

把问题问对

从三个被朋友否决的问题，看清自己提问时的思维陷阱

思维方式训练

费曼式写法 · 由多个 AI 代理撰写与互相审校

导读

从三个被朋友否决的问题，看清自己提问时的思维陷阱

先别急着找答案——这本书想让你做的第一件事恰恰相反：**停下来，看看你手里的问题是不是真的。**

这本书起于一顿不太愉快的晚饭。作者连着提了几个问题，被朋友当场否决：「你提出的问题本身，是有问题的。」第二天把同样的问题抛给 AI，AI 绕开答案，先指出：你的提问逻辑有点小问题。

一个人加一个机器，指着同一处。这就值得较真了。

我们这一辈子都在练习「怎么回答」：考试答题、工作交方案、被人问住时快速圆场。但几乎没人教过另一头的功夫——**一个问题在任何人开口回答之前，就可能已经错了。**它驮着一个没说的前提，前提不成立，后面的讨论全是认真生产的废料。

这本书把那顿晚饭当成一台事故来做调查：拆解「以偏概全」「夹带私货的为什么」「错位的类比」「一词两义」这四个反复出现的陷阱，然后把教训翻过来，攒成四件能带走的工具——怎么把藏起来的前提挖出来、一个好问题的四个零件长什么样、怎么把 AI 调教成专挑你毛病的陪练。

贯穿全书的问题只有一个：**一个问题凭什么算「问对了」？**

读法上没有门槛：每章能独立看，但合起来是一条线——从事故报告，到零件清单，到训练方法。如果你最近也被人说过「你这个问题问得不对」，或者隐隐觉得自己的讨论常常原地打转，这本书就是写给你的。

第一章 · 一顿被否决的晚饭

那天晚上，我和老韩、杨明一起吃饭。酒过三巡，我照例抛出几个盘旋在脑子里的问题。聊到后面，杨明忽然有点激动。他没有回答我的问题，而是说了一句让我当场愣住的话：

你提出的问题本身，就是有问题的。

我以为讨论的套路永远是：我问，你答；答得好不好、对不对，都是答的事。可杨明把矛头掉转了一百八十度，指向了提问的人——我。那一刻我没反应过来问题到底出在哪，只觉得哪里不对劲，又说不上来。回家复盘，我决定把这顿饭当成一条线索，顺着它去看清一件我从没认真想过的事：**一个问题，是可以在任何人回答之前，就已经错了的。**

三个被当场驳回的问题

先把现场还原一下。我那晚一共问了三个问题，三个都被杨明判了「不成立」。

第一个，我问：为什么那么多律师，最后都转去做刑事诉讼？杨明说，这个问题站不住。我身边确实碰到过几个人——有从媒体转行的，有从老师转行的——转去做了刑辩律师。可那只是我恰好遇见的几个人，更多转行的律师根本不是这样。我拿几个个例，当成了一个普遍现象来发问。

第二个，我顺着他给的一条新信息往下问：很多法务总监在挑选合作律所时，是不是只认那几家「红圈所」（红圈所，指中国业内公认规模和声誉领先的头部律师事务所，数量就那么几家）？讨论到这里开始打转。杨明反驳：进法律顾问库的往往不是「律所」，而是某个具体的「律师」；而且入库常常只是后面一整套程序里的第一步，入了也不等于拿到活。这两句话，一句说我把「律所」和「律师」混为一谈，一句说我把「入库」和「中选」当成了一回事。

第三个，是关于离婚的一个问题。具体问的什么我没来得及展开，但杨明同样摇头：你又问错了。

三连否决。更让我在意的是第二天的事——我把这几个问题原封不动抛给了 DeepSeek，想看看机器怎么说。它绕开了我期待的答案，先提醒了一句：你这几个问题的提问逻辑，本身

有点小问题。

两方指同一处，就不是偶然

一个老朋友当面说我问错了，一个 AI 隔着屏幕也说我问错了，而且指的几乎是同一处毛病。连着两方命中同一个靶心，我没法再把它当成「他们没听懂我」。这更像是我自己身上一个稳定的、反复发作的思维模式。

大白话

我提问的毛病，大概可以写成一句话：**拿身边三五个个例，直接推成一个「普遍都这样」的结论，再把这个没验证过的结论当成不言自明的前提，在上面接着往下追问。**于是问题问得越深、越细，离事实反而越远——因为地基本身是斜的，楼盖得越高，歪得越离谱。

「错的问题」到底错在哪

我们从小被训练的是「把答案做对」。考卷上，题目是给定的、被默认成立的，你的任务只是在它底下填对的字。久而久之，我们形成一种错觉：问题永远是中性的、无辜的，对错只发生在答案那一端。

但真实世界里，没有谁替你出题。你得自己把问题从一团模糊里拎出来——而就在「拎」这个动作里，你已经悄悄塞进了一堆没说出口的假设。前提，就是一个问题里那些你没明说、却默认为真、并以它为地基继续往下问的东西。「为什么那么多律师转做刑诉」这个问题，默认了「确实有很多律师这么做」。如果这个默认是假的，那么无论你给出多精妙的回答，都是在为一件根本没发生的事找原因。

错的问题，不是答不出的问题，而是**答对了也没用、因为它问的那件事压根不成立**的问题。

这本书想顺着这条线做什么

我不想把这顿饭的尴尬翻篇就算了。我想把「我是不是总在提出错误的问题」当成一把手术刀，解剖开我自己提问时的思维，看看里面到底藏着哪几种反复出现的陷阱，以及——更要紧的——怎么才能提出一个前提站得住、值得被认真讨论的问题。

所以这本书贯穿始终的那个问题是：**一个问题凭什么算「问对了」**？接下来的每一章，都在拆解我那晚踩中的某一个坑——以偏概全、夹带前提的「为什么」、偷换结构的类比、一词两义的含混——然后把它翻过来，变成一条下次能用上的检查动作。读完你不一定能回答世上所有难题，但你大概能在开口之前，先掂一掂：我这个问题，地基正不正。

先从最要命、也最隐蔽的那条说起——下一章，我们把一个问题拆开，看看它背后到底驮着多少你没留意的前提。

第二章 · 每个问题都驮着一个前提

第一章里杨明说「你的问题本身有问题」。这句话听起来像绕口令，其实指向一个很具体的结构。这一章我们不动我那些毛病，先把「问题」这个东西放到解剖台上，看看它由什么构成。

问题的水面之下

随便拿一个问题：「为什么律师都转去做刑事诉讼？」

这个句子说出口的只有一层：我在求一个解释。但它能问出口，靠的是水面之下一个没说出口的句子：**「很多律师都转去做了刑事诉讼」这件事是真的。**

这个没说出口、但必须为真、整个问题才站得住的东西，叫前提——大白话讲，就是「你问这个问题时，默认已经成立的事实」。

几乎每个问题都驮着一个或多个前提：

- 「为什么现在的年轻人不愿意结婚？」驮着「年轻人结婚意愿在下降」。
- 「怎么才能让孩子不输在起跑线上？」驮着「人生存在一条起跑线，而且输了有后果」。
- 「这家公司为什么留不住人？」驮着「这家公司的人员流失确实高于正常水平」。

注意这些前提的共同点：它们听起来都像真的。这正是麻烦所在——前提越顺耳，越没人去检查它。

前提错了，会发生什么

这是本章最核心的一条因果链：

前提不成立 → 问题没有真实对象 → 任何回答都是在解释一个不存在的现象 → 讨论越深，离事实越远。

用我那顿晚饭来走一遍这条链。我的前提是「律师转型普遍选刑事诉讼」。假如这个事实不成立（第三章会认真查它），那么「为什么」就没有对象——任何一个热心回答我的人，都

只能编一个理由去解释一个没发生过的趋势。理由可以编得很精彩：刑事诉讼有对抗性、有戏剧性、能快速建立个人品牌……每一条都头头是道，每一条都在加固一个假地基。

大白话

大白话：一个错前提的问题就像问「为什么这根冰棍是热的」。你可以请来物理学家讲半天分子运动，讲得都对，但冰棍根本不是热的——你们讨论得越深入，只是在错误上盖越高的楼。

「问题的问题」比「答案的问题」贵得多

工程上有个常识：bug 发现得越晚，修起来越贵。需求错了，代码写得再漂亮也是白写。

提问是一样的流水线：**前提 → 问题 → 讨论 → 结论 → 行动**。前提是这条流水线的第一道工序。答案错了，返工一次讨论就行；前提错了，后面所有工序全是废料——而且因为讨论本身看起来很认真，你根本察觉不到废料在生产。

这就是为什么杨明会激动。争论答案，双方在同一个世界里切磋；而前提错了的问题，会把一桌子人拖进一个虚构的世界里认真开会。

一个立刻能用的自检动作

从今天起，任何一个问题出口之前，做一个动作：把问题里的「为什么」「怎么」「是不是」暂时遮住，问自己——

这句话要想问得出口，有哪些事必须先是真的？

把它们一条条写下来，那就是这个问题驮着的全部前提。第二章之后的整本书，其实就是围绕这个动作展开的：哪些前提最容易造假（第三到第六章），怎么把藏得深的前提挖出来（第七章），以及合格的前提长什么样（第八章）。

下一章先看造假率最高的那一种：拿几个身边的个例，直接充当成全世界的普遍现象。

第三章 · 三个朋友不等于「律师都」

上一章结尾留了个作业：造假率最高的前提，是拿几个身边个例充当普遍现象。这一章就拆它。它在我那晚的第一个问题里登场——「为什么那么多律师都转去做刑事诉讼」——但它绝不只出现在饭桌上，它是人类出厂自带的思维捷径。

大脑是一台「三例成团」的机器

先还原我脑子里的加工过程。我见过：一个从媒体转行的朋友，做了刑辩律师；一个从老师转行的朋友，也做了刑辩律师；大概还有一两个类似的。然后我的大脑自动输出：**「律师转型，普遍选刑事诉讼」**。

注意中间没有任何一步是我决定要这么推的。它是自动发生的。心理学里有个说法叫可得性偏差——大白话讲：大脑判断一件事「多不多」，不去数真实数量，而是看「我能多快想起几个例子」。想得越顺，就觉得越多。三个活灵活现的熟人案例，在记忆里的「可提取速度」极快，于是它们被大脑当成了「很多」的证据。

这台机器在原始环境里是好用的：在草原上，见过三次「那片草丛里窜出过狮子」，押注「那草丛危险」能救命。样本小，但代价不对称，宁可错杀。问题在于，现代饭桌上讨论「律师行业趋势」时，这台老机器还在用同样的算法——**用记忆的流畅度，冒充统计的真实性**。

大白话

大白话：你的大脑不数数，它只凭感觉。「我能一下子想起仨」和「这事很普遍」，在它那儿是同一个信号。而这两件事之间，隔着十万八千里。

三个熟人为什么几乎必然骗你

就算不挑大脑的毛病，光从算术上看，「我身边几个案例」也是极差的证据，原因有两层。

第一层：样本太小，噪声淹没信号。假设真相是「转型的律师里有 10% 选刑事诉讼」。你随机认识 5 个转型律师，撞上 2 个刑辩的概率并不低——但你也可能撞上 0 个或 4 个。小样

本的结果在真实比例附近剧烈晃动，你撞见的那次只是其中一种晃动。拿它下结论，等于摇了一次骰子就宣布骰子有鬼。

第二层，更要命：你的「身边」不是随机抽出来的。这叫样本代表性问题——大白话讲：你认识人的渠道，本身带着偏好。我为什么认识转行的律师？因为我们有共同的朋友圈、共同的饭局。而「会出现在我饭局上的转行律师」，和「全体转行律师」，是两个不同的群体。爱出来吃饭的、愿意讲自己故事的、恰好进了同城的——每一道过滤都在悄悄筛选。刑辩律师故事性强、饭桌上爱聊，被我记住的概率本身就高。

这两层叠起来，结论很扎心：**「我身边几个」不仅是小样本，还是有偏向的小样本。它在算术上几乎注定要骗你。**

代表性的两个土办法检验

不需要学统计学，下次想用「我见过」做前提，先过两关：

- **数量关：**我见过的，是几个？如果一只手数得过来，它撑不起任何带「都」「普遍」「越来越多」的问题。把句子里的「都」改成「我见过的这几个里」，问题往往自己就散了。
- **来源关：**这些例子是怎么来到我眼前的？是我自己找来的，还是它们自己撞进来的？凡是「撞进来」的（饭局、朋友圈、新闻推送），都带着渠道偏好——越是戏剧性、越是反常的事越容易撞进来。还记得吗，你会先注意到「转行的做了刑辩」，恰恰因为这事有点反常、值得聊。

第二关有个反直觉的推论：**越是让你印象深刻、脱口而出的例子，作为证据的质量往往越差。**因为「印象深刻」本身就是一种筛选——筛选的偏好是「反常」，而你要论证的却是「普遍」。

把第一个问题救回来

按这两关改造我那句「为什么律师都转去做刑事诉讼」：「都」字先砍掉；「为什么」先挂起。剩下一个可以查证的小问题：**「转型做律师的人里，选刑事诉讼的比例大概是多少？和其他方向比呢？」**——这个问题的前提（有人转型、方向可统计）站得住，答案也有地方找（律协数据、行业报告、招聘市场）。它不一定能聊出戏剧性，但它配得上一个认真的回答。

「为什么」这个词，是下一个陷阱的重灾区。它骗你的方式和个例完全不同——下一章拆它。

第四章·先问「是不是」，再问「为什么」

上一章的结尾说，「为什么」是另一个陷阱的重灾区。这一章专门解剖这个词。它看起来是求知欲的象征，实际上是最擅长夹带私货的提问句式。

「为什么」是一个特洛伊木马

对比两个句子：

- 「海豚为什么会救人？」
- 「海豚会救人吗？」

第一句听起来更有深度，第二句听起来像小学生。但第一句在语法里藏了一个动作：它跳过了「这件事是否成立」，直接开始要解释。听者一旦开始回答「因为海豚智商高、有亲社会行为……」，就等于签字承认了「海豚确实救人」——而这个前提从未被检查过。（顺便说，「海豚救人」的报道大量是幸存者讲的故事，被海豚推向深水区的人没有机会讲故事。）

这种把未验证断言偷运进问题的句式，逻辑学里叫复合问题——大白话讲：一句话里塞了两个问题，但只留了一个回答的位置，于是前一个问题的答案被默认为「是」。法庭上的经典款是「你打老婆是什么时候停的？」——无论回答哪年哪月，都等于承认打过。

大白话

大白话：「为什么 X」这句话的真实结构是「X 成立，请解释」。你答应解释的那一刻，X 就免费过关了。「为什么」不是疑问词，是前提的走私通道。

我们不是故意走私，是句式替我们偷的

需要替我自己说句公道话：饭桌上问「为什么律师都转去做刑诉」时，我没有一点「设套」的意思。复合问题的可怕之处恰恰是无意的——「为什么」句式在汉语里太顺了，顺到它自动把「确认事实」这一步折叠掉。

背后还有一个心理推手：问「为什么」显得深刻，问「是不是」显得迟钝。饭桌上，「为什么年轻人不结婚了」能开启一场高谈阔论；「年轻人结婚率到底降了多少」听起来像查户口。我们天然偏好前者——它让谈话立刻进入「输出观点」的舒适区，跳过了「核对事实」的体力活。我那晚的三个问题，全都是「为什么/怎么样」开头，没有一个是「是不是/有多少」开头。这不是巧合，是句式偏好。

把 why 拆成 whether 和 how

改造方法很机械，机械到可以做成口诀。任何一个「为什么 X」，拆成两问：

1. **whether: X 是真的吗？** 程度如何？证据在哪？（「转型律师选刑诉的比例是多少？」）
2. **how: 如果 X 是真的，它是怎么发生的？** 机制是什么？（这一步才是原来那个「为什么」真正想问的东西）

顺序不能反。**whether 不过，how 不开工**。这像软件里的断言（assert）：前置条件不满足，后面的函数直接抛异常，不许执行。

用另一个常见问题走一遍：「为什么程序员都爱熬夜？」

- **whether**：程序员平均入睡时间真的比其他职业晚吗？——查一下会发现，「爱熬夜的程序员」和「转行做刑辩的律师」是同一种东西：故事性强的那部分人被看见了，安静早睡的大多数不在样本里。
- **how**：只有上面过了，才轮到讨论「晚睡的成因是工时、时区协作还是文化」。

一个顺手的副产品

养成「先 whether」的习惯后，你会发现一个意外收获：**很多原本要吵半天的争论，在 whether 阶段就死了**。饭桌上大量争执，争的都是 how 层（「到底是这个原因还是那个原因」），而双方共享的 whether 从来没验证过。把 whether 摆上台面，要么发现现象本身不存在（争论解散），要么发现双方说的根本不是同一个现象（争论换轨）。无论哪种，都省下了一晚上。

下一章换一个陷阱：就算前提是真的、句式也没问题，问题照样可以坏掉——如果你用来推理的那个类比，本身就是错的。

第五章 · 你的类比是不是偷换了东西

回到那顿晚饭的第二战场。讨论法务怎么挑律所时，杨明举了个例子：我们公司入驻中粮储代理商合作库的经历。他举这个例子是为了反驳我，但复盘时我发现一个微妙的事——**这个例子恰恰展示了另一种思维陷阱：用错位的类比推进讨论**。而且这次踩坑的不只是我，是桌上的双方。

类比是怎么工作的

类比是人类推理的主力发动机。遇到陌生的 A，找一个熟悉的 B，说「A 应该跟 B 差不多」。它之所以能用，依赖一个隐藏等式：**A 和 B 在「和这个问题相关的那几个结构」上相似**。

关键词是「相关」。两个东西可以在一百个维度上相似，但只要相似的不是和当前问题有关的那几个维度，这个类比就是哑弹。讨论「A 会不会像 B 那样发展」，需要的是因果结构相似；而大多数脱口而出的类比，相似的是表面——都带「入库」两个字、都发生在企业采购场景、都有个名单。

可以给它起个名字：表面相似，指两个事物在词汇、场景、外观上相像；与之相对的是结构相似，指两者在「谁决定、谁掏钱、错了谁负责」这类因果关节上相像。类比要有效，必须结构相似；而我们随手抓来的类比，几乎总是表面相似。

中粮储入库 ≠ 法务挑律所

把这两个东西摆上解剖台：

- **我们入驻中粮储代理商库**：我们提供的是相对标准化、基础的服务，在这个细分领域里规模算大的。采购方对这类服务的核心诉求是「稳定、合规、别出事」，入库门槛主要看资质和规模。一旦入库，长期合作，换供应商的麻烦大。
- **企业法务挑律所**：法律服务的核心是「具体的人」。同一个红圈所里，不同合伙人的水平、投入度天差地别。采购方真正挑的是「那个律师」，律所招牌只是初筛的门票。而且「入库」

（进名单）和「中选」（拿到具体案子）是两道工序，中间隔着具体的人情、信任和个案比选。

两者表面多像啊：都是企业采购外部服务，都有个「库」。但结构差远了：一个是买标准化的「东西」，一个是买具体人的「手艺」；一个库就是终点，一个库只是起点。**拿中粮储的经验去推「法务怎么挑律所」，相当于拿自动售货机的选品逻辑，去推餐厅怎么选厨师。**

大白话

大白话：类比像充电器转接头。样子对不上当然插不进去；但更要命的是样子对上了、电压不对——插得进去，一用就烧。表面相似管「插得进去」，结构相似才管「能不能用」。

类比陷阱和第三章的区别

值得分清：第三章的以偏概全，是「拿三个个例当普遍现象」——错在**证据的数量**；这一章的类比错位，是「拿一个 B 当 A 的预言机」——错在**证据的种类**。前者问你「数够了吗」，后者问你「这是一回事吗」。一个喂给问题错的事实，一个喂给问题错的推理通道。

类比错位还特别难防，因为它有传染性：讨论中一方抛出错位的类比，另一方很容易顺着这个类比往下打——就像那顿饭，我们沿着「入库」讨论了半小时，而这个轨道本身就是错的。纠正类比的成本远高于抛出它。

三个校验问题

下次自己用类比、或别人对你用类比，先问三句：

1. **这个类比里，「决定结果的那个人/机制」是谁？两边是同一个角色吗？**（中粮储：采购流程；挑律所：具体合伙人的信任。不同。）
2. **两边「选错了的代价」和「更换的成本」一样吗？**这两个变量几乎决定了一切采购行为。
3. **如果把表面特征全换掉（把「入库」改名「登记」，把「律所」换成「设计工作室」），这个推理还成立吗？**成立，说明抓到了结构；不成立，说明靠的是字面。

第三句最好用：它逼你把类比「翻译」掉表面词汇，看看剩下的骨架还站不站得住。

不过那顿饭上，类比错位还不是唯一的坑。杨明还指了一个更细小、也更常见的毛病：我说的「律所」和他说的「律师」，压根不是一个词——下一个陷阱，一个词，两种用法。

第六章 · 一个词，两种用法

上一章结尾提到，那顿饭还埋着一个更细小的坑：我说的是「律所」，杨明纠正的是「律师」。就差一个字，但正是这一个字，让那场讨论原地打转了半小时。这一章拆的陷阱叫概念含混——大白话讲：一个词在你的句子里干两份不同的活，而你却没有察觉。

一个词，两份工作

「律所会怎么入选企业的法律顾问名单？」这个问题里，「律所」至少扛着两份完全不同的工作：

- **律所 = 机构**：一个品牌、一套资质、一份名录上的名字。这层意义上，「入选」看的是规模、排名、行业口碑。
- **律所 = 具体办案的律师**：某个合伙人和他的团队。这层意义上，企业真正「选」的是人——你信不信任他、他懂不懂你的行业、出了事他顶不顶得上。

当我问「是不是只有红圈所才能入选」时，我用的是第一层意思；杨明反驳「入选的往往不是律所而是律师」时，用的是第二层。**我们争了半天，其实没有分歧——只是同一个词在两个轨道上跑**。这是概念含混最气人的地方：它不产生错误答案，它产生「伪争论」。双方都觉得自己在反驳对方，其实像两条平行铁轨上的火车，呼啸着互相错过。

那顿饭上还有一个同款：「入库」。我说「进入法律顾问名单」，默认入库就是「被选上了」；杨明说「入库往往只是程序化的一步」，意思是入库只是「有资格被挑」。一个是终点，一个是起点。又是一个词，两份工作。

为什么大脑容忍一词两用

因为省劲。语言给每个概念发了一个词当标签，大脑默认「标签一样 → 东西一样」。这在日常里九成够用——谁会在点菜时纠结「番茄」是植物学果实还是烹饪学蔬菜？但在讨论里，一个词的两份工作如果**结论不同**（机构层面「红圈所有优势」，人的层面「牌子只是门票」），含混就不再是无害的了：它让你把两个层面各自的证据，错装到对方的结论上。

工程师对这事其实不陌生——这就是**变量遮蔽**：同一个变量名，内层作用域偷偷换了值，外层的计算全错。只不过代码里编译器会警告，饭桌上没有编译器。

大白话

大白话：概念含混的讨论就像两个人打乒乓球，但用的是两只不同的球。你以为在回对方那一板，其实每次挥拍打的都是自己那只球——打得再热闹，比分永远是零比零。

拆弹动作：给关键词「点名」

防这个坑的办法土得掉渣，但极其有效：**讨论前，把问题里的关键词挨个拉出来，强制说出它指什么。**

拿「法务是不是只认红圈所」做示范，先点名三个词：

- 「认」= 入库的资格筛选？还是具体案子的委托决定？——这是两道工序。
- 「红圈所」= 机构品牌？还是「出自红圈所的那个合伙人」？
- 「只认」= 100% 排除其他所？还是「同等条件下优先」？

点完名，原来的问题会自动裂变成几个更小、更清楚的问题，比如：「在入库资格筛选这道工序上，机构品牌占多大权重？」——这个小问题的前提可查、概念有边界，终于配得上认真讨论了。

有个好用的测试叫换词测试：把关键词换成一个更具体的说法，看句子意思变没变。「律所入库」换成「张三律师所在的机构进了名单」，如果你的结论忽然说不通了，说明刚才的推理全靠那个含混的词撑着。

至此，四个陷阱齐了

回过头数：第三章的以偏概全（证据数量造假）、第四章的夹带为什么（前提走私）、第五章的类比错位（推理通道用错）、这一章的概念含混（词汇一词两用）。这四个就是那晚饭桌上坏掉的全部零件，也是大多数人提问时最高频的四个坑。

知道坑长什么样还不够——坑在暗处，你得有工具把它挖出来。下一章开始从「认坑」转向「拆弹」：一套把前提从问题里拆出来的三步操作。

第七章·把前提从问题里拆出来

前四章认完了坑。从这一章开始进入「怎么办」。第一个要练的动作，是把前提从问题里拆出来——因为前提最大的保护伞就是「没说出来」，一旦写成白纸黑字，它的一半魔力就消失了。

整个操作只有三步：**找前提** → **查前提** → **给前提找反例**。拿一个新鲜例子全程走一遍：「现在的年轻人为什么都不买房了？」（饭桌上又一个典型的高谈阔论发动机。）

第一步：找前提——把「为什么」翻译成陈述句

操作：把问句改写成它背后那个陈述句。语法上很粗暴——把「为什么」删掉，把问号换成句号：

- 「年轻人为什么都不买房了」→「年轻人都不买房了」。
- 「律师为什么都转去做刑诉」→「律师都转去做刑诉」。

然后对这个陈述句做第六章的「点名」：里面的词各指什么？「年轻人」是多少岁？「都」是多大比例？「不买」是从未买，还是推迟买？一个问句往往拆出不止一个前提——这个例子里至少有两个：① **年轻人购房率显著下降了**；② **下降的原因是「不想」而不是「买不起/不需要」**。很多人忘了第②条：一个「为什么」问题，除了现象前提，通常还驮着一个**归因前提**——它默认原因在某处。第②条不点名，讨论就会直奔「观念变了」而去，完全绕开「首付变了」。

第二步：查前提——给每条前提标一个「证据等级」

前提拆出来后，逐条问：我凭什么相信它？诚实回答通常只有四档：

1. **查得到的硬数据**：统计、报告、公开记录。（「购房率」可以查到，虽然口径有坑。）
2. **可靠转述**：懂行的人、当事人说的，但没法立刻核实。（杨明那句「北大法学院很多人去了金融和法务」属于这档。）
3. **亲身见闻**：我见过的几个例子。——第三章说了，这档证据撑不起任何带「都」的前提。

4. **感觉**：「大家都这么说」「网上都在传」。这档其实不是证据，是回声。

规则只有一条：**前提的证据等级，决定问题能问多大**。手里只有第三档证据，问题里的词就得降级：「都」变「我见过的几个」，「普遍」变「可能」，「为什么」先降级成「是不是」。这不是认怂，是让问题的大小和证据的厚度匹配——小证据驮大问题，就是那晚我翻车的力学原理。

大白话

大白话：前提像桥墩，问题像桥面。三步操作就是：先把桥墩从水下捞出来（找），再敲敲看是不是空心的（查），最后开辆重车上去试试（找反例）。捞不出来的桥墩，你敢开车上桥吗？

第三步：给前提找反例——主动当自己的杠精

前两步是防守，这一步是进攻：**花两分钟，认真搜肠刮肚，给每条前提找一个反例**。

「年轻人都不买房」——反例并不难找：二三线城市家里资助首付、结婚上车的同龄人，你大概认识几个。找到一个，前提里的「都」就死了；前提降级成「一部分」，问题的形状立刻变了。

这一步的反直觉之处在于：它和你想讨论的方向对着干。你兴冲冲想聊「为什么不买」，却要主动去找「买了」的例子——心理上像拆自己的台。但这正是它值钱的地方：反例是前提的压力测试，而且便宜，两分钟一个。饭桌上那个更贵的版本是：你不自己找，你的杨明会帮你找，当着一桌人的面。

如果认真找了两分钟，一个反例都找不到呢？恭喜，这反而是前提结实的好迹象——但只在你真的认真找了的前提下成立。

三步合成一个习惯

串起来，以后任何重要的问题出口前，过一遍流水线：

问句 → 翻译成陈述句并点名（找） → 每条标证据等级、问题大小随证据降级（查） → 每条认真找两分钟反例（攻）。

走完这三步还站着的问题，就基本「及格」了。但及格和优秀之间还有差距——下一章给「好问题」定四个零件，让你不只避开坏问题，还能主动造出好问题。

第八章·好问题的四个零件

上一章的三步操作是「排雷」——把坏问题拆掉。这一章反过来，正面回答那个贯穿全书的问题：**一个问题凭什么算「问对了」？**把前七章的教训攒起来，一个好问题有四个零件。它们不是玄学标准，每一个都能当场检验。

零件一：前提可查证

第一到第四章都在说这件事的反面，这里说正面。好问题的每条前提，都得能回答「去哪查」。「转型律师选刑诉的比例」——查得到（律协、招聘数据）；「律师都转刑诉」——查不到，因为「都」这个量级在任何数据里都不会出现。**检验方法：对着每条前提问「如果我要向一个不信的人证明它，我去哪找证据？」**答不上来的前提，就是悬空的。

注意「可查证」不要求「已查证」。饭桌上不可能现场翻数据库，但你得知道证据存在吗、在哪。「可查而未查」是效率问题，「根本不可查」是结构问题——后者的问题不该出口。

零件二：概念有边界

第六章的正面版。好问题里的每个关键词，都能通过**换词测试**：换成更具体的说法后，句子意思不变、结论还成立。

「年轻人不买房」——「年轻人」换成「25 到 35 岁、一二线城市无房者」，问题还成立吗？如果一换就散架，说明原来的词是个橡皮口袋，什么都装得下。边界不需要精确到小数点，但需要「反例落在哪」是清楚的：什么算「不买」？这辈子没买，还是五年内没买？**概念有边界的标志是：你能说清楚什么情况算这个问题被「答错」了。**

零件三：答案可证伪

这是最容易被忽略、也最区分高下的一件。一个问题的答案，必须有可能是错的——存在一个「什么样的证据能推翻它」的想象。

反例来得最快的地方是自证循环的问题：「为什么靠谱的人都离开了这家公司？」——如果有人举出一个留下的靠谱同事，提问者随时可以把他划出「靠谱」的定义。定义跟着结论

走，答案就刀枪不入，也就一文不值。这种问题和答案之间没有摩擦力，讨论它不会产生任何新东西。

大白话

大白话：好问题像一把能卡住的卡尺——有些刻度它就是读不出来，所以它读出来的数才有意义。什么问题都能「解释」的问题，等于什么都没解释。

检验方法：问「如果答案是错的，世界会长什么样？」想不出那个世界的样子，这个问题就没有证伪面。

零件四：结果有分量

最后一件和逻辑无关，和「值不值」有关：**这个问题的不同答案，会导致不同的行动吗？**

「法务是不是只认红圈所」——如果你是一家小律所的合伙人，答案直接决定你的获客策略，有分量。但如果只是饭桌谈资，四个零件凑齐三个也行，别苛求。反过来，有些问题前三个零件都合格，却无论怎么答都不改变任何事（「宇宙的尽头是不是编制」聊完照原样生活）——它不是坏问题，是轻问题。值得识别，因为人的讨论预算有限，应该优先把认真花在重问题上。

四个零件合起来用

把四个零件做成出口前的最后一道闸：

1. **前提可查证：**每条前提知道去哪找证据。
2. **概念有边界：**关键词过换词测试，知道什么情况算答错。
3. **答案可证伪：**想得出「答案错时世界的样子」。
4. **结果有分量：**不同答案指向不同行动。

回到最初那句「为什么那么多律师转去做刑事诉讼」，用四件一照：前提不可查（「都」「那那么多」） $\times$ ；概念还行 $\triangle$ ；可证伪性勉强 $\triangle$ ；分量看你是谁 $\triangle$ 。四个零件坏一个核心件，整个问题返厂——改造后的「转型律师选各方向的比例是多少」则四个全绿。

零件清单在手，还差一个陪练。自己一个人练提问，最大的障碍是「自己看不出自己的毛病」——下一章介绍一个 24 小时在线、不知疲倦、专挑你毛病的陪练：AI。但要先用对姿势，否则它比饭桌上的老好人还害人。

第九章·把 AI 当杠精用

上一章结尾埋了个转折：AI 这个陪练，用错姿势比不用还害人。这一章讲姿势。先说清楚一个很多人没意识到的事实——**你平时用 AI 的方式，恰好是放大你提问毛病的最佳方式。**

AI 的默认人格是「顺从的优等生」

回忆第二章那条链：前提错了的问题，会把认真的回答者拖进虚构世界开会。人类朋友里，只有杨明这种不怕得罪人的才会掀桌子；大多数老好人会顺着你的问题往下聊。而 AI——绝大多数聊天 AI——被训练成的默认人格，就是一个**永不掀桌的老好人**：你问什么，它答什么，答得又快又周全又客气。

这意味着：你问「为什么律师都转去做刑诉」，它会立刻给你三条理由，每条都有模有样。你的错误前提不但没被挑战，还被它用流畅的论述**加固了**——现在你不光有个假前提，还有了一整套听起来很专业的「解释」。这比不问还糟：不问，前提只是悬空；问了，前提被浇筑了水泥。

大白话

大白话：直接拿问题问 AI，等于雇了一个永远不反驳你的下属。它会把你每一句「我觉得」都发展成一份论证充分的报告——包括那些本来就该被拦下来的。

换姿势：别让它回答，让它攻击

转机在于，AI 的人格是被你的指令设定的。你让它当顺从的助手，它就是助手；你让它当杠精，它能当得比杨明还敬业——不知疲倦、不怕你难堪、凌晨两点也火力全开。关键是吧「请回答」换成「请攻击」。三段可以直接抄走的提示词：

第一段：前提攻击。

我接下来要问你一个问题。但请先不要回答它。请先做一件事：找出这个问题里所有默认成立、但没有被我验证过的前提，逐条列出来，并说明每条前提如果是假的，我的问题会变成什么样。

第二段：概念点名（第六章的动作外包给它）。

请检查这段话里的关键词：有没有一个词在两处表达不同意思？有没有哪个词换成更具体的说法之后，整段话的意思会发生变化？

第三段：反例征集（第七章第三步外包）。

请站在反方，为我的前提找三个最有力的反例或反证方向，越具体越好，不要客气。

这三段本质上就是把第七、第八章的手工活，外包给一个廉价又卖力的陪练。它帮你把「找前提、查前提、找反例」从两分钟的心理挣扎，变成一次回车。

三个必须守住的边界

但「AI 杠精」不是免检产品，用的时候守住三条边界：

- **它的攻击也可能是错的。** AI 找出的「前提问题」需要自己再核一遍——它有时会为了显得犀利而硬挑毛病，把没问题的词也说成含混。把它当陪练，别当裁判。最终拍板的还是你。
- **它查不了你的「身边」。** 前提里涉及你私人的见闻、行业内部情况，它无法验证，只能推理。第三、四章讲的「数量关」「来源关」，还是得自己过。
- **别让它替你提问。** 「帮我想一个好问题」是合法的用法，但如果每次都由它生成问题，你练的是它，不是你。第十章会说，提问是要长成自己肌肉的东西，AI 是哑铃，不是假肢。

一个诚实的复盘

回到这本书的起点：我那晚之后把问题抛给 DeepSeek，它指出了我提问逻辑的问题——这是它做对的地方。但公平地说，如果我当时用的不是「你看看我的问题有没有问题」，而是直接问「为什么律师都转刑诉」，它多半已经洋洋洒洒替我解释了一个不存在的现象。

同一个工具，两种姿势，结果相反。区别只在一件事：**你让它站在你问题的对面，还是站在你问题的后面。**

工具和清单都齐了。最后一章解决最难的一件事：怎么让这套功夫不靠清单、变成条件反射。

第十章 · 提问是可以训练的思维

九个零件、三步操作、一个陪练，全都在手了。但还有最后一道坎，也是最难的一道：**饭桌上的你，没有清单。**

复盘时人人都能当诸葛亮——我现在把那晚的问题看得清清楚楚，可当时我一个都没察觉。问题出口只要两秒，没人来得及掏「四零件检查表」。所以这本书的最后一章只解决一件事：怎么把「问对」从一张清单，变成条件反射。

思维模式为什么「看不见自己」

先接受一个让人不太舒服的事实：**你在提问的那两秒里，是看不见自己的思维模式的。**模式这个词的意思，就是「不假思索、自动运行的流程」——眼睛看不见眼睛本身。以偏概全不是我做的一个「决定」，它是我大脑后台的默认算法，运行时没有任何提示音。

所以指望「下次提问时注意点」是没用的——「注意点」这件事本身，也需要被注意才能发生。能改模式的，从来不是「当场更努力」，而是**在事后留下记录，让模式在复盘时显形，再靠重复把修正后的版本写成新的默认。**这和健身一个道理：你没法在发力那一秒纠正姿势，但你可以录像、回放、下次空杆重来。

工具：一本「问题日记」

具体的做法朴素到有点不好意思：准备一个本子或一个文档，只记三类条目。

第一类：被否决的问题。每次有人（或 AI）说「你这个问题不对」，原样记下来——不是记教训，是记原句。原句里保留着作案现场的全部细节：那个「都」字、那个「为什么」、那个一词两用的关键词。这本书的第一章，其实就是我这类日记的第一页。

第二类：侥幸过关的问题。问出口了、没被反驳、但事后想想不太对的问题。这类最珍贵也最容易丢——没有外部耳光，全靠你自己补记。它们是你模式里「还没被任何人指出」的部分，是你的杨明还没坐到你旁边的那些晚上。

第三类：改造成功的问题。出口前拦下来、按三步操作改写过的问题。记一对：原句 vs 改后。这类条目是你的正反馈——没有它，日记会变成一本纯纯的耻辱柱，坚持不了两周。

大白话：问题日记就是给自己的提问装行车记录仪。开车时你感觉不到自己的变道习惯，但回放三个月的录像，「哦，我并线从不看右后」这件事自己会跳出来。

怎么用：每月一次「模式审计」

日记的价值不在记，在定期回看。每月花二十分钟，把当月的条目扫一遍，只回答一个问题：**这些条目里，有没有重复出现的同一个毛病？**

单个条目是事故，重复出现的条目才是模式。我的日记如果记下去，大概率会看到「都」「普遍」「为什么」高频出现——那就说明我的默认算法里，「小样本直接升级成全称判断」这条捷径权重太高。审计的产出也极简：**下个月，只盯这一个毛病**。一次修一个，因为模式是默认值，改默认值的全量重写注定失败，补丁式迭代才有机会。

坚持几个月后会发一件事：某个晚上，一个问题到了嘴边，你会在**出口前**听见一声很轻的「等等」。那声「等等」，就是清单长成条件反射的声音。它不快——提问这件事，大脑的默认设置是进化了几百万年的，你才练了几个月——但它会来得越来越早：从事后复盘，到出口后的「糟了」，到出口前的「等等」。这个迁移过程本身，就是训练的全部意义。

收尾：回到那顿晚饭

这本书从一顿被否决的晚饭开始。现在可以给杨明那句「你的问题本身有问题」一个更准确的翻译了：他不是在我那三个句子有语病，他是在说——**提出它们的那套流程，该检修了**。

贯穿全书的问题「一个问题凭什么算问对了」，答案收拢成一句话：**前提可查、概念有边、答案可证伪、结果有分量——以及，提问的人养得起一本问题日记**。前四条管单个问题，最后一条管提问的人。问题会过时，人会一直提问。

顺便说，我至今不知道那晚第三个关于离婚的问题错在哪——没来得及展开。但我现在有了工具：下次见到杨明，我可以把它重新问一遍。这次，先问「是不是」。

祝你下一顿饭局，问出一个让全桌安静三秒的好问题。

