清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

如何让Embodied Agent从经验中*有效学习*并*持续进化*

Speaker: 丁鑫

2026年7月3日

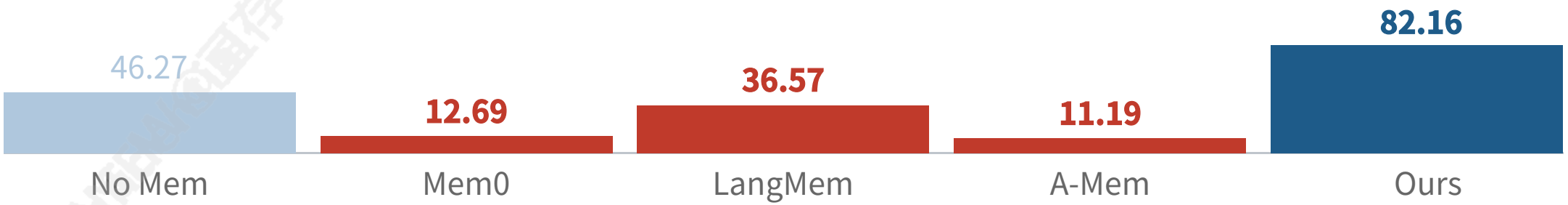
更多的记忆，不一定更好

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

反直觉现象

给智能体注入历史经验，超过一半的情况下表现反而变差。

这说明：经验的价值，需要有效的利用才能体现出来



AlfWorld + Qwen-2.5-14B

为什么具身经验管理难？

语言智能体常把记忆当作文本上下文；具身智能体面对的是动态视觉—动作流。

部分可观测

物体位置、状态和可见性随动作变化；当前观察并不等于完整环境。

长期规划的必要性

复杂任务需要现在做的事情为将来铺路，但不知道路上会遇到什么情况

具身动作的不可逆性

动作是不可逆的，必须在行动前就判断好，不能依赖试错

稀疏且延迟的奖励信号

很多任务，只有在最后才知道成功还是失败

具身任务中的“经验”不仅是文字，还包括空间、视角、物体状态和动作前提。

“用”与“炼”

推理时 · INFERENCE

静态注入，日益脱节

现有方案在任务开始前一次性塞入所有记忆，全程不变。随着任务推进，注入内容与当前状态越来越不相关——有用的经验被无关条目稀释。

- 任务动态推进的
- Attention dilution 效应
- 无法感知智能体当前状态
- 开销大、噪声多

任务后 · POST-EPISODE

盲目更新，覆盖正确知识

失败轨迹被无差别地用来更新技能，但失败有两种根因——技能本身有误，还是执行时没跟上？混淆二者，好的知识会被错误推翻。

- Skill defect vs. Execution lapse
- 正确内容被错误覆盖
- 演化质量无法保证

“用”与“炼”

推理时 · INFERENCE

静态注入，日益脱节

现有方案在任务开始前一次性塞入所有记忆，全程不变。随着任务推进，注入内容与当前状态越来越不相关——有用的经验被无关条目稀释。

任务后 · POST-EPISODE

盲目更新，覆盖正确知识

失败轨迹被无差别地用来更新技能，但失败有两种根因——技能本身有误，还是执行时没跟上？混淆二者，好的知识会被错误推翻。

这两个问题，一个关于"*用*"，一个关于"*炼*”
共同构成了智能体知识生命周期的两个关键环节

MemCompiler

EmbodiSkill

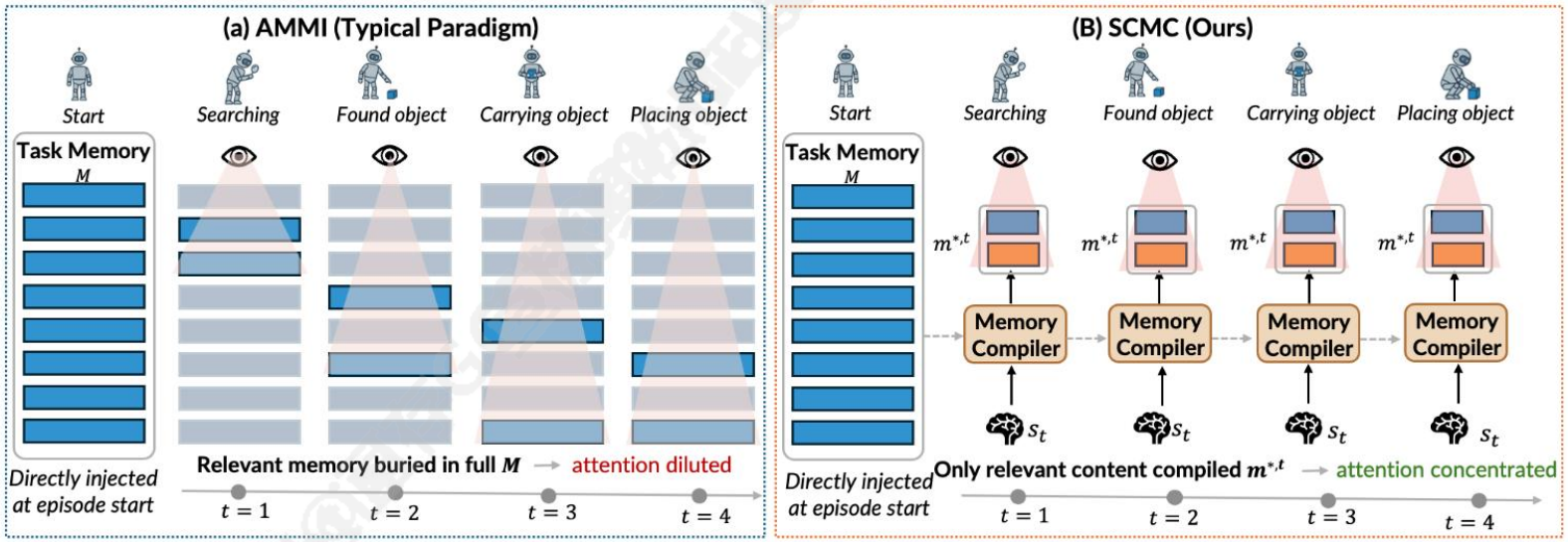
推理时，如何用好已有经验?

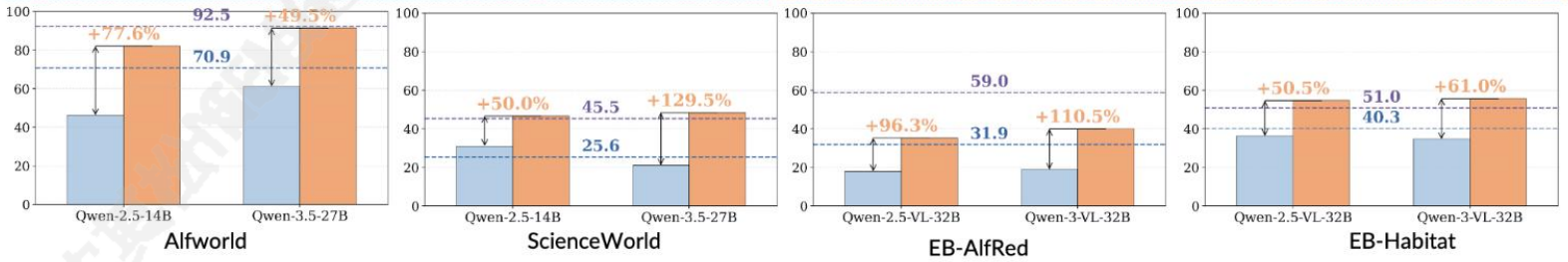
不要在任务开始时 dump 所有记忆——像编译器一样，根据当前状态按需编译。

MemCompiler

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

同一份源码面对不同运行环境，会编译出不同目标指令；同一份经验面对不同执行状态，也应该编译出不同指导





Before · AMMI

- **静态性**：任务是动态的，但这样带来的经验是静态的。
- **单一化**：文本形式的长篇大论
- **资源浪费**：模型要处理大量此刻用不到的信息 浪费计算资源，还会分散注意力

After · SCMC

- **根据智能体当前的状态，动态决定此刻需要哪些记忆。**
- **形式并不局限于文本，而是以最合适的形式提供给它。**

记忆交付从“预先一次性”变成“随状态变化的可执行指导”

MemCompiler

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

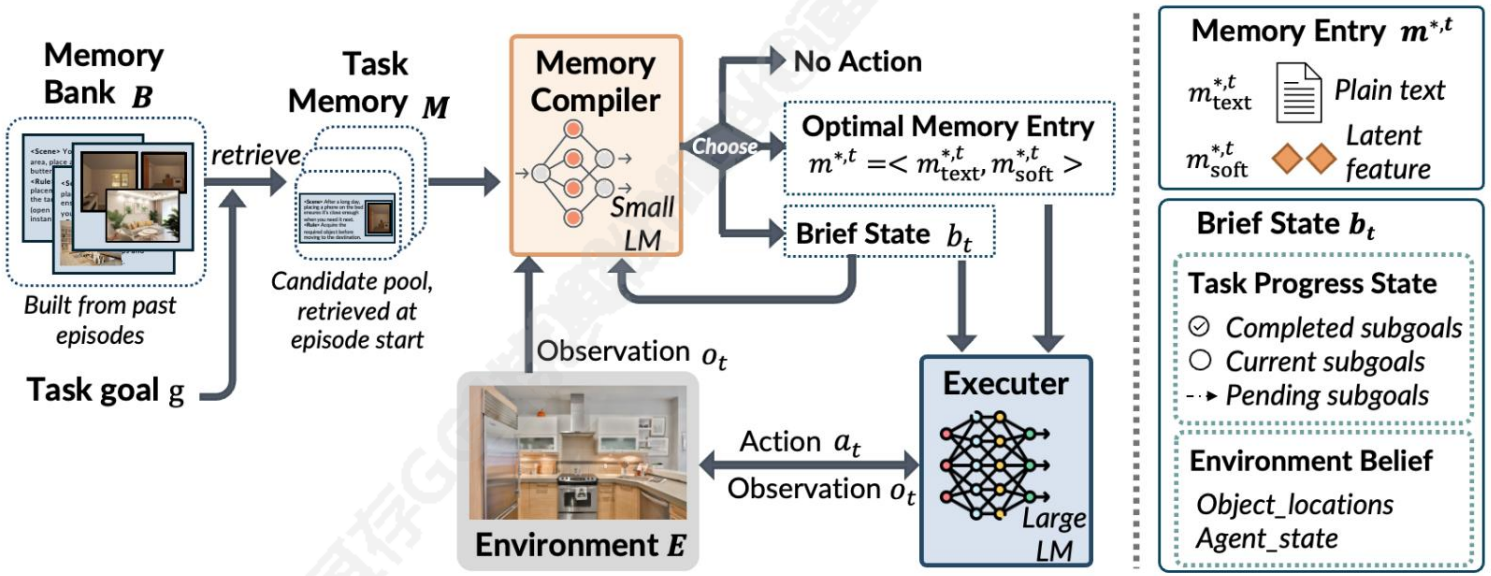


Figure 2 | Overview of MemCompiler at step t. The Memory Compiler reads runtime state $s_t = (o_t, b_t)$ and task memory $\mathcal{M}$ (retrieved once at episode start), then delivers the compiled result $m^{*,t}$ to the Executor through two parallel channels: a text channel ($m^{*,t}_{\text{text}}$) and a latent soft channel ($m^{*,t}_{\text{soft}}$), which are fused at the Executor's embedding level. Brief State b_t is dynamically maintained via structured operations and fed back as part of the next step's state. Note, at each step the Memory Compiler decides among four output types (Section 3.2): EXPERIENCE ($m^{*,t}$ only), BRIEF (Δb_t only), HYBRID (both jointly), or NOACTION (neither, when no entry in $\mathcal{M}$ is currently applicable).

1 检索一次

episode 开始按 task goal 从 memory bank 取候选池 M

2 状态驱动——When to use

Brief State b ： 记录任务的进度以及我对当前环境的理解是什么

3 双通道交付——How to use

文本解释 + 隐空间感知线索一起送入 executor

4 持续维护

Brief State 被结构化更新，进入下一步决策

效果：注意力不再被“静态记忆”稀释

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

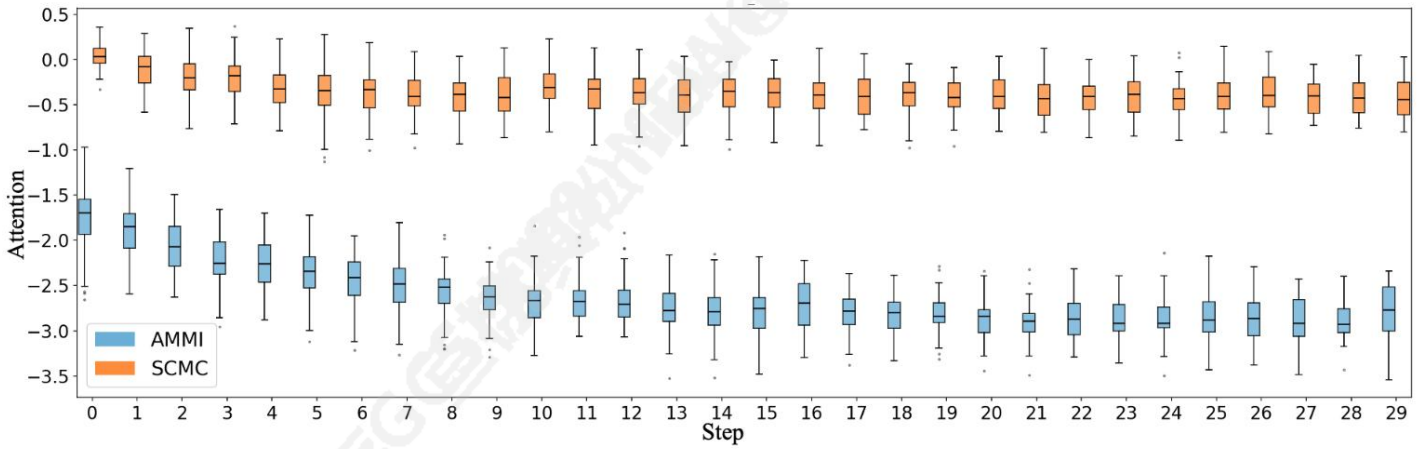


Figure 3 | Mean pre-softmax attention logit (higher values mean more attention weight) of the Executor over memory tokens across Alf World episodes (steps 0–29). AMMI (blue) exhibits monotonic decay, reflecting increasing misalignment between static memory and the agent's evolving state. In contrast, SCMC (orange) maintains stable high attention throughout, indicating consistent alignment between memory and the Executor's current needs.

stable

SCMC 高注意力

动态编译后的记忆始终与当前状态对齐

−86.7%

AMMI 注意力衰减

到 step 29，executor 对 memory tokens 的关注大幅下降

从机制上看，MemCompiler 把“记忆检索”从 episode-level 提升到了 state-level

实验结果

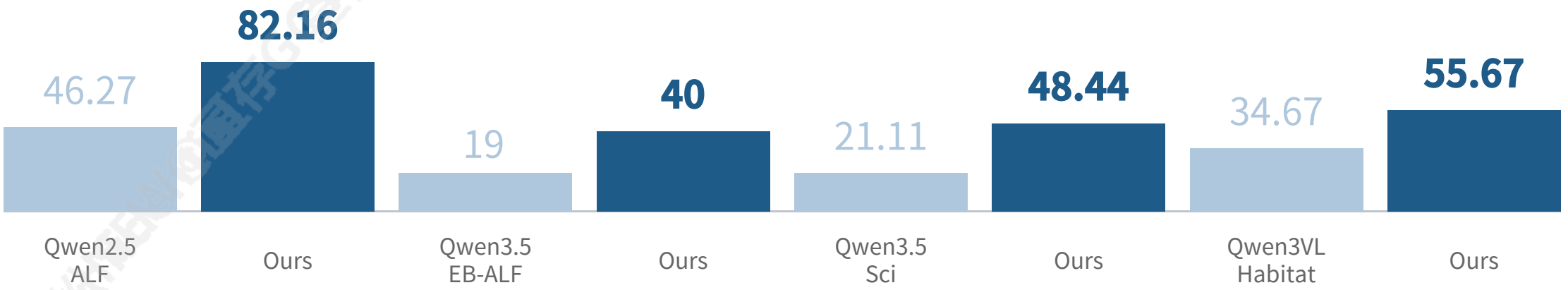
清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

+77.6%
AlfWorld
Qwen-2.5-14B：46.27 → 82.16

+110%
EB-ALFRED
Qwen-3-VL-32B：19 → 40

+129%
ScienceWorld
Qwen-3.5-27B：21.11 → 48.44

+61%
EB-Habitat
Qwen-3-VL-32B：34.67 → 55.67

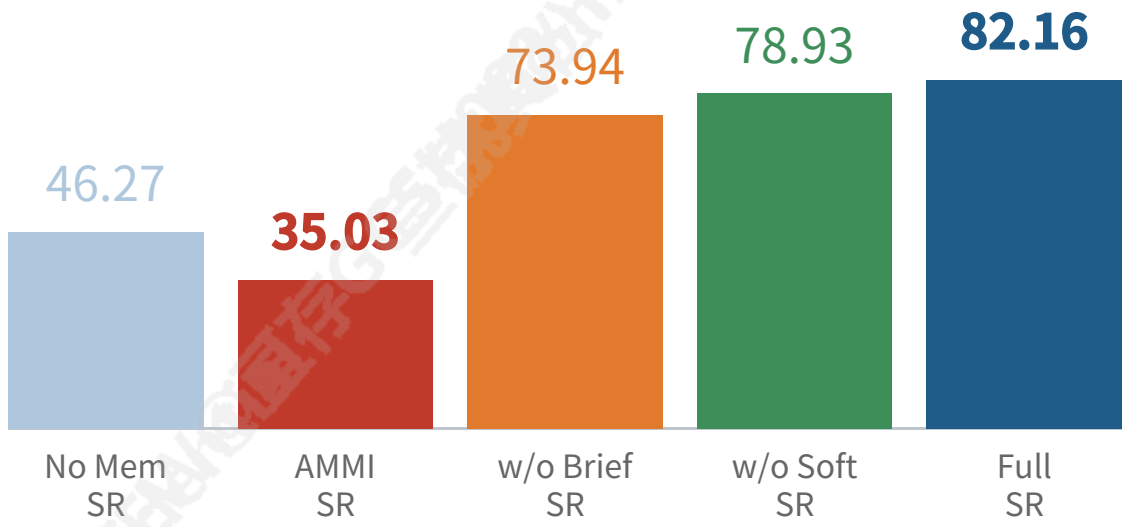


中等规模开源模型 + 正确的记忆交付，可以接近甚至超过前沿闭源系统的 no-memory 表现

消融与效率

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

在相同 **memory pool** 上，动态编译比静态注入更准、更快。



输入 token

2481.1 → 1003.2（减少约 60%）

executor latency

0.30s → 0.12s（编译开销被执行降本抵消）

success rate

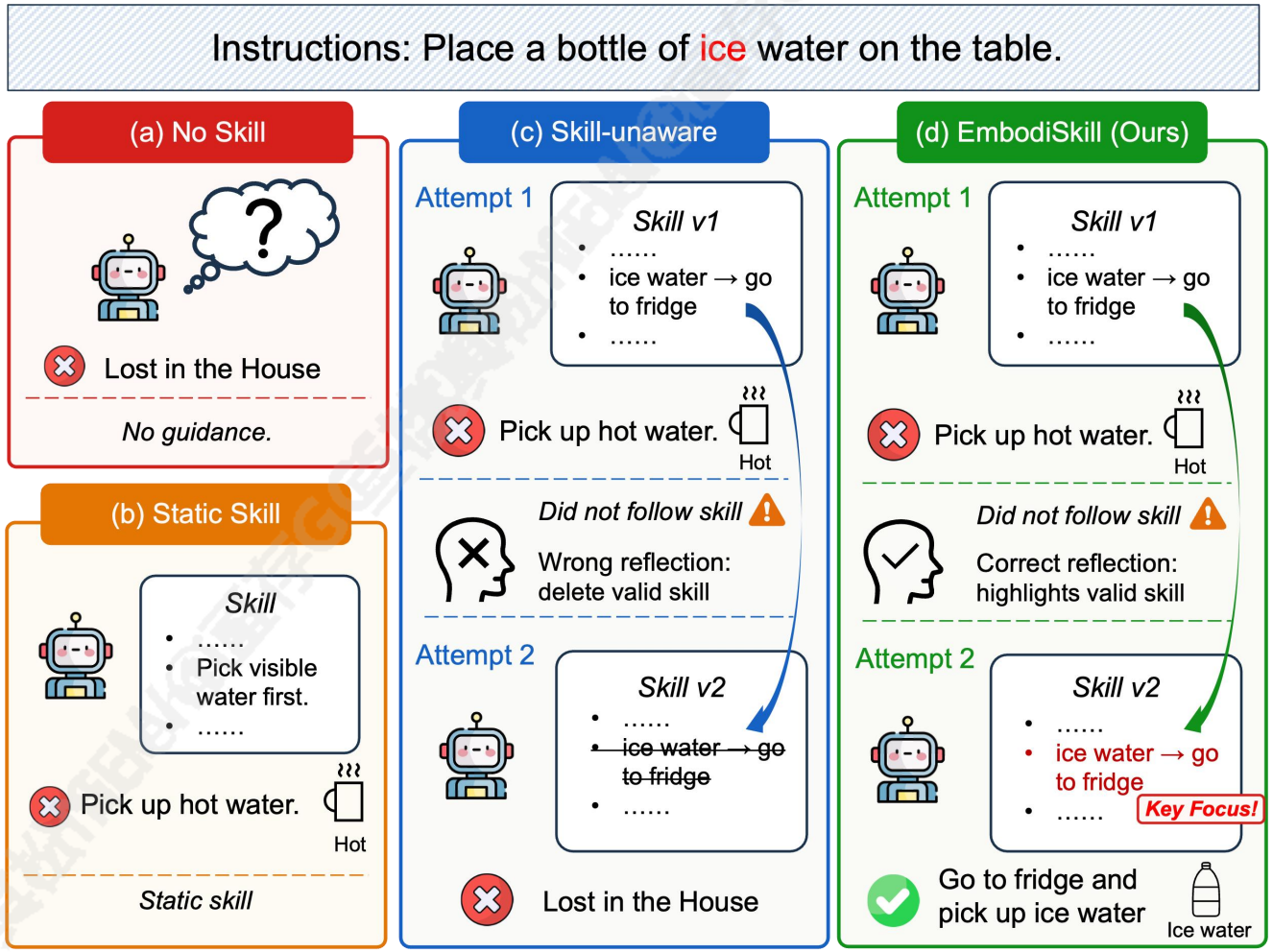
35.03% AMMI → 82.16% SCMC

Brief State 与 SCMC 是核心；Soft-Mem 在含视觉/空间信息的任务上进一步增强。

任务后，如何从轨迹中中学到正确东西?

失败不总是技能错了：可能是技能正确，但 executor 没有跟上。

Motivation example



No Skill / Static Skill

没有指导，或技能不随轨迹更新，容易在房间中**低效探索**。

Skill-unaware

很多情况下往往分不清是经验本身还是自身执行的问题，导致**盲目更新**。

EmbodiSkill

Skill-aware Reflection
识别 执行疏漏（技能本身正确，执行时没有遵循）：不改 body，而是强调“去冰箱取冰水”。

具身任务结果受到感知、空间 grounding、物体状态和执行可靠性共同影响。
因此**不能用成败粗暴改写技能**。

EmbodiSkill

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University

Skill是持久且可修订的 procedural specification，用于指导搜索、动作前提、状态变化与恢复策略。

Current *Skill*$^{(n)}$

Body
know how to do it

智能体真正需要的任务规则
- 搜索策略
- 子目标顺序
- 动作前提
- ……

可由 Discovery / Optimization / Skill Defect 更新

Appendix
do not forget to follow it

不引入新规则；只强调 body 中已经有效、但 executor 容易忽略的内容。

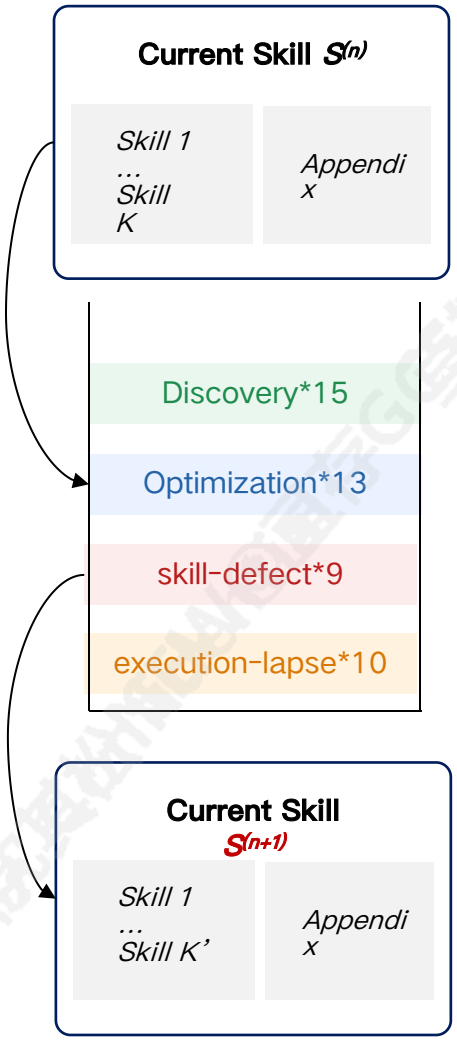
只由 Execution Lapse 更新。

执行失败 ≠ 技能错误；先归因，再修订

避免将正确技能内容从 body 中删除，也能让执行中容易被忽略的关键步骤得到强化

Targeted Skill Revision

Reflection不是立刻修改Skill，而是先积累、分组整合，再做局部、有证据、有边界的修改。



Step 1. 积累
积累足够多的Reflection才会进行Skill的修订。

Step 2. 分组
按类型分组reflection： Discovery, Optimization, Skill Defect, and Execution Lapse。

Step 3. 合并
去除冗余内容，合并重叠的建议，并解决冲突。

Step 4. 修订
用 Discovery / Optimization / Skill-Defect 针对Skill body编辑。
使用 Execution-Lapse 强调Skill body内容，但不更改Skill body内容。

Skill-Aware Evolution Spiral

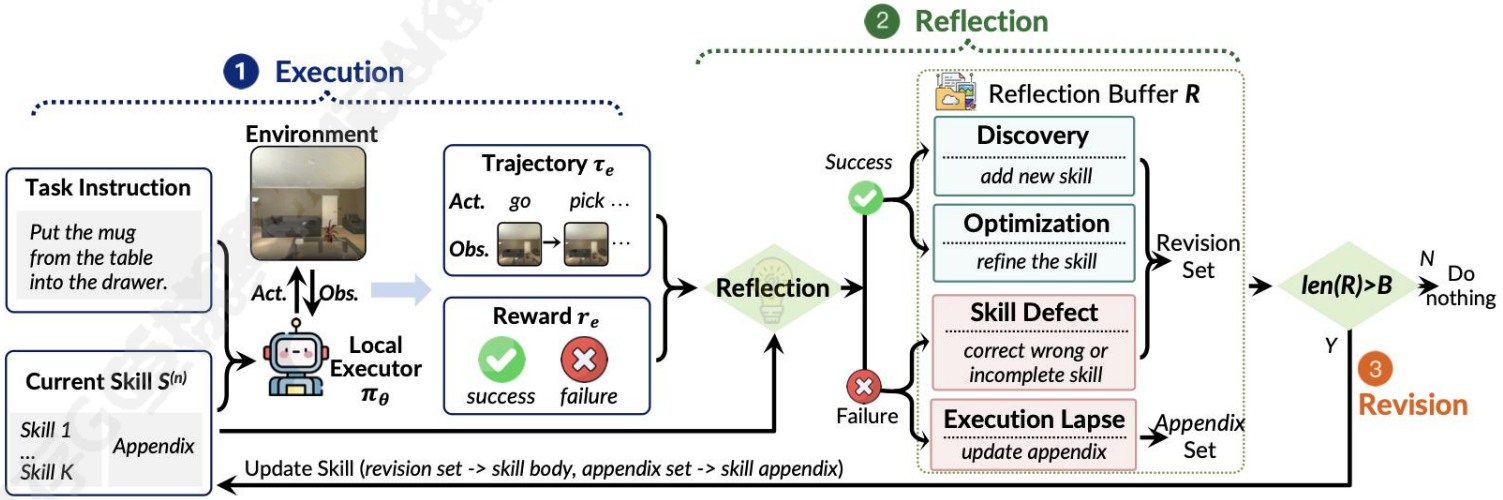
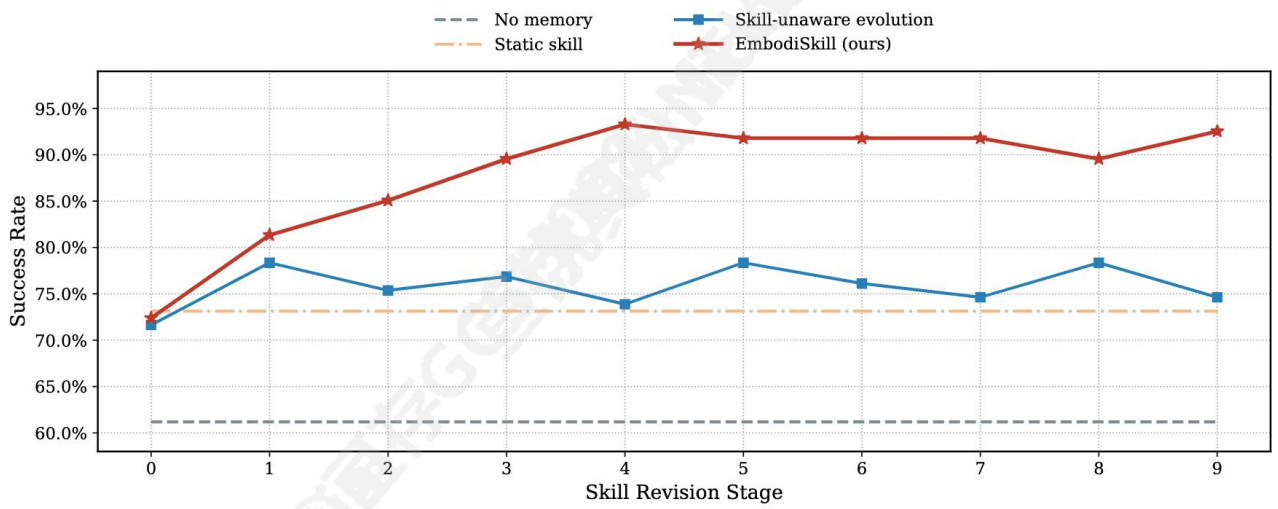


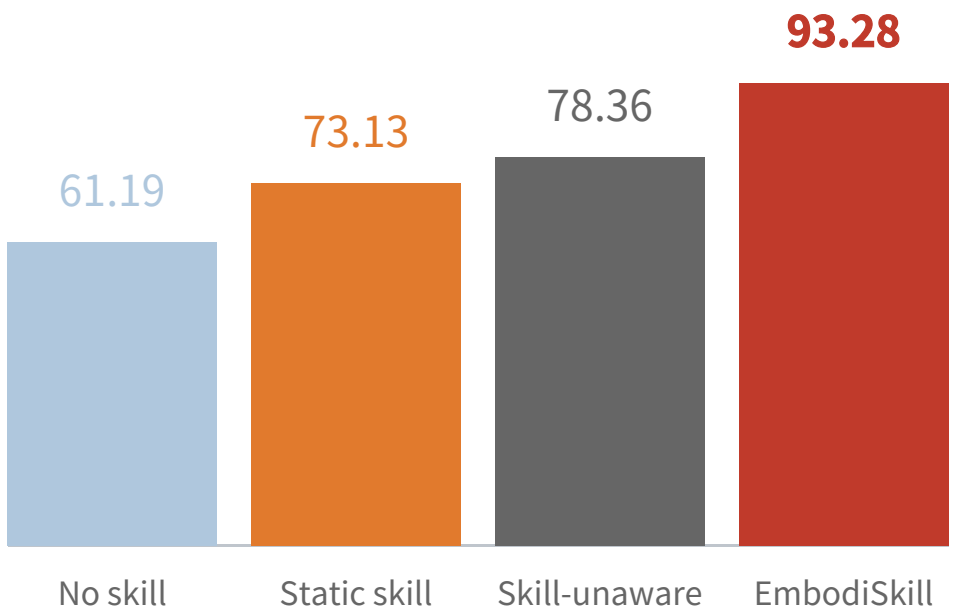
Figure 2 | Overview of EmbodiSkill. The executor uses the current skill to perform embodied tasks and generate trajectories. Skill-aware reflection uses each trajectory to produce targeted reflection records. Accumulated reflections are consolidated into body-level revisions and skill-appendix updates, forming the next skill version. The revised skill then guides subsequent task execution, creating a Skill-Aware Evolution Spiral.

技能与轨迹互相塑造，整体效果螺旋上升

消融

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University





Static skill 有用

从61.19%提升到73.13%，说明skill本身很重要

Skill-unaware 不够

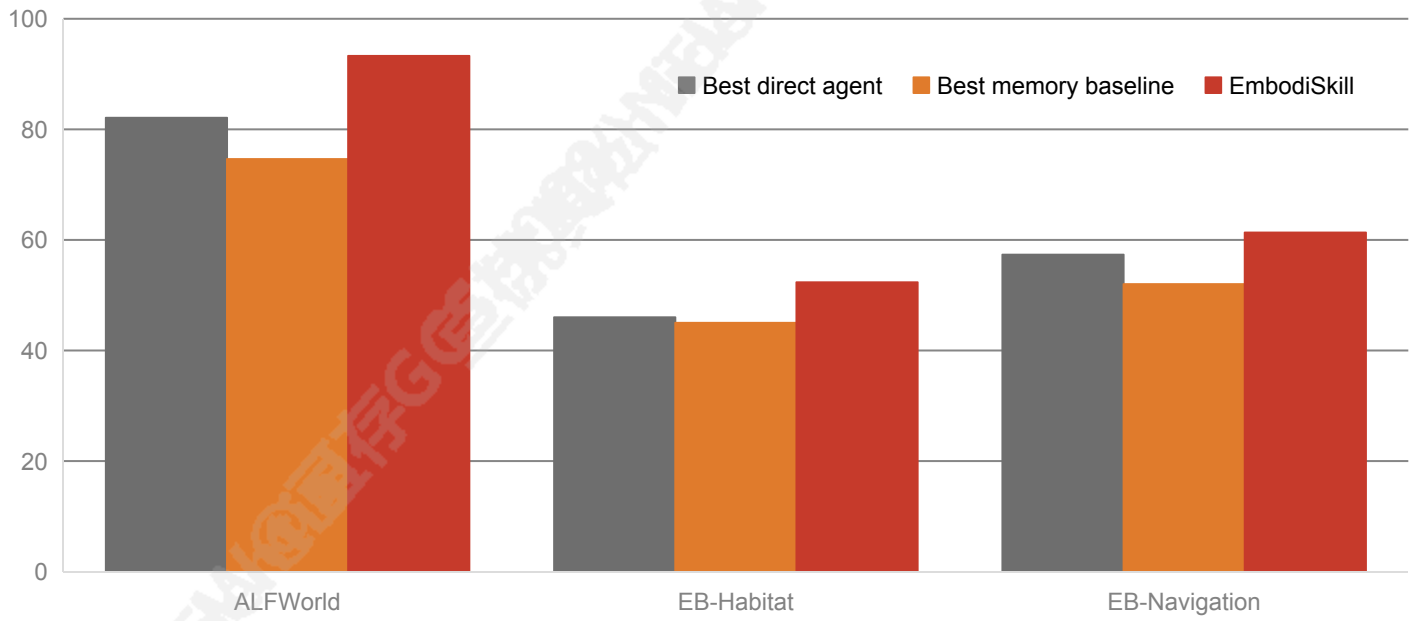
粗粒度演化会获得一定的性能提升，但会波动且收敛到低性能区间

Skill-aware 是核心

从78.36%提升到93.28%，相对skill-unaware提升19.04%

实验结果

清華大學 智能产业研究院
Institute for AI Industry Research, Tsinghua University



ALFWorld · 93.28%

+31.58% vs GPT-5.2 direct
+25.01% vs G-Memory

EB-Habitat · 52.33%

+13.76% vs strongest direct agent
+16.29% vs strongest memory baseline

EB-Navigation · 61.33%

+6.98% vs strongest direct agent
+17.94% vs strongest memory baseline

冻结executor

训练 free：收益来自技能演化

胜过强模型直接执行

直接用强模型执行，不如外部技能演化后的小 executor

优于memory

trajectory-level memory 不如 procedural skill

完整的知识生命周期闭环

清华大学 智能产业研究院
Institute for AI Industry Research, Tsinghua University

01

积累经验

智能体执行任务，产生成功与失败的轨迹，形成原始经验库。

02

动态编译
MemCompiler

推理时根据当前状态按需选取记忆，减少干扰，提升利用效率。

03

执行任务

智能体在精准记忆引导下完成任务，产生新的轨迹和反馈。

04

精准演化
EmbodiSkill

区分失败根因，定向修订技能，知识质量螺旋上升。

使用中等规模的开源模型，配合好的知识管理，可以超越当前最强的前沿闭源系统。

AI 智能体能力的边界，或许不由模型有多大来决定。

Thanks !
Q & A