

ReAct: 将思考与行动相结合

赋能大型语言模型以更高能力、可靠性和可信度

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narashan, Yuan Cao
Princeton University & Google Research

大语言模型(LLM)的两个世界：思考者 vs. 行动者



“思考者” (The Thinker) - 纯推理模型

通过生成思维链 (Chain-of-Thought) 来解决复杂问题。模型利用其内部知识和推理能力来推导出答案。

关键优势

擅长分解问题，进行多步推理。

代表方法

Chain-of-Thought (CoT) Prompting.

“行动者” (The Doer) - 纯行动模型

生成特定领域的动作指令，与外部环境 (如API、游戏) 进行交互来完成任务。

关键优势

能够与外部工具和信息源互动，获取实时、准确的数据。

代表方法

Action Plan Generation.

割裂的范式：纯思考产生幻觉，纯行动陷入僵局



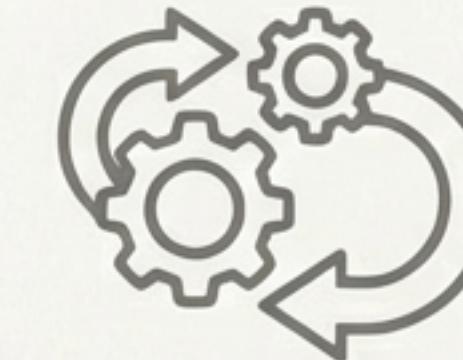
CoT（纯思考）的致命缺陷：事实幻觉

任务：HotpotQA

问题：除了苹果遥控器，还有什么其他设备可以控制它最初设计的程序？

Thought: Let's think step by step. Apple Remote was originally designed to interact with Apple TV. Apple TV can be controlled by iPhone, iPad, and iPod Touch. So the answer is iPhone, iPad, and iPod Touch.

分析：“模型在没有外部信息验证的情况下，自信地进行了错误的推理，产生了事实性幻觉（fact hallucination），最终导致答案错误。”



Act-Only（纯行动）的局限性：缺乏规划

任务：ALFWorld

任务：将胡椒瓶放到抽屉里。

Act 4: Take peppershaker 1 from sinkbasin 1

Obs 4: Nothing happens.

Act 5: Take peppershaker 1 from sinkbasin 1

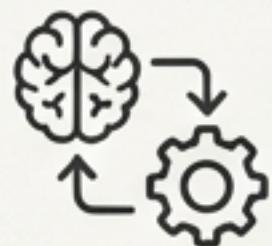
Obs 5: Nothing happens.

分析：“模型无法理解‘水槽里没有胡椒瓶’这一事实，缺乏推理和规划能力，导致其陷入无效的重复动作循环。”

ReAct：一种协同思考与行动的全新范式

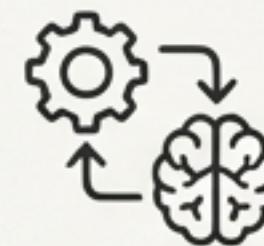


核心概念：ReAct通过提示LLM以交错的方式生成推理轨迹（reasoning traces）和任务动作（actions），从而实现二者的动态协同。



思考以更好地行动（Reason to Act）

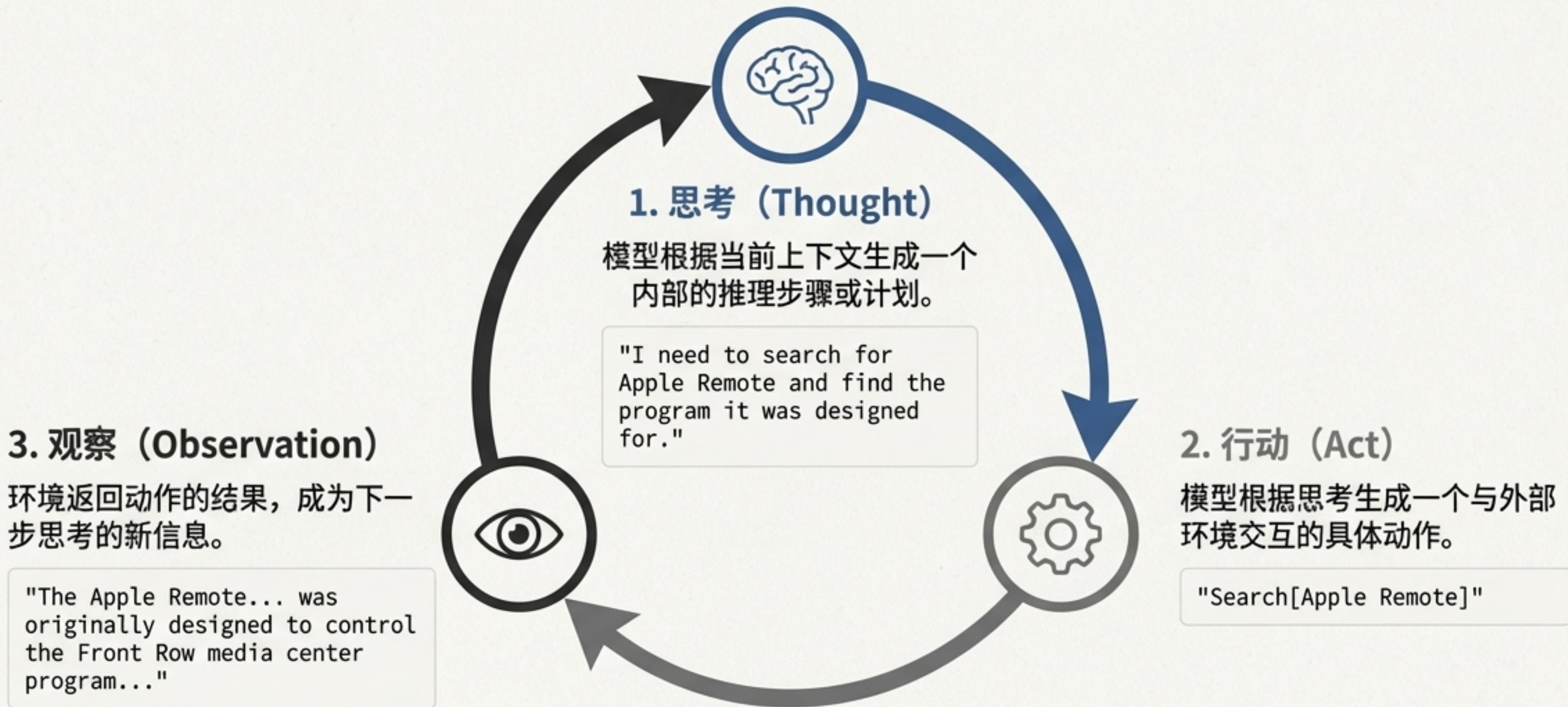
推理轨迹帮助模型创建、跟踪和调整行动计划。



行动以更好地思考（Act to Reason）

与外部环境（如维基百科API）的交互为模型提供了真实、可靠的信息，从而修正和指导推理过程。

ReAct 的核心循环：思考 → 行动 → 观察



核心洞见：“这个循环使得模型能够动态地根据外部世界的反馈来调整其内部的推理过程，从而实现更可靠、更具适应性的任务解决能力。”

实例分析（知识密集型任务）：ReAct 如何克服幻觉

任务：HotpotQA - "除了苹果遥控器，还有什么其他设备可以控制它最初设计的程序？"



Thought 1

我需要搜索苹果遥控器，找到它最初设计的程序。



Act 1

Search[Apple Remote]



Obs 1

...最初设计用于控制 Front Row 媒体中心程序...



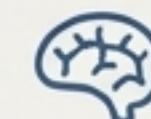
Thought 2

苹果遥控器最初是为 Front Row 设计的。我需要搜索 Front Row，找到其他可以控制它的设备。



Act 2

Search[Front Row]



Thought 3

除了苹果遥控器，键盘功能键也可以控制 Front Row。我需要确认是否还有其他设备。



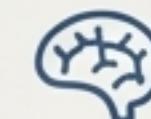
Act 3

Search[Front Row control devices]



Obs 3

...Front Row can be controlled by... the Apple Remote... and keyboard function keys...



Thought 4

Front Row 可由键盘功能键控制。所以答案是键盘功能键。



Act 4

Finish[keyboard function keys]

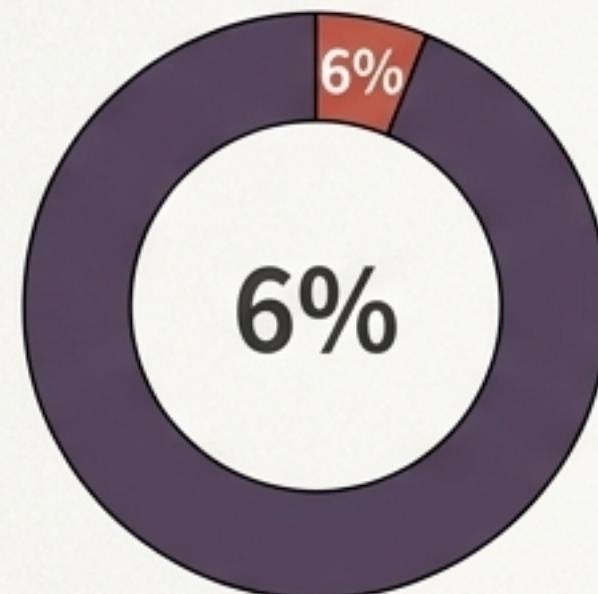
结论：“与CoT的内部幻觉不同，ReAct的每一步推理都基于从外部环境中获取的真实信息，使其解答过程更加扎实、可信。”

数据证明：ReAct 显著减少幻觉，提升解答可靠性

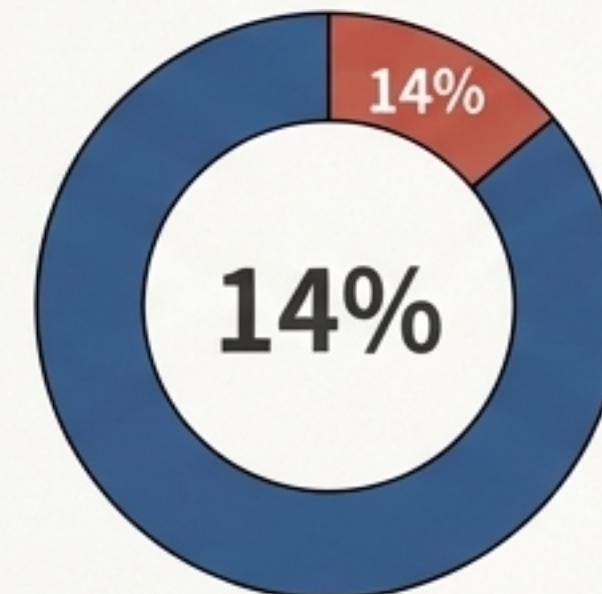
对HotpotQA成功与失败案例的人工分析

成功案例中的假阳性率

推理或事实存在幻觉



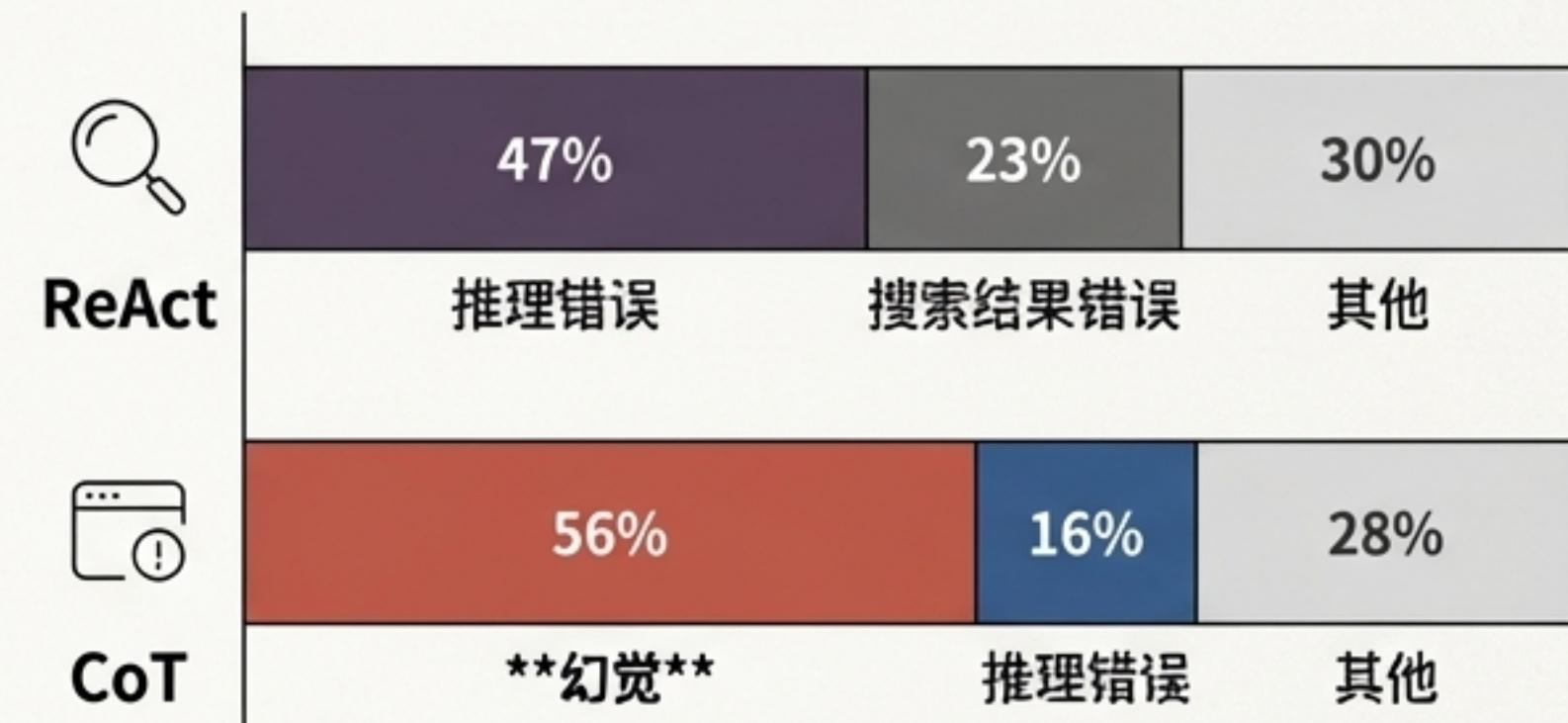
ReAct



CoT

洞见：“即使在最终答案正确的情况下，CoT的推理过程也更容易包含虚假信息。”

失败案例的主要原因



洞见：“幻觉是CoT失败的首要原因。而ReAct的失败模式更多与推理灵活性和信息检索策略有关，而非凭空捏造事实。”

融合内外知识：结合 ReAct 与 CoT 实现最佳性能

核心思想：ReAct擅长利用外部知识避免幻觉，而CoT在构建复杂的推理结构方面更灵活。将二者结合，可以取长补短。

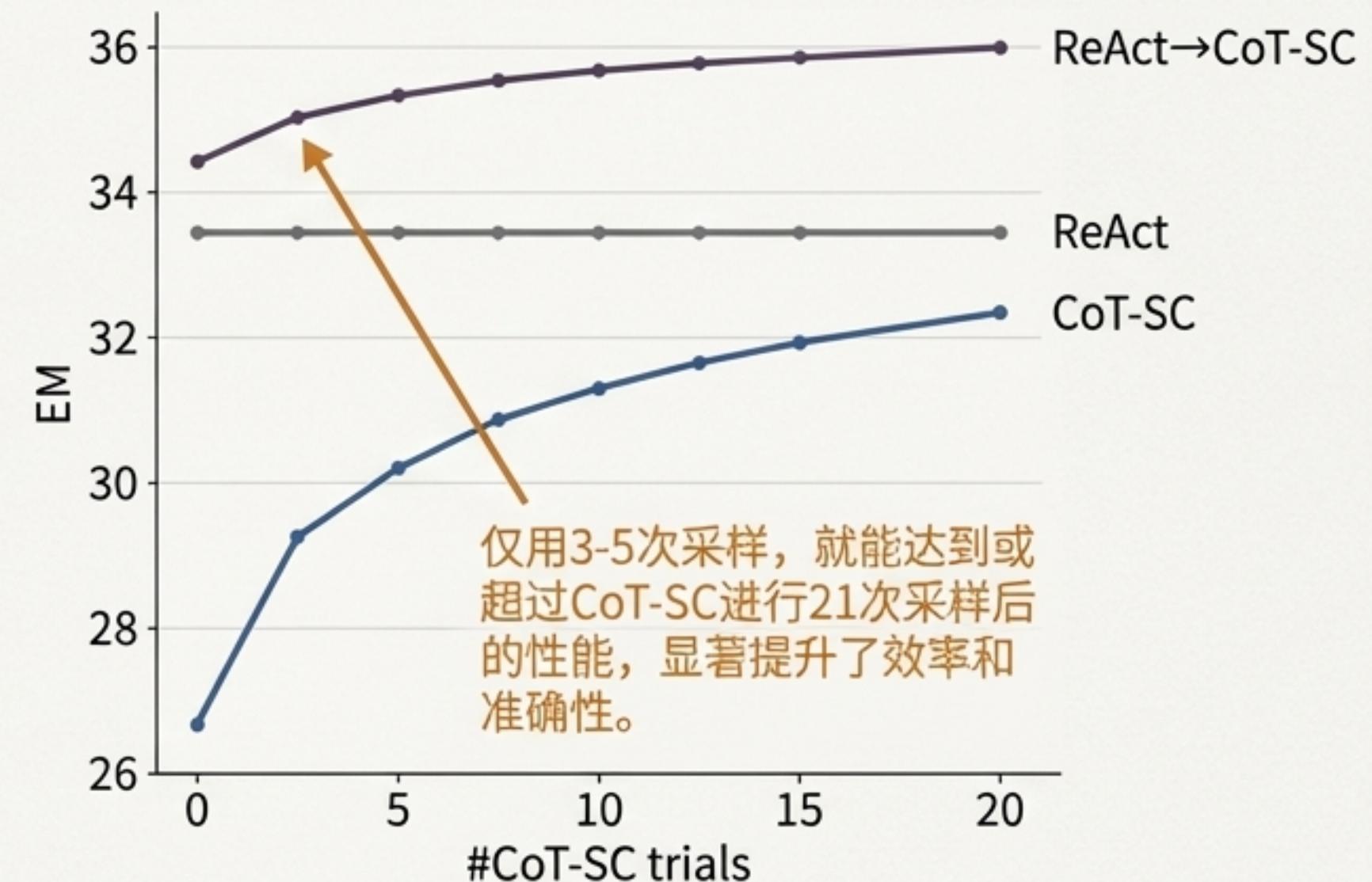
HotpotQA

‘ReAct→CoT-SC’ 组合方法达到 **35.1%** 的EM
超越了单独的ReAct (27.4%) 和 CoT-SC (33.4%)

Fever

‘CoT-SC→ReAct’ 组合方法达到 **64.6%** 的准确率
表现最佳

性能 vs. CoT-SC 采样次数 (HotpotQA EM)



实例分析（决策制定任务）：ReAct 注入规划与策略

任务：ALFWorld - "将一个胡椒瓶放到抽屉里。"

Act-Only (无思考)



"...Go to drawer 1 -> Open drawer 1 -> Go to sinkbasin 1 ->
Take peppershaker 1 from sinkbasin 1 (失败) ->
Take peppershaker 1 from sinkbasin 1 (失败)..."

“缺乏高层规划，无法追踪子目标，陷入无效循环。”

ReAct (有思考)



Think: 首先，我需要找到一个胡椒瓶...它可能在橱柜(1-6)，台面(1-3)...

Act: Go to countertop 3

Obs: ...在台面3上，你看到了一个胡椒瓶1...

Think: 现在我找到了胡椒瓶1。下一步，我需要把它放到抽屉1里。

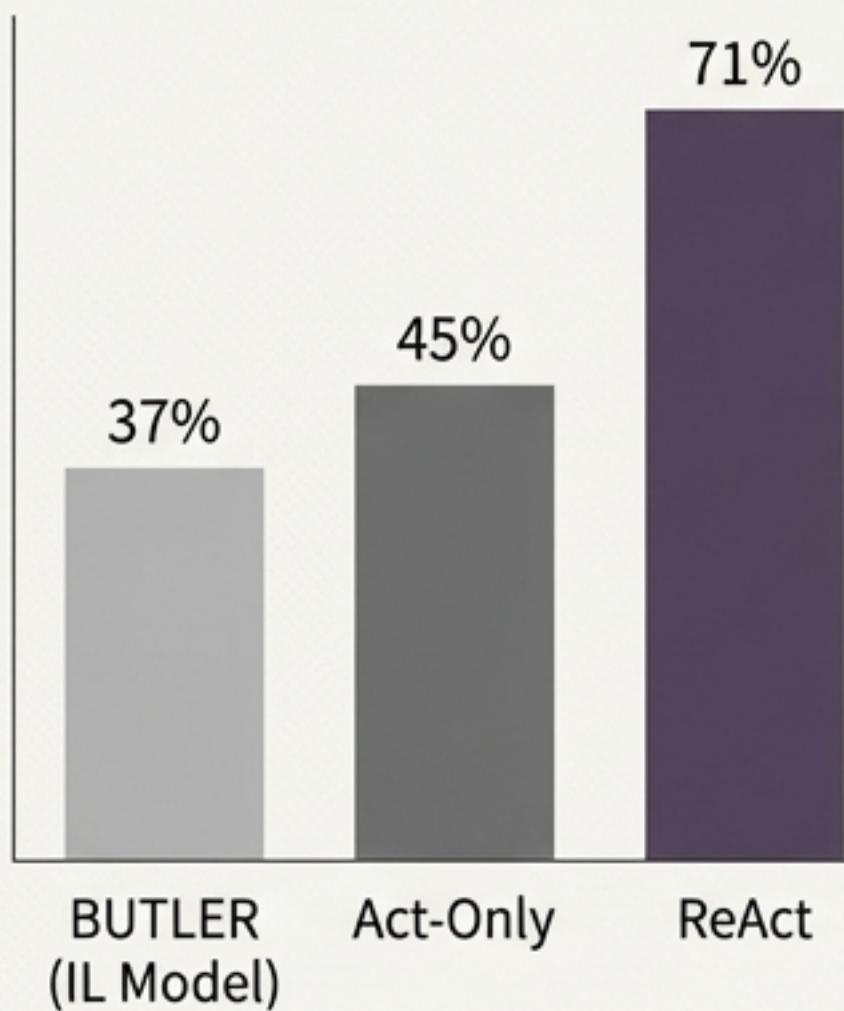
Act: Go to drawer 1...

“通过思考来分解目标、制定探索策略、追踪任务进展，实现了高效、成功的任务执行。”

少量样本学习的巨大成功：ReAct 在决策任务中超越专用模型

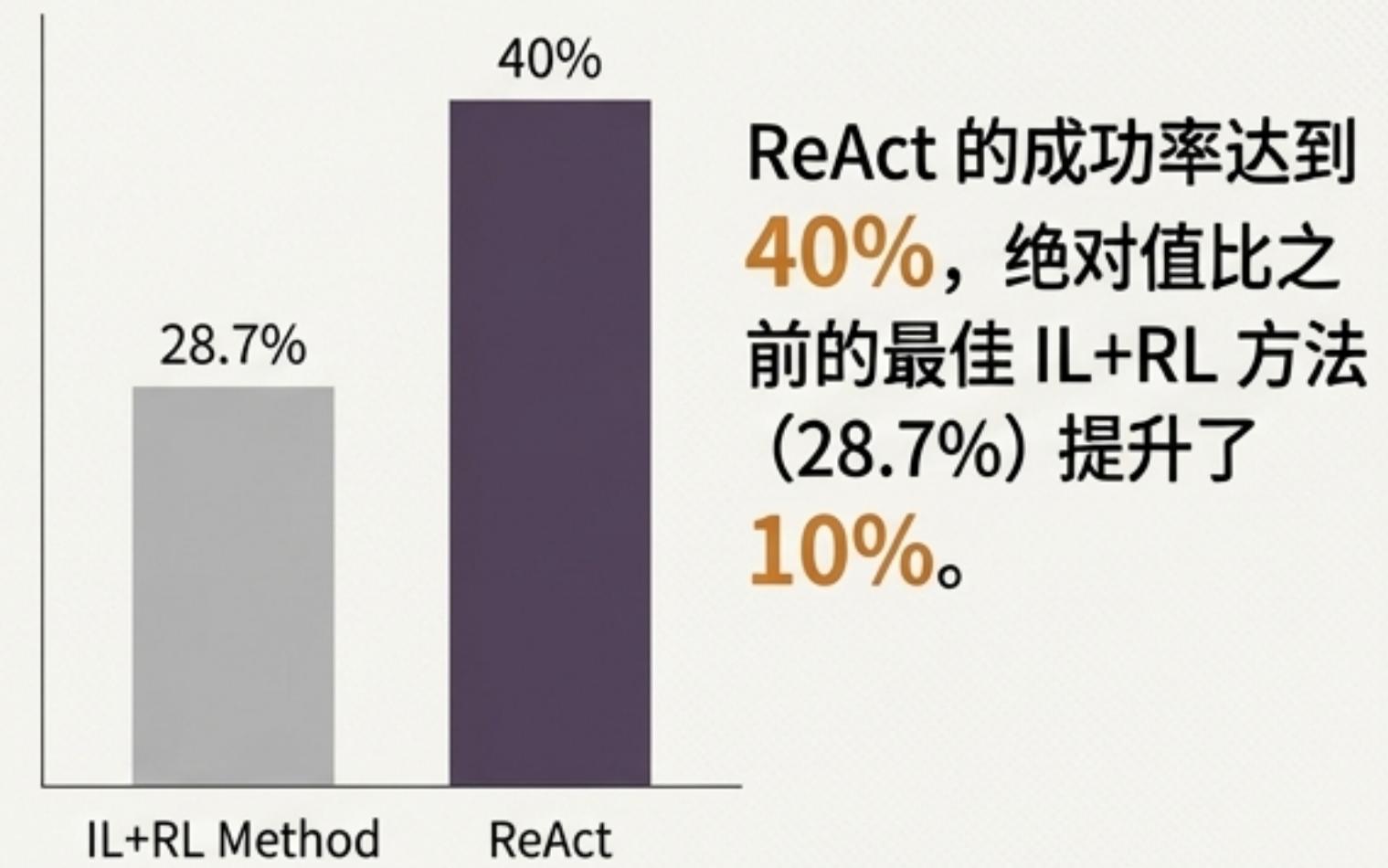
仅用1-2个上下文示例，ReAct的表现远超经过大规模训练的模仿学习(IL)和强化学习(RL)基线。

ALFWorld 成功率



ReAct 的成功率达到 **71%**，显著高于 Act-Only (45%) 和在 10^5 专家轨迹上训练的 BUTLER 模型 (37%)，绝对提升达 **34%**。

WebShop 成功率



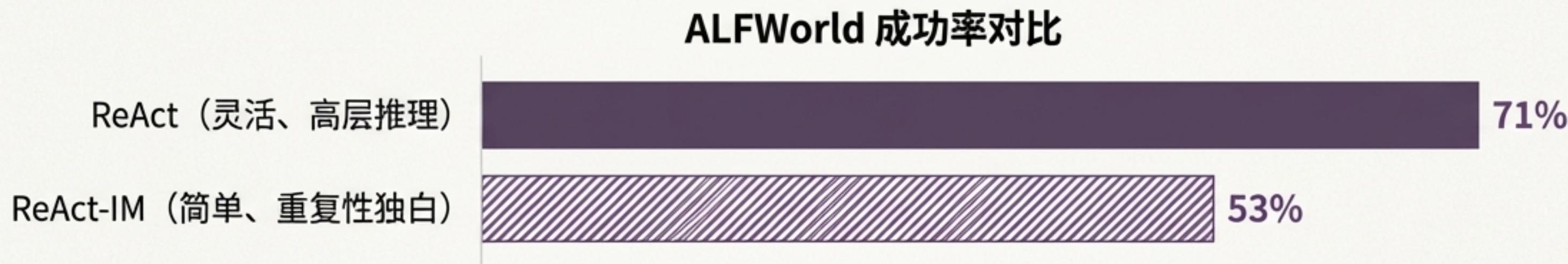
ReAct 的成功率达到 **40%**，绝对值比之前的最佳 IL+RL 方法 (28.7%) 提升了 **10%**。

“这证明了ReAct范式具有强大的通用性和样本效率，能够将LLM的常识知识和规划能力有效应用于复杂的交互式环境中。”

ReAct的思考深度：超越简单的环境反馈

ReAct的优势仅仅来自于外部观察的反馈，还是其内部生成的复杂推理至关重要？

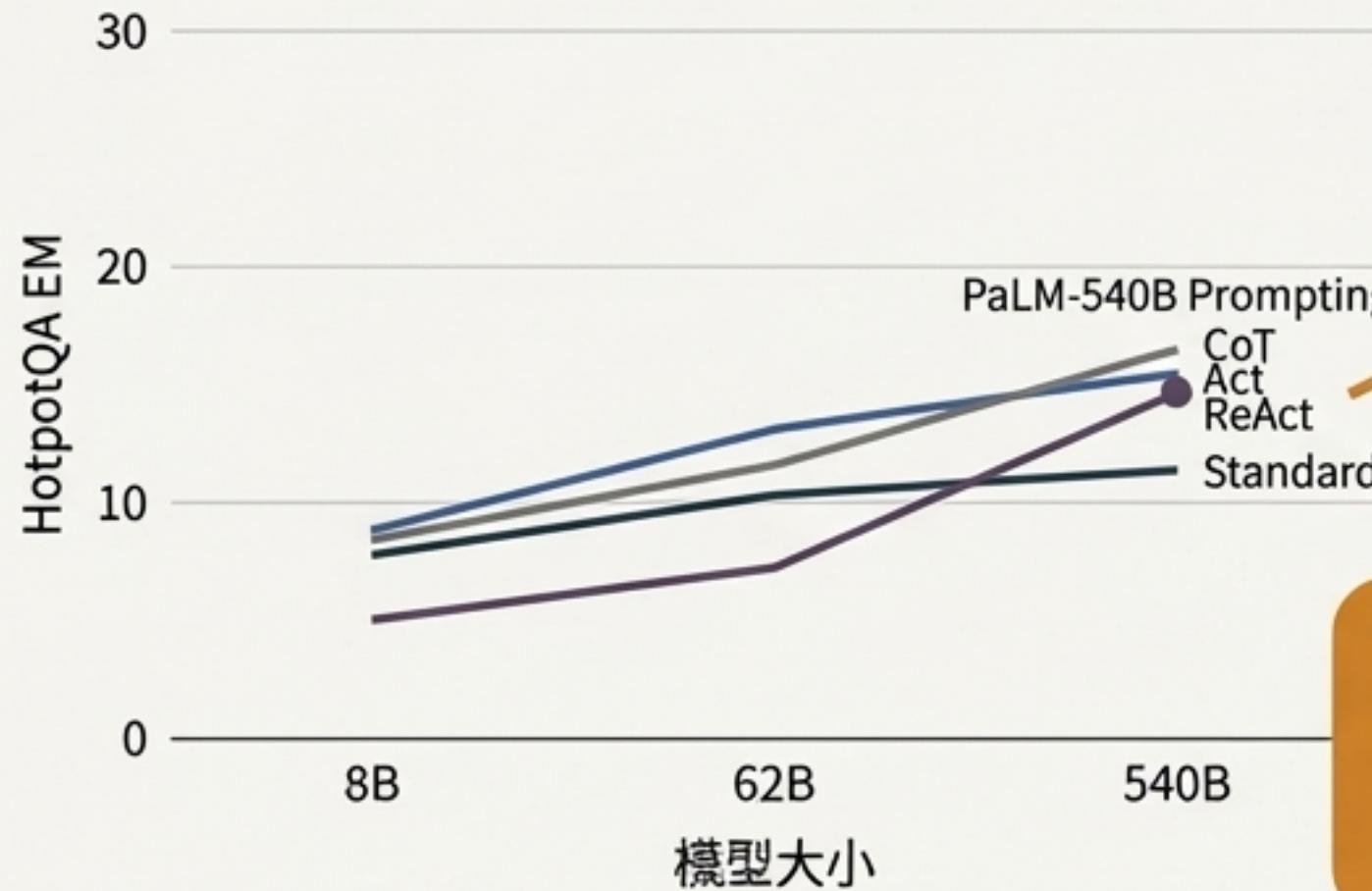
将 ReAct 与一种更简单的“内部独白（Inner Monologue）”式思考进行对比。这种独白仅限于重复当前目标和环境状态，缺乏高层规划和常识推理。



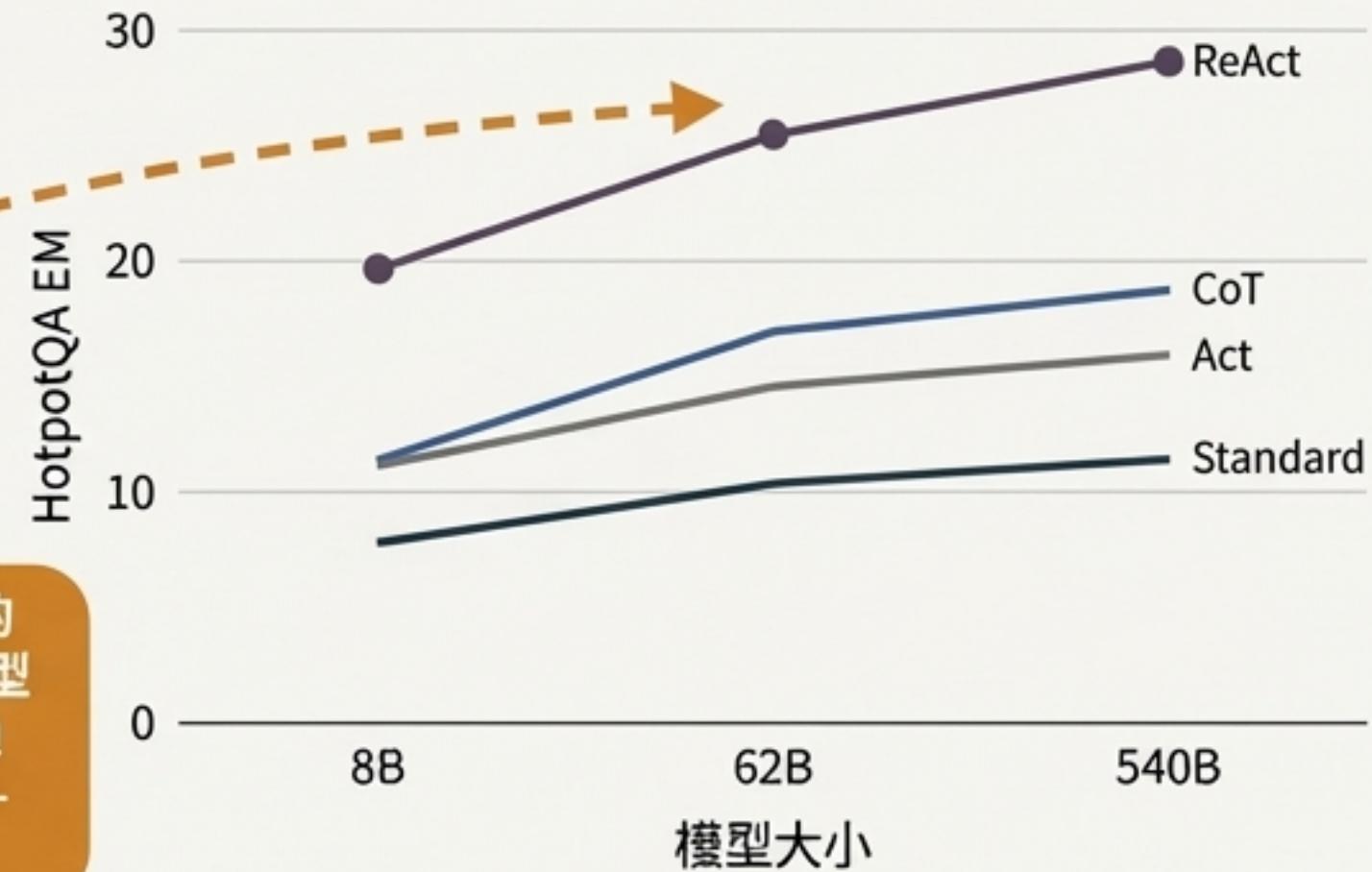
分析：“ReAct的表现显著优于IM风格的提示（71% vs 53%）。IM风格的轨迹常常因为缺乏高层目标分解和常识知识（例如，‘台灯通常在哪里’）而失败。这证明了ReAct生成的灵活、多样的内部推理是其成功的关键。”

ReAct 的可扩展性：从提示学习到微调的飞跃

Prompting (少量样本提示)



Finetuning (微调)



惊人结果：“微调后的
PaLM-62B ReAct 模型
的性能甚至超过了通
过提示学习的 PaLM-
540B 模型。”

观测：在小模型（8B/62B）上，通过少量样本提示来学习ReAct（同时学习推理和行动）是困难的，性能不如更简单的CoT或Act方法。

观测：经过仅仅3,000个样本的微调后，ReAct成为所有方法中表现最好的。

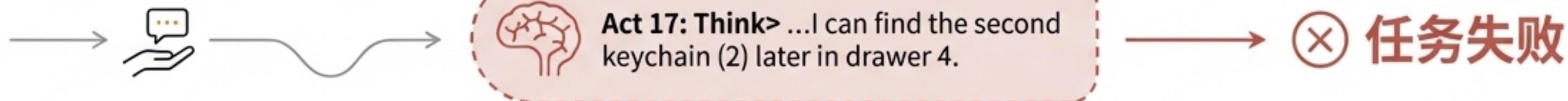
“微调教会了模型一种可泛化的技能——如何协同思考与行动来获取信息，而不是仅仅记忆知识。这表明ReAct范式具有巨大的潜力，可以通过更多高质量数据进一步提升能力。”

更值得信赖的AI：ReAct 提升了可解释性与可控性

ReAct生成的思考轨迹为我们打开了模型的“黑盒”，让我们可以清晰地理解其决策过程。

人机协作修正错误

(a) ReAct 失败轨迹



(b) 人类编辑后的成功轨迹



“通过编辑思想，人类可以轻松地引导和修正模型的行为，这为实现与人类意图对齐的、更安全、更可控的AI代理开辟了新的可能性。”

ReAct 范式总结



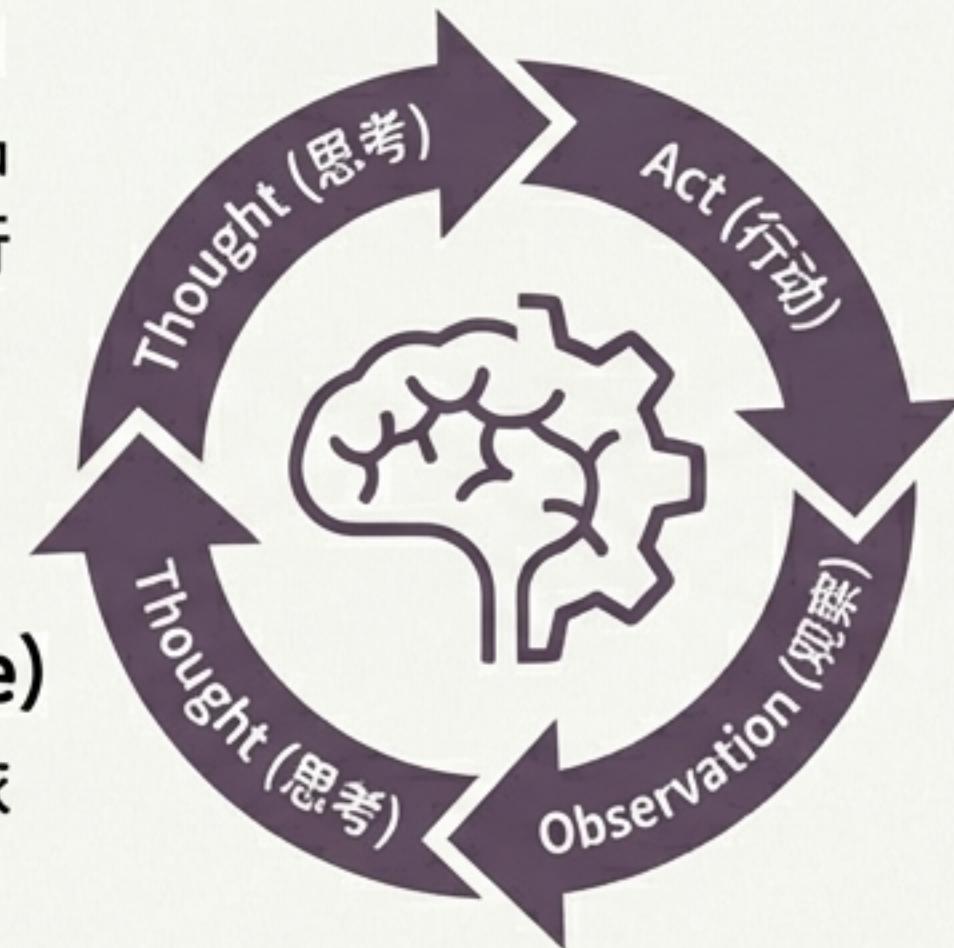
更高性能 (Performant)

在知识推理和决策制定任务中均超越了以往的纯推理或纯行动方法。



更易理解 (Interpretable)

思考轨迹提供了清晰的决策依据，使模型行为透明化。



更强适应性 (Robust)

通过与外部世界交互获取事实，减少幻觉，并根据环境反馈动态调整策略。



更易控制 (Controllable)

允许人类通过编辑思考过程来引导和修正智能体的行为。

What is ReAct?: 一个通用、简单且有效的框架，通过交错生成思考和行动，将LLM的推理能力与交互能力协同起来。

“ReAct 使语言模型能够真正地‘边想边做’，实现了思考与行动的紧密结合。”

未来之路：迈向更通用的智能体

ReAct 为构建能够像人类一样无缝结合语言推理和物理/虚拟行动的智能体奠定了基础。

Future Directions



多任务扩展 (Scaling with Multi-Task Training) :

在更多样的任务上训练ReAct，以提升其通用性。



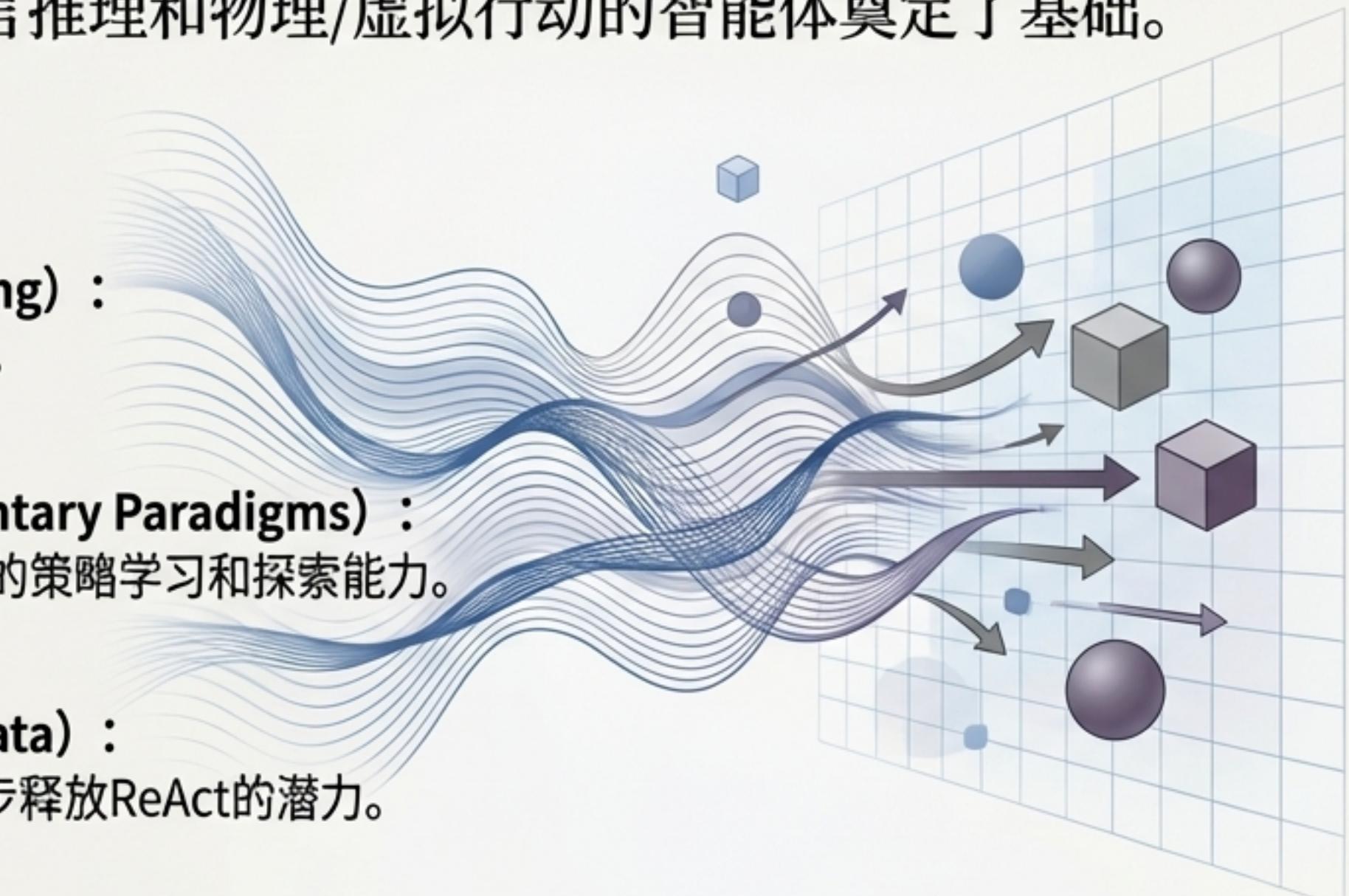
融合互补范式 (Combining with Complementary Paradigms) :

将ReAct与强化学习等方法结合，以实现更强大的策略学习和探索能力。



高质量数据 (Learning from High-Quality Data) :

通过更多高质量的人类标注轨迹进行微调，进一步释放ReAct的潜力。



“通过协同思考与行动，我们正在解锁大型语言模型的全部潜力，构建能够更深入地理解世界并与之有效互动的下一代人工智能。”