北京大學
PEKING UNIVERSITY

Mem^p: Exploring Agent Procedural Memory

Mem^p: Exploring Agent Procedural Memory

Procedural memory
程序性记忆

- The type of long-term memory responsible for knowing how to perform tasks and skills.

- Markov Decision Process (MDP)

$$\tau = (s_0, a_0, o_1, s_1, a_1, o_2, \ldots, s_T), \qquad (1)$$

$$r = R(env, s_T, \tau) \in [0, 1] \qquad (2)$$

$$\pi(a_t|s_t) \longrightarrow \pi_{m^p}(a_t|s_t)$$

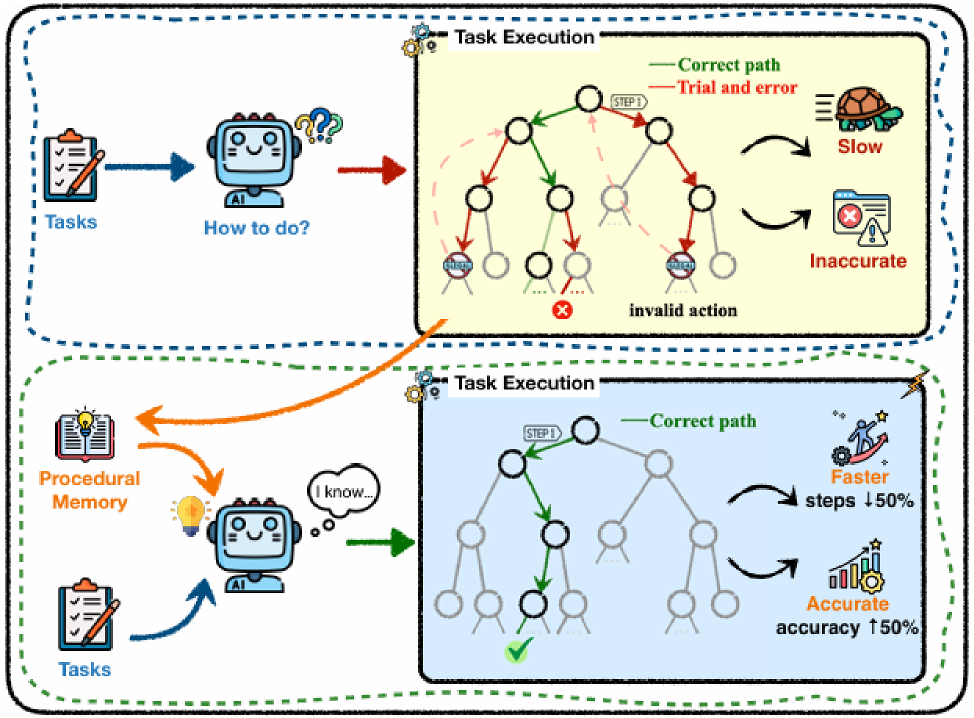


Figure 1: With **procedural memory**, agents can improve both the success rate (accuracy ↑) and execution efficiency (steps ↓) when solving similar tasks.

Mem^p: Exploring Agent Procedural Memory

北京大學 PEKING UNIVERSITY

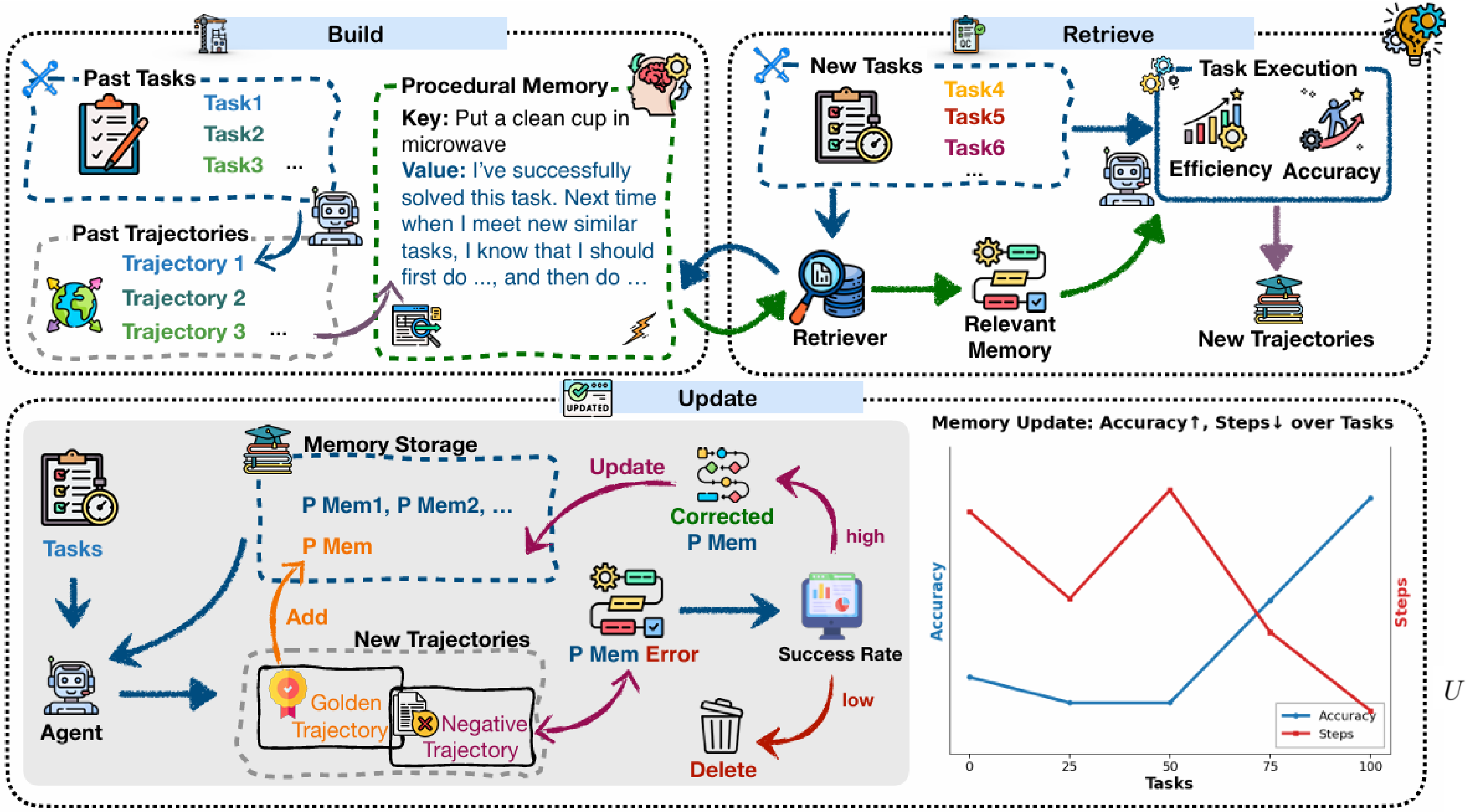


Figure 2: The procedural memory framework consists of **Build**, **Retrieve**, and **Update**, which respectively involve encoding stored procedural memory, forming new procedural memories, and modifying existing ones in light of new experiences.

Build

$$Mem = \sum_{t=1}^{T} m^{p_t}, where\ m^{p_t} = B(\tau_t, r_t)$$

Retrieve

$$m_{retrieved} = \arg\max_{m^{p_i} \in Mem} S(t_{new}, t_i)$$

Update

$$M(t+1) = U\left(M(t), E(t), \tau_t\right)$$

$$U = Add\left(M_{new}\right) \ominus Remove\left(M_{obso}\right) \oplus Update(M_{exist})$$

Mem^p: Exploring Agent Procedural Memory

Model	Granularity	TravelPlanner			ALFWorld		
		#CS ↑	*#HC* ↑	Steps ↓	Dev ↑	Test ↑	Steps ↓
GPT-4o	No Memory	71.93	**12.88**	17.84	39.28	42.14	23.76
	Script	72.08	5.50	15.79	66.67	56.43	18.52
	Trajectory	76.02	8.25	14.64	67.17	74.29	16.49
	Proceduralization	**79.94**	9.76	**14.62**	**87.14**	**77.86**	**15.01**
Claude-3.5-sonnet	No Memory	63.49	**33.06**	18.84	39.20	34.97	24.12
	Script	62.08	29.61	19.21	56.13	53.59	19.38
	Trajectory	65.76	29.61	17.72	69.28	71.78	15.97
	Proceduralization	**65.46**	30.14	**15.29**	**82.50**	**74.72**	**15.79**
Qwen2.5-72b	No Memory	56.57	7.34	18.32	44.91	41.25	21.38
	Script	58.59	7.34	18.53	66.24	61.88	17.13
	Trajectory	63.41	12.66	18.12	64.49	69.57	16.40
	Proceduralization	**63.82**	**14.19**	**17.94**	**85.71**	**77.19**	**15.32**

Table 1: Results on **Build Policy**. *#CS*, *#HC* denote Commensense and Hard Constraint, respectively. ↑ indicates the higher values are better, and ↓ denotes the lower values are better. The best results among all methods with similar settings are **bolded**, and the second-best results are underlined.

- **No Memory**
- **Trajectory**: filter complete gold trajectories
- **Script:** distill abstract knowledge using LLM
- **Proceduralization**: Trajectory + Script

Mem^p: Exploring Agent Procedural Memory

Model	Policy	#CS ↑	#HC ↑	Steps ↓
GPT-4o	No Memory	71.93	**12.88**	17.84
	Random Sample	74.59	6.72	15.12
	Key=Query	73.38	8.95	15.44
	Key=AveFact	**76.02**	8.25	**14.64**
Claude-3.5-sonnet	No Memory	63.49	**33.06**	18.84
	Random Sample	63.99	29.91	17.93
	Key=Query	64.93	28.56	**17.60**
	Key=AveFact	**65.76**	29.61	17.72
Qwen2.5-72b	No Memory	56.57	7.34	18.32
	Random Sample	59.76	8.43	18.31
	Key=Query	61.71	11.97	18.54
	Key=AveFact	**63.41**	**12.66**	**18.12**

Table 2: Results on **Retrieve Policy** on TravelPlanner.

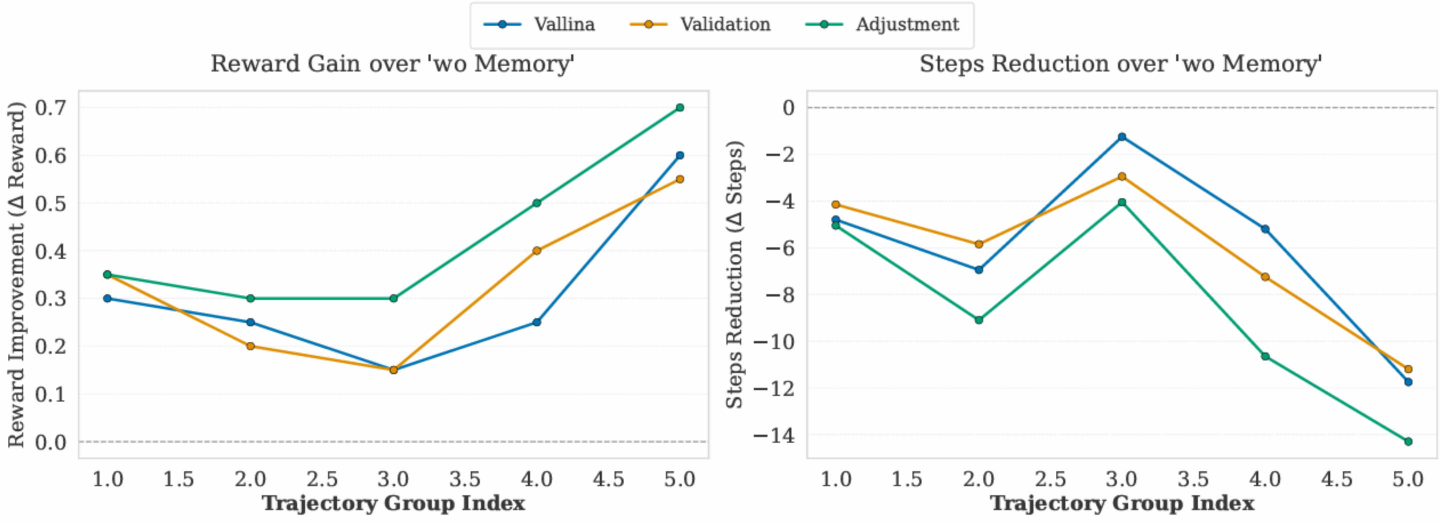


Figure 3: Reward gain and steps reduction vs. trajectory group index with **procedural memory**.

- **Random Sample**
- **Query**
- **AveFact:** extract keywords from queries

- **Vanilla Memory Update:** all trajectories
- **Validation:** only successful trajectories
- **Adjustment:** reflection for erroneous trajectories

Mem^p: Exploring Agent Procedural Memory

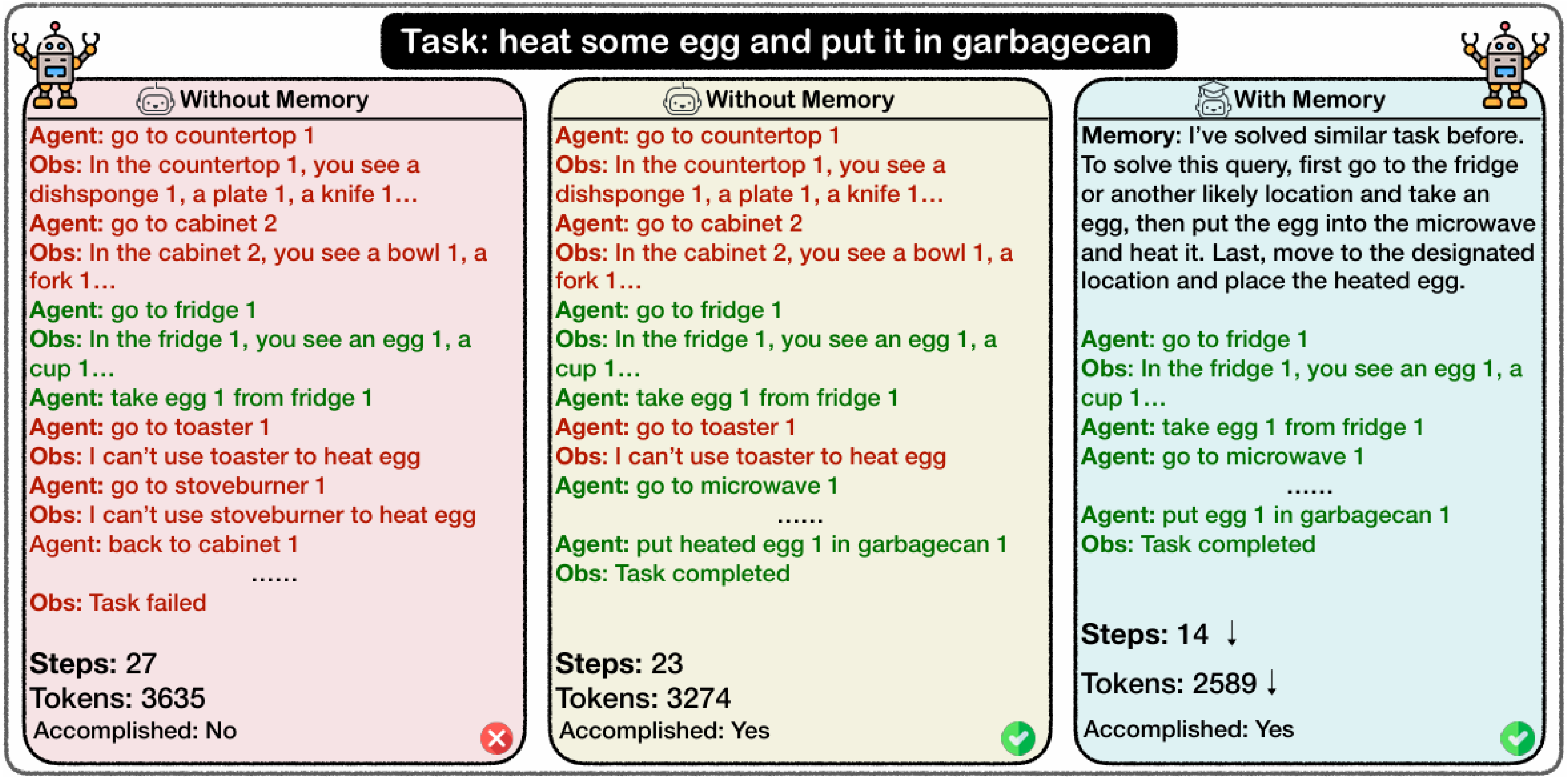


Figure 5: Compare trajectories with and without procedural memory, shortens the process by **9** steps and saves **685** tokens.

Mem^p: Exploring Agent Procedural Memory

北京大学 PEKING UNIVERSITY

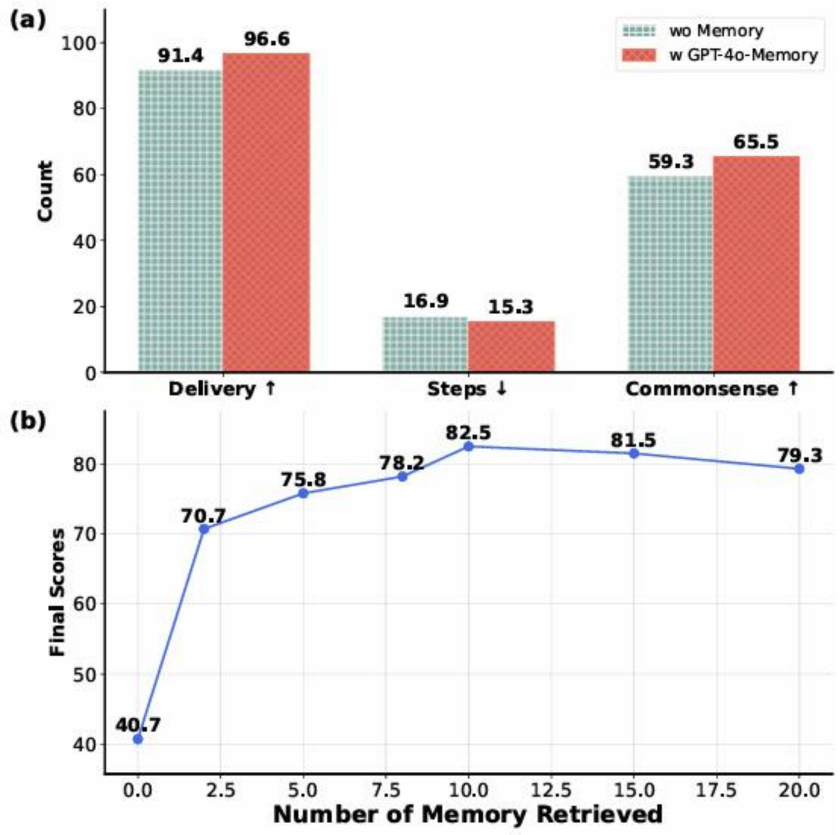


Figure 4: **(a)** Transfer result of GPT-4o's procedural memory to Qwen2.5-14B-Instruct and its performance on TravelPlanner dataset.**(b)** The relationship between the quantity of procedural memory retrieved for GPT-4o's performance on the ALFWorld dataset.

- Procedural memory exhibits transferability from strong models to weaker ones.
  - procedural memory generated by GPT-4o was employed by Qwen2.5-14B.
- Scaling Memory Retrieval Improves Agent Performance.

北京大學
PEKING UNIVERSITY

CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

北京大学 PEKING UNIVERSITY

PPO算法

$$L_i^{CLIP+VF}(\theta) = \mathbb{E}_i[L_i^{CLIP}(\theta) - c \cdot L_i^{VF}(\theta)]$$

$$\max_\theta \Sigma_{t=1}^T \mathbb{E}_{s \sim S_t, a \sim \pi_\theta(\cdot|s)} \left[r_t(s,a)\right]$$

In CL setting

$$\max_\theta \mathbb{E}_{s \sim S_t, a \sim \pi_\theta(\cdot|s)} \left[r_t(s,a)\right] - \mathbb{E}_{s \in S_{t-1}} D_{\mathrm{KL}}(P_{\pi_\theta}(a|s) \parallel P_{\pi_{t-1}}(a|s))$$

- 策略学习的目标是最大化模型生成高奖励结果的概率，而知识保留的目标是保留生成高奖励结果的知识

$$\max_\theta \mathbb{E}_{(s,a) \in D_1} r_t(s,a) - \mathbb{E}_{(s,a) \in D_2} D_{\mathrm{KL}}(P_{\pi_\theta}(a|s) \parallel P_{\pi_{t-1}}(a|s))$$

$$D_1 = \{(s,a) \mid s \sim S_t, a \sim \pi_\theta(\cdot|s), P_{\pi_\theta}(a|s) > \mu_a[P_{\pi_\theta}(a|s)] + k\sigma_a[P_{\pi_\theta}(a|s)]\}$$

$$D_2 = \{(s,a) \mid s \sim S_{t-1}, a \sim \pi_{t-1}(\cdot|s), r_t(s,a) > \mu_a[r_t(s,a)] + k\sigma_a[r_t(s,a)]\}$$

CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

$$\max_{\theta} \mathbb{E}_{(s,a)\in D_1} r_t(s,a) - \mathbb{E}_{(s,a)\in D_2} D_{\mathrm{KL}}(P_{\pi_\theta}(a|s) \parallel P_{\pi_{t-1}}(a|s))$$

$$D_1 = \{(s,a) \mid s \sim S_t, a \sim \pi_\theta(\cdot|s), P_{\pi_\theta}(a|s) > \mu_a[P_{\pi_\theta}(a|s)] + k\sigma_a[P_{\pi_\theta}(a|s)]\}$$

$$D_2 = \{(s,a) \mid s \sim S_{t-1}, a \sim \pi_{t-1}(\cdot|s), r_t(s,a) > \mu_a[r_t(s,a)] + k\sigma_a[r_t(s,a)]\}$$

- 将KL散度计算简化为L2距离计算

$$L_i^{KR}(\theta) = (\log P_{\pi_\theta}(x_i) - \log P_{\pi_{t-1}}(x_i))^2$$

- 整合后的目标函数

$$\begin{aligned}\mathbf{J}'(\theta) &= L_i^{I_{D_1}\cdot CLIP + I_{D_2}\cdot KR + VF}(\theta) \\ &= \mathbb{E}_i[\underbrace{I_{D_1}(x)}_{\alpha(x)} \cdot L_i^{CLIP}(\theta) - \underbrace{I_{D_2}(x)}_{\beta(x)} \cdot L_i^{KR}(\theta) - c \cdot L_i^{VF}(\theta)] \\ &= \mathbb{E}_i[\alpha(x) L_i^{CLIP}(\theta) - \beta(x) L_i^{KR}(\theta) - c \cdot L_i^{VF}(\theta)]\end{aligned}$$

CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

北京大学
PEKING UNIVERSITY

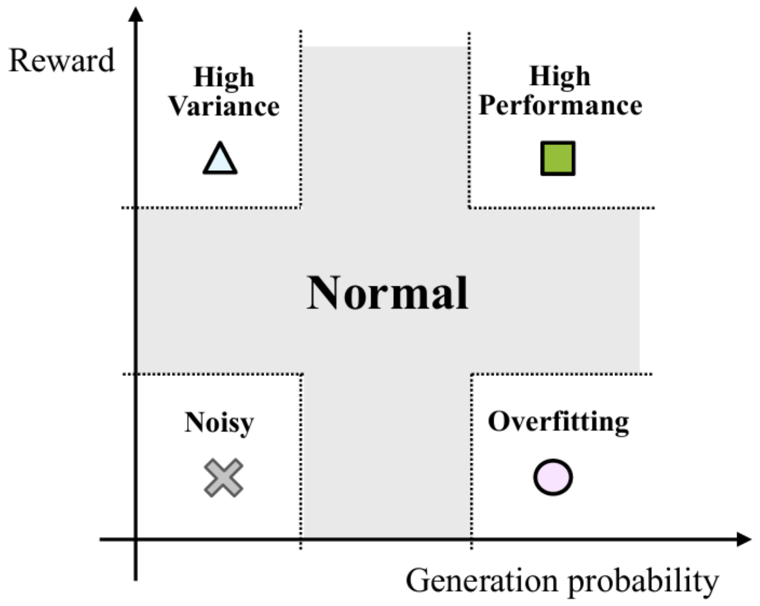


Figure 1: Five types of the rollout are utilized in our method.

Table 1: The determining condition of rollout type and corresponding weight strategy to balance policy learning and knowledge retention. We monitor the generating probability $\mathbf{P}_{\pi_\theta}(x)$ and the corresponding reward score $\mathbf{R}(x)$. The rollout type of sample x depends on whether the $\mathbf{P}_{\pi_\theta}(x)$ and $\mathbf{R}(x)$ fall in or outside the discriminant interval $(F[\cdot], G[\cdot])$.

ID	Rollout Type	Determining Condition		Weight Strategy	
r_1	High-performance	$\mathbf{P}_{\pi_\theta}(x) \geq G[\mathbf{P}_{\pi_\theta}]$	$\mathbf{R}(x) \geq G[\mathbf{R}]$	$\alpha(x) \uparrow$	$\beta(x) \uparrow$
r_2	Overfitting	$\mathbf{P}_{\pi_\theta}(x) \geq G[\mathbf{P}_{\pi_\theta}]$	$\mathbf{R}(x) \leq F[\mathbf{R}]$	$\alpha(x) \uparrow$	$\beta(x) \downarrow$
r_3	High-variance	$\mathbf{P}_{\pi_\theta}(x) \leq F[\mathbf{P}_{\pi_\theta}]$	$\mathbf{R}(x) \geq G[\mathbf{R}]$	$\alpha(x) \uparrow$	$\beta(x) \downarrow$
r_4	Noisy	$\mathbf{P}_{\pi_\theta}(x) \leq F[\mathbf{P}_{\pi_\theta}]$	$\mathbf{R}(x) \leq F[\mathbf{R}]$	$\alpha(x) \downarrow$	$\beta(x) \downarrow$
r_5	Normal	$\mathbf{P}_{\pi_\theta}(x)$ or $\mathbf{R}(x) \in (F, G)$		–	–

$$F[\cdot] = \mu[\cdot] - k\sigma[\cdot] \qquad G[\cdot] = \mu[\cdot] + k\sigma[\cdot]$$

$$\begin{aligned}\mathbf{J}(\theta) &= L_i^{\alpha\cdot CLIP+\beta\cdot KR+VF}(\theta) \\ &= \mathbb{E}_i[\alpha(x)L_i^{CLIP}(\theta) - \beta(x)L_i^{KR}(\theta) - c\cdot L_i^{VF}(\theta)]\end{aligned}$$

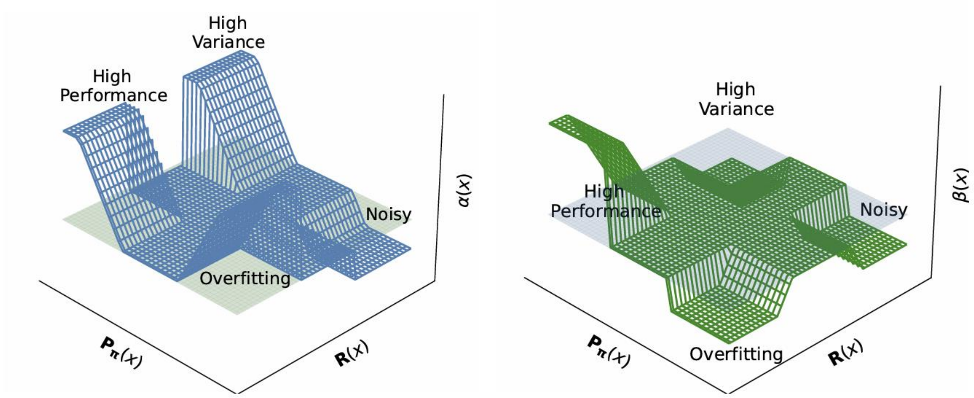
CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

Table 2: The constraint of weights and heuristic weights.

ID	Constraint of $\alpha(x)$	Constraint of $\beta(x)$	Heuristic $\alpha(x)$	Heuristic $\beta(x)$
r_1	$\alpha(x_{r_5}) - \alpha(x_{r_1}) < 0$	$\beta(x_{r_5}) - \beta(x_{r_1}) < 0$	$\min(ub, \frac{P_{\pi_\theta}(x) - \mu[P_{\pi_\theta}]}{k\sigma[\pi_\theta]})$	$\min(ub, \frac{\mathbf{R}(x) - \mu[\mathbf{R}]}{k\sigma[\mathbf{R}]})$
r_2	$\alpha(x_{r_5}) - \alpha(x_{r_2}) < 0$	$\beta(x_{r_2}) - \beta(x_{r_5}) < 0$	$\min(ub, \frac{P_{\pi_\theta}(x) - \mu[P_{\pi_\theta}]}{k\sigma[\pi_\theta]})$	$\max(lb, 2 + \frac{\mathbf{R}(x) - \mu[\mathbf{R}]}{k\sigma[\mathbf{R}]})$
r_3	$\alpha(x_{r_5}) - \alpha(x_{r_3}) < 0$	$\beta(x_{r_3}) - \beta(x_{r_5}) < 0$	$\min(ub, \frac{P_{\pi_\theta}(x) - \mu[P_{\pi_\theta}]}{k\sigma[\pi_\theta]})$	$\max(lb, 2 + \frac{\mathbf{R}(x) - \mu[\mathbf{R}]}{k\sigma[\mathbf{R}]})$
r_4	$\alpha(x_{r_4}) - \alpha(x_{r_5}) < 0$	$\beta(x_{r_4}) - \beta(x_{r_5}) < 0$	$\max(lb, 2 + \frac{P_{\pi_\theta}(x) - \mu[P_{\pi_\theta}]}{k\sigma[\pi_\theta]})$	$\max(lb, 2 + \frac{\mathbf{R}(x) - \mu[\mathbf{R}]}{k\sigma[\mathbf{R}]})$
r_5	–	–	1	1
All	$\mathbb{E}_{x \sim \pi_{t-1}}[\alpha(x)] = 1$	$\mathbb{E}_{x \sim \pi_{t-1}}[\beta(x)] = 1$	–	–

- Heuristic $\alpha(x)$ and $\beta(x)$
- Learnable $\alpha(x)$ and $\beta(x)$

$$\mathbf{L}_{coef}(\phi) = \mathbb{E}_{x \sim \pi_{t-1}}[(\alpha_\phi(x) - 1)^2 + (\beta_\phi(x) - 1)^2] + \tau(\alpha(x_{r_5}) - \alpha(x_{r_1}) + \beta(x_{r_5}) - \beta(x_{r_1}) + \alpha(x_{r_5}) - \alpha(x_{r_2}) + \beta(x_{r_2}) - \beta(x_{r_5}) + \alpha(x_{r_5}) - \alpha(x_{r_3}) + \beta(x_{r_3}) - \beta(x_{r_5}) + \alpha(x_{r_4}) - \alpha(x_{r_5}) + \beta(x_{r_4}) - \beta(x_{r_5}))$$



(a) Surface of heuristic $\alpha(x)$ (b) Surface of heuristic $\beta(x)$

Figure 2: The surfaces of heuristic weights. The weights are equal to 1 when rollout samples fall in the normal zone.

CPPO: CONTINUAL LEARNING FOR REINFORCEMENTLEARNING WITH HUMAN FEEDBACK

Method	Task-1 (M_{π_1}) rPMS$_1$ (↑)	rouge (↑)	AT (↓)	Task-2 (M_{π_2}) rPMS$_2$ (↑)	rouge (↑)	SFR (↓)	Final eval (M_{π_2}) rPMS (↑)	rouge (↑)
Human	2.958	–	–	2.805	–	–	2.903	–
ChatGPT	3.298	0.197	–	3.189	0.191	–	3.242	0.193
SFT (In order)	1.499 ±0.130	**0.248** ±0.006	–	1.543 ±0.067	**0.237** ±0.007	–	1.498 ±0.051	**0.237** ±0.009
SFT (multi-tasks)	1.524 ±0.041	0.254 ±0.011	–	1.536 ±0.092	0.234 ±0.009	–	1.505 ±0.011	0.236 ±0.008
PPO (In order)*	2.629 ±0.183	0.196 ±0.050	0.052 ±0.044	2.546 ±0.201	0.151 ±0.022	0.144 ±0.024	2.502 ±0.242	0.186 ±0.016
Iterated RLHF†	2.629 ±0.183	0.196 ±0.050	0.052 ±0.044	2.732 ±0.163	0.211 ±0.011	0.061 ±0.018	2.666 ±0.124	0.200 ±0.010
PPO	*2.629* ±0.183	*0.196* ±0.050	*0.052* ±0.044	*2.687* ±0.126	*0.184* ±0.017	*0.080* ±0.017	*2.612* ±0.191	*0.188* ±0.013
PPO+OnlineL2 Reg	*2.758* ±0.121	*0.206* ±0.042	*0.042* ±0.042	*2.701* ±0.205	*0.180* ±0.012	*0.062* ±0.013	*2.700* ±0.114	*0.196* ±0.011
PPO+EWC (Kirkpatrick et al., 2017)	*2.833* ±0.122	*0.201* ±0.043	*0.047* ±0.039	*2.823* ±0.192	*0.175* ±0.022	*0.040* ±0.015	*2.801* ±0.202	*0.196* ±0.023
PPO+MAS (Aljundi et al., 2018)	*2.712* ±0.132	*0.211* ±0.051	*0.034* ±0.037	*2.726* ±0.189	*0.157* ±0.021	*0.039* ±0.020	*2.714* ±0.167	*0.179* ±0.011
PPO+LwF (Li & Hoiem, 2018)	*2.822* ±0.126	*0.197* ±0.051	*0.048* ±0.050	*2.832* ±0.179	*0.169* ±0.036	*0.030* ±0.019	*2.824* ±0.192	*0.179* ±0.019
PPO+TFCL (Aljundi et al., 2019)	*2.867* ±0.109	*0.202* ±0.039	*0.043* ±0.046	*2.864* ±0.169	*0.169* ±0.020	*0.054* ±0.022	*2.842* ±0.211	*0.178* ±0.014
PC (Kaplanis et al., 2019)	*2.692* ±0.117	*0.209* ±0.048	*0.036* ±0.055	*2.723* ±0.195	*0.165* ±0.019	*0.047* ±0.017	*2.703* ±0.191	*0.187* ±0.016
HN-PPO (Schöpf et al., 2022)	*2.859* ±0.105	*0.212* ±0.034	*0.036* ±0.042	*2.868* ±0.132	*0.171* ±0.017	*0.028* ±0.029	*2.846* ±0.177	*0.201* ±0.011
NLPO (Ramamurthy et al., 2022)	*2.784* ±0.102	*0.185* ±0.041	*0.060* ±0.050	*2.796* ±0.116	*0.172* ±0.021	*0.012* ±0.012	*2.799* ±0.146	*0.181* ±0.022
CPPO (Heuristic)	*3.020* ±0.137	*0.213* ±0.024	*0.035* ±0.023	*2.978* ±0.113	*0.174* ±0.019	***-0.164*** ±0.009	*3.099* ±0.153	*0.179* ±0.016
CPPO (Learn)	***3.180*** ±0.154	*0.220* ±0.040	***0.028*** ±0.042	***3.085*** ±0.134	*0.164* ±0.024	*-0.161* ±0.008	***3.207*** ±0.113	*0.179* ±0.008

Table 7: Ablation study. PPO is a special case of CPPO ($^*\alpha \equiv 1, \beta \equiv 0$).

Method	Task-1 rPMS$_1$ (↑)	rouge (↑)	AT (↓)	Task-2 rPMS$_2$ (↑)	rouge (↑)	SFR (↓)
CPPO / **Heuristic**	3.020 ±0.137	0.213 ±0.024	0.035 ±0.023	2.978 ±0.113	0.174 ±0.019	**-0.164** ±0.009
CPPO / **Learn**	**3.180** ±0.154	**0.220** ±0.040	**0.028** ±0.042	**3.085** ±0.134	0.164 ±0.024	-0.161 ±0.008
PPO / $\alpha \equiv 1, \beta \equiv 0$	2.629 ±0.183	0.196 ±0.050	0.052 ±0.044	2.687 ±0.126	0.184 ±0.017	0.080 ±0.017
CPPO / $\alpha \equiv 1$	2.837 ±0.124	0.196 ±0.029	0.047 ±0.041	2.745 ±0.121	0.169 ±0.020	-0.031 ±0.010
CPPO / $\beta \equiv 1$	2.476 ±0.117	0.185 ±0.021	0.063 ±0.025	2.520 ±0.119	**0.186** ±0.017	0.051 ±0.009
CPPO / $\beta \equiv 0$	2.012 ±0.186	0.209 ±0.022	0.038 ±0.045	2.436 ±0.141	0.174 ±0.021	0.142 ±0.015