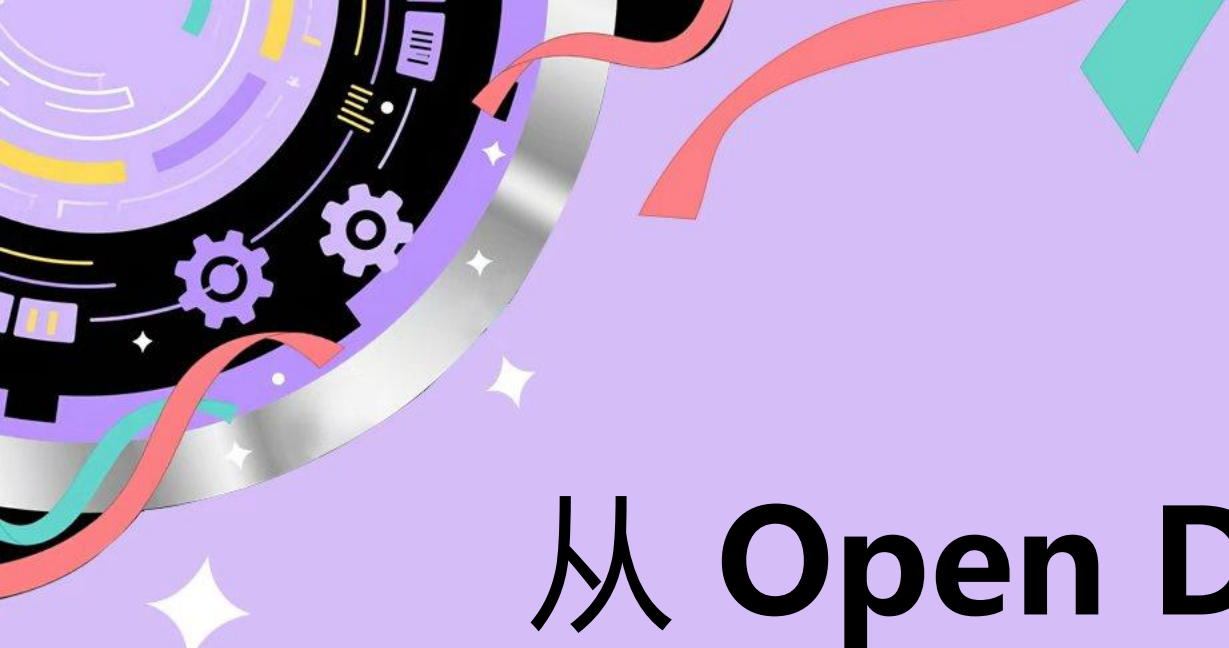




从 Open Deep Research

张海立

LangChain Ambassador

看上下文工程



🦜 LangChain | 🔬 Open Deep Research 开源项目

 langchain-ai / open_deep_research

 | 

[Code](#) [Issues 16](#) [Pull requests 7](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)

 **open_deep_research** Public

 Watch **58** 

 GitIngest

 Fork **1.2k** 

 Starred **8.7k** 

 main  

[Go to file](#)  [Code](#)

 **rlancemartin** Update uv.lock (#191) ✓

b419df8 · 2 weeks ago 

 .github/workflows	Add Claude Code GitHub Workflow (...)	last month
 examples	Enable gpt-4.5-preview	7 months ago
 src	fix: AttributeError: type object 'dateti...	last month
 tests	Add failed gpt5 runs	last month
 .env.example	Clean up env file	2 months ago
 .gitignore	Add evals, MCP, and improvements (...)	3 months ago

About

No description, website, or topics provided.

-  Readme
-  MIT license
-  Activity
-  Custom properties
-  **8.7k stars**
-  **58 watching**
-  **1.2k forks**
- [Report repository](#)

🦜 LangChain | 🔬 Open Deep Research 最新更新

- August 14, 2025: See our free course [here](#) (and course repo [here](#)) on building open deep research.
- August 7, 2025: Added GPT-5 and updated the Deep Research Bench evaluation w/ GPT-5 results.
- August 2, 2025: Achieved #6 (#1 among OSS) ranking on the [Deep Research Bench Leaderboard](#) with an overall score of 0.4344.
- July 30, 2025: Read about the evolution from our original implementations to the current version in our [blog post](#).
- July 16, 2025: Read more in our [blog](#) and watch our [video](#) for a quick overview.



DeepResearch Bench: A Comprehensive Benchmark for Deep Research Agents

The research aims to comprehensively evaluate the capabilities of Deep Research Agents.

[Code](#) | [Website](#) | [Paper](#) | [Eval Dataset](#) | Total models: 21 | Last Update: 02 August 2025

Race judge model: gemini-2.5-pro | Fact-checking models: gemini-2.5-flash

🏆 Leaderboard

🔗 Side-by-Side Viewer

🔍 Data Viewer

💬 Prompt-to-Leaderboard

Model Search

Entering model name to search...

Model Categories

☒ Deep Research Agent

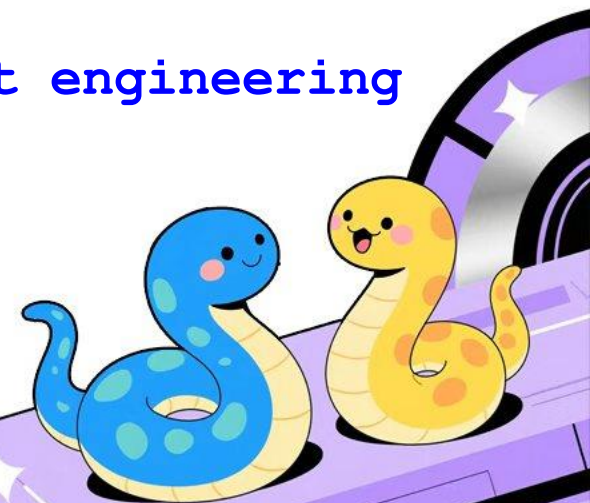
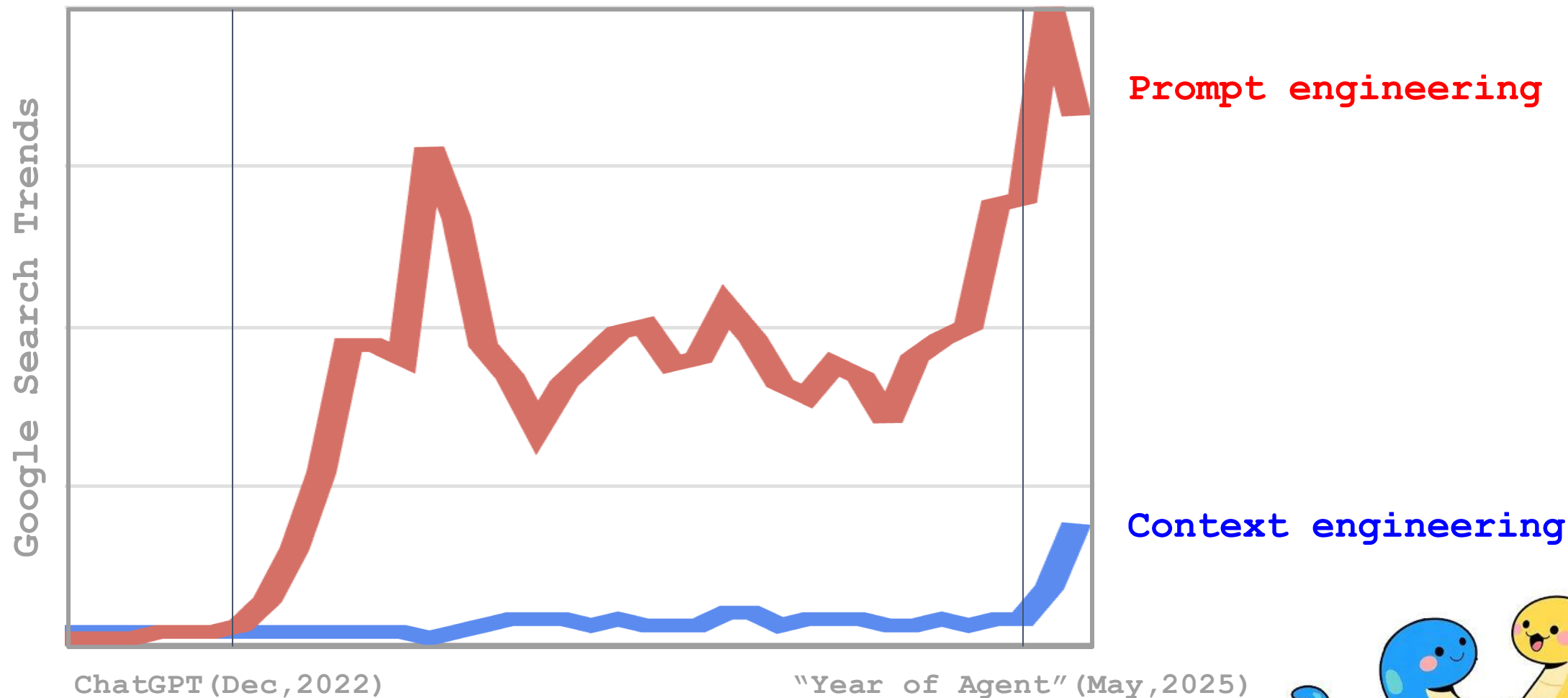
☒ LLM with Search

Rank	model	overall	comp.	insight	inst.	read.	c.acc.	eff.c.	categ
1	🚀 gemini-2.5-pro-deepresearch	49.71	49.51	49.45	50.12	50	78.3	165.34	Deep
2	🚀 openai-deepresearch	46.45	46.46	43.73	49.39	47.22	75.01	39.79	Deep
3	🚀 claude-research	45	45.34	42.79	47.58	44.66	-	-	Deep
4	🚀 kimi-researcher	44.64	44.96	41.97	47.14	45.59	-	-	Deep
5	🚀 doubao-deepresearch	44.34	44.84	40.56	47.95	44.69	52.86	52.62	Deep
6	🚀 langchain-open-deep-research	43.44	42.97	39.17	48.09	45.22	-	-	Deep
7	nvidia-aiq-research-assistant	40.52	37.98	38.39	44.59	42.63	-	-	LLM w
8	🚀 perplexity-Research	40.46	39.1	35.65	46.11	43.08	82.63	31.2	Deep
9	🚀 grok-deeper-search	38.22	36.08	30.89	46.59	42.17	73.08	8.58	Deep
10	sonar-reasoning-pro	37.76	34.96	31.65	44.93	42.42	45.19	9.39	LLM w

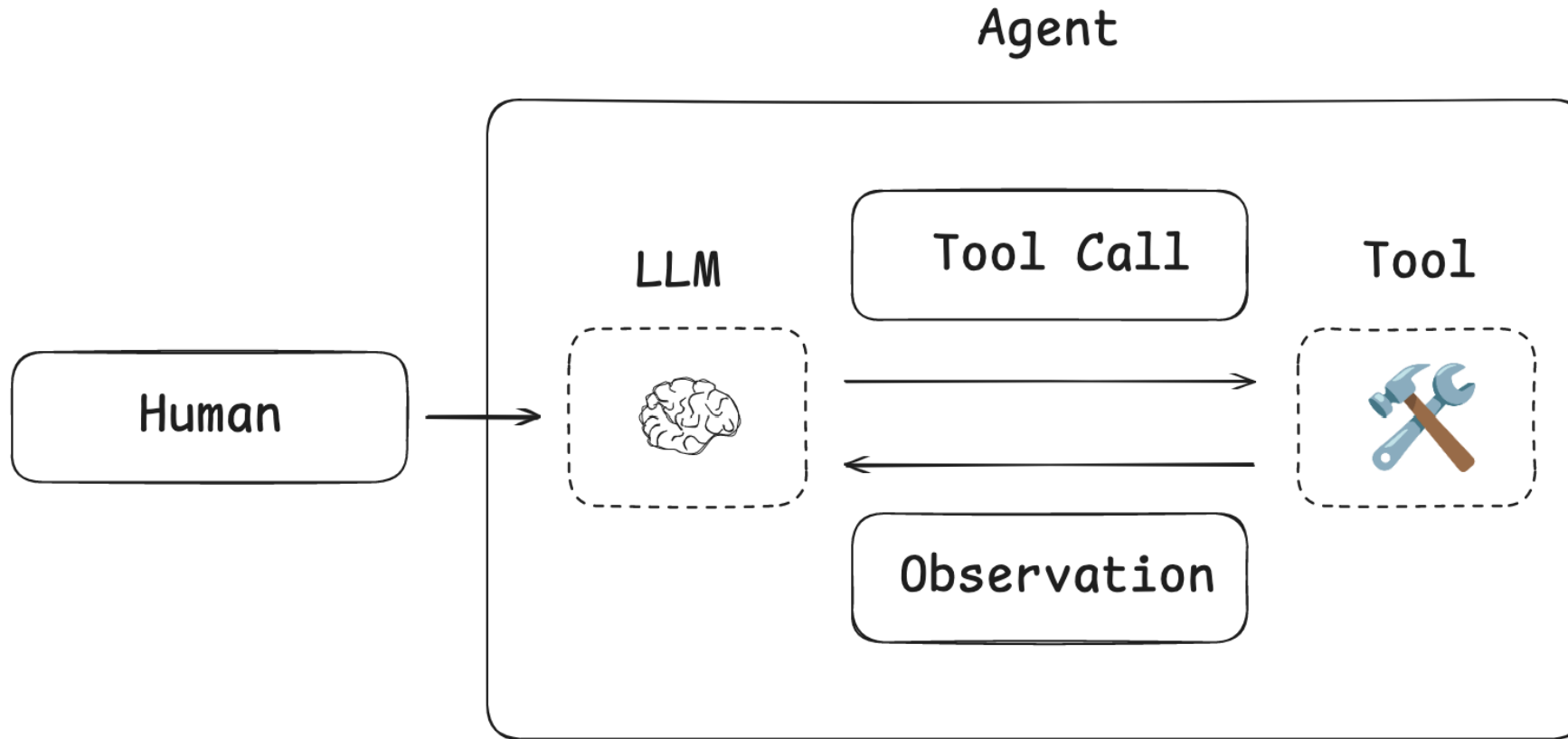
- 01 上下文工程的源起
- 02 上下文工程的策略
- 03 上下文工程的实践



Context Engineering 是何时出现的？



Context Engineering 为何会出现？



Context Engineering 没有又会如何？

- 上下文污染

例：Gemini 在玩《宝可梦》时出现幻觉，虚构了一个道具并尝试重复使用它。

- 上下文干扰

例：当上下文超过 10 万 Token 时，Gemini 更倾向于重复动作而不是制定新计划。

- 上下文混淆

例：模型在使用更多工具时表现更差，尤其是当工具相似时。

- 上下文冲突

例：如果模型连续调用的工具产生矛盾，则表现会更差。



Context Engineering 有官方定义吗？



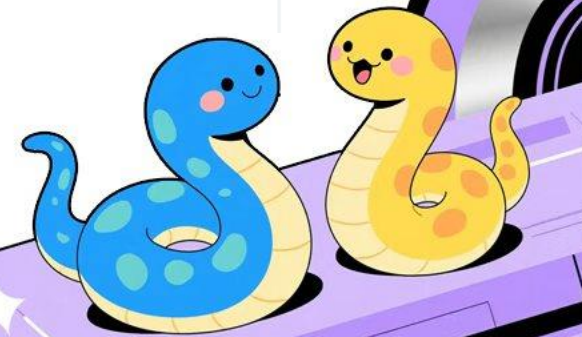
Andrej Karpathy ✓

@karpathy



+1 for "context engineering" over "prompt engineering".

People associate prompts with short task descriptions you'd give an LLM in your day-to-day use. When in every industrial-strength LLM app, context engineering is the delicate art and science of filling the context window with just the right information for the next step. Science



策略 1 Offload Context)

- 使用文件系统记录笔记 (参考: Anthropic 多智能体系统)
- 使用文件系统 (例如: todo.md) 规划/跟踪进度 (参考: Manus)
- 使用文件系统读写高负载上下文 (参考: Manus)
- 使用文件存储长期记忆 (参考: [LangGraph Ambient Agents 课程](#))



策略2：缩减上下文 (Reduce Context)

- 总结智能体消息历史记录 (参见: Claude Code)
- 修剪消息历史中不相关的部分 (参见: [Drew 的文章](#))
- 总结/修剪工具调用的输出内容 (参见: [open-deep-research 项目](#))
- 在智能体与智能体交接时进行总结/修剪 (参见: [Cognition 的文章](#))
- 但要注意避免信息丢失 (参见: Cognition 和 Manus)



策略3：检索上下文 (Retrieve Context)

- 结合多种检索方法并进行重新排序 (参考: [Varun](#) 在 Windsurf 的观点)
- 将检索内容组合成提示的系统 (参考: Cursor 中的 [Preempt](#))
- 根据工具描述检索相关工具 (参考: Drew 的帖子)



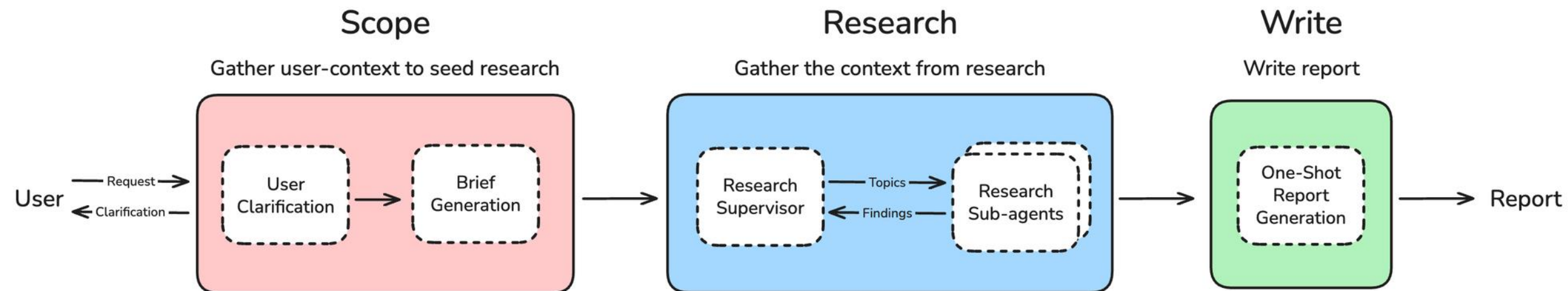
策略4：隔离上下文 (Isolate Context)

- 在多智能体之间分割上下文 (参考: Drew的文章, Anthropic)
- 但要谨慎 (参考: Cognition / [Walden Yan](#) 的推文)
- 多智能体可能会做出冲突的决策 (参考: Cognition / Walden Yan)
- 如果子智能体避免做决策, 则风险较低 (参考: open-deep-research)



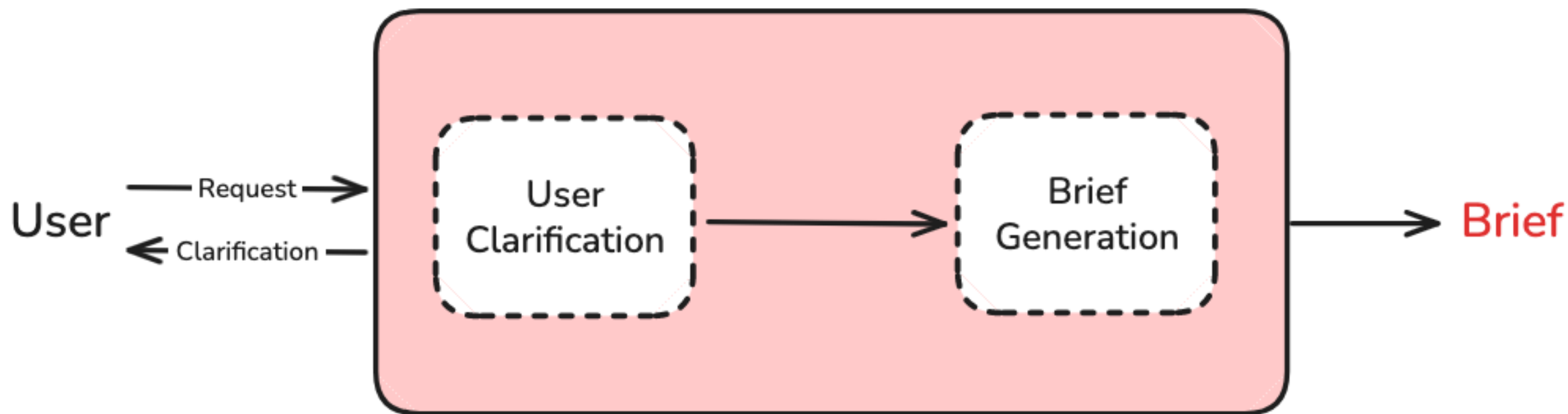


Open Deep Research 的上下文管理实践

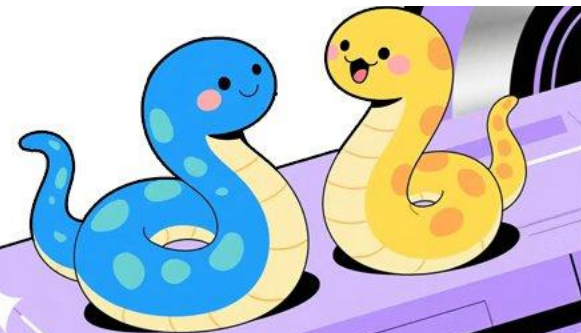
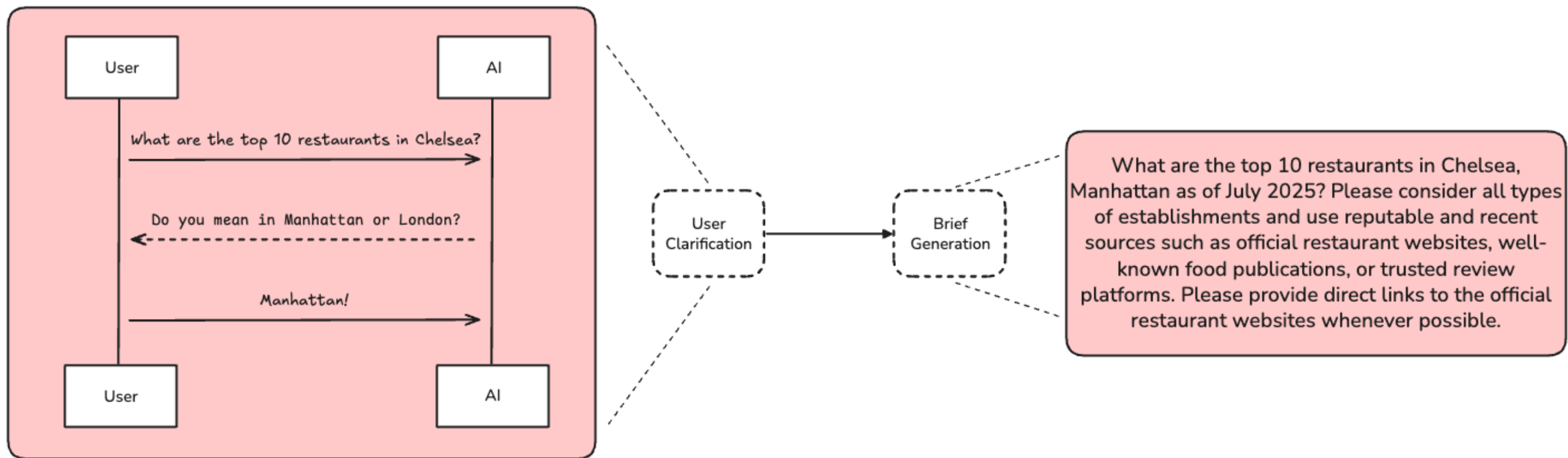




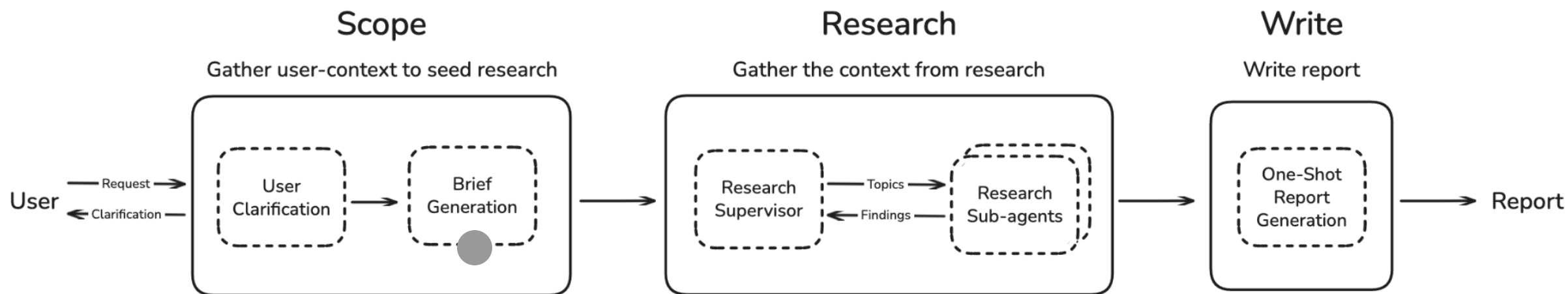
Open Deep Research 的上下文管理 | 确定议题



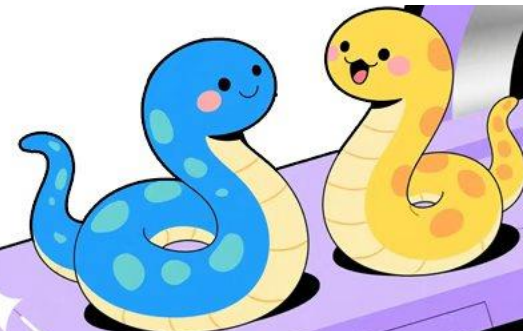
Open Deep Research 的上下文管理实践 | 确定议题



Open Deep Research 的上下文管理实践 | 确定议题

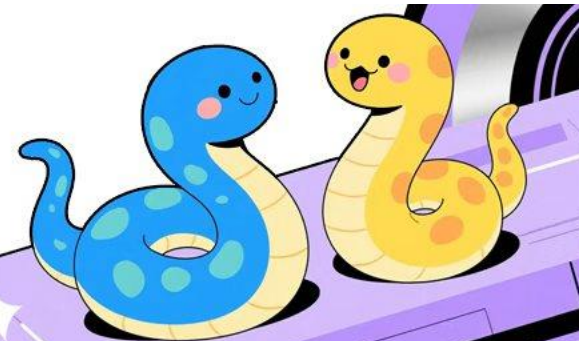
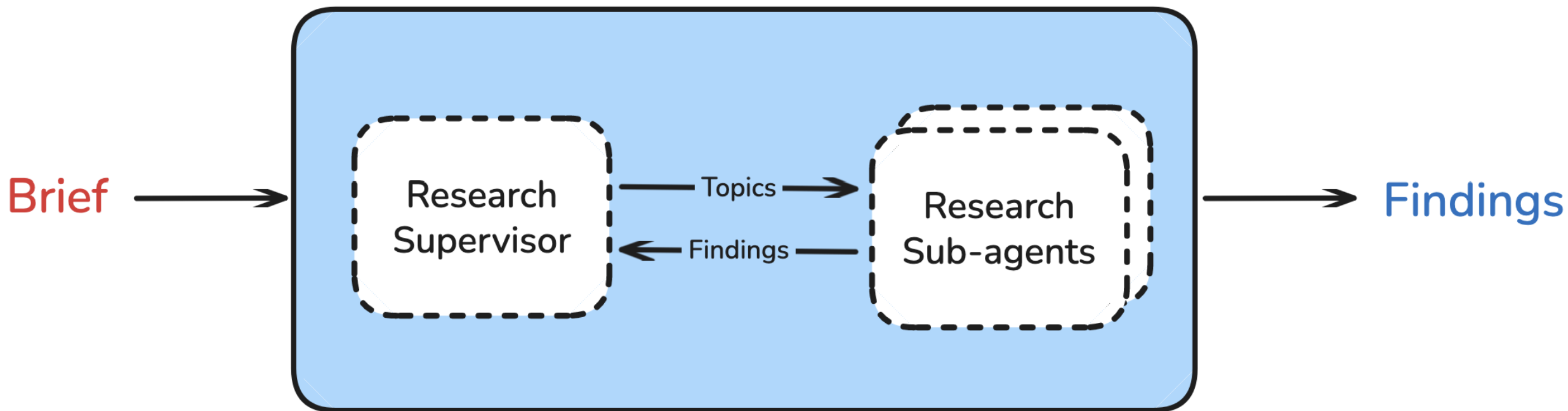


Offload: Create brief from chat and save to state.



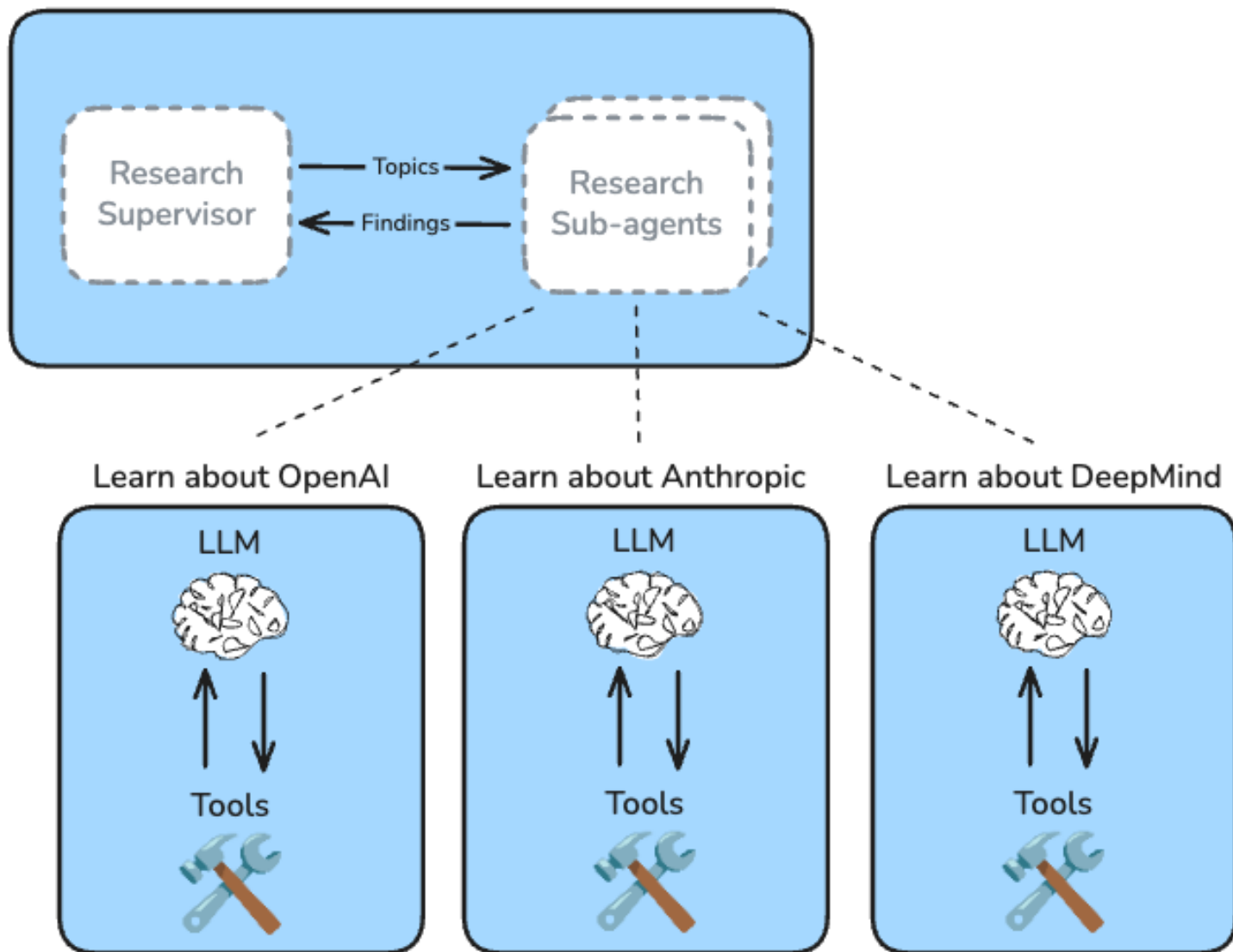


Open Deep Research 的上下文管理 | 执行调研

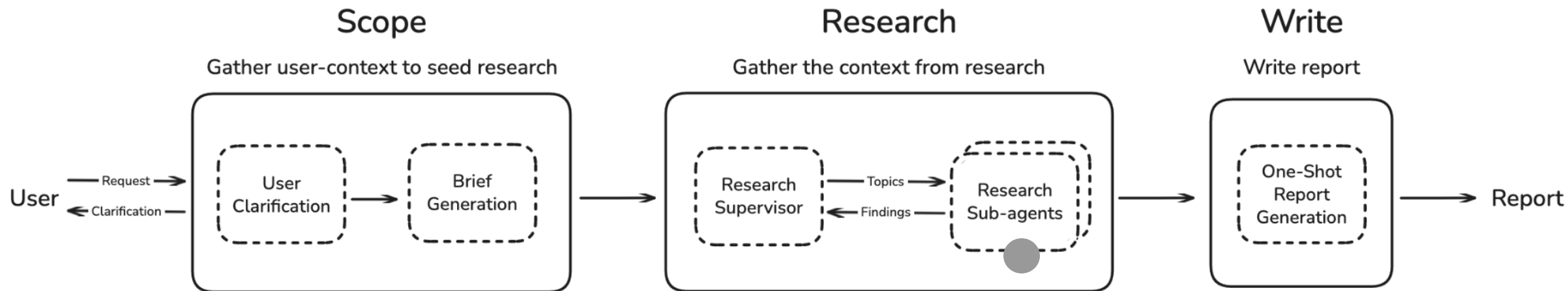




Open Deep Research 的上下文管理 | 执行调研



Open Deep Research 的上下文管理实践 | 执行调研



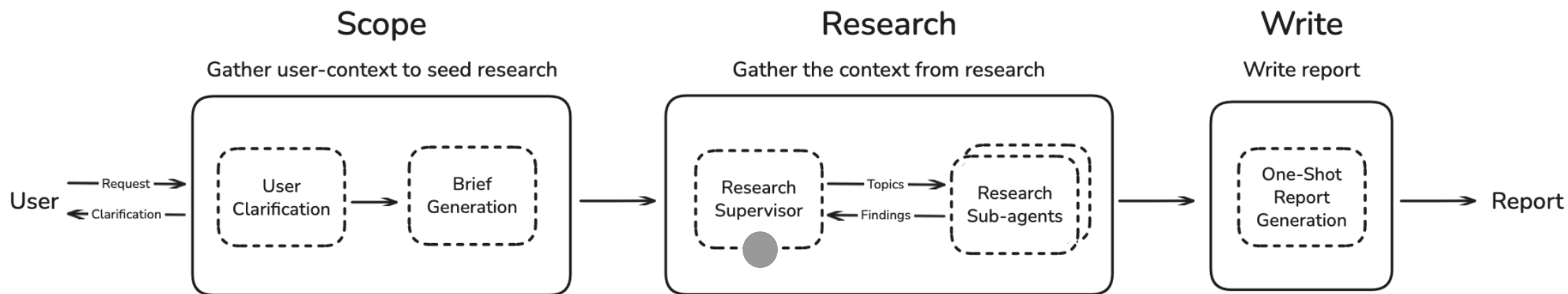
Offload: Use brief later to steer writing/research.

Isolate: Isolate context across sub-agents.

Reduce: Compress findings before passing to supervisor.



Open Deep Research 的上下文管理实践 | 执行调研

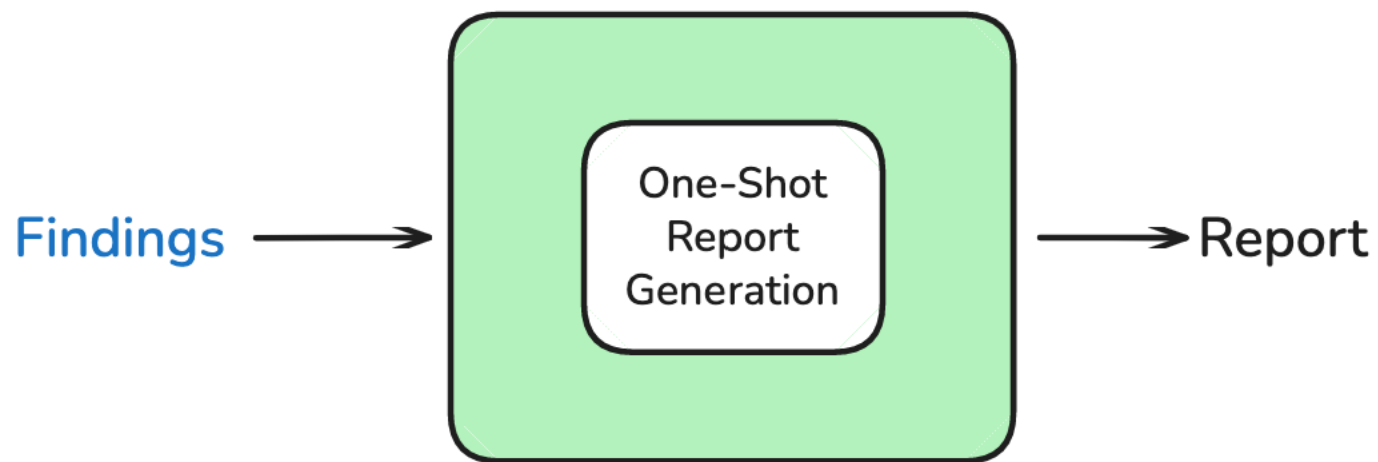


Retrieve: Pick proper tools or sub-agents from a big pool.



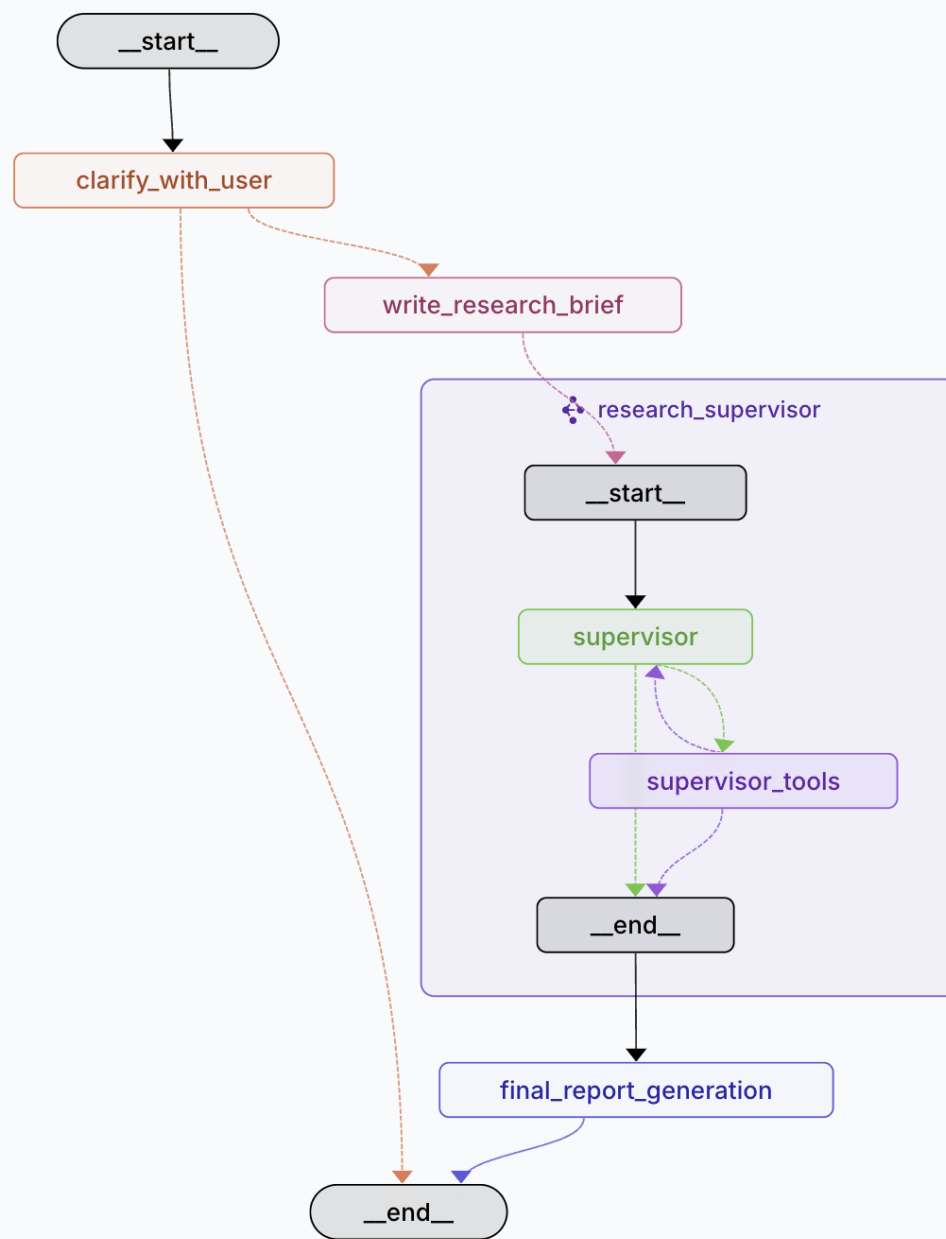


Open Deep Research 的上下文管理 | 编写报告





Open Deep Research 的上下文管理 | 状态流图



行业内的一些实践汇总

	Offload	Reduce	Retrieve	Isolate	Cache
Drew's Post	Discussed	Discussed	Discussed	Discussed	Not discussed
Manus	Used	Discouraged [1]	Not discussed	Not discussed	Used
Anthropic-researcher	Used	Used	Used	Used	Not discussed
Cognition	Not discussed	Used	Not discussed	Discouraged [2]	Not discussed
LC open-deep-research	Used	Used	Used	Used	Not discussed

