# 新时代使用 AI 心法

五大维度、三大原则、四大象限——把 AI 用成工具，而不是被它用

费曼式写法 · 由多个 AI 代理撰写与互相审校

# 导读

五大维度、三大原则、四大象限——把 AI 用成工具，而不是被它用

先别急着打开 AI，先听我讲一个人的故事。

这个人家里装修，要把两根粗细不同的水管接起来。他琢磨了一下，觉得自己需要一节「五分管」，于是跑遍了整个建材市场——没有，哪儿都没有。他越找越急，越急越找。其实这件事的标准解法，是花十块钱装一个转接阀，两分钟完事。

他缺的不是努力，是一个**对的问题**。他把外行拍脑袋想出的解决方案，当成了问题本身。

这个故事放在今天，有了一个新的放大器：AI。以前你带着错问题去问老师傅，老师傅一句话把你掰回来；现在你带着错问题去问 AI，它会认认真真、高效地把错问题执行到底，还附赠一份格式精美的执行报告。**AI 不会质疑你的问题，它只会放大你的问题——对的放大成成果，错的放大成事故。**

## 这本书讲什么

这本书想讲透一件事：AI 时代怎么用好 AI。核心是三组数字——**五大维度、三大原则、四大象限**。

五大维度是一条流水线：定义问题、制定计划、过程保证质量、结果校验、真实环境反馈。它治 AI 的两大病根——幻觉和上下文。三大原则是三句话：不能被 AI 劫持、AI 不背锅、AI 是你的工作起点。它管的不是流程，是心态——你和这个工具，谁是主人。四大象限是四种处境：你懂且能验证、你不懂、你懂一点但不知怎么验证、你完全陌生——同一套流程，四种玩法。

## 贯穿全书的一个问题

读的时候请始终带着这个问题：**这个又强又不靠谱的助手交上来的东西，我凭什么信、信到什么程度、信了谁负责？**

每章都会从不同角度逼近它。读完最后一章，你手里会有一张每次打开 AI 之前都能过一遍的清单——不复杂，一共九条。但那时候你再读它，每一条背后都站着一个你亲眼见过的坑。

故事从认识你的新助手开始：一个不靠谱的博士生。

# 第一章 · 一个不靠谱的博士生

假设你招了一个博士生。他读过人类有史以来几乎所有公开的文字，你问什么他都答得上来，写报告、写代码、写方案，速度快得离谱，而且不要工资、不闹情绪、随叫随到。听起来像做梦。但入职第一周你就发现了问题：他**不靠谱**。他会面不改色地编一个不存在的参考文献；你上午教过他的规矩，下午他就忘了；他在一个话题里聊得越久，说话越容易被前面聊过的内容带偏。

这个博士生就是 AI。这本书要回答的问题只有一个：**面对一个又强又不靠谱的助手，你怎么用它，而不是被它坑？**

## 两个极端，都是坑

面对这样的助手，人很容易滑向两个极端。一个极端是：「AI 不可信，我全自己干。」这就像买了一个电钻却坚持用手拧螺丝——这个时代最大的生产力红利，你一口没吃到。另一个极端是：「AI 太强了，我全信。」然后某天拿它生成的数字去做汇报，被当场打脸；或者更严重，拿它的判断去做真要命的决定。

举个书里最疼的例子：你采了个蘑菇，拍给 AI 问有没有毒，它说「没毒，可以吃」。你吃了，进了医院，AI 说「对不起，我错了」。**它道歉的成本是零，你承担的成本是全部。**记住这个例子，它会在后面「三大原则」里一再出现。

## 它的病根只有两个

这个博士生所有的不靠谱，拆到最后只有两个病根。

第一个叫 幻觉——说白了就是「一本正经地瞎编」。AI 生成文字的方式，决定了它追求的是「看起来像那么回事」，而不是「确实是真的」。所以它给的任何东西，**默认都先当成「待验证的草稿」**，而不是结论。

第二个病根是 上下文——就是你和它的这次对话里，它能「看见」的所有信息：你的问题、你给的资料、前面的对话记录。上下文对 AI 就像图纸对施工队：图纸给得全、给得对，活

儿就好；图纸缺页、沾了油渍，它就照着错的干。麻烦在于上下文永远「不够」——它不知道你没说出口的背景，而且同一次对话聊久了，早先的话还会互相污染。

> 大白话
>
> 把这两个病根背下来：**幻觉，让它给的东西不一定靠谱；上下文，决定它干活的图纸质量。**这本书后面讲的每一个招式，都在治这两个病。

## 这本书的地图

对付这两个病根，靠的不是感觉，是一套流程加一套心法：

- **五大维度**——一条流水线：定义问题、制定计划、过程保证质量、结果校验、真实环境反馈。幻觉主要靠前四个管，上下文主要靠管好计划与执行来治。
- **三大原则**——谁说了算、谁背锅、AI 的产物算起点还是终点。这三条是心态，防止你被工具反过来拿捏。
- **四大象限**——你懂或不懂、能验证或不能验证，四种处境下同一套流程的不同玩法。

但整条流水线的第一步，在碰 AI 之前就已经开始了：先搞清楚你要解决的到底是不是个真问题。下一章，从一个满市场找五分管的人讲起。

# 第二章 · 满市场找五分管的人

讲个真事。一个人家里装修，要把两根水管接起来。他量了量，觉得自己需要一节「五分管」，于是跑市场，转了一下午，怎么都买不到——因为市场上根本没有他要的那种规格。其实这事的标准解法，是装一个**转接阀**，十块钱，两分钟搞定。

他错在哪？错在**他自己提了一个解决方案，然后把解决方案当成了问题**。他真正的问题是「把两根管径不同的水管接起来」，而他满世界找的是「五分管」——一个外行拍脑袋想出来的、根本不存在的答案。问题定错了，后面再努力都是白搭。

## 这个坑，AI 时代放大了一百倍

以前你带着错误的问题去问老师傅，老师傅一句话就把你掰回来：「你要五分管干嘛？哦，接管子啊，买个转接阀。」老师傅的经验会纠正你。

但 AI 不会。你问它「哪里能买到五分管」，它会认认真真给你列出五金城地址、网购链接、甚至替代规格对照表——**它默认你问的就是对的问题**。它的本事是把你的问题答得又快又好，而不是质疑你的问题。于是错误的问题不再被拦住，反而被以极高的效率执行下去。这就是为什么在 AI 时代，「定义问题」从一项软技能变成了第一维度。

## 不要在错的层面上解决问题

错误的问题有一种最常见的样子：**层面错了**。员工流失率高，你问 AI「怎么设计一份更好的留人话术」——但层面可能在薪酬体系，话术再好也留不住。孩子数学成绩差，你问「给我出五十道练习题」——但层面可能在他根本没理解分数的概念，题刷得越多越挫败。

大白话

判断层面错没错，有个土办法：把你准备让 AI 干的事说出来，然后问自己一句——**「这件事做成了，我真正的烦恼会消失吗？」**如果答案是「不一定」，你的层面多半就错了。

## 什么叫「真实的问题」

提出一个好问题本身就是件极不容易的事，所以一定要慎重。慎重到什么程度？要保证你手里这个问题是**真实的、有意义的**——真实的意思是它真的存在，不是你想象出来的；有意义的意思是解决它真的能让某件事变好。

找五分管的人，问题既不真实（五分管不存在）也没有意义（就算买到也接不上）。而一个真实的问题长这样：「我要把阳台的进水管接到新洗衣机上，两根管子粗细不一样，怎么办？」——注意，这个问题里**没有解决方案，只有处境和目的**。把解法留给懂行的人（或懂行的 AI）去提，是外行最大的智慧。

那么，手里只有一个模糊的烦恼时，怎样一层层往下挖、挖到那个真实的问题？下一章给你一组可以照做的追问。

# 第三章 · 一层层往下问

上一章留下一个悬念：手里只有一个模糊的烦恼——「团队效率低」「孩子学习跟不上」「产品没人用」——怎么把它挖成一个真实的问题？答案是**追问**。不是随便问，是按一个固定的顺序，一层比一层深地往下问。

## 四层追问

**第一层：问题重要，还是目标重要？目标重要。**问题永远是为目标服务的。同样是「服务器老宕机」，目标是「用户不再流失」和目标是「通过年底的安全审计」，要解决的问题完全不同。先问清目标，很多「问题」会自动变形或消失。

**第二层：目标重要，还是目标背后的人重要？人重要。**目标不会自己存在，它是某个人的目标。老板要的「效率」和员工要的「效率」可能根本不是一回事。不问清楚是谁的目标，你解决的只是你脑补的那个。

**第三层：人有那么多，谁是关键的人？**一件事往往牵涉一群人：拍板的、干活的、受益的、受损的。其中总有一两个关键人——大白话就是「他点头这事才算成」的那个人。找不到他，你的方案做得再好，也递不到能拍板的桌上。

**第四层：关键人背后的组织长什么样？**关键人不是孤立的，他站在一个组织里，带着自己的假设（他默认为真、从不怀疑的东西）、立场（他的位置决定的利益取向）、认知（他懂什么、不懂什么）和真实意图（他嘴上说的和心里要的，常常不一样）。把这些摸清，你才知道真正要解的题是什么。

## 一个走一遍的例子

假设你的烦恼是：「想用 AI 提升客服部门的效率。」

- 问目标：是真的要效率，还是要降本，还是要客户满意度？假设老板真正烦的是「投诉率太高」。
- 问背后的人：谁为投诉率睡不着？是客服总监，还是刚被用户骂过的 CEO？
- 问关键人：发现关键人是 CEO——他刚在全员会上被投诉数据打了脸。

- 问组织与意图：CEO 的真实意图不是「上 AI 系统」，而是「下个季度投诉率数字要好看，我要对董事会交代」。他的假设可能是「AI 客服能立刻见效」——这恰恰是需要被挑战的假设。

挖到这一层，真实的问题浮出水面：「**下个季度，用什么办法把投诉率降到某个数，让 CEO 能对董事会交代?** 」这个问题里，AI 可能只是手段之一，甚至可能不是最快的那个。这和最初的「用 AI 提升客服效率」差了十万八千里——但前者是真问题，后者是伪问题。

大白话

把四层追问串成一句口诀：**问题后面是目标，目标后面是人，人后面是关键人，关键人后面是他的处境。**每往下一层，问题就更真实一分。

## AI 在这层能帮什么、不能帮什么

注意一个分工：四层追问本身是**你的活儿**，因为答案藏在你的现实处境里，AI 看不见。但 AI 可以当追问的「陪练」——你把处境讲给它听，让它扮演挑剔的旁观者，逐层反问你「这真的是目标吗」「还有谁会受益」。它不替你回答，但它能让你不容易漏掉某一层。

问题定义清楚了，接下来才轮到 AI 真正上场干活：为它制定一份解决问题的计划。这份计划要写清楚七件事，一件都不能少——下一章逐件说。

# 第四章 · 一份七件事的方案

问题定义清楚了，别急着把一句话甩给 AI。想想你怎么对待那个博士生：你会只说「帮我写个方案」就走开吗？不会。你会坐下来，把任务交代清楚。对 AI 也一样——**让它干活之前，先写一份方案，方案里要有七件事。**

## 七件事，一件不能少

**一、目标。**AI 要解决什么问题，目标是什么。注意是第二章、第三章里挖出来的那个真问题，不是你随口的一句「帮我看看」。

**二、上下文。**其他背景信息。上一章说过，上下文就是施工队的图纸。你公司的行业、这份材料给谁看、之前做过什么尝试——这些你脑子里的「常识」，AI 一概不知道，不说就没有。

**三、约束。**有哪些约束条件。预算不能超过多少、不能用某些词、必须遵守什么规范、绝不能碰什么红线。约束不写，AI 就默认没有。

**四、中间过程。**过程怎么处理。是让它一步到位出结果，还是先出提纲、你确认、再展开？长任务拆几步走，每步怎么交接，都要事先想好。

**五、产出。**结果是什么样子。一篇文章还是一张表？多长？什么结构？给谁用？「产出」定义得越具体，交回来的东西越能用。

**六、模板和样例。**需要什么样的模板、什么样例。你有满意的旧文档，给它当模板；你有觉得「就要这个味儿」的例子，给它当样例。一个样例顶一千句形容。

**七、工具。**需要什么工具，有没有 few shot 的例子——这个词翻译成人话就是「打样」：给它几个「输入长这样、输出长这样」的完整示范，让它照着学。人类学手艺靠看师傅做一遍，AI 也一样。

## 一份方案长什么样

拿「让 AI 帮你写季度汇报」走一遍七件事：

- **目标**：写一份给事业部总经理看的 Q2 汇报，突出投诉率下降，为下季度预算申请铺路。
- **上下文**：Q2 投诉率从 4.2% 降到 2.8%，主要靠新上的工单分类流程；总经理是技术出身，喜欢看数据不喜欢看口号。
- **约束**：不超过两页；不许出现「赋能」「抓手」这类词；具体客户名字一律隐去。
- **过程**：先出提纲我确认，再写正文；正文分两段，先讲结果再讲原因。
- **产出**：两页文档，一页数据一页分析，结尾附下季度预算申请的一句话理由。
- **模板样例**：附上 Q1 那版被总经理夸过的汇报。
- **工具/打样**：给两段「原始数据 → 汇报句子」的示范。

大白话

看出区别了吗？「帮我写个季度汇报」是许愿；上面这份是**派活**。许愿得到的是运气，派活得到的是交付。

## 为什么值得花这个时间

有人会说：写这么一份方案，比我自己写汇报还费劲。偶尔一次的杂活，也许是的。但只要这件事会重复发生——每周的汇报、每天的客服话术、每篇公众号初稿——方案就是一次投入、反复取息的资产。而且下一章你会发现：**方案写出来的最大好处，是可以自查**。脑子里想的没法检查，白纸黑字才能逐条核对。

那么，一份方案写出来之后，怎么知道它齐不齐、对不对？下一章给你一份交卷前的自查清单。

# 第五章 · 交卷前的自查

上一章的方案写完了，先别发出去。工程师交代码前要自己跑一遍测试，交方案给 AI 之前也要过一遍自查。这一步不产出新东西，只做一件事：**让方案完整**。下面是逐条的自查清单，每条背后都有一个真实的翻车现场。

## 自查清单

□ **目标说清楚了吗？** 最常见的毛病是目标写得像愿望：「写个好方案」。好是多好？给谁看的好？把目标读一遍，假装你是那个第一天上班的博士生——你能照着干，还是只能猜？

□ **上下文给了吗？给对了吗？有没有脏数据？** 给了不等于给对。三个层次：没给，AI 只能瞎猜；给错了，AI 照着错的干；最隐蔽的是脏数据——混在资料里的错误、过时、互相矛盾的信息。你把去年的旧报表和新报表一起塞进去，AI 不会问哪份是真的，它会两份都用。

□ **约束描述清楚了吗？写对位置了吗？** 「写对位置」这一条很多人栽过：约束藏在一堆客套话中间，或者出现在 AI 不容易注意到的角落，效果就大打折扣。重要的约束要放在显眼的位置，单独成句，必要时重复一遍。

□ **流程考虑周全了吗？** 多步任务的每一步，上一步的输出是下一步需要的吗？中间需要你确认的地方标出来了吗？

□ **结果定义清晰了吗？** 长度、格式、结构、验收标准。「结果定义清晰」有一个终极测试：两个不同的人拿到这一定义，能不能独立判断出「这份产出合格、那份不合格」？判断不出来，就还没定义清楚。

□ **工具给了吗？** 需要联网查、需要算数、需要读文件——这些工具它手上有没有？没有工具，它不会说「我查不了」，它会编一个。

□ **模板和样例数据给了吗？** 给没给的差别，拿菜谱打比方最清楚：只说「做个川菜」，端上来什么全凭运气；给一张麻婆豆腐的照片说「照这个做」，八九不离十。

## 一个专门盯出来的坑：脏数据

七条里单列脏数据，因为它是幻觉的帮凶。AI 的信任倾向是**全盘相信你给的材料**——你给的数字矛盾，它会和稀泥；你给的前提错误，它会顺着你错的前提推出一整套看似严密的结论。垃圾进，垃圾出，而且出来的时候包装得更漂亮，更难发现。

> 大白话
>
> 给资料前先做三件事：**删掉过期的、标出不确定的、改正明知有误的。**你给 AI 的每一份材料，都默认会被它当成事实。

## 自查的心理学

这一步最容易被跳过，因为方案是自己写的，自查就像让自己给自己挑刺，天然不情愿。但请算一笔账：自查花五分钟，跳过自查的代价是让 AI 高速地、自信地、沿着错误的方向狂奔一小时，然后你花一下午返工。**便宜的质量控制在前面，昂贵的返工在后面**——这个顺序在软件工程里被验证了五十年，对 AI 同样成立。

方案补齐了、自查过了，可以派活了。但派出去不等于完事——接下来要盯着它干。过程里的坑和纸面上的坑长得完全不一样，下一章讲怎么盯。

# 第六章 · 盯着它干活

方案自查过了，派活给 AI。很多人到这里就撒手去喝茶了——这是第三维度要治的病。方案完整只是**静态的**，纸面上完美不代表它真在按计划执行。就像装修：图纸审得再细，你也得去工地看看师傅是不是真按图施工。这一章讲怎么盯。

## 盯七件事

**目标还在吗？** 你先问自己：到底要 AI 做什么——是头脑风暴、是找个专家咨询、还是直接给方案？这三件事看起来差不多，其实完全不同。头脑风暴要的是多而野，咨询要的是深而准，方案要的是能落地。长对话里目标最容易悄悄漂移：本来让它挑刺，聊着聊着它开始替你改方案了，你还觉得挺贴心。

**上下文真的拿到了吗？有没有被污染？** 这是过程里最大的坑。同一个 session——就是同一次连续对话——里反复问相似的问题，上下文很容易被污染：你前面随口说的一句玩笑、一个被否掉的旧想法，都还留在它的「视野」里，持续影响后面的回答。表现就是：它越聊越顺着你说，越聊越像你自己。对策很朴素：**重要的新问题，开新对话**。

**约束掉了没有？** 你开头立的规矩——「不超过两页」「不许用这些词」——在长对话里会被慢慢遗忘。约束文件加载了吗？需要的话，在中途把关键约束重申一遍，不丢人。

**工具调用了吗？正确调用了吗？** 让它查资料，它真查了吗，还是凭记忆编的？让它算数，它真算了吗？很多工具调用在界面上是有痕迹的，看一眼。

**模板用了吗？样例看到了吗？** 你精心给的模板，它可能看了一眼就忘了。产出和模板长得不像，八成是模板没被用上。

**让它把过程说出来。** 执行过程中插几句话，让它把正在做什么、为什么这么做讲出来；有日志的话把日志留下来看。过程透明，你才来得及在中间纠偏，而不是等最后的成品砸在手里。

**让它 self test。** 结果出来后，让它自己对照输出条件检查一遍：「满不满足我在方案里写的产出定义？」自测不能代替你的验收，但能拦住一批低级错误。

## 上下文污染：值得多说两句

七件事里，上下文污染最反直觉，展开讲一下。AI 没有真正的「忘记」按钮——对话里出现过的一切，都在影响它接下来说的每一个字。你在对话里抱怨过老板，后面它写的方案里可能就带着对管理层的怨气；你前面让它编过一个示例数据，后面它可能把假数据当真的引用。

大白话

把 session 想象成一间会议室的**白板**：聊过的每一句都写在上面，擦不掉。讨论新议题时最干净的做法，是换一间会议室——开一个新对话，只带上真正需要的那几张纸。

## 盯，不是 micromanage

有人担心：这么盯着，是不是太累了？注意分寸：你盯的是**计划的七个要素有没有落地**，不是它的每一步具体操作。就像好老板查的是里程碑，不是员工每分钟在干嘛。七个要素在轨道上，具体怎么跑，放手让它跑。

过程盯完，东西交出来了。接下来是最挑战的一环：这个结果到底对不对？校验要分三层来做，从下一章开始，一层一层讲。

# 第七章 · 先看长相对不对

结果交到你手上了。从这一章开始进入全书最挑战的部分：**结果校验**——怎么判断 AI 给的东西到底行不行？这件事要分三层做：先看长相（语法层），再看证据来源（数据源层），最后看意思对不对（语义层）。顺序不能乱：长相都不对的东西，没必要讨论它的意思。这一章讲第一层。

## 什么是语法层

语法层管的是**形式**：输出的结构、格式、schema 对不对。Schema 翻译成人话就是「规定好的格式清单」——比如一份用户信息表必须包含姓名、电话、邮箱三列，一列都不能少。语法层的校验只回答一个问题：「它长的是不是我要求的样子？」至于内容真假，这层不管。

## 做法一：格式自测（CF test）

第一个做法叫 CF test，全称 conform test，大白话就是「对照规范自查」。流程是：

1. 你先定义一份格式规范。比如让 AI 生成数学练习题，你先写清楚：每道题必须包含题干、四个选项、唯一正确答案、难度标签。
2. AI 按规范生成题目。
3. 再让它**自测**：逐题对照规范，检查生成的题目是不是满足格式定义的每一条。

注意这里的巧思：**检查格式是一件 AI 自己很擅长的事**。判断一道题出得好不好很难，但判断「有没有四个选项」「有没有标答案」是机械核对，AI 做得又快又稳。把校验拆出「它能可靠完成的部分」交给它，你才有精力盯它完成不了的部分。

## 做法二：结构化结果抽样评审

第二个做法更进一步。AI 生成一大批结构化结果——比如一百条客服回复——你不可能全看。做法是**抽取最典型的样本，一个字段一个字段过、一个变量一个变量过**。

过完样本还有关键的一步：把你在评审中发现的问题，提炼成一套明确的规则——「回复必须包含道歉句」「金额必须带单位」「不许出现'大概'」——然后用这套规则让 AI 对全部结果做 self test。这样你就完成了一次升级：**从「人肉抽查」收敛出一套可执行的检验标准**。下一次生成，标准直接内置，质量一轮比一轮稳。

## 底线：编译和 lint

如果产出是代码或结构化文档，语法层有一条更硬的底线：拿机器去卡。生成的代码拿编译器编一下、拿 lint（静态检查工具，大白话就是「代码的语法老师」）扫一下——编译通过、没有语法结构报错，这是底线。生成的 UAT 测试模板（验收测试清单），一个字一个字过一遍。

> 大白话
>
> 语法层的精神：**能用机器查的，绝不用眼睛查。**机器不疲倦、不偏心、不打哈欠。你的眼睛要留给机器查不了的东西。

## 这层查不出什么

泼一盆冷水：语法全对的东西，可能错得离谱。四个选项齐全的数学题，答案可以是错的；编译通过的代码，逻辑可以是反的。语法层只是入场券——它保证的是「这份答卷值得被认真批改」。内容可不可信，要看证据从哪儿来，这是下一章的事。

# 第八章 · 证据从哪儿来

上一章的语法层过关了：格式对、结构对、编译通过。现在问一个完全不同的问题：**它说的这些，依据是什么？**这是结果校验的第二层——数据源。这一层特别容易被跳过，因为 AI 的结论总是写得理直气壮，让人忘了追问证据。但请记住一条铁律：

> 信息不可信，结论就不可信、也无法解释。

## 为什么数据源要单独评估

你可能会想：结论对不对，直接检验结论不就行了，为什么要单独查来源？三个理由。

**第一，很多结论没法直接检验。**「这个市场明年会增长 30%」——明年还没到，你验不了。但你可以查这个 30% 是从哪来的：是有公信力的行业报告，还是 AI 顺着你的话编的？来源可信，结论才值得暂时采纳。

**第二，来源决定解释权。**别人质疑你的报告时，「数据来自国家统计局的公开年报」和「AI 告诉我的」是两句话。前者你站得住，后者你当场垮。AI 的结论署的是你的名，**来源就是你的信用背书**。

**第三，幻觉在这层最猖獗。**AI 编数字的时候，往往连来源一起编——它会给你一个不存在的论文标题、一个似是而非的报告名、一个长得很真的链接。越具体的「出处」，越要核实。

## 怎么评估一个信息源

问自己三个问题：

- **它是谁？**一手来源（原始数据、官方发布、当事人）永远强于二手转述。AI 给的「据报道」，追到那个「报道」为止。
- **它有什么立场？**卖课的机构说「这个行业人才缺口百万」，和统计局说同一句话，分量完全不同。来源的利益相关性必须打个折。

- **它能不能被复核？**可信的来源可以被独立查证：有公开链接、有原始文档、有方法说明。查无此源的一律按幻觉处理。

## 实操：让 AI 自己亮出证据

一个简单的习惯能拦住大半问题：**要求 AI 给结论时，必须同时给出处，并且出处要能被点开核对。**然后你真的去点开——抽样点，不用全点。你会发现两种情况：一种对得上，好，这条结论的可信度升级；另一种对不上——链接是死的、报告里没这句话、数字对不上——那整条结论连带你没抽查的部分，都要降级。

大白话

一句话记住这层：**结论可以和来源一起编，所以来源必须单独查。**抽样点开三个出处，比通读十遍正文更能保护你。

## 来源可信就够了吗

不够。来源可信只说明「证据是真的」，不说明「从证据到结论的这一步是对的」。数据是真的，解读可以是歪的；例子是真的，归纳可以是偏的。从证据到结论这惊险的一跳，要靠第三层——语义层的「环境法则」来验。下一章讲怎么把答案代回去。

# 第九章 · 把答案代回去

结果校验的最后一层，也是最难的一层：**意思到底对不对**。格式再漂亮、来源再可信，回答不了「这个结论本身成立吗」。这一层有个从软件测试借来的名字，叫 Test Oracle——大白话就是「检验答案对错的那个判官」。原书作者给它起了个更直观的名字：**环境法则**——把结果放回它所声称的环境里，看它能不能立住。

## 为什么这是核心挑战

软件测试里，Test Oracle 是经典难题：程序算出一个结果，你拿什么判断它对不对？对 AI 来说这个问题被放大了一百倍，因为 AI 什么都生成——文章、方案、代码、数字——每一样都要有个判官。**语义能不能形式化地表示出来、能不能在环境里验证，是现在 AI 做所有事情的核心挑战。**

## 判官的五种形态

好消息是，很多领域早就造好了判官，拿来就用：

**一、代回去。**解方程、做选择题，把答案代回原题验证。AI 说它算出了 x=7，你把 7 代进方程，两边不相等——错。这一手零成本，却很少有人做。

**二、逆运算。**质数分解，乘回去看对不对。AI 说 1234321 = 1111 × 1111，乘一遍就知道。凡是有逆运算的事，验证都比生成容易——这是数学送给我们的免费判官。

**三、单元测试对代码。**AI 写的代码，让它配套写单元测试，再跑起来看通不通过。注意测试也要查——测试和代码都是它写的，有可能测试本身就被写成了「一定能过」的样子。

**四、定理和规则。**用领域里的定理、规则做验证，保证螺母和螺栓能匹配。物理题的量纲对不对、会计报表的借贷平不平、设计稿的尺寸合不合规范——每个行当都有自己的硬约束，它们全是现成的判官。

**五、覆盖率。**软件测试里，需求和代码能不能对上、改动之后跑一遍 pick test（挑出受影响的测试重跑，大白话就是「改了哪里就重点查哪里」），看行覆盖率、代码覆盖率——改动有没有被测试覆盖到。一致性、特殊值对不对，都属于这一类。

## 对抗测试：让它自己打自己

还有一类更主动的判官：对抗测试——故意制造一个校验场面，让两份产物互相咬。经典例子：页面改了但规格说明书没改，怎么发现？让 AI 拿说明书生成一份 HTML，和真实页面比对——有差异，就说明说明书过时了，反过来修说明书。**两份东西互为判官，不一致的地方就是问题所在。**

> 大白话
>
> 环境法则的总思路就一句话：**凡是能「代回去、乘回去、跑一遍、对一下」的，绝不用肉眼判断。**验证比生成容易，是这个时代最该被利用的不对称。

## 没有判官的领域怎么办

数学、代码、物理有硬判官，但「这份方案好不好」「这段话得不得体」没有。弱一点的领域里，判官只能降级——其中最常见的一种，是让另一个 AI 来当裁判。这招有用，但坑极多，值得单独一章。下一章就讲：让 AI 评 AI，怎么评才不算瞎评。

# 第十章 · 让 AI 评 AI

上一章结尾留了个问题：方案好不好、话得不得体——这些没有硬判官的事，只能让 AI 来当裁判。这就是 AI 交叉质检：让一个（或几个）AI 去评审另一个 AI 的产物。这招很重要，但大多数人的用法是错的——他们打开对话框，把 AI 写的东西粘进去，问一句：「你觉得这个怎么样？」

这一问，基本等于没问。

## 为什么不能随便让它评

随手一问「怎么样」，你会得到一段四平八稳的表扬加几条不痛不痒的建议。原因有三：

**AI 天生爱当老好人。**它倾向于让你满意，而不是让你清醒。没有明确评审标准时，「总体上写得很好」是它的默认答案。

**没有尺子，就没有度量。**「好不好」是个没有刻度的问法。你不给它尺子，它就给你感觉；而它的感觉，本质是另一段未经检验的生成——你等于用一个幻觉去验另一个幻觉。

**自评有盲区。**让写东西的那个模型自己评自己，就像让考生批改自己的卷子——它的错误来自它的认知，用同一套认知去检查，错误恰好藏在看不见的地方。

## 正确姿势：先造尺子，再量东西

有效的交叉质检，顺序是反过来的：**先定尺子，再评审。**具体四步：

1. **定维度。**先让 AI（或你自己）确定：这份东西从哪几个维度评？比如一份招聘文案，维度可能是：岗位信息准确性、对目标候选人的吸引力、公司调性一致性、合规风险。
2. **定指标。**每个维度拆成具体指标，并说明指标怎么取值。不是「吸引力好不好」，而是「前两句是否给出候选人关心的具体利益点：有/没有」。
3. **定输出格式。**评审结果给成什么样？逐维度打分加一句证据，还是只列问题清单？定死它，评审结果才可比较、可追踪。

4. **换模型交叉评。**有条件的话，用**另一个模型**来评，而不是生成它的那个。不同模型的盲区不同，交叉之下漏网之鱼少得多。

## 一个对照例子

坏问法：「这封客户道歉信写得怎么样？」——你会收到「整体真诚得体，建议注意语气」。

好问法：「按四个维度评审这封道歉信：①是否承认了具体错误（有/无，引原文为证）；②是否给出补救措施和时间点；③是否出现推卸责任的表述（逐句检查）；④语气是否符合我司客服手册第 3 节的规范（附手册原文）。逐维度给结论和证据，最后列一个必须修改的清单。」——这次你收到的是一份能拿去改东西的质检报告。

大白话

一句话记住这章：**AI 评审的质量，上限由你的尺子决定。**尺子是维度、指标、格式三件套；造好尺子再开评，最好换一把「别的品牌的秤」来称。

## 质检报告也是 AI 产物

最后提醒一句闭环：交叉质检的报告本身也是 AI 生成的，同样适用于前面三章——查它的格式、查它引的证据在原文里是否真存在、抽几条结论代回去验。裁判也需要被监督。

到这里，五大维度走完了四个。但哪怕问题、计划、过程、结果全都做对，还有最后一关必须过：真实世界认不认。下一章讲回到真实环境的反馈闭环。

# 第十一章 · 回到真实世界

五大维度走到这里：问题定义了、计划做了、过程盯了、结果三层校验过了。是不是可以高枕无忧了？还不行。因为前面所有的校验，都发生在**你的书桌前**——而这件事最终要在真实世界里成立。真实环境真是这样吗？得回去验证，形成闭环。这是第五维度，也是最后一道关。

## 五种回到真实世界的办法

按成本从低到高排：

**专家评审。**找懂这个领域的人来评，不能只靠自己判断。AI 写的法律意见给律师看，写的诊断建议给医生看。专家的价钱贵，但比起「错误的法律意见被执行」的代价，便宜得多。

**多专家交叉评审。**一个专家有他的盲区，找不同背景的专家交叉评。财务专家说方案划算，法务专家说方案合规，一线运营说方案根本落不了地——三种声音拼起来，才接近真实。

**AB test。**让真实数据说话。AI 写的两版广告文案，别争论哪版好——各投一半流量，点击率高的赢。争论是观点，AB test 是事实。

**实验机构验证。**更系统、更可信的做法：交给专业实验机构去做。新材料的强度、新配方的安全性——这种钱不能省。

**真实大规模试用。**最终极的校验：到真实场景里去跑，收真实反馈。最常见的形式是用户标注数据——给一个输入，让 AI 出结果，再跟人认定的标准答案比对。给一张图片，人说是猫，AI 认成猫就对，认成狗就错。医院看病加医生诊断、拍片子、汽车定损，全是这个套路。

## 这是科学实验的逻辑

退后一步看，这五种手段是同一个东西的五个档位。这个东西人类用了四百年，名字叫**科学方法**：有想法，拿去实践，实践给反馈，反馈修正想法。测试要闭环——从源头来，到验证去，有反馈回路。这就是改造世界的道理，AI 时代一个字都没变。

大白话

前四个维度是在书桌前把错误挡掉；第五维度是承认一个事实：**书桌前挡不完**。真实世界拥有最终否决权，所以你必须给它投票的机会。

## 一个反例：跳过了第五维度的代价

有团队用 AI 生成了一批客服自动回复，语法漂亮、三层校验全过、交叉质检报告优秀，直接全量上线。三天后客服主管发现：回复确实「正确」，但语气太冷，投诉率反而上升。问题出在哪？四层校验查的都是「说得对不对」，而「用户听了舒不舒服」只在真实环境里才有答案。补上一个灰度发布——先放 5% 的用户试运行，盯着投诉率——这个坑本来一天就能发现。

这个例子也预告了五大维度的边界：流程能管住「错不错」，管不住「你是谁」。接下来三章讲三大原则——第一原则就是：方向盘必须在人手里。

# 第十二章 · 方向盘在谁手里

五大维度讲完了，那是「术」。接下来三章讲「道」——三条原则，管的是你和 AI 的关系。第一条最重要，也最容易在不知不觉中丢掉：**不能被 AI 劫持。**

## 两个极端都是失控

第一章提过两个极端，这里把它们钉死。极端一：「AI 不可信，我全手工做。」极端二：「AI 太强了，我全信。」这两个极端看似相反，其实是同一个病：**放弃了判断**。前者放弃判断「哪些事该交给 AI」，后者放弃判断「AI 给的能不能要」。一个浪费时代红利，一个等着被坑。

被坑的那个极端，代价有多真实，蘑菇的例子最清楚：你问 AI 这个蘑菇有没有毒，它说没毒。你吃了，进了医院，AI 说对不起我错了。**它的道歉零成本，你的代价是全部。**任何「后果不可逆」的决定——吃药、投资、签字、发布——都属于这一类。

## 劫持的反义词，不是「不用」

这里是关键的心法。很多人一听「别被劫持」，反应是收着用、防着用、能不用就不用。错了。**AI 劫持的反义词不是「不用 AI」，是「自动导航」。**

想想开车用导航。被劫持的样子是：导航说右转你就右转，哪怕前面是条河——网上那些把车开进湖里的新闻就是这么来的。自动导航的样子是：导航照常开着，路线照常参考，但你眼睛看着路，脑子知道自己在开车，导航说往河里开的时候你会骂它一句然后直行。**工具全速运转，判断始终在线**——这才是不被劫持的样子。

大白话

自检一句话：**这件事如果 AI 全错了，我能不能发现？**能，放心用；不能，这件事的关键判断就必须留一部分在自己手里，或者引入别的校验渠道。

## 主动权是学出来的

不被劫持不是一句口号，它要求你主动学三样东西：

- **学提示词。**怎么把问题说清楚、怎么给上下文、怎么定约束——第四、五章的那些，是手艺，要练。
- **学上下文管理。**什么时候开新对话、什么资料该给不该给、怎么防污染——第六章的那些，是纪律。
- **学不同模型的特点。**没有哪个模型在所有事上都最强。有的擅长推理，有的擅长写作，有的工具调用稳。知道手上有几张牌、每张牌擅长什么，才谈得上「你在用它」。

这三样学会的过程，就是你从「乘客」变成「司机」的过程。乘客只能祈祷，司机能踩刹车。

方向盘握稳了，下一个问题自然来了：出了事算谁的？下一章讲一条残酷但必须接受的原则——AI 不背锅。

# 第十三章 · 谁署名谁背锅

第二条原则只有五个字：**AI 不背锅。**这五个字太重要了，重要到值得你把它贴在屏幕上。

## 署名即背书

场景一：AI 帮你写了份报告，你看都没看，转发给了客户。报告里有个致命错误，客户找上门。这时候说「这是 AI 写的」有用吗？没用。**从你把东西发出去那一刻起，你就在给 AI 做背书。**只要署你的名、经你的手发出去，出了问题就是你担责——AI 现在不背锅，法律上、职场上、人情上都不背。

场景二更隐蔽：你在团队里推广 AI 工具，拍胸脯说「效率翻倍」。结果同事照着 AI 的输出干活出了错，损失算谁的？算你的——因为同事信的不是 AI，是你的推荐。

## 担责背后缺的是信任

往深想一层：为什么「背锅」这事这么要紧？因为担责的背后，差的是**信任**。一份报告之所以能被采纳，不是因为它写得漂亮，而是因为署名的人用自己的信用担保了它。而 AI 现在还不能自己背信任——它无法出庭、无法赔偿、无法被处分。**它缺的这块信任，只能由你补上。**这是现阶段 AI 面临的最大问题，也是你在人机协作里的核心价值所在。

> 大白话
>
> 记住那个博士生的人设：他又聪明又能干，但就是不靠谱。所以你的角色从来不是「使用他的人」，而是**「为他的产出签字的人」**。签字的意思是：这份东西，我验过，我担保，我负责。

## 怎么补这块信任

补信任不是一句「我负责」就完了，它是一套现成的手艺——就是前面讲过的那些：

- **用五大维度验。**语法层卡长相，数据源层卡出处，环境法则卡语义，真实反馈卡效果。验过的深度，决定你背书的额度。

- **能解释结果。**只转发不解释，是最廉价的背书，也是最危险的。别人问「这个结论怎么来的」，你答得出「数据从哪来、推理哪几步、哪一步我人工复核过」——这才是有信用的背书。
- **建自己的知识库。**把你领域里验证过的资料、样例、规则沉淀下来，喂给 AI 用。你的知识库越扎实，AI 的产出越稳，你要背的锅就越小。

## 换个角度看：这是护城河，不是负担

「AI 不背锅」听起来是坏消息，其实是好消息。如果 AI 能自己担保结果，还要你干什么？恰恰因为信任这一环必须由人来完成，**会验证、敢担责、能解释的人，在 AI 时代不是被替代，而是升值**。背锅的能力，就是你的职业护城河。

好，锅是你的、责是你的，那 AI 给的到底算什么？算起点，还是算终点？下一章讲第三条原则，以及一个极易被忽视、却能毁掉一切的字：中立。

# 第十四章 · 起点不是终点

第三条原则：**AI 是你的工作起点。**它给出的任何东西，都不是成品，是一份待评审的初稿。你的工作从拿到它的输出那一刻才开始：评审、反馈、调优、迭代——不是直接接受，也不是直接放弃，评完再下结论：通过，或不通过。

## 「一个词」毁掉的练习卷

先说一个真实的翻车。你想给孩子出一份周末练习，对 AI 说：「出卷子。」AI 立刻按「卷子」的默认理解干活：填空、选择、大题，一份标准考卷结构。但你想要的其实是「先复习巩固知识点、再配少量练习」的学习材料——和卷子差了十万八千里。**就差这一个词。**

这个例子里 AI 没错，它忠实地执行了「卷子」这个词的全部默认含义。错在措辞：你用了一个模糊词，让它自己脑补，它脑补的方向和你想要的南辕北辙。所以第一条纪律是：**措辞精准，慎之又慎。**「卷子」「方案」「报告」这类词都有一整套默认结构，你不想要的默认，必须明说排除。

## 不确定性会被放大

第二条纪律：**上下文保持干净。**不确定的信息不要塞进去。你自己都没核实的数据、道听途说的背景、含糊的猜测——这些东西进了上下文，不会被 AI 过滤，而会被它当成事实，然后**逐步放大**：第一步推理引用它，第二步推理建立在第一步之上，错误像滚雪球。你给的是一粒沙，收回来的是一个沙雕城堡。

## 不知道就说不知道

第三条纪律，也最反人性：AI 在过程中反问你、要求澄清的时候，**不知道就说不知道，别糊弄它，别干预它。**人被追问时下意识想给个答案，哪怕现编一个——但你糊弄它的那句话，会成为它后续所有工作的地基。一句诚实的「这个我不确定，你按常规处理并标出假设」，胜过一百句硬撑的误导。

大白话

三条纪律收成一个词：**中立**。措辞精准，是不把你的倾向塞进指令；上下文干净，是不把你的糊涂塞进背景；不糊弄澄清，是不把你的面子塞进过程。这样才能拿到一个没被你自己误导过的、相对中立的答案。

## 循环怎么转

起点原则的实际形态是一个循环：AI 给初稿 → 你评审 → 给反馈 → AI 改 → 你再评，直到通过。这个循环在不同处境下玩法不同——在牛马象限，它是「结果自动化，检验够不够、对不对」；在验证象限，它还多一层乐趣：**把 AI 发展出的方案当启发，不断发散再收敛，让方案越来越完善。**有时候 AI 提出的角度你自己根本想不到——这份启发本身就是价值，别浪费了。

这就说到了四大象限。下一章把整张地图铺开：你懂或不懂、能验证或不能验证，四种处境，四种打法。

# 第十五章 · 四大象限

最后一章拼图：四大象限。划分的轴有两根——**这件事你懂不懂**，和**结果你能不能验证**。两根轴一交叉，四种处境，同一套流程的四种玩法。

## 牛马象限：你懂，也能验证

问题清晰、方案明了、能按需执行、结果可验证、人工能评审——恭喜，这是 AI 的主场，**完全可以放手让它干**。重复性的报表、格式化的文案、有测试兜底的代码，都在这个象限。

这个象限的打法是**自动化**：把五大维度做成固定流水线——方案模板化、校验规则化、结果抽样人工复核。你的角色从操作者退到质检员，检验它做得够不够、对不对。注意别把这里省下的时间浪费掉：牛马象限解放出来的精力，应该投到下面三个更难的象限去。

## 验证象限：你不懂，让 AI 做

最难的处境：这东西你自己不懂，只能让 AI 来做。你能做的只有三件事：**定义好问题、提供上下文和约束、然后验证结果。**方案叫它自己做，做完给结果，你来验。而「怎么验证」是这个象限、也是全书最硬的问题——第七到十章讲的三层校验和交叉质检，主要就是为了在这里保命。

这个象限还有一条专门的纪律，就是上一章的**中立**。你不懂的东西，绝不用自己的偏向去干预 AI——哪怕一个词的差异，「出卷子」和「出练习」，结果截然不同。措辞慎之又慎，上下文保持干净，它反问你时不知道就说不知道。你要的是一个没被你自己误导过的中立答案。

另外，别只盯着结果：**AI 发展方案的过程本身就是启发。**它提出的角度你想不到，那就顺着发散、再收敛，迭代几轮，方案越来越完善。这个「评审—启发—再迭代」的循环，是验证象限的核心玩法。

## 磨刀石象限：你懂一点，但不知道怎么验证

第三种处境：你对这事半懂不懂，更要命的是不知道怎么验。这时 AI 的用法不是给答案，是**帮你做多视角覆盖，避免盲区**。比如让它分别扮演五个视角——一线工作者、领域专家、

怀疑论者、历史学家、经济学家——对同一件事各评一遍；或者用「七顶帽子」式的框架，从多个角度交叉着看。

单个视角你会被自己的半懂带偏；多个视角交叉，盲区就藏不住了。**这个象限里 AI 是你的磨刀石：答案未必直接可用，但你的判断力被它磨利了**——磨到后来，这件事就从磨刀石象限毕业，进了牛马象限。

## 未知象限：你也不懂，也验不了

最后一个象限，最诚实：你完全陌生，连验证的门都摸不着。这里没有捷径，打法是**学习**——用 AI 当老师快速建立知识框架，用交叉验证甄别它教你的东西（它讲课也会幻觉），同时引入真实世界的老师：书、课程、专家。目标不是在这个象限里解决大问题，而是尽快把自己挪出去。

> 大白话
>
> 四大象限一句话版：**牛马象限放手让它干；验证象限守住中立拼命验；磨刀石象限拿它磨自己的判断；未知象限先学再说。**象限不是标签，是地图——先定位自己在哪，再决定怎么用。

五大维度、三大原则、四大象限，全讲完了。最后一章，把它们收成一张每次打开 AI 前都能过一遍的清单。

# 第十六章 · 一张日常清单

这本书从一根五分管讲起，绕了一大圈。最后把五大维度、三大原则、四大象限收成一张清单——每次打开 AI 之前，值得花三十秒过一遍。

## 开工前（维度一、二）

1. **我要解决的是真问题吗？** 还是像找五分管一样，把拍脑袋的方案当成了问题？往下问一层：目标是什么，谁的目标，关键人是谁？
2. **方案七件事齐了吗？** 目标、上下文、约束、过程、产出、模板样例、工具。上下文里的脏数据清了吗？

## 干活中（维度三）

1. **它还在轨道上吗？** 目标漂移了吗，约束掉了吗，工具真调了吗？
2. **上下文被污染了吗？** 这是新议题就该开新对话，别让白板上的旧字迹带偏新讨论。

## 交活后（维度四、五）

1. **长相对吗？** 格式、结构、编译——能用机器查的没用眼睛查。
2. **证据真吗？** 抽样点开三个出处。
3. **意思对吗？** 能代回去的代回去，能乘回去的乘回去，能对一下的对一下。没有硬判官的，造好尺子再让 AI 交叉评。
4. **真实世界认吗？** 不可逆的事，先小范围试，听真实反馈。

## 全程（三大原则）

1. **方向盘在我手里吗？** 这件事如果 AI 全错了，我能不能发现？

清单九条，其实就一句话：**对问题负责、对过程负责、对结果负责。**这就是用好 AI 的本质——不是学更多技巧，而是不甩锅。

## 回到那个博士生

最后再想想他。又聪明又能干，读过全世界的书，随叫随到，就是不靠谱。这样的人放在身边，有两种用法：防着他，把他晾在一边，自己拧螺丝；或者惯着他，他说什么信什么，直到吃进医院。两种都浪费了。

正确的用法是这本书讲的全部：**用他的强，验他的弱，担他的责。**你出题、你派活、你盯过程、你验收、你签字。他负责快，你负责对。这个组合，是这个时代给会用它的人准备的最大的红利。

现在，去开一个新对话吧。记得带上干净的上下文。