

固定模型规模下的 AI 编程能力跃迁：ES-DSI 社会级跨代进化机制的公共证据与可检验假说

版本：V1.1

摘要

当前对 AI 能力进步的解释，通常集中于规模扩展（scaling）、训练算法改进、自训练与推理计算等方向。然而，软件工程领域出现了一个值得单独研究的现象：即使参数规模与模型架构不再扩大，甚至沿用同一预训练基础模型，后训练仍可带来数十个百分点的软件工程基准增益。

本文所谓**固定模型规模**，特指参数规模与模型架构容量不增加；它并不意味着模型权重内容、有效能力、后训练计算或推理策略保持不变。本文基于知识—结构—约束（KSC）框架对既有知识与固化结构（K）和新结构生成（S）的区分，对 DSI 与 ES-DSI 给出完整定义和操作化说明。****结构分布内化（Distributional Structural Internalization, DSI）****描述训练环境中可迁移问题解决结构的分布如何转化为模型在面对结构相关新任务时所表现出的不同初始生成倾向。****外部选择—结构分布内化（ES-DSI）****描述候选结构在现实世界中经过差异化保留、传播和后续有效训练暴露，进而可能改变后代模型初始结构倾向的跨代过程。在经验分析部分，本文采用逐层控制的同系列代际比较：依次控制参数规模、架构、基础模型和部分训练条件，观察在这些控制下仍然保留的能力增量。配套描述性数据集目前包括 21 个模型或版本和 12 组同系列纵向比较。

在代表性案例中，DeepSeek V4 Flash 的架构和尺寸保持不变，官方明确说明仅重新进行后训练；其官方同源 DeepSWE 评测结果从 7.3 提高到 54.4，增加 47.1 个百分点。GLM-5.3 使用与 GLM-5.2 相同的基础模型，其官方 DeepSWE v1.1 结果从 46.2 提高到 66.9。作为数量级参照，在 DataCurve 统一评测中，Gemini Flash 和 Claude Opus 等若干前沿模型相邻版本的 DeepSWE 增量约为 11–18 个百分点。由于不同模型系列的纵向结果并非全部来自同一评测实现，且本文尚未依据完整样本分布定义“典型前沿版本增量”，这些跨系列差值只用于提供数量级参照，不作为严格的横向性能比较。

进一步的受控研究表明，在模型、训练流程或计算预算受到不同程度控制后，任务构造、执行环境和问题解决轨迹的差异仍可显著改变未见任务上的泛化能力。这些结果与 DSI 关于训练结构分布影响后续结构生成倾向的预测一致，但尚不构成对初始结构倾向变化的直接测量。与此同时，编程智能体已经大规模参与真实软件工程；AI 生成、工程选择、数字结构传播以及后续训练暴露等 ES-DSI 所需环节，也已分别出现可观察实例。

基于这些证据，本文提出中心机制主张：

ES-DSI 应作为一种有别于规模扩展、训练算法创新和封闭式自训练的社会级 AI 跨代进化机制，单独加以建模和检验。

其基本时间单元不是一次模型训练，而是“模型—现实世界—后续模型”：AI 与人类持续产生候选问题解决结构，现实环境完成外部选择，胜出结构改变未来数字环境中的结构分布，后续模型再通过 DSI 将这种变化转化为新的初始结构倾向。

需要强调的是，当前公共证据尚未围绕同一批结构完整追踪“生成 → 选择 → 传播 → 后续训练暴露 → 初始结构倾向变化 → 性能变化”的全链因果过程。因此，固定模型规模下的性能跃迁不能直接归因为 ES-DSI；它首先说明，模型规模不继续扩大时仍存在能力增长空间，而其他证据则分别支持 ES-DSI 各关键环节的现实可行性。

这些观察使一个更强的假说值得直接检验：

ES-DSI 可能已经成为现阶段前沿 AI 编程与智能体能力增长的主要驱动机制之一。

本文目前的直接证据主要来自软件工程，因此这一判断能否推广到总体 AI 能力增长，仍需数学、科学推理及其他领域的独立检验。现有证据也尚不能完成 ES-DSI、规模扩展、训练算法和推理计算之间的贡献率分解。因此，本文将该判断明确视为一个可观察、可证伪、可量化的研究假说，而不是已经得到证明的经验结论。

关键词：KSC；DSI；ES-DSI；AI 跨代进化；编程能力；后训练；DeepSWE；外部选择

1. 引言：AI 能力增长是否存在一条被低估的路径

过去数年，规模扩展是解释 AI 能力进步的重要轴：增加参数、数据与训练计算，模型能力通常随之提高。

随着训练体系复杂化，强化学习、验证器、自训练、合成数据以及测试时计算也逐渐成为重要的能力来源。这些机制并不互斥，也仍在持续发展。

但这些路径是否足以涵盖当前 AI 能力增长的主要来源，仍是一个开放问题。

软件工程特别适合研究这一问题。一方面，AI 编程能力已经从局部代码生成扩展到代码库检索、跨文件修改、长程任务处理和环境交互；另一方面，代码执行、测试、CI 和生产反馈使许多候选方案能够接受外部验证和长期选择。

这使一个基础问题逐渐具备经验检验的条件：

如果部署模型的参数规模和架构不继续扩大，其能力还能增长到什么程度？

本文将这种条件简称为“固定模型规模”。它只表示参数规模和架构容量不继续增加，并不意味着后训练、模型权重内容、推理预算或实际能力保持不变。如果模型规模固定后的改进始终只是边际性的，那么模型规模仍可近似视为能力增长的主要约束之一。反之，如果固定参数、固定架构乃至固定基础模型后，仍能持续出现大幅能力增益，那么当前 AI 能力增长图景至少需要额外考虑规模之外的能力增长轴。

KSC 框架将复杂目标任务中的功能区分为当前任务可调用的知识与固化结构（K）、新结构生成（S）和约束保持（C）。[1] 在这一坐标中，本文使用 DSI 与 ES-DSI 描述一个时期发现的结构如何改变后代模型解决问题的起点，并在后文完整定义两个机制及其经验变量与证据边界。

本文围绕三个研究问题展开：

RQ1： 固定参数规模、架构甚至基础模型后，是否仍存在大幅编程能力增益？

RQ2： 在进一步控制模型、训练流程、计算预算或数据规模以后，任务构造、执行环境和问题解决轨迹是否仍会系统影响泛化能力，并与 DSI 的结构分布预测一致？

RQ3： 当 AI 生成、外部选择、传播、后续训练暴露和能力变化等关键环节已经分别出现时，ES-DSI 是否应作为一种独立的社会级跨代进化机制被建模，并进一步检验其是否已经成为当前编程与智能体能力增长的重要来源？

前两个问题构成经验基础，第三个问题包含两个不同层级：ES-DSI 作为独立研究机制的提出，以及“主要驱动机制之一”这一更强的待检验假说。

2. 分析框架：KSC、DSI 与完整 ES-DSI 机制

2.1 本文使用的 KSC 坐标

知识—结构—约束（KSC）框架将复杂目标任务中的关键功能分为三个具有递进依赖关系的功能角色：[1]

- **K（知识）：** 系统当前能够访问和调用的信息、状态与已固化的结构；
- **S（结构生成）：** 在 K 的基础上形成新的问题表示、变量关系、抽象、解决方案和行动路径；
- **C（约束保持）：** 使既定目标、前提、边界、不变量和验收条件持续约束生成与行动。

本文主要使用 K 与 S 的关系。既有 K 改变 S 可用的材料和起点；S 产生的新结构则可能被保存在模型之外，并在满足后文条件时成为后代模型可用 K 的一部分。C 不是本文提出的跨代机制中的主要变量，但必须将其与外部选择区分：C 在任务或系统内部使既定条件持续有效；外部选择则发生在候选结构生成以后，改变这些结构被长期保留的概率。

DSI 与 ES-DSI 用于进一步说明历史结构经验如何影响后续 K 与 S。为使本文的论证能够独立闭合，本章将完整说明分析单位、传递链、可观察变量，以及该机制能够或不能产生正向能力积累的条件。

2.2 分析单位：可迁移的问题解决结构

本文所说的**问题解决结构**，是能够在表面细节不同、但底层关系相关的任务之间复用的问题表示、变量关系、程序、设计模式、解决方案、搜索方向或验证逻辑。

仅复制答案、复现固定表达或输出特定任务结果，并不足以构成本文意义上的结构经验。训练所得内容必须能够影响模型对结构相关新实例的处理。在软件工程中，所有权与生命周期关系、异步任务协调方式、回滚

结构、测试策略和跨组件变更的拆分方式，都可能构成这种结构；同一结构可以通过不同代码、术语、代码库和局部约束表达。

这一迁移标准将本文机制与简单重复区分开来，也明确了后续实验应测量什么：不仅要观察训练样本是否被复现，还要观察相关暴露是否改变了模型面对新问题时优先或较早生成哪些结构。

2.3 DSI：从结构经验到初始生成倾向

本文将**结构分布内化（Distributional Structural Internalization, DSI）**定义为：

训练将数据或交互环境中可迁移问题解决结构的分布转化为模型面对结构相关新问题时的初始生成倾向。

这里关注的不只是模型能否陈述某个概念，而是在大规模在线搜索、工具反馈或多轮修正开始以前，哪些问题表示、变量关系、解决模式和探索方向已经比其他候选结构更容易进入初始候选集合。

例如，在处理异步加载共享资源的生命周期时，一个模型可能首先使用分散的局部标记，另一个模型则可能直接围绕所有权、引用计数或显式生命周期协议组织问题。DSI 关注的是历史结构经验如何改变这些候选结构出现和被优先考虑的相对倾向。

DSI 不意味着结构决定论。既有 K 改变的是 S 的起始分布，并不强制最终答案，也不排除新结构生成。DSI 也不是对“模型学习训练分布”这一一般事实的重新命名；它只讨论问题解决结构跨表面实例变化的迁移。模型架构、提示词、解码设置和训练目标同样可能改变生成倾向；只有在某种倾向的变化与相关训练暴露差异建立联系后，它才构成 DSI 的证据。

本文以**初始结构倾向**作为主要行为表述；“结构先验”只作为同一生成不对称性的简写，并不预设已经识别出某种具体的贝叶斯先验或内部表征。

2.4 外部选择：生成之后的差异化保留

AI 与人类持续产生候选结构，但生成本身并不决定哪些结构会改变未来环境。一些方案通过测试、在生产环境中稳定运行、保持可维护性、获得采用并被复制到更多系统；另一些方案则会消失。

本文将**外部选择（External Selection, ES）**定义为：

候选结构生成后，由目标模型训练闭环之外的过程造成的差异化保留，包括工程执行、测试、生产反馈、人类判断、用户采用、同行审查和长期实践。

ES 不需要统一的中心选择者或单一适应度函数。只要不同结构在现实过程中获得不同的存续、复用和传播概率，外部选择就已经存在。它也不保证选择方向必然与结构质量一致：可见性、市场激励、制度条件和网络效应都可能保留较弱结构或压制有效结构。因此，正向积累要求外部选择与相关领域中的实际有效性具有足够强的相关性。

2.5 传播与有效训练暴露

外部选择与 DSI 之间还隔着两个必要的传递条件。

第一，胜出结构必须得到**传播**。结构被采用、复制、写入文档、进入库或框架，并持续存在于实际系统中，才会改变未来数字环境里的结构分布。一个结构即使在局部取得成功，如果没有留下可持续痕迹，也很难显著改变后代模型面对的环境。

第二，传播后的结构必须获得**有效训练暴露**。公开可访问不等于进入特定训练集；进入训练集也不等于在训练中拥有足够权重和足够多样的表达，或出现在能够改变后续行为的训练阶段。模型训练者可以选择数据来源和训练流程，却很难以足够规模直接构造具有高结构价值的训练实例。现实选择与传播因此能够改变后续训练可以吸收的候选结构池，而训练设计仍决定其中哪些部分真正被内化。

传播与有效训练暴露是传递条件，而不是独立的进化机制。任一环节失效，ES 的结果都无法进入 DSI。

2.6 完整 ES-DSI 链条

完整机制可以表示为：

$G(t) \rightarrow \text{Sel}(t) \rightarrow R(t) \rightarrow E(t+1) \rightarrow D(t+1) \rightarrow K(t+1) \rightarrow S(t+1)$

其中：

- **G**：人类、AI 或人机协作生成候选问题解决结构；

- **Sel**: 外部选择造成的差异化保留;
- **R**: 保留结构跨制品、项目、组织和时间的传播;
- **E**: 这些结构在后代系统训练或知识环境中的有效暴露;
- **D**: 通过 DSI 形成的初始结构倾向变化;
- **K(t+1)**: 后代模型可用知识与固化结构的变化;
- **S(t+1)**: 从改变后的 K 出发进行的后续结构生成。

可观察的下游结果是:

$D(t+1) \rightarrow \Delta P(t+1)$

其中, **DeltaP** 表示在指定任务分布、基准、智能体执行框架和推理配置下测得的代际性能变化。DeltaP 不是传递机制本身; 它可能同时受到 D、训练算法、验证器、推理计算、评测条件和任务构成变化的影响。

因此, ES-DSI 的基本时间单元不是一次训练, 而是:

模型生成 $\rightarrow$ 现实生产与选择 $\rightarrow$ 后续模型

一个时期需要经过高成本结构搜索 (S) 才能发现的结构, 在经历选择、传播、暴露与 DSI 后, 可以成为后代模型 K 中概率更高的初始候选。后代模型由此可能从“重新发现结构”转向“识别成熟结构并进行局部适配”, 把 S 留给任务中仍未解决的部分。

2.7 一般生态继承与 AI 自反馈

ES-DSI 并不限于 AI 生成结构。人类产生的代码、算法、理论、设计和工程方法同样可以经过这一过程, 成为后代模型的结构经验。因此, 完整机制首先描述的是开放的人类—AI 环境中的结构继承。

其中, AI 自反馈只是范围更窄的子过程:

AI 或人机生成 $\rightarrow$ ES $\rightarrow$ R $\rightarrow$ E $\rightarrow$ 后代 AI 中的 DSI

要把后续能力变化归因于这一反馈, 必须识别胜出结构的来源和新颖性、传播与训练暴露路径, 并建立相关反事实: 如果没有该 AI 生成贡献, 后代模型是否仍会通过既有人类知识获得同一结构? 如果无法区分来源, 现有证据只能支持一般生态继承, 不能单独估计 AI 自反馈的贡献大小。

2.8 适用条件与证据层级

该机制不意味着能力必然单调增长。外部选择可能较弱或方向错误, 有效结构可能无法传播, 后代训练可能没有接触这些结构, 缺乏有效选择的递归生成数据也可能造成模型质量下降。[15]

因此, 本文将论证划分为四个证据层级:

1. **直接性能观察**: 在固定模型规模或基础模型的条件仍可能出现较大的 DeltaP。这证明规模不扩张时仍存在能力增长空间, 但不证明增长原因。
2. **与 DSI 一致的受控证据**: 在一定程度上控制训练条件后, 任务构造、执行环境和交互轨迹仍会影响泛化。这缩小了解释范围, 但没有直接测量 D。
3. **组成环节的现实可行性**: 软件工程中已经分别出现 G、Sel、R 与 E 的可观察实例。各环节分别存在, 不能替代端到端因果链。
4. **贡献假说**: ES-DSI 可能已经成为前沿编程与智能体能力增长的主要驱动机制之一。验证这一判断, 需要追踪同一批结构的完整链条, 并将其贡献与规模扩展、训练算法、验证器和推理计算分解开来。

在这些边界内, 本文的中心机制主张是: ES-DSI 已经足够独立、可观察并可证伪, 应作为一种社会级 AI 跨代进化机制单独建模和检验。关于其当前贡献大小的更强判断仍是研究假说。

3. 观察方法: 逐层控制的同系列代际比较

直接比较两个模型版本无法进行可靠归因。一次模型升级可能同时改变:

- 参数和激活规模;
- 架构;
- 预训练数据与计算;
- 后训练数据;
- 强化学习方法;

- 验证器；
- 优化器；
- 推理预算。

因此，本文依次加强控制条件。

L0：同模型系列

减少产品定位和目标变化带来的干扰。

L1：参数规模相同

若显著增益仍存在，则参数增加不是该增益的必要条件。

L2：架构相同

进一步排除架构替换作为必要条件。

L3：基础模型相同

若能力继续明显提高，则重新训练一个更强的基础模型也不是必要条件。

L4：模型与部分训练条件固定

利用开放研究继续控制模型、训练流程、训练预算或数据规模，比较不同任务构造、执行环境和问题解决轨迹带来的差异。

这一方法不等于随机控制实验，也不意味着每一层都完成了严格因果隔离。特别是在 L3 条件下，后训练仍可能同时改变数据、强化学习、验证器、采样和优化细节。其目标更为有限：

观察常见能力增长变量被逐层控制以后，仍有多少性能增量能够保留下来，并进一步缩小可解释变量的范围。

本文同时区分两类基准证据：

1. **同一模型系列内部的同源纵向评测**，用于计算版本差值；
2. **DataCurve 等独立统一评测**，用于交叉验证最终模型能力，并为若干同期前沿模型相邻版本的变化提供统一数量级参照。

不同来源的纵向差值只用于比较增量数量级，不进行严格的跨来源统计比较。当前 12 组比较构成描述性样本；本文尚未给出足以定义“典型前沿版本增量”的完整纳入规则与统计分布，因此不以这些样本估计前沿模型版本迭代的总体分布。

4. 证据一：固定模型规模下的大幅能力跃迁

本文配套描述性数据集目前包含 21 个模型或版本以及 12 组同系列相邻比较。

该数据集在本文中用于描述性比较，而非估计“前沿模型典型版本增量”的统计分布。未来公开版本应同时提供完整样本、纳入与排除规则、版本日期、基准版本、智能体执行框架（agent harness）、推理强度（reasoning effort）等可获得的元数据，以提高纵向比较的可复现性。

无法验证的参数、训练方法和数据来源统一标记为 Unknown；闭源模型的 API 价格只作为产品档位和推理成本的辅助信息，不作为参数量的替代指标。

部分代表性结果如下：

模型变化	DeepSWE 增量	主要控制条件	评测来源
Gemini 3.5 Flash → 3.6 Flash	+11.0 个百分点	同系列、同 high 配置	DataCurve 统一评测 [2]
Gemini 3.6 Flash → 3.7 Flash	+18.0 个百分点	同系列、同 high 配置	DataCurve 统一评测 [2]
Claude Opus 4.8 → Opus 5	+15.0 个百分点	同产品档位、同 max 配置	DataCurve 统一评测 [2]
GLM-5.1 → GLM-5.2	+28.2 个百分点	同公开参数规模	Z.ai 官方纵向评测 [17]
GLM-5.2 → GLM-5.3	+20.7 个百分点	相同基础模型	Z.ai 官方同源评测 [5]
DeepSeek V4 Flash Preview → 0731	+47.1 个百分点	相同架构和尺寸，仅重新后训练	DeepSeek 官方同源评测 [3][4]

注：不同模型系列的增量并非全部来自同一评测实现。表中每一行内部具有纵向可比性；跨行差值只用于说明增量数量级，不构成严格的横向基准比较。

因此，本节使用“与若干前沿模型相邻版本同量级”描述这些增量，而不将 11-18 个百分点定义为统计意义上的“典型前沿版本增量”。

4.1 DeepSeek V4 Flash

DeepSeek 官方更新说明指出，V4-Flash-0731 与 Preview 保持相同模型结构和尺寸，并重新进行了后训练。[3]

其官方模型卡中的同源 DeepSWE 结果为：

7.3 → 54.4 [4]

同时，Terminal Bench、CyberGym 等其他工程相关指标也同步提高。[4]

DataCurve 对正式版 V4 Flash 的独立测试结果约为 53%，与厂商报告的 54.4 基本一致，可作为正式版能力的独立交叉验证；但 Preview 的 7.3 来自厂商评测，因此完整的 47.1 个百分点纵向增量仍应视为**厂商同源测试结果**，而不是独立基准对整条变化轨迹的复现。[2][4]

还需要保留一项评测配置限制：公开模型卡说明，0731 的 Code Agent 评测使用 DeepSeek Harness 的 minimal mode 与 max reasoning effort，但公开资料并未同样明确说明 Preview 的 7.3 是否在完全相同的智能体执行配置下重新评测。[4] 因此，这一纵向结果构成较强的厂商同源证据，但尚不能视为外部研究者在完全锁定执行框架与推理配置后的严格复现。

这一案例支持两项关于必要条件的较强判断：

更大的参数规模和新的模型架构，并不是该次巨大能力跃迁的必要条件。

但公开资料不足以进一步将后训练变化在数据、强化学习、奖励设计和优化细节之间进行分解。

4.2 GLM-5.2 → GLM-5.3

GLM-5.3 提供了控制程度更高的固定基础模型案例。

Z.ai 明确说明，GLM-5.3 使用与 GLM-5.2 相同的基础模型，改进来自后训练。[5]

其官方 DeepSWE v1.1：

46.2 → 66.9 [5]

公开说明还显示，GLM-5.3 延续了 5.2 已经采用的核心强化学习策略，重点扩展了更真实、更复杂的软件工程环境、自动验证器、求解轨迹与训练任务。[5]

因此：

重新训练一个新的基础模型，也不是获得这一量级的软件工程能力增益的必要条件。

GLM-5.1 与 GLM-5.2 均公开为总参数量 744B、激活参数量 40B，但 5.2 同时引入了架构和训练变化，因此这一组只能用于排除“参数数量增长是必要条件”，不能用于排除架构变化。[17]

5. 证据二：训练数据构造与交互轨迹结构具有高杠杆效应

商业模型的公开资料只能将变化范围缩小到后训练阶段；开放研究则允许进一步控制训练过程。

5.1 SWE-Lego

SWE-Lego 在同一个 Qwen3-32B 基础模型上，以 SWE-bench Verified 为主要软件工程评测，报告：

- 基础成绩：23.2%；
- 新的软件工程数据：+25.6 个百分点；
- 改进后的监督微调：额外 +3.8 个百分点；
- 测试时扩展：额外 +6.2 个百分点。[6]

这一结果不能外推为所有前沿模型中数据与算法贡献的固定比例。

它能够支持的是：

在固定基础模型上，训练经验变化本身已经能够产生数十个百分点的软件工程能力增量，而不要
求同量级的训练方法创新。

5.2 Terminal-Lego

Terminal-Lego 进一步固定学生模型和训练流程，仅改变用于训练的问题解决轨迹。[7]

研究发现，教师模型自身的基准成绩并不能直接预测其轨迹的训练价值。其中一个重要差异来自轨迹中的环境交互结构：更频繁呈现 inspect -> act -> verify（检查—行动—验证）模式的训练轨迹，能够带来更好的后续终端任务表现。[7]

因此，训练数据的价值不仅取决于任务最终是否成功，也与问题解决过程中暴露了哪些环境状态、操作关系和验证过程有关。

这为 DSI 提供的是**间接支持**：不同结构分布对应不同的后续行为和泛化能力，但研究尚未直接测量某一可迁移结构初始生成概率的变化。

5.3 数据规模不能单独解释结果

OpenThoughts-Agent 进行了大量受控数据实验，并在控制计算量的不同训练集规模比较中，持续观察到不同数据构造方式产生的性能差异。[8]

Patches-to-Trajectories 以 SWE-bench Verified 等任务进行实验，并通过等规模消融和等计算量分析表明，轨迹中间过程的质量不能被单纯轨迹数量或额外训练计算完全解释。[9]

这些结果共同说明：

训练规模与训练数据/轨迹的构造方式不能被视为同一个变量。

相同或相近规模的数据，因为包含不同的问题表示、观察—行动关系、失败路径与验证过程，可以产生明显不同的泛化效果。按照第 2 章的定义，这些结果与“训练中的结构分布会影响后续结构生成倾向”的 DSI 预测一致；但这种对应关系仍属于本文的理论映射，现有研究并未直接测量某一可迁移结构初始生成概率的变化。

6. 证据三：AI 参与的问题解决产出已经达到可测规模

ES-DSI 首先要求前代 AI 持续参与新的问题解决，并产生足够规模的数字化结果。

软件工程已经提供直接的规模代理证据。

Khosravani 与 Mockus 对超过 1.8 亿个 Git 仓库进行了多信号检测。研究发现，到 2026 年 4 月，可明确归因于编程智能体的提交已经超过：

每月 32 万次。

其中 Claude Code 累计识别：

- 886,122 次提交；
- 17,295 个公共项目。[10]

由于大量 AI 辅助开发行为不会表现为独立机器人身份，这些数字更接近可检测活动的下界。[10]

AIDev 数据集则收集了：

932,791 个由智能体产生的 PR（Agentic PR），覆盖 116,211 个代码仓库。[11]

这些结果直接测量的是 AI 参与工程产出的规模，而不是其中“新问题解决结构”的数量。

因此，对于第 2 章定义的 G，更准确的判断是：

当前已经具有较强的规模代理证据，但结构新颖性和结构价值仍缺少直接测量。

7. 证据四：ES、传播与后续训练暴露已有现实实例

大规模生成只是 ES-DSI 的输入条件。下一步需要观察候选结构是否经历差异化保留、传播和再次训练。

7.1 外部选择

软件工程具有多层现实选择机制：

- 编译；
- 自动测试；
- CI；
- 基准测试；
- 代码审查；
- 人类验收；
- 生产运行；
- 用户反馈；
- 缺陷修复和回滚。

因此，AI 参与产生的工程结果不会无差别地长期保留。

Ogenrwot 与 Businge 分析了 24,014 个已经合并的 Agentic PR，共涉及 440,295 次提交。[12]

这一数据直接说明：

一部分有 AI 参与的工程修改已经获得项目级接受，并进入长期代码历史。

但“被合并”仍不能完全等同于“高价值结构被正确选择”。

因此，现有证据较强地支持 **ES 的存在性**，但对外部选择的强度、长期质量与结构偏好仍缺少系统量化。

7.2 传播

通过选择的结构可以继续进入代码库、测试、API、软件库、框架、文档、工作流与后续项目，由此产生跨时间、跨项目甚至跨组织传播。

目前公共研究主要能够统计智能体提交、PR 和项目采用规模，对某个特定高价值结构被复制、派生和长期传播的倍率仍缺少系统追踪。

因此：

传播通道已经存在，但 R 仍是当前证据链中明显较弱的一环。

7.3 后续训练暴露

GLM-5.3 公开说明，其后训练扩展了新的长程软件工程任务、执行环境、自动验证器和求解轨迹。[5]

SWE-Lego、Terminal-Lego、OpenThoughts-Agent 等开放研究也直接展示了代码仓库、执行环境和交互轨迹如何被转化为后训练输入。[6][7][8]

因此，**新工程结构重新成为后训练输入**已经存在直接实例。

但需要保持一个重要边界：

公共代码中存在 AI 生成或 AI 参与生成的结构，并不意味着这些结构已经进入任何特定商业模型的训练集。

所以 E 当前具有**后续训练暴露的直接实例**，但尚未与前述 G、Sel 样本建立同源的完整时间链。

8. ES-DSI 链条的证据映射

第 2 章已经将传递机制分解为 G、Sel、R、E 和 D，并将 DeltaP 区分为下游性能结果。当前公共证据可以映射为：

变量	当前软件工程证据	证据状态
G	编程智能体提交、Agentic PR	规模代理证据较强；结构新颖性尚未直接测量
Sel	已合并 PR、测试、CI、代码审查、生产反馈	ES 的存在性有直接证据；选择强度与质量偏好尚未量化
R	代码、软件库、API、工具和文档传播	通道存在，尚缺系统量化
E	新工程环境、任务和轨迹进入后训练	存在直接实例，但尚未与 G、Sel 建立同源链路

变量	当前软件工程证据	证据状态
D	不同训练轨迹形成不同后续泛化行为	间接证据，尚缺初始结构倾向的直接测量
DeltaP	固定模型规模或固定基础模型下显著的软件工程性能变化	直接性能证据较强

目前尚无针对某一前沿商业模型的全链因果记录能够完整追踪：

G -> Sel -> R -> E -> D

并进一步在受控评测条件下检验 D 与 DeltaP 之间的因果联系。多个关键环节已经分别出现可观察的现实证据，但这些证据证明的是各环节的存在性和机制可行性，不能直接拼接为同一条已经闭合的因果链。

9. 尚未闭合的因果环节与解释边界

现有公开信息尚不能将观测到的能力增量在规模扩展、训练算法、验证器、推理计算和 ES-DSI 之间进行可靠的贡献率分解。

因此，本文不能判断 DeepSeek、GLM 或其他具体模型某次性能提升中有多少比例来自 ES-DSI。

递归使用模型生成数据本身也可能造成分布退化甚至模型崩溃。[15]

所以：

有效外部选择是 ES-DSI 形成正向累积的必要条件，而不是可以省略的环节。

本文目前建立的是关于现象存在性、增益幅度与机制可行性的公共证据，而不是完整的单模型全链因果证明。

10. 为什么现有证据不仅仅是“后训练有效”

“后训练可以让模型变强”并不是本文的新发现。当前证据把问题进一步推进了三步。

第一，模型规模固定后的增益可以很大。

在参数规模、架构甚至基础模型受到不同程度控制后，仍可以出现数十个百分点的软件工程基准变化，并达到若干同期前沿模型相邻版本变化的同一数量级。

第二，观察到的是未见任务上的泛化表现。

相关评测包括长程软件工程和终端任务，因此讨论对象并不是训练样本的简单记忆或复现。

第三，训练经验不能只以数量衡量。

受控研究表明，在相同模型和相近训练预算下，不同数据构造、环境交互和轨迹结构可以形成显著不同的后续能力。

这三类观察仍然不能直接证明固定模型规模下的性能增量来自 ES-DSI。它们建立的是一个更有限但重要的经验基础：模型规模扩张不是大幅能力增长的必要条件；训练经验的构造方式具有高杠杆；而现实社会中 ES-DSI 所需的生成、选择、传播与后续暴露环节也已经分别出现。由此，ES-DSI 从纯理论可能性推进为一个可以进行同源链追踪和贡献率检验的机制假说。

11. 讨论：ES-DSI 作为社会级 AI 跨代进化机制

第 2 章已经完整定义机制，第 4 至第 10 章则建立了当前证据基础及其边界。本节进一步展开本文的中心机制主张：

ES-DSI 应作为一种有别于规模扩展、训练算法创新和封闭式自训练的社会级 AI 跨代进化机制，单独加以建模和检验。

本文所谓**社会级**，是指结构生成、选择和传播分布于大量独立的人类组织、真实任务与数字系统中，而非局限于单一模型训练闭环内部。

11.1 与跨代积累研究的关系

已有研究已经讨论过 AI 的跨代积累。Cook 等提出 Artificial Generational Intelligence，在受控强化学习环境中研究个体探索、社会学习和跨代信息传递，并展示了人工智能体实现文化积累的可能性。[13]

更接近本文问题的是 Ferbach 等对递归使用自身生成数据的模型（self-consuming generative models）的研究：作者分析经过用户或奖励模型筛选的数据如何进入迭代再训练，并证明在其理论设定下，数据筛选（curation）可以形成隐式偏好优化，同时也可能放大选择函数本身的偏差。[18]

因此，本文并不主张“AI 可以发生跨代积累”或“筛选后的模型生成数据可以影响后代模型”这些概念由本文首次提出。

ES-DSI 的区别在于：

它关注开放的现实生产环境中由 AI 与人类共同参与的结构生成、独立于模型训练目标的现实外部选择、跨项目和跨组织的数字结构传播，以及这些结构在后续通用模型训练中的再次暴露。

也就是说，ES-DSI 不是一个封闭的迭代再训练协议，而是试图描述现实中的数字社会如何成为跨模型代际的结构选择与积累环境。

11.2 全社会共同支付结构搜索与选择成本

新的问题解决结构并不只由模型实验室产生。

现实社会持续开展软件开发、科学研究、工程设计、产品决策、数据分析，并不断创造新工具和新 workflow。这些活动本身就需要寻找新的问题表示、变量关系、解决路径和验证方式。

也就是说：

现实生产活动本身就在持续支付结构搜索成本。

随着 AI 深度参与这些活动，越来越多搜索结果会以代码、论文、测试、设计、工具和其他数字形式被保存。

当胜出结构进入未来训练环境时，后续模型便可能利用：

整个社会付出搜索成本后获得、并经现实选择保留下来的问题解决结构。

这使 ES-DSI 与为改善模型而额外开展的训练实验具有明显不同的成本结构。

它不意味着所有结构都会公开共享。私有代码、封闭数据和组织内部经验仍然存在。更准确地说：

结构生成和选择发生于整个数字化社会，并具有显著的跨组织外溢潜力。

11.3 与规模扩展的区别：扩大模型规模与积累结构经验

规模扩展通过增加参数、数据和训练计算推动模型能力增长。[16]

这种路径仍然重要，但增加训练规模需要额外资源，更大的部署模型通常也会带来显存、通信和推理计算方面的资源压力。

ES-DSI 关注的是另一个变量：

在模型规模不继续扩大的条件下，未来训练环境中哪些经过选择的问题解决结构被重复暴露，并由后续模型内化。

因此，两者可以近似区分为：规模扩展改变参数、数据与训练计算所提供的表示和优化资源；ES-DSI 则试图解释跨代结构经验如何改变既有模型规模下的学习分布与初始结构倾向。

本文观察到的固定模型规模跃迁只说明：能力增长并不要求部署模型规模同步增长。它本身并不能证明这些增益来自 ES-DSI。

这也不意味着 ES-DSI 没有训练成本，或实际智能体的总推理成本一定保持不变。更准确地说：

ES-DSI 所描述的能力增益路径不要求部署时的模型规模随能力增益同步扩大。

因此，如果该机制的贡献率最终得到验证，它将提供一种使能力增长与部署模型规模部分解耦的解释。

11.4 与自训练的区别：外部选择

自训练、合成数据和自博弈已经可以在一些具备可靠验证器的领域形成有效学习闭环。

例如，Self-Rewarding Language Models 允许模型生成自身训练样本，并利用模型本身作为评判器进行迭代优化。[14] Ferbach 等进一步表明，经过外部筛选的生成数据也可以在迭代再训练中改变后续模型分布。[18]

因此，ES-DSI 与自训练或经筛选的迭代再训练之间，区别都不在于“是否使用 AI 生成数据”。

关键区别在于：

决定哪些结构长期胜出的主要选择信号，是否来自模型训练闭环之外。

典型自训练可以近似表示为：

模型生成 -> 内部奖励 / 验证器 -> 再训练

而 ES-DSI 进一步包含：

模型与人类生成 -> 真实世界外部选择 -> 传播 -> 后续训练

一个工程结构能否长期存在，不只由训练系统判断其是否正确，还取决于系统是否实际运行、人类是否维护、用户是否采用、是否发生故障、是否被回滚以及是否被其他项目继续复用。

因此：

ES-DSI 的关键不是 AI 自我复制，而是经过独立的现实选择后实现的跨代继承。

11.5 与训练算法创新的区别：离散突破与持续积累

新的架构、优化器、强化学习算法和推理方法，通常需要经过方法设计、训练实验、失败探索、工程实现和规模验证。

这类能力增长通常表现为离散创新，其出现时间和收益幅度难以预先确定。

ES-DSI 的输入则来自现实世界持续开展的问题解决活动。只要 AI 持续参与新的真实任务、外部选择保持有效、胜出结构能够保存和传播，并被后续模型再次接触，新的可继承结构就可以持续产生。

因此，训练算法进步往往以新的方法、实现或训练配置变化出现；ES-DSI 则在理论上允许现实问题解决持续提供新的可继承结构。二者并不互斥，也不能仅凭时间连续性进行贡献归因。

11.6 方向性累积与自增强

ES-DSI 并不保证模型能力在数学意义上单调上升。选择机制可能失效，错误结构也可能传播。

但如果现实外部选择总体提高了有效结构的存续概率，未来数字环境中的结构分布就会沿相关有效性方向持续重新加权。

随后：

更高 K -> 更强生成能力 -> 更复杂的新问题得到解决 -> 更多候选结构 -> 新一轮 ES

于是形成反馈。

这种“自动性”不意味着无人参与，而是：

它不要求任何单一主体以推动 AI 进化为目标；企业、研究者、开发者和用户各自为了现实目标解决问题，进化过程仍可以作为社会生产活动的副产物持续运行。

因此，在 ES 持续有效的条件下，ES-DSI 具有方向性累积和自增强倾向。

11.7 与其他能力增长机制的比较

机制	主要作用层级	核心新增资源	是否要求扩大模型容量	对推理资源的要求	主要选择信号	时间特征
规模扩展	单代模型	参数、训练计算、数据	通常相关	通常带来更高资源压力	训练目标	规模扩张，通常呈递减边际收益
训练算法创新	单代训练	新算法、实验与工程	否	不一定	人工设计目标 / 验证器	离散创新
自训练	单代或递归训练	模型生成数据	否	不一定	模型或训练系统内部反馈	依赖闭环质量
ES-DSI	跨模型代际与现实社会	现实问题解决中生成并胜出的结构	否	不要求随能力增益同步扩大模型容量	工程、人类、用户与现实环境	持续、跨组织、方向性累积

固定模型规模下的AI编程能力跃迁

这些机制并不互斥，当前前沿 AI 很可能同时从多个机制中获得能力增益。

ES-DSI 的独特之处在于：

其结构搜索和选择主体已经不再局限于模型实验室。

11.8 核心假说：ES-DSI 可能已经成为主要能力增长驱动机制之一

前述证据支持三类观察：第一，固定模型规模甚至固定基础模型后仍可出现大幅软件工程能力增益；第二，训练数据构造、执行环境与交互轨迹在受控研究中具有显著影响；第三，AI 结构生成、现实外部选择、数字传播与后续训练暴露已经分别出现可观察实例。

这三类证据尚未在同一批结构上闭合 $G \rightarrow Sel \rightarrow R \rightarrow E \rightarrow D$ 传递链，也未进一步建立 D 与 DeltaP 之间的因果联系。因此，它们不能直接推出 ES-DSI 已经解释了固定模型规模下的性能增量，更不能直接推出其贡献率已经超过其他机制。

这些观察共同为本文提出的核心假说提供了可以直接检验的现实基础：

ES-DSI 可能已经不是未来才会显著出现的辅助机制，而是现阶段前沿 AI 编程与智能体能力增长的主要驱动机制之一。

这是本文提出的核心假说；现有公共证据尚不足以证明这一判断。

当前直接经验基础集中于软件工程。因此，一个进一步但尚未得到本文直接支持的外推是：

ES-DSI 也可能成为总体前沿 AI 能力增长的重要驱动机制。

这一更广泛判断需要数学、科学推理、开放式研究等领域的独立复现。

对“主要驱动机制”假说的严格验证，最终需要比较：

Contribution_ES-DSI

与：

Contribution_Scaling

Contribution_TrainingAlgorithm

Contribution_Inference

之间的实际数量关系。

如果未来研究发现，在控制其他变量后 ES-DSI 只能解释极小部分能力变化，该假说应被削弱。反之，如果同源追踪能够证明经过 ES 的结构传播和训练暴露持续引起可测的初始结构倾向变化，并进一步解释固定模型规模下的大比例性能增量，则 ES-DSI 应被提升为理解 AI 能力进步的核心变量之一。

12. 可证伪预测与量化路线

ES-DSI 的研究价值在于，各环节既可分别测量，也可组合成一条能够整体检验的完整时间链。

12.1 G：来源与结构新颖性

选择一批能够前瞻性观察其产生过程的候选结构，记录：

- 结构由人类、AI 还是人机协作生成；
- 它对相关环境而言是真正的新结构，还是既有结构的重新表达；
- 其中哪些部分需要新的搜索，而不是直接检索；
- 如果没有 AI 参与，同一结构出现的反事实概率有多高。

单纯的输出规模并不足够。如果难以从 AI 生成制品中识别出新的问题解决结构，这些制品只能证明 AI 的参与规模，不能支持对 AI 跨代自反馈贡献的可靠估计。

12.2 Sel：外部选择贡献

固定模型、训练算法、数据规模和训练计算，比较：

未经现实选择的 AI 生成结构

与：

经测试、人类筛选、真实执行或生产反馈选择的结构。

如果 ES-DSI 的外部选择作用成立，后者应产生更稳定的跨任务泛化收益。

实验还应区分测试、生产存续、人类审查、用户采用和模型内部奖励等不同选择来源，因为 ES-DSI 预测选择信号的来源和质量会影响哪些结构长期保留。

12.3 R：结构传播倍率

跟踪经过 ES 的结构：

- 被复制多少次；
- 进入多少项目和组织；
- 是否进入软件库、框架、文档或标准实现；
- 是否形成派生版本；
- 存活时间多长。

这些指标可以估计胜出结构对后续数字环境的重加权程度。一个结构可能在局部成功，却因为传播倍率很低而几乎不产生跨代继承效应。

12.4 E：有效训练暴露

对已经传播的结构进一步测量：

- 是否进入后续训练或知识环境；
- 暴露次数及发生时间；
- 表达形式的多样性；
- 与竞争结构相比的相对权重；
- 所处训练阶段及对应目标。

E 是现实结构分布变化进入 DSI 的桥梁。公开可访问本身不能证明特定模型获得了有效训练暴露。

12.5 D：初始结构倾向变化

选择明确的问题解决结构，控制其：

- 暴露频率；
- 表达多样性；
- 上下文环境；
- 训练阶段。

随后测量面对新任务时：

该结构被自然生成或优先调用的概率。

测量应尽量发生在大规模工具调用、外部反馈和多轮修正以前，以捕捉 S 的起始分布，而不是最终任务成功率。这可以将目前通过基准间接推断的 DSI 推进到初始结构倾向的直接行为测量。

12.6 全链追踪与贡献率分解

最强研究设计应选择一批时间上真实新出现的问题解决结构，连续追踪：

G：首次生成与来源

```
-> Sel
-> R
-> E
-> D
```

随后检验测得的 D 是否产生下游性能变化：

D -> DeltaP

该设计应比较接触和未接触目标结构的后代模型或受控训练结果，并尽量固定模型规模、算法、验证器、推理配置和评测条件。只有到这一阶段，才能开始严格比较 ES-DSI、规模扩展、训练算法和推理计算的贡献率。

如果端到端追踪反复发现：经过外部选择和传播的结构很少进入后续训练，进入后也没有改变初始结构倾向，或在控制其他变量后只能解释极小的 DeltaP，那么“主要驱动机制”假说应被削弱。反之，如果同一结构可以被连续追踪，并解释固定模型规模下相当比例的能力增量，该假说将获得更强支持。

13. 简要产业含义

如果 ES-DSI 最终被证明对当前编程与智能体能力增长作出了重要贡献，那么模型公司的关键资产便不能只用训练算力和参数规模衡量。

能够持续产生、筛选、传播并重新吸收高价值结构的真实任务网络，也可能构成重要的能力增长基础设施。

固定模型规模下的显著增益还提示：模型进入一定规模区间以后，继续扩容与改善训练数据、环境和结构经验之间的最优资源配置，值得成为独立研究问题。

14. 结论

本文从一个经验现象出发：在参数规模、模型架构甚至基础模型固定的条件下，软件工程能力仍然可以获得数十个百分点的基准增益，并达到若干同期前沿模型相邻版本变化的同一数量级。

本文使用 KSC 框架中的 K—S 关系，完整说明了分析这一现象所需的结构单位与跨代机制。一个时期生成的可迁移问题解决结构，可能经过外部选择，在数字环境中传播，获得后续有效训练暴露，并通过 DSI 改变后代模型的初始结构倾向。在分析上，应将这一传递机制与其下游性能结果区分开来。

逐层控制的模型比较表明，参数和架构扩张不是这些增益的必要条件；受控研究进一步表明，训练数据构造、执行环境与交互轨迹会显著影响后续泛化；现实软件工程生态中，AI 生成、外部选择、数字传播和后续训练暴露等环节也已经分别出现可观察实例。

这些证据尚未将上述环节连接为某一具体前沿模型的完整同源因果链。因此，本文不把 DeepSeek、GLM 或其他模型的固定规模性能跃迁直接归因为 ES-DSI。

在这一边界下，本文提出：

ES-DSI 应作为一种有别于单纯规模扩展、训练算法创新和封闭式自训练的社会级 AI 跨代进化机制，单独加以建模和检验。

其关键不是 AI 无条件重新学习自己的输出，而是现实问题解决过程中产生的候选结构经过独立外部选择后，改变未来数字环境中的结构分布，并再次进入后续训练，从而提高其成为下一代模型初始候选的概率。

当前公共证据进一步使一个更强的假说值得直接检验：

ES-DSI 可能已经成为现阶段前沿 AI 编程与智能体能力增长的主要驱动机制之一。

这一判断尚未闭合同源因果链，也未完成贡献率量化，因而不能直接外推为总体 AI 能力增长的既成事实。

因此，下一阶段研究应分成两个连续问题：

1. **同源链闭合**：能否围绕真实新结构，连续建立 $G \rightarrow \text{Sel} \rightarrow R \rightarrow E \rightarrow D$ 各环节的时间与数量关系，并进一步检验 D 与 DeltaP 之间的因果联系？
2. **贡献率分解**：在控制规模扩展、训练算法、验证器和推理计算以后，ES-DSI 实际解释了多少能力增量？

只有完成第一步，第二步的“ES-DSI 已经贡献了多少”才具备严格的因果含义。如果这条链能够被稳定测量，AI 跨代进化就可以从宏观叙事进一步转化为一条能够与规模扩展、训练算法和推理计算并列进行贡献率分析的能力增长轴。

参考文献

[1] 刘明. KSC: 知识、结构与约束——人类—AI 差异与协同的功能框架. Zenodo, 2026. DOI: 10.5281/zenodo.21834579.

- [2] Huang, W., & Jiang, P. **DeepSWE v1.1: A Cleaner, More Reproducible Benchmark for Frontier Coding Agents**. DataCurve, 2026.
- [3] DeepSeek. **DeepSeek API Change Log: DeepSeek-V4-Flash Update**. 2026.
- [4] DeepSeek-AI. **DeepSeek-V4-Flash-0731 Model Card**. Hugging Face, 2026.
- [5] Z.ai. **GLM-5.3: Frontier Coding with Emergent Cyber Capabilities**. 2026.
- [6] Tao, C., Chen, J., Jiang, Y., et al. **SWE-Lego: Pushing the Limits of Supervised Fine-tuning for Software Issue Resolving**. arXiv:2601.01426, 2026.
- [7] Yang, S., Tao, C., Chen, J., et al. **What Makes Interaction Trajectories Effective for Training Terminal Agents?** arXiv:2606.03461, 2026.
- [8] Raoof, N., Zhuang, R., Nezhurina, M., et al. **OpenThoughts-Agent: Data Recipes for Agentic Models**. arXiv:2606.24855, 2026.
- [9] Ma, M., Chen, T., Lin, Y., et al. **From Patches to Trajectories: Privileged Process Supervision for Software-Engineering Agents**. arXiv:2605.21996, 2026.
- [10] Khosravani, A., & Mockus, A. **Detecting AI Coding Agents in Open Source: A Validated Multi-Method Census of 180 Million Repositories**. arXiv:2606.24429, 2026.
- [11] Li, H., Zhang, H., & Hassan, A. E. **AIDev: Studying AI Coding Agents on GitHub**. arXiv:2602.09185, 2026.
- [12] Ogenrwot, D., & Businge, J. **How AI Coding Agents Modify Code: A Large-Scale Study of GitHub Pull Requests**. arXiv:2601.17581, 2026.
- [13] Cook, J., Lu, C., Hughes, E., Leibo, J. Z., & Foerster, J. **Artificial Generational Intelligence: Cultural Accumulation in Reinforcement Learning**. arXiv:2406.00392, 2024.
- [14] Yuan, W., Pang, R. Y., Cho, K., et al. **Self-Rewarding Language Models**. arXiv:2401.10020, 2024.
- [15] Shumailov, I., Shumaylov, Z., Zhao, Y., et al. **AI Models Collapse When Trained on Recursively Generated Data**. Nature, 631, 755-759, 2024.
- [16] Kaplan, J., McCandlish, S., Henighan, T., et al. **Scaling Laws for Neural Language Models**. arXiv:2001.08361, 2020.
- [17] Z.ai. **GLM-5 Official Repository and Model Specifications**. 2026.
- [18] Ferbach, D., Bertrand, Q., Bose, A. J., & Gidel, G. **Self-Consuming Generative Models with Curated Data Provably Optimize Human Preferences**. Advances in Neural Information Processing Systems, 37, 2024. arXiv:2407.09499.