

# 《2026 最新Claude Skill 仓库合集》

大家好，我是吴哥，专注AI编程、AI智能体。立志持续输出、帮助小白轻松上手AI编程和AI智能体。

2026年了，吴哥汇总了最新仓库合集。

本文重点内容：用一张对比表带你选对最热 Skill 仓库、搞清 Skills vs Prompt vs MCP，并在 10 分钟内装好一套“能持续复用”的技能栈。建议看完就去做：先装 3 个，再写 1 个自己的。

- 阅读信息：预计 8 分钟 | 适用人群：小白/进阶

记住，别再把“会写提示词”当能力，把“能复用流程”才叫落地，真实力。

## 核心问题清单

- Q1：Skills 到底是什么？和 Prompt、MCP 有什么本质区别？
- Q2：这么多仓库，谁是“生产级样板”，谁是“百科导航”，谁是“黑马能力”？
- Q3：怎么避免“装得越多越卡”？怎么管理版本、审计风险、持续升级？

先搞清“选型逻辑”，再谈“装哪些仓库”。

## 工具卡

- anthropics/skills（官方示范库）
  - 用途：直接提供一批可运行的 Skills（文档处理、测试、工作流等），也是“Skills 应该怎么写”的参考实现。
  - 适合：第一次入坑、想要“拿来就能跑”的人；也适合作为团队内 Skill 编写规范的模板。
- ComposioHQ/awesome-claude-skills（高密度导航/分类）
  - 用途：把大量技能按场景分类，提供“从需求到技能”的检索入口（你不需要自己在 GitHub 大海捞针）。
  - 适合：已经知道自己要做什么（如：代码审查、TDD、文档流水线），需要快速选技能的进阶用户。
- agentskills/agentskills（开放标准/规范与SDK）
  - 用途：当你想让技能“写一次，多处可用”（Claude / Copilot / VS Code 等），这里是规范与参考 SDK 的总开关。
  - 适合：团队/组织要落地；你不想被单一平台锁死。

官方库教你“怎么做对”，导航库帮你“选得更快”，标准库保证你“走得更远”。

## 底层逻辑：

- 判断：**Skills** 解决的不是“模型会不会写代码”，而是“你的流程能不能被稳定复用”。
- 证据：Anthropic 明确把 Skills 定义为“文件夹化的指令/脚本/资源”，并强调三层渐进式加载：先加载元数据、触发后加载指令、必要时再加载附加文件；这样你可以装很多技能而不一次性挤爆上下文。
- 启示：很多人说“AI 编程不落地”，其实是把问题当成“换个更强模型”——但真正的瓶颈是**缺少可复用的工程做法**（测试、评审、发布、规范、回滚、文档）。Skills 让这些做法变成“随用随取”的技能卡片。

建议：先用官方技能跑通一个闭环，再把你自己的团队流程固化成 **1 个 Skill**。

2026 热门仓库对比表				
仓库	星标	核心价值	最适合谁	一句话评价
anthropics/skills	44.2k	官方“生产级示范 + 可直接用”的技能集	新手/团队起步	先抄作业，再改成你自己的流程
ComposioHQ/awesome-claude-skills	20.9k	高密度分类导航（选技能更快）	进阶/多场景	你不缺技能，你缺“目录”
agentskills/agentskills	6.1k	<b>开放标准</b> ：规范/文档/SDK	团队/平台化	写一次，多处跑，降低锁定
muratcankoylan/Agent-Skills-for-Context-Engineering	7.3k	上下文工程（诊断/优化/评估）	想把Agent做稳的人	<b>不做上下文工程，Agent必然后漂</b>
heilcheng/awesome-agent-skills	1.4k	跨平台“技能/教程/工具”百科	想全景扫描	适合“逛”，别一口气全装
gotalab/skillport	229	技能管理/分发（CLI/MCP 思路）	多机器/多项目	管理一次，多处同步
kirodotdev/powers	137	为 Kiro 代理做的“按需加载能力包”	Kiro 深度用户	解决“工具太多=上下文过载”
K-Dense-AI/claude-scientific-skills	6.4k	科学/数值/数据/ML 向技能集合	科研/量化/工程分析	技能栈一旦专业化，效率会断层领先

三个最常见误区（以及怎么避免）

1. 误区：装得越多越强纠正：Skills 的设计原则就是“渐进式披露”，理论上你可以装很多，但前提是**只在需要时触发加载**；否则你会把上下文浪费在“用不到的说明书”上。
2. 误区：**Skills = Prompt** 模板纠正：Prompt 更像一次性对话指令；Skills 是“文件系统里的可组合资源”（说明 + 脚本 + 模板），可以被多个工具重复触发。
3. 误区：只看星标不看风险纠正：官方工程博客直接提醒“恶意 skills 风险”，尤其是带脚本/依赖/外联指令的技能，装之前要审计。

## 10 分钟安装：一条命令把技能放到正确位置

不同平台的“技能目录”略有差异，但核心就两类：**项目级**（跟仓库走）与**个人级**（跟你走）。GitHub Copilot 的文档把这两类路径讲得很清楚（例如 `.claude/skills`、`.github/skills`、`~/.claude/skills` 等）。

最通用的做法（个人级）：

```
mkdir -p ~/.claude/skills
git clone <你选中的技能仓库> ~/.claude/skills/<repo-name>
```

（路径与“Skill 是文件夹 + SKILL.md”这一结构保持一致即可。

如果你懒得手动装，社区也在做“统一安装器”方向（例如 OpenSkills 的 npx 工具），适合“先跑起来再说”的人。

## 可复制提示词：让 AI 帮你“选 3 个技能 + 组 1 条 workflow”

**Prompt：技能选型（从仓库到清单）**

```
你是我的“Agent Skills 选型顾问”。我的场景是：【在这里写你的场景】。
目标：用最少技能跑通一个可复用闭环（10-30分钟内可验证）。
请输出：
1) 我应该优先选的 3 个仓库（说明理由：生产级/导航/标准/管理各取其一）
2) 每个仓库里我应该先装的 2-3 个技能（写出用途/触发关键词/预期产出）
3) 一个最短 workflow：输入 -> 过程 -> 输出（含验证点）
约束：不要推荐“装很多”，只给能立刻用的最小集合。
```

**Prompt：把你的团队流程固化成第一个 SKILL.md**

```
你是“Skill 作者”。我要把下面流程固化成一个 Skill：
流程名称：【例如：PR 代码评审+单测补齐+变更说明】
输入信息：【仓库结构/语言/测试框架/规范链接或片段】
输出要求：【产物格式：Checklist、PR 模板、命令、报告结构】
请生成：
1) SKILL.md 的 YAML frontmatter (name/description)
2) SKILL.md 主体：分步骤、含检查点、含失败回滚策略
3) 可选：建议我拆分成哪些附加文件（比如 reference.md / checklist.md）
```

这套结构正是官方“SKILL.md + 元数据 + 渐进加载”的推荐方式。

合集清单：

[1]: <https://platform.claude.com/docs/en/agents-and-tools/agent-skills/overview> "Agent Skills -

Claude Docs"

备注：官方文档 | 解释 Skills 是什么、基本概念与使用方式 | 适合作为“定义/规则/术语”权威出处（非开源仓库）

[2]: <https://www.anthropic.com/engineering/equipping-agents-for-the-real-world-with-agent-skills> "Equipping agents for the real world with Agent Skills \ Anthropic"

备注：官方工程博客 | 讲清“为什么要 Skills（渐进式披露/可治理）”的设计动机与方法论 | 适合引用做“趋势与原理”（非开源仓库）

[3]: <https://github.com/anthropics/skills> "GitHub - anthropics/skills: Public repository for Agent Skills"

备注：开源仓库（官方精选） | 生产级 Skills 示例与最佳实践 | 新手抄作业/团队定规范首选

[4]: [https://github.com/anthropics/skills?utm\\_source=chatgpt.com](https://github.com/anthropics/skills?utm_source=chatgpt.com) "anthropics/skills: Public repository for Agent Skills"

备注：同 [3]（带参数链接） | 建议正文引用时统一用 [3]，避免“带追踪参数”

[5]: <https://github.com/ComposioHQ/awesome-claude-skills> "GitHub - ComposioHQ/awesome-claude-skills: A curated list of awesome Claude Skills, resources, and tools for customizing Claude AI workflows"

备注：开源仓库（Awesome 导航） | 分类清晰、覆盖面广 | 你知道要做什么时，用它快速“找技能/找资源”

[6]: [https://github.com/agentskills?utm\\_source=chatgpt.com](https://github.com/agentskills?utm_source=chatgpt.com) "Agent Skills" 备注：GitHub 组织主页 | 聚合标准/生态相关仓库入口 | 适合做“生态

[7]: <https://github.com/ComposioHQ/awesome-claude-skills/activity> "Activity · ComposioHQ/awesome-claude-skills ·

GitHub" 备注：活动页（更新强度） | 看维护是否活跃、近期提交频率 | 写“热门/持续更新”时可用作旁证

[8]: <https://github.com/muratcankoylan/Agent-Skills-for-Context-Engineering/activity> "Activity · muratcankoylan/Agent-Skills-for-Context-Engineering · GitHub"

备注：活动页（黑马仓库更新强度） | 用于判断作者是否持续迭代 | 建议配合主仓库页一起引用

[9]: <https://github.com/heilcheng/awesome-agent-skills> "GitHub - heilcheng/awesome-agent-skills: A curated list of skills, tools, tutorials, and capabilities for AI coding agents (Claude, Codex, Copilot, VS Code)" 备注：开源仓库（跨平台 **Awesome**） | 覆盖 Claude/Codex/Copilot/VS Code 等 | 适合做“全景扫描”，不建议一口气全装

[10]: <https://github.com/gotalab/skillport> "GitHub - gotalab/skillport: Bring Agent Skills to Any AI Agent and Coding Agent — via CLI or MCP. Manage once, serve anywhere."

备注：开源仓库（技能管理/分发工具） | 更偏“管理器/同步器”，不是技能合集本体 | 适合多项目、多机器统一管理

[11]: <https://kiro.dev/powers/> "Powers - Kiro"

备注：产品能力页 | Kiro 的 Powers（类 Skills 能力包）说明与用法 | 适合对比“Claude Skills vs Kiro Powers”（非开源仓库）

[12]: <https://github.com/K-Dense-AI/claude-scientific-skills/blob/main/docs/open-source-sponsors.md> "claude-scientific-skills/docs/open-source-sponsors.md at main · K-Dense-AI/claude-scientific-skills · GitHub"

备注：开源仓库内文档页 | 赞助/开源声明等信息入口 | 若要介绍科学技能集，

[13]: <https://docs.github.com/copilot/concepts/agents/about-agent-skills> "About Agent Skills - GitHub Docs" 备注：官方文档（**GitHub Copilot Agents**） | 从 Copilot 视角解释 Agent Skills | 用于论证“Skills 跨平台落地”（非开源仓库）

[14]: [https://github.com/numman-ali/openskills?utm\\_source=chatgpt.com](https://github.com/numman-ali/openskills?utm_source=chatgpt.com) "numman-ali/openskills: Universal skills loader for AI coding ..."

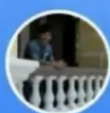
备注：开源仓库（通用 **Skills Loader**） | 偏“加载器/运行时”，解决不同平台/目录结构的兼容问题 | 适合工程化落地与自动安装脚本化

这是今天分享内容，希望整理内容对你有所帮助，感谢阅读。



该二维码7天内(1月11日前)有效，重新进入将更新

如果你对AI编程感兴趣，欢迎交流，进群领取吴哥AI编程手册详细资料福利(PS:群已超200人)。要是觉得今天这碗饭喂得够香，随手点个赞、在看、转发三连吧！



吴可行

我已加入AI破局俱乐部  
邀你免费体验三天 >>>



扫码即可领取