Cross-task Cross-environment Knowledge Adaptation for Language Agents

Dongwon Kelvin Ryu[♠] **Gholamreza Haffari**[♠] **Meng Fang**[♣] **Ehsan Shareghi**[♠♡]
[♠]Department of Data Science & AI, Monash University
[♣]University of Liverpool [♡]University College London
firstname.lastname@monash.edu
meng.fang@liverpool.ac.uk ehsan.shareghi@ucl.ac.uk

Abstract

Current agentic workflows often suffer from knowledge silos, where insights are localized to specific task-environment pairs and lost during transitions, leading to inefficient learning. In this paper, we propose Cross-task Cross-environment Knowledge Adaptation (CKA), a framework designed to bridge this gap. CKA selectively stores high-quality causal hypotheses and employs a trio-mechanism for retrieval: leveraging same-task memory, task-similar knowledge, and high-performing cross-environment insights. To optimize the learning process, we introduce a curriculum learning approach that reorders task exposure based on knowledge quality to guide the agent from easier to more challenging environments. Our experiments on ScienceWorld and ALFWorld demonstrate that CKA effectively fosters mutual reinforcement across diverse environments, leading to substantial improvements in both success rates and overall performance.[1]

1 Introduction

The emergence of language model based agents has shifted the paradigm of learning from parameter fine-tuning to non-parametric memory adaptation (Yao et al., 2023; Shinn et al., 2023; Zhao et al., 2024; Majumder et al., 2024; Wang et al., 2024). By constructing, storing and retrieving natural language knowledge, agents can perform in-context learning to solve complex, multi-step tasks (Yao et al., 2023; Shinn et al., 2023; Kim et al., 2025). In these settings, the ability to extract insights from past experiences is critical for navigating partially observable environments.

Despite these advancements, a significant gap persists between the generation of knowledge and its effective utilization. Agentic workflows typically follow a pattern where memory is either adapted through multiple attempts within a single task-environment pair (Shinn et al., 2023; Kim et al., 2025) or transferred to initialize a novel task, only to be subsequently updated within a narrow, temporally localized window of recent episodes (Majumder et al., 2024; Zhao et al., 2024). This creates knowledge silos, where valuable insights are lost once an agent transitions to a new task-environment. Consequently, agents suffer from inefficient learning, failing to leverage the cross-dimensional relationships between different tasks and environments.

We hypothesize that causal hypotheses (Majumder et al., 2024) acquired in one task-environment contains transferable insights that can benefit performance in another. Transferring causal hypotheses as knowledge can enforce the behaviour of the agent that would not emerge otherwise, thereby enriching its decision-making capabilities. However, effective memory management remains a bottleneck, since agents often struggle to discern which specific pieces of knowledge are worth storing or retrieving. To address this, we introduce a scoring mechanism where the agent's performance serves as a proxy for knowledge quality.

We propose a framework, *Cross-task cross-environment Knowledge Adaptation (CKA)*, that selectively stores the highest-scoring knowledge per task-environment and facilitates them for knowledge refinement. Its retrieval follows trio-mechanism: previous knowledge of the same task-environment, cross-task knowledge sampled based on task similarity, and cross-environment knowledge sampled based on high score performance. CKA jointly adapts knowledge across a diverse set of task-environments to foster mutual reinforcement, enhancing the agent's overall performance.

Furthermore, we extend the utility of our quality proxy to govern the adaptation order. To facilitate this, we propose a *Performance-indexed Scheduler (PS)* that functions as a form of curriculum

[1]Code is available at https://github.com/ktr0921/cka

learning. This scheduler reorders task exposure, guiding the agent from high-performing (easier) environments to low-performing (more difficult) ones. Our experiments on ScienceWorld and ALFWorld demonstrate that this approach consistently improves score performance in ScienceWorld and increases success rates across both benchmarks.

2 Related Work

Interactive Learning Environments serve as foundational testbeds for evaluating language agents' capabilities across diverse task paradigms. Text-based simulations, including TextWorld (focused on everyday tasks) (Côté et al., 2018), ScienceWorld (scientific discovery tasks) (Wang et al., 2022), and DiscoveryWorld (Jansen et al., 2024), enable agents to leverage commonsense reasoning for sequential decision-making. Complementarily, embodied environments like ALFWorld (Shridhar et al., 2021) integrate multimodal observations (images and text) with textual actions, simulating real-world physical interactions. All these frameworks are formalized as partially observable Markov decision processes (POMDPs) to learn and test the policies for navigation, object manipulation, and contextual reasoning under uncertainty. Among these, ScienceWorld provides more controlled task-environment variations. We use both ScienceWorld and AlfWorld in our work.

Reasoning in Language Agents have focused on integrating reasoning loops into the decision-making process. Frameworks such as ReAct (Yao et al., 2023) integrate Chain-of-Thoughts (CoTs) (Wei et al., 2022) with action generation, where the agent generates thoughts to guide its immediate next step. In contrast, ReflAct focuses on reasoning about the current state relative to a long-term goal (Kim et al., 2025). While these methods enhance local decision-making, they are primarily designed for intra-episode interaction. Due to the knowledge deficiency from limited intra-episode experiences, they are often implemented with inter-episode interaction (Shinn et al., 2023) and inter-task execution (Majumder et al., 2024; Zheng et al., 2023).

Natural Language Memory has emerged as a dominant paradigm for encoding agent knowledge due to its interpretability and ease of manipulation. By storing and retrieving linguistic feedback, language agents dynamically adjust policies during complex tasks, enabling recovery from impasses through self-reflection or action refinement. Reflexion (Shinn et al., 2023) represents an early form of episodic memory, where failures are parsed to inform the subsequent episode within a localized context (e.g., same task in the same environment). Moving toward more generalized memory, ExpeL (Zhao et al., 2024) constructs natural language rules from past experiences to facilitate transfer to novel tasks. CLIN (Majumder et al., 2024) further advances this by constructing causal hypotheses from predefined task or environment configurations, which are then adapted within a specific task. Crucially, minimal research exists on how knowledge distilled across different tasks and environments in interactive simulation environments, leaving a critical void in continual learning methodologies for language agents. Our work move towards filling this gap.

3 Preliminaries

Text-based Simulation environments are modelled as a partially observable Markov decision process (POMDP) of a tuple of $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{O}, \Omega, \mathcal{R}, \gamma \rangle^{(\tau,\omega)}$, where $\mathcal{S}, \mathcal{A}, \Omega$ are sets of states, actions and observations while $\mathcal{P}$, $\mathcal{O}$, $\mathcal{R}$ and γ denote the transition probabilities, set of conditional observations probabilities, reward function and discount factor. τ and ω signify a task type and an environment configuration, and a task-environment refers to a unique tuple of them, (τ, ω). For every (τ, ω), there exists a unique POMDP per task-environment pair, which we will denote superscript of each variables.

The agent ought to select an appropriate action from a policy $a_t^{(\tau,\omega)} = \pi_\theta \left(. | \kappa^{(\tau,\omega)}, \xi_{t-1}^{(\tau,\omega)} \right)$, where $\kappa^{(\tau,\omega)}$ and $\xi^{(\tau,\omega)}$ are the learned knowledge and trajectory of (τ, ω). A trajectory at time t' is represented as $\xi_{t'}^{(\tau,\omega)} = (d, s, o_0, a_0, o_1, \cdots, o_{t'})^{(\tau,\omega)}$ with a task description in a given environment configuration denoted as $d^{(\tau,\omega)}$ and the total score denoted as $s^{(\tau,\omega)} = \sum_{i=0}^{T} r_i^{(\tau,\omega)}$.

Causal Hypotheses as Knowledge as a means of representing structured knowledge within a specific task-environment context was first proposed in the CLIN agent (Majumder et al., 2024). They capture explicit causal insights learned from past experiences to represent positive knowledge (`X MAY BE NECCESSARY to Y`, `X SHOULD BE NECCESSARY to Y`), negative knowledge (`X DOES NOT CONTRIBUTE to Y`) and uncertain speculative knowledge (`X MAY BE CONTRIBUTE to Y`) (Majumder et al., 2024).

Formally, let $\kappa_e^{(\tau,\omega)} \in K^{(\tau,\omega)}$ denote the set of causal hypotheses regarding (τ, ω) at episode e.

4 Methodology

4.1 Knowledge Storage

While we encode knowledge as causal hypotheses, to prevent memory redundancy and store useful insights, we maintain only the highest-quality knowledge per task-environment inside the memory, defined as $\hat{\kappa}^{(\tau,\omega)} = \arg\max_{\kappa_e^{(\tau,\omega)} \in K^{(\tau,\omega)}} Q(\kappa_e^{(\tau,\omega)})$, where $Q : K \to \mathbb{R}$ represents a quality metric. Given that the accumulated reward serves as a direct proxy for the effectiveness of the underlying causal hypotheses, we define the quality metrix as $Q(\kappa_e^{(\tau,\omega)}) \equiv s_e^{(\tau,\omega)}$, where $s_e^{(\tau,\omega)}$ is the total score that the agent achieved using $\kappa_e^{(\tau,\omega)}$. Thereby, for a given N number of τ and M number of ω, there exists $N \times M$ number of $\hat{\kappa}^{(\tau,\omega)}$ entailed with $\hat{s}^{(\tau,\omega)}$. Under this formulation, the agent's performance serves a dual purpose: evaluating task success and quantifying the reliability of the learned knowledge.

4.2 Unified Knowledge Generator

Cross-task Cross-environment Knowledge Adaptation (CKA) framework facilitates the construction of causal hypotheses for every episode to represent structured environmental insights and evaluate them in the subsequent episode. While baseline approaches like CLIN focus on localized knowledge refinement, CKA introduces a continuous transfer mechanism that shares cross-task and cross-environment insights to synthesize learned knowledge for the target task-environment. We propose a unified generator. More formally, for a given target (τ, ω) at episode e, CKA generates knowledge using:

$$\kappa_e^{(\tau,\omega)} = G_\theta \left(\xi_{T,e-1}^{(\tau,\omega)}, \hat{\kappa}^{(\tau,\omega)}, \hat{\kappa}^{(\mathcal{T},\omega)}, \hat{\kappa}^{(\tau,\Omega)} \right), \quad (1)$$

where $\hat{\kappa}^{(\mathcal{T},\omega)}$ and $\hat{\kappa}^{(\tau,\Omega)}$ are transferred cross-task and cross-environment knowledge. This unification enables a continuous transition between knowledge refinement and application.

4.3 Knowledge Retrieval

Cross-Task Knowledge Transfer via Semantic Embedding To facilitate transfer between task types, we sample $\hat{\kappa}^{(\mathcal{T},\omega)}$ based on semantic similarity. We utilize a Sentence Transformer (Reimers and Gurevych, 2019) to encode task descriptions $d^{(\tau,\omega)}$ into embeddings. The agent samples $\tau' \in \mathcal{T}$ within the same environment configuration ω that exhibit high cosine similarity to the current task embedding, $\Pr(\tau') \propto d^{(\tau,\omega)} \cdot d^{(\tau',\omega)}$. This ensures contextually relevant knowledge transfer, but also adds stochasticity in causal hypotheses generation.

Cross-Environment Knowledge Transfer via Performance Weighting To leverage insights across different environment configurations, we acquire $\hat{\kappa}^{(\tau,\Omega)}$ using a performance-weighted sampling mechanism. We apply the softmax function over the scores $\hat{s}^{(\tau,\omega')}$ achieved using $\hat{\kappa}^{(\tau,\omega')}$ to sample $\omega' \in \Omega$, $\Pr(\omega') \propto s^{(\tau,\omega')}/\epsilon$, where ϵ is the temperature. This prioritizes the transfer of high-quality, successful causal hypotheses that contributed to the agent's score performance.

4.4 Adaptation Order Dynamics

In CKA, because knowledge update and utilization occur simultaneously, the temporal sequence of task exposure becomes a critical factor. Knowledge acquired in early stages fundamentally shapes heuristic quality of subsequent acquisitions. We investigate two distinct adaptation paradigms: *Sequential Adaptation Order (SEQ)*, where the agent completes all episodes within a single task-environment before transitioning, and *Joint Adaptation Order (JNT)*, where the agent iterates through all task-environment pairs in every episode. While SEQ is a constraint of localized methods like Reflexion and CLIN, CKA's unified generator allows for dynamic task-environment switching. This flexibility enables the agent to transition between different task types and environment configurations mid-sequence, optimizing the learning trajectory based on real-time knowledge utility.

4.5 Performance-Indexed Scheduling

Our assumption behind knowledge storage mechanism is that higher quality knowledge matters more than lower quality, and that the quality can be effectively approximated by the agent performance score. Based on these two assumptions, we go one step further and propose *Performance-Indexed Scheduling (PS)*, an algorithm that dynamically prioritize task-environment pairs based on the agent's performance, systematically reschedule the order of learning. Formally, for N tasks (τ) and M environments (ω), PS schedules task-environment sequence via the function:

$$\text{Schedule}_Q(K) := (\kappa_1, \kappa_2, \ldots, \kappa_{N \times M}), \quad (2)$$

		ScienceWorld									ALFWorld		
		Score			SR: $70 \leq s$			SR: $s = 100$			SR		
		s_*	$\frac{s_*}{T_*}$	$\frac{p_*}{e_*}$	p_{70}	$\frac{p_{70}}{T_{70}}$	$\frac{p_{70}}{e_{70}}$	p_{100}	$\frac{p_{100}}{T_{100}}$	$\frac{p_{100}}{e_{100}}$	p_s	$\frac{p_s}{T_s}$	$\frac{p_s}{e_s}$
GPT-4.1-Mini	ReAct	7.78	0.828	–	0.0	*null*	–	0.0	*null*	–	0.0	*null*	–
	Reflexion	39.13	2.007	34.53	26.7	1.413	21.4	15.6	1.268	12.1	12.7	2.153	10.9
	ExpeL	38.28	1.525	–	25.6	1.533	–	16.3	1.116	–	10.4	1.351	–
	CLIN	51.8	3.258	24.33	46.7	2.937	21.3	24.4	1.616	8.9	30.2	2.013	16.8
	CKA_{SEQ}	43.42	2.426	18.79	33.3	1.85	16.1	20.0	1.136	7.2	35.7	2.112	15.7
	CKA_{JNT}	49.16	2.643	26.02	44.4	2.707	**31.7**	24.4	2.068	10.7	**51.0**	**3.954**	**29.0**
	CKA_{SEQ}+PS	60.2	3.382	**38.71**	48.9	3.544	23.9	**26.7**	**2.119**	10.7	–	–	–
	CKA_{JNT}+PS	**62.4**	**3.448**	31.84	**60.0**	**4.286**	29.9	**26.7**	1.949	**11.9**	–	–	–
Gemini-2.5-Flash	ReAct	6.93	1.155	–	2.2	0.183	–	2.2	0.183	–	0.0	*null*	–
	Reflexion	49.13	2.444	33.5	33.3	1.982	25.0	22.2	1.552	17.1	14.9	2.525	10.3
	ExpeL	42.42	1.844	–	34.9	1.278	–	20.9	1.0	–	14.9	2.258	–
	CLIN	54.8	4.349	22.62	48.9	3.976	22.4	23.3	2.402	10.7	74.5	5.56	44.6
	CKA_{SEQ}	56.78	4.937	28.3	48.9	4.405	23.9	22.2	2.337	12.3	75.6	5.6	**48.2**
	CKA_{JNT}	57.87	**5.121**	27.98	50.3	**4.931**	29.6	22.2	2.313	**17.5**	**76.7**	**6.728**	43.8
	CKA_{SEQ}+PS	59.15	4.929	28.42	53.3	4.89	29.0	**24.4**	**2.49**	15.7	–	–	–
	CKA_{JNT}+PS	**60.62**	5.094	**29.23**	**54.1**	4.874	**30.9**	**24.4**	2.44	16.3	–	–	–
Qwen3-8B	ReAct	0.0	*0.0*	–	0.0	*0.0*	–	0.0	*0.0*	–	0.0	*0.0*	–
	Reflexion	0.0	*0.0*	*0.0*	0.0	*0.0*	*0.0*	0.0	*0.0*	*0.0*	0.0	*0.0*	*0.0*
	ExpeL	0.0	*0.0*	–	0.0	*0.0*	–	0.0	*0.0*	–	0.0	*0.0*	–
	CLIN	38.27	4.88	19.35	24.4	3.156	16.8	24.4	4.493	14.1	71.7	6.578	39.6
	CKA_{SEQ}	36.02	3.07	16.97	22.2	1.935	10.6	18.9	2.616	8.5	76.8	6.4	40.2
	CKA_{JNT}	40.91	3.54	18.41	26.7	2.249	11.6	24.4	3.74	11.0	**78.9**	**6.982**	**40.8**
	CKA_{SEQ}+PS	51.19	5.66	24.0	**37.8**	**4.607**	17.6	30.0	4.779	14.7	–	–	–
	CKA_{JNT}+PS	**54.23**	**5.99**	**24.65**	**37.8**	4.266	**19.3**	**31.1**	**4.846**	**16.6**	–	–	–

Table 1: Main results; s, p, T and e signify score, success rate (SR) by %, step and episode. In ScienceWorld, s_* is the mean of maximum score achieved across task-environment, $s_* = \mathbb{E}_{\tau,\omega}[\hat{s}^{(\tau,\omega)}]$, whereas T_* and e_* are the mean time step and episode when s_* is reached. We also define two success rate, partial completion for 70 score, p_{70}, and full completion for maximum 100 score, p_{100}. T_{70} and T_{100} are the average step when partial and full completion are achieved whereas e_{70} and e_{100} are the average episode. Similarly, p_s in ALFWorld is the success rate (full completion) while T_s and e_s are the average step and episode when the agent accomplishes the simulation.

	Episodic	Cross-task/environment	
		$e = 1$	$e > 1$
ReAct	✗	✗	✗
Reflexion	✓	✗	✗
ExpeL	✗	✓	✗
CLIN	✓	✓	✗
CKA	✓	✓	✓

Table 2: Comparison of knowledge transfer methods. Episodic utilizes knowledge extracted from past episodes within the same task-environment, whereas cross-task cross-environment shares learned knowledge across different task types and environmental configurations, serving as knowledge initialization at $e = 1$ and knowledge refinement at $e > 1$. ✓ means the knowledge is 'transferred and used' while ✗ signifies the opposite.

such that $Q(\kappa_1) \geq Q(\kappa_2) \geq \cdots \geq Q(\kappa_{N\times M})$. In cases of equal quality scores $Q(\kappa_i) = Q(\kappa_j)$ where $i \neq j$, relative ordering is arbitrary. This prioritization leverages high-performance knowledge to bootstrap low-quality representations early in the learning cycle, which in parallel, effectively promotes easier task-environment pairs over harder ones, simulating curriculum learning.

5 Experimental Setup

We conducted experiments within ALFWorld (Shridhar et al., 2021), an embodied interactive world for daily tasks in households, and ScienceWorld (Wang et al., 2022), a text-based interactive simulation demanding complex interactive reasoning to complete science-theory assignments. We selected the full test set of ALFWorld, comprising 6 task types across 134 unique environment configurations. Due to the minimal similarity between environment configurations, only cross-environment generalization was evaluated under the same task type. On contrarily, since ScienceWorld provides finer control over task types and environment con-

figurations, we selected diverse 9 task types from 6 classes in 5 environment configurations, forming a grid-like task-environment set shown in Appendix.

We evaluate our framework against baselines: ReAct (Yao et al., 2023), Reflexion (Shinn et al., 2023), ExpeL (Zhao et al., 2024), and CLIN (Majumder et al., 2024). While CLIN uses causal abstractions as knowledge source like CKA, we also include ExpeL since it utilizes insights transferred from different tasks. The differences between how agents transfer knowledge are shown in Table 2. For all experiments, we utilize GPT-4.1-Mini (OpenAI, 2023), Gemini-2.5-Flash (Comanici et al., 2025) and Qwen3-8b (Yang et al., 2025) as our language model (LM) backbones. To ensure reproducibility and reduce stochastic variance, all the experiments are ran three times with zero temperature for all models, ensuring deterministic outputs across all trials.[2]

6 Results and Analysis

6.1 Main Results

To evaluate the effectiveness of our proposed method, we implemented ReAct, Reflexion and ExpeL provided by (Zhao et al., 2024) and CLIN by (Majumder et al., 2024), integrated with our own LM backbones. For our CKA framework, we retrieved 5 distinct knowledge components for knowledge refinement: previous same task-environment knowledge, two cross-task knowledge, and two cross-environment knowledge. CKA is evaluated under SEQ and JNT with and without PS in ScienceWorld, leveraging its intermediate reward signals. For ALFWorld, we only evaluated without PS due to ALFWorld's binary reward constraints.

The results, averaged over three runs, are summarized in Table 1, reporting the score performance for ScienceWorld and the success rate (SR) for both simulation environments. In ScienceWorld, we defined two success metrics: partial completion when a score reached 70, and full completion when a score reached 100. To account for temporal efficiency, we included the ratio of score performance to the number of time steps taken, referred to as step efficiency ($\frac{s}{T}$). This provides another basis for comparison: when agents achieve identical scores, a higher ratio indicates greater temporal efficiency, whereas when agents terminate at the same step, a higher ratio indicates superior task performance. We further extended this logic to episodes, reporting the ratio of score per episode ($\frac{s}{e}$) for ScienceWorld, SR per time step ($\frac{p}{T}$) and episode ($\frac{p}{e}$) for ScienceWorld and ALFWorld.

Baseline vs CKA. As illustrated in Table 1, the performance of the baseline models varies significantly across environments and backbones. ReAct demonstrated minimal effectiveness, yielding near-zero score and SR in most configurations. While Reflexion and ExpeL achieved non-zero performance, their results were inconsistent, with one outperforming the other depending on the specific simulation environment and evaluation metric. Notably, when utilizing the Qwen3-8B backbone, all three models failed to acquire any reward signals or successes in task completion. We observed that this is mainly due to overthinking of Qwen3-8B, where the agents continue generating CoTs without action selection. In contrast, CLIN provided more consistent results and outperformed the aforementioned baselines. Ultimately, CKA consistently outperforms all baseline models across all benchmarks, regardless of the underlying LM backbone.

Language Model Backbones. The score and SR gains achieved by CKA vary across different LM backbones. Specifically, Table 1 shows that when compared to the best-performing baseline, Gemini-2.5-Flash yields the smallest gains in both ScienceWorld and ALFWorld, whereas Qwen3-8B and GPT-4.1-Mini achieve the highest score and SR increases, respectively. However, these variations do not suggest that CKA's effectiveness is model-dependent, but rather, the gains appear to be inversely correlated with the performance of the model within a given environment. For example, because GPT-4.1-Mini performs poorly in ALFWorld, the relative improvement provided by CKA is significantly higher there than in ScienceWorld. Conversely, since Gemini-2.5-Flash already demonstrates high baseline proficiency across both environments, the margin for improvement is naturally narrower. This suggests that CKA provides a substantial performance boost by compensating for model-specific deficiencies through cross-task cross-environment knowledge transfer.

Adaptation Order Dynamics. One critical observation from Table 1 is that joint adaptation (JNT) consistently yields scores, SR, and episodic efficiency on-par with or superior to its sequential counterpart (SEQ). This suggests that simultaneous knowledge adaptation facilitates a more effec-

[2]The full details of experiments setup are in Appendix.

tive mechanism for agent's performance across episodes, driven by a synergistic effect where knowledge refinement in one task-environment actively improves another through knowledge transfer. In contrast, SEQ limits the agent's experiential breadth because task-environment encountered in the early stages of the adaptation cannot leverage insights generated in later stages. Another notable observation is that within ScienceWorld, JNT achieves higher scores despite full SR remaining comparable to SEQ. The fact that partial completion gains outpace full completion suggests an asymmetric performance improvement, where JNT disproportionately benefits poorly performing task-environment pairs over well-performing ones. Consequently, in complex environments like ScienceWorld, we speculate that JNT serves to elevate underperforming, highly complex task-environment pairs to a level of reasonable performance.

However, a nuanced trade-off emerges regarding step efficiency. While JNT achieves higher overall performance, SEQ requires fewer time steps per episode when it is on-par with JNT. For instance, within the same SR in ScienceWorld, SEQ maintains a higher step efficiency, indicating faster task completion within an episode. We attribute this divergence to the differing levels of knowledge stationarity between the two approaches. In SEQ, knowledge remains stationary during the adaptation of a specific task-environment pair, fostering more consistent causal hypothesis generation for more stable and focused behaviour for the target task-environment. Conversely, JNT introduces non-stationarity as knowledge is refined across all task-environment pairs simultaneously, the agent generates more diverse causal hypotheses. This results in more exploratory behaviour with higher time steps per episode, yet ultimately leads to superior task completion. These findings suggest that by jointly adapting across task-environment pairs, agents can achieve higher performance through integrated knowledge utilization in shorter episodes with slightly higher temporal cost per episode.

6.2 Performance-indexed Scheduler

As evidenced in Table 1, the integration of our PS consistently improves CKA's performance across all score and SR in ScienceWorld. To investigate these dynamics, we analyzed the adaptation curves for CKA_{JNT} with and without PS in Figure 1. The results indicate that CKA with PS achieves higher scores more rapidly during the early stages

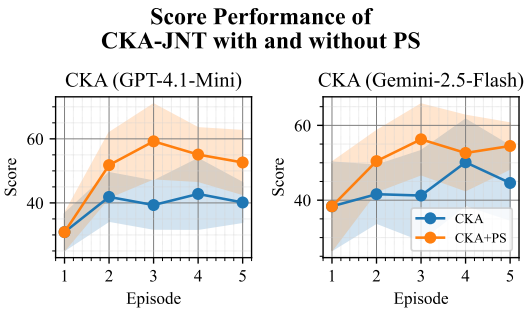

Figure 1: An adaptation curve of score over episode for CKA_{JNT} with and without PS. Two LM backbones, GPT-4.1-Mini and Gemini-2.5-Flash are used for the left and right plots, respectively.

of adaptation. This accelerated learning is driven by the PS's ability to reorder task-environment sequences following the prior performance. By prioritizing easier task-environment pairs, the agent effectively undergoes a curriculum-style adaptation that builds a robust knowledge foundation before tackling more complex task-environment. In contrast, the adaptation curve for CKA without PS exhibits greater irregularity. For instance, while GPT-4.1-Mini shows rapid adaptation between episode 1 and 2, Gemini-2.5-Flash demonstrates higher variance and a delayed learning curve between episodes 3 and 4. These observations suggest that the PS mechanism optimizes the learning trajectory by ensuring more stable and robust knowledge acquisition through strategic sequencing of simpler task-environment.[3]

6.3 Knowledge Type Ablation

To investigate the effectiveness of knowledge transfer and adaptation, we conducted an ablation study focused on the types of knowledge sharing within ScienceWorld using Gemini-2.5-Flash backbone. We expanded our experimental scope to include a wider range of 17 tasks within the same class set, resulting in 85 unique task-environment combinations. We first independently constructed causal hypotheses across these 85 task-environment pairs using CKA, then for every evaluation, we transferred knowledge excluding the target task-environment. Within CLIN, 5 sets of cross-task knowledge transfer within the same class (CLS) or cross-environment knowledge transfer (ENV)[4] has been assessed. In contrast, we evaluated CKA un-

[3]Reversed PS has been investigated and reported in the Appendix.

[4]CLIN only transfers knowledge between different task types within the same class or different environment configurations for knowledge initialization.

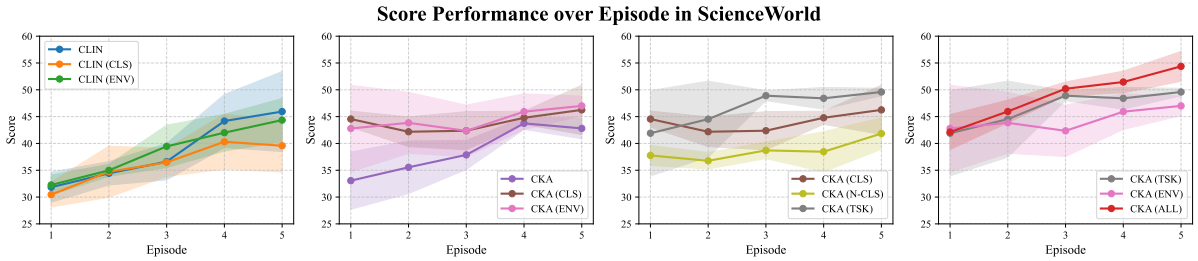

Figure 2: An adaptation curve of different knowledge types; CLIN and CKA are the agents without knowledge transfer. Specific type of knowledge transfer is denoted between parentheses. In these cases, the terms within the brackets can be CLS, N-CLS, TSK, ENV, and ALL, which represent cross-task within the same class, cross-task excluding tasks within the same class, cross-task with all available tasks, cross-environment and cross-task cross-environment knowledge transfer settings, respectively.

der 5 sets of knowledge across (1) solely cross-task knowledge within the same class (CLS), (2) cross-task knowledge excluding tasks within the same class (N-CLS), (3) cross-task knowledge with all available tasks (TSK), (4) cross-environment knowledge (ENV) and (5) a combined cross-task cross-environment knowledge (ALL) transfer strategy. These tests were conducted across 3 subsets of the 5 environment configurations to ensure robust results and represented in Figure 2.

As illustrated in Figure 2, CLIN demonstrates that sharing knowledge does not contribute to a score improvement in ScienceWorld. In contrast, CKA exhibits substantial performance gains through knowledge transfer, particularly during the initial episode of adaptation. While transferring CLS knowledge provides a measurable benefit, our ablation study reveals that the broader transfer of task types (TSK) is more effective than both CLS and N-CLS. Notably, while CLS knowledge facilitates superior zero-shot performance, TSK surpasses CLS as the agent progresses beyond the initial episode. Ultimately, the integrated ALL knowledge outperforms both TSK and ENV, which yields the most robust learning trajectory, characterized by a smooth, consistent adaptation curve and the lowest variance across episodes.

6.4 Shared Causal Hypotheses

To further elucidate the mechanics of our knowledge transfer framework, we additionally ran CKA_{JNT}+PS with cross-environment (ENV) transfer using Gemini-2.5-Flash and conducted an analysis of shared causal hypotheses against CKA_{BASE}, and CKA_{JNT}+PS with combined cross-task/cross-environment (ALL). We utilized both exact matching for HARD overlaps and DBSCAN[5] to cluster embeddings of hypotheses for SOFT overlaps within the same task type. As illustrated in Table 3, BASE primarily generates unique causal hypotheses with minimal overlap while ENV and ALL facilitate shared ones. ENV transfer results in causal hypotheses that are shared across 4-5 environments, whereas the ALL transfer produces more localized knowledge, characterized by a higher proportion of unique and partially shared hypotheses across 2–3 environments. These results suggest that while cross-environment transfer maximizes the breadth of shared insights within the same task type, the inclusion of cross-task knowledge introduces more specialized, environment-specific hypotheses. Crucially, these results imply that the framework can be used to optimize memory efficiency by identifying and removing redundant causal hypotheses, thereby reducing the overall memory footprint without sacrificing knowledge utility.

	SOFT			HARD		
# of ENV	BASE	ENV	ALL	BASE	ENV	ALL
5	0.0	**0.433**	0.3	0.0	**0.187**	0.166
4	0.0	**0.156**	0.098	0.0	**0.177**	0.147
3	0.013	0.097	**0.167**	0.0	**0.139**	0.109
2	0.064	0.052	**0.108**	0.004	0.106	**0.141**
1	**0.923**	0.262	0.326	**0.996**	0.392	0.438

Table 3: A proportion of generated causal hypotheses shared across five distinct environments inside memory. HARD refers to exact matching and SOFT signifies embedding matches between causal hypotheses. # of ENV is the number of environments in which causal hypotheses are shared.

[5]Hyperparameters and experiment details are in Appendix

7 Discussion

7.1 Cross-environment as Generalization

As shown in Table 3, in the cross-environment setting, CKA demonstrates an ability to distill localized insights into environment-agnostic causal hypotheses. By leveraging cross-environment knowledge through in-context learning, the agent moves beyond task-specific heuristics toward generalized principles. If causal hypotheses are well-generalized for optimal policy, theoretically, they can be directly applied to novel environment configurations within the same task class.

7.2 Cross-task as Proposal

Conversely, cross-task transfer serves a different functional role: it acts as a mechanism for hypothesis proposal. While task-specific requirements vary significantly, our findings in Figure 2 indicate that while learning from dissimilar tasks (N-CLS) can be detrimental to performance, the combination of similar and dissimilar tasks (TSK) yields superior results compared to either alone. This suggests that CKA does not merely perform semantic transfer, but also it uses transferred knowledge as a springboard to propose novel, creative causal hypotheses tailored to the current environment.

7.3 Knowledge Transfer as Exploration

Beyond mere performance gains, we speculate that the cross-task and cross-environment transfer facilitated by CKA induces a more diverse and sophisticated exploratory behaviour. Without such transfer, agents often become trapped in localized behavioural patterns, limiting their ability to discover novel states or representations. While certain task-environment requirements may be unreachable under a static, localized policy, transferred knowledge allows more dynamic policy modulation for further exploration. This expansion of state-space coverage enables the agent to transcend the limitations of its parameter-tuned policy.

Furthermore, we posit that knowledge transfer serves as a form of implicit exploration. In cross-environment settings, if causal hypotheses are sufficiently generalized and environment-agnostic, they can be applied to novel configurations without the need for direct trial-and-error. This allows the agent to effectively broaden its experiential horizon through zero-shot or few-shot application of high-quality knowledge, bypassing the inefficiencies of explicit exploration.

7.4 Performance as Curriculum

The Performance-indexed Scheduler (PS) is built upon the principle of prioritizing task-environment pairs that yield high performance metrics. Critically, these high-performing task-environment often correspond to simple tasks for the agents that their policy aligns well with. By deliberately sequencing task exposure from high-to-low performing task-environment pairs, PS implements a top-down curriculum, moving the agents from simpler problems towards increasingly more complex problems.

7.5 Asymmetric Knowledge Refinement

While PS functions as a curriculum, it can also be reconceptualized as a mechanism for asymmetric knowledge refinement. This perspective suggests that high-quality knowledge is more effective at refining low-quality knowledge than vice versa. This aligns with the intuition that injecting noisy or poor-quality knowledge into a robust knowledge is counterproductive. Instead, PS induces a high-to-low refinement cascade: by using high-quality insights to stabilize and improve weaker knowledge, we create an incremental strategy for global quality enhancement. While high-quality knowledge does not guarantee improvement in every instance, it provides a statistically superior foundation for progressive learning.

8 Conclusion

In this paper, we introduced Cross-task Cross-environment Knowledge Adaptation, a framework designed to overcome the limitations of knowledge silos in agentic workflows. By implementing a selective storage mechanism and a tripartite retrieval strategy, CKA enables agents to transcend localized learning through the effective transfer of causal hypotheses across varying tasks and environments. Furthermore, the Performance-indexed Scheduler demonstrates that leveraging performance as a curriculum can effectively guide agents from simpler to more complex environments through asymmetric knowledge refinement. Our results on ScienceWorld and ALFWorld demonstrate that this cross-task cross-environment knowledge sharing fosters mutual reinforcement, substantially boosting score performance and success rates. Ultimately, CKA provides a robust pathway for developing more resilient and adaptive language agents capable of leveraging past experiences to navigate complex virtual simulation environments.

9 Limitations

This work focuses on multi-task multi-environment text-based simulation settings, where environments must accept natural language commands, generate textual observations, and exhibit diversity in objectives as well as spatially or physically distinct worlds. Consequently, the applicability of our approach to non-textual or single-task environments may be limited. In addition, because our method does not involve fine-tuning, it relies on relatively large language models as the backbone of the framework, which may increase computational cost and restrict deployment in resource-constrained settings.

Finally, our approach may scale poorly to long-horizon tasks, as task sampling relies on Sentence-BERT similarity between task descriptions. Since only existing tasks whose descriptions closely match the target task are sampled, functionally related tasks described in dissimilar ways may be overlooked, and complex long-horizon tasks may lie far from any task in the pool in the embedding space. In such cases, the sampled tasks fail to cover the knowledge required by the target task, limiting transfer.

10 Potential Risks

Our work may contribute to development of agent that can understand and generate language, which could be repurposed for malicious activities, e.g., social engineering, misinformation generation or deceptive interaction. If deployed without sufficient safeguards, these agents may amplify existing biases and produce misleading contents.

References

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit S. Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, Luke Marris, Sam Petulla, Colin Gaffney, Asaf Aharoni, Nathan Lintz, Tiago Cardal Pais, Henrik Jacobsson, Idan Szpektor, Nan-Jiang Jiang, and 3279 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *ArXiv*, abs/2507.06261.

Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew J. Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. 2018. Textworld: A learning environment for text-based games. In *Computer Games - 7th Workshop, CGW 2018, Held in Conjunction with the 27th International Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, July 13, 2018, Revised Selected Papers*, volume 1017 of *Communications in Computer and Information Science*, pages 41–75. Springer.

Peter A. Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. 2024. Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Jeonghye Kim, Sojeong Rhee, Minbeom Kim, Dohyung Kim, Sangmook Lee, Youngchul Sung, and Kyomin Jung. 2025. Reflact: World-grounded decision making in LLM agents via goal-state reflection. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 33433–33465. Association for Computational Linguistics.

Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Peter Jansen, Oyvind Tafjord, Niket Tandon, Li Zhang, Chris Callison-Burch, and Peter Clark. 2024. CLIN: A continually learning language agent for rapid task adaptation and generalization. In *First Conference on Language Modeling*.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew J. Hausknecht. 2021. Alfworld: Aligning text and embodied environments for interactive learning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024. Voyager: An open-ended

embodied agent with large language models. *Transactions on Machine Learning Research*.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. 2022. ScienceWorld: Is your agent smarter than a 5th grader? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11279–11298, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

Longtao Zheng, R. Wang, and Bo An. 2023. Synapse: Trajectory-as-exemplar prompting with memory for computer control. In *International Conference on Learning Representations*.

A Appendix

A.1 Language Models

We evaluated both baseline and proposed agents using GPT-4.1-Mini, Gemini-2.5-Flash (via API), and Qwen3-8b (quantized to Q8 via Ollama). For all models, the temperature was set to zero, with all other hyper-parameters maintained at their default values. To accommodate varying reasoning patterns during causal hypothesis generation, we set the maximum token limit to 2048 for GPT-4.1-Mini and Gemini-2.5-Flash. Due to overthinking in Qwen3-8b, we increased it to 8192 for Qwen3-8b to prevent premature termination. For Qwen3-8b Q8 with Ollama, we used a single L40S GPU for each experiment for approximately 12 - 36 hours.

A.2 ALFWorld

ALFWorld is designed for embodied problem under 7 different task types:

- Look: look_at_obj_in_light
- Pick: pick_and_place_simple
- Move: pick_and_place_with_movable_recep
- Clean: pick_clean_then_place_in_recep
- Cool: pick_cool_then_place_in_recep
- Heat: pick_heat_then_place_in_recep
- Pick2: pick_two_obj_and_place

Its testing dataset consists of 6 different tasks (Look, Pick, Clean, Cool, Heat, Pick2) across 134 unique environment configurations. In ALFWorld, even when location names (e.g., kitchen, bathroom) were identical, other configuration differences rendered environments effectively dissimilar. Since cross-task transfer in our setting requires substantial environmental similarity for cross-task setting, we only included cross-environment transfer.

A.2.1 PS Implementation Barrier in ALFWorld

ALFWorld's binary reward system (only 0 or 1) creates a challenge for PS. The absence of intermediate rewards eliminates any meaningful metric to estimate knowledge quality, as PS requires graded performance signals. Consequently:

- Successful tasks (score of 1) need no repetition, making scheduling redundant
- Failed tasks (score of 0) provide no actionable performance metric for scheduling adjustments

Thus, the binary reward structure prevents effective implementation of PS in ALFWorld.

A.3 ScienceWorld

ScienceWorld (Wang et al., 2022) is a text-based simulation testbed, featuring tasks grounded in standardized elementary science curricula. The benchmark includes 30 distinct challenges spread across 10 scientific topics, each requiring agents to engage in multi-turn interactions and make long-term strategic decisions. For instance, completing a task such as 'boil the water' necessitates sequentially planning and executing several intermediate

steps (e.g., finding a pot, filling it with water, placing it on a stove, and turning on the heat) demonstrating the need for sophisticated, step-by-step reasoning. We selected a diverse set of tasks from 6 distinct classes up to 5 environments (indices 0–4) under a 'difficult' setting to maximize knowledge acquisition and transfer.

A.3.1 Main Experiments

For main results, 9 tasks were selected across the same 5 environments, executed for 5 episodes per task-environment. The default task sequence follows a structured order:

- States of matter (Class 1): freezing / melting (C1-T9/22)
- Electric circuits (Class 3): powering bulbs without (C3-T25); testing conductivity of known substances (C3-T27)
- Growing organism (Class 5): growing fruits (C5-T11)
- Chemical compounds (Class 6): mixing chemical compounds of varying complexity, easy / intermediate (C6-T2/3)
- Identifying organism (Class 8): identifying one or more animals (C8-T12)
- Mendelian genetics (Class 10): applying Mendelian genetics to known plants (C10-T23)

where 'C' and 'T' refer to the class and task index. In sequential adaptation, sequencing switched between cross-environment (when task type is the same) to cross-task (when task type transitions to another), e.g., (C1-T9-W0), (C1-T9-W1), ..., through (C10-T23-W4).

In CLIN, $k = 5$ for previous knowledge retrieval for causal hypotheses generation, while cross-task leverage used all prior tasks within the same class (e.g., C1-T22 relies on other C1 tasks), and cross-environment leveraged all explored environments.

CKA employed dynamic sampling:

- At episode of zero (knowledge initialization), upto 3 cross-task knowledge and 3 cross-environment knowledge were sampled.
- At episode more than zero (knowledge refinement), 1 same task-environment, 2 cross-task and 2 cross-environment knowledge were sampled

Our CKA agent is implemented using code base from (Majumder et al., 2024). Cross-task knowledge is sampled from task similarity distribution (a softmax over sentence embedding of task description using Sentence-Bert (Reimers and Gurevych, 2019)), while cross-environment sampling prioritized performance scores (a softmax over peak score for environments).

Also, to make a fair comparison, we shared the same initial trajectory at episode 0 in CKA_{SEQ}+PS, CKA_{JNT} and CKA_{JNT}+PS. This will reduce variance since their initial attempt and its subsequent PS scheduling will be the same across different versions.

A.3.2 Knowledge Type Ablation

For our knowledge type ablations, causal hypotheses per every task-environment are acquired inside memory using CKA. We included more number of tasks than the main experiments, 17 tasks spanning 5 distinct environments, with each task executed for 5 episodes, resulting in 85 unique task-environment pairs. The tasks include diverse scientific activities such as:

- States of matter (Class 1): boiling / changing / freezing / melting (C1-T0/1/9/22)
- Electric circuits (Class 3): powering bulbs without / with renewable energy (C3-T25/26); testing conductivity of known / unknown substances (C3-T27/28)
- Growing organism (Class 5): growing plants / fruits (C5-T10/11)
- Chemical compounds (Class 6): mixing chemical compounds of varying complexity, easy / intermediate / hard (C6-T2/3/4)
- Identifying organism (Class 8): identifying one or more animals / plants (C8-T12/13)
- Mendelian genetics (Class 10): applying Mendelian genetics to known / unknown plants (C10-T23/24)

For evaluation, both CLIN and CKA were tested on these tasks but confined to 3 environments (indices 0–2) across 5 episodes. During evaluation, we excluded the knowledge of target task-environment from memory (e.g., for a target $\tau =$ C1-T0 in $\omega =$ W0, we excluded C1-T0 of W0 from the memory).

Similarly, CKA employed dynamic sampling, where at episode of zero:

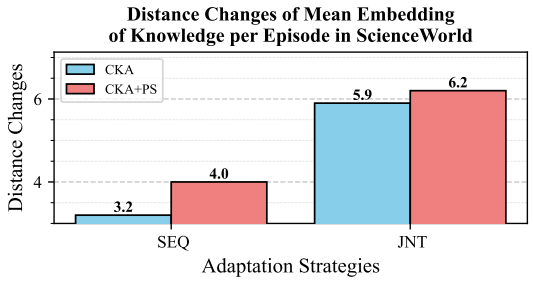

Figure 3: A comparison of expectation of knowledge embedding distance changes.

- Upto 3 cross-task knowledge for cross-task setting
- Upto 3 cross-environment knowledge for cross-environment setting
- Upto 3 cross-task knowledge and 3 cross-environment knowledge for cross-task cross-environment setting.

At episode of non-zero:

- 1 same-task same-environment and 4 cross-task knowledge for cross-task setting
- 1 same-task same-environment and 4 cross-environment knowledge for cross-environment setting
- 1 same-task same-environment, 2 cross-task and 2 cross-environment knowledge for cross-task cross-environment setting.

These ensure to transfer the same number of knowledge across different settings in CKA.

A.4 Knowledge Diversity

To quantify the diversity of causal abstractions generated during adaptation, we first transform each abstraction at episode $\kappa_{e,i}$ into a sentence embedding $f(\kappa_{e,i})$ using sentence transformers (Reimers and Gurevych, 2019). The knowledge representation for episode e is then defined as the average embedding across all causal abstractions: $f_\kappa(\kappa_e) \equiv \mathbb{E}_{\kappa_{e,i} \in \kappa_e}[f(\kappa_{e,i})]$. We measure the evolution of knowledge diversity by computing the expected Euclidean distance between consecutive episodes' representations, $D = \mathbb{E}[d(f_\kappa(\kappa_e), f_\kappa(\kappa_{e+1}))]$, where $d(\cdot,\cdot)$ is the Euclidean metric and $f_\kappa(\kappa_e) \neq \mathbf{0}$. This distance metric captures how significantly new knowledge deviates from prior causal hypotheses, as illustrated in Figure 3.

The results demonstrate that PS enhances knowledge diversity across both sequential and joint adaptation settings. By prioritizing high-performing task-environment during scheduling, PS encourages CKA to refine causal abstractions more actively. This increased diversity reflects the agent's ability to explore novel abstractions beyond previously established patterns, which aligns with our observation in Table 1 that PS consistently outperforms default ordering in adaptation outcomes. The emergence of broader knowledge exploration suggests that diversity-driven escape from local minima is a key mechanism behind performance gains of PS.

We draw an analogy between PS-induced knowledge diversity and optimization algorithms with high learning rates. Just as elevated learning rates enable models to avoid shallow local optima through larger parameter updates, PS generates more diverse knowledge hypotheses that test alternative reasoning pathways. While this increased exploration does not guarantee immediate performance improvements (similar to how high learning rates may cause instability), it prevents the agent from stagnating in suboptimal solution spaces.

Class-Task	Default s^*	PS Reversed s^*	PS s^*
C1-T9	7.2	58.8	**68.2**
C1-T22	27.3	31.8	**45.0**
C3-T25	78.3	**85.8**	82.8
C3-T27	40.0	22.0	**80.2**
C5-T11	10.4	22.0	**26.8**
C6-T2	**86.6**	73.2	**86.6**
C6-T3	**100**	**100**	90.0
C8-T12	**75.4**	51.0	68.0
C10-T23	17.3	**100**	13.8
MEAN	49.2	60.5	**62.4**
MEAN w/o T23	53.2	55.6	**68.5**

Table 4: Performance comparison of CKA_{JNT} for each task in ScienceWorld; **Bold** indicates the best performing score. Mean is the average over all tasks and Mean w/o T23 signifies mean without C10-T23.

A.5 Reversed Performance-indexed Scheduling

We additionally ran a reversed order of PS using GPT-4.1-Mini. The results are in Table 4, where Default is CKA_{JNT}, PS is CKA_{JNT}+PS and PS Reversed is reversed order of CKA_{JNT}+PS. As shown, reversed order initially appears effective due to its unexpectedly strong score performance, but a closer analysis reveals this is an illusion driven by the exceptional results on `'mendelian-genetics-known-plant [T23]'` Specifically, T23 alone yielded a perfect 100 score

CKA with Default Order	CKA with PS-based Order
• CONNECTING WIRES TO THE CORRECT TERMINALS ON COMPONENTS SHOULD BE NECESSARY to create an electrical circuit for testing conductivity. • DISCONNECTING AND RECONNECTING COMPONENTS MAY CONTRIBUTE to troubleshooting and ensuring proper circuit function before making a final decision.	• CONNECTING wires properly between the battery, aluminum foil, light bulb, and switch SHOULD BE NECESSARY to complete the electrical circuit for testing conductivity. • CONNECTING terminals or wires incorrectly or ambiguously MAY NOT CONTRIBUTE to successfully testing conductivity.
• ACTIVATING THE SWITCH SHOULD BE NECESSARY to complete the circuit and test if the aluminum foil conducts electricity.	• ACTIVATING the switch MAY BE NECESSARY to test if the aluminum foil conducts electricity by observing the light bulb. • OBSERVING the light bulb's state after activating the switch SHOULD BE NECESSARY to determine the conductivity of the aluminum foil before placing it in a box.

Table 5: Qualitative analysis on causal abstractions extracted from `test-conductivity` [T27] in environment configuration [W0] in ScienceWorld

for reversed PS, whereas default order and PS order only achieved approximately 17.3 and 13.8 scores respectively. When T23 is excluded from evaluation (Mean w/o T23 in Table 4), the reversed approach shows no significant advantage over default ordering, demonstrating that its apparent efficacy was entirely task-specific and not generalizable to broader adaptation task-environment.

A.6 Qualitative Analysis

Our qualitative analysis in Table 5 reveals a direct correlation between the quality of causal abstractions (knowledge) and score performance, demonstrating that higher-performing agents derive knowledge with superior structural accuracy. Poor performance outcomes stem predominantly from inadequate knowledge representations rather than hallucinations during action execution. Semantically, high-quality causal abstractions encapsulate detailed and precise task/environment details, whereas low-quality knowledge manifests as erroneous, sparse, misleading information, or complete absence of relevant concepts. This underscores that robust knowledge refinement is critical for mitigating performance degradation.

A.7 Soft Matching of Causal Hypotheses

We employed Density-Based Spatial Clustering of Applications with Noise (DBSCAN) to cluster embeddings of causal hypotheses generated by CKA. It is unsupervised non-parametric algorithm to group points that are closely packed together in a feature space. Embeddings are acquired the same way as Appendix A.4, and we set maximum distance (ϵ) to be 20 and minimum point to be 1 to allow a unique causal hypothesis, but cluster reasonable causal hypotheses to each other.

A.8 Out-of-Distribution Testing

For the CLIN and CKA$_{\text{JNT}}$+PS configurations described in Section 6.1, we evaluate out-of-distribution generalization on 5 *unseen* environment configurations in ScienceWorld:

- CLIN (Baseline): $s_0 = 39.62 \pm 8.4$
- CKA$_{\text{JNT}}$+PS (Ours): $s_0 = 48.53 \pm 6.5$

CKA retains a clear advantage on unseen configurations, demonstrating that it captures genuinely reusable causal knowledge rather than merely exploiting selection bias in the sampled environments.

A.9 Inference Cost

We quantify the token overhead of CKA$_{\text{JNT}}$+PS associated with knowledge generation (GPT-4.1):

- Mean input tokens: 803.2 at $e = 1$ and 1834.9 at $e > 1$
- Mean output tokens: 149.4 at $e = 1$ and 336.1 at $e > 1$

Retrieving one additional knowledge entry incurs roughly 336.1 extra input tokens, which is comparable to increasing the reflection k-window in Reflexion or causal hypothese k-window in CLIN. The overall token usage of CKA is therefore essentially on par with CLIN and Reflexion, and

the additional overhead is minimal. Thus, CKA can be viewed as a lightweight, complementary component within a larger agent framework.

Finally, while task-embedding matching introduces some computational cost, it is negligible in our experiments. The dominant latency arises from the simulation environment's response time rather than from retrieval or embedding computation.

A.10 Language Model Assistant

The manuscript was refined for grammatical accuracy and clarity through the assistance of various large language models, specifically Qwen3-Next-80B-A3B-Instruct, GLM-4.5-Air, GPT-5, and Gemma-4-26b.

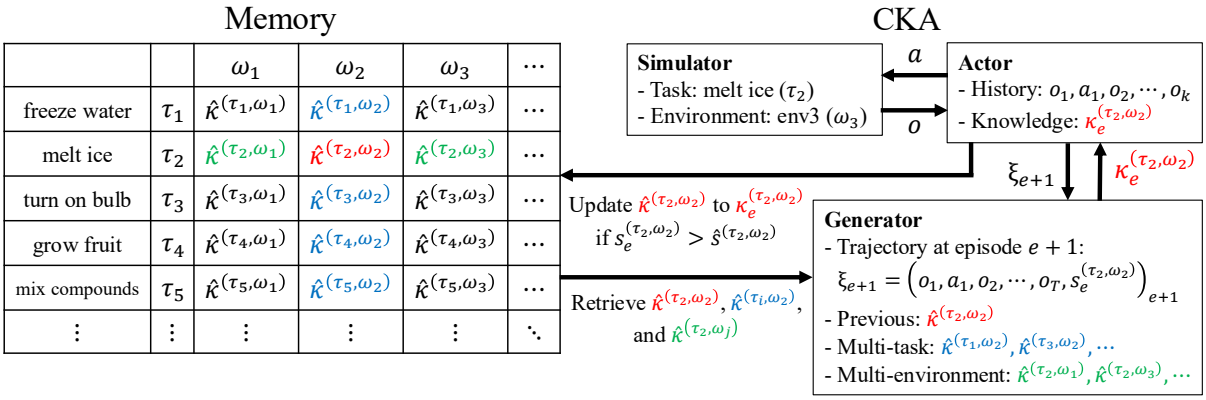

Figure 4: A CKA workflow on update, retrival and evaluation of knowledge; τ is the task type and ω is the environment configuration. Generator proposes the causal hypotheses knowledge of the target task-environment based on the previous knowledge and newly sampled trajectory from Actor. It also uses transferred insights from cross-task and cross-environment knowledge. Actor interacts with Simulator to evaluate the generated causal hypotheses knowledge and sample trajectories for the next generation.

You are given CURRENT TRACE, a sequence of actions that an agent made in a world to accomplish a task.

Task is detailed at the beginning. For each action, there is a rationale why the agent made that action. There is an observation that provide details about the new state of the world after each action was executed.
The CURRENT TRACE is accompanied by an EVALUATION REPORT indicating the success of the attempt to the task.

You will be provided with learnings from the previous attempts by the agent.
1. PREVIOUS LEARNINGS: SAME TASK in SAME WORLD, indicating exact task and world match.
2. CROSS-TASK LEARNINGS: DIFFERENT TASK in SIMILAR WORLD, indicating different tasks in the similar worlds/environments.
3. CROSS-WORLD LEARNINGS: SIMILAR TASK in DIFFERENT WORLD, indicating very alike tasks in different worlds/environments.

TASK indicates the task description and the LEARNINGS contains the summary of what has been learned from performing TASK. EVALUATION REPORT shows the final reward with LEARNINGS and how successful the respective attempt was for solving the TASK.

PREVIOUS LEARNINGS
{previous_knowledge}

CROSS-TASK LEARNINGS
{cross_task_knowledge}

CROSS-WORLD LEARNINGS
{cross_environment_knowledge}

CROSS-TASK LEARNINGS may contain useful environmental knowledge to solve the current task in the similar worlds/environments, such as presence of objects, starting location, navigational information, etc. On the other hand, CROSS-WORLD LEARNINGS may contain useful task principles to solve the current task in the current world/environment, but they may carry different environmental knowledge from the current one you are in, in terms of presence of objects, starting location, navigational information, etc.

Generate a summary of learning, as a numbered list, that will help the agent to successfully accomplish the current task in the current world.

Each numbered item in the summary can ONLY be of the form:
X MAY BE NECESSARY to Y.
X SHOULD BE NECESSARY to Y.
X MAY CONTRIBUTE to Y.
X DOES NOT CONTRIBUTE to Y.

The CURRENT TRACE has its current task and the interaction (action-observation), then EVALUATION REPORT at the end.

CURRENT TRACE
{trace}

Summary of learning as a numbered list:

Table 6: CKA's generator prompt.