PREPING: Zero-Task Agent Memory Construction via Self-Practice

Yumin Choi [1] **Sangwoo Park** [1] **Minki Kang** [1] **Jinheon Baek** [1,†] **Sung Ju Hwang** [1,2,†]

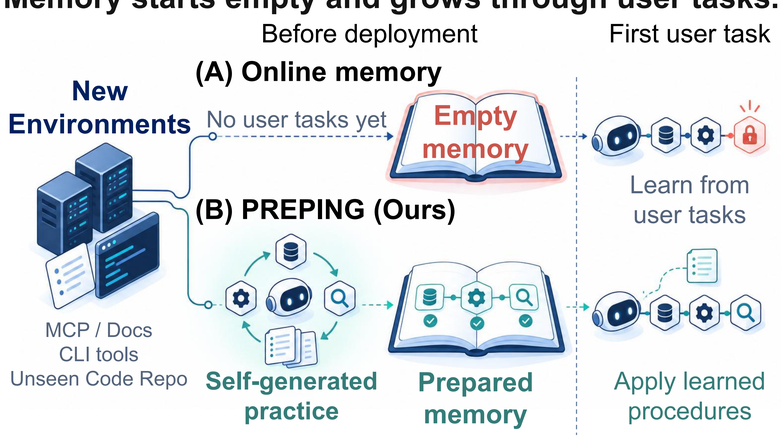
[1] KAIST [2] DeepAuto.ai [†] Equal advising

Project Page

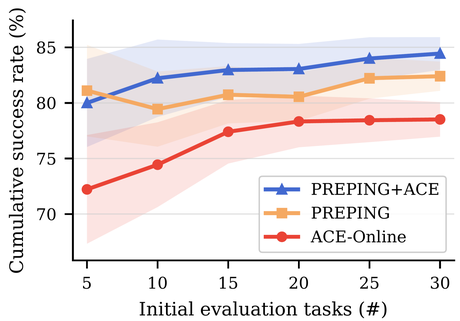
TL;DR: Can agents learn useful workflows and tool-use strategies in a new environment **without human intervention?**
PREPING constructs procedural memory through **self-practice** by controlling **what to practice** and **what to store.**

Motivation: The Cold-Start Gap

Memory starts empty and grows through user tasks.



Starting with empty memory can limit task performance.



First 10 AppWorld tasks

Cumulative success (%)

ACE-Online	74.4%
PREPING	**79.4%**
PREPING+ACE	**82.2%**

Our Approach: Self-Practice in New Environments

Desiderata for Effective Self-Practice

Diversity

Repetitive tasks limit the breadth of procedural memory.
Self-practice should cover diverse tools and workflows.

Feasibility

Infeasible tasks can yield misleading procedural memory.
Use known states and constraints to avoid infeasible tasks.

PREPING: Control What to Practice and What to Store

What to Practice?

Generated tasks can be **repetitive** or **infeasible**, limiting the usefulness of the resulting memory.

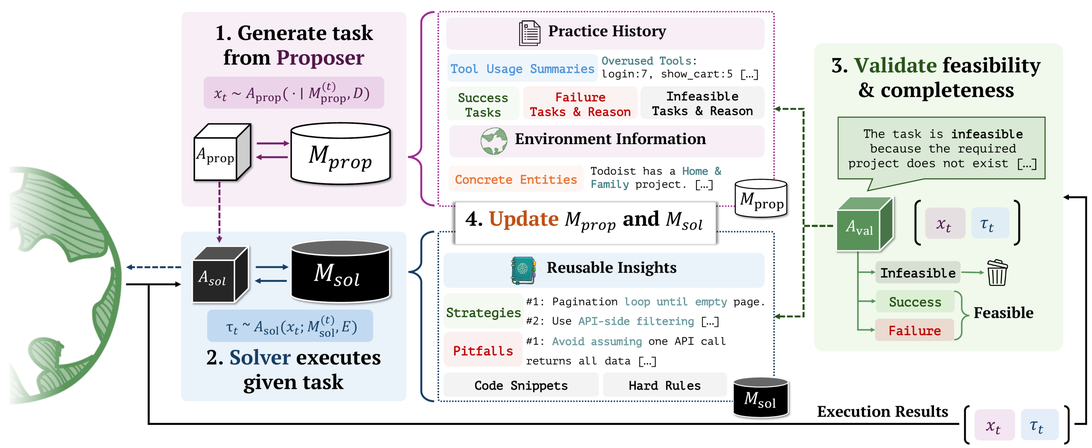
$$M_{\text{prop}}^{(t+1)} = U_{\text{prop}}(M_{\text{prop}}^{(t)}, x_t, \tau_t, v_t),$$

Proposer memory uses prior **practice history** and **environment information** to condition the proposer and reshape the task distribution.

What to Store?

$$M_{\text{sol}}^{(t+1)} = \begin{cases} U_{\text{sol}}(M_{\text{sol}}^{(t)}, x_t, \tau_t, v_t), & \text{if Feasible}(v_t), \\ M_{\text{sol}}^{(t)}, & \text{otherwise.} \end{cases}$$

The Validator checks **task feasibility** and prevents **misleading insights** from infeasible tasks before they enter solver memory as guidance.

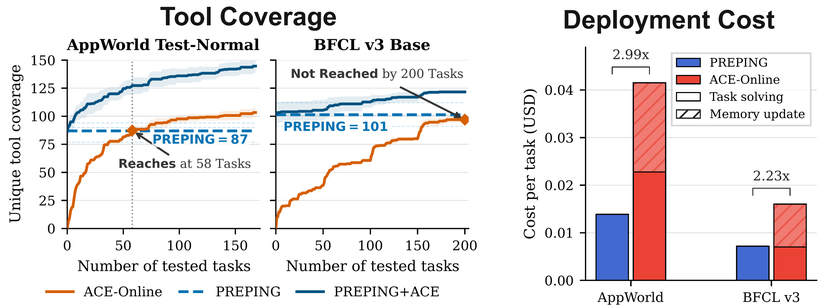


Main Results

Method	AppWorld	BFCL v3	MCP-Universe	Avg.
Pre-Task Methods				
Base	53.1	41.9	32.1	42.4
Direct Memory	54.7	42.8	32.8	43.4
Random Exploration	54.7	55.2	33.1	47.7
Guided Exploration	55.6	53.4	34.6	47.9
PREPING	**70.2**	**62.3**	**37.5**	**56.7**
Task-Informed Methods				
ACE-Online	71.3	58.7	41.2	57.1
PREPING+ACE	**76.3**	**64.9**	–	–

PREPING builds effective procedural memory without human tasks.
It outperforms exploration baselines and further improves ACE-Online.

Analysis



Self-practice covers a broad range of tools than the coverage gained in test-time.
PREPING reduces deployment cost per task compared to online memory.

Ablation: Balancing Diversity and Feasibility

Validator	Env. Info	Practice History	Appworld TGC ↑	Infeasible Tasks (%) ↓	Task Diversity Metrics: Unique Tools ↑	Tool Entropy ↑	Weighted Recall ↑
✗	✗	✗	47.8	—	65.7	5.611	0.653
✓	✗	✗	**78.2**	26.3	69.0	5.586	0.632
✓	✗	✓	81.5	**33.3**	**81.7**	**5.775**	**0.674**
✓	✓	✗	80.3	**5.3**	**42.3**	4.711	**0.368**
✓	✓	✓	**83.7**	**16.0**	**87.0**	**5.919**	**0.703**

Useful memory depends on filtering out infeasible tasks and covering diverse workflows.

Validator controls memory quality

Coverage alone does not ensure useful memory.
Validation raises TGC from 47.8% to 78.2%.

History expands beyond familiar tools

unique tool coverage improves: 69.0 → 81.7,
but infeasibility also rises: 26.3% → 33.3%

Environment info improves feasibility

Infeasibility drops: 26.3% → 5.3%,
but coverage narrows: 69.0 → 42.3 tools