Google

进化级智能体蓝图：Harness2 运行机机制深度拆解

详解 Propose-Probe-Compose

详解 Propose-Probe-Compose

三步曲与开放世界下的并行/顺序探索流



基于 Google Cloud AI Research 前沿学术成果
"Recursive Agent Harnessing for an Open World" 提炼

Gemini Notebook

智能体的能力边界，不仅取决于底层模型，更取决于其运行脚手架。



大模型（冻结权重）

Harness（动态脚手架）

智能体（Agent）

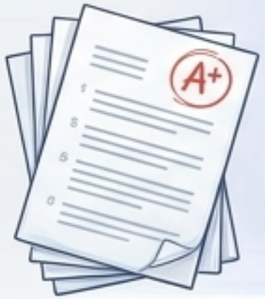
Harness2 核心思想：在不改变模型参数的前提下，通过自动化、递归式地迭代这 8 个外围组件文件，让智能体持续进化。

Gemini Notebook

封闭测试（有标准答案 Ground-Truth）



状态：像在考场做选择题，对错分明，易于自动化评分和优化。



真实开放环境（无标准答案）





致命拷问：在没有验证器、不知道绝对正确答案的情况下，如果修改了脚手架，Agent 如何判断自己是改好了还是改坏了？

传统的预设 Harness 无法应对动态变化；我们需要一种能在无监督状态下自我验证的迭代引擎。

Gemini Notebook

寻找行为差异：Propose-Probe-Compose 对比式引擎

放弃寻找绝对正确，转而捕捉行为差异 (Behavioral Contrasts)

Propose（提议）
- 起草不同修改方案

Compose（重组）
- 仅采纳缓解缺陷的证据

前置输入： 基础 Harness 的初始执行轨迹（e_0）



产出： 基于证据重组的新 Harness

Probe（试探）
- 同等环境下空跑测试

Harness2 将优化过程转化为一场精密的对比实验，让每一步修改都建立在可观测的证据之上。

Gemini Notebook

Step 1: Propose（生成提议）—— 双线并发的诊断机制

输入：当前任务 + 历史轨迹 + 改进库



Invariant（机制不变性提议）

- **关注点：**Agent 如何工作（How it works）
- **修改范畴：**工具调用的错误处理、上下文记忆管理、防止死循环的防呆逻辑
- **目标：**寻找跨领域通用的、防御性的架构机制



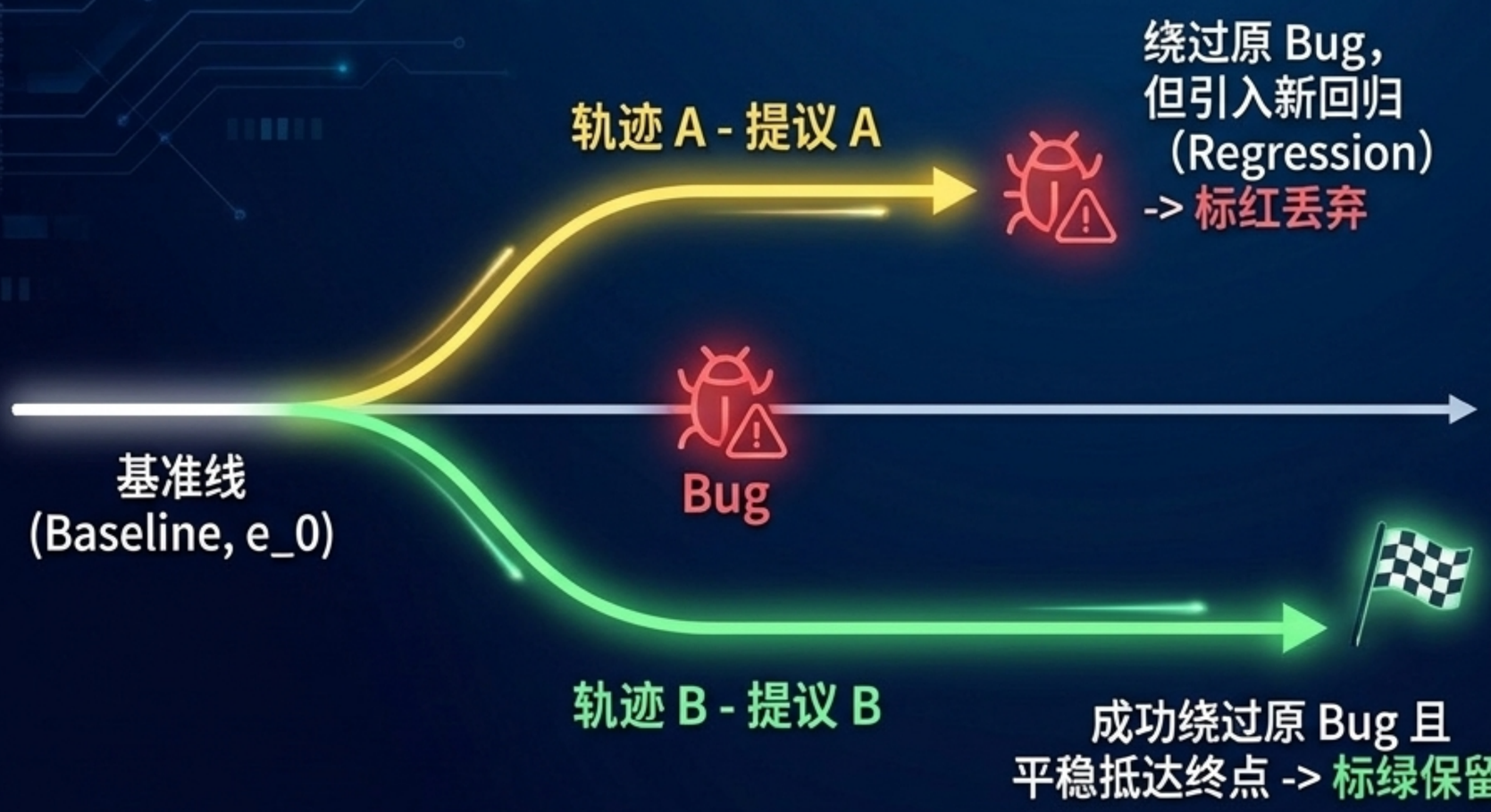
Domain-specific（领域特定提议）

- **关注点：**从业者知识（What a practitioner knows）
- **修改范畴：**特定行业的排版约束、专有名词的映射关系、特定任务的验收结构
- **目标：**将高昂的推理成本转化为固化的、可直接读取的环境事实

重要提醒：提议的产物不是几句文本建议，而是完全可执行的 Harness 配置文件。

Gemini Notebook

Step 2: Probe（试探验证）—— 无答案环境下的天平



判定逻辑

不看什么：

~~不盲目评估最终产出是否完美（因为没有标准答案）。~~

只看什么：

1. 目标缺陷是否被有效缓解？
2. 相关联的健康行为是否保持原样？
3. 是否意外引入了新的能力退化？

结论：行为对比（Behavioral Contrasts）成为无监督环境下的『代理真理』。

Gemini Notebook

Step 3: Compose（重组融合）—— 基于证据的精密拼装

基础记录 (e_0)

候选 Harness 集合

Diff 代码差异

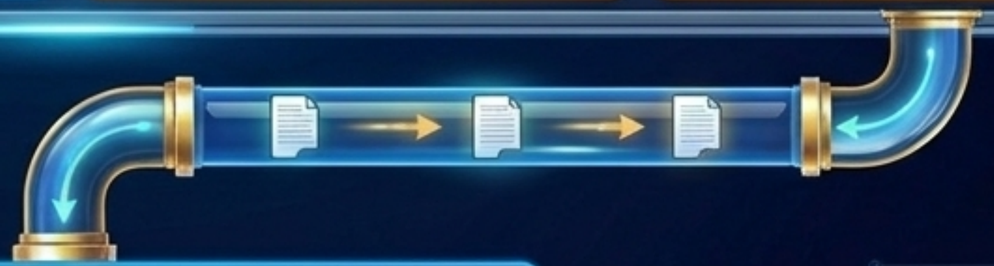
试探执行轨迹

🛠️ **Merge（合并）：** 将证明有效的底层机制与正确的领域知识缝合。

✂️ **Revise（修正）：** 解决不同提议编辑同一文件产生的冲突。

🎯 **Adopt（采纳）：** 某一提议表现出压倒性优势时，直接整体采用。

🛡️ **Keep Base（保持）：** 若证据都不充分，果断放弃修改，退回原始状态。

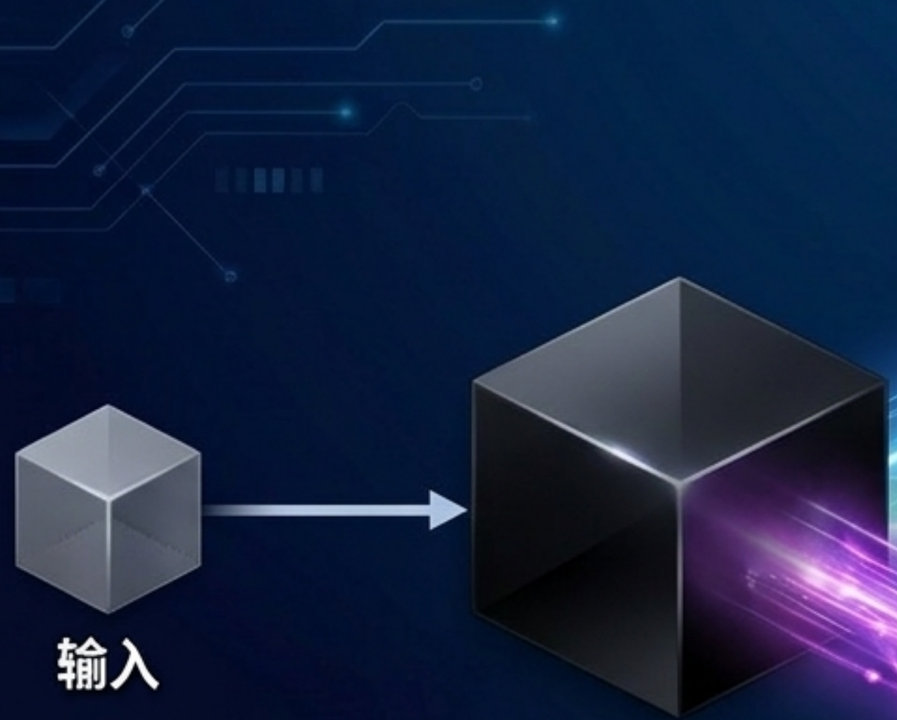


单一精致的后继 Harness 组件

没有任何一项决策基于虚构的打分提升——每一行代码的修改，都必须绑定在具体的试探观测证据之上。

Gemini Notebook

引擎运转的结晶：一次成功迭代的输出清单



新版脚手架（H’）

经过验证和重组的、可直接用于下一回合的可执行配置。

新轨迹记录（e’）

在新脚手架下运行产生的、更优的系统执行过程与最终交付物。

宝贵的步骤记录（ξ）

包含『为什么这么改』、『对比证据是什么』的元经验萃取。

一次 P-P-C 足以修复浅层错误。
如果给系统投入更多算力（预算 k），它还能进化出怎样的形态？...

Gemini Notebook

算力换取智能：引入任务级递归（Task-Level Recursion）

单次 P-P-C 循环视野有限。
Harness2 将算力预算引入系统，
升级为任务级递归引擎。

模式 A：并行（Parallel）

广度优先，榨干并发算力，发挥群体智慧。



模式 B：顺序（Sequential）

深度优先，踩石过河，实现逻辑的深度纠偏。



算力预算参数
（Budget k）

这不是简单的多次重试，而是对探索空间的系统性拓展。

Gemini Notebook

模式 A：并行递归（Parallel）— 广度优先的群体智慧



并行获取 k 个初始 Harness 的执行基线。

同时启动 k 个 Invariant 提议和 k 个 Domain 提议。

开启 k 个互相隔离的 Compose 议程，生成 k 个候选者。

关键环节：裁判在不知道来源的情况下，对最终交付物进行盲态评估，选定唯一胜出者者（H^*_t）。

性能注脚：最大化利用云端高并发算力，极大地拓展探索宽度，且物理延迟极低。

Gemini Notebook

模式 B：顺序递归（Sequential）— 深度优先的迭代进化

链式推进: 将 k 次 P-P-C 对比步骤首尾相连，串行执行。

状态传递: 第 r 步的输入是第 r-1 步产生的最新 Harness。

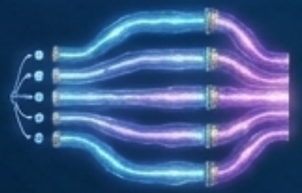
记忆累积 (Memory Accumulation)
随着台阶上升，系统不仅带着最新配置，还携带着之前所有回合的累积执行历史 (E^{r-1})。



能力特写： 具备『自我纠偏』能力。后一步能用新证据推翻或修正前几步的错误，极度适合逻辑链条深、阻碍嵌套的任务。

Gemini Notebook

架构抉择：并行 vs. 顺序 探索模式对比矩阵

	并行 (Parallel)	顺序 (Sequential)
运行逻辑 (Logic)	 多路径独立发散，终局盲选决胜 (广度优先)	 单一主干持续推进，踩石过河 (深度优先)
证据视野 (Evidence Horizon)	 仅依赖当前单回合的多样本对比	 拥有所有历史回合的累积记忆与证据链
算力与时间特征 (Compute & Time)	 极速消耗高并发算力，系统延迟 (Latency) 极低	 算力消耗随时间线性增长，需要串行等待
最佳适用场景 (Ideal Use Case)	 任务目标宽泛、解法发散且不确定性高时	 任务存在极深层的逻辑链条，需要反复试错排雷时

Gemini Notebook

宏观顿悟时刻：Harnessing-Level 的跨任务递归

刚才的步骤只是解决当前任务，Harness2 的真正伟大在于它的『改进级』进化。



1. 🔍 Reflect（反思萃取）
回顾成功的 Harness 和步骤记录，反思为什么这样改生效了？

2. 📚 Update（更新改进库）
将经验教训作为规则存入改进库（Improvement Library）。

3. 🚀 Forward（反哺未来）
全新任务到来时，带着前人的经验法则去引导新的生成。

> 真正的智能不是解决眼前的问题，
> 而是通过解决问题，实现了『改进机制本身』的递归进化。

Gemini Notebook

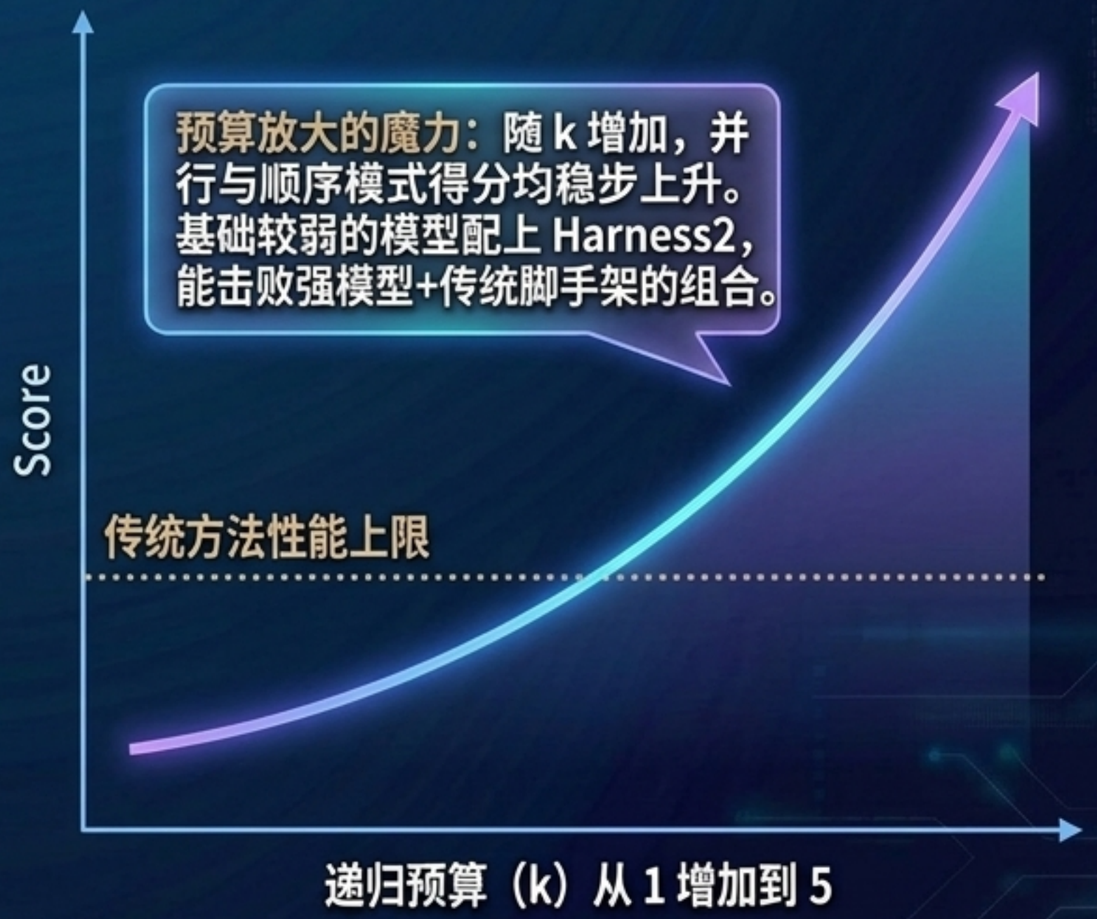
效能验证：无需标注数据的跨越式提升

Metrics Board



- WorkBuddy-Bench 巅峰跃升：单次 P-P-C 让 GLM-5 实现惊人的 +11.5 分 绝对提升。
- JobBench（专业长线任务）：+4.5 分
- LAB（极难法律文本域）：+11.8% 局部通过率

Scaling Trend



Gemini Notebook

总结：重塑大模型智能体（Agent）工程的未来范式



告别盲目调试，拥抱证据主义

Propose-Probe-Compose 架构打破了无标准答答案的死局，用行为对比代替绝对判分，让脚手架的每次迭代基于确凿证据。



推理算力转化为系统稳定性

通过并行与顺序两种扩展模式，Harness2 证明了在测试时投入算力，能够稳定转化为复杂任务的成功率。



从做题家到进化工厂

宏观的经验改进库实现了经验跨任务迁移。系统不仅在完成任任务，更在沉淀优化方法，实现 AI 自我改进的飞轮效应。



Gemini Notebook