

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分



Memory and Skills

How agents improve without starting over

Carnegie Mellon University
Language Technologies Institute

Daniel Fried
11-768: AI Agents

智能体如何在不从头开始的情况下持续改进

Daniel Fried

11-768: AI 智能体

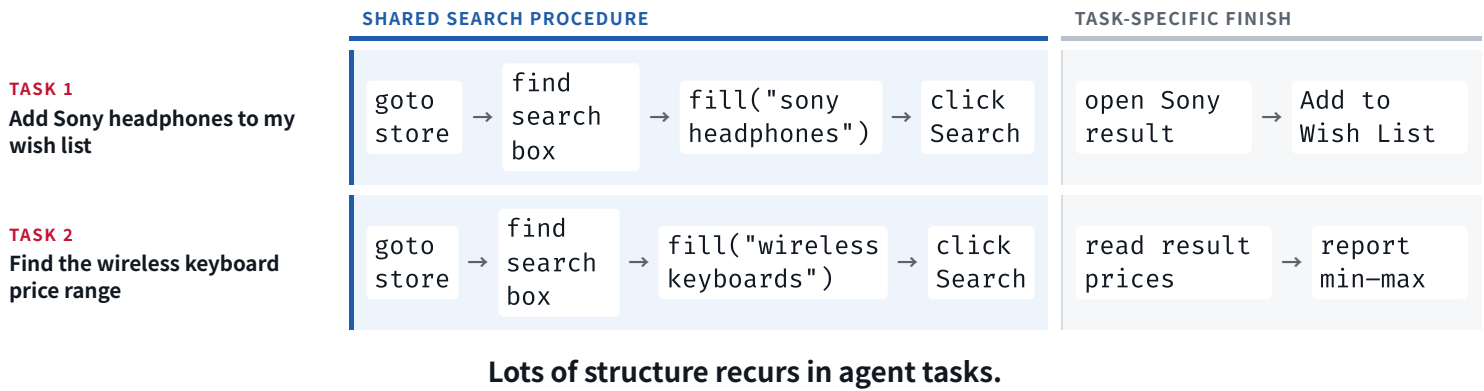
记忆与技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Shared task structure



TODAY'S QUESTIONS:

REPRESENTATION Should the shared structure be text or code?	INDUCTION How can the agent induce it from experience?	LIFECYCLE When should a skill be retrieved, checked, revised, or forgotten?
---	--	---

任务 1：将 Sony 耳机添加到我的愿望清单。共享搜索过程：进入商店 → 找到搜索框 → 填入 `sony headphones` → 点击 Search → 打开 Sony 耳机结果。任务特定的收尾：添加到愿望清单。

任务 2：查找无线键盘的价格区间。共享搜索过程：进入商店 → 找到搜索框 → 填入 `wireless keyboards` → 点击 Search → 阅读结果价格。任务特定的收尾：报告最小值至最大值。

智能体任务中有大量结构反复出现。

今天的问题：表示——共享结构应是文本还是代码？归纳——智能体如何从经验中归纳出它？生命周期——何时检索、检查、修订或遗忘技能？

共享任务结构

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Methods of updating the agent

LOCATION	PROS	CONS
Context window LAST LECTURE	high fidelity to what happened	costly, noisy, does not decide what matters
External artifacts THIS LECTURE	inspectable, editable, retrievable, portable	must be induced, selected, and maintained
Model weights SFT LECTURE	faster inference, broad behavior change	slower updates, opaque, model-specific

What experience is worth turning into an external artifact?

3

上下文窗口（上一讲）：高度忠实于实际发生的事情；但成本高、噪声多，也不会自行判断哪些内容重要。

外部制品（本讲）：可检查、可编辑、可检索、可移植；但必须归纳、筛选和维护。

模型权重（SFT 讲）：推理更快、行为改变范围广；但更新更慢、不透明，并依赖具体模型。

哪些经验值得转化为外部制品？

更新智能体的方法

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Types of experience

Episode

A full trace, every observation and action

Useful if the exact product, prices, or sequence will matter again.

Fact

“This store shows prices only on the product page.”

Useful if the fact recurs in future tasks, or can be updated.

Skill

fill the search box → click search → open a result

Useful if later tasks will repeat the same pattern.

情节 (Episode)：完整轨迹，包含每次观测和行动。当确切产品、价格或步骤顺序未来仍有时适合保存。

事实 (Fact)：例如 ‘这家商店只在商品页面显示价格。’ 当事实会在未来任务中重复出现或可以更新时适合保存。

技能 (Skill)：例如 ‘填写搜索框 → 点击搜索 → 打开一个结果。’ 当后续任务会重复同一模式时适合保存。

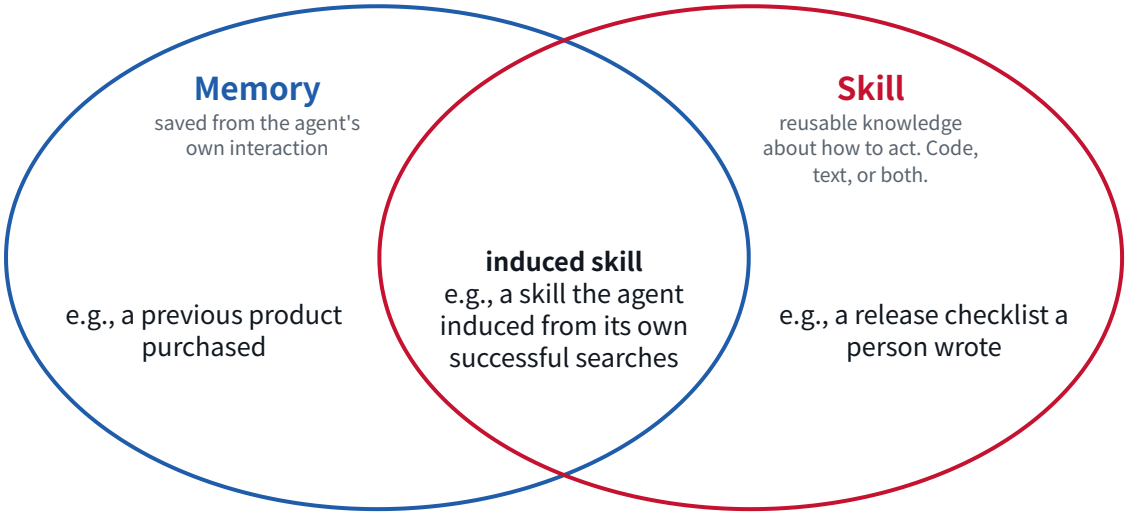
经验类型

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Memory versus skill



5

记忆：来自智能体自身交互中保存的信息。

技能：有关如何行动的可复用知识，可以是代码、文本，或两者兼有。

示例：之前购买过的产品；智能体从成功搜索中归纳的技能；由人编写的发布检查清单。

记忆与技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Skill creation pathways

	SKILLS AS INSTRUCTIONS	SKILLS AS LEARNING
Author	a person or organization	the agent
Source	existing expertise, policy, documentation	its own episodes and outcomes
Strength	known intent and provenance	works where the skill must be discovered
Main burden	expert time to write and maintain	judging outcomes, verification, curation

6

指令式技能 | 学习式技能。

作者：个人或组织 | 智能体。

来源：现有专业知识、策略、文档 | 智能体自身的情节和结果。

优势：意图和出处明确 | 适用于必须通过发现得出技能的场景。

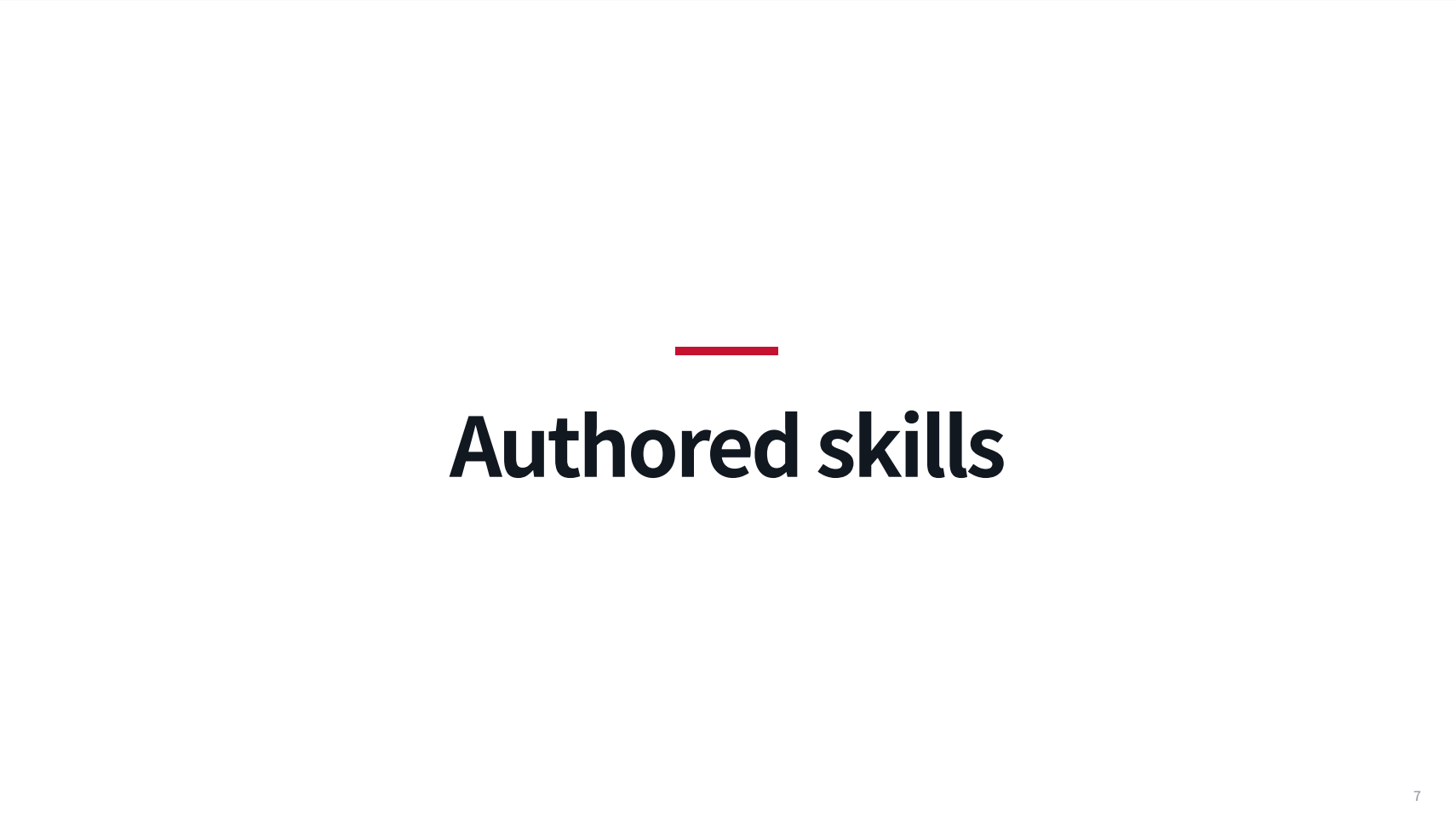
主要负担：专家撰写和维护的时间 | 判断结果、验证和筛选。

技能创建路径

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分



人工编写的技能

人工编写的技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

What is a skill?

THE SAME REQUEST, TYPED INTO EVERY PULL-REQUEST CONVERSATION

Please check this Python code using:

- Black formatting with 88-char line limit
- Ruff linting with our custom rules
- Type hints for all public APIs
- Google-style docstrings
- Pytest for all new functions

- **When you want to do something repeatedly following a fixed spec.** e.g. adding tests, reviewing pull requests, upgrading dependencies
- **Here a person wrote it.** Later in the lecture the agent is asked to notice the repetition and write the skill itself
- **Implemented using a collection of files.** Documentation, scripts, and resources that provide guidance and automation
-

OpenHands · Michelini & Neubig, “How to Create Effective Agent Skills”, 2026 ↗

8

同一请求被反复写入每次 Pull Request 对话：请按以下要求检查这段 Python 代码：使用 Black 格式化，行长限制为 88 个字符；使用带有自定义规则的 Ruff 检查；所有公共 API 都要有类型提示；使用 Google 风格的文档字符串；所有新函数都要有 Pytest 测试。

当你想按照固定规范反复完成某件事时适合使用技能，例如添加测试、审查 Pull Request、升级依赖。

此处技能由人编写。讲座后续会要求智能体发现这种重复，并自行编写技能。

技能由一组文件实现，包括提供指导和自动化的文档、脚本和资源。

OpenHands · Michelini 与 Neubig, 《如何创建有效的智能体技能》, 2026 ↗

什么是技能？

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Example of an authored skill

PACKAGE

python-review/

SKILL.md

scripts/

references/

assets/

Required: instructions + metadata

Optional: executable code

Optional: documentation

Optional: templates, resources

SKILL.MD

name: python-review

description: This skill should be used when the user asks to "review Python code", "check Python style", "lint Python", or mentions code quality.

triggers:

- python review

- code review

- lint python

Python Code Review

Review Python code using company standards...

LEVEL	FILE	CONTEXT WINDOW
1	SKILL.md metadata (YAML)	Always loaded
2	SKILL.md body (Markdown)	Loaded when the agent triggers the skill
3	Bundled files (text, scripts, data)	Loaded when the agent reads the files

Progressive disclosure: Names and descriptions load up front; the rest is read only when the agent calls for it.

Example: OpenHands, “How to Create Effective Agent Skills” ↗

Table: Anthropic, “Equipping agents for the real world with Agent Skills” ↗

AgentSkills specification ↗

9

包结构 `python-review/`：必需的 `SKILL.md`（指令 + 元数据）；可选的 `scripts/`（可执行代码）、`references/`（文档）、`assets/`（模板、资源）。

`SKILL.md` 示例元数据：名称 `python-review`；描述：用户要求 ‘审查 Python 代码’、 ‘检查 Python 风格’、 ‘检查 Python lint’ 或提及代码质量时使用；触发词包括 python review、code review、lint python。正文标题为 Python Code Review，并说明按公司标准审查 Python 代码。

层级 1：`SKILL.md` 元数据 (YAML)，始终加载。层级 2：`SKILL.md` 正文（Markdown），智能体触发技能时加载。层级 3：捆绑文件（文本、脚本、数据），智能体读取文件时加载。

渐进式披露：名称和描述先行加载；其余内容仅在智能体调用时读取。

示例：OpenHands，《如何创建有效的智能体技能》↗；表格：Anthropic，《为真实世界配备智能体技能》↗；AgentSkills 规范 ↗

人工编写技能的示例

个人学习译文 · 非 CMU 官方译本 · 译注独立标记 · 原始课件来自 cmu-agents.com

双语 PDF 9 | 原页 9

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Progressive Disclosure: Indexing skills

```
SYSTEM PROMPT

## Skills
Before replying, scan the skills below. If a skill matches or is even
partially relevant to your task, you MUST load it with skill_view(name)
and follow its instructions. ...

<available_skills>
  code-quality:
    - python-review: This skill should be used when the user asks to
      "review Python code", "check Python style", "lint Python", or
      mentions code quality.
  workflow:
    - submit-task: When the task is complete, take the required steps to
      submit the solution. DO NOT submit without invoking this skill.
    - release-notes: ...
</available_skills>

Only proceed without loading a skill if genuinely none are relevant.
```

系统提示示例：回复前，扫描下面的技能。如果有技能与任务匹配或部分相关，必须通过 `skill\_view(name)` 加载并遵循其指令.....

可用技能索引：`code-quality` 下的 `python-review`：用户要求 ‘审查 Python 代码’ 、 ‘检查 Python 风格’ 、 ‘检查 Python lint’ 或提及代码质量时使用。
`workflow` 下的 `submit-task`：任务完成后执行要求的步骤；未调用该技能不得提交。另有 `release-notes`.....

只有确实没有相关技能时，才可以不加载技能而继续。

摘编自 HermesAgent · `agent/prompt\_builder.py` (MIT) ↗

渐进式披露：技能索引

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Progressive Disclosure: loading skills with tools

TURN	WHAT ENTERS THE CONTEXT
1 System Prompt	the index of skills: names and descriptions only <i>≈3k tokens in Hermes</i>
2 skill_view('python-review')	the SKILL.md body, as a tool result : the review steps, as text <i>once; invoke_skill in Assignment 1</i>
3 TERMINAL python scripts/run_checks.py src/	the script's output : Black and Ruff findings <i>the code itself never enters</i>
4 skill_view('python-review', 'references/docstring-style.md')	the docstring rules <i>only when the model calls for them</i>

[HermesAgent · tools/skills_tool.py ↗](#)

[Assignment 1 · § 1.4 Load skills ↗](#)

[Anthropic, Agent Skills ↗](#)

第 1 步，系统提示：将技能索引（仅名称和描述）放入上下文；Hermes 中约 3K 词元。

第 2 步，`skill\_view('python-review')`：以工具结果形式加入 `SKILL.md` 正文，即审查步骤；只加载一次。作业 1 中使用 `invoke\_skill`。

第 3 步，终端执行 `python scripts/run\_checks.py src/`：只将 Black 和 Ruff 的检查结果放入上下文，代码本身不进入上下文。

第 4 步，`skill\_view('python-review', 'references/docstring-style.md')`：只有模型需要时才加入文档字符串规则。

HermesAgent · `tools/skills\_tool.py` ↗；作业 1 · §1.4 加载技能 ↗；Anthropic · Agent Skills ↗

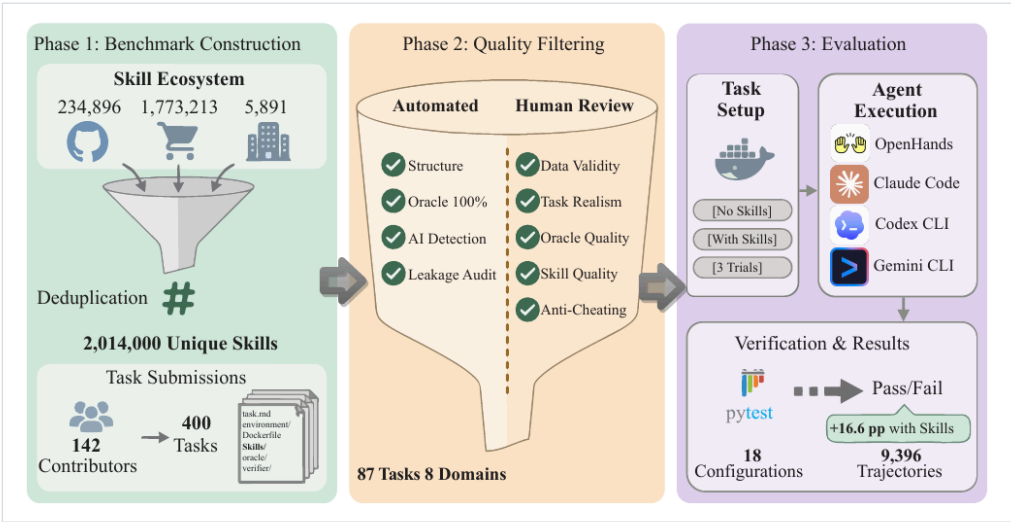
渐进式披露：使用工具加载技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Measuring improvements from skills



- **SkillsBench: 87 real-work tasks across 8 domains.**
- **Curated task-skill pairs.** Humans choose the skill bundle supplied with each task
- **Main experiment: no skills vs. curated skills.** 18 model-harness configurations × 87 tasks.

Figure: SkillsBench · [arXiv 2602.12670](#)

12

阶段 1：基准构建。技能生态系统：234,896；1,773,213；5,891。去重后得到 2,014,000 个独特技能。任务提交：142 位贡献者、400 项任务。

阶段 2：质量筛选。自动检查：结构、Oracle 100%、AI 检测、泄漏审计。人工审查：数据有效性、任务真实性、Oracle 质量、技能质量、反作弊。共 87 项任务、8 个领域。

阶段 3：评估。任务设置：不提供技能、提供技能、3 次试验。智能体执行：OpenHands、Claude Code、Codex CLI、Gemini CLI。验证与结果：pytest、通过 / 失败、使用技能增加 16.6 个百分点；18 种配置、9,396 条轨迹。

SkillsBench：来自 8 个领域的 87 项真实工作任务。策划任务与技能的配对；由人选择为每个任务提供的技能包。主要实验：不提供技能与提供策划技能的对比；18 种模型—执行框架配置 × 87 项任务。

图：SkillsBench · [arXiv 2602.12670](#)

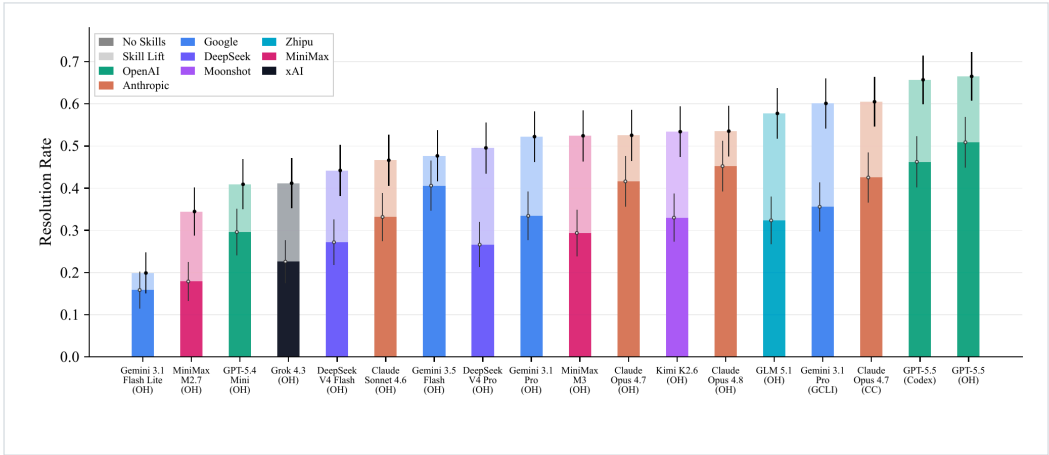
衡量技能带来的改进

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Gains and limits of human-authored skills



Skills raise the average pass rate from 33.9% to 50.5%.

- **Focused skills correlate with larger gains.** Tasks with 1–3 skills improve more than tasks ≥ 4 skills; shorter SKILL.md groups also show larger gains.
- **Skills still *hurt* on 13 of 87 tasks.** A skill can displace a better default strategy.

SkillsBench · arXiv 2602.12670 ↗

技能将平均通过率从 33.9% 提高到 50.5%。图例：No Skills、Skill Lift、OpenAI、Anthropic、Google、DeepSeek、Moonshot、Zhipu、MiniMax、xAI；纵轴为解决率。横轴模型包括 Gemini 3.1 Flash Lite、MiniMax M2.7、GPT-5.4 Mini、Grok 4.3、DeepSeek V4 Flash、Claude Sonnet 4.6、Gemini 3.5 Flash、DeepSeek V4 Pro、Gemini 3.1 Pro、MiniMax M3、Claude Opus 4.7、Kimi K2.6、Claude Opus 4.8、GLM 5.1、Gemini 3.1 Pro、Claude Opus 4.7、GPT-5.5。

针对性强的技能与更大的收益相关。只提供 1–3 项技能的任务比提供 ≥ 4 项技能的任务改善更多；较短的 SKILL.md 组合也显示出更大收益。

在 87 项任务中的 13 项上，技能仍会造成负面影响。技能可能取代更好的默认策略。

SkillsBench · arXiv 2602.12670 ↗

人工编写技能的收益与局限

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Sources of skill packages

Bundled

Ships with the harness as part of its default set.

Installed or written

A person writes one for their own work, checks it into the repository under `.agents/skills/`, or installs it from another repository or a registry.

Created by the agent

The agent writes the same package itself. Hermes exposes a `skill_manage` tool that can create, patch, or delete; OpenHands has a `/skill-creator` command that drafts one from the workflow the agent just finished. A person can review either before it ships.

OpenHands' advice: don't write a skill from scratch. Create it right after the agent finishes a workflow you want to repeat, then test it a few times.

[HermesAgent skills documentation ↗](#)

[OpenHands, “How to Create Effective Agent Skills” ↗](#)

14

随包提供：作为执行框架默认技能集的一部分。

由人安装或编写：个人为自己的工作编写技能，将其提交到仓库的 `.agents/skills/` 下，或从其他仓库或注册表安装。

由智能体创建：智能体自行编写相同的技能包。Hermes 提供 `skill_manage` 工具，可创建、修补或删除技能；OpenHands 提供 `/skill-creator` 命令，根据智能体刚完成的工作流起草技能。技能发布前，人可以审查这两种方式产生的技能。

OpenHands 的建议：不要从零开始写技能。在智能体完成你想复用的工作流后立即创建，然后测试几次。

HermesAgent 技能文档 ↗；OpenHands, 《如何创建有效的智能体技能》↗

技能包的来源

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Case study: maintaining a code-review skill

OpenHands runs a review skill of the `python-review` kind on every pull request in its own repos.

MAINTAIN review skill → log each run → score each run → find recurring failures → patch SKILL.md

- **Evaluate the outputs.** After each PR merges, a model judge counts how many of the agent's suggestions the humans kept.
- **Have a model analyze failures.** It reads the run notes and names what recurs: ignoring repo conventions (~15%) and approving PRs with a critical flaw (~10%).
- **Update** SKILL.md . The same model drafts it: request changes whenever a critical issue is found, and check a suggestion against the repo before posting it.

OpenHands · Micheline & Neubig, "How to Create Effective Agent Skills", 2026 ↗

15

OpenHands 在自己的仓库中对每个 Pull Request 都运行 `python-review` 类型的审查技能。

维护流程：审查技能 → 记录每次运行 → 为每次运行评分 → 找出反复出现的失败 → 修补 `SKILL.md`。

评估输出：每个 PR 合并后，由模型评审器统计人类保留了多少条智能体建议。

让模型分析失败：读取运行记录并找出复现的问题，例如忽略仓库约定（约 15%）以及在存在严重缺陷时仍批准 PR（约 10%）。

更新 `SKILL.md`：同一模型起草修改；发现严重问题时要求提出修改；发布建议前对照仓库检查。

OpenHands · Micheline 与 Neubig, 《如何创建有效的智能体技能》, 2026 ↗

案例研究：维护代码审查技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

记忆：事实、教训与情节

Memories: facts, lessons, and episodes

16

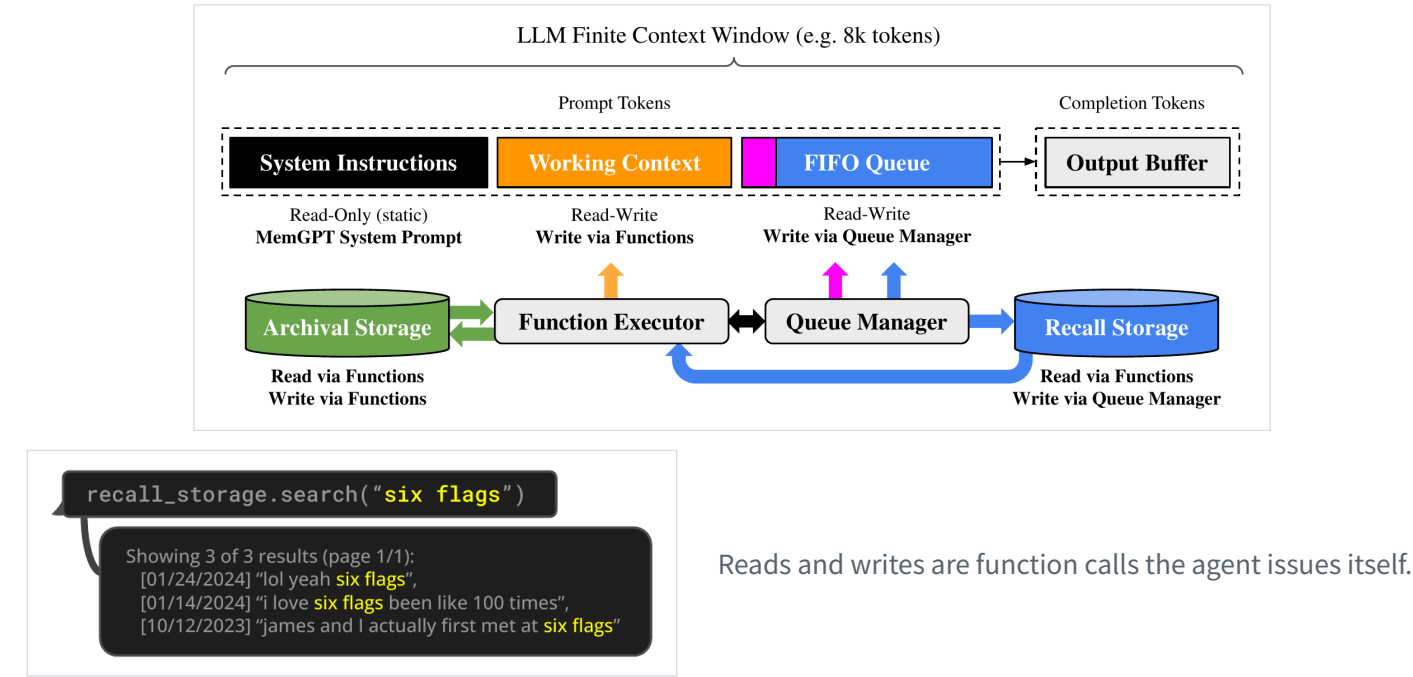
记忆：事实、教训与情节

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Managing an external memory



Figures 3 and 2: MemGPT · Packer et al., 2023 ↗

LLM 有限上下文窗口（例如 8K 词元）由提示词元和补全词元组成。提示区包含系统指令（只读、静态、由 MemGPT 系统提示提供）、工作上下文（可读写，通过函数写入）、FIFO 队列（可读写，由队列管理器写入）；补全区为输出缓冲区。

归档存储：通过函数读取和写入。函数执行器与队列管理器进行交互。回忆存储：通过函数读取，由队列管理器写入。

查询示例：`recall\_storage.search("six flags")`。显示 3 条结果（第 1/1 页）：
[01/24/2024] “I love six flags” ； [01/14/2024] “i love six flags been like 100 times” ； [10/12/2023] “james and I actually first met at six flags” 。

读取和写入都是智能体自行发出的函数调用。

图 3 和图 2：MemGPT · Packer 等, 2023 ↗

管理外部记忆

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Handling memory conflicts

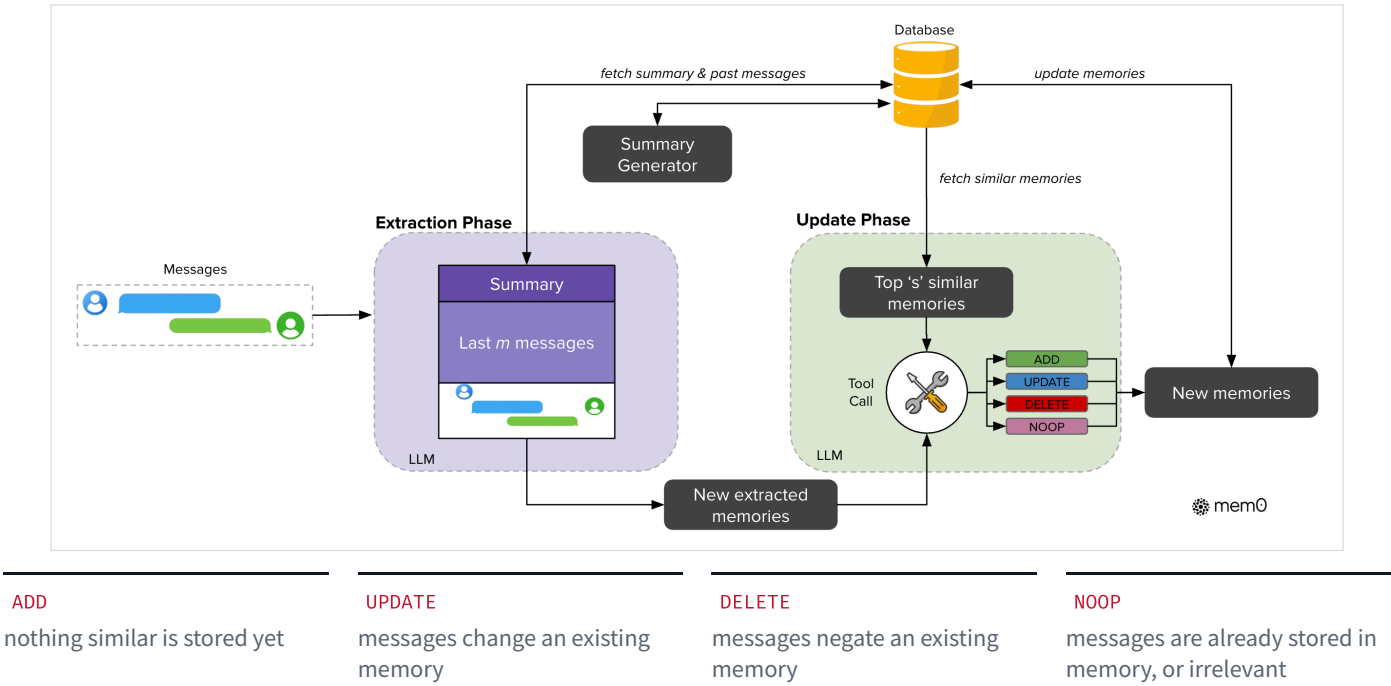


Figure: Mem0 · Chhikara et al., 2025 ↗

添加 (ADD)：尚未存储相似内容。

更新 (UPDATE)：新消息改变了现有记忆。

删除 (DELETE)：新消息否定了现有记忆。

不操作 (NOOP)：新消息已存储在记忆中，或与记忆无关。

图：Mem0 · Chhikara 等, 2025 ↗

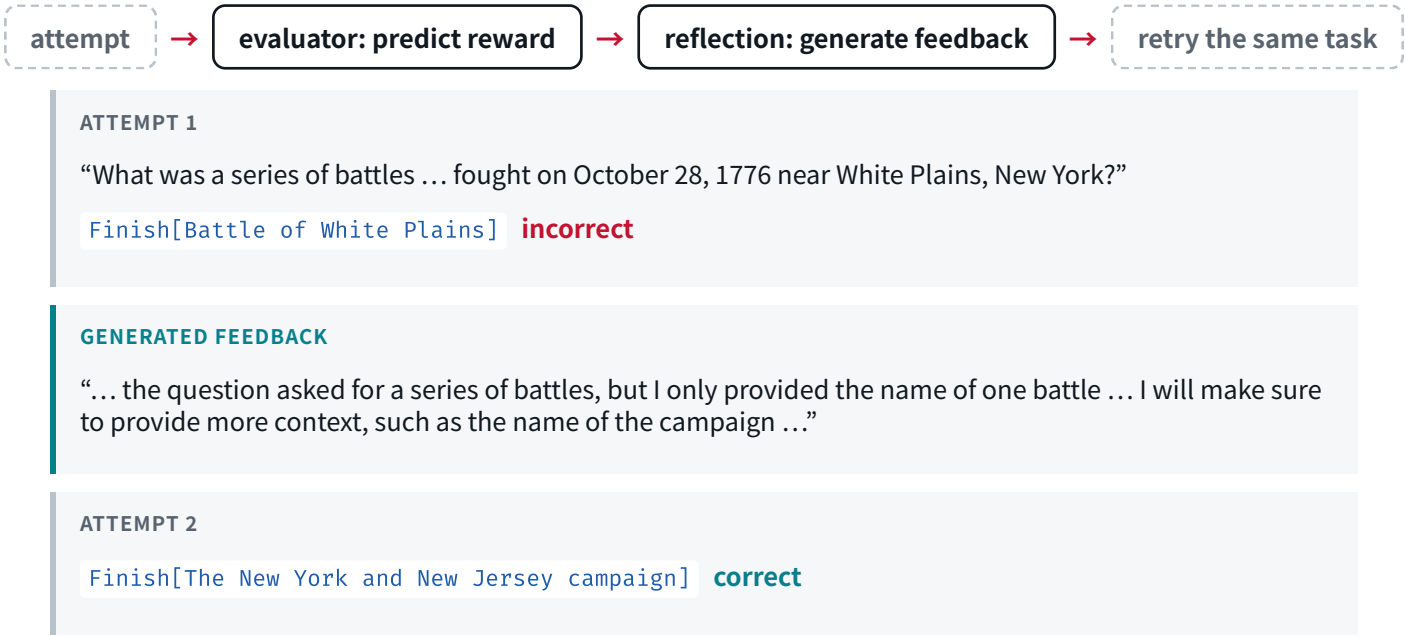
处理记忆冲突

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Remembering feedback on trajectories



Reflexion · Shinn et al., NeurIPS 2023 ↗

19

尝试任务 → 评估器预测奖励 → 反思生成反馈 → 重试同一任务。

尝试 1： ‘一系列战役是什么.....这些战役于 1776 年 10 月 28 日在纽约州怀特普莱恩斯附近发生？ ’ 回答 `Finish[Battle of White Plains]`： 错误。

生成的反馈： ‘.....问题询问的是一系列战役，但我只给出了一场战役的名称.....我会确保提供更多背景，例如战役的名称.....’

尝试 2： 回答 `Finish[The New York and New Jersey campaign]`： 正确。

Reflexion · Shinn 等, NeurIPS 2023 ↗

记住对轨迹的反馈

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Similar-trajectory conditioning

- **What is stored:** whole trajectories, both successes and failures, sometimes rewritten and annotated before storage.
- **Where the pool comes from:** collected offline in a training phase, grown while the agent works, and/or hand-written in advance.
- **How it is used:** retrieve the runs most similar to the new task using text embeddings, and include them as few-shot examples.

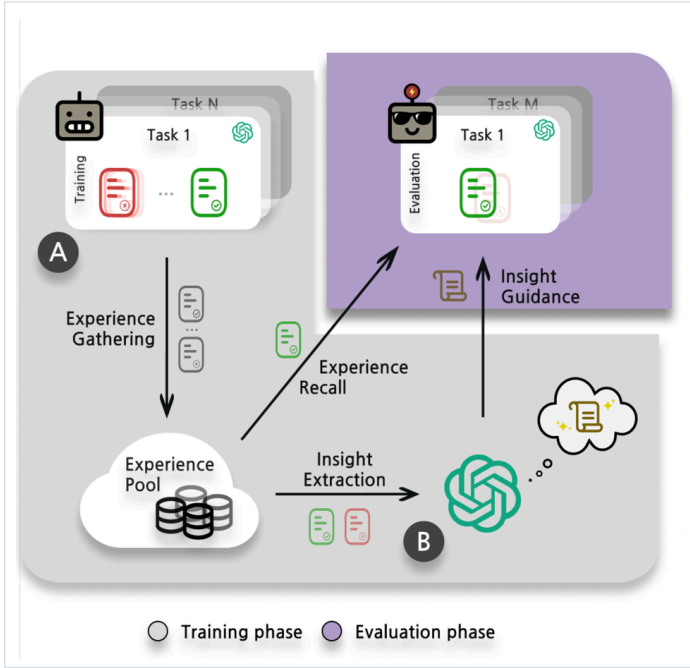


Figure from ExpeL, Zhao et al. 2024

[ExpeL · Zhao et al., AAAI 2024 ↗](#)

[Agent S · Agashe et al., ICLR 2025 ↗](#)

[Synapse · Zheng et al., ICLR 2024 ↗](#)

[ICAL · Sarch et al., NeurIPS 2024 ↗](#)

保存内容：完整轨迹，包括成功和失败；有时在保存前会重写并添加注释。

记忆池来源：在训练阶段离线收集；智能体工作时逐步积累；和 / 或由人预先编写。

使用方式：通过文本嵌入检索与新任务最相似的运行记录，并将其作为少样本示例。

图：ExpeL · Zhao 等, 2024。ExpeL · Zhao 等, AAAI 2024 ↗；Agent S · Agashe 等, ICLR 2025 ↗；Synapse · Zheng 等, ICLR 2024 ↗；ICAL · Sarch 等, NeurIPS 2024 ↗

基于相似轨迹进行条件化

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分



归纳技能

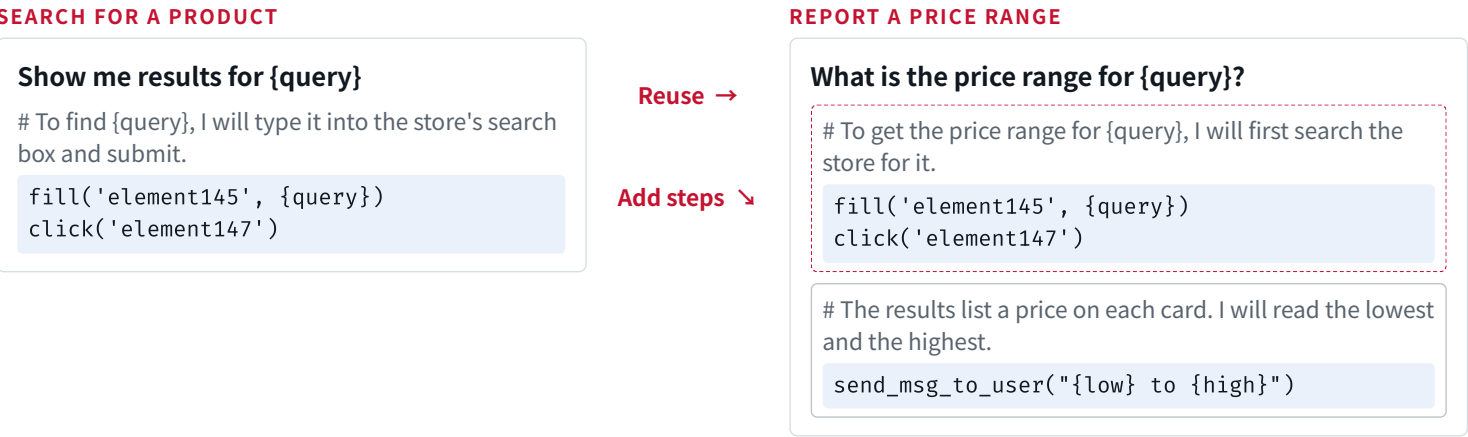
归纳技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Skills as reusable chunks of a task



How should a skill be represented? We'll compare text vs code.

22

搜索商品： ‘显示 {query} 的搜索结果。’

步骤说明： ‘要查找 {query}，我会在商店搜索框中输入并提交。’ 调用 `fill('element145', {query})`，再调用 `click('element147')`。

复用 → 报告价格区间： ‘{query} 的价格区间是多少？’ 在商店搜索商品，然后读取每张卡片上的价格，报告最低价和最高价： `send\_msg\_to\_user("{low} to {high}")`。

追加步骤：填写 `element145` 并点击 `element147`。

技能应如何表示？ 接下来比较文本和代码。

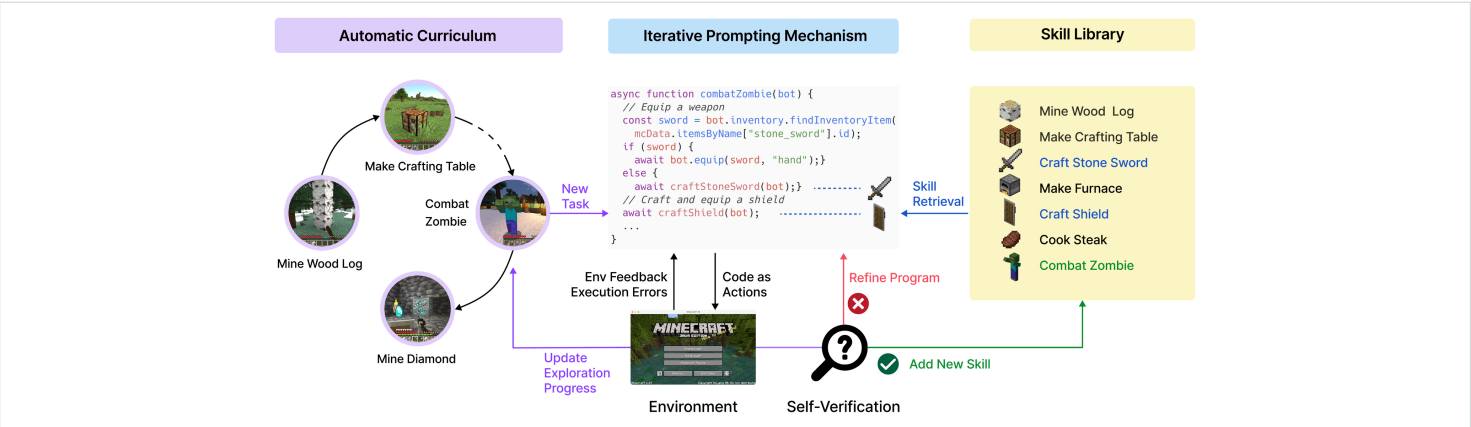
作为任务可复用片段的技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Skills / program induction in past work



- **Voyager.** Induce functions for increasingly complex action sequences in Minecraft, each calling earlier ones.
- **DreamCoder.** Induce functions that compress a corpus of solved programs; later search reuses them.
- **Stitch.** Fast symbolic compression: find functions that capture the most shared structure in a corpus.
- **LAPS.** Natural-language task descriptions guide which functions are induced and how programs are searched for.
- **LILO.** An LLM writes programs; Stitch compresses them into functions; the LLM names and documents each.

Figure: Voyager · Wang et al., 2023 ↗ DreamCoder · Ellis et al., 2021 ↗ Stitch · Bowers et al., 2023 ↗ LAPS · Wong et al., 2021 ↗ LILO · Grand et al., 2024 ↗

Voyager: 在 Minecraft 中归纳越来越复杂的行动序列函数，每个函数都调用早先归纳的函数。

DreamCoder: 归纳能够压缩一组已解决程序的函数；后续搜索会复用它们。

Stitch: 快速符号压缩，寻找能捕获语料中最多共享结构的函数。

LAPS: 自然语言任务描述指导要归纳哪些函数，以及如何搜索程序。

LILO: LLM 编写程序；Stitch 将其压缩为函数；LLM 为每个函数命名并编写文档。

图: Voyager · Wang 等, 2023 ↗; DreamCoder · Ellis 等, 2021 ↗; Stitch · Bowers 等, 2023 ↗; LAPS · Wong 等, 2021 ↗; LILO · Grand 等, 2024 ↗

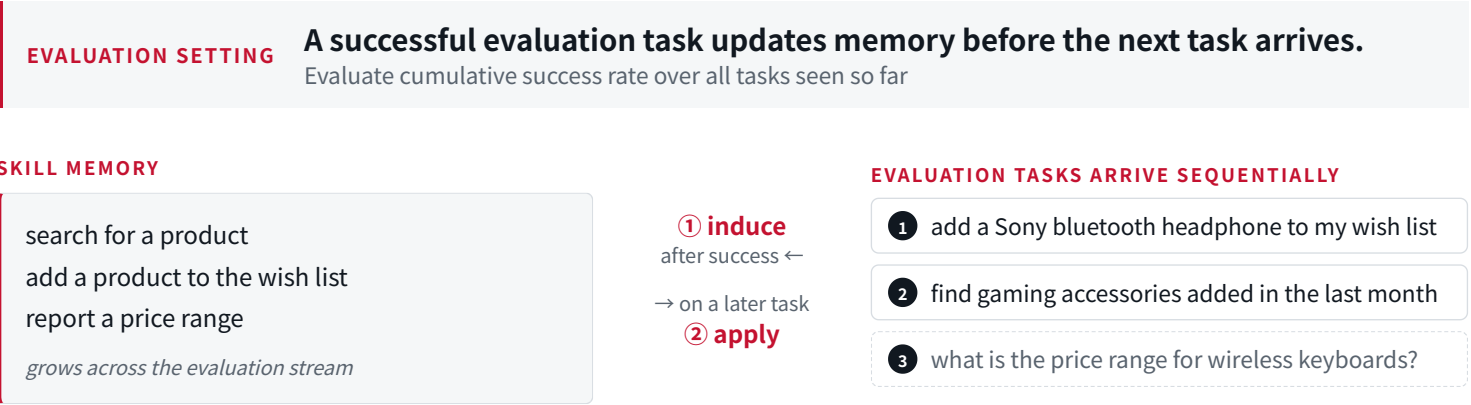
以往研究中的技能 / 程序归纳

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Inducing skills online



24

评估设置：一项成功的评估任务会在下一项任务到来之前更新记忆；评估截至当前所见所有任务的累计成功率。

技能记忆：搜索商品；将商品加入愿望清单；报告价格区间；技能会随着评估任务流不断增长。

评估任务按顺序到来：1. 将 Sony 蓝牙耳机加入愿望清单；2. 查找上个月添加的游戏配件；3. 无线键盘的价格区间是多少？

成功后归纳；稍后的任务可以复用；② 应用。

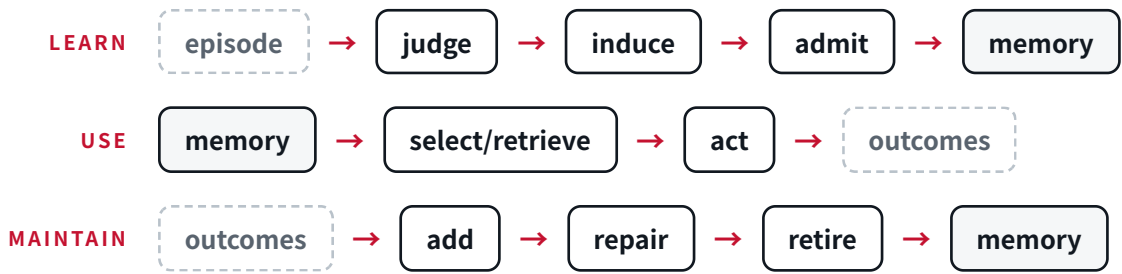
在线归纳技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

The lifecycle of an induced skill



25

学习：情节 → 评判 → 归纳 → 准入 → 记忆。

使用：记忆 → 选择 / 检索 → 行动 → 结果。

维护：结果 → 添加 → 修复 → 退役 → 记忆。

归纳技能的生命周期

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Inducing textual representations of skills

LEARN

judge → induce → admit

episode

NL query

Tell me the number of reviews that our store received so far that mention the term “satisfied”

+

Actions

click(“Marketing”)

click(“All Reviews”)

Memory

fill(757, “satisfied”)

click(“Search”)

send_msg_to_user(“2”)

Judge

Final state predicted to be correct?

YES ▶ NO ▼ pass

Few-Shot Prompting

LM

Induce

Task: Search for reviews matching “{term}”

Actions:

To find reviews for {term}, I will search for “{term}” in the reviews page

click(“All reviews”)

fill(757, “{term}”)

click(“Search”)

Memory

Agent Workflow Memory · Wang et al., ICLR 2025 ↗

26

Zora Wang

学习阶段：评判 → 归纳 → 准入。

自然语言查询： ‘告诉我到目前为止，本店收到的评论中有多少条提到 “满意” 一词？ ’

情节行动： 点击 `Marketing`； 点击 `All Reviews`； 在 `757` 处填写 `satisfied`； 点击 `Search`； 调用 `send\_msg\_to\_user("2")`。

少样本提示下，语言模型归纳出的技能： 任务——搜索包含 “{term}” 的评论； 行动——在评论页面搜索 “{term}” ， 点击 `All reviews`， 填写 `757` 处的 `{term}` ， 点击 `Search`。

评判： 预测的最终状态是否正确？ 是 / 否； 通过。

Agent Workflow Memory · Wang 等, ICLR 2025 ↗

归纳技能的文本表示

个人学习译文 · 非 CMU 官方译本 · 译注独立标记 · 原始课件来自 cmu-agents.com

双语 PDF 26 | 原页 26

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

An induction prompt



INDUCTION PROMPT FROM AGENT WORKFLOW MEMORY

Given a list of web navigation tasks, your task is to extract the common workflows.

Each given task contains a natural language instruction, and a series of actions to solve the task. You need to find the repetitive subset of actions across multiple tasks, and extract each of them out as a workflow.

Each workflow should be a commonly reused sub-routine of the tasks. Do not generate similar or overlapping workflows. Each workflow should have at least two steps. Represent the non-fixed elements (input text, button strings) with descriptive variable names as shown in the example.

Agent Workflow Memory 的归纳提示：给定一组网页导航任务，请提取它们的共同工作流。

每个任务包含自然语言指令和一系列解决任务的行动。你需要找到多个任务中重复的行动子集，并将每个子集提取为工作流。

每个工作流应是任务中常复用的子程序。不要生成相似或重叠的工作流。每个工作流至少包含两个步骤。按照示例，使用描述性变量名表示非固定元素（输入文本、按钮文本）。

AWM · Wang 等, ICLR 2025——附录 A.1 ↗

归纳提示

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Cross-task memory compounding

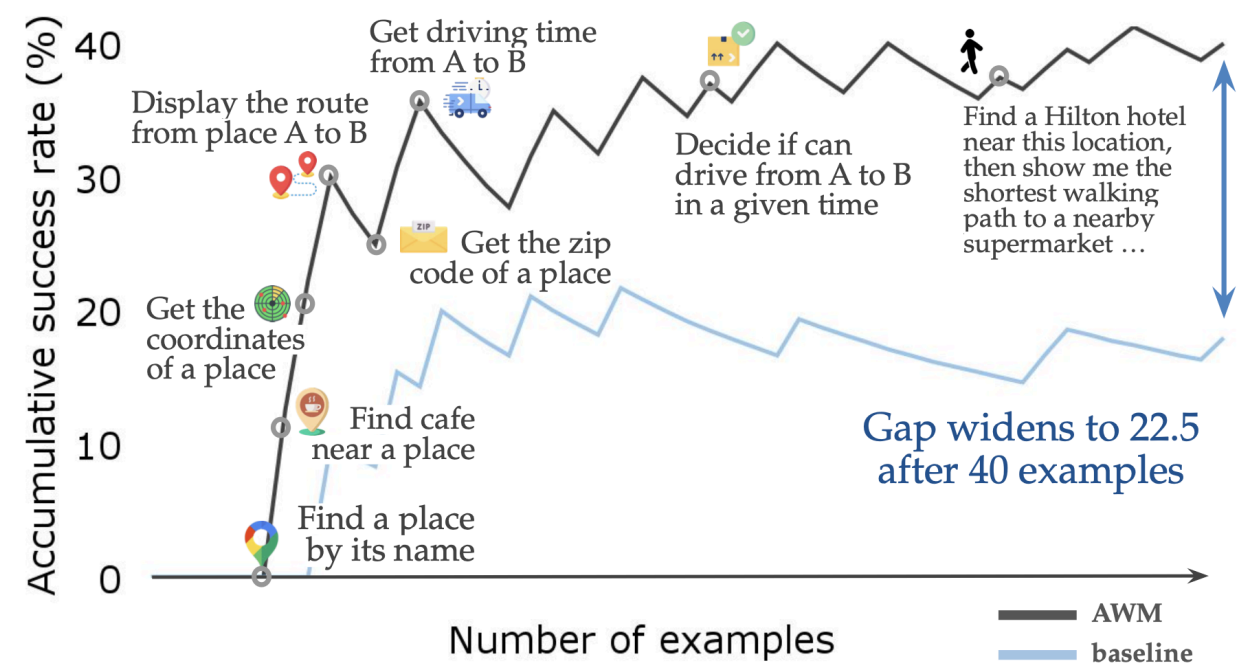


Figure: AWM · Wang et al., ICLR 2025 ↗

28

图表：跨任务记忆复利。横轴为样例数量，纵轴为累计成功率（%）；对比 AWM（深色线）与基线（浅色线）。

任务示例沿着路线规划能力逐步扩展：按地点名称查找地点；查找附近的咖啡馆；获取地点坐标；显示 A 到 B 的路线；获取 A 到 B 的驾驶时间；判断给定时间内能否从 A 开到 B；查找该地点附近的 Hilton 酒店，并显示到附近超市的最短步行路线。

40 个样例后，两者差距扩大至 22.5 个百分点。

图：AWM · Wang 等, ICLR 2025 ↗

跨任务记忆复利

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Skills as text vs skills as code

Text + examples

Task: Search for gaming accessories within a date range

Action trajectory:

```
click(1274) # Navigate to the Video Games category
fill(473, {search_terms}) # Enter search terms including product name and year
click(478) # Execute the search
```

- + **Flexible:** a hint the agent interprets and adapts to the page in front of it
- **Manual:** the agent still issues every low-level action itself, copying and modifying

Code

```
def search_product(search_box_id: str, query: str):
    """Search for a product using the search box.
    Args:
        search_box_id: ID of the search input field
        query: Search query string to enter
    Returns:
        None
    Examples:
        search_product('595', 'sony bluetooth headphones')
    """
    click(search_box_id)
    fill(search_box_id, query)
    keyboard_press('Enter')
```

`search_product('595', 'Macbook')` Usable as a tool

- + **Testable:** run it before storing it
- + **Hierarchical:** skills call skills
- + **Efficient:** one call replaces many model steps
- **Rigid:** does exactly what it says, even on a page that has changed

文本 + 示例：任务——在日期范围内搜索游戏配件。行动轨迹：点击 `1274` 进入电子游戏类别；填写 `473` 处的 `{search\_terms}`（包括商品名和年份）；点击 `478` 执行搜索。

文本的优点：灵活，智能体可以理解提示并根据当前页面调整。缺点：人工编写；智能体仍需自行发出每个底层行动，并复制、修改指令。

代码示例 `search\_product(search\_box\_id: str, query: str)`：使用搜索框搜索商品。参数：搜索输入框 ID、要输入的查询字符串；无返回值。示例 `search\_product('595', 'sony bluetooth headphones')`。函数点击搜索框、填写查询并按 Enter；可作为工具使用。

代码的优点：可在保存前运行测试；技能可以分层调用技能；一次调用可替代多个模型步骤，执行高效。缺点：刚性较强；页面变化后仍会机械地执行既定操作。

Workflows · AWM, Wang 等, ICLR 2025 ↗; Skill · ASI, Wang 等, COLM 2025 ↗

技能：文本与代码

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

LEARN judge → induce → admit

episode

NL query
Tell me the number of reviews that our store received so far that mention the term “satisfied”

Actions

- click(“Marketing”)
- click(“All Reviews”)
- fill(757, “satisfied”)
- click(“Search”)
- send_msg_to_user(“2”)

Few-Shot Prompting

LM

Induce

skills

```
def search_reviews(search_box_id, search_button_id, search_term):
    """Search for reviews with a specific term.
    Args: search_box_id: ...
    Examples: search_reviews('757', '704', 'great')"""
    fill(search_box_id, search_term)
    click(search_button_id)

def open_marketing_reviews():
    """Navigate to All Reviews under Marketing.
    Examples: open_marketing_reviews()"""
    click("Marketing")
    click("All Reviews")

open_marketing_reviews()
search_reviews('757', '704', "satisfied")
send_msg_to_user("2")
```

test

Judge

Final state predicted to be correct? Skills used?

YES → NO → pass

Memory

学习阶段：评判 → 归纳 → 准入。自然语言查询：‘告诉我到目前为止，本店收到的评论中有多少条提到“满意”一词？’

情节行动: 点击 `Marketing` 和 `All Reviews`; 填写 `757` 处的 `satisfied`; 点击 `Search`; 向用户发送 `2`。

少样本提示下归纳出 `search\_reviews(search\_box\_id, search\_button\_id, search\_term)`，用于搜索指定词语的评论；另归纳 `open\_marketing\_reviews()`，导航到 Marketing 下的 All Reviews。

先测试技能，再使用`open\_marketing\_reviews()`和`search\_reviews('757','704','satisfied')`；最后`send\_msg\_to\_user("2")`。评判最终状态是否正确、是否使用技能；是 / 否；通过。

改编自 ASI·Wang 等, COLM 2025 ↗; 沿袭关系: Voyager 在 Minecraft 中积累代码技能记忆 ↗

归纳代码技能

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Code skills allow testing



```
CODE SKILL ADMISSION
candidates = propose_skill(task)
episode = practice(candidates, task)

if reward_model(episode.actions, episode.screenshots, task):
    memory.add(candidate)
else:
    candidate = revise(candidate, task)
```

```
EXAMPLE OF LINTING AND TESTING A CODE SKILL
identify_pill(page, imprint="M366", color="White")

WARNING parameter 'color' is defined but never used in the function body

+ if color:
+     await page.get_by_role("group", name="Color and shape (optional)") \
+         .get_by_role("combobox", name="Color (optional)").select_option(color)
```

After SkillWeaver · Zheng et al., 2025 — Figure 1 ↗

31

代码技能准入：`candidates = propose\_skill(task)`；`episode = practice(candidates, task)`。如果 `reward\_model(episode.actions, episode.screenshots, task)` 为真，则 `memory.add(candidate)`；否则修订候选技能。

代码技能的 lint 和测试示例：`identify\_pill(page, imprint="M366", color="White")`。警告：函数定义了参数 `color`，但函数体中从未使用。

修复代码通过 `if color:`，再以 `page.get\_by\_role("group", name="Color and shape (optional)").get\_by\_role("combobox", name="Color (optional)").select\_option(color)` 选择颜色。

改编自 SkillWeaver · Zheng 等, 2025——图 1 ↗

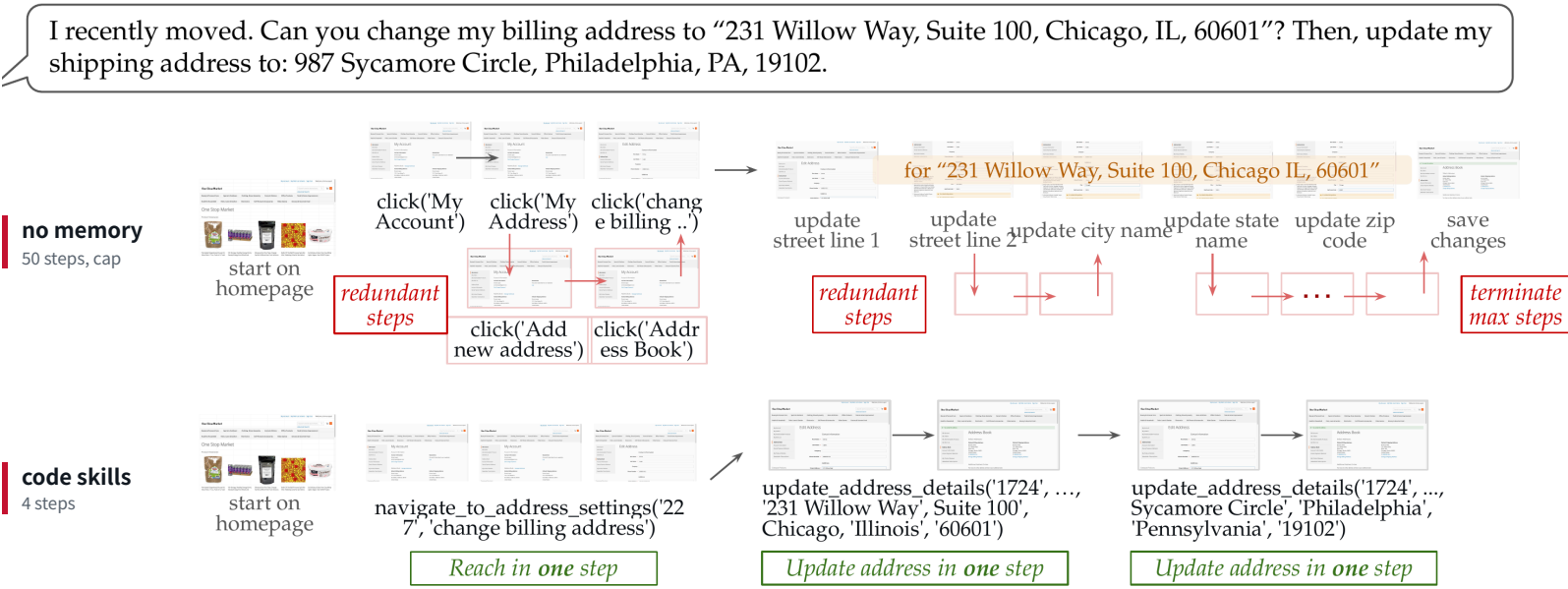
代码技能可供测试

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Task compression with code skills



ASI · Wang et al., COLM 2025 — Figure 5, shopping site ↗

32

无记忆：50 步，达到上限。

代码技能：4 步。

ASI · Wang 等，COLM 2025——图 5，购物网站 ↗

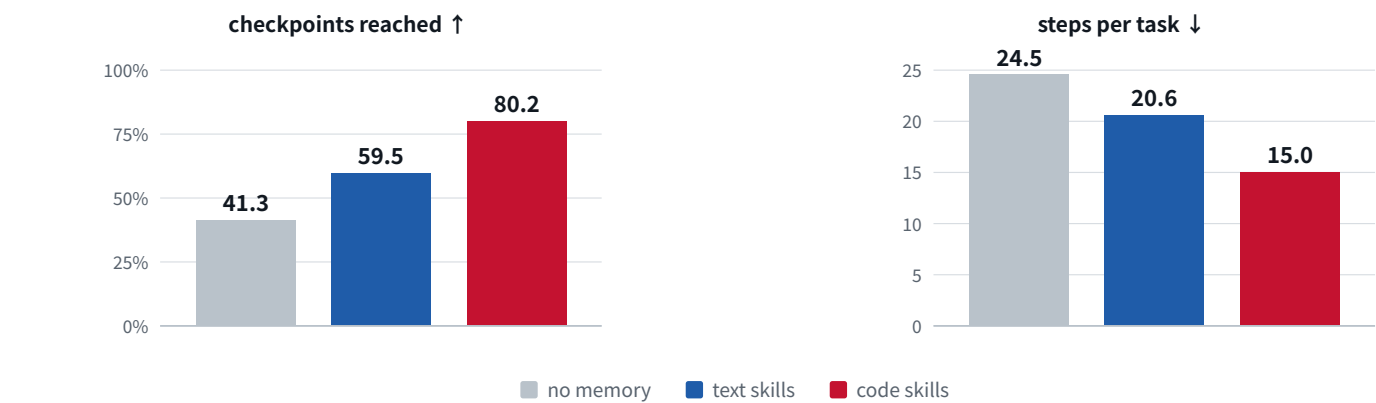
用代码技能压缩任务

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Code skills raise success and cut steps



- Comparison between text and code skills on web tasks with repeated structure.
- Code skills generally improve success (checkpoints reached) over text skills.
- They also use fewer steps by the agent.

ASI · Wang et al., COLM 2025 — Table 2, averaged over five WebArena sites ↗

33

左图指标：达到的检查点（%），右图指标：每项任务的步骤数。对比组为无记忆、文本技能、代码技能。

图中达到的检查点数值：41.3、59.5、80.2；每项任务步骤数：24.5、20.6、15.0。

比较具有重复结构的网页任务中的文本技能和代码技能。总体而言，代码技能相较文本技能提高成功率（达到检查点），也减少智能体执行的步骤。

ASI · Wang 等, COLM 2025——表 2, 五个 WebArena 网站的平均值 ↗

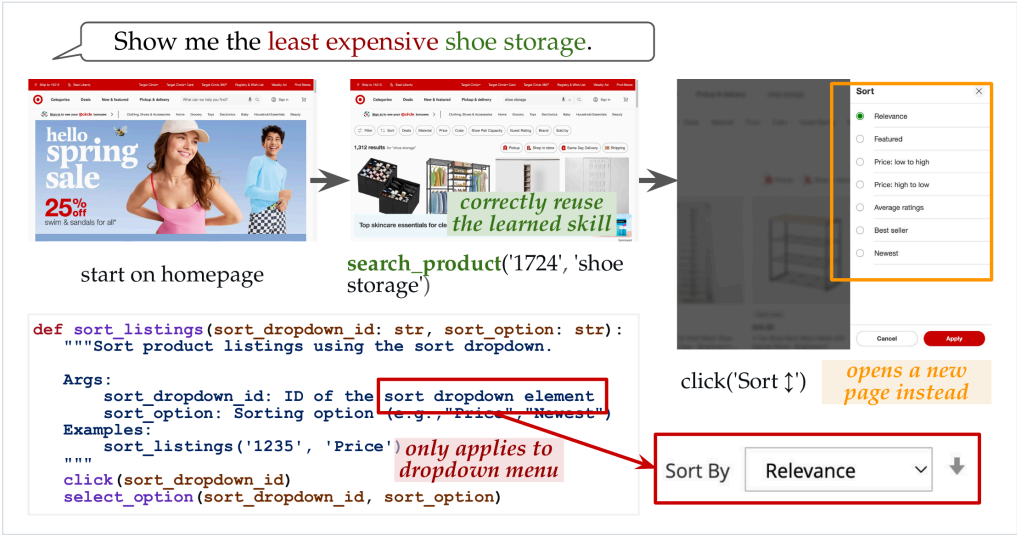
代码技能提高成功率并减少步骤

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

How general are skills?



- Induce skills on tasks on one website; evaluate on another.
- sort_listings clicks a dropdown; Target's sort opens a sidebar.

ASI · Wang et al., COLM 2025 — Figure 4, WebArena skills on target.com ↗

34

示例请求： ‘告诉我最便宜的鞋子收纳产品。’ 从首页开始；在复用已学技能时调用 `search\_product("1724", "shoe storage")`。

技能代码 `sort\_listings(sort\_dropdown\_id: str, sort\_option: str)`： 按排序下拉菜单对商品列表排序；参数为排序下拉元素 ID 和排序选项（例如价格由低到高）； 示例 `sort\_listings("1235", "Price")`； 调用 `click(sort\_dropdown\_id)` 和 `select\_option(sort\_dropdown\_id, sort\_option)`。

跨网站归纳技能并测试： 在 Target 网站中， 点击排序按钮会打开侧边栏， 排序控件与原技能假设的下拉菜单不同； 代码注明 ‘只适用于下拉菜单’ ， 因此不能直接泛化。

在一个网站的任务上归纳技能， 再在另一个网站上评估。 `sort\_listings` 点击下拉菜单； Target 的排序功能则会打开侧边栏。

ASI · Wang 等， COLM 2025——图 4, Target.com 上的 WebArena 技能 ↗

技能的泛化程度如何？

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

PolySkill: one interface, many implementations



PolySkill Abstract Class	PolySkill Implementation
<pre># The Abstract Domain Class, defining a "schema" for shopping sites. class AbstractShoppingSite: def search_product(self, query: str): """Searches for a product.""" def add_to_cart(self, item_id: str, quantity: int): """Adds a specified item to the shopping cart.""" def checkout(self): """Initiates the checkout process.""" ## ----- ## Compositional Skills ## ----- def find_and_add_to_cart(self, query: str, item_id: str): """Searches for a product and adds it to the cart.""" self.search_product(query) self.add_to_cart(item_id) def purchase_item(self, query: str, item_id: str): """Finds a specific product, adds it to the cart, and starts checkout.""" self.find_and_add_to_cart(query, item_id) self.checkout()</pre>	amazon <pre>class AmazonWebsite(AbstractShoppingSite): def search_product(self, query: str): click(search_bar_id) fill(search_bar_id, query) keyboard_press('Enter') def add_to_cart(self, item_id: str, quantity: int): click(add_to_cart_button_id) def checkout(self): """NOTE: Implements checkout as a composite action: first cart, then clicks checkout.""" self.view_cart() click(checkout_button_id)</pre> TARGET <pre>class TargetWebsite(AbstractShoppingSite): def search_product(self, query: str): fill(search_bar_id, query) click(search_button_id) def add_to_cart(self, item_id: str, quantity: int = 1): click(add_button_id) fill(quantity_bar_id, quantity) click(add_to_cart_button_id) def checkout(self): click(checkout_button_id)</pre>

Compositions are written once against the abstract methods; each website implements only the primitive browser actions.

抽象类示例：`AbstractShoppingSite` 定义购物网站的 schema；
`search\_product` 的文档字符串为 “Searches for a product.” ；
`add\_to\_cart` 的文档字符串为 “Adds a specified item to the shopping cart.” ；
`checkout` 的文档字符串为 “Initiates the checkout process.” 。

维护：修复 → 退役。表格对比 PolySkill 抽象类与网站实现。抽象购物网站类定义
搜索商品 `search\_product(self, query: str)`、将指定商品加入购物车
`add\_to\_cart(self, item\_id: str, quantity: int)`、发起结账 `checkout(self)`，以及
组合技能 `find\_and\_add\_to\_cart(self, query: str, item\_id: str)` 和
`purchase\_item(self, query: str, item\_id: str)`。

AmazonWebsite 实现：通过搜索框 ID 输入查询并按 Enter；通过按钮 ID 加入
购物车；结账是组合操作，先查看购物车，再点击结账按钮。

TargetWebsite 实现：通过搜索框 ID 输入查询并点击搜索按钮；添加商品时还填
写数量栏，再点击加入购物车按钮；结账时点击结账按钮。

组合只针对抽象方法编写一次；每个网站只需实现底层浏览器行动。

PolySkill · Yu 等, ICLR 2026——表 1 ↗

PolySkill：一个接口，多种实现

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Consequences of skill representation

REPRESENTATION	STRENGTHS	WEAKNESSES
Retrieved episode	preserves concrete behavior	long, hard to transfer
Text skill	provides flexible guidance	doesn't improve efficiency
Code skill	executable, composable, efficient	can be brittle

How much of the behavior is left to the model, and how much is fixed in the artifact?

36

检索到的情节：保留具体行为；缺点是很长，且难以迁移。

文本技能：提供灵活指导；缺点是不提升效率。

代码技能：可执行、可组合、高效；缺点是可能脆弱。

有多少行为留给模型处理，又有多少被固定在制品中？

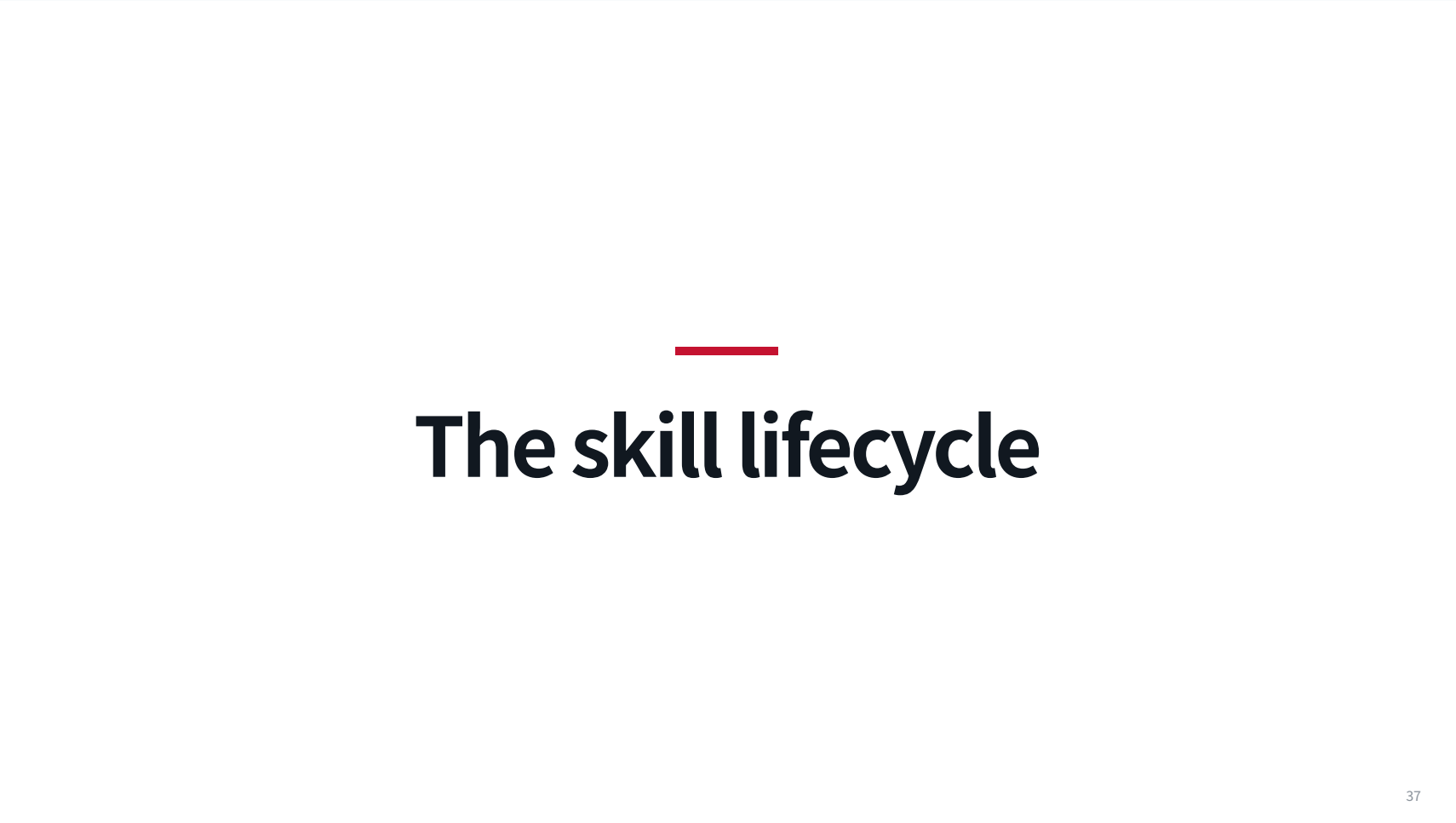
技能表示方式的后果

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

技能生命周期



技能生命周期

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Learning from failures



EPISODE

“Provide me with the complete names of Bluetooth headphones from Sony, and also share the price range for the available models.”

search “Bluetooth headphones Sony” → 5,578 results from every department of the store, 12 per page → next page → next page → ... → step limit, nothing sent to the user

Failure.

INDUCED STRATEGY

“Search query optimization” — a vague query buried the answer under thousands of loosely matched results. Tighten the query before browsing.

“Adjust number of items displayed per page” — twelve per page makes every “next” cost a step for little coverage.

“Use filters available” — the sidebar’s category filters shrink the list without paging.

ReasoningBank · Ouyang et al., ICLR 2026 — appendix case study ↗

38

学习阶段：评判 → 归纳 → 准入。

情节： ‘请提供 Sony 蓝牙耳机的完整名称，并给出所有在售型号的价格区间。’ 搜索 `Bluetooth headphones Sony` 得到来自商店所有部门的 5,578 条结果，每页 12 条；不断翻页直到触及步骤限制，未向用户发送任何内容。失败。

归纳的策略： ‘优化搜索查询’ ——模糊的查询使答案淹没在数千条匹配结果中；浏览之前先缩窄查询。

‘调整每页显示的商品数’ ——每页 12 条导致每次点 “下一页” 都需要一步，却只覆盖很少内容。

‘使用可用筛选条件’ ——侧栏类别筛选可以缩小列表，不必逐页翻找。

ReasoningBank · Ouyang 等, ICLR 2026——附录案例研究 ↗

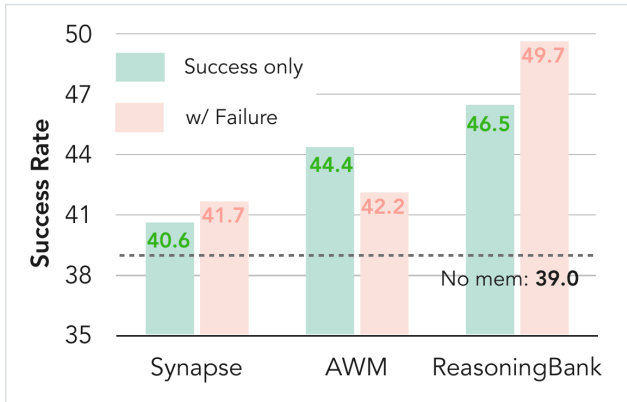
从失败中学习

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Failures help some memory types and hurt others



- Failures help substantially when inducing strategies (ReasoningBank)
- Failures are less helpful with trajectories (Synapse) or workflows (AWM): demonstrations of what not to do?

ReasoningBank · Ouyang et al., ICLR 2026 — WebArena shopping, Gemini-2.5-flash ↗

39

图表：成功率。对比仅使用成功经验（Success only）和包含失败经验（w/ Failure）；无记忆基线为 39.0。Synapse: 40.6 与 41.7；AWM: 44.4 与 42.2；ReasoningBank: 46.5 与 49.7。

在归纳策略时，失败能带来显著帮助（ReasoningBank）。

对轨迹（Synapse）或工作流（AWM）而言，失败帮助较小；它们是否可以作为‘不该怎么做’的示范？

ReasoningBank · Ouyang 等, ICLR 2026——WebArena 购物任务, Gemini-2.5-flash ↗

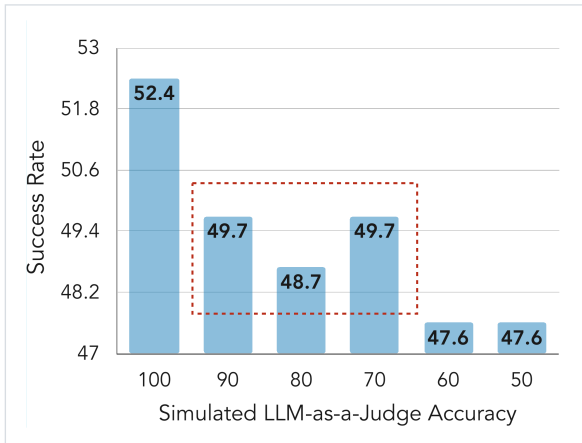
失败对某些记忆类型有帮助，对另一些则有害

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

How much does judge accuracy matter?



- **Setup:** the real judge is imperfect. To ask how much that matters, they swap in simulated judges: the true label, flipped with a fixed probability, from oracle to coin flip.
- **Result:** performance depends on the judge accuracy, but performance improves for a wide range of judge qualities.

ReasoningBank · Ouyang et al., ICLR 2026 — judge calibration analysis ↗

40

图表：成功率。模拟 LLM-as-a-Judge 准确度依次为 100、90、80、70、60、50；对应成功率为 52.4、49.7、48.7、49.7、47.6、47.6。

设置：真实评判器并不完美。为研究这一点，作者换用模拟评判器：以固定概率翻转真实标签，范围从 oracle 到抛硬币。

结果：性能取决于评判器准确度，但在较宽的一段评判器质量范围内，性能都会提升。

ReasoningBank · Ouyang 等, ICLR 2026——评判器校准分析 ↗

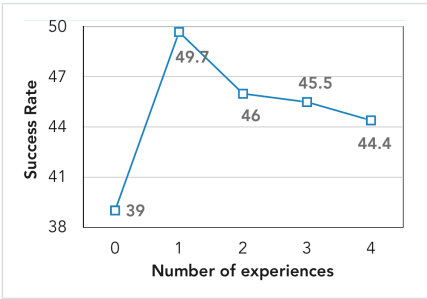
评判器准确度有多重要？

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

The cost of over-retrieval



- Retrieving additional experiences can cause performance to degrade!
- Example above from ReasoningBank.
- SkillsBench found the same for human-curated skills.
- Some experiences may be irrelevant: limitations of the memory
- The agent may be limited in what it can condition on

ReasoningBank · Ouyang et al., ICLR 2026 — WebArena shopping ↗

41

使用阶段：检索 → 行动。图表：成功率随检索经验数量变化：0 项经验为 39；1 项为 49.7；2 项为 46；3 项为 45.5；4 项为 44.4。

检索额外经验可能导致性能下降！

上图示例来自 ReasoningBank；SkillsBench 在人工策划技能上也发现相同现象。

某些经验可能无关；智能体可用于条件化的信息也可能有限。

ReasoningBank · Ouyang 等, ICLR 2026——WebArena 购物任务 ↗

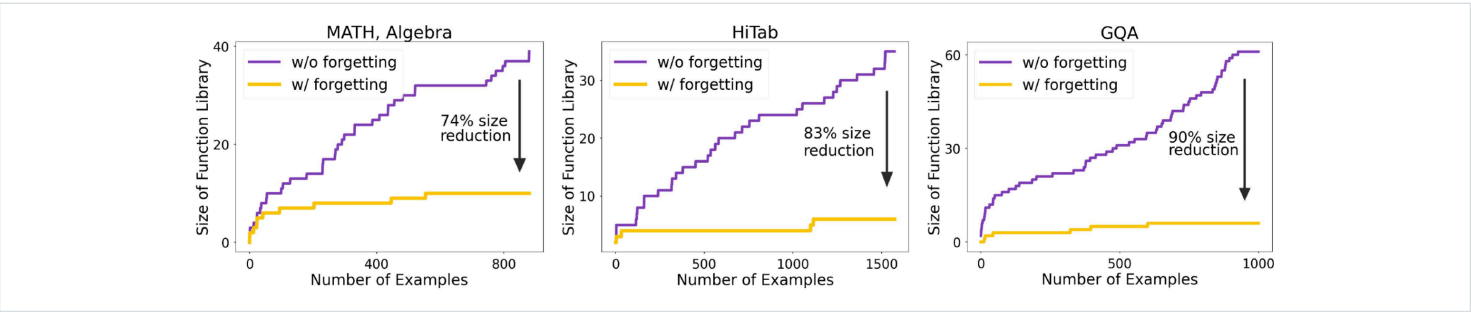
过度检索的代价

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Memory consolidation



- TroVE drops any function used fewer than $\frac{1}{2} \cdot \log_{10}(n)$ times, where n is the number of examples seen so far.

[TroVE · Wang et al., ICML 2024 ↗](#)

[Also: Not All Skills Help — a skill that helps one task type hurts another ↗](#)

42

维护：修复 → 退役。

TroVE 会删除使用次数少于 $\frac{1}{2} \cdot \log_{10}(n)$ 的函数，其中 n 是迄今见过的样例数。

TroVE · Wang 等, ICML 2024 ↗；另见《并非所有技能都有帮助》：有助于一种任务类型的技能可能损害另一种任务类型 ↗

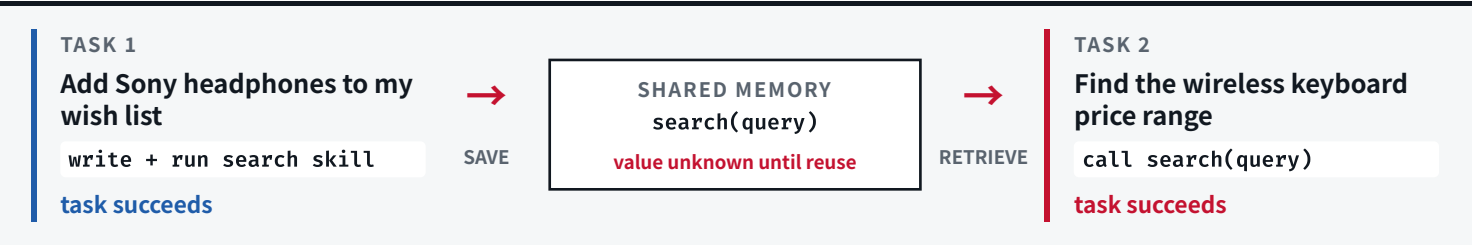
记忆整合

第 4 讲 · 智能体能力 3：技能与记忆

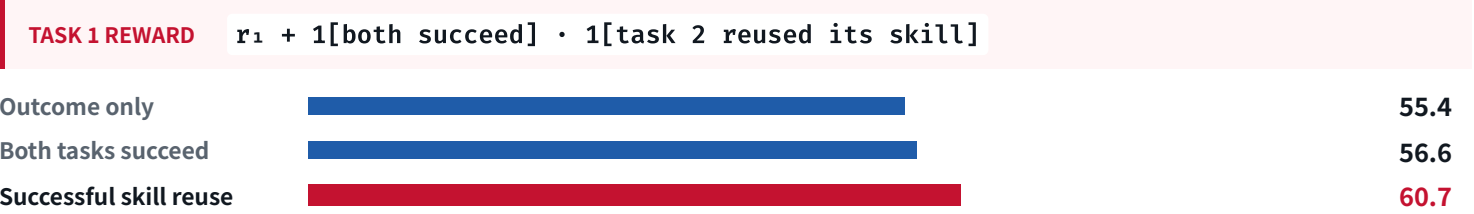
EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Training the skill inducer



Task 1's reward says whether the task succeeded, not whether saving part of the solution was worthwhile — only a later task can show that. So put the later task inside the training example: two related tasks in one rollout [SAGE], or one long trajectory [AgeMem].



Scenario completion · AppWorld test-normal

SAGE · Wang et al., ACL 2026 — AppWorld, Qwen2.5-32B ↗ AgeMem · Yu et al., ACL 2026 ↗

任务 1：将 Sony 耳机加入愿望清单；编写并运行搜索技能，任务成功。保存到共享记忆中的 `search(query)`；其价值要等到复用时才知道。

任务 2：查找无线键盘的价格区间；检索并调用 `search(query)`，任务成功。

任务 1 的奖励只能说明任务成功，不能说明保存解法的一部分是否值得；只有后续任务能揭示这一点。因此，要把后续任务放进训练样例：在一次 rollout 中安排两个相关任务 [SAGE]，或放在一条长轨迹中 [AgeMem]。

任务 1 奖励：`r\_1 + 1[both succeed] · 1[task 2 reused its skill]`。

场景完成率（AppWorld test-normal）：仅结果 55.4；两个任务都成功 56.6；成功复用技能 60.7。

SAGE · Wang 等, ACL 2026——AppWorld, Qwen2.5-32B ↗；AgeMem · Yu 等, ACL 2026 ↗

训练技能归纳器

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Discussion: what should persist?

exact episode	↔	general skill
flexible guidance	↔	committed execution
reuse what exists	↔	explore and replace
grow the memory	↔	update, merge, delete

- 1. What would you store as an episode, a fact, a lesson, a text skill, or code — and why?
- 2. When should an agent delete a skill? Revise it?
- 3. What's the right balance between agent and human effort in creating and using skills?

44

确切情节 ↔ 通用技能。

灵活指导 ↔ 已确定的执行。

复用现有内容 ↔ 探索并替换。

增长记忆 ↔ 更新、合并、删除。

1. 你会把什么保存为情节、事实、教训、文本技能或代码技能？为什么？ 2. 智能体何时应该删除或修订技能？ 3. 创建和使用技能时，智能体与人类投入多少精力才合适？

讨论：应该保留什么？

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分

Ownership of memory decisions

DECISION	HUMAN EFFORT IS VALUABLE WHEN...	AGENT EFFORT IS VALUABLE WHEN...
Author or induce	the skill or policy is already known	it must be discovered through interaction
Verify and admit	correctness is normative or high-stakes	outcomes are executable and cheap to test
Select and use	rare exceptions need contextual judgment	routing repeats and produces feedback
Maintain	accountability requires an owner	drift is visible in outcomes and repairable

45

人工投入有价值的情形 | 智能体投入有价值的情形。

编写或归纳：技能或策略已知 | 必须通过交互发现。

验证与准入：正确性由规范决定或后果重大 | 结果可执行且测试成本低。

选择与使用：罕见例外需要结合上下文判断 | 路由会反复发生并产生反馈。

维护：问责要求明确负责人 | 结果中的漂移可见且能够修复。

记忆决策的责任归属

第 4 讲 · 智能体能力 3：技能与记忆

EN / 英文官方原页 · 保持原样

ZH / 中文译文 · 第 1/1 部分



下一讲：规划。

问题