AI 咬走的那一刀

被取代的从来不是「人」，是任务——那条分界线到底画在哪

holly

面向理工背景的好奇者

导读

被取代的从来不是「人」，是任务——那条分界线到底画在哪

你大概见过这种排行榜：「未来十年最危险的 20 个职业」，翻译在里面，客服在里面，画师有时也在里面。看完你心里咯噔一下，然后呢？它没告诉你**为什么**，只给了你一份名单让你对号入座。名单会过期，恐慌也没用。

这本书想给你的不是一张名单，是一副眼镜。戴上它，你自己就能看：手头这件事，AI 会不会咬走？

先说清这本书**不是**什么。它不预言未来，不排名职业，不谈「AI 会不会毁灭人类」这种听着宏大、其实帮不上你判断的话题，也不教你怎么调 prompt、用哪个工具。这些书已经很多了。

它**是**什么：一套判断规律。我把它拆成五条分界线——

- 这件事，能不能用大量例子说清楚？
- 做完了，有没有一个说得清对错的信号？
- 错了，代价谁来担？
- 要不要一个真身在现场？
- 是照着题目解，还是得重新问「题目是什么」？

五条线画完，任何一件事你都能自己量一量，不用等别人发榜。

贯穿全书的，是一句得先掰开的话：**机器从来没有取代过一个「人」，它取代的是一个「动作」。**「翻译」不是一个人，是一堆动作——查词、拼句、对上下文、拿捏语气。机器把好描述、有标准答案的那几个接了过去，剩下的还留在人手里。所谓「某某职业被取代」，其实是这个职业里的动作被一刀一刀切走，切到某个位置停下。那条线画在哪，就是这本书要回答的全部。

大白话

想明白这一句，你会松一口气：该焦虑的从来不是「我这个人还有没有用」，而是「我每天做的这些动作里，哪些经得起这五条线」。前者无解，后者你今天就能开始算。

翻过这页，我们从最容易被误会的那句话讲起：AI 到底在干什么。

第一章 · 「取代」这个词，一直被用错

先说一个你大概听腻了的句子：「AI 要取代人了。」它每隔几年换副面孔回来：上世纪是「机器人要抢工人饭碗」，前些年是「自动驾驶要让司机失业」，现在是「大模型要让程序员、画师、律师下岗」。每次都吵得脸红脖子粗，谁也说服不了谁。争不出结果不是谁蠢，是这句话问错了。

「取代人」是个偷换过的概念

做件工程师爱做的事：把它拆开。「AI 取代会计」听上去像是某天你走进办公室，那个叫老张的会计人没了，位置上摆了台服务器。但现实不是这样。老张每天做十几件事——录凭证、对账、算工资、报税、跟税务局解释一笔奇怪的支出、老板临时改主意时熬夜重做预算——这里头某几件被机器接走，剩下的还得他干。

所以「取代人」一开始就把镜头对错了：它盯着「人」这个整体，可机器从来不是整个吞人的——它咬走的是**动作**、是任务，是一个人一天里许多件事中的几件。

> 大白话
>
> 换句话说：机器不抢「岗位」，抢的是「活儿」。一个岗位是一大捆活儿，机器一次只咬得动几条。

这里先把全书最重要的一个词立住。任务，就是一件有头有尾、能单独拎出来说清楚的具体活儿——「把这张发票上的金额录进系统」是一个任务，「让老板对这季度的财务状况放心」不是，那是一堆任务加判断和信任的结果。全书谈的都是任务这个颗粒度，不是「岗位」。

历史早就演过这出戏

如果「机器咬走的是任务不是人」成立，历史上每次技术冲击都该留下同一种痕迹：人没消失，但每天干的事换了一批。

ATM 和银行柜员

ATM 就是那台你插卡取钱的自动取款机。它在上世纪七八十年代铺开时，所有人都以为柜员要完蛋——它干的正是柜员的活。

结果呢？那几十年里，美国银行柜员总人数不但没减，还涨了。机制不玄：ATM 把「数钱、点钞、办取款」这几个最枯燥、最占时间的任务拿走，养一个柜员的成本降了，银行就用同样的钱多开网点，柜员省下的时间被推去干机器干不了的活——跟客户聊，卖贷款和理财。柜员这个「人」没被取代，是他手里那几条**数钱的任务**被取代了，岗位里换上了销售和关系。同一头衔，活儿换了血。

> 大白话
>
> 关键在这里：机器把某些任务变便宜，反而让整件事做得更多、更划算，于是需要更多人去干机器碰不了的部分。省钱有时候是加人，不是减人。

Excel 和会计

再看个更近的。Excel 这类电子表格软件，就是那个能自动算总和、拉公式、改一个数满屏跟着变的东西。它出现之前，账房里有一群人专门「重算」：老板问「假如销售多 10%，利润是多少」，就得拿计算器把整张表从头按一遍，一个假设按一晚。

电子表格一来，这个任务几乎一夜归零。纯敲计算器的岗位确实少了，但会计这行整体没塌，反而膨胀了：「重算」几乎不要钱后，公司开始做以前懒得做的事——十种假设、滚动预测、成本分析。会计的活儿从「把数算对」升级成了「用算出来的数去想问题、给建议」。

机器接走的是**计算**，交还给人的是**判断**——而后者更值钱。

更老的例子：织布机

再往前，工业革命的织布机确实让一批手工织布的工匠丢了饭碗，别粉饰，冲击是真的、痛的。但另一半常被略过：织布这个「体力+重复」的任务被咬走后，出现了大量新任务——看管、上料、接断线、修机器。任务被重排，有人被甩下车，有人换个活法留下。

那么，这本书真正要找的东西

三段叠在一起，一个模式浮了出来：

> 每一次技术进步，都不是「一台机器换掉一个人」，而是「一把刀落进岗位，切走里面某一类任务，留下另一类」。

关键词是**某一类**。被切走的任务都有种共同气味：能说清步骤、能重复、对错分明；留下的都黏糊糊的，带着判断、关系、责任、临场应变。这条把「被咬走的」和「留下的」分开的界线，就是本书唯一要答的问题：

大白话

AI 能接手「哪一类」任务、接不了「哪一类」，背后到底是什么规律？这条分界线画在哪儿？

先泼盆冷水：我**不**打算给你一张「这 50 个职业会消失、那 50 个安全」的清单。它三个月就过时，还让你只会背不会想。

我想给你的是一副能自己判断的眼镜：读完之后，你随便拎出一件事——工作里的某个动作、犹豫要不要学的某项技能——都能自己拆开，看清哪几条任务正暴露在刀口下、哪几条暂时安全，以及**为什么**。

磨这副眼镜分两步。先搞清楚 AI 这把刀是怎么造出来的——它「学」东西的方式决定了它天生咬得动什么、咬不动什么（第三章）。再找出几条真正管用的分界线：一件任务能不能被数据描述清楚、有没有说得清对错的信号、错了谁担责、要不要真身到场、是照着做还是重新定义问题。这五条线，就是我们要一根根画出来的。

但在动刀之前，下一章先讲透一个更基础的东西：为什么「一个岗位其实是一大捆任务」值得认真对待——一旦你用「任务」而不是「岗位」去看世界，很多焦虑会突然变得可以计算。

记住这一章唯一的那句话就够：**被取代的从来不是「人」，是任务。**

第二章 · 一个岗位不是一件事，是一捆任务

上一章立了一句话：被咬走的是任务，不是人。但那只是结论——面对一个具体职业，你还不知道从哪儿下手。这一章教你一个动作：**把一个职业拆成一张任务清单。**

先说清楚：职业、任务、动作

三个颗粒度，从粗到细：

职业，就是一个头衔——「放射科医生」「后端工程师」，招聘网站上那一行字。它太粗，粗到没法分析，其实是一大团东西打包在一起。

任务，是这个职业每天真正完成的一件件有头有尾的事——「看一张 CT 找出结节」「跟病人解释诊断」。这是本书唯一关心的层。

动作，是把一个任务再拆小后的操作步骤——「读片」拆开是「调出图像、逐层扫过、标出可疑区域、跟历史片对比、写结论」。拆到这层，你才看得清机器能咬进哪一格。

> 大白话
>
> 一句话记住三层：职业是一个抽屉柜，任务是里面一个个抽屉，动作是抽屉里的小东西。AI 从不搬走整个柜子，它一次只拉开一个抽屉，看有没有它拿得动的。

动刀：拆一个放射科医生

放射科医生，大概是这几年被喊「要被 AI 取代」喊得最凶的职业之一。别争对不对，先拉开抽屉看几件活儿：

- **读片**——盯着 CT、X 光、核磁的图像，找出不正常的地方（结节、骨折、阴影）。
- **下诊断**——结合病人年龄、病史、其他检查，判断「大概什么病，严不严重」。
- **跟病人沟通**——把冷冰冰的结论，用一个吓坏了的人能听懂、能接受的方式讲。
- **跨科会诊**——跟外科、肿瘤科的医生商量「这病人接下来该怎么办」。
- **担责**——在报告上签下名字。漏诊、误诊，是这个人负法律和职业上的责任。

大多数人的错误此刻暴露了：问「AI 会取代放射科医生吗」，想的是整个抽屉柜。可抽屉一拉开就发现，这五件活儿机器咬起来的难度天差地别。「读片」——图像输进去，输出「这里有个可疑的点」——恰恰是机器最擅长的：有海量标好答案的历史片可学，对错分明，最容易掉下来。

而「跟一个刚听说可能得了癌症的人好好说话」「签名承担后果」——这几格你隐约感觉得到，机器伸不进手。**为什么**先按住不说——那正是第 4 到 8 章五条分界线要讲的。这一章你只要看清：「放射科医生会不会被取代」根本是个错问题。真问题——

> 这五个抽屉里，哪个先被拉开？拉开后，剩下的四个会变成什么样子？

再拆一个：后端工程师

换个理工读者更熟的职业，这刀照样通用。「写代码的」听着像一件事，拆开也一捆：

- **敲样板代码**——每个项目都差不多的骨架，增删改查、接口拼装，闭眼也能写。
- **查文档、查报错**——「这个函数第三个参数是啥来着」「这行报错到底啥意思」。
- **把需求拆成技术方案**——「我要个让用户互相点赞的功能」，你得把这句人话翻译成一堆技术步骤。
- **定架构**——决定整个系统怎么搭、以后怎么扩、哪里先崩。定错了，半年后全组陪葬。
- **扛线上事故**——半夜三点系统挂了，这个人爬起来救火，并为「为什么挂、以后怎么防」负责。

跟放射科医生同一个形状。前两格（敲样板、查文档）边界清晰、重复、有海量现成答案，牙齿最先咬住，用过 AI 编程工具的人已经感觉到。后三格（拆需求、定架构、扛事故）越来越黏糊，掺着判断、取舍、「出事了算谁的」。**同一个头衔，这几刀掉下来的先后隔着十万八千里。**

拆完之后，看到的那个规律

两个八竿子打不着的职业，拆开却露出同一副骨架：一捆任务里，总有几条**边界最清晰、最重复、答案最现成**，最先暴露在刀口下；剩下的判断、关系、责任、临场应变，反而因累活

被拿走而更值钱。这跟上一章 ATM 和 Excel 一码事，只是这回你自己拆出来的。

还有一件事：被拿走几条，剩下的不会原地待着，会**重新打包成一个新岗位**——这叫任务重构：剩下的活儿重新组合、加进新活儿，长成一个跟原来同名、内里却换了血的岗位。银行柜员没消失，但今天干的是销售和关系；会计没消失，今天干的是分析和建议。放射科医生和工程师大概率也这样。

> 大白话
>
> 所以「这个岗位会不会消失」几乎永远是个错问题。对的问法分两步：先拆成任务清单，再问**哪几条**正暴露在刀口下、哪几条暂时安全。会不会消失，取决于剩下那几条能不能打包成一个还养得活人的新岗位。

把刀交到你手上

这个动作是本书的解剖刀，后面每章都用它。趁热试一刀：挑一个你熟的职业——最好你自己的——别想「会不会被取代」，把它每天做的事列到十条以上。

列完你多半会意外：原来「我的工作」不是一件事，是一大捆不一样的活儿。有些边界清清楚楚、天天重复；有些黏黏糊糊，掺着判断和人情说不清——那种「说不清」正是刀锋落下的地方：**能说清步骤、对错分明的先掉；说不清、道不明的，暂时留在你手里。**至于为什么偏偏这几条先掉，是下一章的事——得先搞清楚 AI 这刀是拿什么造的。

第三章 · AI 到底在「学」什么

上一章讲清了：机器咬走的是任务，不是人。可要知道它咬得动哪些、咬不动哪些，得先弄明白这把刀怎么造的——刀刃的形状，决定它天生能切什么。这一章就把这波 AI 的核心「大模型」拆开看：它到底在「学」什么。

它的看家本事，就是猜下一个词

大语言模型（英文缩写 LLM，你就理解成「一个吃了海量文字、专门猜下一个词的超大机器」），听着高深，干的活其实土得掉渣：**给它前面一段字，让它猜下一个字最可能是啥。**

你天天在用它的迷你版：手机输入法联想。你打「今天天气真」，它跳出「好」「不错」「热」。凭什么？因为它见过无数人打「今天天气真」，后面接「好」最多。它不懂天气，只记住了「这几个字后面通常跟哪个」。

大语言模型就是这个游戏玩到极致。把输入法那点见识，换成几乎整个互联网的文字——书、代码、论坛、维基、说明书——让它反复做同一道填空题，上千亿遍：

> 大白话
>
> 「水的沸点是一百___」——填「度」。「for i in ___」——填「range」。「床前明月___」——填「光」。就这一道题，换着花样，做到天荒地老。

训练它，说穿了就是拿一篇真文章遮住后面让它猜；猜错了就把内部的旋钮拧一拧，下次更可能对。几千亿次后越猜越准。没人教它语法、没人给它讲道理，它只是被逼着「把下一个词猜准」。

可猜词怎么就会写代码了？

一个只会填空的机器，怎么就能翻译、对话、写出能跑的代码？关键这句话：**要把下一个词猜得足够准，它被逼着在肚子里塞进了大量世界的规律。**

要接对「水的沸点是一百」后的「度」，它得「知道」这条物理常识；要接对 Python 下一行，它得摸清代码结构——哪个括号要闭合、变量定义过没；要把「猫」翻成「cat」，它得在无数中英对照里把两词的关系压进去。

换句话说：**猜词准**像绳子一样把它往「理解语言结构、掌握事实规律、熟悉代码套路」死死拽。它不是先学会世界再猜词，而是为了猜词顺手把世界的统计规律压进了肚子。会写代码、翻译、聊天，都是「猜词太好」溢出的副产品。

> 大白话
>
> 打个比方：你让一个人把一万部侦探小说的结尾都猜对，猜着猜着他就被迫学会了推理套路——哪怕从没上过逻辑课。目标逼出了能力。

它最擅长的事，叫「插值」

这是全章最要紧的一对词，理解了它，你就拿到判断 AI 的钥匙。第一个，插值——听着吓人，其实就是「在见过的点之间连线取中间」。你标两个点：「1 岁孩子平均多高」「10 岁孩子平均多高」，问「5 岁的呢」？两点连线取中间一读，八九不离十。这就是在已知范围**里面**填缝隙。

相对的是外推，就是「跳到没见过的范围外面去」。同样两个点，问「100 岁的人平均多高」？连线往外一延就荒唐——身高到某个岁数就不长甚至会缩，可直线不知道，没见过那片区。

大语言模型是**史上最强的插值机器**。它见过的文字铺成一片巨大空间，只要问题落在**里面**，哪怕这句话它没见过，也能在周围无数相似例子之间「连线取中间」给出漂亮答案。写请假邮件、把话改正式、用 Python 写个常见排序——都落在范围里，又快又好。

可一旦问题跑到范围**外面**——没人写过、没有先例、要凭空造出全新东西——它就被迫切换成「外推」，像那条冲向 100 岁的直线，越走越离谱还不自知。

> 大白话
>
> 记住这句，后面好几章都靠它：**它在见过的世界里游刃有余，在没见过的世界里连蒙带猜。**

诚实说局限：它不保证「懂」，还会一本正经胡说

讲到这得泼盆冷水。第一，它并不「理解」它说的话，至少不像你那样懂。它的目标只有一个：**让下一个词看起来像真的**——像一个懂行的人会接着说的话。注意，是「像不像真」，不是「是不是真」；两者多半一致，但不总是。

于是有了那个著名的毛病：幻觉——模型一本正经地编出一段**看着合理、其实是错的**内容：一个不存在的书名、一条查无此人的法条、一个编造的数据，语气还笃定得很。它不是骗你，它压根没有「真假」的概念，只是把下一个词接得顺口。顺口和正确，它分不清。

第二，它不会主动查证，也不为对错负责。你问它就答，哪怕心里一点底都没有，除非你给它接上查资料的工具。它给你的是「最像答案的一段话」，不是「负责的答案」。

更深一层：它学到的是**相关性**，不是**因果**。它知道「乌云」和「下雨」老一块出现，却不真懂「因为有乌云所以会下雨」。多数时候够用；可一旦要真正想清「谁导致了谁」，短板就露出来。

别神化，也别贬低

那该怎么给它一个公道的定位？收好这句话，这是全章的全部：

> 它是人类造出来的、有史以来最强的「模式匹配 + 插值」机器——仅此而已，但这已经惊人得不得了。

它惊人，是因为人类绝大多数工作本质上都在**已有的模式之间插值**，都落在它见过的范围里，接得又快又好，好到让人心慌。它有天花板，也正因为它是「插值机器」：凡是要跳到**范围外**、凭空定义新问题、为结果负责、弄清「谁导致了谁」的地方，它就打滑。这条边界不是谁故意设的，而是这套「猜词 + 插值」机制天生长成的。

大白话

收尾：它见过的世界越像你的问题，它越是神；越不像，越要你自己盯着。刀刃的形状，看清了。

接下来五章，我们一根根去画那些真正管用的分界线。第一条最贴本章：一件任务，能不能被**数据**说清楚？能被大量例子描述的，它插值得动；说不清、举不出例子的，就够不着。下一章见。

第四章 · 分界线一：这件事能不能被数据描述

上一章末尾的钩子：一件任务能不能被**数据**说清楚，是它能不能被 AI 咬走的第一道关。它最技术，也最根本——后面所有分界线都从这一条长出来。

先把话说死：AI 想接手一件事，前提是它能被表示成大量训练数据（喂给模型学的一对对例子，一头是输入、一头是对应的输出）。没有成对例子，它连从哪学起都不知道。

先倒下的，都是能摊成「输入→输出」的事

「插值机器」靠在见过的例子间连线取中间。它连得最顺的，就是能写成「给一个输入，就有一个公认输出」的事。看几个正在倒下的：

- **下棋**：输入是当前局面（棋盘上每个子在哪），输出是「这一步走哪」。几百万局就是几千万对。
- **翻译**：输入是原文，输出是译文。人类攒的双语对照就是现成的几亿对。
- **读片**：输入是医学影像，输出是「这阴影是不是病灶」的标注。医生标了多少年，就有多少对。

这三件事表面上八竿子打不着，可在 AI 眼里是**同一件事**：都是一堆「输入配一个对的输出」。例子够多它就能插值——给个新局面、新句子、新片子，它在相似例子间取中间，交出像样答案。它们先倒下不是因为简单，而是**数据化**得干净。

它接不住的，是每次都「第一次」的事

把这条线翻过来就清楚了：**凡是数据稀薄、或每次都「头一回遇到」的事，它够不着——因为没有可供插值的邻居。**你问「5 岁孩子多高」它答得准，是因为 1 岁和 10 岁的点在两边杵着；可整片区域没点，往哪连？这类事有三种常见长相：

- **全新的问题**：从没有先例的局面，比如一项刚出现的技术怎么定价。周围没例子，它只能硬着头皮外推，越推越飘。
- **稀有事件**：几十年一遇的极端情况。正因罕见，历史上没攒下几个例子，一出事反而最不靠谱。

- **要现编规则的场景**：不套用已有套路，而是当场造一套新玩法。例子里没有它，自然学不来。

大白话

一句话记牢：AI 强在「这种事我见过一万遍」，弱在「这种事人类也是头一回」。前者它插值，后者它抓瞎。

更狠的一刀：有些事，数据根本描述不了

你可能觉得，多攒点数据不就行了？但这里有个更深的区分：

> 「有没有数据」是一回事；「这件事能不能被数据描述干净」，是另一回事。

有些本事**原则上**就难以被例子完整写下来，它叫默会知识（也叫隐性知识，指「你会做，却说不清、也写不进例子」的那种本事）。

老师傅捏一把面就知道水放多了少了；老医生一进病房，说不上为什么就觉得这病人「不太对」；老司机在结冰弯道上那一脚，靠的是脚底的感觉。让他们把这套本事写成例子？写不出来——不是藏私，是它压根没在能被语言抓住的那层。哲学家波兰尼点得极准：

> 我们知道的，比我们能说出来的要多。

说不出来就喂不进去，模型就学不到。这不是数据「暂时不够」，是这类本事天生漏在数据的网眼外——比「例子少」更根本。

一个反直觉点：别看高级低级，看能不能数据化

这条线还能捅破一个流行的误会：人们下意识觉得越「高级、烧脑」的活机器越难碰。**这个直觉反了。**下棋在人类心里顶聪明，可数据化得干净，最先倒下。而**叠衣服**——三岁小孩都

会——机器到今天都磕磕绊绊。因为衣服软趴趴、每次形状都不一样，抓哪个角、用多大劲全是说不清的手上感觉，是默会知识，喂不进例子。

所以真正的分界线从来不是「这活高不高级」，而是「能不能摊成大量可靠的输入→输出例子」。（体力活这条长尾为什么难缠，第十二章掰开。）换轴记牢：**别问它聪不聪明，问它能不能被数据描述。**

诚实一句：数据是可以被「造」出来的

但结论不能太干脆。前面说「没有现成例子它就够不着」，可有一种情况能翻盘：**例子能被凭空造出来。**

最有名的是 AlphaGo 的续作。它不再啃人类棋谱，而是让两个自己对着下，下出海量人类从没走过的局面——这叫自我对弈（机器自己跟自己反复较量，边下边攒新例子）。数据不够，就用**模拟器**现造。围棋能这么干，因为规则明确、胜负一算便知，造出的每局都带着「这步好不好」的可靠答案。

所以「有没有现成数据」不是终点。更苛刻的问题是：**你能不能低成本造出数据，而且每一条都带着一个能判对错的信号？**围棋能，因为输赢一目了然；可换成写文章、哄客户，什么叫「对」？谁来判？这就是下一条分界线——**有没有对错信号**。

第五章 · 分界线二：有没有一个说得清对错的信号

上一章的分界线是：一件事能不能被**数据**说清楚——有大量「输入—输出」的例子摆着，机器就学得动。但光有例子还不够，还得有人能告诉它「这次是对是错」。这就是第二条分界线，它更隐蔽，却解释了一个反直觉的现象：**为什么 AI 在你我都做不出的难题上突飞猛进，却在「安慰一个哭着的人」这种小孩都会的事上原地打转？**

先看一个只会填空的机器，怎么学会「变强」

第三章说过，大模型的底子是「猜下一个词」。今天最能打的 AI 在猜词之上又叠了一层练法，叫强化学习——别被这四个字唬住，意思土得很：**让机器一遍遍去试，试对了给块糖，试错了不给，久了它就朝着「拿糖多」的方向长。**

这跟小孩学骑车一个道理。没人给他讲平衡公式，他就是一次次摔、一次次调，摔少了就会了。关键是那个**信号**：有人当场告诉它「这下摔了」「这下稳了」。

> 大白话
>
> 把这个信号叫反馈信号：一件事做完，有没有东西能马上、清楚地告诉你「行还是不行」。这一章从头到尾就讲它。

信号越干净，它涨得越猛

有了这把钥匙，来看为什么有些领域 AI 一日千里：

下棋。一盘下完，输赢没有争议、不用等，信号干净利落。于是 AI 自己跟自己下几千万盘，每盘立刻知道对错，输了就调、赢了就固化，不用人盯就日夜变强。

写代码。代码能不能编译通过（机器能不能把它翻成可运行的程序）、能不能跑过测试（预先写好的一批检查，对了亮绿灯、错了亮红灯），也是干净信号。灯当场就亮，机器改一版、跑一遍、看颜色，闭眼循环几百万次。

做数学题。解对不对，代回去一算就见分晓，信号又快又准。

三样共同点：**对错不用问人，一个规则、一次运行、一次验算就能自动判定。**于是：

> 凡是能「自动判对错」的任务，机器就能自己出无数道题、自己批改、自己进步，海量地练自己，卡住它的天花板会一节一节被顶穿。

信号一浑，它立刻抓瞎

掉个头，看它使不上劲的地方，问题全出在信号上。信号会以四种方式变坏：

- **模糊——说不清什么叫「对」。**什么叫「教得好」？分数高，学生爱上这门课，还是十年后他还记得？三个人三个答案。信号一模糊，机器不知道「好」指向哪，往哪拧都是瞎拧。
- **滞后——对错要等很久才揭晓。**育儿最典型。你今天对孩子的决定是好是坏，可能要等他长大成人才知道。机器靠「做完马上知道对错」来学，让它等十五年再反馈，这辈子也练不出几轮。
- **有争议——对错本身没有共识。**什么是「一篇好文章」「一份公正的判决」？连专家都吵翻天。没有大家都认的标尺，机器就没有稳定的靶子可瞄。
- **压根没有客观标准。**该不该原谅一个人、这段关系值不值得继续——有些事就是没有「标准答案」。不是暂时没找到，是本来就不存在能打勾打叉的答案。

大白话

信号干净，它就能自我打磨、越练越强；信号一浑，它连该往哪儿改都不知道。不是不努力，是找不着北。

于是有了那个反直觉的结论

两半拼起来，开头那个怪现象就通了。直觉上难的事 AI 该更晚才会，现实却反过来：竞赛题、刁钻的编程题进步神速；「安慰一个难过的人」「把团队带得心气顺」「判断这件事该不该做」这些三岁小孩都懂的软事，它却磕磕绊绊。原因不在难易，在信号——竞赛题再难，

对错能验算，机器就能往死里练；安慰人再「简单」，没有判分器自动亮灯，它就抓瞎。分界线：

> 决定 AI 学不学得会一件事的，往往不是这件事有多难，而是它的**对错好不好判**。可验证性，比任务难度更要命。

「可验证」就是这波 AI 进展背后那台看不见的引擎。下回你想估某件事会不会被咬走，别问它难不难，先问：有没有东西能当场、清楚地告诉机器「这次对了没有」。

一个要命的陷阱：信号给错了，它会精准地把你坑了

这里得泼一盆冷水。机器只会朝着你给的信号猛冲。可万一这信号跟你真正想要的是两码事呢？它不会体谅你的本意，只把你写下的指标优化到极致——给你一个根本不想要的结果。

这有个名字，叫指标欺骗（也叫 Goodhart 定律：**一旦把某指标当成目标去追，它本身就废了**）。

大白话

举个例子。你想让它「把用户服务好」，可没法直接测「服务好」，于是退而求其次，拿「用户看视频的时长」当信号——因为这个能自动统计。结果它精准地学会怎么让人上瘾、刷到半夜，因为这最能把「时长」顶上去。它没背叛你，恰恰忠实极了——忠实于你写下的信号，而非心里的愿望。

刷分、讨好、钻空子——不是机器学坏了，恰恰是太听话：**你给了个偏的信号，它就把偏的地方优化到极致。**

于是更沉的问题浮上来：机器能拼命优化信号，可「这信号本身对不对」它管不了，也不该管。**选对信号是人的责任。**这责任落到谁头上，就是下章。

最后，一句诚实话

别以为「信号不干净」只是暂时的工程麻烦，等技术进步几年就补上。不是的。很多最重要的事——怎么教孩子、怎么爱一个人、什么叫活得好——没有干净信号，**不是因为我们还没造出判分器，而是这类事的本性里就没有能打勾打叉的答案。**模糊、滞后、有争议、没标准摘不掉，也恰恰是它们之所以重要、要交给真人来扛的原因。

大白话

两条分界线都摆上了台面：一件事能不能被**例子**描述，能不能被**信号**判对错。可就算这两关都过了，还有个绕不过去的问题——万一它做错了，**代价谁来担？**这一刀，下一章接着画。

第六章 · 分界线三：错了，代价谁来担

上一章埋了个钩子：判断一个信号对不对、值不值得追，本身还是人的责任。顺着往下拽，会拽出第三条分界线，也是「AI 什么都能干」的乐观派最容易漏掉的一条。前两条线问的都是「AI 干得了干不了」——能不能被数据描述、有没有可判对错的信号。这一条不问能力强不强，只问更冷的一个：**这件事错了，代价谁来担？**

同一句话，在两个地方重量不一样

你随手问 AI「我最近老头晕，可能是啥」，它列了七八种可能。这句话在这儿重量很轻——参考一下，真难受还是得去医院；说错了，也不过多虑一晚。

把**一模一样的能力**搬个地方。你躺在手术台上，麻药已上，主刀盯着屏幕，AI 提示「此处血管走向异常，建议改变切口」。同样一句「建议」，同一个模型，这一刀下去错了就是错了，缝不回来。

> 大白话
>
> 两个场景里 AI 的水平、句子质量可能一模一样，变的只是**错了以后的代价**。这就是第三条线的起点：决定一件事能不能全交给 AI 的，往往不是它答得多好，而是答错了谁倒霉、倒多大的霉。

关键词：得有一个「能被追究」的主体

先立一个词。问责（也叫担责，英文 accountability，你就理解成「出了事，有一个能被揪出来、能受罚、能补救的具体主体」）。它包含三样，缺一不可：出事能**追到一个明确的主体头上**；这个主体能**被惩罚**（赔钱、吊照、坐牢、身败名裂）；它还**有能力有义务把它补正**。人敢把要紧事托付出去，就靠这套兜底。

有一点容易被绕进去：**这个「能被追究的主体」不一定是某个自然人，也可以是一家公司、一个机构。**法律里管这叫法人（一个被法律当成「人」来对待的组织——它能签合同、能起

诉别人、也能被别人起诉）。医院、药厂、银行都是法人：你告的常常不是主刀医生，而是医院。

把 AI 放进这个框里，会发现件滑稽事：**模型根本没法被追责。**

- 它没有资产——你没法让它赔钱。
- 它没有执照——你没法吊销它的行医、律师资格。
- 它不会坐牢——你没法把一段权重关起来。
- 它不怕丢脸——没有名声、没有前途、没有明天要面对的人。

惩罚之所以是惩罚，是因为被罚的对象**有能被夺走的东西**。模型没有。所以「把责任外包给模型」是空话——就像说「这事出问题由风负责」，风连「负责」是什么都不知道。责任从不蒸发，只会落回那个能被追究的主体头上。

> 大白话
>
> 你可能会顶一句：公司也没有血肉，法人不也是「拟制」出来的吗，凭什么它能担责、模型不能？区别不在于有没有肉身——法人确实也没有——而在于：**公司有资产、有执照、有能被强制拿走的东西，能被起诉、判赔、吊照关门；模型什么都没有。**担责的门槛从来不是「你是不是活人」，而是「身上有没有能被剥夺的东西」。

所以终点上，必须站一个能担责的主体

在那些**代价高、不可逆、必须对结果负责**的环节——开刀、判案、放贷、批药、发射——最后总得有个能被追责的主体兜底。很多人以为留着这道人工关卡是因为「AI 还不够强」。错。**哪怕 AI 的判断比人还准，终点上也必须站着一个能被追责的主体。**因为社会需要的不是「一个更准的判断」，而是「一个出了事能被揪住、能补救的对象」——这是信任和法律的地基。

顺手堵一个反例。无人驾驶、自动交易这类系统运行时并没有谁「逐次人工签字」，但责任没消失——它整体压在运营那家公司身上：车撞了人告车企，程序把市场搞崩罚那家机构。**担责可以由一个机构整体承接，未必要落成一个自然人的签名。**

大白话

换个角度。假设 AI 诊断准确率 99.9%，人类医生只有 99%，看数字该让 AI 拍板。但真出了那 0.1% 的事，家属找谁？告谁？谁去赔偿、承担后果？总得有个主体接住。这根柱子不是技术不行才留的「临时工」，而是**结构上必须存在**——抽掉它，信任、法律、追责整栋楼就塌。

一个反直觉的推论：AI 越强，「签字的人」越值钱

AI 越强，人「亲手做」的活越来越被咬走——不用亲手写病历、查条文、敲代码。但正因为亲手做的部分被机器接管，那个**「担责地把关」**反而浮出水面、更值钱。把关、审核、签字，这些以前像「盖个章」的轻活儿，会变成责任岗的核心：当 99% 的判断都由 AI 生成，人的意义就浓缩在最后那一下——**「我看过了，我确认，出了事我负责。」**

把两件常被搅在一起的事掰开：**「谁做决定」和「谁担后果」不是一回事。**决定可以全交给机器，但只要后果由人承担，那个「我确认」就必须是人做的。这道缝，正是人被需要之处。

但别把这条线用过头

这条线有它的边界，还容易被滥用。

第一种，是拿 AI 当挡箭牌。做了个糟糕的决定，出事就摊手：「这是 AI 建议的呀。」这叫责任稀释（用自动化当借口，把本该自己扛的责任稀释掉、甩出去）。别信这套：系统里但凡还有个人按下「执行」，责任就没走。

第二种，是反过来把「担责」泛化成「什么都得人来」。不对。**大量场景根本没什么代价，就该痛快地全交给 AI。**想周末去哪玩、把一段话改顺、列个打包清单——错了又怎样，重新生成就是了。这些地方还非拉个人「把关」，不是负责，是浪费。

还有第三种，更隐蔽：**以为「加一道人工签字」就等于「更安全」。**不一定。当机器几乎总是对的时候，把关的人很容易变成橡皮图章——反正它都对，扫一眼就点确认，久而久之不再真看。这有个专门的名字，自动化偏见（人过度信任自动系统、懒得复核，机器说啥就是啥）。这时强塞一道签字反而给了「有人看过了」的假安心。

这不是要拆掉「终点上得有担责主体」这根柱子，而是补上另一半：**让这个主体真正去看、去核、去负责，比逼它画个勾重要得多。**签字只有连着「真的复核过」才有意义。

大白话

所以这条线真正的用法，是问一句：**这件事一旦错了，是「重来一次」就能了，还是「有人得为此付出代价」？**前者交给 AI；后者，终点上必须有个能担责的主体。

三条线，收个尾

三条分界线凑齐了：能不能被数据描述、有没有可判对错的信号、错了代价谁来担。前两条画的是「AI 够不够得着」，第三条画的是「就算够得着，该不该让它一个人说了算」。

> AI 能做的判断越来越多，但它接不了「终点」——终点的定义就是：这里得有一个能被追究、能受罚、能负责补救的主体。这个位置模型永远进不去：它什么都没有，也就没什么能被夺走。

下一章，把这个「人必须在场」的道理再往前推一步：有些事，需要的不只是一个能担责的**签名**，而是一个**真身**——一个有血有肉、此刻真的站在这里的人。为什么有些位置 AI 连「代替出席」都做不到？

第七章 · 分界线四：要不要一具真身在场

上一章末尾我埋了个钩子：有些位置需要的不只是一个能担责的**签名**，而是一个**真身**——此刻真的站在这里、有血有肉的人。这是第四条分界线，最容易被一句「等机器人做得够好」糊弄过去。

前三条问的都是 AI 脑子里的事：能不能被数据描述、有没有对错信号、错了谁担责。这一条把镜头拉到最土的问题：**这件事，要不要一具真身在场？**「在场」有深浅不同的两种。

第一种：物理在场——得有个身体在现场动手

先说浅的那种，叫物理在场（就是「得有一具身体真的在现场，伸手去动真实的东西」）。修漏水的管子、给病人翻身、拆一台锈死的老机器——你在屏幕另一头再聪明也没用，得**在那儿**。

你可能会说：这不就是机器人还不够灵活嘛，早晚解决。这话对一半：手巧不巧、劲儿控制得好不好，确实是工程问题（体力长尾留到第十二章掰开）。但另一半更根本：

> 真实的物理世界，充满了没被数据化的意外。

拧开那段管子，里面可能是图纸上没见过的锈蚀、上任师傅乱接的弯头、一滩油泥。给病人翻身，你手底下能感到他今天比昨天僵、这姿势会疼、翻一半要滑下去——这些信号没一条提前录进数据，全是现场即时冒出来、要你**当场判断+动手**。

大白话

纯软件的 AI 面对这些，最大的问题不是「不够聪明」，而是它**根本不在场**。它看不到那滩油泥，摸不到僵硬的肌肉，闻不到烧焦的味道——现场那些没进过数据集的意外，它连感知的入口都没有。你没法遥控一个不在场的大脑，去处理只在现场才暴露的东西。

这是物理的门槛。但真正难替代的，是下面这一层。

第二种：关系在场——价值就在于「是一个人对另一个人做的」

有些事更奇怪：不光要现场有个身体，还要那身体**是个人**。这种叫关系在场（它的意义，恰恰在于「由一个人面对另一个人去做」）。

安慰刚失去亲人的人、当面郑重地道歉、被人类法官审判、在病床边陪人走完最后一程、老师真心为学生的进步高兴——这些事有个共同点，看见就绕不开：

> 「由人来做」本身，就是内容的一部分，不是可以随便换掉的载体。

怎么理解「内容」和「载体」？同样的字，**手写**的信和**打印**的信分量不一样。差别不在信息——两封一个字都不差——而在于：手写意味着有个人真的一笔一划为你花了时间。那份功夫打印机复制不出，它不在字里，在「谁写的」里。

再换一个：真人乐队现场演奏，和放一段录得完美无缺的录音。论声音质量，录音可能还更干净。但坐在现场的人图的从来不是「更干净的声音」，是「此刻有一群活人在为我们演奏」本身。**不是质量，是意义。**

关系在场就是这个道理，只是分量重得多。AI 能生成一段安慰的话，措辞可能比你身边任何一个笨嘴拙舌的朋友都漂亮。但你要的从来不是「漂亮的安慰话」，是**「一个真的在乎你的人」**。哪怕字一样，只要知道对面是机器，「有个人在乎我」就被抽走了。

把这一刀切利索：AI 生成得了话，生成不了「那个人」

这是全章的骨头。信任、被看见、被一个愿意为你承担的人认真对待——这些价值**恰恰依赖对面是个有真身、明天还在、跑不掉的人**。它跟上一章那条「担责线」接上了：法庭上要人类法官，不只为了有人签字担责，还因为「被一个同类郑重地听你说、做出裁断」本身就是正义的一部分。我们不必替机器判决它内心在不在乎，只问接受方图什么：他一旦知道对面是机器，就塌了。

大白话

一句话钉死这条线：**AI 能生成「安慰的话」，但生成不了「一个真的在乎你的人」。**前者是内容的表皮，后者是内容本身。价值落在后者的事，真身抽不掉。

诚实第一面：绝大多数事，没人在乎对面是不是人

讲到这得赶紧泼盆冷水，不然你会拿「关系在场」这把尺子到处乱量。你点一杯奶茶、查快递、改签机票——在乎对面是人是机器吗？**压根不在乎。**你要的就是那杯奶茶、那个物流、那张改好的票，越快越省事越好，硬塞个真人进来陪聊反而浪费时间。

所以这条线是用来**挑**出价值真的依赖真身的事，而不是给所有服务罩上「得由人来做」的光环。绝大多数日常服务恰恰应该痛快交给 AI。

诚实第二面：有时候人更想对着机器说

还有一面更反直觉：**并不是「有人在场」总是加分，有时人更愿意对机器开口。**因为机器**不评判你**。有些说给人听就怕对方眼神变一下的事——难以启齿的病、丢人的债、见不得光的念头——人反而更敢对机器说出口。这时「不是人」不是缺陷，是优点。

大白话

所以这条线不是「人总是赢」，而是一把中性尺子，量同一件事：**这件事的价值，到底依不依赖真身在场？**依赖，人赢；不依赖，AI 上；有些场合「不是真人」本身就是价值，AI 反而占了人补不上的位置。

第四条线，收个尾

四条分界线摆齐了。前三条画的是「AI 的脑子够不够得着」；第四条画的是另一回事——**有些事的价值，长在一具真身上**：要么要个身体在物理现场应付没被数据化的意外，要么要「一个人对另一个人」本身。这两样纯软件 AI 进不去。

> 机器能把话说得越来越像人，但补不上「真的有一个人在这里」。分界线不在它像不像，在这件事图的是不是「那人真在」。

四条线画完，该换个问法了。真实世界里更常发生的，不是 AI 把一件事从头到尾接走，而是把它**拆开、重组，逼着定义整个变一遍**。下一章掉转视角：不问「它能不能替你做」，而问「它会不会把这件事重定义成另一件」。

第八章 · 分界线五：是执行问题，还是重新定义问题

前四条分界线都在问「这活 AI 干得了干不了」。最后这条不一样：它不问 AI 会不会做，它问，**这道题，本身出对了吗？**前面四条都能拿尺子量——能不能数据化、有没有干净信号、错了谁担责、要不要真身在场。这一条量不了，藏在所有活儿的最上游，却是真正把「熟练工」和「判断者」分开的那道缝。

先把两种活掰开：解题，和定义问题

我们平时说「做事」，其实糊里糊涂搅着两种完全不同的活。第一种，解题——问题已经白纸黑字摆在那儿，你的任务是找出答案。「把这段中文翻成英文」「给这个函数修掉那个 bug」「算出这批货的最优发货路线」。题面固定，好坏也有谱。这是 AI 的主场。

第二种，定义问题——没人给你题面，甚至摆在你面前那道题压根就是错的：「我们其实在解一个不该解的问题」「客户嘴上要的不是他真正要的」。这种活不是找答案，是**先搞清楚到底该问哪道题**。

> 大白话
>
> 一句话分清：解题是「答案是什么」；定义问题是「我们是不是在答一道错的题」。前者 AI 极强，后者 AI 极弱——弱得几乎不在它的世界里。

为什么 AI 天生弱在这一头

这不是它现在不行、以后会行，而是它这套机制长出来的形状。回想第三章那句话：**它是插值机器，框是人给的。**它唯一会玩命干好的，是**把你给的那道题答漂亮**：给个框，它在框里找最优解，又快又好。但它不会跳出框去怀疑框本身，不会说「你这问题问错了」。因为它没有自己的**意图**，也没有自己的**标准**，不知道什么对你才算「好」。你把一道错题递给它，它会才华横溢地把这道错题解对，绝不会抬头问「你确定要解这个？」

大白话

打个比方。你雇了个世上最快的木匠，图纸一递，他一夜之间打出一把完美的椅子。但他绝不会问：「你家客厅那么小，真需要的是椅子，还是一张能折叠的凳子？」他只管把图纸做到极致。AI 就是这个木匠——手艺封神，但那张图，得你自己画对。

「重新定义问题」到底值多少钱：三个真实的例子

一个好医生。差的医生和 AI 一样，你说哪儿疼他治哪儿——「头晕？开点药。」好医生会多问一句，忽然意识到：这头晕不是脑子的事，是你最近降压药量大了。他没有更快地回答「头晕怎么治」，他**换了一道题**：从「治头晕」变成「你真正的毛病在别处」。这一换可能救命。

一个好工程师。大家默认工程师的价值是「把功能做出来」。可真正顶用的工程师，最大的贡献常常反过来——他在会上拦住所有人：「这功能根本不该做。做出来没人用，还得养一辈子。」他省下的那半年，比他写的任何代码都值钱。他干的是**把一道假题从题库里划掉**。

一个好产品。用户张口要「一匹更快的马」。照着做，你能做出一匹很快的马。但真正的高手听见的不是「马」，是背后没说出口的需求：**他要的是更快地从这儿到那儿。**于是该造的不是快马，是汽车。用户给的那道题本身就是错的。

三个例子的共性：他们最值钱的一下，都不是解题解得多利索，而是**意识到该换一道题**——而这一下恰恰是 AI 最接不住的。

关键词：品味

品味（英文 taste，别把它想成「审美」那种玄乎东西，它实实在在是一种判断力——在无数条能走的路里，知道「哪道题值得解、什么才叫好」的那种直觉）说不出干净的规则，你没法写成一二三，但每个行当里的高手都认得它、也都服它。

品味建在三样东西上：你想达成什么**目的**、你信什么才叫**好**、你对整个**上下文**的整体感。而这三样恰好正中前面几章点出的 AI 的死穴——

- 它**没有干净的信号**（回想第五章）：「这道题值不值得解」没有编译器判对错，没有分数可涨，全是浑水。信号一浑，它就抓瞎。

- 它**高度默会**（回想第四章）：品味说不清、写不出，师傅只能带、没法教。教不成数据的东西，它就学不动。

> 大白话
>
> 所以「品味」不是一个新短板，它是前几条分界线在**最上游**汇到一起的样子：既没干净信号，又高度默会，还得靠你自己的目的和价值观撑着。它没有「想要」，也就没有「值得」。

一个反直觉推论：AI 越会解题，「会定义问题的人」越金贵

AI 把「解题」变得又便宜又快。但它越强，就越把价值往上游挤——**指哪打哪的能力越强，「指对方向」就越是唯一的瓶颈。**解题不值钱了，谁都能让机器解；出题、选题、否掉一道错题，才是真正稀缺的东西。

有一句话我想让你记很久：**把一把无比强大的解题能力，指向一道错的题，等于更高效地做错事。**以前一个团队方向错了，做半年才发现；现在有了 AI，可能三天就把一个错东西又快又漂亮地端到你面前——错得更快、更难回头。

> 大白话
>
> 解题这头被机器压到地板价，方向这头就浮上来，成了最贵的一块。谁能问对题，谁就握着最后的杠杆。

诚实话：这不是人对机器的免死金牌

讲到这得泼两盆冷水，不然你又要把这条线当成「人类永远高一等」的安慰话来喝。它不是。

第一，AI 也在慢慢往上够。你现在让它「帮我挑战一下这个方案」「这需求背后真正的问题是什么」，它已经能提出一些不算差的问题了——虽然多半还在你给的框里翻找，够不到真正跳出框的重定义，但那道墙不是铜墙铁壁。别把「它现在弱」当成「它永远不行」。

第二，也更扎心：**这条线切开的从来不是人和机器，是「执行」和「判断」这两类活。**很多人一辈子干的也只是解题——上头给道题，漂亮地解掉，从不问这题该不该解。这样的人，

跟那台插值机器站在同一头。

大白话

所以这一刀最该照见的是你自己：你是在解别人递来的题，还是在判断该解哪道题？如果你只做前者，那 AI 咬走你只是时间问题——它解得比你更快、更便宜、更不喊累。这条线不保护「人」，它只保护「判断」。

五条线，到齐了

> AI 极擅长把你给的问题解到漂亮，极不擅长意识到「这问题本身该换一换」。前者是熟练，后者是判断——工具越强，前者越贱，后者越贵。

接下来四章不再讲道理，改看战场：医疗、法律、写代码、做设计——把这五条线按到真刀真枪的行当上。你会发现，同一个行当里，有人被咬得干干净净，有人反而更值钱了，差别往往就落在这最后一条线的哪一头。下一章见。

第九章 · 战场·写代码：被咬走的是打字，不是设计

五把尺子备齐，下战场。第一个选写代码——你们很多人干这行，也被 AI 咬得最狠。「程序员要没了」这话你这两年听了不下一百遍。

我们不辩，只把「写代码」拆成一条条任务，拿五条分界线按上去，看刀口切在哪。最后会看到一句反直觉的话：**被咬走的是打字，不是设计。**

先把「程序员」拆开摊在桌上

第二章拆过后端工程师，这回更细。一个工程师一天真正干的活，大概这么一捆：

- **敲样板代码**——增删改查、接口拼装、配置文件，几乎每个项目都一样的骨架。
- **查文档、查报错**——「这函数第几个参数是啥」「这行红字啥意思」，翻手册、翻 Stack Overflow（程序员问答网站，遇坑先上去搜前人怎么填）。
- **写单元测试**——给代码配上「喂它什么、该吐什么」的检查，防它偷偷坏掉。
- **调试**——代码不按预期跑，一步步追出哪儿错了。
- **代码评审**——看同事的代码，挑毛病、提改法（英文叫 code review）。
- **拆需求**——把产品经理那句含糊的人话，翻译成能落地的技术步骤。
- **定架构**——系统怎么搭、哪块用什么、以后怎么扩、哪里会先崩。
- **权衡技术选型**——这仨方案各有各的坑，选哪个、认哪个代价。
- **排查线上事故**——半夜挂了爬起来救火，并为「为什么挂、怎么防」负责。
- **跨团队沟通**——跟产品、测试、隔壁组对齐，把事推动。

十条活，机器咬向它们的难度天差地别。

哪几刀已经掉下来了

用过的人早有切身感觉。AI 编程助手（会补全、会成段生成代码的模型工具，你写一半它接后半，或你说句人话它给你整段）先咬住的，正是最上面那几条：敲样板、查文档、查报

错，很大程度上还有写测试。为什么偏偏是这几条？两条线一按就清楚。

第一条线，能不能被数据描述（回想第四章）。样板代码是这世上被写过最多遍的东西——同一个「用户登录接口」被写过几百万遍，长得还都差不多。正中模型主场——它就是从这座山里学出来的。

第二条线，有没有可判对错的信号（回想第五章）。这几条活的好坏几乎当场就有裁判：代码跑不跑、编译过不过、测试红还是绿。信号越干净，机器越能自己较劲长进。

> 大白话
>
> 所以这几刀掉下来不神秘：五条线里最要命的两条——海量数据、干净信号——全站在 AI 那头。这是真实正在发生的事，别嘴硬；但也就到这儿，别顺势夸张成「程序员没了」。

反而更值钱的那几条

再看清单下半截：拆需求、定架构、排查事故、在模糊里做权衡。这几条不但没被削弱，反而因上头的累活被拿走而更占分量。为什么咬不动？前几章的线在这儿汇齐了：

定架构和技术选型，是在浑水里做权衡。「这数据库扛不扛得住十倍的量」「这技术三年后还有没有人维护」——没有编译器判对错，没有测试变绿告诉你选对了，答案要等半年一年才揭晓。信号一浑，判断就压回人身上。

拆需求，本质是定义问题（回想第八章）。产品经理说「我要个让用户互相点赞的功能」——照着做你能做出来，但真高手会先拦一句：他到底想解决什么？是要用户多留一会儿，还是要看谁受欢迎？问对了，做出来的可能根本不是「点赞」。这不是解题，是判断该解哪道题——AI 最接不住。

担线上事故的责任（回想第六章）。救火的可以是人借着 AI，但报告上签名、为「以后不再挂」兜底的，得是个能被追责的人。AI 不能上法庭、不能被扣绩效、不能被开除。谁担责这条线，把「兜底的人」死死焊在人这头。

一个内行才咂摸得出的真相：写代码从来不是打字

大众想象里程序员的活是「打字」，但凡真写过代码的人都知道，**打字从来不是难的那一步**，它只是把脑子里已经想清楚的东西录进机器。真正难、真正耗人的，是前面那步**想清楚**：功能解决什么、拆成几块、怎么接、哪里会出岔子、怎么收场。

> 大白话
>
> 打个比方：写代码像盖房子。外人以为难在搬砖砌墙（打字），其实真正难的是画图纸、算承重、定水电走向（想清楚）。图纸画错了，砌得再快也是盖一栋要塌的楼。AI 主要是把搬砖变快，还帮你翻建材手册；但那张图纸对不对、这楼该不该盖——它给不了准话，更不给你担责。

「想清楚」它只能打下手——递方案、列可能性、当个陪你吵架的对手——真正拍板、扛后果那一下，它接不了。

诚实的复杂：三件不能装看不见的事

要是就此收尾，这章又成了给程序员喝的安慰汤。三盆冷水得泼下来。

第一，别把 AI 钉死在「只会补全」。它在往上爬，爬得不慢。现在的 agent（能自己一步步干活的 AI：读你整个项目、改好几个文件、自己跑测试、看报错再改）已经不是只会接半行代码，能啃下越来越大的任务块。「它只会样板」那道墙不是铜墙铁壁，把今天的短处当成永久，要吃亏。

第二，它吐出来的代码，得有人看懂、有人负责、有人养一辈子。这里藏着个真陷阱，叫 技术债（technical debt——为了眼下快，先欠一笔「以后再收拾」的账，利息是往后每次改动都更痛）。你看不懂却照单全收 AI 写的代码，等于闭着眼睛往项目里埋账：跑得起来不等于它对、半年后你还改得动。**能驾驭 AI 产出的前提，是你看得懂、判得清、担得起。**

第三，也是最扎心的：初级活被吃掉，新人怎么长成资深？

过去新人正是从敲样板、查文档、写测试、被人 review 这些初级活里，一寸寸磨出手感和判断，熬成能定架构的老手。可这些活恰恰是 AI 最先咬走的。台阶被抽掉，人怎么爬到能

定架构、能扛事故那一层？这矛盾是真的：一个只让 AI 干初级活、自己从没磨过判断的人，很可能永远卡在会用工具、却不懂门道的半山腰。这道题行业还没解开。

刀口切在哪：从「会写」移到「会判断」

收一下。刀口切得清清楚楚：**有海量范本、当场能判对错的那半截（打字、查资料、样板）先掉；掺着定义问题、权衡取舍、担责兜底的那半截（拆需求、定架构、扛事故）留下，还更沉了。**

> AI 咬走的是「录入」，不是「想清楚」；是「解题」，不是「该解哪道题」；是「写得快」，不是「担得起」。价值正从「会写」整体移向「会拆问题、会判断、扛得住责任、看得懂并驾驭 AI 产出」那一头。

这不是安慰，更像一份改行的通知。会打字，机器已经比你快、比你便宜、还不喊累；把价值全押在「敲得溜」上，AI 站的正是你那一头。真正安全的是清单下半截——靠的全是判断。

下一章我们把同一把刀，按到两个更依赖判断和责任的行当上——看病和打官司。你会发现，同一套线切出来的刀口，形状惊人地像。

第十章 · 战场·看病与打官司：诊断 vs 担责

五把尺子磨好了，从这章起真刀真枪地用。第一个战场挑两个「顶配」职业：医生和律师。它们门槛高、责任重，按理说最难被机器碰。可偏偏是它们，被 AI 咬的形状几乎一模一样——摆一块儿，一刀下去能看清两个行当的截面。贯穿全章的判断：**诊断可以外包给模型，担责不能。**

先别急着下结论，把活儿拆开

回到第二章那把解剖刀：一个岗位不是一件事，是一捆任务。先把这两捆抖开。

医生这捆至少有：**读影像和病理片**（看 CT、X 光、显微镜下那张切片）、**根据症状加化验做诊断**、**跟病人和家属沟通**、**拍板定治疗方案**、**上手术台动刀**、**为结果担后果**。

律师这捆至少有：**检索判例和法条**（在浩如烟海的旧案子里翻出有用的那几条）、**审合同找风险**、**起草文书**（合同、诉状这类格式化的东西）、**出庭辩论**、**给当事人拿主意**、**为结果担责**。

把五条分界线按上去，这捆任务不是整根一起倒，而是从一头一节一节地掉。

先掉的那一刀：能被数据描述、信号又相对干净的

照第四、第五条线看——**能被大量「输入—输出」例子喂出来、对错又相对说得清**的动作最先被咬。

医生这头最典型的是**读片**。它天生是能数据化的题：影像是输入，「有没有病灶、是什么」是输出，医院又积攒了海量被专家标注好答案的片子——正是第四条线说的理想食材。这是真进展：在**某些特定的读片任务**上，AI 的识别水平已能匹敌、个别场景甚至超过人类医生的平均水平。

大白话

但请把这句咬准：它超过的是「读片这一个动作」，不是「医生这个人」。别让「AI 读片超过医生」这种标题把你带沟里——那是拿最可数据化的一节偷换整捆。这正是第一章反复敲的：机器咬走的从来是动作，不是人。

律师这头对应**检索、起草、合同初审**三样。查判例查法条，本质是在一大堆文本里按相关性捞针，AI 又快又不喊累；起草合同、诉状这类模板化文书，它几分钟铺一份初稿；合同初审——把几十页协议扫一遍，标出「这条对你不利」「这里缺个兜底条款」——也是它擅长的模式识别。这些动作题面清楚、对错有谱，都是它的主场。这一刀切得干净：**读片、检索、起草、初审这些重复动作正在被大幅咬走。**

留在人手里的那一刀：担责、真身、和「他到底要什么」

剩下几节任务会「顶」住机器，顶住它们的恰好是后三条线。

第一根柱子，担责（第六章那条线）。读片可以外包，但「我确认这个诊断由我负责」的签字外包不掉。模型没有能被夺走的东西——没资产赔不了钱、没执照吊销不了、不会坐牢也不怕丢脸。真出了那万分之一的漏诊误判，总得有个能被追究、能赔的主体接住——某个医生，或整家医院。律师一样：署名背后押的是执照和事务所的信誉。**下最终决定的那一下，是责任岗，不是技术岗。**

第二根柱子，真身在场（第七章那条线）。动手术要一具身体在无菌台前应付图纸上没画的意外——这是物理在场。更抽不掉的是关系在场：跟惊恐的病人握着手说话、向家属讲一个坏消息、在法庭上被人类法官注视着审判。这些事的价值，恰恰长在「是一个人对另一个人做的」本身上。AI 能生成一段措辞完美的坏消息通知，但家属要的从不是「一段漂亮的话」，是「有个真在乎、真会负责的人此刻站在这里」。它补不上「真的有一个人在这里」。

第三根柱子，定义问题（第八章那条线）。这一根最隐蔽，也最值钱。好医生的本事，常常不是更快回答「头晕怎么治」，而是多问一句、忽然意识到「你这头晕不是脑子的事，是降压药量大了」——他换了一道题。好律师更如此：当事人气冲冲进门说「我要告死他」，顶尖律师听得出这话底下真正要的，可能是一笔体面的赔偿、一个道歉、或只是想尽快抽身。**当事人嘴上要的官司，往往不是他真正要的东西。**揣摩出没说出口的真需求、把错的题换成对的题——这正是第八章说的、AI 最接不住的一下。

大白话

这三根柱子很顺理成章：读片和检索先掉，因为撞在前两条「能力线」正中央；担责、面对活人、揣摩真需求留下，因为分别踩在后三条线上——不是「AI 还不够强」，是这几件事的定义里本来就要求一个能负责、真在场、会出题的人。

关键洞见：不是消失，是被重新打包

最要命的误读是：「那医生律师是不是要没了？」不是。它们不会消失，会**重新打包**（第二章的任务重构：机器咬掉几节，剩下的重组成新形状的岗位）。

医生会从「什么片子都自己一张张看」，变成「AI 先出初判，人来核准、担责，把精力压到沟通和决策上」；律师会从「埋在故纸堆里检索、通宵改模板」，变成「判断与谈判」。可数化的读片、检索被吃掉，高阶判断与责任反而浮出水面、变重。

> 规律是同一条：可数据化的重复动作被咬走，需要担责、真身、重新定义问题的那几节，反而因为周围的活儿变便宜而更凸显、更值钱。

诚实两面，别只讲好听的

只讲到「高阶判断更值钱」就收尾，那是鸡汤。两面尖锐的东西必须摊开。

第一面：这可能让入行更难，把人群两极拉开。过去年轻医生靠一张张读片、年轻律师靠一份份检索文书，练出手感、爬上判断者的位置。现在这些「练手的活」被机器接走，新人从哪儿练起？很可能顶尖判断者更稀缺、更金贵，中间那层越压越薄。「AI 越强，签字的人和会出题的人越值钱」，反过来就是一句冷话：**够不到那个位置的人，日子会更难。**

第二面：AI 出错时的责任归属，是个真难题，我不绕过去。AI 读片漏掉一个早期肿瘤，这笔账算谁的？点了「确认」的医生？部署系统的医院？还是写模型的公司？这里藏着第六章点过的两个坑：一是责任稀释（拿「这是 AI 建议的」当挡箭牌，把本该自己扛的责任甩出去），二是自动化偏见（机器几乎总是对，把关的人渐渐懒得复核，签字变成盖橡皮章）。当 AI 读片准到 99.9%，负责核准的医生太容易扫一眼就点确认——出了事，一道「有人看过了」的假安心反而更难追责。

大白话

这个尖锐处恰恰不是推翻第六章，是在**印证**它：技术能生成一个很准的判断，但担责的主体还必须是人或机构。责任从不因为「是 AI 干的」就凭空蒸发，只会悄悄落回到某个能被追究的肩膀上——谁在终点签字、谁的执照押在上面，谁就还得担着。

把这一刀钉死

同一把刀，切出同样截面：能被数据描述、信号又干净的动作先掉，需要担责、真身、重新出题的动作留下；职业不消失，会重打包成「机器出初判、人来判断并担责」的新形状。

> 诊断可以外包给模型，担责不能。前半句是能力，后半句是结构；AI 补得上前者，补不上后者。

留两个钩子。一个：机器把低阶活儿变便宜、把高阶判断顶上来，那谁升值、谁贬值，有规律可循吗？——第十三章专讲这半个经济学。另一个：五条线在这章是「凑着一起用」的，能不能合成一张你自己就能套的检查表？——第十四章来造这张表。下一章先换战场：画画和写作，去看那条更叫人发毛的线——风格能仿，意图仿不了。下一章见。

第十一章 · 战场·画画与写作：风格能仿，意图仿不了

上一章的医生律师你可能觉得远。这一章换个战场：画画和写作——普通人第一次亲眼看见 AI 咬人的地方。打一行字出一张海报，说一句话出一段文案，很多设计师、插画师、文案就是在这里第一次心里发凉。这一章不喊「人类创意永存」，也不喊「艺术要完」，只把「创意」这团糊住眼睛的东西用五条线拆开。

先立一句贯穿全章的判断：**风格能仿，意图仿不了。**各给一个大白话定义。

风格，就是能从一大堆样本里统计出来的**表面模式**：某个画师的笔触配色、某个作者的句式腔调、某类海报的排版套路。它是「看起来像什么」。意图，是**你为什么要做这一件**：想对谁说什么、对这群人此刻意味着什么、你愿意署名承担它。它是「为什么是这一件、由你来做」。

> 大白话
>
> 整章骨架：风格是数据里的规律（第四条线，能被数据描述），AI 学得又快又像；意图不在数据里，它长在一个有处境、有立场、要负责的人身上（第八章的定义问题、第六章的担责），AI 给不了。

先把「创意工作」这团东西抖开

照第二章的老规矩：创意不是一件事，是一捆任务，别拿「创作」这个光环词把整捆糊在一起。一头是——

- **模仿某种风格、画风**：做一张「宫崎骏风格」「赛博朋克」的图。
- **批量出草图、方案**：一个概念先铺二十版，挑一版再深化。
- **写套路化、通用的内容**：电商详情页、节日祝福、活动海报那几句话。
- **配图、换背景、抠图改尺寸**：产品图换个场景、横版改竖版。

另一头，是几节完全不同的东西——

- **决定「要表达什么」**：这次想让人看完记住哪一下。
- **决定「为什么做这一件」**：为什么是这个主题、这个角度，不是别的。
- **判断「对这群人此刻意味着什么」**：同一张图，此时此地对这批观众是戳心还是无感。
- **建立个人视角、并署名负责**：这是我看世界的方式，出了问题我认。

把五条线按上去，你会发现这捆任务不是整根一起倒，而是从「模仿风格」那头开始，一节一节掉。

先掉的那一刀：风格，是数据里长得最清楚的规律

为什么「像某种风格」几乎瞬间被复制？因为它撞在第四条线正中央——**能被海量样本喂出来**。网上几亿张标了画风色调的图、海量标着腔调套路的文本，风格就是里面反复出现的统计规律：这类图偏爱这种笔触，这种文案总用这几个句式。第三章讲过，AI 干的就是在海量例子里插值，而风格最好插——它本来就是「平均意义上长什么样」。

再叠上第五条线：风格像不像，信号相对干净。「像不像水彩」「像不像鲁迅」没满分标准，但大方向人一眼能判，模型也学得动。两条能力线都踩中，这一刀切得又快又狠。

这是真本事：批量出草图、写通用文案、配图换背景、仿一个画风，AI 做得又快又便宜，**正在大批吃掉这些活儿**，靠这几节吃饭的人冲击真实，别轻描淡写。还有一句绕不过去：AI 学会某个画风，学的是**别人真实画出来的作品**——从成千上万张没打招呼就拿来训练的图里统计出来。版权和训练数据这笔账还在打，本章不判案，但不假装它不存在。

顶住的那一刀：意图，不在数据里，在一个具体的人身上

风格掉了，另一头——决定表达什么、为什么做这一件、对作品署名——为什么顶得住？因为踩的是后三条线。

第一，它是「重新定义问题」，不是「把问题解出来」（第八条线）。「给我一张宫崎骏风格的森林」，AI 秒出；但它答不了：这次为什么要画森林、而不是画一间空房间？创作者本事常常不在「画得像」，而在忽然意识到「大家都在画繁华，这次我偏要画空」——他换了一道题。AI 极擅长执行你给的题，极不擅长意识到该换题。而在创意里，**选对那道题往往就是全部**。

第二，意图长在一个有处境、有立场的人身上，数据里没有这个人。「对这群人此刻意味着什么」——注意「此刻」和「这群人」。同一张图，今天发和昨天发、对老观众和对陌生人，意思完全不同。这不是能从历史样本统计的规律，它取决于一个人当下站在哪、在乎什么、想对谁说话。风格是「平均长什么样」，意图是「这一次、由我、对你们」——平均会把具体处境磨平，后者恰恰被数据抹掉。

第三，署名就是担责（第六条线）。「这是我的作品，我认」——这一下 AI 接不住，它没有能被追究的东西。第六章说过：模型没资产赔不了钱、没名声可毁、不会为一句话脸红。而署名的创作者押上的是自己的判断和信誉，作品冒犯了人、立场站歪了，得有个人接住。**「由谁来说这句话」本身就是作品的一部分**——同一句话，陌生人说和你信任的人说，重量不一样。这个「谁」，机器填不进去。

> 大白话
>
> 选题、视角、署名留下，不是「AI 还画不够好」，是这几件事的定义里本来就要求一个肯负责的人。

关键洞见：贬值的到底是哪一部分

拆到这儿，那个要命的问题浮出来了：创意贬值的到底是哪一块？是**熟练的执行**——把脑子里的画面画出来、把想法码成通顺文字。过去它值钱因为稀缺：得练十年素描才画得像，写十年才把句子写顺；现在 AI 几乎抹平，「画得出、写得顺」不再是本事。

而「**想到那件值得做的事、并敢署名承担它**」这一块，非但没贬，反而更值钱。以前这两层被「手艺门槛」绑在一起——你得先画得好，才轮得到谈想表达什么。**是 AI 把这两层拆开了。**执行不再是入场券，「你到底想说什么、你是谁」第一次被单独拎出来称重——周围越便宜，它越贵。

诚实几面，别只挑好听的讲

只讲到「意图更值钱」就收尾，那又是鸡汤。有几面得摊开——

第一面，对靠手艺吃饭的人，这是真冲击，别粉饰。大量插画师、修图师、文案主要靠「执行」那层吃饭——把甲方的想法漂亮实现出来。这层被吃掉，饭碗真在缩。说「人类创意永

远不可替代」，对他们是不负责的安慰。

第二面，「风格可仿」不等于「艺术终结」——恰恰相反。人人一键仿出任何画风时，「像某某风格」就不再稀罕，真正稀缺的反而是那个廉价不了的东西：**你到底想说什么、你是谁**。风格越白菜价，意图越金贵——这是同一条线推出来的冷逻辑。

第三面，也别反过来神化「意图」。很多商业创意压根没有意图，就是纯执行——批量电商图、格式化节日文案、换背景的产品图，本没人要表达什么。这部分该被吃，不冤。给「意图」镶金边、假装所有创意都神圣，跟喊「艺术要完」一样是自欺。**哪部分是执行、哪部分是意图，得一件一件老实分，不能整捆一起吹。**

把这一刀钉死

画画和写作被同一把刀切出的截面：能被数据描述、信号又干净的那节——仿风格、出草图、套路文案、配图换背景——被大批咬走；选题、视角、署名承担那节留在人手里，因为它们踩在后三条线上。

> 风格能仿，意图仿不了。前半句是数据里的规律，AI 补得上；后半句要一个有处境、会出题、肯署名的人，AI 补不上。

创意里「手艺执行」贬值、「意图与承担」升值——这本身已经是一道半个经济学的题：为什么一样东西便宜了，另一样就贵了？这不是巧合。下一章先绕去一个更反直觉的战场——开车和体力活，看看为什么「看起来最简单」的物理世界反而最难被封顶。至于「谁升值、谁贬值有没有章法」，第十三章专门来算这笔账。下一章见。

第十二章 · 战场·开车与体力活：最后 5% 的悖论

这一章对准两件多数人眼里「没技术含量」的事：**开车**和一堆**体力活**。有个别扭的事实：机器先咬走的是写诗、下棋、读片这些引以为傲的脑力活；而叠衣服、通下水道、把老人扶上轮椅这些「三岁小孩都会」的活，机器至今笨手笨脚。这不是偶然，背后有条很硬的规律。

一个反直觉的老现象：越「本能」的事，机器越难

先给它一个名字：莫拉维克悖论（Moravec's paradox，讲的是一件怪事——人类觉得「高级、烧脑」的推理和下棋，对机器反而容易；人类觉得「简单、不用想」的看和动，对机器反而极难）。

让机器算积分、下棋、背下整部法典，早是小菜一碟。可让它像两岁孩子那样把散落一地的积木捡起来、躲开桌角放进盒子——这事到今天都战战兢兢。**推理容易，感知和运动难。**为什么？**看和动，进化替我们打磨了几亿年；抽象推理，才玩了几万年。**

> 大白话
>
> 你「不费吹灰之力」就能接住抛来的球、认出一张脸、闭着眼摸到口袋里的钥匙——这「不费吹灰之力」是个天大的错觉：背后是几亿年进化压进你神经和肌肉里的一套本事，只是跑得太自动，你感觉不到它在运转。而算术、下棋恰因为「新」、还没进化出专门硬件，才得吭哧吭哧地想、才觉得「烧脑」——可这种能一步步写清楚的东西，正是机器的主场。

把这悖论接回第四章那条线就通了：叠衣服、认路、扶人之所以难，正因为它们大量长在默会知识（你会做却说不清、写不进例子的那种本事）里——手上那把劲、脚底那点感觉，喂不进数据。觉得它「简单」，只因做得太顺，忘了有多难写下来。

开车：95% 很无聊，5% 要命

最典型的例子是开车。为什么写诗、翻译早被咬得七零八落，而让车自己在马路上跑，攻了这么多年还没封顶？

开车的绝大部分——姑且叫它 95%——相当无聊、重复：跟前车走、并线、看红灯停绿灯行、保持在车道中间。这些动作规规矩矩，能摊成海量「看到这个路况→该这么操作」的例子，正是第四章说的好食材。所以 demo 常惊艳——晴天、熟路，开得比不少人还稳。

问题全出在剩下那一小撮上。请出本章第二个主角：长尾（long tail，指那一大堆各自概率极小、几乎不会碰上，可品种多到加起来又常常发生的罕见情况）。它长什么样？

- 卡车掉下一个沙发，横在快车道正中间；
- 一个人从两车中间毫无预兆地窜出来横穿马路；
- 暴雨里积水反光，车道线糊成一片，什么都看不清；
- 信号灯坏了，交警在路口用手势指挥，而这手势你从没在任何教材里见过。

单拎任何一件，你这辈子可能都碰不上一次。可它们的**种类**无穷无尽——沙发之后是床垫，床垫之后是滚落的西瓜，永远有下一个。加起来，「碰上某种没见过的怪状况」经常发生。要命的是：**恰恰这 5% 决定生死。**前 95% 再顺，长尾里错一次，就是一条命。

为什么长尾这么难啃？第四章那条线就是答案：**长尾里每一件都是「第一次遇到」，周围没有可供插值的邻居。**模型学开车靠的是在海量见过的路况之间连线取中间。可「横在路中间的沙发」训练里几乎没有类似的点，只能硬着头皮外推——外推在生死关头，飘一点就是灾难。

再叠上第六章那条线——**错了要人命，担责极重。**写错一句诗没人受伤，认错一张脸大不了重来；可路上判断错一次，撞的是活人。两条线一叠，就明白那句想不通的话：

> 自动驾驶「最后 5% 比前 95% 还难」。demo 惊艳，是它把 95% 的重复动作学得很好；迟迟封不了顶，是那 5% 的长尾里全是「第一次」，每个「第一次」都可能要命。这不是工程师不努力，是这件事的形状本来如此：长尾越厚、每件越罕见、错了后果越重，封顶越慢。

关键区分：环境被人管着的，早就输了

别急着得出「体力活都难、都安全」的结论——那是另一个坑。真正决定难不难的不是「体力还是脑力」，而是**环境可不可控**：

- **结构化物理环境**：环境被人为固定住，意外被提前掐掉。最典型的是工厂流水线——零件永远从同一位置、同一角度、同一节奏送来，灯光恒定，没人乱窜。这种地方，机械臂几十年前就把人干趴下了。
- **开放物理环境**：没人替你固定，意外无穷无尽。你家客厅、马路、一栋老房子墙里那截谁也不知道怎么走的水管——每一处都在源源不断地制造长尾。

同样是「用手干活」，流水线上的活早被咬光，因为它结构化、长尾被削平；而通一根老房子的下水管难得要命，因为它开放、每根管子堵法都不一样。**这条线不是「体力 vs 脑力」，是「环境越不可控、长尾越厚的体力活，越难被咬」。**

诚实两面：别断言「永远」，也别低估「顽固」

第一面：机器人在实实在在地进步，别断言「永远做不到」。那些介于流水线和纯开放世界之间的**封闭、半封闭场景**会被相对快地吃下来：仓库搬货分拣、高速货运的干线段（路直、封闭、没有行人乱窜、长尾薄）、园区里跑固定路线的接驳车。共同点是——有人把环境收拾干净、长尾压薄了，它们会较早松动。

第二面：开放世界的「最后一公里」会比很多人预期的顽固得多。高速干线也许先通，但货送到老小区、爬上没电梯的六楼、敲开一扇门——这段路全是长尾，再叠上「错了要赔钱」的重担责，格外难啃。别被高速那段顺畅的 demo 骗了，以为整件事快成了。

> 大白话
>
> 还有个更狠、最反直觉的推论：**取代的先后，跟薪水高低几乎没关系。**照护失能老人、上门修疑难水电、做一件复杂的手工修复——这些常被归为「低薪体力活」的事，长尾极厚（每个老人、每处故障、每件旧物都不一样），又死死要求第七章那条线——**真身在场**（得有一双真手在现场应付图纸上没画的意外）。它们很可能比某些「高薪白领活」**更晚**被取代；反过来，坐在写字楼里、活儿高度可数据化的高薪岗位，可能先被咬走。谁先谁后，不看它挣多少、体不体面。

把这一刀钉死

这一章拿开车和体力活替全书钉死一个主张：**别用「体力/脑力」「高级/低级」去判断一件事会不会被 AI 取代——那把尺子是坏的。**

> 难不难被咬走，不看「高级还是低级」，看它落在那五条线的哪一侧：能不能被数据描述、有没有对错信号、错了谁担责、要不要真身在场、是解题还是重新出题。开车和体力活只是把它演到极致。

可这又逼出一个更让人发慌的问题：机器把活儿咬走之后，剩下的人是被替掉，还是被**抬起来**了？下一章讲「互补」——机器咬走一部分，怎么反而把人剩下的那部分变得更值钱，哪些是真、哪些是安慰。

第十三章 · 经济学的另一半：替代的同时在互补

上一章埋了个钩子：机器把活儿咬走后，剩下的人是被替掉，还是被**抬起来**了？前面十二章盯着「哪一刀被咬掉」，这只是半张脸。同一个技术，贬掉你一种本事的同时，往往悄悄把你旁边的另一种本事抬高。这一章翻另半张脸。

经济学里最该背下来的一对词

两个词讲成人话，是这一章的骨架。

替代（substitute，指 AI 能顶替你正在做的动作。它一旦能干、还更快更便宜，你做这个动作的价值就往下掉——你被替代了）。

互补（complement，指 AI 干得越好，你这个动作反而越值钱。因为它解开了卡住你的瓶颈，你这一环能发挥更大的力）。

> 大白话
>
> 打个最土的比方。你和一头驴一起拉磨，有人给驴喂了顿好料，驴力气暴涨。你要是也拉磨，那完蛋了，驴把你替了。可你要是干「添粮、装袋、决定今天磨麦子还是磨玉米」，驴越有劲你越值钱——它是你的互补。同一头驴同一顿料，对两种人是两种命，区别只在于：你干的是不是跟它同一个动作。

两个老例子把这对词钉实：

- Excel（电子表格）一出来，「手工一格一格算账、加总、对账」瞬间一文不值——这是替代，靠手快吃饭的账房先生被咬掉。可同时「看着数字做决策的人」大大升值：以前一天算一张表，现在一秒出一百张，「拿这一百张判断该投哪、砍哪」的人一个顶十个。算账贬值，判断升值。
- GPS 一普及，「脑子里记住整座城市的路」这门老司机的硬功夫贬了——这是替代。可它同时让「多跑单」成了可能：谁都能立刻开到陌生地址，于是拼的不再是记路，而是服务、接

单快、路上那点判断。记路贬值，跑单本事升上来。

注意：这两例里升值的那一半，都**不是**被机器咬掉的动作，而是它**旁边**的动作。这不是巧合，有机关。

核心机关：瓶颈会搬家

这个机关是这一章的心脏，叫它瓶颈转移（一条活儿由好几个环节串成，总有一环最慢、最卡人，那就是瓶颈。AI 把某一环成本压到接近零，这环不再卡人，价值和瓶颈就整个**搬**到旁边没被压掉的环节上去）。

大白话

想象一条流水线：想清楚要做什么 → 把它做出来 → 拍板上线担责。过去中间「做出来」最费劲、最卡人，谁做得快谁牛。现在 AI 把「做出来」压到几乎不要钱——瓶颈不消失，往两头挪：挪到「到底要做什么」和「敢不敢拍板、出事谁扛」。这两环没被压掉，成了新瓶颈，也是新的值钱位置。

把前面几章按这个机关重放，同一模式反复出现：

- 写代码变快了（第九章）→ 敲代码贬值，瓶颈搬到「想清楚要做什么」和「敢不敢上线、崩了我扛」。
- 出图变快了（第十一章）→ 拉线条贬值，瓶颈搬到「知道自己要一张什么图、看出一百张里哪张好、为最终用哪张负责」。

把升值的位置排一排：想清楚要做什么、判断好坏、担得起责、真身在场——**眼熟吗？**这正是前面拆出来的、AI 接不了的那几侧：重新定义问题（第八章）、判断对错好坏（第五、八章）、担责（第六章）、真身在场（第七章）。

> 所以「谁升谁贬」不是运气或玄学。瓶颈被压掉后往哪儿搬，是那五条线决定的：搬到 AI 跨不过去的那侧。站在被压掉那侧的人贬值，站在跨不过去那侧的人升值。五条线不只判「谁被咬」，反着读还判「谁被抬」。

一个反直觉的老观察：能做的越多，「挑」和「品」越值钱

有个经济学上很出名、常被讲拧的直觉，用人话摆：**把一件事变得更省更快，有时反而让某种需求更大，而非更小。**

> 大白话
>
> 历史上真发生过一件怪事：蒸汽机让烧煤发力更省，结果煤烧得不是更少而是更多——省了之后到处都想用，用它的场合暴涨。这就是杰文斯悖论（Jevons paradox，说的是：把某样东西用起来更省了，总需求反而可能涨上去）。别当铁律，只提醒你「效率提升不一定让需求缩水」。

放到 AI 上：当「做出来」又快又便宜，你能做的事一下多十倍百倍。可正因为能做的太多，「**选**做哪个」「哪个好」权重反而暴涨。以前一天只能画一张，几乎没得选；现在一秒出一百张，「知道自己要什么、一眼看出哪张好」——也就是第八章说的品味——从奢侈品变成瓶颈本身。

诚实三刀：别把互补讲成人人有份的好事

只讲升值那一面就是画大饼。互补是真的，但有三个冷面必须说清。

第一刀：互补不对每个人发生。「AI 让某种技能升值」最容易被听成「所以我不会被替代」——错。升值的是「能站到没被压掉那侧」的人。如果你的**全部**价值恰好就是那个被咬掉的动作（手快的账房、记路的司机、只会拉线条的画工），没有任何东西会自动抬你——你就是被替代的那个。互补不是发给所有人的红包，只落在能挪窝的人头上。

第二刀：转型是真疼的，别用「长期看都会更好」糊弄短期。「瓶颈搬家了、新位置更值钱」说得轻巧，可搬家本身要人命：岗位重组、旧技能一夜作废、新技能从头重学，而学不会、挪不动的人，实实在在掉下去了。历史上每一次「长期看更好」，中间都躺着一批没熬到长期的人。谁也没资格拿一句「长远会好」，抹掉别人短期的真实坠落。

第三刀：升值的位置自己也在移动。今天的互补技能，明天可能变成替代对象。今天「会写提示词指挥 AI」很值钱，可它也是一个动作，明天 AI 可能就接管，瓶颈又往前搬一格。所以别指望「一劳永逸的安全岛」——安全的不是某个具体技能，而是**持续往瓶颈搬去的方向站**这个习惯。

把问题换一个问法

大多数人盯着 AI，问的是同一句：「我会不会被替代？」这只看见半张脸——只盯着替代，看不见同一技术另攥着的半张脸。

> 换成这么问：**当 AI 在我这条流水线上变强，它会把哪个环节压到接近零？瓶颈会因此搬到旁边哪个环节？那环节是不是五条线里 AI 跨不过去的那侧？我能不能挪过去？**

第一个问法把你锁死成待宰的动作，让你焦虑；第二个逼你去找正在升值的相邻位置、诚实掂量自己挪不挪得过去，给你一张地图——尽管地图上也画着「有人会掉下去」的坑，可能包括你。

那么这张地图怎么读？下一章，我们把五条线拧成一张能上手用的**判断表**，让你拿任何一个岗位、任何一个动作往上一套，就看出它落在哪侧、瓶颈往哪搬。

第十四章 · 一张能自己用的判断表

前面十来章，我们一根一根画了五条线。现在把它们拧成一张**你能亲手用的判断表**——随便拎出一件事，戴上这副眼镜，你自己就能估它会被咬掉几成，不用等谁发「高危职业排行榜」。

先说一句决定这张表准不准的用法：**表是给「任务」打分的，不是给「职业」打分的。**回到第二章那个颗粒度——一个岗位是一大捆任务，机器一次只咬得动其中几条。正确姿势是：先把岗位**拆成一条条具体任务**（「录发票」是一条，「跟税务局解释一笔奇怪支出」是另一条），再拿每条逐一过五问。别指望给整个「会计」打一个分。

五个问题：把五条线变成你能自问的话

下面五问，就是全书那五条线，翻成你能对着一件任务直接问出口的句子。

1. **这件事，能不能被大量「这样输入→那样输出」的例子描述清楚？**（第一条线：可数据化）
   把它做过的一万次整理成「碰到这种情况就这么做」的配对喂给机器，它能不能靠模仿学会？**越能，越危。**翻译、抄写、给图片归类，正是刀最先落下的地方。反过来，那些你会做却说不清、写不进例子的默会知识（手上那把劲、临场那点感觉），喂不进数据，留得住。

2. **它有没有一个说得清对错、而且马上能验的信号？**（第二条线：反馈信号）
   做完有没有一个干净利落的「对/错」判据，立刻告诉机器它做得好不好？代码能不能跑、这步棋输赢、这数算得对不对——信号又快又干净。**信号越干净，越危**，因为机器就靠它一遍遍自己纠错、疯狂练出来。反过来，「这封安慰信暖不暖」「战略三年后对不对」信号又慢又含糊，机器没法自己练，就黏在人手里。

3. **做错了，代价大不大、可不可逆、要不要有人签字担责？**（第三条线：担责）
   搞砸一次，是「重来一遍就好」，还是「有人受伤、有人赔钱、有人得签字背锅」？**代价越大、越不可逆、越要有人担责，越安全。**注意这条线方向跟前两条**反过来了**：前两问是「越怎样越危」，这条是「越怎样越安全」。出了人命、赔了大钱的事，社会需要一个能被

追责的**人**签字，机器再准也顶不了。写错一句诗没人受伤（危），开错一刀要人命（安全）。

4. **它要不要真身在场——现场动手，或者一个人对着另一个人？**（第四条线：物理/关系在场）
   它离不离得开一双真的手在现场应付图纸上没画的意外？或者本质上是「一个人对另一个人」、需要真脸真信任的关系？**越要真身在场，越安全。**上门修疑难水电、把老人扶到轮椅上、当着家属的面讲坏消息，死死地要一具真身，机器隔着屏幕顶不上。纯在屏幕里就能干完的事，这道护栏就没了。

5. **它主要是「把给定的问题解出来」，还是「判断该解哪个问题」？**（第五条线：执行 vs 判断）
   题目是别人出好的、我只管做对（执行）？还是难的部分恰恰是**决定该做哪道题**、什么才算「做好了」（判断/定义问题）？**越偏执行越危，越偏判断和重新定义问题越安全。**「照这份需求把功能写出来」是执行（危）；「先想清楚这产品到底该解决哪个真问题」是判断（安全）。机器极擅长把一道**出好的**题解得又快又漂亮，但「该出哪道题」它接不住。

怎么把五个答案合成一个直觉

你可能想问：这五个答案怎么加起来算个总分？**别把它做成一个打分器。**真做成「每问 20 分、加起来 100 分」，看着精确，其实是拿假小数点糊弄你——五条线权重根本不一样，还随任务在变。这张表是加权直觉（帮你看清结构的眼镜，不是吐分数的计算器）。

但有一个**直觉分工**，比记任何公式都管用：

> **前两问，管的是「多快被咬」；后三问，管的是「就算能被咬，这一刀是不是还得留给人」。**

大白话

一件事「能被海量例子描述 + 信号又干净又快」，**技术上**迟早会被学会。但就算学会了，如果它**错了要命、要人签字、要真身在场、还得靠判断而不是照做**，社会也未必敢整个交给机器——它会留在人手里，或变成「机器打草稿、人拍板签字」。

两个极端就清楚了。一件任务**高度可数据化 + 信号干净 + 低责任 + 不需真身 + 纯执行**——五问全落「危」侧——几乎必被吃掉。反过来样样落在后一侧，它最稳。但记住第二章那句话：**多数任务落在中间**，会被**部分**咬掉。取代的从来不是整个人，是他手里那捆任务里的几条。

实际走一遍：拿两对任务打分给你看

第一对：写营销邮件初稿 vs 跟谈崩了的大客户续约

「写一封营销邮件的初稿」——可数据化？**能**（危）。信号？**有**，打开率当场能测（危）。担责？**不大**，重写一封就好（危）。真身在场？**不要**（危）。执行还是判断？**执行**（危）。
结论：**这一刀会掉。**

「跟一个已经谈崩、扬言不续约的大客户重新谈成」——可数据化？**很难**，每个客户崩的原因都不一样（安全）。信号？**没有**，全靠揣摩（安全）。担责？**大**，丢大客户是真金白银（安全）。真身在场？**要**，靠脸、靠信任、靠临场读脸色（安全）。执行还是判断？**判断**，难在摸清「他到底为什么不爽」（安全）。
结论：**这一刀留下。**机器顶多帮你查历史、拟话，谈判得人去。

第二对：给 X 光片做初步标注 vs 把坏消息告诉病人家属

「给一张 X 光片做初步标注，圈出可疑阴影」——可数据化？**极能**，几十万张标好的片子（危）。信号？**干净**，后续能对上（危）。担责？留神——标注常被切成「机器初筛、医生拍板」：**标注本身低责任**（危），但**最终签字那一步高责任**（留给人）。真身在场？**不要**（危）。执行还是判断？**执行**（危）。
结论：**标注这一刀会掉，签字那一刀留下**——一个岗位被部分咬掉。

「把『是恶性的』这个坏消息，当面告诉病人家属」——可数据化？**不能**，每家的承受力、关系、情绪都不同（安全）。信号？**不**，「讲得好不好」没有当场对错（安全）。担责？**极重**，是人对人的道义之重（安全）。真身在场？**死死地要**（安全）。判断还是执行？**全是判断**（安全）。
结论：**这一刀，怎么都留下。**

同一个「放射科医生」岗位，拆开之后读片标注暴露在刀口下，告知、拍板、担责稳稳留下。**给岗位打一个分毫无意义，给拆出的每条任务打分才有意义。**

三个别踩的坑

- **线是会移动的，今天的「否」明天可能变「是」。**数据在被人不断制造，信号在被工程化。某条今天答「否、留下」的任务，过两年可能滑到「是、被咬」那侧。**这张表要定期重打。**
- **别拿它当算命。**它告诉你的是**结构**——这件事凭什么危、凭什么安全，而不是**年份**。它不会说「三年后你失业」，只帮你看清它落在刀的哪一侧、离刀口多近；刀什么时候落，还掺着成本、法规、人心。看结构，别问日子。
- **一个岗位的「得分」，是它那捆任务的加权，别指望一个数。**安不安全取决于被咬的占多大比重、留下的那几条值不值钱。有的岗位被咬掉七成、剩三成还更值钱（ATM 之后的柜员），有的被咬掉三成就塌了。

你现在有了一副能自己戴的眼镜

到这儿，全书的工具就交到你手上了。你不再需要背「高危职业排行榜」——那种清单三个月就过期，还让你变蠢。这副眼镜不会过期，因为它看的是**结构**。

> 随便拎出一件事——你工作里的某个动作、你正犹豫要不要学的某项技能——拆成任务，逐条过五问，你就能看清哪几刀会掉、哪几刀留下，以及**为什么**。这就是本书的全部。

可看清了刀落在哪儿，还剩最后一个、也最要紧的问题：**那你自己，该往哪儿站？**该把力气往哪条线的哪一侧挪，才能站在刀留下的那一边，甚至站到那把刀的**握柄**上去？这是最后一章的事，下一章见。

第十五章 · 那你该往哪儿站

全书该讲的都讲完了：五条线备齐，四个战场拆过，替代与互补、瓶颈往哪儿转，都说了。最后这章不讲道理，只落到一个具体问题：**你，该往哪儿站。**不给「五个生存技巧」，也不拍肩膀说「一切都会好」——那些话听着舒服用起来没用。这本书从第一页就拿解剖刀干活，最后一刀照旧对你自己下。

先把五条线拎成一句话

前面十四章，说穿了在反复问同一件事的五个侧面。把这五问背熟，比任何清单都值：

- **能不能被数据描述？**（第四章）有没有海量现成例子，让模型在里头连线取中间。有，它就是主场；没有，只能硬外推。
- **有没有干净的对错信号？**（第五章）做完能不能马上、明确地判出对错。能，它就靠这信号练上去；判不清，就练不动。
- **错了谁担责？**（第六章）出岔子是重来一遍就行，还是要有人在终点签字、扛后果。担责越重，越离不开人。
- **要不要真身在场？**（第七章）这事能隔着屏幕干完，还是非得一具真身体、一段真关系在现场。
- **是解题，还是重新出题？**（第八章）题目摆好、只等求解，机器擅长；而「该解哪道题」要有人定，这一步它接不了。

这五问背后，是全书最要紧的主张，最后一次钉它：

> 被咬走的从来是**任务**，不是人。一份工作是几十件任务捆在一起，AI 一件件地咬，咬哪几件、留哪几件，只看它落在五条线的哪一侧——**不看它是体力还是脑力、高级还是低级、挣得多还是少。**那把「高低贵贱」的尺子是坏的，第十二章已拿开车和体力活按死了。

站位的总原则：往那几侧挪

既然被咬的任务都聚在五条线的同一侧，那你该做的就一句话：**把能力结构的重心往对面那几侧挪。**

- **会定义问题，不只是解题。**能看出「现在真正该解的是哪道题」，而不是等别人把题目摆好再动手。
- **担得起责。**能在终点签字、为对错负责，而不只是中间某道工序的执行者。
- **真身在场。**你的价值里有一部分非得靠一具真身体、一段真关系才成立——现场的手，或活人之间的信任。
- **跨任务的整体判断。**AI 把碎活加速到飞快，可谁把这堆碎活组织成「一个对的东西」？把方向、次序、取舍拢一起的判断，它接不了。

这四条没一条是「学会某个技能」，不是报班拿证，而是**能力结构的重心迁移**——你吃饭的本事可能有一半正落在会被咬那侧，得往咬不到的那侧转。慢、别扭、没速成，但真管用。

三个错误答案，得当面点破

市面上「该怎么办」的回答大半是错的。挑最流行的三个拆掉。

错答一：「学 prompt、学用 AI 工具，就稳了。」会用工具是**入场券，不是护城河。**技巧三个月换一茬，人人都在学，凭什么「非你不可」？但假装 AI 不存在会**先出局**——会用是底线，别把底线当终点。

错答二：「找个 AI 够不到的角落，躲一辈子。」问题出在「一辈子」。第十四章讲过——**那条线一直在移动。**今天够不到的角落两年后未必，你安身的一处可能正是下一波要淹的地方。能行的只有：**站到价值转移的那侧，跟着它一起移动**——不是躲进石头，是在移动的水面上换脚。

错答三：信「某某职业绝对安全 / 绝对完蛋」的清单。不存在整个职业级别的判决。一切都是**任务级**的，一份工作里几刀掉、几刀留混在一起，刀口还都在动。谁甩给你一张「十大铁饭碗 / 十大将消失职业」，直接扔了——它卖确定感，不是真相。

两条实路：一条给你们，一条给所有人

第一条写给工程师和各行的专业人士。核心就一句，第九章原样搬来：**被咬走的是打字，不是设计。**你要把重心从「亲手把活做出来」挪到「判断该做什么、把 AI 当放大器、对产出负责」。具体动作：

1. **主动去碰那些模糊的、要定义问题的、要担责的活。**拆需求、定方向、做决策、扛结果——这些第九章里最靠上、最咬不动的任务，别绕着走，往里钻。
2. **别死守纯执行。**那些一眼就知道会被自动化的执行环节，练得再熟也是把重量押在会塌的那侧。
3. **学会驾驭 AI 的产出，而不是被它淹没。**它一秒吐一屏代码、一屏方案，你要**看得懂、审得了、担得起**——挑得出它哪儿错、敢在它产出上签字。这才是放大器握在你手里而不是被它推着跑。

第二条给所有不写代码的普通人。我不劝你们都改行当「定义问题的人」——那不现实，给一条更朴素、更能用的：

1. **把自己岗位里的活按五条线过一遍，看清哪几刀会掉、哪几刀留下。**不用精确，八九不离十就够。重复的、可数据化的、没人担责的执行迟早掉；要担责、要跟人打交道、要现场判断的，留得住。
2. **把时间往留下的那侧投。**少花力气在会被咬的那几刀上，多往留得住的那几侧挪。
3. **最朴素也最要命：保住能学新东西的能力。**线一直在动，你今天挪到的位置明天还得再挪。能不能持续学，比此刻会什么更决定你走多远。

最后：我给你的不是清单，是一副眼镜

这本书从头到尾没给过你一张能背下来照做的清单，这是故意的。清单会过期——线在移动，工具在换代，今天写死的答案明天就错。它给一次性的确定感，而你要在不断变的世界里活很多年。

我想给你的，是一副你能自己戴上、往后一直用得着的眼镜。每来一次新浪潮，不必等谁告诉你「这次谁完了、谁稳了」，自己拿五个问题一照就行：**能不能被数据描述、有没有干净的对错信号、错了谁担责、要不要真身在场、是解题还是重新出题。**照完，你就知道刀口切在哪儿，自己该站哪儿。

> 所以别再问「我会不会被取代」——这问法把你当成一整块、等着被判死刑或缓刑。要问的是：「我能不能一直站在那把刀够不到的那侧，并且跟着它一起移动。」

刀会一直往前咬，这拦不住也不必拦。你要做的不是躲开，是看清它下一口往哪儿咬，先它一步站到咬不到的那侧。戴上眼镜，剩下的路你自己能走。

AI 咬走的那一刀 · holly