive intrusion detection models demonstrate limited effectiveness because they can only handle temporally isolated labeled or unlabeled data.
- Reinforcement learning has been used for detecting intrusion.
  - The intrusion detection problem is converted to a state value prediction task.
  - RL-based intrusion detection shows higher accuracy and lower computational costs compared to supervised learning methods.

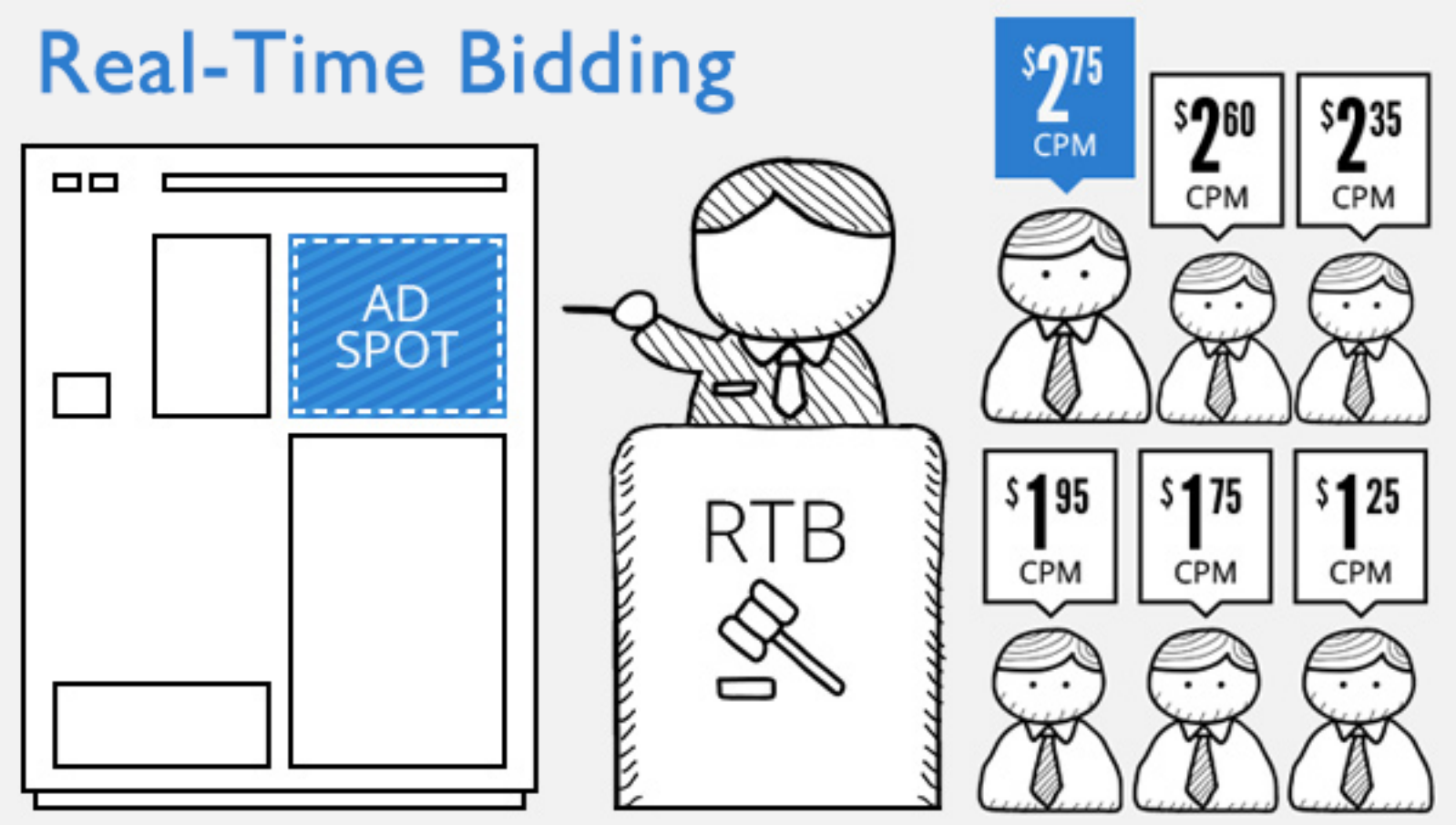
ROBOT INTELLIGENCE LAB

Marketing Advertising

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

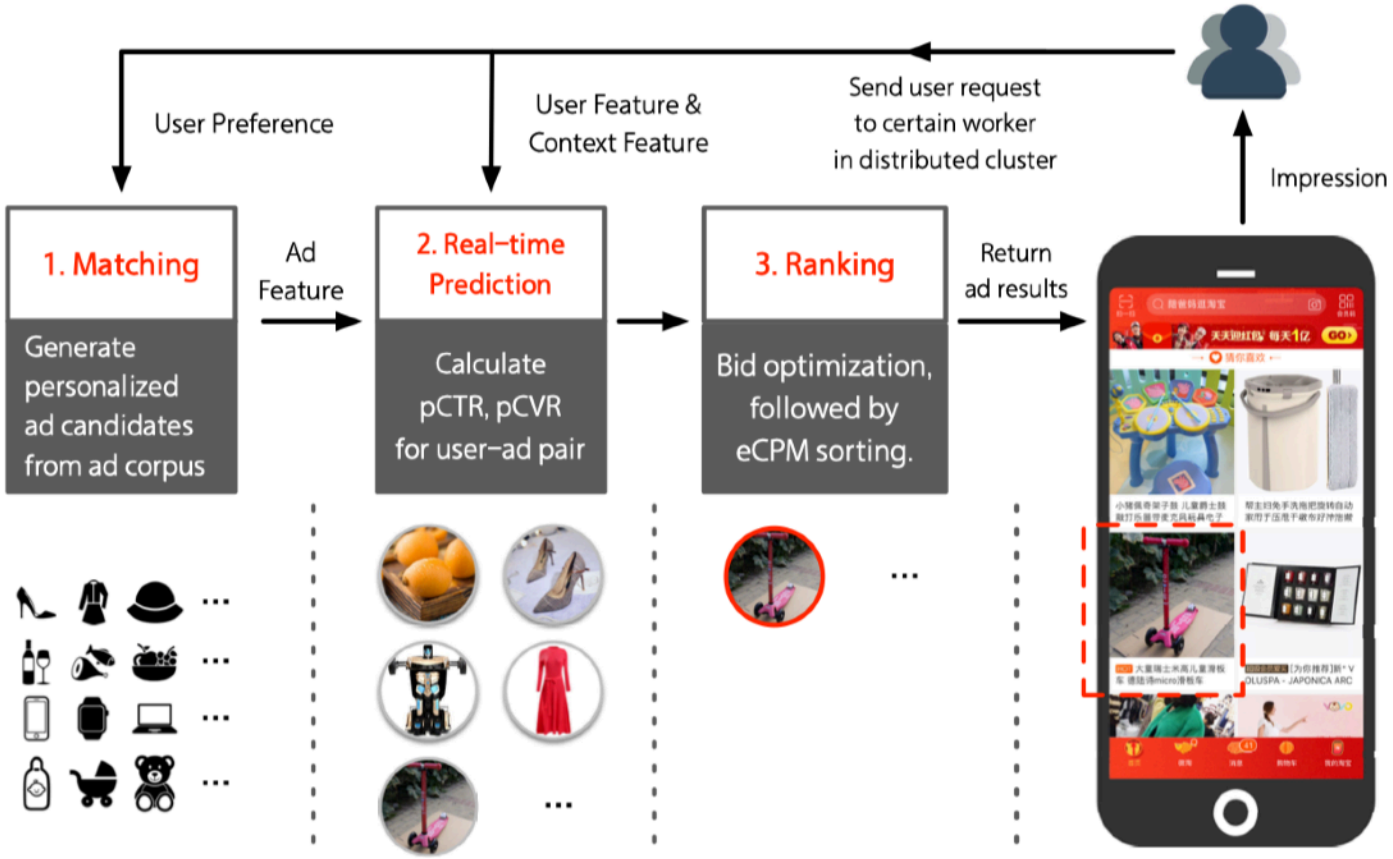
https://arxiv.org/pdf/1802.09756.pdf

Real-Time Bidding (RTB)



https://www.fraudlogix.com/affiliate-blog/marketers-real-time-bidding/

Taobao Display AD System



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

Real-Time Bidding (RTB)

- RTB is basically a multi player game with 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 **mult-agent advertising bidding**.

RTB as a Markov Game

ROBOT INTELLIGENCE LAB

- **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 competiveness or cooperativeness)
- **Action**: bid adjustment ratio
- **Reward**: total revenue of all bidding agents

Thank You