Incremental Learning of Retrievable Skills For Efficient Continual Task Adaptation

Daehee Lee, Minjong Yoo, Woo Kyung Kim, Wonje Choi, Honguk Woo
Sungkyunkwan University

Presented as a Poster at NeurIPS 2024 (Accepted, 6/6/6/5)

https://openreview.net/forum?id=RcPAJAnpnm

2025.03.09
TaeYoon Kwack
njj05043@g.skku.edu

Table of Contents

분석 중심으로 리뷰하라는 말씀을 반영하여 생략

Motivation

Conventional CiL

1. **데이터 비효율성:**
   - Continual Imitation Learning (CiL)에서는 다양한 전문가 시연 데이터를 통해 여러 작업을 학습해야함.
   - 그러나 전문가 시연 데이터는 획득 비용이 높고, 모든 가능한 시나리오를 포괄하기 어렵다는 문제 존재[1]

2. **비정상성(non-stationarity):**
   - 계속해서 변하는 비정상성(non-stationarity)의 현실공간에서 기존의 CiL 접근 방식은 적응하는 데 어려움이 있음
   - 반복되는 학습 과정에서 과거 학습된 정보를 쉽게 잃는 Catastrophic Forgetting 문제 발생[2]

3. **프라이버시 문제:**
   - 기존의 CiL 모델은 모든 시연 데이터를 저장하거나 재학습하는 방식을 주로 사용해왔으며, 이는 개인화된 사용자 데이터를 저장하게 되어 프라이버시 문제가 발생[3].

Motivation

Proposed Method: IsCiL

1. **Prototype-based Skill Incremental Learning:**
   기존에 학습한 스킬을 재활용하며 새로운 스킬을 효율적으로 추가 학습하도록 설계.

2. **Task-wise Selective Adaptation:**
   비정상적인 환경에서도 적절한 스킬을 선택해 변화하는 환경에 빠르게 적응할 수 있도록 합니다.

3. **Task Unlearning:**
   특정 작업에 관련된 스킬만 제거할 수 있어, 프라이버시 문제를 효과적으로 해결 가능.

Method: IsCiL

Overview

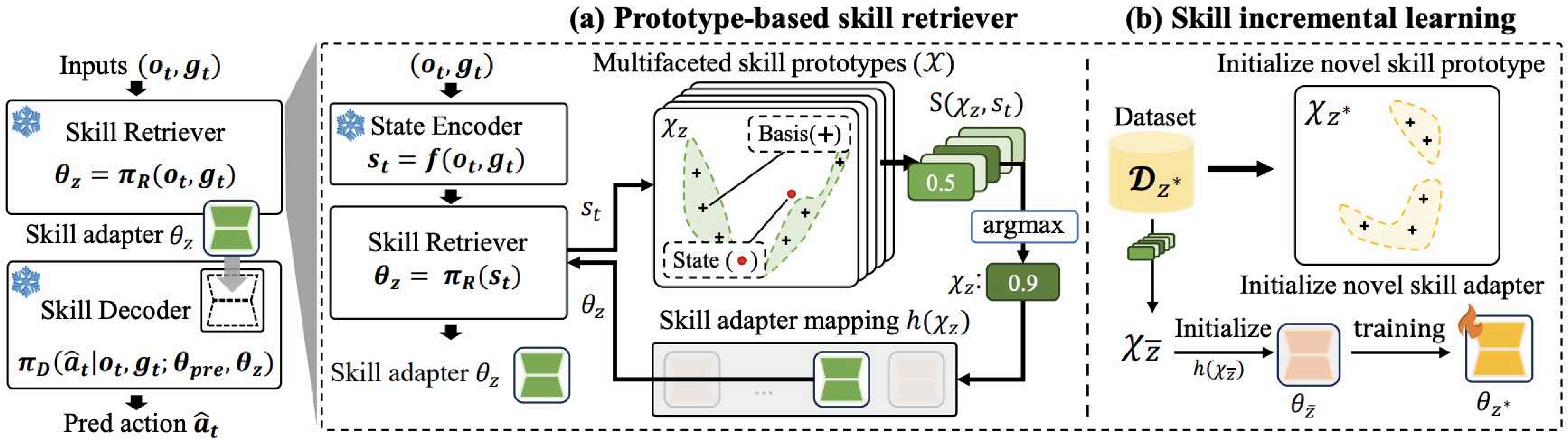


Figure 2: Overview of the IsCiL framework: (a) The prototype-based skill retriever sequentially utilizes a state encoder f, multifaceted skill prototypes $\mathcal{X}$, and a skill adapter mapping function h to identify the skill adapter θ_z. (b) Skill incremental learning involves the initialization and updating of the skill prototype χ_{z^*} and its corresponding adapter θ_{z^*}.

Components

- **State Encoder**
- **Multifaceted Skill Database**
- **Prototypes**
- **Skill Retriever**
- **Skill Adapter**
- **Pre-trained Basemodel**
- **Skill Decoder**

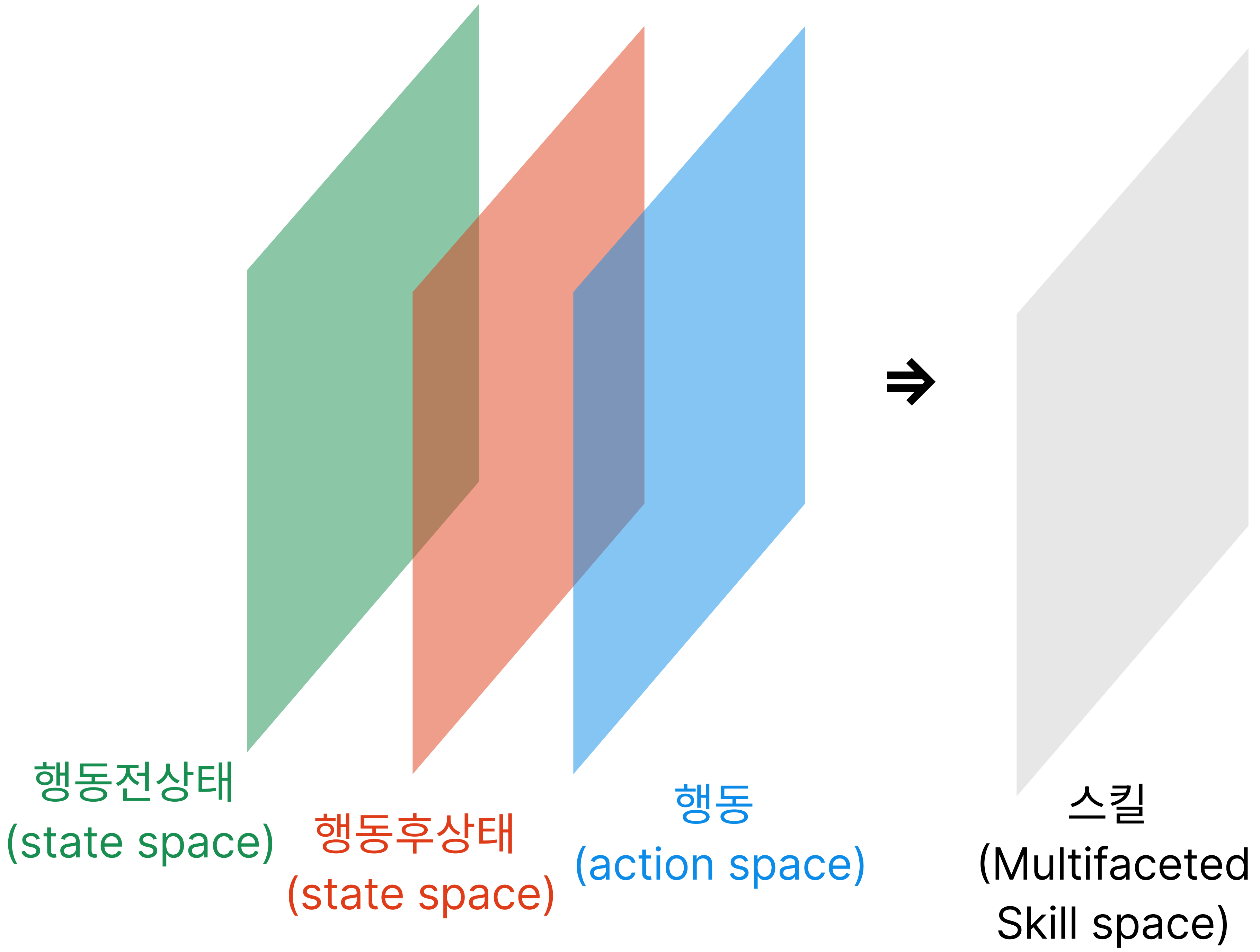
Method: IsCiL

1. Multifaceted Skill & Database

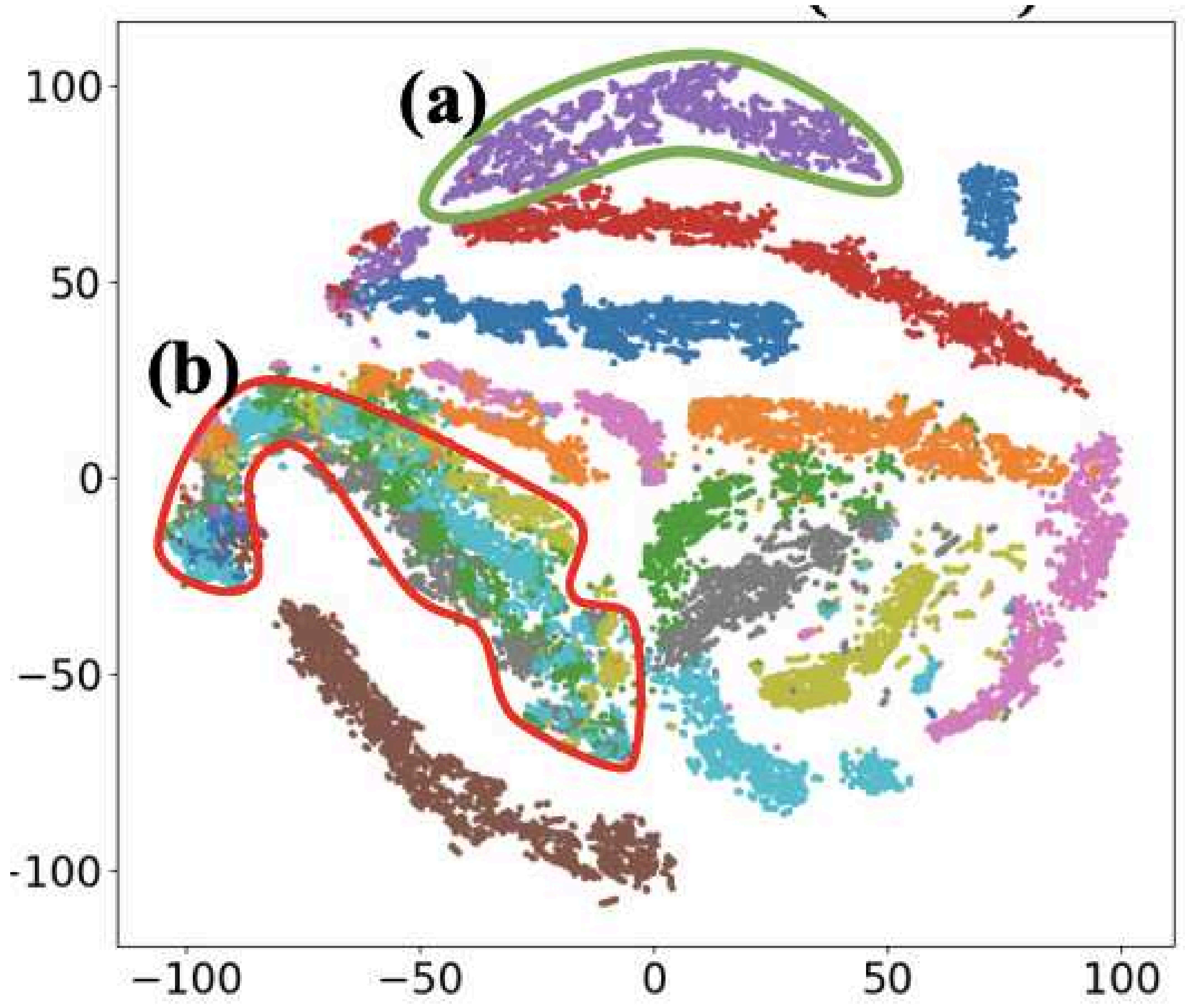
본 연구에서는 전문가의 시연 데이터 혹은 agent의 활동 기록 데이터를 **[행동전상태+행동후상태+행동]**의 벡터 형태로 저장하며 합쳐진 형태를 **Skill**이라 칭합니다. 이때 **[행동전상태+행동후상태]**는 **basis 혹은 state**라고 합니다.

Basis

Skill = [[o (행동 전 상태) | g (행동 후 상태)] | a (행동)]



Skill 공간은 action space와 state space가 포개어진 공간이다.

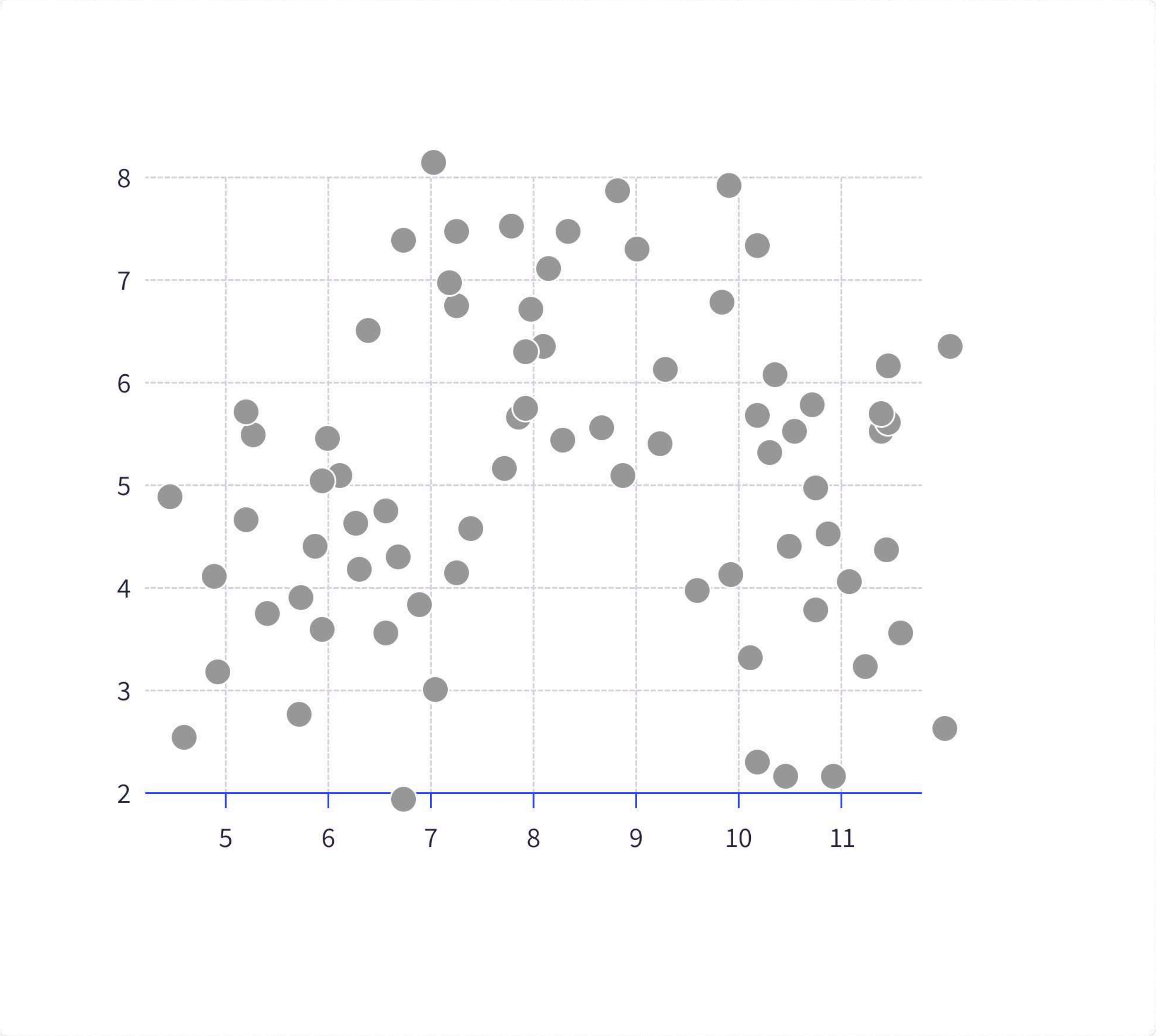


실제 skill들과 skill 공간

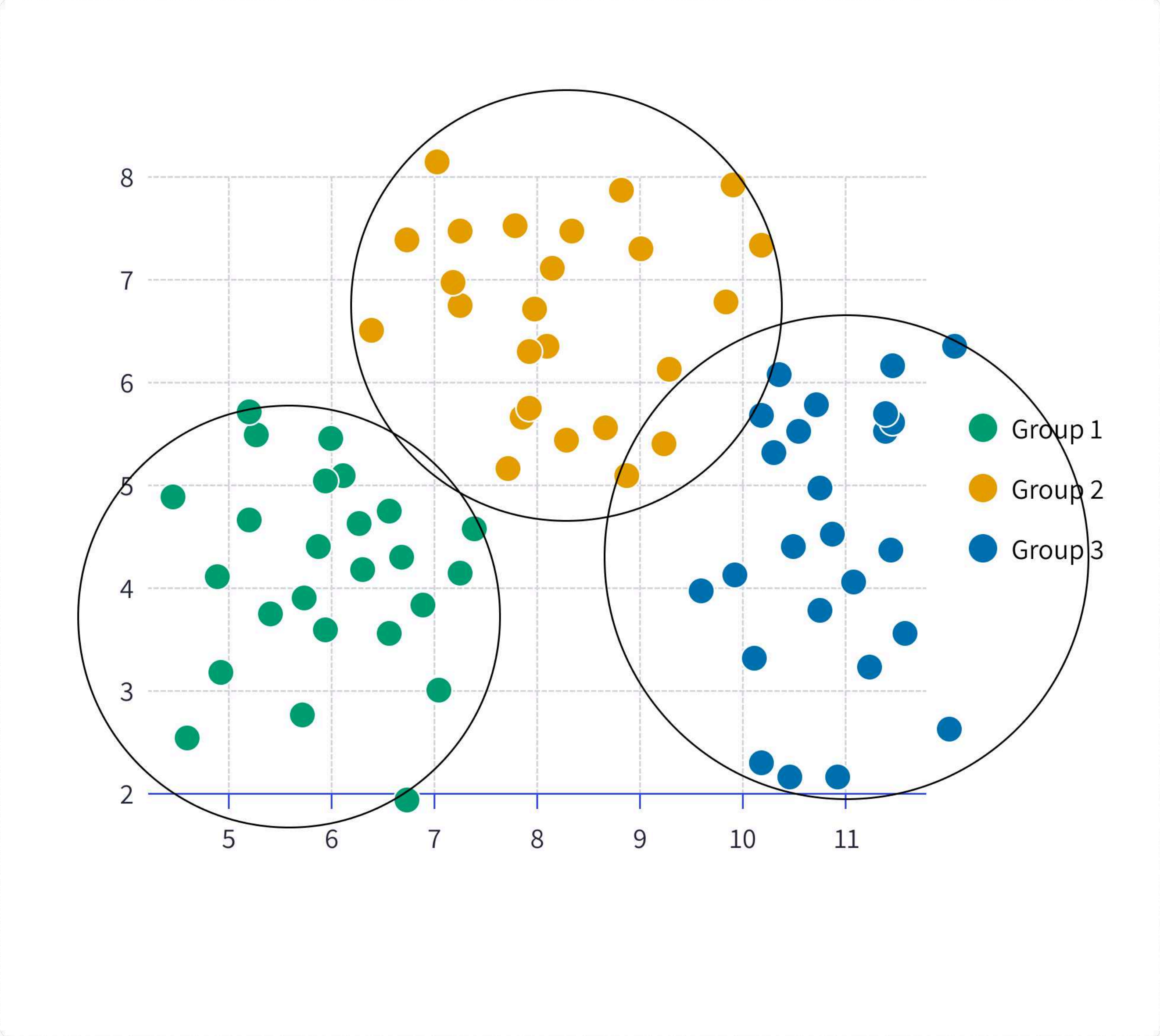
Method: IsCiL

2. Prototype

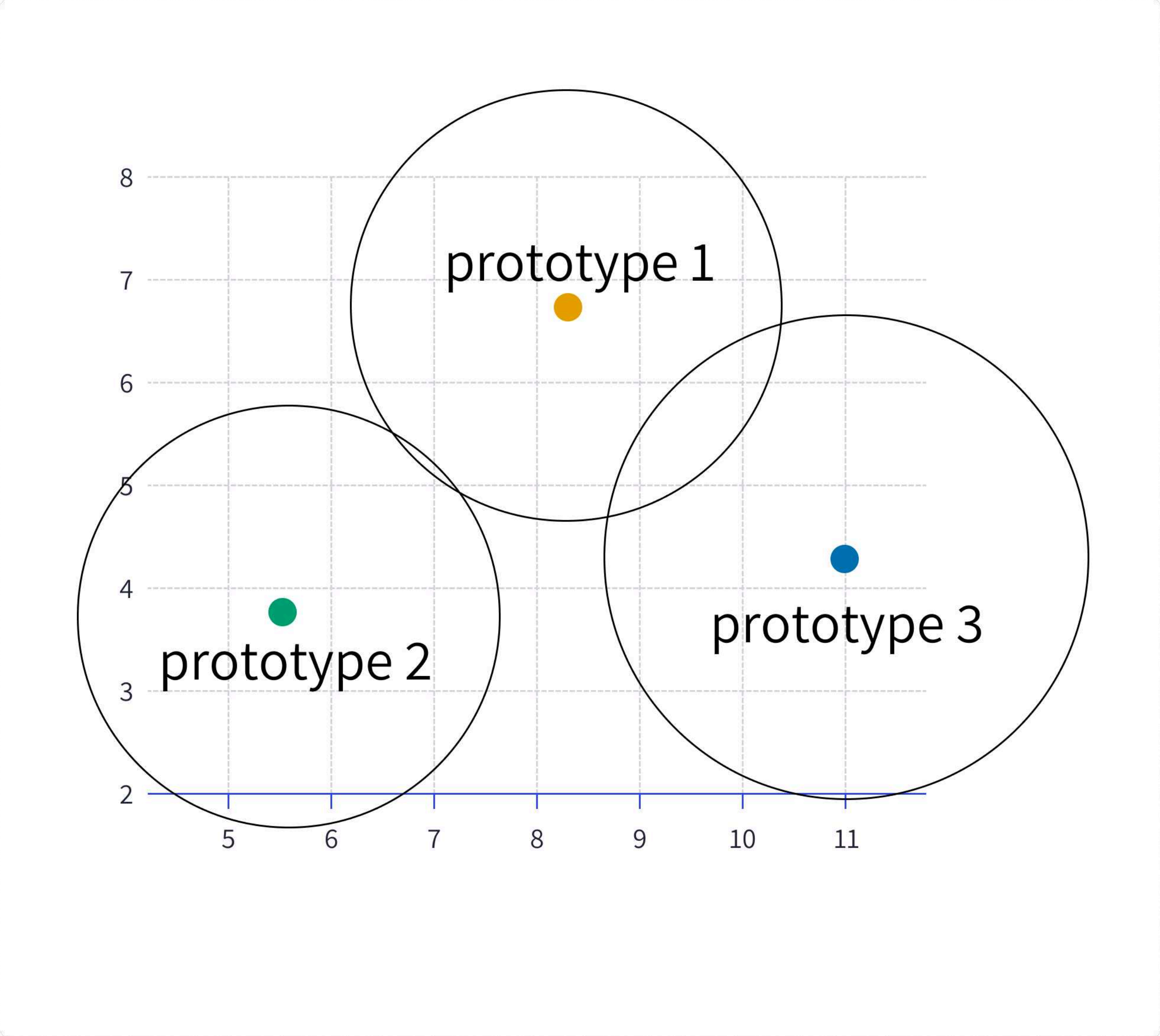
IsCiL은 이러한 **skill들을 더 일반화된 형태로 만들기 위해** 클러스터링을 진행하여
각 클러스터의 centroid를 **Protype(특정 유형의 skill들을 대표하는 skill)**로 지정한다.



1. skill space에 흩어져있는 skill들



2. KNN을 통해 skill을 클러스터링

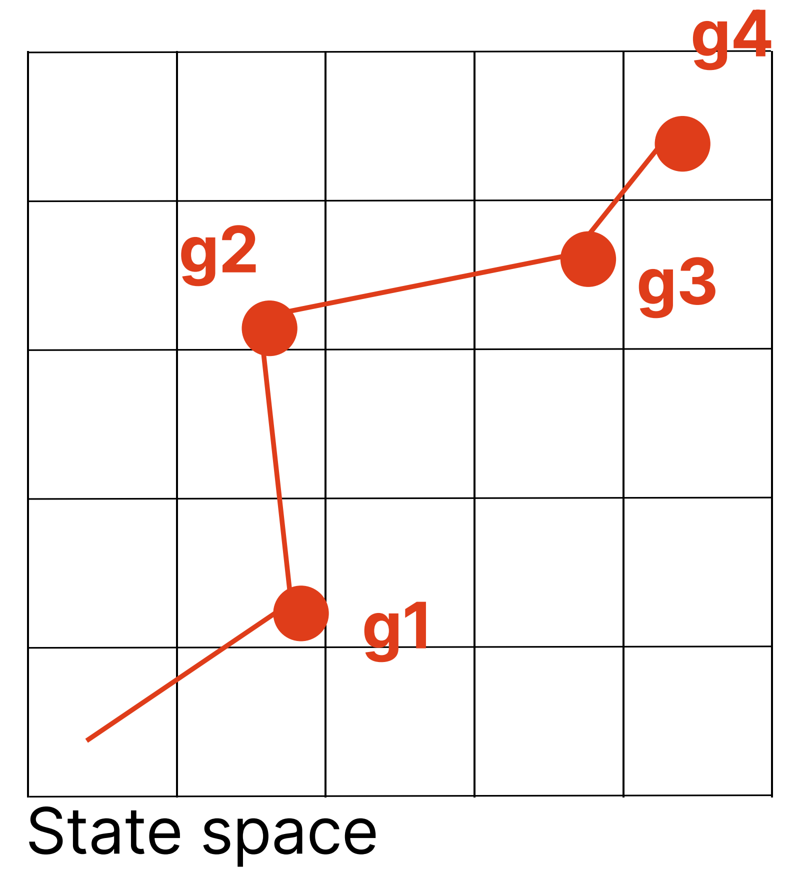


3. 각 클러스터의 중심을 prototype으로 지정

Method: IsCiL

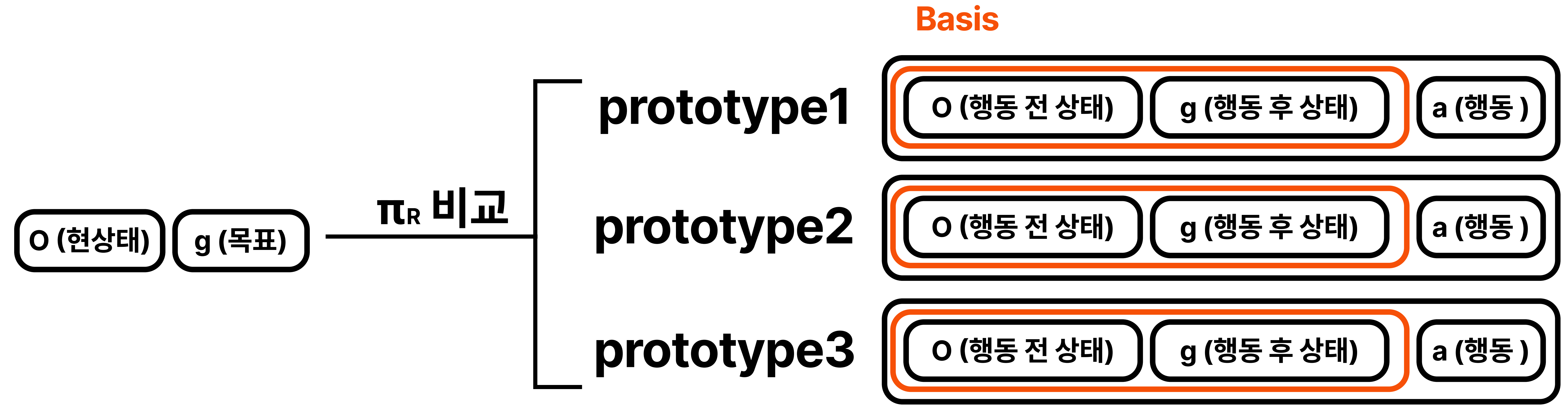
3. Sub Goal List & Skill Retriever

본 연구에서 해결과제로 설정한 **데이터셋**은 **해결과제(Task: τ)**들로 이루어져있고, **해결과제(Task: τ)**는 다시 **소목표(sub-Goal: g / state space위의 한 점)**들로 이루어져있다.

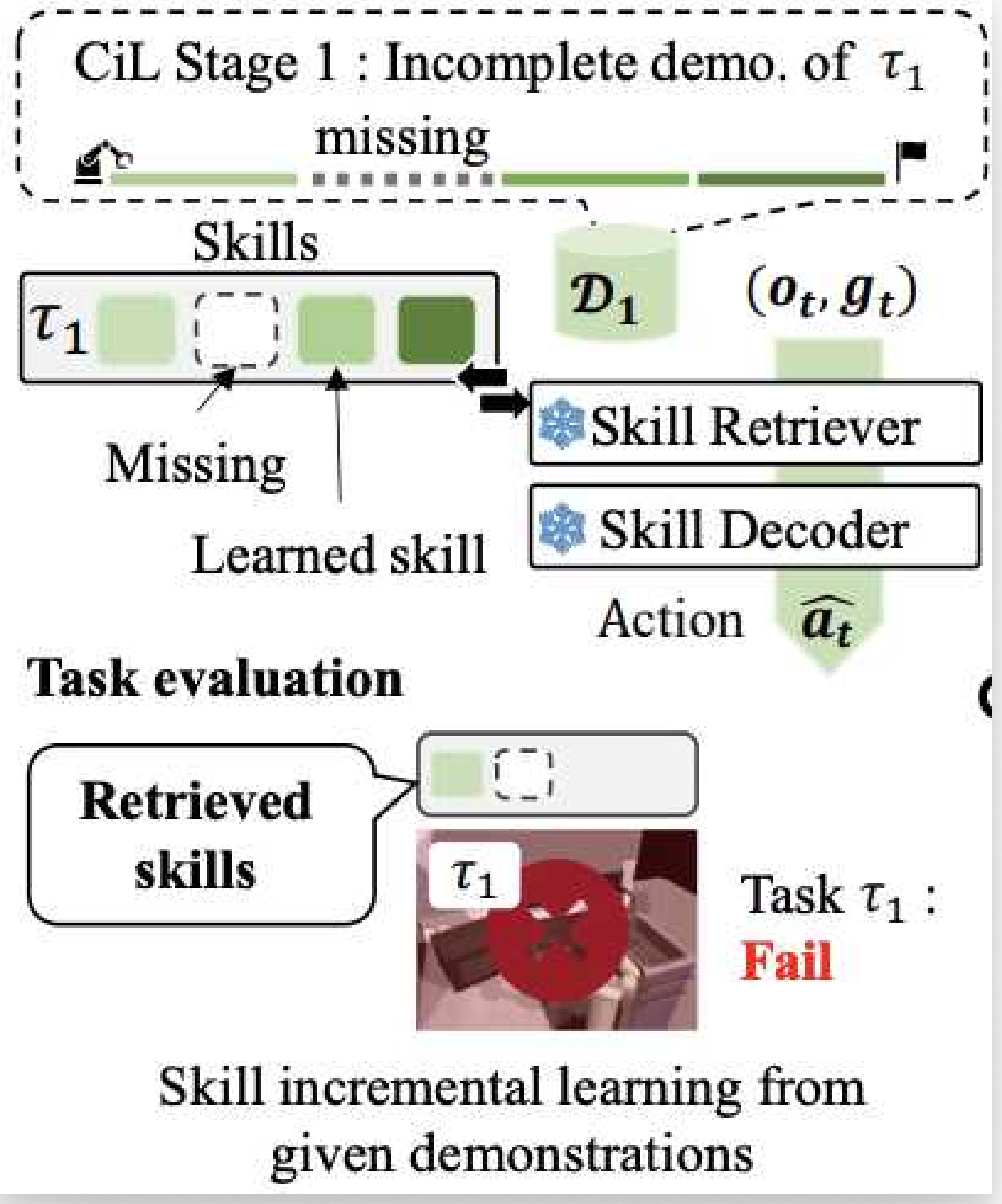


τ1 = {g1, g2, g3, g4...}

Skill Retriever(π_R)는 **(o(현상태), g(목표))**와 **Prototype들의 Basis**를 비교하여, **유사도가 가장 높은 Prototype을 가져온다**. 이 과정을 반복하여 Skill Retriever는 해결과제 **τ = {g1, g2, g3, g4} 를 해결할 수 있는 prototype의 배열 {p1, p2, p3, p4}**를 완성한다.

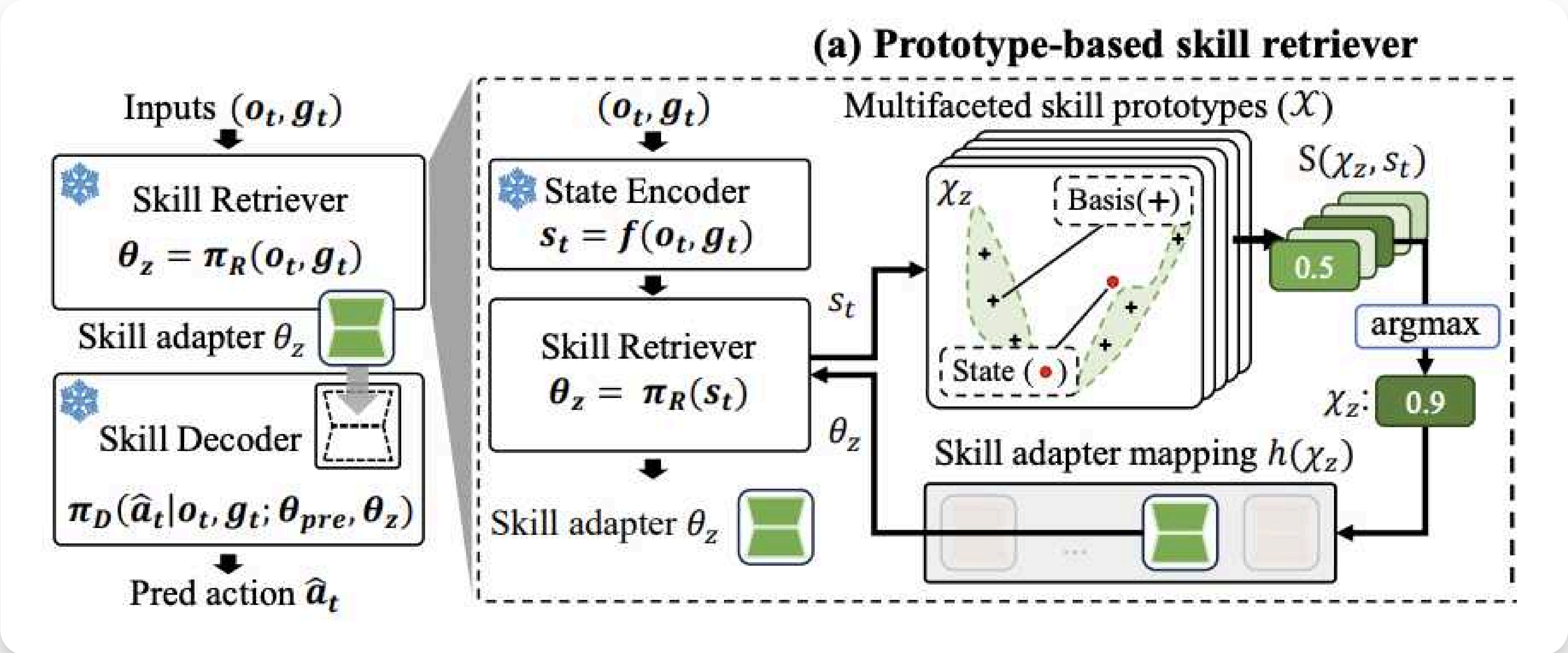


$$\pi_R \rightarrow \text{argmax}_{\text{prototype}}\{\text{similarity}((o,g), \text{Basis})\}$$



Method: IsCiL

4. Skill adapter & Base Model



Skill Retriever (π_R)를 통해 선택된 Prototype은 가중치 변환함수 h를 통해 **Skill adapter의 가중치 θ_z로 변환**된다.

$$\theta_z = \pi_R(s_t; \mathcal{X}) = h\left(\text{argmax}_{\chi_z \in \mathcal{X}} S(\chi_z, s_t)\right) \quad \text{where} \;\; S(\chi_z, s_t) = \max_{b \in \chi_z} \text{sim}(b, s_t) \tag{2}$$

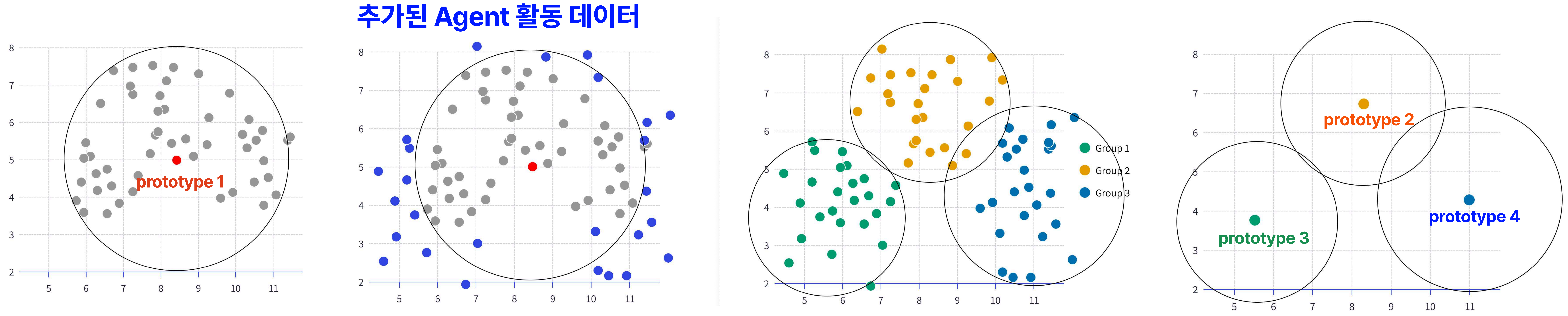
이후 action은 **Skill adapter (θ_z)**와 사전학습된 **Basemodel (θ_{pre})** 를 통해 **Skill Decoder(π_D)**가 결정한다.

$$\pi_D(\hat{a}_t|o_t, g_t; \theta_{pre}, \theta_z)$$

Method: IsCiL

5. Skill Incremental Learning

IsCiL의 장점은 **Agent의 활동 데이터 누적**을 통해 **Skill Prototype을 계속해서 얻을 수 있다**는 것이다.



1. 기존의 큰 클러스터

2. Agent의 활동 데이터가 클러스터 내외로 누적

3. 큰 클러스터를 KNN을 통해 작은 클러스터들로 분할

4. 새로운 프로토타입 3개 생성

prototype2	prototype2의 basis	a (prototype1의 행동)
prototype3	prototype3의 basis	a (prototype1의 행동)
prototype4	prototype4의 basis	a (prototype1의 행동)

이때 새로 생성된 prototype들의 action 은 **기존 prototype1의 action과 동일하게 초기화**한다. 이후 **지도학습**을 진행하여 최적화한다.

$$\mathcal{L}(o_t, g_t, a_t; \theta_z) = \|a - \pi_D(\hat{a}_t \mid o_t, g_t; \theta_{\text{pre}}, \theta_z)\|, \text{ where } \theta_z = \pi_R(o_t, g_t) \quad (3)$$

Method: IsCiL

6. Skill Incremental Learning / Task-wise Selective adaptation

IsCiL은 Incremental Learning과 π_R, π_D 의 최적화를 통해
이전에 실패한 Task도 성공할 수 있고, **처음 본 task도 성공**할 수 있다.

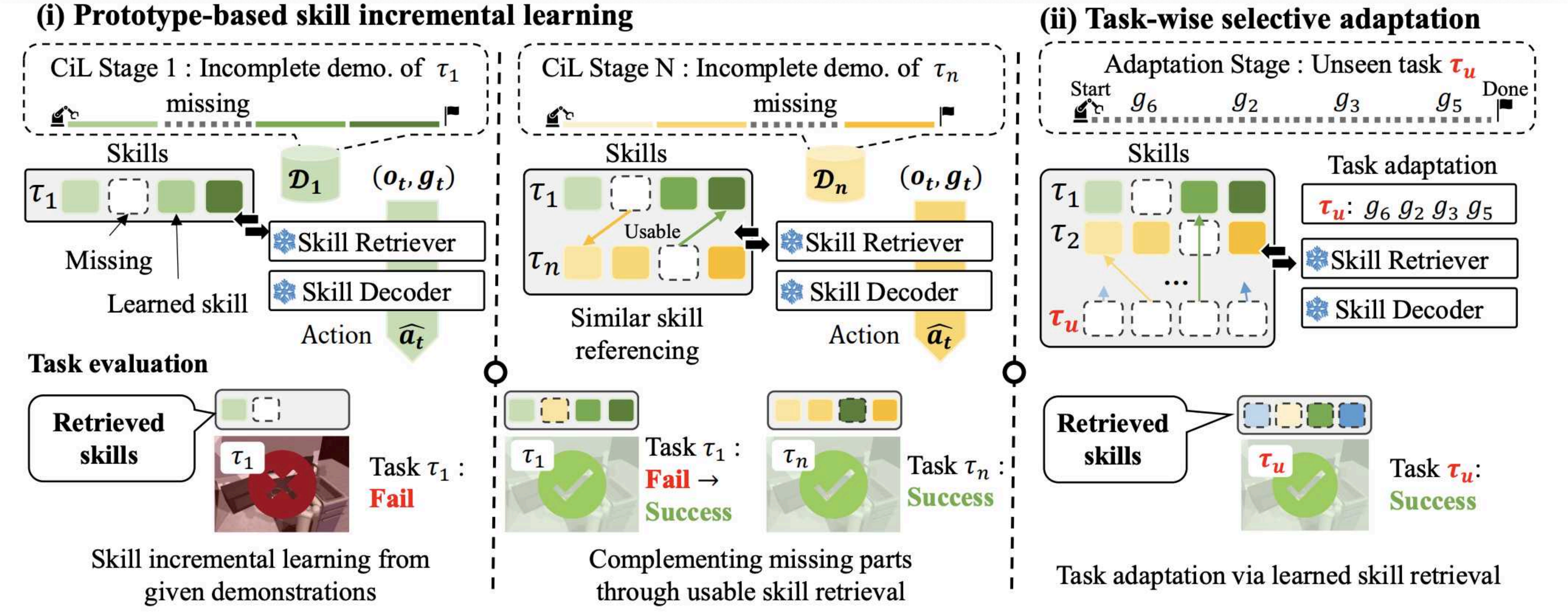
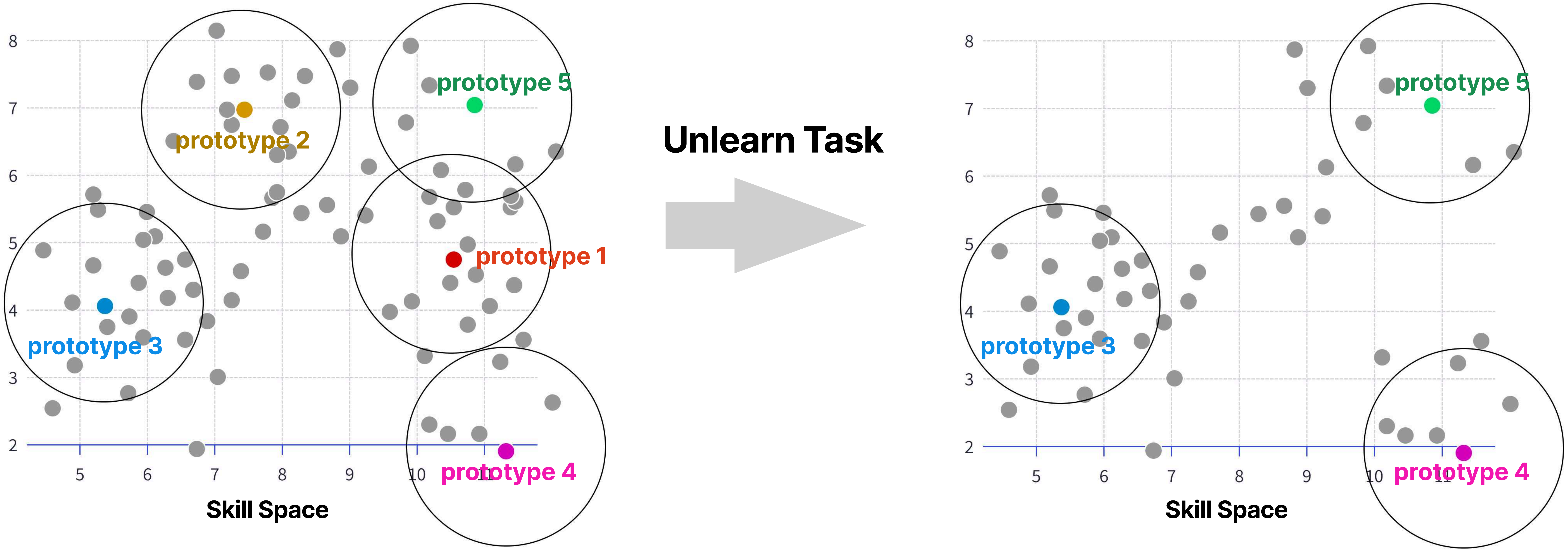


Figure 1: The scenario demonstrating how IsCiL enhances continual imitation learning efficiency through retrievable skills: (i) Prototype-based skill incremental learning: despite the failure of τ_1, skills are incrementally learned from the available demonstrations. In later stages, missing skills for τ_1 are retrieved from other tasks, achieving the resolution of τ_1 and illustrating the reversibility and efficiency of retrievable skills. (ii) Task-wise selective adaptation: IsCiL effectively retrieves relevant learned skills, facilitating rapid task adaptation.

Method: IsCiL

7. Task Unlearning

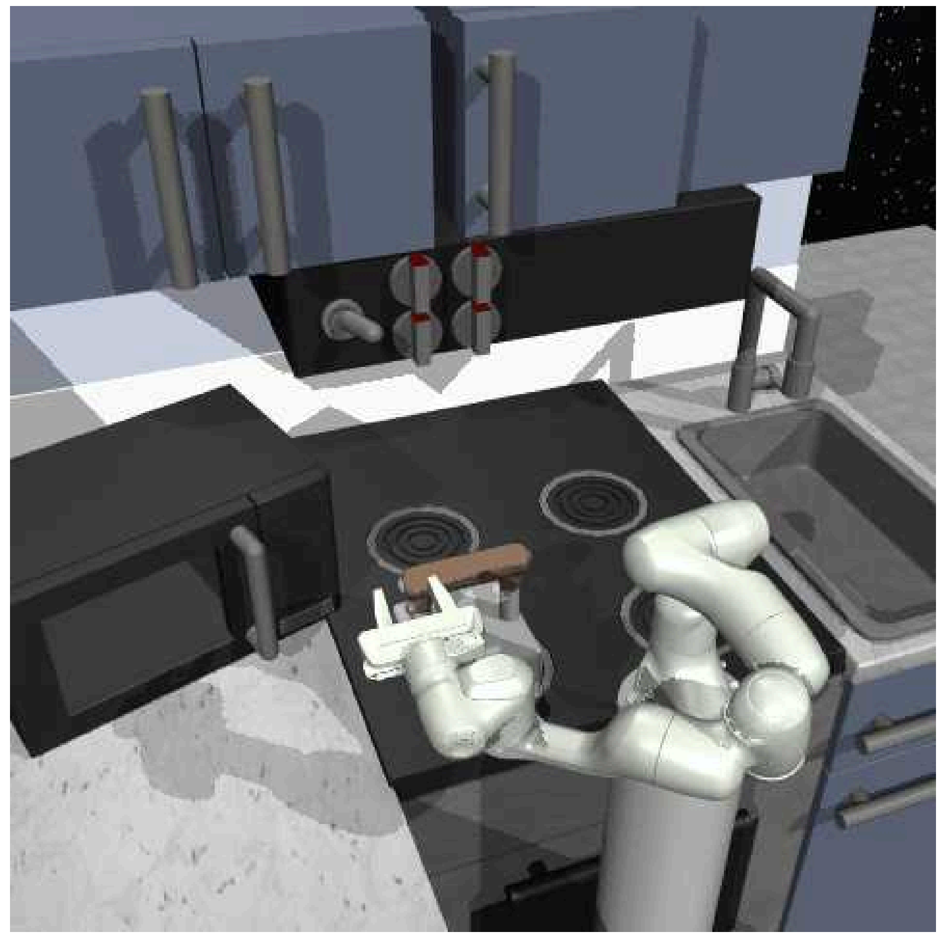
IsCiL의 **Task Unlearning** 과정은 매우 단순하다.
Unlearn하고 싶은 Task가 τ라고 할때, τ = {g1, g2}를 해결하기 위해 {p1, p2} 프로토타입이 각각 사용되었다면, **두 프로토타입을 skill space에서 제거**함과 동시에 **두 프로토타입과 같은 클러스터에 속한 skill들을 제거**하면 된다.



Experimental Settings: Benchmark

Evolving Kitchen

- Franka-Kitchen 환경에서 수행되는 데이터 스트림[45]
- 장기적 목표 수행(long-horizon tasks)을 포함
- 각 작업은 7개의 서브골 중 4개 서브골을 순차적으로 달성해야함
- **환경:** 로봇 팔 (7-DOF) 기반의 시뮬레이션 환경
- **관측(o):** 60차원 벡터



Evolving World[6]

- Meta-World 환경을 기반으로 한 데이터 스트림
- Evolving Kitchen과 유사한 장기 목표 수행(long-horizon tasks)을 요구
- 각 작업은 8개의 서브골 중 4개 서브골을 순차적으로 달성해야함
- **환경:** 로봇 팔 (7-DOF) 기반의 시뮬레이션 환경
- **관측(o):** 60차원 벡터

Experimental Settings: CiL Scenario

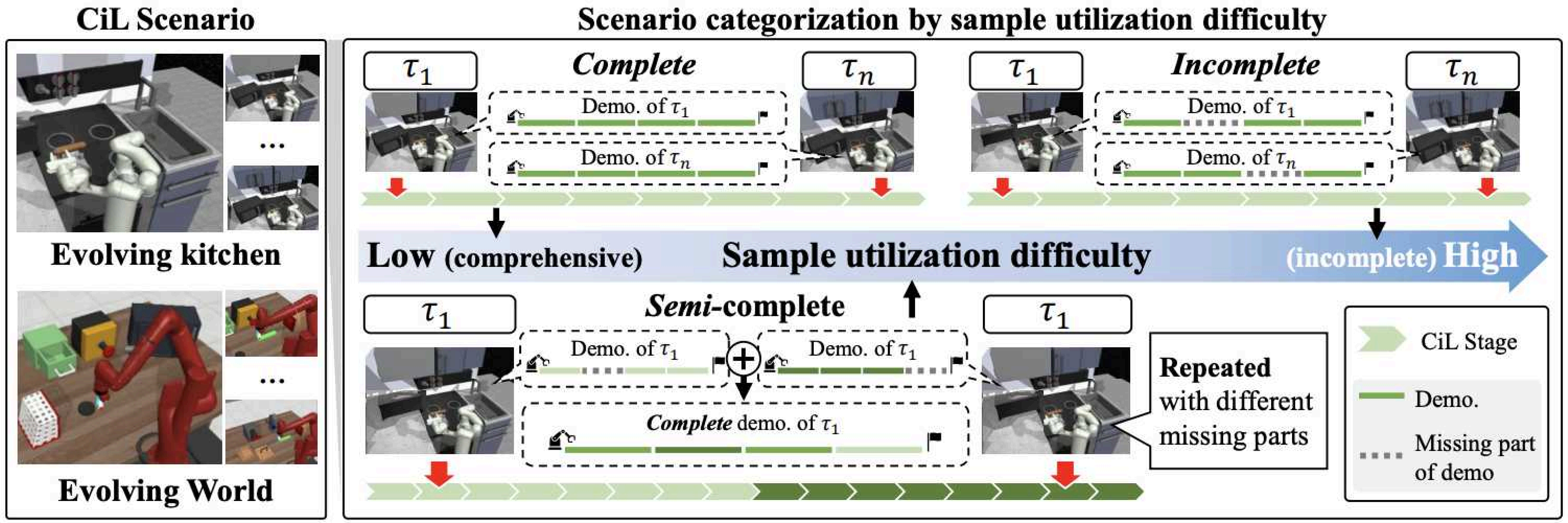


Figure 3: CiL scenarios including *Complete*, *Semi-Complete*, and *Incomplete*, categorized by sample utilization difficulty, based on the completeness of the demonstration for task performance: In *Complete*, each of the 20 CiL stages incrementally introduces new tasks featuring objects not encountered in the pre-training stage, along with full, comprehensive demonstrations for each task. In *Semi-Complete*, the first 10 stages are repeated twice, with tasks presented alongside incomplete demonstrations, where specific sub-goals are missing from the trajectories. In *Incomplete*, the same sequence of tasks from the *Complete* scenario is used, but all stages feature incomplete demonstrations, requiring the system to handle tasks with missing sub-goal trajectories.

모든 샘플 활용 가능 / 일부 활용 불가능 / 극히 제한된 활용

Experimental Settings: Metrics

1. FWT (Forward Transfer)

모델이 이전에 학습한 데이터 없는 상태에서 초기학습 능력을 평가하는 지표

- 0.3~0.5: 중간 정도의 초기학습 능력
- 0.6 이상: 우수한 초기학습 능력

2. BWT (Backward Transfer)

새로운 작업을 학습한 후, 이전 작업의 성능이 얼마나 유지되었는지를 평가하는 지표

- 0 : 새로운 작업 학습이 이전 학습에 영향을 주지 않음
- 양수: 새로운 작업 학습이 과거 작업에 긍정적 영향을 미침 (이전 성능 향상)
- 음수: 새로운 작업 학습이 과거 작업 성능을 저하시킴 (Catastrophic Forgetting 발생)
- 0 ~ 0.1이 평균적인 수준이며, 0.1 이상이면 뛰어난 유지 성능을 의미

3. AUC (Area Under the Curve)

학습 시작부터 종료 시점까지의 누적 성능을 평가하는 종합적인 지표

- 0.5: 평균적인 성능
- 0.7 이상: 매우 우수한 성능 (일관된 성능 유지)
- 0.4 이상: 학습 성능이 불안정하거나, Catastrophic Forgetting이 심각한 경우

Experimental Settings: Baseline

1. **Seq-FT[7]**
   순차적 학습 방식
2. **Seq-LoRA[7]**
   Seq-FT의 변형 버전, 사전 학습된 모델의 지식을 효과적으로 활용
   64-rank adapter 사용
3. **EWC (Elastic Weight Consolidation)[8]**
   Fisher 정보 행렬을 이용해 가중치를 정규화 → 과거 학습 내용을 보존
   하이퍼파라미터 α=10 설정
4. **L2M (Learning to Memorize)[9]**
   입력을 쿼리로 변환하여 키(key) 검색을 통해 적절한 어댑터를 선택
   L2M-g에선 상태 벡터와 서브골 정보를 결합한 값을 키로 지정
   4-rank LoRA adapter 100개를 사용
5. **TAIL (Task-Adaptive Incremental Learning)[10]**
   작업별(Task-specific) 어댑터를 추가해 유연하게 학습하는 방식
   TAIL-g: 4-rank adapter / TAIL-τ는 16-rank adapter
6. **Multi-task[11]**
   모든 작업을 동시에 학습하는 멀티태스킹 방식
   이전 데이터와 새로운 데이터를 1:1로 섞어서 학습 반복
7. **ER (Experience Replay)[12]**
   과거 데이터를 선택적으로 저장하고 이를 다음 학습에 재활용
   이전 데이터와 새로운 데이터를 1:1로 섞어서 학습 반복
8. **CLPU (Continual Learning with Private Unlearning)[13]**
   특정 작업의 데이터를 요청에 따라 제거하는 개인정보보호 특화 방식
   특정 작업 삭제 요청 → 해당 데이터와 관련된 파라미터를 완전히 제거
   TAIL-τ 정보 기반 탐색(search) 방식으로 제거된 데이터 효과적 처리

Experiment Results: Overall Performance

Table 1: Overall performance on CiL scenarios of Evolving Kitchen and Evolving World: The rows represent baselines, categorized into sequential adaptation and adapter-based approaches, and oracle, respectively. The columns represent continual learning scenarios, where each scenario has 20 stages. Each scenario in the environment is categorized into *Complete*, *Semi*-complete, and *Incomplete*. The highest performance is highlighted in **bold** and the second highest performance is underlined.

Stream	Evolving Kitchen-*Complete*			Evolving Kitchen-*Semi*			Evolving Kitchen-*Incomplete*		
CiL-algorithm	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)
Pre-trained	-	-	24.3±0.5	-	-	29.1±0.9	-	-	24.3±0.5
Seq-FT	90.9±2.6	-63.7±2.7	35.0±0.7	37.1±2.1	-25.1±2.7	16.5±0.7	32.7±4.3	-19.6±3.0	15.7±0.5
EWC	34.2±0.8	-19.5±4.2	17.1±2.7	27.2±1.3	-18.0±1.3	12.2±1.4	19.3±2.3	-3.2±11.3	10.4±1.7
Seq-LoRA	77.5±2.6	-55.2±1.8	28.3±1.5	37.4±3.8	-25.5±3.2	15.9±1.6	32.9±2.5	-19.9±2.9	14.5±0.2
L2M	24.7±4.8	-2.5±4.5	22.7±1.6	19.2±4.4	0.2±1.3	19.1±4.8	17.5±4.0	-2.0±3.2	15.8±4.8
L2M-*g*	38.2±3.4	-6.5±3.7	32.3±1.4	37.9±3.7	-4.5±3.1	32.1±1.2	37.5±10.0	-6.5±6.9	31.0±8.8
TAIL-*g*	85.3±8.0	-49.9±6.7	41.5±1.7	55.0±1.5	-21.1±2.2	37.2±2.4	53.2±1.7	-20.0±2.0	<u>35.4±0.7</u>
TAIL-τ	86.2±5.6	0.0±0.0	<u>86.2±5.6</u>	41.2±2.5	0.0±0.0	<u>41.2±2.5</u>	33.8±3.0	0.0±0.0	33.8±3.0
IsCiL (ours)	79.3±1.7	11.0±1.6	**89.8±0.5**	68.1±2.2	8.6±0.6	**75.8±1.8**	61.8±0.9	13.7±2.9	**74.0±1.9**
Multi-task	93.3±1.7	-1.6±2.3	92.3±1.8	75.4±4.5	8.0±5.5	83.2±1.1	71.7±1.1	12.6±0.8	83.0±1.1

Stream	Evolving World-*Complete*			Evolving World-*Semi*			Evolving World-*Incomplete*		
CiL-algorithm	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)
Pre-trained	-	-	0.0±0.0	-	-	0.0±0.0	-	-	0.0±0.0
Seq-FT	88.9±3.1	-73.6±4.2	24.9±0.4	38.9±5.9	-27.5±5.5	13.2±0.9	41.4±2.0	-33.0±2.0	12.2±0.8
EWC	25.7±3.8	-18.0±0.2	10.5±3.5	13.9±1.4	-9.1±1.8	6.2±1.8	18.2±2.8	-11.6±2.1	8.5±0.9
Seq-LoRA	85.6±2.9	-75.1±2.3	21.4±1.2	32.2±5.2	-18.2±4.9	16.0±2.3	38.1±1.6	-30.6±0.9	11.7±0.9
L2M	72.1±5.3	-6.6±2.1	65.9±3.3	41.0±2.1	6.3±3.0	47.0±0.7	26.1±1.1	5.7±2.8	31.4±2.0
L2M-*g*	64.2±3.9	-19.3±4.4	48.6±2.0	44.5±2.0	3.4±2.5	<u>48.2±0.2</u>	33.2±2.0	-0.6±0.9	33.1±2.2
TAIL-*g*	90.0±3.0	-56.8±0.4	39.5±2.9	43.2±7.8	-17.6±3.5	27.4±5.1	51.4±2.5	-21.4±0.6	32.5±2.3
TAIL-τ	85.7±5.9	0.0±0.0	**85.7±5.9**	27.5±0.7	0.0±0.0	27.5±0.7	39.7±1.0	0.0±0.0	<u>39.7±1.0</u>
IsCiL (ours)	81.7±0.4	2.7±0.9	<u>84.3±1.1</u>	60.0±1.1	9.3±1.4	**68.9±0.5**	63.2±1.5	8.7±2.7	**71.2±4.2**
Multi-task	88.6±3.6	2.8±3.5	90.7±1.2	55.0±3.6	27.6±4.1	80.9±0.3	73.2±1.7	12.6±1.2	84.2±1.3

- 모든 시나리오에서 높은 AUC (누적 성능) 유지
- Overwriting 문제 완화로 안정적 성능
- 샘플 효율성 우수
- 특정 시나리오에서의 최고 성능은 일부 베이스라인에 밀림

Experiment Results: Task Adaptation

Table 2: Task adaptation performance with unseen tasks: This is based on the existing Evolving World-*Complete* and Evolving Kitchen-*Complete*. In Evolving World, four novel tasks are introduced every four stages, while in Evolving Kitchen, two novel tasks are introduced every five stages. Metrics with the suffix -A denote performance based solely on adaptation tasks, while other metrics report performance across all tasks.

Stream	Evolving Kitchen-*Complete* Unseen					Evolving World-*Complete* Unseen				
Algorithm	FWT (%)	BWT (%)	AUC (%)	FWT-A (%)	AUC-A (%)	FWT (%)	BWT (%)	AUC (%)	FWT-A (%)	AUC-A (%)
Seq-FT	72.3$\pm$1.6	-47.7$\pm$1.6	30.4$\pm$0.2	27.8$\pm$0.6	19.5$\pm$0.1	52.9$\pm$3.6	-26.7$\pm$1.8	30.1$\pm$2.1	16.3$\pm$1.8	24.0$\pm$2.6
EWC	21.0$\pm$15.9	-14.0$\pm$2.0	16.8$\pm$1.6	18.1$\pm$4.2	14.4$\pm$1.6	16.5$\pm$1.9	-8.1$\pm$0.8	9.6$\pm$2.6	6.1$\pm$1.3	8.3$\pm$2.1
Seq-LoRA	62.4$\pm$3.8	-41.5$\pm$3.3	25.4$\pm$0.9	28.1$\pm$0.0	18.2$\pm$0.0	45.2$\pm$0.4	-35.8$\pm$1.3	14.5$\pm$0.9	6.4$\pm$2.5	8.2$\pm$1.8
L2M	22.3$\pm$2.3	0.3$\pm$1.5	22.7$\pm$3.5	15.3$\pm$3.2	21.2$\pm$4.1	55.1$\pm$3.7	-1.4$\pm$3.3	53.6$\pm$1.0	40.3$\pm$2.4	41.2$\pm$2.0
L2M-*g*	33.8$\pm$0.9	-4.3$\pm$1.2	30.0$\pm$0.4	22.2$\pm$0.6	24.1$\pm$0.7	43.3$\pm$1.6	-8.2$\pm$3.6	35.7$\pm$1.6	24.2$\pm$1.5	25.7$\pm$1.7
TAIL-*g*	67.6$\pm$7.4	-34.9$\pm$5.4	36.8$\pm$3.2	34.7$\pm$2.2	30.1$\pm$1.0	53.2$\pm$1.4	-27.1$\pm$1.2	29.2$\pm$0.3	18.6$\pm$0.9	19.1$\pm$0.6
IsCiL (ours)	69.5$\pm$2.5	16.3$\pm$2.2	**84.4**$\pm$1.3	**52.1**$\pm$7.5	**72.8**$\pm$2.1	64.3$\pm$2.6	-0.5$\pm$3.5	**63.9**$\pm$0.6	**45.8**$\pm$4.7	**45.3**$\pm$0.9
Multi-task	85.3$\pm$1.7	3.7$\pm$1.8	88.8$\pm$0.0	70.8$\pm$0.0	79.0$\pm$0.1	85.4$\pm$0.9	5.6$\pm$0.5	90.4$\pm$0.5	78.3$\pm$2.9	85.9$\pm$0.4

새로운 평가 지표

- **FWT-A:** 이전에 학습한 작업의 지식만으로 신규 작업을 수행하는 초기 성능을 의미
- **AUC-A:** 신규 작업에 대해서만 성능 곡선을 계산한 것

- IsCiL은 두 가지 시나리오 모두에서 뛰어난 Task Adaptation 성능을 보임
- Evolving Kitchen에서 **최고 수준의 FWT-A를 달성** : FWT-A: 52.1
- 동시에 Evolving Kitchen에서 **최고 수준의 성능 향상 (AUC-A) 달성** : AUC-A 72.8
- **신규 작업 적응 & 전체 학습 곡선에서 높은 AUC-A 성능을 기록**하며 지속적인 학습 성능 입증

Experiment Results: Task Unlearning

Table 3: Overall performance with task unlearning as task adaptation: Additional stages for unlearning tasks that were learned during other stages are included for tests.

Stream	Evolving Kitchen-*Complete* Unlearning			Evolving Kitchen-*Incomplete* Unlearning		
Algorithm	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)
TAIL-τ CLPU	86.2±5.6	0.0±0.0	86.2±5.6	33.8±3.0	0.0±0.0	33.8±3.0
IsCiL (ours)	75.0±7.2	11.2±5.5	85.2±1.8	61.4±2.9	12.4±2.9	72.7±2.9

- 실험 설명: 특정 작업 능력을 제거 한 후 적응 능력 평가
- 각 지표의 의미
    a. FWT: task 제거 후 신규 작업에 빠르게 적응 능력
    b. BWT: task 제거의 영향으로 인한 성능 저하를 평가
    c. AUC: task 제거 후의 장기적 안정성을 평가

- **IsCiL은 Incomplete 시나리오에서 TAIL-τ CLPU보다 115% 더 높은 AUC를 기록**
- **Unlearning 이후에도 IsCiL의 성능 저하는 최소화됨 (1.8% ~ 5.2%)**

Experiment Results: Sample Efficiency

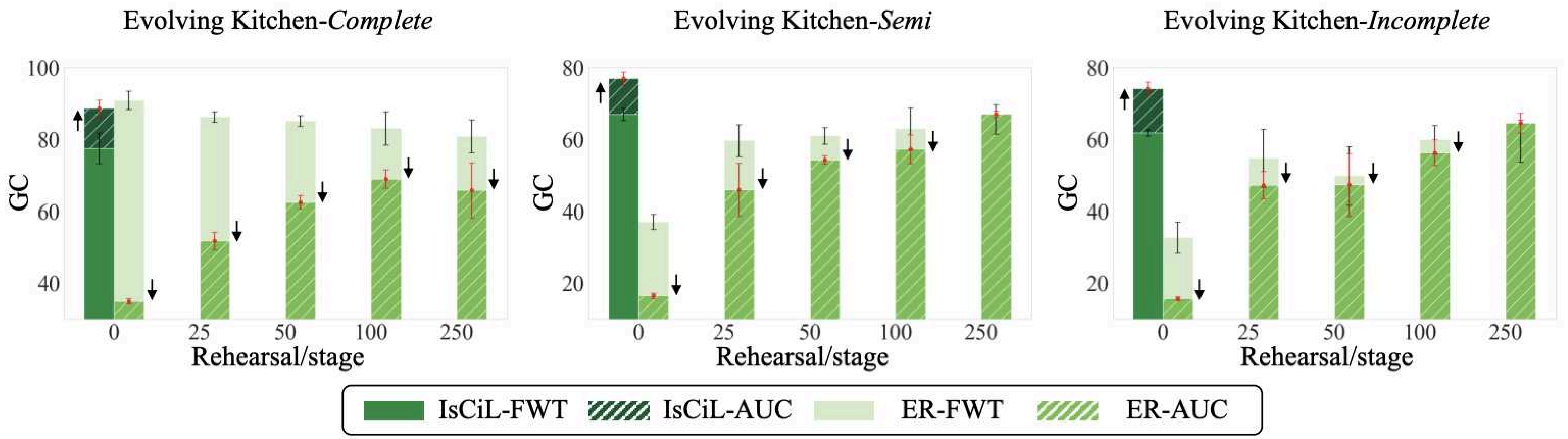


Figure 4: Comparison w.r.t. the number of rehearsals: The horizontal axis represents the amount of stored rehearsal data at each stage, while the vertical axis indicates goal-conditioned success rates (GC).

- IsCiL은 **모든 환경에서 최고 AUC를 기록**하며, **AUC가 FWT를 초과한 유일한 방법**
- ER은 저장하는 샘플 수가 많아질수록 FWT 성능이 감소하→리허설이 많을수록 샘플 효율성이 감소 의미
- ER의 FWT가 IsCiL에 근접하지만, 동시에 AUC는 거의 개선되지 않음

Experiment Results: Training resource

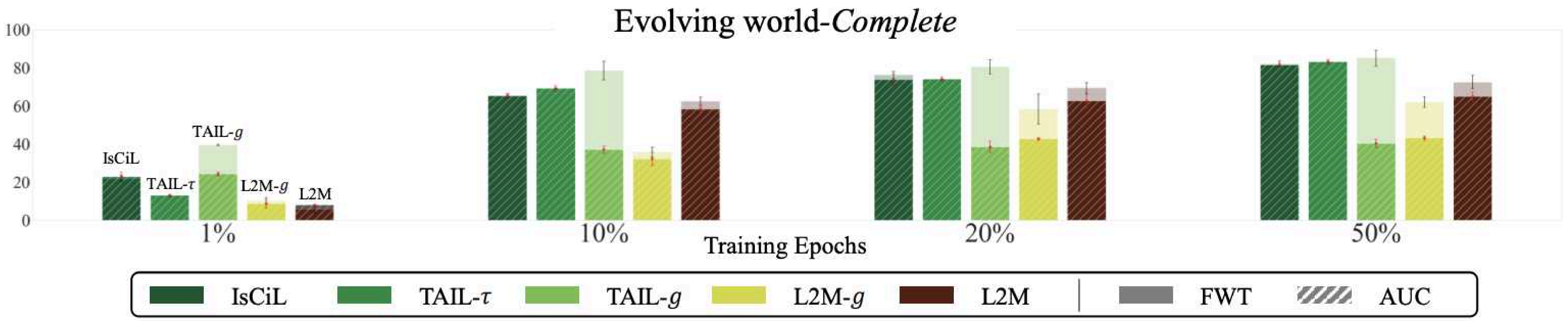


Figure 5: Comparison w.r.t. training resources: In all baselines, the plain bar graph represents FWT, while the bar graph with hatch marks represents AUC. The vertical axis indicates goal-conditioned success rates (GC).

- IsCiL, TAIL-τ 은 제한된 학습 자원에서도 안정적 성능 유지
- TAIL-g → 초기 적응 능력(FWT)은 높지만, 지식 유지 실패로 AUC 저하

Ablation Study

Table 4: Ablation on IsCiL skill prototype.

Stream	Evolving kitchen-*Complete*		
Ablations	FWT (%)	BWT (%)	AUC (%)
IsCiL $g, \|\chi_z\| = 20$	79.3±1.7	11.0±1.6	89.8±0.5
IsCiL $g, \|\chi_z\| = 1$	28.9±10.2	2.3±5.7	30.6±12.2
IsCiL $g, \|\chi_z\| = 5$	63.1±4.0	2.7±7.3	66.5±7.9
IsCiL $g, \|\chi_z\| = 10$	76.4±6.5	8.2±4.0	83.9±2.7
IsCiL $g, \|\chi_z\| = 25$	77.1±1.8	11.9±1.7	88.2±1.1
IsCiL $g, \|\chi_z\| = 50$	81.5±2.8	7.9±4.9	89.4±1.2
IsCiL $\tau, \|\chi_z\| = 20$	57.8±16.6	10.9±1.8	67.2±17.2
IsCiL $\tau, \|\chi_z\| = 80$	84.3±6.7	5.0±7.5	89.5±8.3

IsCiL g (일반 설정)

- 베이스 수가 적을수록 FWT와 AUC 성능이 크게 저하
- 베이스 수가 약 10~20개에서 최고 성능을 기록
- 베이스 수 20개 → 최고 AUC (89.8%) 및 양호한 FWT 유지

IsCiL τ (Skill Trajectory 기반 설정)

- τ 설정에서는 스킬 트래젝토리(skill trajectory) 기반으로 새로운 스킬을 구성하므로 더 많은 베이스가 필요
- 베이스 수 증가 시 AUC 89.5%로 안정적 성능을 기록
- 상대적으로 적은 베이스에서는 AUC가 67.2%로 성능 하락

Limitation

Future Works

Reference

1. Byeonghwi Kim, Minhyuk Seo, and Jonghyun Choi. Online continual learning for interactive instruction following agents. In The Twelfth International Conference on Learning Representations (ICLR), 2024.
2. Suneel Belkhale, Yuchen Cui, and Dorsa Sadigh. Data quality in imitation learning. In Advances in Neural Information Processing Systems (NeurIPS), 2023.
3. Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. In Conference on Lifelong Learning Agents (CoLLAs), 2022.
4. Mohit Shrivastava, Alex Irpan, and Sergey Levine. Franka-Kitchen environment for robotic learning. In Proceedings of the Conference on Robot Learning (CoRL), 2020.
5. Sergey Levine, Pieter Abbeel, and Chelsea Finn. Long-horizon tasks in robotic learning. In Advances in Neural Information Processing Systems (NeurIPS), 2021.
6. Tianhe Yu, Deirdre Quillen, and Chelsea Finn. Meta-World: A benchmark for multi-task and meta-reinforcement learning. In Conference on Robot Learning (CoRL), 2020.
7. J. Kirkpatrick, R. Pascanu, N. Rabinowitz, et al., Overcoming catastrophic forgetting in neural networks, Proceedings of the National Academy of Sciences, 2017.
8. C. Finn, K. Xu, and S. Levine, Probabilistic model-agnostic meta-learning, Advances in Neural Information Processing Systems (NeurIPS), 2018.
9. M. Ren, M. Iuzzolino, and R. S. Zemel, Learning to Memorize: Continual Learning with Memory, Conference on Neural Information Processing Systems (NeurIPS), 2019.
10. Y. Liu, B. Schiele, and Q. Sun, Task-Adaptive Incremental Learning: Reducing Catastrophic Forgetting with Dynamic Architectures, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.
11. Y. Netzer, T. Wang, A. Coates, et al., Reading Digits in Natural Images with Unsupervised Feature Learning, Advances in Neural Information Processing Systems (NeurIPS), 2011.
12. M. Ranzato, A. Szlam, J. Bruna, M. Mathieu, R. Fergus, and Y. LeCun, Experience Replay for Continual Learning, International Conference on Machine Learning (ICML), 2018.
13. B. Liu, Q. Liu, and P. Stone, Continual Learning with Private Unlearning, Conference on Lifelong Learning Agents (CoLLAs), 2022.

Incremental Learning of Retrievable Skills For Efficient Continual Task Adaptation

Daehee Lee, Minjong Yoo, Woo Kyung Kim, Wonje Choi, Honguk Woo
Sungkyunkwan University

Presented as a Poster at NeurIPS 2024 (Accepted, 6/6/6/5)

https://openreview.net/forum?id=RcPAJAnpnm

2025.03.09
TaeYoon Kwack
njj05043@g.skku.edu

Appendix

Task Configuration

Table 5: Evolving Kitchen-*Complete & Incomplete* data stream task configuration

Evolving Kitchen-*Complete & Incomplete*				
Task	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
τ_1	microwave	kettle	top burner	light switch
τ_2	kettle	bottom burner	top burner	slide cabinet
τ_3	microwave	bottom burner	top burner	slide cabinet
τ_4	kettle	bottom burner	light switch	slide cabinet
τ_5	microwave	kettle	light switch	slide cabinet
τ_6	kettle	bottom burner	top burner	hinge cabinet
τ_7	microwave	kettle	top burner	hinge cabinet
τ_8	microwave	kettle	slide cabinet	hinge cabinet
τ_9	kettle	light switch	slide cabinet	hinge cabinet
τ_{10}	microwave	kettle	bottom burner	hinge cabinet
τ_{11}	kettle	bottom burner	slide cabinet	hinge cabinet
τ_{12}	kettle	bottom burner	light switch	hinge cabinet
τ_{13}	microwave	top burner	light switch	hinge cabinet
τ_{14}	microwave	kettle	bottom burner	slide cabinet
τ_{15}	microwave	kettle	light switch	hinge cabinet
τ_{16}	microwave	bottom burner	top burner	light switch
τ_{17}	kettle	top burner	light switch	slide cabinet
τ_{18}	microwave	bottom burner	top burner	hinge cabinet
τ_{19}	microwave	bottom burner	slide cabinet	hinge cabinet
τ_{20}	microwave	bottom burner	light switch	slide cabinet

Table 6: Evolving Kitchen-*Semi* data stream task configuration

Evolving Kitchen-*Semi*				
Task	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
τ_1	microwave	kettle	top burner	light switch
τ_2	kettle	bottom burner	top burner	slide cabinet
τ_3	microwave	bottom burner	top burner	slide cabinet
τ_4	kettle	bottom burner	light switch	slide cabinet
τ_5	microwave	kettle	light switch	slide cabinet
τ_6	kettle	bottom burner	top burner	hinge cabinet
τ_7	microwave	kettle	top burner	hinge cabinet
τ_8	microwave	kettle	slide cabinet	hinge cabinet
τ_9	kettle	light switch	slide cabinet	hinge cabinet
τ_{10}	microwave	kettle	bottom burner	hinge cabinet
τ_{11}	microwave	kettle	top burner	light switch
τ_{12}	kettle	bottom burner	top burner	slide cabinet
τ_{13}	microwave	bottom burner	top burner	slide cabinet
τ_{14}	kettle	bottom burner	light switch	slide cabinet
τ_{15}	microwave	kettle	light switch	slide cabinet
τ_{16}	kettle	bottom burner	top burner	hinge cabinet
τ_{17}	microwave	kettle	top burner	hinge cabinet
τ_{18}	microwave	kettle	slide cabinet	hinge cabinet
τ_{19}	kettle	light switch	slide cabinet	hinge cabinet
τ_{20}	microwave	kettle	bottom burner	hinge cabinet

Table 7: Evolving World-*Complete & Incomplete* data stream task configuration

Evolving World-*Complete & Incomplete*				
Task	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
τ_1	door	handle	button	box
τ_2	puck	drawer	stick	lever
τ_3	handle	puck	lever	door
τ_4	button	drawer	box	stick
τ_5	door	handle	box	button
τ_6	lever	stick	drawer	puck
τ_7	lever	puck	handle	door
τ_8	stick	button	drawer	box
τ_9	handle	button	box	door
τ_{10}	drawer	stick	lever	puck
τ_{11}	puck	lever	door	handle
τ_{12}	stick	button	box	drawer
τ_{13}	handle	button	door	box
τ_{14}	drawer	lever	stick	puck
τ_{15}	puck	lever	handle	door
τ_{16}	stick	box	button	drawer
τ_{17}	handle	door	box	button
τ_{18}	stick	drawer	puck	lever
τ_{19}	door	puck	lever	handle
τ_{20}	box	drawer	button	stick

Table 8: Evolving World-*Semi* data stream task configuration

Evolving World-*Semi*				
Task	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
τ_1	door	handle	button	box
τ_2	puck	drawer	stick	lever
τ_3	handle	puck	lever	door
τ_4	button	drawer	box	stick
τ_5	door	handle	box	button
τ_6	lever	stick	drawer	puck
τ_7	lever	puck	handle	door
τ_8	stick	button	drawer	box
τ_9	handle	button	box	door
τ_{10}	drawer	stick	lever	puck
τ_{11}	door	handle	button	box
τ_{12}	puck	drawer	stick	lever
τ_{13}	handle	puck	lever	door
τ_{14}	button	drawer	box	stick
τ_{15}	door	handle	box	button
τ_{16}	lever	stick	drawer	puck
τ_{17}	lever	puck	handle	door
τ_{18}	stick	button	drawer	box
τ_{19}	handle	button	box	door
τ_{20}	drawer	stick	lever	puck

Table 9: Evolving Kitchen-*Complete* Unseen task configuration

Evolving World-*Complete* Unseen				
Stage	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
5	microwave	kettle	top burner	slide cabinet
5	microwave	kettle	top burner	top burner
10	microwave	kettle	top burner	light switch
10	kettle	bottom burner	top burner	light switch
15	kettle	top burner	slide cabinet	hinge cabinet
15	microwave	top burner	slide cabinet	hinge cabinet
20	microwave	top burner	light switch	slide cabinet
20	kettle	top burner	light switch	hinge cabinet

Table 10: Evolving World-*Complete* Unseen task configuration

Evolving World-*Complete* Unseen				
Stage	**Sub-goal 1**	**Sub-goal 2**	**Sub-goal 3**	**Sub-goal 4**
4	door	handle	button	box
4	puck	drawer	stick	lever
4	handle	puck	lever	door
4	button	drawer	box	stick
8	door	handle	box	button
8	puck	lever	drawer	stick
8	handle	lever	puck	door
8	box	drawer	stick	button
12	box	handle	door	button
12	lever	drawer	stick	puck
12	handle	lever	puck	door
12	box	drawer	stick	button
16	door	handle	box	button
16	puck	drawer	stick	lever
16	handle	puck	lever	door
16	box	drawer	stick	button
20	door	handle	box	button
20	lever	drawer	stick	puck
20	door	handle	lever	puck
20	box	drawer	stick	button

Appendix

IsCiL Incremental Learning & Evaluation Algorithm

Algorithm 1 IsCiL Skill Incremental Learning

State encoding function f, Skill retriever π_R
Skill decoder π_D, Pre-trained parameter θ_{pre}
Skill adapter mapping function h
for each stage i in CiL Stages **do**
  for each sub-goal g in task dataset D_i **do**
    $D_i^g \leftarrow \{(o, g') \in D_i \mid g' = g\}$ // filter transitions related to the current sub-goal g
    $S_i^g \leftarrow \{f(o_t, g_t) \mid (o_t, g_t) \in D_i^g\}$ // encode states from the filtered dataset into state embeddings
    $\mathcal{X}^g \leftarrow \{\text{argmax}_{\chi_z \in \mathcal{X}} S(\chi_z, s_t) \mid s_t \in S_i^g\}$ // retrieve the most relevant skill prototypes for each state s_t
    $\chi_{\bar{z}} \leftarrow \text{Mode}(\mathcal{X}^g)$ // select the most frequently retrieved skill prototype from the set
    $\theta_{z^*} \leftarrow h(\chi_{\bar{z}})$ // map the selected skill prototype $\chi_{\bar{z}}$ to its skill adapter via h
    Update θ_{z^*} using Eq. (3) // update the skill adapter based on task-specific learning
    $\mathcal{X} \leftarrow \mathcal{X} \cup \chi_{z^*}$ // append the new skill prototype to the skill set for future retrieval
    Update the mapping function h to map χ_{z^*} to the updated adapter θ_{z^*} // update h with the new skill adapter
  end for
end for

Algorithm 2 IsCiL Evaluation

State encoding function f, Skill retriever π_R
Skill decoder π_D, Pre-trained parameter θ_{pre}
while not done **do**
  $s_t = f(o_t, g_t)$ // encode state
  $\theta_z = \pi_R(s_t)$ // retrieve skill
  $\hat{a}_t \sim \pi_D(o_t, g_t; \theta_{\text{pre}}, \theta_z)$ // decode the skill
end while

Appendix

Metric Specifics

- **FWT**: $\text{FWT}_\tau = \frac{1}{|I_\tau|}\sum_{i \in I_\tau} C_{\tau,i}$ where τ is task and $C_{\tau,i}$ represents the GC score of task τ at stage i. and I_τ is set of stage indices where task τ is trained in the CiL scenario.
- **BWT**: $\text{BWT}_\tau = \frac{1}{|I_\tau|}\sum_{i \in I_\tau}\left(\frac{1}{p-i-1}\sum_{j=i+1}^{p}(C_{\tau,j} - C_{\tau,i})\right)$, where p is the final stage at which task τ is available. In the case where $p = i$ BWT is 0.
- **AUC**: $\text{AUC}_\tau = \frac{1}{|I_\tau|}\sum_{i \in I_\tau}\left(\frac{1}{p-i}\sum_{j=i}^{p} C_{\tau,j}\right)$, represent the the overall performance of continual learning, internally including FWT and BWT. In the case where $p = i$, AUC_τ is FWT_τ.

Appendix

Additional Ablation Study

Table 14: Ablation study on the skill adapter rank in Evolving Kitchen-*Complete* and Evolving World-*Complete*.

Stream		Evolving Kitchen-*Complete*			Evolving World-*Complete*		
Rank	CiL-algorithm	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)
1	L2M-*g*	30.2±2.1	2.6±1.0	33.0±1.6	56.8±3.5	-16.9±5.2	41.6±1.3
	TAIL-*g*	93.2±2.5	-54.3±1.6	45.7±1.3	77.0±5.0	-47.9±1.9	34.6±3.7
	IsCiL	89.2±4.0	2.7±3.0	**91.6**±1.8	73.6±5.1	-3.3±3.9	**70.9**±3.3
4	L2M-*g*	38.2±3.4	-6.5±3.7	32.3±1.4	64.2±3.9	-19.3±4.4	48.6±2.0
	TAIL-*g*	85.3±8.0	-49.9±6.7	41.5±1.7	90.0±3.0	-56.8±0.4	39.5±2.9
	IsCiL	79.3±1.7	11.0±1.6	**89.8**±0.5	81.7±0.4	2.7±0.9	**84.3**±1.1

Table 15: Ablation study on the quality of the skill decoder pre-trained model in Evolving Kitchen-*Complete* and *Incomplete*.

Stream		Evolving Kitchen-*Complete*			Evolving Kitchen-*Incomplete*		
CiL-algorithm	Pre-training	FWT (%)	BWT (%)	AUC (%)	FWT (%)	BWT (%)	AUC (%)
TAIL-τ	1 object	72.8±7.9	0.0±0.0	72.8±7.9	28.8±0.7	0.0±0.0	28.8±0.7
	2 object	87.2±4.6	0.0±0.0	87.2±4.6	35.9±2.6	0.0±0.0	35.9±2.6
	4 object	86.2±5.6	0.0±0.0	86.2±5.6	33.8±3.0	0.0±0.0	33.8±3.0
IsCiL	1 object	60.0±4.0	2.1±4.2	62.1±0.8	42.1±7.3	5.4±3.2	47.0±4.6
	2 object	78.9±5.1	6.4±3.7	84.9±1.3	56.7±3.2	12.0±2.3	67.3±1.6
	4 object	79.3±1.7	11.0±1.6	89.8±0.5	61.8±0.9	13.7±2.9	74.0±1.9

Appendix

Visualization of State/Task/Retrieval

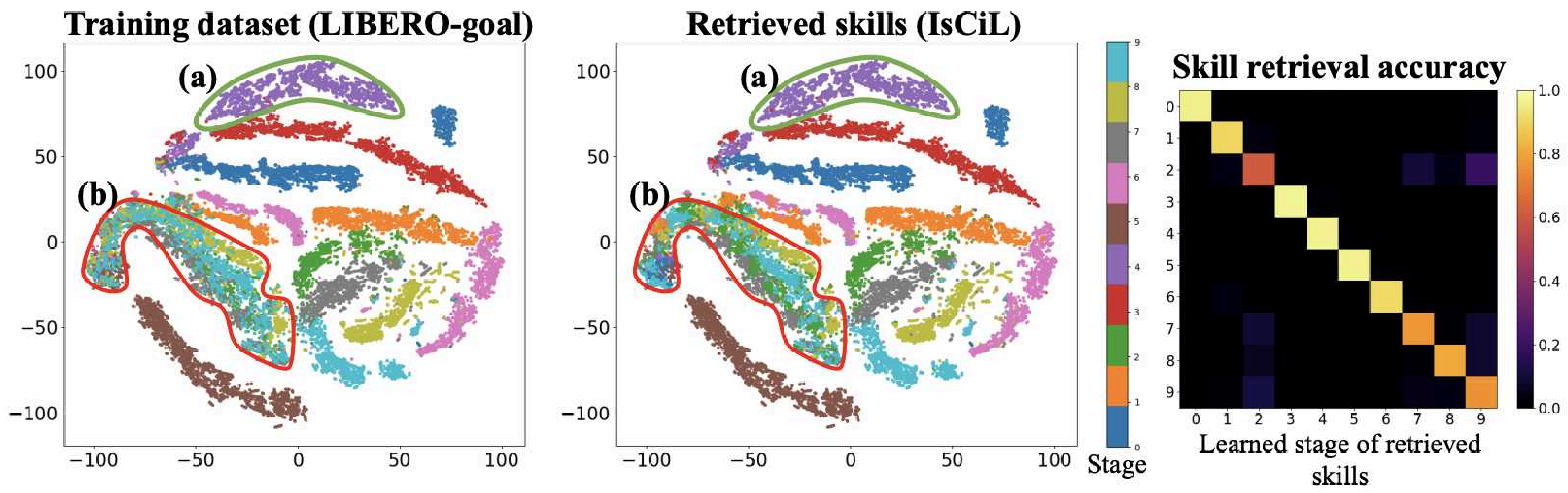


Figure 8: **Visualization of Skill Retriever on the LIBERO-goal Scenario. Left**: T-SNE visualization of the state space of the existing dataset for each stage. **Middle**: Visualization of the stages where skills retrieved by the Skill Retriever, after all CiL stages of learning. **Right**: Map showing the stages of each dataset and the retrieved skills. This demonstrates that the Skill Retriever can find skills capable of handling the given state, even in the LIBERO scenario. Additionally, in task-specific parts (a), it accurately retrieves the skills, and in parts showing similar behaviors (b), it shares skills.