RL Research Group 20250403

Jump-Start Reinforcement Learning

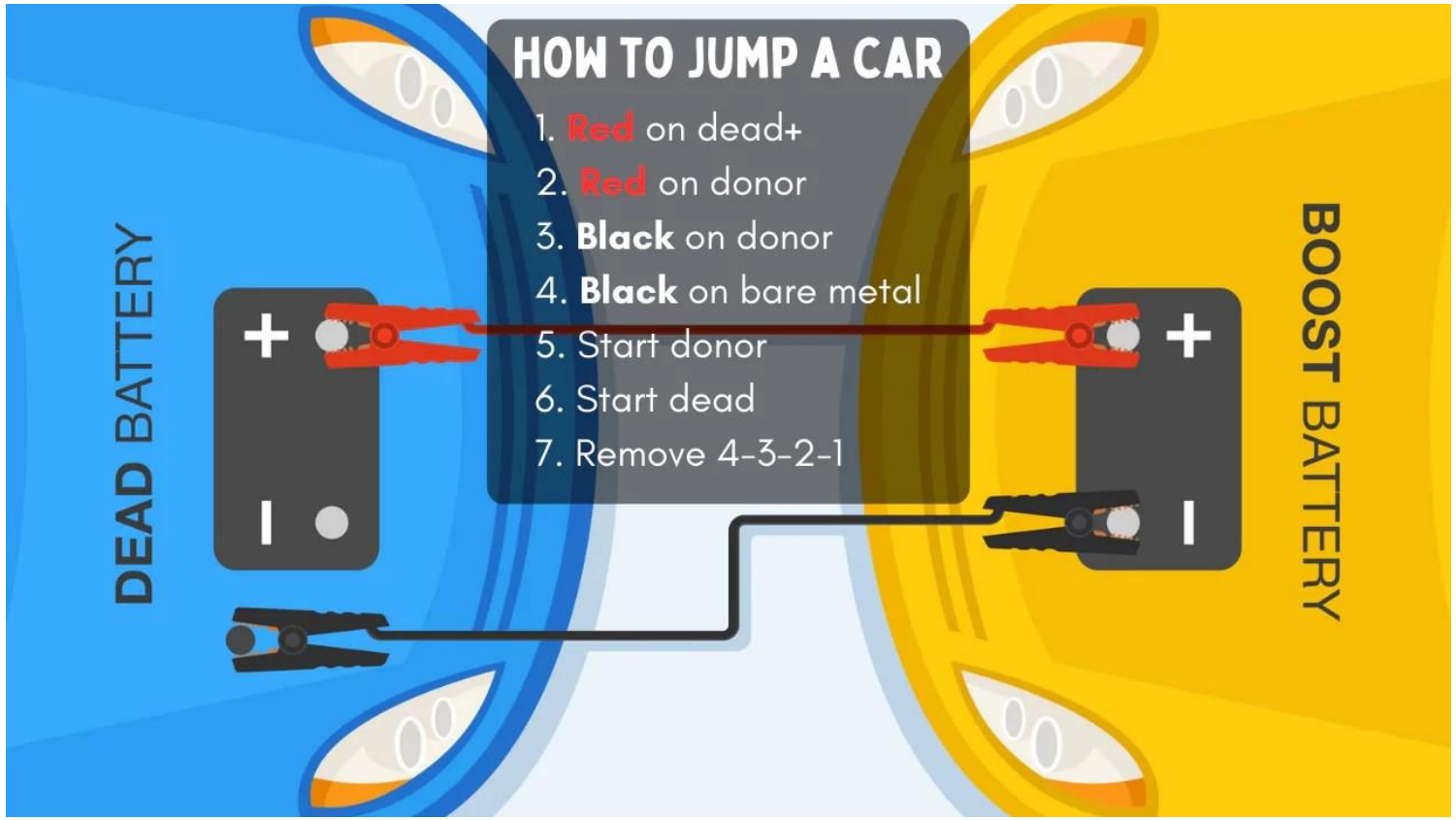
(ICML 2023)

김재훈

Introduction

❖ Jump-Start란?

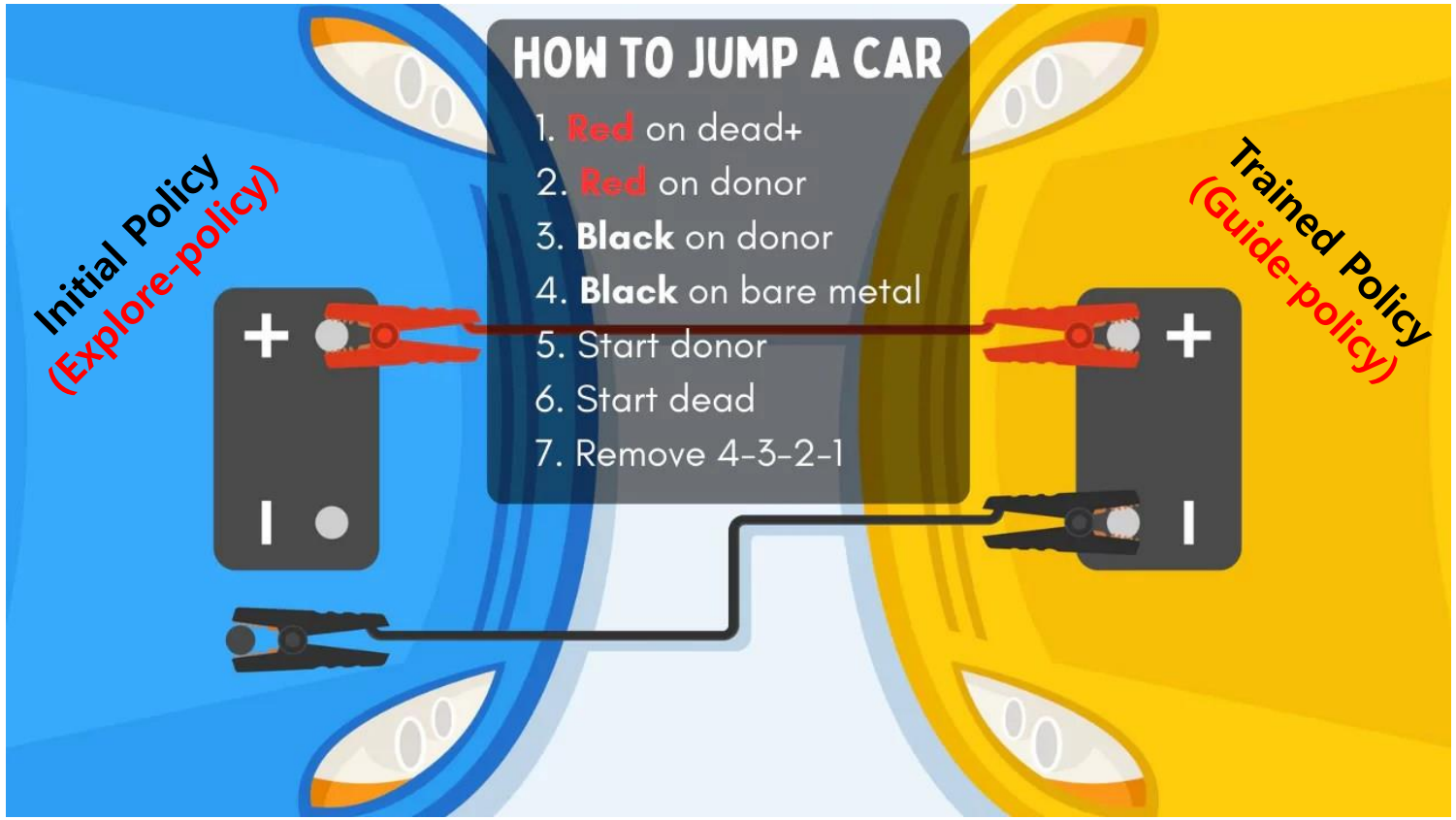
- 자동차가 방전되었을 때 전류를 흘려주어 시동을 거는 것을 의미함
- Jump-start로 시동을 건 뒤에 15분 정도 있으면 배터리가 충전된다고 함



Introduction

❖ Jump-Start Reinforcement Learning (JSRL)

- (일정 수준 이상) 잘 작동하는 정책을 활용하여 초기 정책의 성능을 샘플 효율적으로 끌어올리는 방법
- 모방학습, 강화학습으로 학습되어 있거나 룰-베이스 등 어떠한 정책이라도 잘 작동하는 정책으로 활용 가능



Introduction

❖ JSRL 들어가기 전에...

- 탐험이 어려운 환경에서 샘플 효율적으로 학습하는 방법론 연구
- Online 데이터 + Offline 데이터를 모두 활용하는 방법론
  - 논문에서는 모방학습 혹은 OfflineRL로 학습된 정책을 가지고 진행
  - 특히 데이터가 적은 상황(small data regime)에서 모방학습보다 더 나은 성능을 보인다고 강조
- Non-optimism 방식의 탐험 방법론 (ex. ϵ-greedy)
  - Optimism 방식은 reward bonus를 주는 RND나 ICM 등...
- Value-based method 사용 (Implicit-Q learning, QT-Opt)

Introduction

❖ 시작이 어려우면 사전 지식을 쓰자

- 탐험이 어려운 환경은 초반에 많은 수고를 들여야 함, 특히 ϵ-greedy 처럼 확률적으로 탐험을 수행하는 방법은 더욱…
- 물론 사전 지식을 활용할 수 있는 방법론을 쓰면 극복할 수 있음 (ex. Behavior cloning, Offline RL 또는 그 후 Online RL 수행)

  - Policy gradient 기반의 방법론은 사전 학습 후 추가 학습을 온라인으로 진행해도 무방
  - 반면 value-based 방법론은 OOD 데이터로 학습하는 것에 취약하기 때문에 해당 방식 적용이 어려움 → 너무 좋은 state-action만을 방문

    Bootstrap 방식이기 때문에 적절하지 않은 target Q-value 연산

  - Value-based 방법론은 state-action pair에 대한 정확한 가치 판단이 중요하기 때문에 좋은/나쁜 경우를 모두 학습해야 함

Jump-Start Reinforcement Learning으로 위의 사항들을 해결

Algorithm

❖ 서로 다른 두 정책이 교대하며 학습

- **Guide-Policy**: 사전 지식으로 학습된 정책 (논문에서는 모방학습 기반의 정책을 사용)
- **Exploration-Policy**: 랜덤하게 초기화된 정책 (value-based, non-optimism exploration)

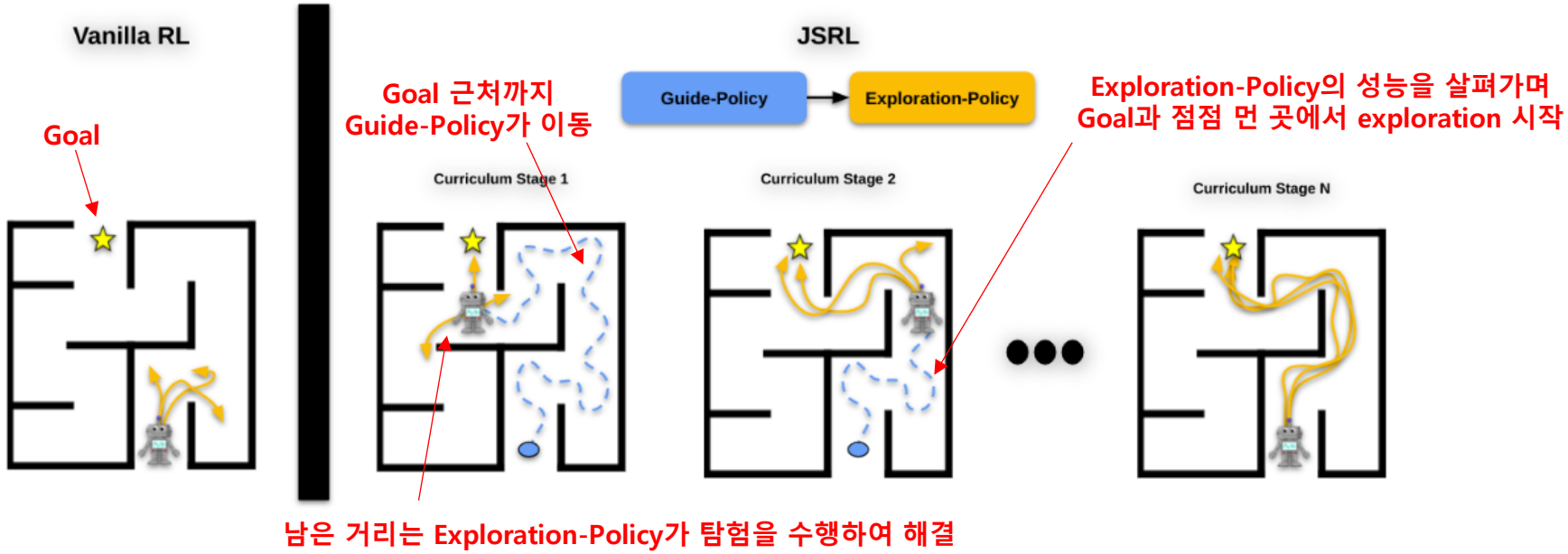


Figure 1. We study how to efficiently bootstrap value-based RL algorithms given access to a prior policy. In vanilla RL (left), the agent explores randomly from the initial state until it encounters a reward (gold star). JSRL (right), leverages a guide-policy (dashed blue line) that takes the agent closer to the reward. After the guide-policy finishes, the exploration-policy (solid orange line) continues acting in the environment. As the exploration-policy improves, the influence of the guide-policy diminishes, resulting in a learning curriculum for bootstrapping RL.

Algorithm

❖ Concern 1. 탐험이 어려운 환경은 초반에 많은 수고를 들여야 함

- Goal 근처까지 도달한 다음에 탐험을 시작하기 때문에 더 빠르게 보상을 접하고 정책을 업데이트할 수 있음

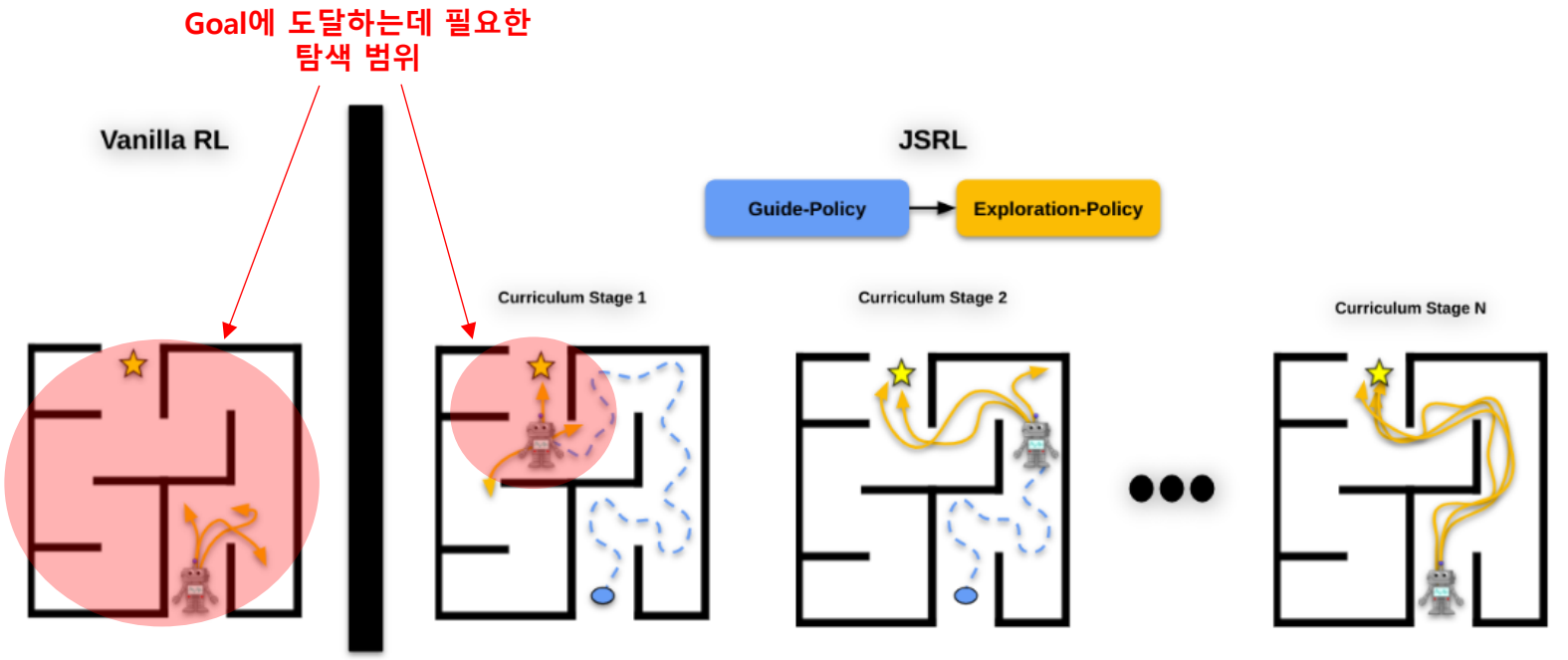


Figure 1. We study how to efficiently bootstrap value-based RL algorithms given access to a prior policy. In vanilla RL (left), the agent explores randomly from the initial state until it encounters a reward (gold star). JSRL (right), leverages a guide-policy (dashed blue line) that takes the agent closer to the reward. After the guide-policy finishes, the exploration-policy (solid orange line) continues acting in the environment. As the exploration-policy improves, the influence of the guide-policy diminishes, resulting in a learning curriculum for bootstrapping RL.

Algorithm

❖ Concern 2. Value-based 방법론은 OOD 데이터로 학습하는 것에 취약

- 남은 거리의 데이터는 exploration-policy가 직접 수집하고 학습하므로 해당 문제가 해결됨
- 또한 랜덤 초기화된 정책으로 시작하기 때문에 초반에는 매우 다양한 시행착오를 겪으면서 학습

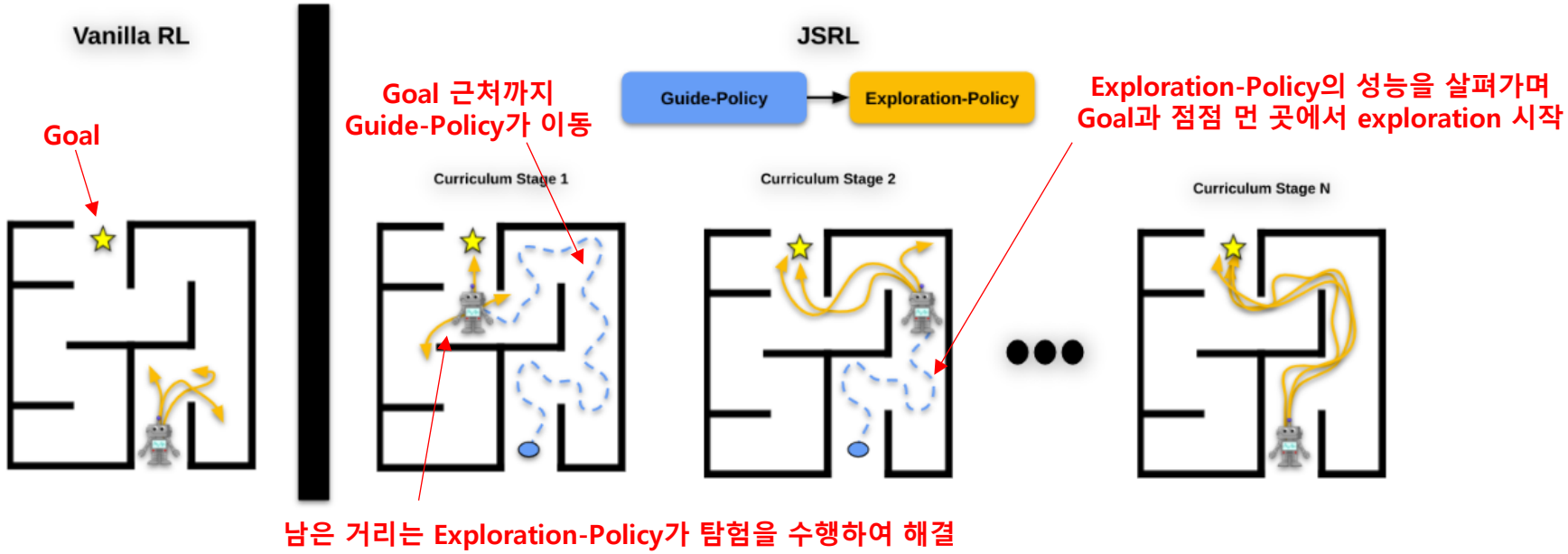


Figure 1. We study how to efficiently bootstrap value-based RL algorithms given access to a prior policy. In vanilla RL (left), the agent explores randomly from the initial state until it encounters a reward (gold star). JSRL (right), leverages a guide-policy (dashed blue line) that takes the agent closer to the reward. After the guide-policy finishes, the exploration-policy (solid orange line) continues acting in the environment. As the exploration-policy improves, the influence of the guide-policy diminishes, resulting in a learning curriculum for bootstrapping RL.

Algorithm

❖ **Pseudo-code**

Task를 완료하는데 필요한 모든 수행 과정 **(개념)**

성능의 moving average를 구하고 tolerance 수준을 넘기면 다음 guide step 진행
(moving average: 3 for grasping, 5 for D4RL / tolerance: 너무 크면 안 좋다 (구체적으로 제시 X))

Algorithm 1 Jump-Start Reinforcement Learning

Input: guide-policy π^g, performance threshold β, task horizon H, a sequence of initial guide-steps $H_1, H_2, \cdots, H_n$, where $H_i \in [H]$ for all $i \leq n$.

Initialize exploration-policy from scratch or with the guide-policy $\pi^e \leftarrow \pi^g$. Initialize Q-function $\hat{Q}$ and dataset $\mathcal{D} \leftarrow \varnothing$.

for current guide step $h = H_1, H_2, \cdots, H_n$ **do**

  Set the non-stationary policy $\pi_{1:h} = \pi^g, \pi_{h+1:H} = \pi^e$

  Roll out the policy π to get trajectory $\{(s_1, a_1, r_1), \cdots, (s_H, a_H, r_H)\}$; Append the trajectory to the dataset $\mathcal{D}$.

  $\pi^e, \hat{Q} \leftarrow \text{TRAINPOLICY}(\pi^e, \hat{Q}, \mathcal{D})$

  if $\text{EVALUATEPOLICY}(\pi) \geq \beta$ **then**

    Continue

  end if

end for

얼만큼의 간격으로 멀어질지는 서치를 통해 결정(...)
에이전트가 특정 구간에 갇혀버릴 때까지 간격을 벌림

Guide-policy가 해당 task를 100회 수행해본 뒤
각 수행마다 완료까지 필요했던 시퀀스 수의 평균

앞선 설명에서는 사전학습된 value-based method의 단점을 말해두고?

Exploration-policy 초기화

제일 먼 곳에서 시작하는 guide-step

Exploration-policy 업데이트

업데이트를 완료한 exploration-policy의 성능을 확인

Experiments

Cold-start 성능
(exploration-policy: **from scratch**)

❖ **Comparison with IL + RL baselines**

- Offline dataset이 적을 때 더 좋은 성능
- Vision-based 환경에서도 동일 (fig. 3)

Environment	Dataset	AWAC[1]	BC[1]	CQL[1]	IQL	IQL+JSRL (Ours)	
						Curriculum	Random
antmaze-umaze-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.2 ± 0.5	**15.6 ± 19.9**	10.4 ± 9.6
	10k	0.0 ± 0.0	1.0	0.0 ± 0.0	55.5 ± 12.5	**71.7 ± 14.5**	52.3 ± 26.7
	100k	0.0 ± 0.0	62.0	0.0 ± 0.0	74.2 ± 25.6	**93.7 ± 4.2**	92.1 ± 2.8
	1m (standard)	**93.67 ± 1.89**	61.0	64.33 ± 45.58	**97.6 ± 3.2**	**98.1 ± 1.4**	**95.0 ± 3.0**
antmaze-umaze-diverse-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	**3.1 ± 8.0**	**1.9 ± 4.8**
	10k	0.0 ± 0.0	1.0	0.0 ± 0.0	33.1 ± 10.7	**72.6 ± 12.2**	39.4 ± 20.1
	100k	0.0 ± 0.0	13.0	0.0 ± 0.0	29.9 ± 23.1	**81.3 ± 23.0**	**82.3 ± 14.2**
	1m (standard)	46.67 ± 3.68	80.0	0.50 ± 0.50	53.0 ± 30.5	**88.6 ± 16.3**	**89.8 ± 10.0**
antmaze-medium-play-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	10k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.1 ± 0.3	**16.7 ± 12.9**	3.8 ± 5.0
	100k	0.0 ± 0.0	0.0	0.0 ± 0.0	32.8 ± 32.6	**86.7 ± 3.7**	56.2 ± 28.8
	1m (standard)	0.0 ± 0.0	0.0	0.0 ± 0.0	**92.8 ± 2.7**	**91.1 ± 3.9**	**87.8 ± 4.2**
antmaze-medium-diverse-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	10k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	**16.6 ± 11.7**	5.1 ± 8.2
	100k	0.0 ± 0.0	0.0	0.0 ± 0.0	15.7 ± 17.7	**81.5 ± 18.8**	67.0 ± 17.4
	1m (standard)	0.0 ± 0.0	0.0	0.0 ± 0.0	**92.4 ± 4.5**	**93.1 ± 3.1**	**86.3 ± 5.9**
antmaze-large-play-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	10k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	**0.1 ± 0.2**	0.0 ± 0.0
	100k	0.0 ± 0.0	0.0	0.0 ± 0.0	2.6 ± 8.2	**36.3 ± 16.4**	17.7 ± 13.4
	1m (standard)	0.0 ± 0.0	0.0	0.0 ± 0.0	**62.4 ± 12.4**	**62.9 ± 11.3**	48.6 ± 10.0
antmaze-large-diverse-v0	1k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
	10k	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	**0.1 ± 0.2**	0.0 ± 0.0
	100k	0.0 ± 0.0	0.0	0.0 ± 0.0	4.1 ± 10.4	**34.4 ± 23.0**	22.4 ± 15.4
	1m (standard)	0.0 ± 0.0	0.0	0.0 ± 0.0	**68.3 ± 8.9**	**68.3 ± 8.8**	58.3 ± 6.5
door-binary-v0	100	**0.07 ± 0.11**	0.0	0.0 ± 0.0	**0.8 ± 3.8**	**0.4 ± 1.8**	**0.1 ± 0.2**
	1k	**0.41 ± 0.58**	0.0	0.0 ± 0.0	**0.5 ± 1.5**	**0.7 ± 1.0**	**0.45 ± 1.2**
	10k	1.93 ± 2.72	0.0	12.24 ± 24.47	10.6 ± 14.1	4.3 ± 8.4	**22.3 ± 11.6**
	100k (standard)	17.26 ± 20.09	0.0	8.28 ± 19.94	**50.2 ± 2.5**	28.5 ± 19.5	24.3 ± 11.5
pen-binary-v0	100	3.13 ± 4.43	0.0	**31.46 ± 9.99**	18.8 ± 11.6	24.3 ± 12.1	**29.1 ± 7.6**
	1k	1.43 ± 1.10	0.0	**54.50 ± 0.0**	30.1 ± 10.2	36.7 ± 7.9	**46.3 ± 6.3**
	10k	2.21 ± 1.30	0.0	**51.36 ± 4.34**	38.4 ± 11.2	44.3 ± 6.2	**52.1 ± 3.3**
	100k (standard)	1.23 ± 1.08	0.0	59.58 ± 1.43	**65.0 ± 2.9**	62.6 ± 3.6	60.6 ± 2.7
relocate-binary-v0	100	0.0 ± 0.0	0.0	0.0 ± 0.0	0.0 ± 0.0	**0.0 ± 0.1**	0.0 ± 0.0
	1k	**0.01 ± 0.01**	0.0	0.0 ± 0.0	0.0 ± 0.0	**0.0 ± 0.1**	0.0 ± 0.0
	10k	0.0 ± 0.0	0.0	**1.18 ± 2.70**	0.2 ± 0.3	0.6 ± 1.6	0.5 ± 0.7
	100k (standard)	0.0 ± 0.0	0.0	**4.44 ± 6.36**	**8.6 ± 7.7**	0.0 ± 0.1	**4.7 ± 4.2**

Table 1. Comparing JSRL with IL+RL baselines on D4RL tasks by using averaged normalized scores for D4RL Ant Maze and Adroit tasks. Each method pre-trains on an offline dataset and then runs online fine-tuning for 1m steps. Our method IQL+JSRL is competitive with IL+RL baselines in the full dataset setting, but performs significantly better in the small-data regime. For implementation details and more detailed comparisons, see Appendix A.2 and A.3

Experiments

❖ Warm-start vs. Cold-start

- Guide-policy를 exploration policy의 초기값으로 사용했을 때와의 비교
- Task와 offline data 수에 따라 성능 우위가 다름

Environment	JSRL: Random Switching		JSRL: Curriculum		IQL
	Warm-start	Cold-start	Warm-start	Cold-start	
pen-binary-v0	27.18 ± 7.77	$\mathbf{29.12 \pm 7.62}$	25.10 ± 8.73	24.31 ± 12.05	18.80 ± 11.63
door-binary-v0	0.01 ± 0.04	0.06 ± 0.23	$\mathbf{1.45 \pm 4.67}$	0.40 ± 1.80	0.84 ± 3.76
relocate-binary-v0	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	$\mathbf{0.01 \pm 0.06}$	0.01 ± 0.03

Table 3. Adroit 100 Offline Transitions

Environment	JSRL: Random Switching		JSRL: Curriculum		IQL
	Warm-start	Cold-start	Warm-start	Cold-start	
pen-binary-v0	$\mathbf{47.23 \pm 3.96}$	46.30 ± 6.34	34.23 ± 7.22	36.74 ± 7.91	30.11 ± 10.22
door-binary-v0	0.15 ± 0.25	0.45 ± 1.22	0.44 ± 0.89	$\mathbf{0.68 \pm 1.02}$	0.53 ± 1.46
relocate-binary-v0	$\mathbf{0.06 \pm 0.08}$	0.01 ± 0.04	0.05 ± 0.09	0.04 ± 0.10	0.01 ± 0.03

Table 4. Adroit 1k Offline Transitions

Experiments

❖ **Curriculum vs. Random switching (Everyday Robots 자체 시뮬레이터?)**

- Indiscriminate Grasping: 어떠한 물건을 집어도 reward 제공
- Instance Grasping: 특정한 물건을 집어야만 reward 제공
- 특정한 방문 순서보다는 좋은 상태를 방문하는 것이 더 중요함을 보여줌

Environment	Demo	AW-Opt	BC	QT-Opt	QT-Opt+JSRL	QT-Opt+JSRL Random
Indiscriminate Grasping	20	0.33 ± 0.43	0.19 ± 0.04	0.00 ± 0.00	$\mathbf{0.91 \pm 0.01}$	0.89 ± 0.00
Indiscriminate Grasping	200	$\mathbf{0.93 \pm 0.02}$	0.23 ± 0.00	0.92 ± 0.02	0.92 ± 0.00	0.92 ± 0.01
Indiscriminate Grasping	2k	0.93 ± 0.01	0.40 ± 0.06	0.92 ± 0.01	0.93 ± 0.02	$\mathbf{0.94 \pm 0.02}$
Indiscriminate Grasping	20k	0.93 ± 0.04	0.92 ± 0.00	0.93 ± 0.00	$\mathbf{0.95 \pm 0.01}$	0.94 ± 0.00
Instance Grasping	20	0.44 ± 0.05	0.05 ± 0.03	0.29 ± 0.20	$\mathbf{0.54 \pm 0.02}$	0.53 ± 0.02
Instance Grasping	200	0.44 ± 0.04	0.16 ± 0.01	0.44 ± 0.04	0.52 ± 0.01	$\mathbf{0.55 \pm 0.02}$
Instance Grasping	2k	0.42 ± 0.02	0.30 ± 0.01	0.15 ± 0.22	0.52 ± 0.02	$\mathbf{0.57 \pm 0.02}$
Instance Grasping	20k	0.55 ± 0.01	0.48 ± 0.01	0.27 ± 0.20	0.55 ± 0.01	$\mathbf{0.56 \pm 0.02}$

Table 2. Limiting the initial number of demonstrations is challenging for IL+RL baselines on the difficult robotic grasping tasks. Notably, only QT-Opt+JSRL is able to learn in the smallest-data regime of just 20 demonstrations, 100x less than the standard 2,000 demonstrations. For implementation details, see Appendix A.2.2