预训练 LLM 评估：调研者综述

三本头研究报告 · Part: researcher

本报告截稿日期: 2026-05-25　　共 8 章

01 · 评估理论基础：accuracy → IRT → 对抗 eval

1.1 为什么要先谈"评估理论"

在投入到 MMLU 错题率、HellaSwag 饱和、GSM8K 污染等具体 benchmark 的批判性讨论之前，本章先把 LLM 评测放回 NLP/ML 评估方法论的长河里。理由有二。其一，过去十年 NLP 评测的主流变迁——从 string-match metric，到 item-level 量化建模 (IRT)，再到 adversarial filtering 与 dynamic benchmark——并非由 LLM 单独驱动，而是测量理论 (measurement theory) 与 ML 实证传统的交汇；不了解这一脉络，Elazar 等对 WinoGrande "artifact" 的指控、Chizhov 等对 HellaSwag 的 "Lorem ipsum" 检验、Schwartz 等对 Story Cloze 的 endings-only baseline 等 critique 会显得零散。其二，本报告后续章节系统综述具体 benchmark 的设计动机与争议，需要一个统一的"信息量"框架——即 benchmark 到底测出了多少有用信号——来比较不同 critique 的力度。本章给出这套理论工具，作为 §03–§07 的分析基线。

主线有三：accuracy / F1 等"原子级"指标的失败、Item Response Theory 在 NLP 评估中的复兴、以及 adversarial / contrastive eval 作为 distribution-aware 反制的兴起。最后我们落到 framework 设计哲学，对照 DCLM 的"benchmark 绑数据"、OLMES 的"reproducibility 论"和 HELM 的"holistic scenarios"三种 community-scale 范式，引出衡量 benchmark 信息量的 meta-evaluation 工具——以 BHI [^bhi-2026] 为代表。

1.2 第一性原理：accuracy 是什么、不是什么

accuracy 的统计含义清晰：在固定测试集 $\mathcal{D}={(x_i, y_i)}_{i=1}^n$ 上，模型 f 的 accuracy 是 $\frac{1}{n}\sum_i \mathbb{1}[f(x_i)=y_i]$，对应二项均值估计；95% Wilson 置信区间在 $n=500$、observed $p=0.5$ 时约 ±4.4 pp，在 $n=10000$ 时约 ±1 pp。这给出 LLM 评测"题数门槛"的基本算术：OpenBookQA 仅 500 题，单一模型分数的统计噪声边界几乎与"7B vs 13B"真实差异同量级。这也是为什么 frontier 模型作者在 model card 中倾向于报告 ≥3000 题 benchmark，且在 small benchmark 上以 multi-seed 或 bootstrap 给区间——PIQA valid 2000 题正是"小池子，大噪声"代表。

然而 accuracy 的统计干净掩盖了三类深刻问题，是过去十年 NLP/ML 评估批判的核心。

1 / 58

第一类：题目难度不均的隐含权重。 accuracy 把"全班最难的物理化学题"与"小学常识题"等权 1 计入。当 benchmark 内题目难度分布不均时，两个 accuracy 相同的模型在能力空间上可能完全不同：一个解出 60% 的简单题但所有难题都错，另一个 30% 简单题与 30% 难题均散落。前者在 production 难题上几乎全军覆没，后者则可能在 long-tail query 上反而稳健。MMLU 的 57 个 subdomain 分别报告、HELM v1 的 16 core scenario 加权，部分目的就是把这种"难度异质性"显式化；但只要 within-subdomain 仍按 simple mean 聚合，问题就没有消失。

第二类：评分协议的非平凡选择。 MCQ 至少有四种协议给出实质不同的数字：(a) loglikelihood ranking——把每个选项作为 continuation 算 $\log P$ 取 max；(b) byte-length normalization；(c) character-length normalization；(d) unconditional normalization——减去 prior $\log P(\text{option})$；以及 (e) generation-based answer extraction。EleutherAI 自家博客明确承认，**同一模型在同一 MMLU 题集上，不同 normalization 给出 5–15 pp 不同的分数** [^eleuther-mcq-norm]。换言之，**"GPT-4 在 MMLU 上 86%"是不完整命题**：不附 normalization / few-shot prompt / chat-template / answer extractor，数字不可比。OLMES paper [^gu2024olmes] 把这正式 codify 为 "canonical setup" 的标准化诉求并进入 NAACL 2025 Findings；其暗含批评对象——lm-evaluation-harness "presents task setups as open choices"——正是 community 主线工具的核心张力。

第三类：题目本身的标注噪声 / 构造伪信号。 accuracy 把每道题视作可信 ground truth。但 MMLU-Redux 重标 5,700 题后发现 6.49% 含错，Virology 子集错题率 57% [^gema2024mmlu-redux]；Surge AI 对 HellaSwag 的 audit 估计 36% 含错，ActivityNet 子集非语法或荒谬选项达 95% [^surge2024hellabad]；Mousavi 等对 SocialIQa 的精细审计指出超过 28% 样本含结构 / 语义 / 语用缺陷 [^mousavi2025garbage]。这些数字直接侵蚀 accuracy 的"信号"含义：当 benchmark 的"测量上限"被错题压在某个数值之下，frontier 模型继续 hill-climb 本质上是在 fit noise；不同模型在错题上的"恰好答对率"与其能力无关，造成 ranking 抖动。

F1 与 EM (exact match) 在生成任务上各自具有同构问题：F1 把 token-level overlap 当 surrogate 信号，但对同义改写、word order 自由度、答案 span boundary 容忍度都需要任务级 normalization；EM 在"是否找到唯一正确字符串"上敏感于 tokenization 与小写化。LAMBADA 的两套变体（Paperno 原版 vs OpenAI re-tokenized 版本）之所以"在同一模型上给出不一样的分数"，本质就是 EM 协议对 tokenization 的零容忍 [^paperno2016lambada]。

1.3 IRT 进入 NLP 评估：从均值到 item-level 推理

测量心理学 (psychometrics) 自 1960 年代起就发展了一套远比 accuracy 精细的工具，称为 Item Response Theory (IRT)。其最基础的双参数 logistic (2PL) 模型把"被试 s 在题 i 上答对"的概率参数化为：

$$ P(\text{correct}_{i,s} = 1 \mid \theta_s, a_i, b_i) = \sigma(a_i (\theta_s - b_i)), $$

其中 θ_s 是被试 ability、b_i 是题目 difficulty、a_i 是题目 discrimination（判别力）。这个模型一次性给出 accuracy 完全做不到的三件事：(i) 在 item-level 区分"难

题"与"易题"；(ii) 把 ability 估计在一个有数值意义的 latent scale 上（不再受测试集组成绑架）；(iii) 量化每道题"区分能力的力度"，从而识别"对一切被试都几乎一样的题"——这种题对 ranking 无贡献，应被剔除。

Lalor 等 2016/2019 在 EMNLP / ACL 系列工作 [^lalor2016irt] 把 IRT 引入到 SNLI / SQuAD 等 NLP 任务上：他们用 IRT fit 出题目的难度与判别力，进而构造 "small but informative" 测试子集——一种 benchmark 精简方法，与单纯 random subsample 相比能在 1/10 题量下达到几乎相同的模型 ranking 一致性。Vania 等 2021 进一步把 IRT 用到 cross-benchmark 比较：把 GLUE / SuperGLUE 多个任务的题目放进同一 IRT scale 后发现，**任务"难度"在 model space 上不是单调单维的**——某些任务（如 RTE）对小模型很难、对中模型却变易，对超大模型反而又因为 trick prompt 重新变难，提示能力分布不是线性 latent variable [^vania2021irt]。这一观察后来被 Schaeffer 等 2023 关于 "emergent abilities are a mirage" 的工作 [^schaeffer2023emergent] 间接呼应——他们的核心论点是 emergent ability 的 sharp jump 来自不连续 metric (如 EM) 的选择，换成 token-level cross-entropy 等连续 metric 后曲线变光滑。Schaeffer 的批评虽然主要靶子是 metric 而非 IRT，但其底层逻辑——"模型能力是 latent variable，accuracy 是 noisy projection"——恰恰是 IRT 的世界观。

进入 LLM 时代，IRT 应用并未完全主流化，但其遗产以两条隐线渗透：其一是 benchmark 内 subdomain accuracy 的分别报告，本质上是"承认 ability 是 multi-dimensional"；其二是 Benchmark Health Index (BHI) [^bhi-2026] 把 IRT 的 discrimination 概念抽象为 "Capability Discrimination" 维度——一个 benchmark 在 91-model 分布上的 score variance / IQR，被显式 score 化、归一化，与 anti-saturation 和 impact 一起构成 benchmark-level 健康度三轴。BHI 的方法学贡献在于：**它第一次把"benchmark 自身是否值得用"作为 first-class research question 提出**，而不是把 benchmark 视作不可质疑的标尺。其局限——score 依赖 model 分布、impact 用 citation 引入 age bias——也提醒研究者：meta-evaluation 工具本身仍处早期阶段，不能机械套用。

1.4 Adversarial filtering 与 contrastive design：把 distribution shift 纳入测量

IRT 处理的是"已构造好的 item pool 上如何精算 ability"。但 NLP 评测的另一系列问题——annotation artifacts、surface statistical shortcut、reporting bias——发生在 item pool 构造阶段：测试题本身可能允许模型走 lexical / 句法捷径 fake 出能力。这正是 adversarial / contrastive 评测设计的出发点。

Zellers 等 2018-2019 在 SWAG → HellaSwag 的演进 [^zellers2019hellaswag] 是这一思路的范式案例：他们用一个判别器 (BERT 自己) 反复筛选"对模型显然但对人类不显然"的候选续写，把统计共现的 trivial shortcut 主动剔除。HellaSwag 因此在 2019 把当时 SOTA 从 SWAG 的 86% 拉回到 48%，给 BERT 时代模型留下大量提升空间。Sakaguchi 等 WinoGrande [^sakaguchi2019winogrande] 用更轻量的 AfLite (Adversarial Filtering Lite) 把 WSC 扩到 44k 并剔除 embedding 可解释 token；PIQA [^bisk2020piqa] 设计 twin sentences 仅差一两 token 的方式，让 lexical overlap baseline 失灵。这一波工作的共同 epistemic claim 是：**evaluation 不应该假设 distribution 是固定的；模型可以在 same**

distribution 上学到 spurious correlation，所以 benchmark 必须主动从 model 角度做对抗筛选。

然而 adversarial filtering 自身埋下三类隐患，到 2024–2025 集中爆发：

第一，**filter model 的视野有限**：HellaSwag 的 BERT-based filter 在 BERT-era 是有效对抗，但对 5 年后的 GPT-4 / Claude 已不再是 "adversarial"。Chizhov 等 2025 [^chizhov2025goldenswag] 的 "Lorem ipsum" 实验直击这一点——他们发现 65% 以上的模型预测在删除 question / 用 Lorem ipsum 占位符替换后仍不变，说明现代 LLM 已绕过原始 AF 防御，靠 answer-only surface cue 即可作答；他们随后释放 GoldenSwag 子集作为修正。这是一个非常重要的负面 lesson：**adversarial filtering 的对抗强度由 filter model 上限决定，对未来 frontier 模型不保证转移**。

第二，**对抗 filter 自身引入新 artifact**：Elazar 等 EMNLP 2021 [^elazar2021artifact] 的 "Back to Square One" 论文系统证明，WinoGrande 等数据集"看似消除了 WSC 的 artifact"但 AfLite 反而留下另一套 artifact——zero-shot 严格 evaluation 下，大多数 LM 在 WinoGrande 上接近随机，所谓的"进步"主要来自 fine-tuning 对新 artifact 的利用，而非真实常识。WinoWhat [^winowhat2025] 进一步用 paraphrase 复测：保持 schema 推理结构不变、改写 surface form 后，LLM 表现显著退化，说明 surface cue 仍是主要信号。

第三，**filter 引入对人类的偏置**：Schwartz 等 CoNLL 2017 [^schwartz2017stylistic] 在 Story Cloze 上发现，人写的 "right ending" 在风格上倾向更短、更轻量，仅靠 endings-only classifier 即可达 ~72%（random=50%）。这并非 filter 算法的失误，而是众包 annotator 在写正例与负例时的隐性风格分歧——AF 无法解决这种 annotator-level distribution shift。

这套 negative lesson 累积下来，逐步指向更新一代 evaluation design 哲学：**与其 once-for-all 对抗 filter，不如让 benchmark 本身 dynamic**。LiveBench [^white2024livebench] 选择月更，每月从 6 个月内新材料生成 ~50 题，让 contamination 与 distribution shift 都成为时间维度上的对抗；MixEval [^ni2024mixeval] 用真实 user query 加权挑题，把 distribution 从 benchmark 静态对齐到 wild query 真实分布。这两条路线代表了 2024–2026 的两种主流"distribution-aware"对抗范式——并直接进入 §05 的 contamination / saturation 学术辩论。

1.5 Framework 哲学：测量理论如何落到 community-scale 基础设施

抽象的测量理论最终需要 community-scale infrastructure 才能影响 community-wide 数字。lm-evaluation-harness、HELM、OLMES、DCLM、BabyLM 在 framework 层面给出了几条互有张力的设计哲学，值得对比。

lm-evaluation-harness [^biderman2024lessons] 选择"工具中性"路线：抽象 task 为 loglikelihood / generate_until / multiple_choice 四类 request，setup 选择交给研究者。它的优势是覆盖广（v0.4.12 含 60+ task）、社区贡献活跃；其代价是 OLMES paper 直白点出的 "presents task setups as open choices"——同一 task 在不同论文给出不同数字。HuggingFace Open LLM Leaderboard 2025-03 退休公告 [^hf-leaderboard-retirement] 把这

一张力提到了 community 自省层面：HF 团队自承"leaderboard could encourage people to hill climb irrelevant directions in the field"，承认 static benchmark mean-score 的范式正在失去意义。

HELM [^liang2022helm] 走"holistic 评分卡"路线：scenario × metric 二维矩阵强制完整覆盖，把 calibration / robustness / fairness / bias / toxicity / efficiency 七维与 accuracy 并列报告。HELM v1 暴露的 aggregation 缺陷（mean-win-rate 依赖 model 集）在 HELM v2 / Capabilities [^crfm2025helm-capabilities] 自我修正为 mean-score。CRFM blog 在 v2 launch 时坦言 mean-win-rate "dependent on the set of models being compared, and sensitive to small variations"——这是 framework 维护者自身的方法学自省。然而 HELM v2 缩窄到 5 个 scenario，与"holistic"原意构成讽刺。HELM 在 2026-06-01 进入 maintenance mode 也标志着这一路线的活力下降。

OLMES [^gu2024olmes] 站在 lm-eval-harness 肩膀上做"canonical setup"标准化：每个 task 只允许一个 setup，base / chat 模型对称地约定 cloze vs MC 形式。OLMES 的局限——只规范 prompt-level 标准化，不解决 contamination / behavioral validity——是 paper 自身承认的 scope limit；AI2 把 OLMES 越来越绑定到 OLMo 训练 workflow（OlmoBaseEval 43-bench 套件，2025-11 OLMo 3 release），社区中立性下降。OLMES 与 lm-eval-harness 的关系在 issue #2002 (https://github.com/EleutherAI/lm-evaluation-harness/issues/2002) 中暴露：OLMES 建议向 harness 反向传播时，normalization、few-shot example pool 等细节常常无法直接 backport。

DCLM [^li2024dclm] 提出一个 epistemically 不同的范式：**fix model + training code，varying 是数据；benchmark 不再是评模型的标尺，而是评数据 recipe 的反应面板**。这把 evaluation 从 model-centric 重新 frame 为 data-centric——一个把 benchmark"绑定到数据 ablation"的设计哲学。DCLM 53-task suite 因此承担一个新角色：作为 data quality 的"度量空间基底"。其代价是 benchmark health 不再被审计——daVinci-LLM [^qin2026davinci] 2026-03 的回应正是把这件事自省化，明确把 "evaluation protocol choices significantly influence understanding of pretraining progress" 写进 paper limitation。

BabyLM [^warstadt2023babylm-cfp] 则代表另一条极端：把 evaluation 与 training data budget 共同约束。100M-word ceiling 不是"评测对象"，而是"研究问题的边界"。BLiMP + EWoK + GLUE 评测套件本身是为标准 LLM 设计的，把它放到 ≤100M-word tiny LM 上是否仍有信号本身就是开放问题——BabyLM 2024 Findings [^warstadt2025babylm-findings] 直白报告"curriculum learning largely failed"、"BLiMP saturating for top entries"等 negative result。

五种路线对照可提炼一个共同 epistemic insight：**任何 community-scale benchmark 都是测量理论 + 数据集 + 工具 + 社会激励四元体，单独优化任一环都不够**。lm-eval-harness 工程层完美但 setup 开放导致跨论文不可比；HELM 测量哲学完整但 community 跟进成本高、活力下降；OLMES 解决 setup 标准化但绑 OLMo workflow；DCLM 把 evaluation 重 frame 为 data 反应面板但绕过 benchmark health；BabyLM 切换问题边界但结果 transferability 仍开放。

5 / 58

1.6 衡量"benchmark 信息量"的工具与本报告的分析框架

综上，本报告后续章节将沿如下分析框架审视具体 benchmark：

信号轴 (signal)：accuracy / F1 等 raw metric 之外，是否有 item-level discrimination 数据 (IRT 派生的 BHI Capability Discrimination)；是否报告 normalization 敏感度；是否在 small-sample 下给出统计区间。

有效轴 (validity)：题目是否含 annotation artifact（Story Cloze 风格 cue、PIQA twin sentence overlap）；是否对 paraphrase / Lorem ipsum 等 ablation 鲁棒（WinoWhat、GoldenSwag）；是否依赖标注质量 (MMLU-Redux 6.49% 错率)。

对抗轴 (adversarial robustness)：构造时是否经 adversarial filtering；filter model 是否 frontier (HellaSwag BERT-based filter 已被 GPT-4 越过)；是否设计 distribution shift 检验（dynamic refresh、cross-lingual variant）。

生态轴 (ecosystem)：是否被 lm-eval-harness / OLMES / HELM 等 community framework 标准化；是否进入 leaderboard 死亡循环 (HF Open LLM Leaderboard 退休)；是否被 BHI / meta-eval 工具 score 为 high-discrimination + high-anti-saturation。

这套四轴框架将贯穿后续 §02–§07，作为评价每个 benchmark 的统一语言。下一章 §02 即从 pretrain 阶段评测特有的方法论问题（loglikelihood vs zero-shot generation、normalize-by-token-length、few-shot 选样、scaling law 视角下的 benchmark 单调性）继续这一线索。具体到工程实现层面的命令行、log 解读与代码示例，请参考 工程实践者手册 02-HARNESS 与 工程实践者手册 03-FRAMEWORKS。

[^bhi-2026]: Zhu, L., Hua, H., Miao, L., Zhao, B. (2026-02). *Benchmark Health Index: A Systematic Framework for Benchmarking the Benchmarks of LLMs*. arXiv:2602.11674. [^eleuther-mcq-norm]: EleutherAI Blog. *Multiple Choice Normalization in LM Evaluation*. https://blog.eleuther.ai/multiple-choice-normalization/. [^gu2024olmes]: Gu, Y., Tafjord, O., Kuehl, B., Haddad, D., Dodge, J., Hajishirzi, H. (2024). *OLMES: A Standard for Language Model Evaluations*. arXiv:2406.08446 (Findings of NAACL 2025). [^gema2024mmlu-redux]: Gema, A.P. et al. (2024). *Are We Done with MMLU?* arXiv:2406.04127. [^surge2024hellabad]: Surge AI. (2024). *HellaSwag or HellaBad? 36% of this popular LLM benchmark contains errors*. surgehq.ai blog. [^mousavi2025garbage]: Mousavi, S.M., Cecchinato, E., Horníková, L., Riccardi, G. (2025). *Garbage In, Reasoning Out? Why Benchmark Scores are Unreliable and What to Do About It*. arXiv:2506.23864. [^paperno2016lambada]: Paperno, D. et al. (2016). *The LAMBADA dataset: Word prediction requiring a broad discourse context*. ACL 2016. arXiv:1606.06031. [^lalor2016irt]: Lalor, J.P., Wu, H., Yu, H. (2016). *Building an Evaluation Scale using Item Response Theory*. EMNLP 2016. [^vania2021irt]: Vania, C. et al. (2021). *Comparing Test Sets with Item Response Theory*. ACL 2021. [^schaeffer2023emergent]: Schaeffer, R., Miranda, B., Koyejo, S. (2023). *Are Emergent Abilities of Large Language Models a Mirage?* NeurIPS 2023. arXiv:2304.15004. [^zellers2019hellaswag]: Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., Choi, Y. (2019). *HellaSwag: Can a Machine Really Finish Your Sentence?* ACL 2019.

6 / 58

arXiv:1905.07830. [^sakaguchi2019winogrande]: Sakaguchi, K., Le Bras, R., Bhagavatula, C., Choi, Y. (2019). *WinoGrande: An Adversarial Winograd Schema Challenge at Scale*. arXiv:1907.10641 (AAAI 2020). [^bisk2020piqa]: Bisk, Y., Zellers, R., Le Bras, R., Gao, J., Choi, Y. (2020). *PIQA: Reasoning about Physical Commonsense in Natural Language*. AAAI 2020. arXiv:1911.11641. [^chizhov2025goldenswag]: Chizhov, P., Nee, M., Langlais, P.-C., Yamshchikov, I.P. (2025). *What the HellaSwag? On the Validity of Common-Sense Reasoning Benchmarks*. arXiv:2504.07825. [^elazar2021artifact]: Elazar, Y., Zhang, H., Goldberg, Y., Roth, D. (2021). *Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema*. EMNLP 2021. arXiv:2104.08161. [^winowhat2025]: WinoWhat (2025). *A Parallel Corpus of Paraphrased WinoGrande Sentences with Common Sense Categorization*. arXiv:2503.23779. [^schwartz2017stylistic]: Schwartz, R. et al. (2017). *The Effect of Different Writing Tasks on Linguistic Style: A Case Study of the ROC Story Cloze Task*. CoNLL 2017. [^white2024livebench]: White, C., Dooley, S., Roberts, M. et al. (2025). *LiveBench: A Challenging, Contamination-Limited LLM Benchmark*. ICLR 2025 Spotlight. arXiv:2406.19314. [^ni2024mixeval]: Ni, J., Xue, F., Yue, X. et al. (2024). *MixEval: Deriving Wisdom of the Crowd from LLM Benchmark Mixtures*. NeurIPS 2024. arXiv:2406.06565. [^biderman2024lessons]: Biderman, S. et al. (2024). *Lessons from the Trenches on Reproducible Evaluation of Language Models*. arXiv:2405.14782. [^hf-leaderboard-retirement]: clefourrier @ HuggingFace. (2025-03-13). *End of the Open LLM Leaderboard*. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard/discussions/1135. [^liang2022helm]: Liang, P., Bommasani, R., Lee, T. et al. (2022). *Holistic Evaluation of Language Models*. arXiv:2211.09110. [^crfm2025helm-capabilities]: CRFM. (2025-03-20). *HELM Capabilities*. https://crfm.stanford.edu/2025/03/20/helm-capabilities.html. [^li2024dclm]: Li, J., Fang, A., Smyrnis, G. et al. (2024). *DataComp-LM*. NeurIPS 2024 Datasets and Benchmarks. arXiv:2406.11794. [^qin2026davinci]: Qin, Y. et al. (2026-03). *daVinci-LLM: Towards the Science of Pretraining*. arXiv:2603.27164. [^warstadt2023babylm-cfp]: Warstadt, A. et al. (2023). *Call for Papers: The BabyLM Challenge*. arXiv:2301.11796. [^warstadt2025babylm-findings]: Warstadt, A. et al. (2025). *Findings of the BabyLM Challenge*. arXiv:2504.08165.

02 · Pretrain 评估方法论

2.1 base model 评测的"特有问题"为什么需要单独讨论

Pretrain 阶段的评测有一个长期被工程化文献忽视的方法论特殊性：base model（无 instruction tuning、无 RLHF）的输出分布与 chat / instruct model 在结构上不同——前者是 next-token 自回归 LM 的"自然分布"，后者是 chat-template 包裹下、对齐到 helpful-and-harmless 偏好的 conditional 分布。把同一道 MMLU 题目分别喂给 base 与 chat 模型，得到的不是同一类信号：base 模型用 loglikelihood ranking 选 A/B/C/D continuation，chat 模型则倾向于生成完整自然语言回答然后 extract 字母。这导致一系列 pretrain 阶段评测特有

的方法学张力：cloze vs MCQ formulation 不对称（OLMES 把它列为 first-class problem [^gu2024olmes]）、normalization 选择敏感、prompt template 微小扰动带来 ±10 pp 抖动 (Biderman 等 2024 [^biderman2024lessons] 总结的三类问题之一)、few-shot 选样偏差、token-length normalization 之争、以及 scaling law 视角下"哪些 benchmark 真正能反映 capability 单调增长"等问题。

本章按以下顺序展开：(2.2) loglikelihood vs zero-shot generation 的统计含义；(2.3) normalize-by-token-length 之争与其在 Brown 2020 之后的演化；(2.4) few-shot 选样偏差 (Zhao et al. 2021 [^zhao2021calibrate])；(2.5) scaling law 视角下 benchmark 的"单调性 / 有用性"；(2.6) emergent capability 的 metric 依赖性争议 (Schaeffer 2023 [^schaeffer2023emergent])；(2.7) BabyLM 的"sample-efficient"与 DCLM 的"data-centric"两条 pretrain 评估哲学对比；(2.8) 2026 新范式：in-training-probing、MaP、daVinci-LLM 的 self-reflective 评估论。

2.2 loglikelihood vs zero-shot generation：两类"答案"分别在测什么

base model 的 MCQ 评分有两条几乎完全不同的协议路径，二者在统计含义、对 model size 的敏感性、对 prompt 的鲁棒性都有别。

Loglikelihood ranking：把每个 option 拼到 prompt 后面计算 $\log P(\text{option},|,\text{prompt})$，取 argmax。这等同于 base LM 在 evaluation 任务上以"哪一个 continuation 更自然"作答。它的统计含义清晰：直接 query 模型的 unnormalized posterior。优势：(i) 对小模型友好——即使模型还不会按指令输出 A/B/C/D 字母，loglikelihood 也能给出一个有意义的 ranking；(ii) deterministic（无需 sampling temperature）；(iii) 对 chat-template 不敏感，是 base model 的"自然语言"。劣势：(i) 选项长度差异会扭曲 ranking——更长的 option 在 unnormalized $\log P$ 下天然更低 (各 token 概率连乘)；(ii) 与下游 production usage 距离远——chat 模型在 user query 上不是 ranking 4 个 option，而是 free-form generation。

Zero-shot (or few-shot) generation：让模型自回归生成"答案：A"这样的字符串，再用 regex / SymPy / unit test 等 answer extractor 抽取最终答案。这是 GSM8K (Cobbe et al. 2021 [^cobbe2021gsm8k])、HumanEval (Chen et al. 2021 [^chen2021humaneval])、MATH 等 generation-style benchmark 的标准协议，2023 年后被 HELM v2 / OLMES chat-mode 推广到 MCQ。统计含义改变：现在测量的是 "模型在指令理解 + answer formatting + 内容正确" 的联合分布，而非单纯的 next-token posterior。这对 base model 不公平——一个能力很强但未经 instruction tuning 的 7B 模型可能 95% 时间不输出"A"字母而输出"我认为这道题..."，answer extractor 抓不到，accuracy 就降为 0。OLMES paper [^gu2024olmes] 把这一不对称性 codify 为 base / chat 二分协议：base 用 cloze（continuation loglikelihood），chat 用 MCQ formulation（"Question: X\nA. ...\nB. ...\nAnswer:" 再 ranking 字母）；OlmoBaseEval 43-bench 套件正是基于这一区分。

两种协议在 frontier 模型上仍可差出 3–5 pp。HuggingFace Open LLM Leaderboard v1 时代的 ARC / MMLU 数字与 chat 模型在自家 model card 上自报 generation-mode 数字常常

8 / 58

对不上，正是这一不对称的具体体现。Biderman 等 [^biderman2024lessons] 总结 lm-evaluation-harness 三年维护经验时把"模型对 prompt 微小扰动的高敏感性、跨方法学不可比、复现性不足"列为三大类问题——本质都是"协议选择是 evaluation 的隐藏自由度"的体现。

值得引用 Schaeffer 等 2023 [^schaeffer2023emergent] 的批评作为方法论 anchor：他们指出当用 token-level cross-entropy 而非 sharp accuracy / EM 重新绘制 GPT 系列在多 task 上的 capability 曲线时，所谓的 "emergent ability" 大多变成 smooth scaling——"突现"是 metric 不连续性的产物。这一观察的方法论含义是：**loglikelihood-based protocol 比 generation-based protocol 更可能呈现 smooth scaling，因为 loglikelihood 本质是连续的；generation 协议则把"未能产出正确字符串"的连续概率坍缩为 binary 0/1**。Pretrain 评测优先采用 loglikelihood 不只是因为 base model 没指令能力，更因为 loglikelihood 在 scaling law 拟合上更稳健。

2.3 Normalize-by-token-length 之争：Brown 2020 与之后

GPT-3 paper (Brown et al. 2020 [^brown2020gpt3]) 在 MCQ 评分时使用 **byte-length normalized loglikelihood**：把 $\log P(\text{option},|,\text{prompt})$ 除以 option 的 byte 长度，再 argmax。理由很简单：option 长度差异巨大时（如 OpenBookQA 有 1-词到 1-句的 option），unnormalized $\log P$ 系统偏向短 option，accuracy 失真。byte normalization 是一种 zero-cost 的去偏。

但 byte normalization 自身并非完美。EleutherAI 在 multiple-choice-normalization 一文 [^eleuther-mcq-norm] 中系统比较了四种 normalization 在多个 MCQ benchmark 上的影响，得出两个关键 finding：

第一，**normalization 选择影响分数 5–15 pp**。同一 8B 模型在同一 MMLU 题集上，byte / char / unconditioned / acc_norm 四种 normalization 下的 accuracy 可以从 55% 到 70%。这意味着 community 报告的 MMLU 数字若不附 normalization，跨论文比较实质无意义。

第二，**没有一个 normalization 在所有 benchmark 上最佳**。HellaSwag 在 byte-norm 下与人类基线最接近；ARC-Challenge 在 acc_norm (length-normalized acc) 下更稳健；某些任务的 unconditioned normalization（即 $\log P(\text{option},|,\text{prompt}) - \log P(\text{option})$）反而最 robust 因为它显式扣除 option 的 marginal probability，矫正了"模型把 'the' 作为 default option 的偏置"。

社区的 de facto 解决是：lm-evaluation-harness 同时报 `acc` 和 `acc_norm`，留给用户选；OLMES 则在 paper 中明确指定每个 task 用哪种 normalization（base = byte normalization in cloze, chat = pure acc on MC letter）。HELM v1/v2 走另一路线：让 task 自定 metric，由 framework 保证 reproducibility。

这一争议在 reasoning model 时代有新表现。当 frontier 模型用 long chain-of-thought 在生成模式下答题时，最终"answer letter"的选择只占 token 序列极小一部分；length normalization 对总 generation $\log P$ 几乎无意义——community 因此向 generation + answer extraction 协议迁移。但对 base model 中检（数据 ablation、scaling law 拟合）这一场景，cloze + byte normalization 仍是 OLMES / OlmoBaseEval / DCLM 等套件的默认。

9 / 58

2.4 Few-shot 选样偏差：Zhao et al. 2021 与"prompt order matters"

GPT-3 paper 报告 few-shot evaluation 时把"in-context example 的选择 / 顺序"当作 fixed convention。Zhao et al. 2021 [^zhao2021calibrate] 系统实验后揭示了一个让 community 不安的 finding：**改变 few-shot example 的顺序、数量或具体样例，accuracy 可以在同一模型同一题集上波动 30+ pp**。他们提出三类偏差：(i) majority label bias——模型倾向于 predict in-context 中出现频次最高的 label；(ii) recency bias——in-context 末尾的 example 对 prediction 影响最大；(iii) common token bias——某些 token 因 pretrain 频次高，被模型 default 输出。

Zhao 等同时提出 **contextual calibration**：在 evaluation 时插入 "N/A" 等 content-less 输入，测量模型的 "default prediction"，再用它校准实际预测——一种 token-level 的 bias 后处理。该方法把 GPT-3 在 SST-2 / TREC 等 task 上的 accuracy 提升 30+ pp。后续工作 (Lu et al. 2022 [^lu2022prompt-order]) 把 prompt order 视为额外超参，建议在 N! 种排列中报告 distribution 而非单一 accuracy。

对 pretrain 评测的实操含义有三：

第一，**few-shot example 必须固定**。OLMES 强制 fixed example pool（不允许随机），lm-evaluation-harness 默认 `--num_fewshot 5 --seed 42`，HELM 在 task config 中 hardcode 例子；如果一篇论文报告 MMLU 5-shot 但没指定 example 选择规则，数字不可复现。

第二，**bias calibration 仍非主流**。Zhao 2021 的 contextual calibration 没有进入 lm-evaluation-harness 默认 pipeline。原因之一是大模型时代 (≥30B) bias 影响相对缩小（模型 entropy 升高），但近期 Holtzman 等 (2021 [^holtzman2021surface]) 的 Surface Form Competition 工作显示这类偏差在 frontier 模型上仍以更微妙形式存在——模型可能因 "Paris" 比 "paris" 在 pretrain 中频次高而把它作为 default option。

第三，**chain-of-thought 引入新自由度**。BBH (Suzgun et al. 2022) 等 reasoning benchmark 把 CoT exemplar 当 first-class config——选 8 个还是 3 个 exemplar、用哪些题、CoT 思路是否 ground truth-aligned，对最终 accuracy 影响巨大。HF Open LLM Leaderboard v2 报告 BBH 3-shot，OLMES 在 task config 中 fix exemplar——community 仍未收敛到统一约定。

2.5 Scaling law 视角：哪些 benchmark 真"单调"

Hoffmann et al. (Chinchilla, 2022 [^hoffmann2022chinchilla]) 与 Kaplan et al. 2020 [^kaplan2020scaling] 把 pretrain 进展 frame 为 **compute → loss 的 power law**。Chinchilla 在 70B 参数与 1.4T token 上 compute-optimally 训练，发现"小模型 + 多 token"比"大模型 + 少 token"在等同 compute 下 cross-entropy 更低，进而推翻了 GPT-3 时代"模型越大越好"的简单 narrative。这一发现的方法论副产品是把"loss"重新树为 first-class pretrain metric——但 loss 与 downstream benchmark accuracy 之间并不严格单调。

Tay et al. 2022 [^tay2022scaling] 在 architecture variation × scaling 系列实验中发现：**不同 architecture 的 scaling curve 在 loss 上几乎相同，但在 downstream benchmark accuracy 上分化显著**。具体来说，pretraining loss 几乎只反映"句子级 likelihood"，而 downstream benchmark 测的是某种 task-specific abstraction——两者的 mapping 不是恒等。这意味着 loss-based scaling law 不能直接预测 benchmark accuracy。

DataDecide / AI2 与 DCLM [^li2024dclm] 在 2024 年的工作把这一问题工程化：用 small-model proxy（412M / 1B）在多 data recipe 上跑 53-task evaluation，作为 7B / 70B scale 的 reference signal。DCLM-Baseline 在 7B 上达到 64% MMLU 5-shot——这是 data engineering（model-based filtering）的杠杆证据。但 DCLM 53-task 内部 benchmark 在 7B scale 上已部分饱和（BHI [^bhi-2026] 显示某些 task capability discrimination 接近 0）；这意味着 small-model proxy 在 mid-scale 还有信号，但跨到 frontier scale 时会失效。

2026 年涌现的 Prescriptive Scaling (Proteus 2k dataset, arXiv:2602.15327 [^prescriptive-scaling]) 试图直接拟合 compute → downstream accuracy 的 saturating sigmoid，作为 prescriptive prediction tool：给定 compute 预算，回归预测 downstream accuracy 上限，并在早期模型 fit 后在后期模型上做 out-of-sample 验证。这是把 scaling law 从描述性 (descriptive) 推向 prescriptive 的尝试，与 DataDecide 互补——一个是 model-size proxy，一个是 compute-to-capability mapping。

单调性 vs sample-level instability：2510.04848 [^instability-paper] 报告 pretrain 过程中 downstream task performance 在相邻 checkpoint 间的 variance 显著——同一模型在 step N 与 step $N+1000$ 上 GSM8K accuracy 可能差 5+ pp，**这并非 model 真实退步，而是 evaluation 协议的高方差**（generation-based eval + small benchmark）。这一发现直接挑战"用 single-checkpoint 评测决定是否继续训练"的 pretrain workflow。in-training-probing (arXiv:2604.01025 [^liu2026probing]) 据此提出 lightweight probe paradigm，把每 checkpoint 评测从 ~1 小时压到 ~3 分钟（详见 §2.8）。

2.6 Emergent capability 的 metric 依赖性：Schaeffer 2023 的余波

Wei et al. 2022 [^wei2022emergent] 的 "Emergent Abilities of Large Language Models" 在 137B 之上的模型上观察到多个 benchmark 上的 sharp performance jump——arithmetic、word unscrambling、Persian QA 等任务在 sub-100B 模型上几乎不可解，超过某临界点后突现到 60-80% accuracy。这一现象被作为"scale 是 capability 的非线性 driver"的关键证据。

Schaeffer et al. 2023 [^schaeffer2023emergent] 提出尖锐反驳：他们重新分析同一系列任务，论证 emergent abilities 是 **metric 不连续性的产物**而非 capability 真实非线性。具体论据：用 EM (exact match) / accuracy 等 binary metric 时，模型部分正确（如 7 位 arithmetic 中算对 6 位）被视为 fully wrong，曲线呈 sharp jump；换成 token-level edit distance、cross-entropy、Brier score 等连续 metric 后，曲线变 smooth，"突现"消失。他们进一步证明只要 metric 在 capability 上是 non-linear（即"接近正确"与"完全正确"被严格区分），任何 smooth 的 underlying capability 都会在该 metric 上呈现 emergent jump。

这一争议的方法论含义远超 emergent ability 本身。对 pretrain 评测：

第一，**评测 metric 的非线性决定了 capability 曲线形状**。一个 benchmark 若用 EM 评分（GSM8K 的 final answer match、HumanEval 的 pass@1 unit test），会放大 sharp jump；用 cross-entropy 或 token-level metric 则呈现 smooth scaling。研究者选 metric 不仅是工程决策，也是 epistemic 决策。

第二，**emergence 的"reality"仍有争议**。Du et al. 2024、Hu et al. 2024 等多个后续工作认为某些 capability（如 in-context learning、chain-of-thought reasoning）即使在连续 metric 下也呈现非线性 transition，Schaeffer 的论点并非全 emergent ability 都消失。Lambada 在 GPT-2 → GPT-3 上的 accuracy 跳跃（7% → 86%）是 well-documented case，但 Schaeffer 风格的连续 metric reanalysis 还未在 LAMBADA 上系统做过。这是开放问题。

第三，**对 pretrain progress monitoring 的实操影响**：如果用 EM-based benchmark 监控训练，可能在大部分 step 看不到 progress，然后在 final 几个 checkpoint 看到 "突现"——这给训练决策造成 false alarm。改用 token-level 或 probabilistic metric (loglikelihood, perplexity on relevant subset) 可以更早捕捉信号。in-training-probing 选 AUROC 作为 metric 的部分原因正是 AUROC 对模型 confidence 而非二分类的连续敏感。

2.7 BabyLM vs DCLM：sample-efficient vs data-centric 的两条评估哲学

BabyLM Challenge [^warstadt2023babylm-cfp] 与 DataComp-LM (DCLM) [^li2024dclm] 在 pretrain 评估上代表两种几乎对立的哲学，对比它们对理解 2026 的 pretrain 评估格局有教益。

BabyLM：把 data budget 严格约束在 100M-word（Strict track）或 10M-word（Strict-Small）以下，研究问题是"在这点数据上如何最优 pretrain？" 评测套件用 BLiMP（语法）+ EWoK（world knowledge）+ (Super)GLUE（NLU）——一组本质为标准 LLM 设计的 benchmark。BabyLM 2024 / 2025 Findings [^warstadt2025babylm-findings] 报告了若干 negative result：(i) **curriculum learning 在 BabyLM 多届都未带来稳定收益**——这是与 cognitive science 期望相反的发现，但论文继续涌现 curriculum 方法，社区收敛缓慢；(ii) **shorter input sequence + student-teacher distillation 反而最有效**；(iii) **BLiMP 在 100M-word top entry 上接近饱和**——一个为标准 LM 设计的 benchmark 在 tiny LM 上同样会达到 ceiling。LTG-BERT 在 100M-word 数据上 outperform 万亿词训练模型这一事实，是"数据效率 vs scale"辩论的有力证据。

DCLM：反向操作——固定 model architecture（OpenLM transformer + RoPE + SwiGLU）+ training code，变 data recipe。规模到 7B / 2.6T token，53-task evaluation 套件覆盖 MMLU / ARC / HellaSwag / GSM8K / HumanEval 等。Top-level metric 是 Core 子集（22 task）的 centered macro-average。findings：(i) **model-based filter（fastText classifier 过滤 high-quality web text）是最关键因素**，远超 deduplication 与 rule-based filtering；(ii) DCLM-Baseline 7B 在 MMLU 拿 64%，同期 Llama-1-7B 35% / Mistral-7B 60%，说明 data engineering 的杠杆巨大。

12 / 58

两者的对比指向几个 pretrain 评估的深层问题：

问题一：benchmark 的 transferability。 BabyLM 的 LTG-BERT 在 100M-word 上 BLiMP > 万亿词模型，但 BLiMP 是为 generic LM 设计的。这意味着 BabyLM finding 是否能迁移到 frontier-scale 模型？2024–2025 Findings 自己承认这是 open question。同样，DCLM 53-task suite 是否在 70B / 200B model 上仍 informative？BHI 分析表明部分 task 已饱和，benchmark suite 自身需要 stage-specific selection。

问题二：evaluation protocol 是 framework 的产物。 BabyLM 用 BLiMP + GLUE fine-tune，DCLM 用 lm-eval-harness compatible 协议。两者对 normalization / few-shot 等选择各自约定。daVinci-LLM [^qin2026davinci] 2026-03 paper 把这一问题 self-reflectively 写进 limitation："evaluation protocol choices significantly influence understanding of pretraining progress"——一个工业 scale (3B model, 8T token, 200+ ablation) 的开源 framework 公开承认自己的 200+ 消融结论受 protocol 选择约束。这是 pretrain 评测领域罕见的 epistemic honesty。

问题三：data-centric 与 sample-efficient 是否殊途同归？ DCLM 的 model-based filter 本质是把"高质量 token 密度"提高，等价于在等同 token 数下增加 effective sample；BabyLM 在 hard cap 下逼研究者做 architecture / objective innovation。两条路线都指向"data efficiency"是 frontier 的杠杆，但 BabyLM 强调 cognitive plausibility 与小机构友好，DCLM 强调 industrial scale 可重复。

2.8 2026 新范式：in-training-probing 与 daVinci-LLM 的 self-reflective 评估

2026 年涌现的几个工作把 pretrain 评估推向新方向。

in-training-probing (Liu et al. 2026 [^liu2026probing], arXiv:2604.01025)：训练 lightweight probe 输入 LLM checkpoint 的 internal representation，直接预测该 checkpoint 在 downstream task 上的成功率 (pass@1)。在 OLMo3-7B 的 checkpoints 上验证，跨 8 个 benchmark (MMLU / GSM8K / MATH / BBH / AIME / GPQA / HumanEval / MBPP)，Submodel probe 平均 AUROC 0.789，跨 distant future checkpoint 仍 >0.75。把每 checkpoint 评测从 ~1 小时压到 ~3 分钟，对 trillion-token pretrain workflow 是 20x 加速。**critique（来自 paper 自身与外部）**：AUROC 0.78 留 22% 误差，对 "continue vs stop training" 这类关键决策仍需 generative eval verification；仅在 OLMo3 验证，其他架构 (MoE、reasoning-tuned base) 的 internal representation 与 OLMo3 不同，transfer 是 open question；probe target benchmark 自身的 health 影响 probe quality，用 saturated benchmark 训 probe 会 inflate AUROC（BHI [^bhi-2026] 视角）。in-training-probing 与 MaP (arXiv:2510.09295)、benchmark-health-index 形成 "intermediate checkpoint eval" 论文集群——把 pretrain monitoring 从"每 1000 step 跑一次 full eval"转向"高频 lightweight probe + milestone full verification"双层结构。具体的训练 pipeline 与 probe inference 工作流见 工程实践者手册 05-REASONING-ERA。

daVinci-LLM (Qin et al. 2026 [^qin2026davinci], arXiv:2603.27164)：提出 Data Darwinism L0-L9 数据处理 taxonomy（L0-L2 raw acquisition / format / rule-based filter →

13 / 58

L3-L5 model-based filter / generative refinement / cognitive completion → L6-L9 higher-order synthesis），配套 daVinci-3B 模型（3B 参数、8T token、200+ ablation），match OLMo-3 7B 性能。框架明确把 "evaluation protocol choices significantly influence understanding of pretraining progress" 作为 self-reflective limitation。**critique**：L6-L9 仍是理论前沿，daVinci-3B 实际只用到 L0-L5；200+ ablation 用同一 protocol 跑，没做 cross-validate；daVinci-3B vs OLMo-3 7B 对比 fairness 存疑（训练数据量、curriculum、optimizer 不同，"better data" 与 "different model size" 的归因混淆）。但 daVinci-LLM 把 "evaluation 影响 understanding" 写进 paper 的做法，是 pretrain 评估方法论自我成熟的标志——它承认 pretraining 是 experimental science，evaluation 是该 science 的 measurement 工具，而非 ground truth。

Benchmark Health Index (Zhu et al. 2026 [^bhi-2026], arXiv:2602.11674) 把"为 benchmark 选 benchmark" frame 为 first-class research question。三轴 Capability Discrimination / Anti-Saturation / Impact 给出 106 个 validated benchmark 的健康度三元组。对 pretrain 评测的实操含义：选 benchmark 时优先 CD 高 + AS 高的 task；避免 Impact 低 + AS 低（即将 saturate 的 niche）。**critique**：score 依赖 91-model 2025 distribution，新 model（reasoning-tuned）加入后健康度 score 漂移；Impact 用 citation 引入 age bias；Anti-Saturation 假设线性 ceiling-rise，对 o1 / R1 等 reasoning model 出现导致 ceiling 跳跃不 robust；stage-agnostic，pretrain vs SFT/RLHF 需不同 benchmark。daVinci-LLM 论文明确指出 stage-specific 选择是必要。

2.9 小结：pretrain 评测的方法论张力 map

把 §2.2–§2.8 综合起来，2026 时刻的 pretrain 评测方法论可以视作五个张力的交织：

(i) **协议张力**：loglikelihood vs generation 在 base vs chat 上分别天然适配，但 community 缺统一 reporting 约定；OLMES 是 most-mature canonical setup 但社区中立性下降。

(ii) **归一化张力**：byte / char / unconditioned normalization 的选择影响 5–15 pp 分数；社区共识尚未形成。

(iii) **few-shot 张力**：example pool / order / seed 影响 30+ pp accuracy，bias calibration 未主流化。

(iv) **scaling 张力**：loss-based scaling law 不严格映射到 downstream benchmark；某些 benchmark 已饱和无法继续作为 progress proxy；step-level instability 挑战 single-checkpoint 评测。

(v) **元评估张力**：BHI 提出审计 benchmark 自身健康度，但 score 依赖 model distribution；daVinci-LLM 自省 "evaluation 影响 understanding"，但未给出 cross-protocol validation 方案。

后续章节 §03（知识 / 推理 benchmark 学术批评）将沿这五条张力具体切入 MMLU / HellaSwag / ARC / TriviaQA 等单一 benchmark 的 critique；§04 把视角转向数学 / 代码 / STEM；§05 集中讨论饱和与 contamination 学术辩论；§06–§07 讨论 live / dynamic / agent

14 / 58

范式与未来评估辩论。本章关注的"评估如何在数字上自洽"是这一切讨论的 prerequisite——脱离了 protocol-level 严谨，benchmark-level critique 的力度也会被稀释。

工程层面的命令行与 log 解读细节请参考 工程实践者手册 02-HARNESS（lm-evaluation-harness 深度导读）与 工程实践者手册 03-FRAMEWORKS（OLMES / lighteval / HELM v2 / DCLM 框架对比）。

[^gu2024olmes]: Gu, Y., Tafjord, O., Kuehl, B., Haddad, D., Dodge, J., Hajishirzi, H. (2024). *OLMES: A Standard for Language Model Evaluations*. arXiv:2406.08446. [^biderman2024lessons]: Biderman, S. et al. (2024). *Lessons from the Trenches on Reproducible Evaluation of Language Models*. arXiv:2405.14782. [^zhao2021calibrate]: Zhao, T.Z., Wallace, E., Feng, S., Klein, D., Singh, S. (2021). *Calibrate Before Use: Improving Few-Shot Performance of Language Models*. ICML 2021. [^schaeffer2023emergent]: Schaeffer, R., Miranda, B., Koyejo, S. (2023). *Are Emergent Abilities of Large Language Models a Mirage?* NeurIPS 2023. arXiv:2304.15004. [^cobbe2021gsm8k]: Cobbe, K. et al. (2021). *Training Verifiers to Solve Math Word Problems*. arXiv:2110.14168. [^chen2021humaneval]: Chen, M. et al. (2021). *Evaluating Large Language Models Trained on Code*. arXiv:2107.03374. [^brown2020gpt3]: Brown, T. et al. (2020). *Language Models are Few-Shot Learners*. NeurIPS 2020. arXiv:2005.14165. [^eleuther-mcq-norm]: EleutherAI Blog. *Multiple Choice Normalization in LM Evaluation*. https://blog.eleuther.ai/multiple-choice-normalization/. [^lu2022prompt-order]: Lu, Y., Bartolo, M., Moore, A., Riedel, S., Stenetorp, P. (2022). *Fantastically Ordered Prompts and Where to Find Them*. ACL 2022. [^holtzman2021surface]: Holtzman, A. et al. (2021). *Surface Form Competition: Why the Highest Probability Answer Isn't Always Right*. EMNLP 2021. [^hoffmann2022chinchilla]: Hoffmann, J. et al. (2022). *Training Compute-Optimal Large Language Models*. arXiv:2203.15556. [^kaplan2020scaling]: Kaplan, J. et al. (2020). *Scaling Laws for Neural Language Models*. arXiv:2001.08361. [^tay2022scaling]: Tay, Y. et al. (2022). *Scaling Laws vs Model Architectures: How does Inductive Bias Influence Scaling?* arXiv:2207.10551. [^li2024dclm]: Li, J., Fang, A., Smyrnis, G. et al. (2024). *DataComp-LM*. NeurIPS 2024. arXiv:2406.11794. [^prescriptive-scaling]: (2026-02). *Prescriptive Scaling: Compute-to-Capability via Proteus 2k*. arXiv:2602.15327. [^instability-paper]: (2025). *Instability in Downstream Task Performance During LLM Pretraining*. arXiv:2510.04848. [^liu2026probing]: Liu, Z., Lun, T., Wen, Z. et al. (2026). *Fast and Accurate Probing of In-Training LLMs' Downstream Performances*. arXiv:2604.01025. [^wei2022emergent]: Wei, J. et al. (2022). *Emergent Abilities of Large Language Models*. TMLR 2022. [^warstadt2023babylm-cfp]: Warstadt, A. et al. (2023). *Call for Papers: The BabyLM Challenge*. arXiv:2301.11796. [^warstadt2025babylm-findings]: Warstadt, A. et al. (2025). *Findings of the BabyLM Challenge*. arXiv:2504.08165. [^qin2026davinci]: Qin, Y. et al. (2026-03). *daVinci-LLM: Towards the Science of Pretraining*. arXiv:2603.27164. [^bhi-2026]: Zhu, L., Hua, H., Miao, L., Zhao, B. (2026-02). *Benchmark Health Index*. arXiv:2602.11674.

引子：从"百科 quiz"到"validity 危机"

知识 / 推理类 benchmark 是 LLM 评测的元老。它们的共同祖先是 SQuAD、SWAG、WSC 这类 2016–2018 NLP 任务集，2020 年后被 MMLU、HellaSwag、PIQA、WinoGrande、ARC 改写成"巨型百科 MCQ + 常识推断"的统一形式，统治了 GPT-3 → Llama-3.1 这五年的 LLM 报告卡。然而进入 2024–2026 年后，几乎每张牌都被独立学者从三个维度同时戳穿：(1) annotation quality（题目本身错率惊人）；(2) construct validity（题目能否真正测量声称的能力）；(3) data contamination（测试集渗入预训练 corpus）。本章按"广覆盖知识 → 经典常识三件套 → 抽样型推理 → truthfulness"四组，把代表性 benchmark 的设计动机、争议焦点和 2025+ critique 文献串成一条 narrative，目的是让本节服务于 §05 的 saturation/contamination 大辩论和 §07 的未来路线讨论。

一、广覆盖知识：MMLU 家族与 AGIEval

MMLU：百科 MCQ 的奠基与崩塌

MMLU（Hendrycks 等 2020）的设计逻辑是"用一个 14,042 道、跨 57 学科、4 选 1 的题集衡量模型在多大程度上掌握人类各领域知识"[^hendrycks2021mmlu]。从方法论看，它把"general intelligence"操作化为"美国教育系统抽样题的 accuracy"，并选定 5-shot + log-likelihood 作为标准协议。这个范式有两大优势：(a) MCQ 评测便宜、可重复；(b) 57 学科分层允许后续工作做 per-subject diagnostics。从 GPT-3 (43.9%) 到 GPT-4 (~86%) 到 GPT-4o / Claude 3.5 Sonnet (~88.7%)，MMLU 是过去五年最被频繁报告的"capability 单数字"。

但 2023 年起 MMLU 进入 validity 危机。Sainz 等 2023 系统记录了 MMLU 等 MCQ benchmark 在 C4 / Common Crawl 中的污染信号——前沿模型在"缺失选项"的部分填空回忆攻击下能高频复刻 ground-truth，是污染的强证据[^sainz2023nlp]。Gema 等 2024 的 MMLU-Redux 项目重新标注了 5,700 题，发现 **6.49% 题目含错**，其中 Virology 子集错误率达 57%[^gema2024mmluredux]。第三个打击来自 Holtzman 等指出的 prompt 敏感性——MCQ 的 4 选 1 形式让 token surface probability 主导，prompt 模板的微小改动可造成 4-5pp 浮动[^holtzman2021surface]。综合三股力，2024 年后再单独报 MMLU 几乎被视为"compliance ritual"，主社区的关注转向 MMLU-Pro。

MMLU-Pro 是否真的"Pro"？

MMLU-Pro（Wang 等, TIGER-Lab, NeurIPS 2024）刻意做了三件事：(1) 把 4 选项扩到 10 选项（猜中概率 25% → 10%）；(2) 从 STEM 题库、TheoremQA、SciBench 等推理密集源补题至 12,032 题；(3) 显式人工筛去 MMLU-Redux 标注为错的题[^wang2024mmlupro]。结果是 frontier 模型分数下降 16–33%，且 CoT 在 MMLU-Pro 上能拉 +5–10pp（MMLU 上几乎不动），分辨度回归到 GPT-3.5 时代的 MMLU 水平。

但 MMLU-Pro 是否"真的 Pro"？维护方 issue tracker 反映三类争议：拼写/语法错（Issue #76）、小模型异常高分提示 contamination（Issue #69）、10 选项之间互相过近引入 ambiguity[^mmlupro-issues]。更深的方法论问题是：扩选项数本质是"拉低猜中 baseline"，并不直接保证题目测的是更深推理；MMLU-Pro 与原 MMLU 共享同样的 multiple-choice 形

16 / 58

式偏差（surface lexicon、选项长度、stem token bias）。这意味着 MMLU-Pro 也只是把饱和窗口往后推了 1-2 个 model generation，并未根本解决"MCQ 测的是什么"这一构造效度问题。Wang 等也在 NeurIPS 2024 D&B Spotlight 自报，他们的 ablation 显示 MMLU-Pro 对 prompt format 的依赖比 MMLU 略低（±1pp 而非 ±4pp），但仍非"对协议不敏感"。

AGIEval：人类高利害真题

AGIEval（Zhong 等, Microsoft, 2023）走了另一条路：放弃 quiz 形式，直接把 SAT / LSAT / Gaokao / 司考 / GRE 等人类高利害考试当作 LLM 的考场[^zhong2023agieval]。论文核心叙事是"GPT-4 在 SAT Math 95%、LSAT 超均，但在 Gaokao 数学物理仍低于人类"，被 GPT-4 Technical Report 大量引用。AGIEval v1.1（2024）补 2023 Gaokao 真题、修复多答案题。

AGIEval 的争议集中在数据治理：题如何采集、版权如何处置（Issue #7、#11）；部分题"无标准答案"（Issue #29）；样本量子任务化导致分项噪声大（如 Gaokao-Biology 仅几十题，单数字标准误 >5pp）[^agieval-issues]。最关键的 contamination 隐患是 Gaokao / SAT 真题在中英文 web crawl 中长期公开，AGIEval 与 Llama 3 contamination 分析中报告的 NQ / TriviaQA 类型类似——预训练 corpus 中很可能已经含数百份"真题 + 详解"的资料。AGIEval 团队未公布详细的去重/contamination 审计报告，使其作为"中英文 capability 标杆"的可信度被打折扣。这一点在与 C-Eval / CMMLU 对照时尤为明显——后两者明确声明自建题、不来自公开考试。

二、闭卷 QA：TriviaQA、Natural Questions 与 test-train overlap

TriviaQA（Joshi 等, ACL 2017）与 Natural Questions（Kwiatkowski 等, TACL 2019）是 closed-book QA 的"参数化知识"探针：模型只看问题，要从权重中召回答案[^joshi2017triviaqa][^kwiatkowski2019nq]。两者的共同关键引用是 **Lewis, Stenetorp, Riedel (2020)**：在所有主流 open-domain QA dataset 上做 test-train overlap 审计，发现 60–70% 的测试题答案在训练集中已出现，30% 测试题有近似 paraphrase[^lewis2020testtrain]。这意味着 closed-book QA 分数主要反映记忆，不是泛化。

Llama 3 technical report（Dubey 等 2024）做了直接的 contamination 量化：**52% 的 NQ 测试集已出现在 Llama 3 的预训练 corpus 中**[^dubey2024llama3]。这是 MCQ contamination 之外另一个量化点——开放生成式 QA benchmark 同样面临测试集渗漏，且因为答案别名集不完整，closed-book EM 实际上是"被低估的高估"：分数被记忆抬升，但又被别名缺失压低，净效果是被高估，且 noise 来源不明。

方法论上，Lewis et al. 提出三件事：(1) 分离 "question paraphrase overlap"、"answer overlap"、"both" 三类指标；(2) 报告 contamination-controlled subset 的分数；(3) 把 closed-book QA 视作"参数化召回 + 浅泛化"复合任务，而非单一 capability。这套指标在后续 LiveBench、MathArena、SWE-bench Live 等设计中被反复引用作为 contamination 控制

的范式。NQ 仓库已于 2026-04 归档（read-only），意味着这一代 closed-book QA benchmark 的"官方维护时代"也基本结束。

三、经典常识三件套与 ARC

HellaSwag："几何过拟合"的活标本

HellaSwag（Zellers 等 2019）的设计本身就是反 shortcut 的——SWAG (2018) 被 BERT 几月内刷到 86% 后，作者用 adversarial filtering 把 BERT 自己来当判别器迭代 AF 至收敛，再扩到 ActivityNet + WikiHow 两域，长尾上下文 + 跨域 zero-shot 让 BERT 塌回 48% [^zellers2019hellaswag]。但 HellaSwag 自己也在 2024-2025 经历了同样的 validity 反噬。

第一击是 Surge AI 2024 的 36% 错率审计：ActivityNet 子集中 95.7% 续写不合英语语法 [^surge2024hellabad]。第二击是 Chizhov 等 2025 的 GoldenSwag 论文："What the HellaSwag? On the Validity of Common-Sense Reasoning Benchmarks"——他们做了一个简单实验：把 question 删掉只让模型看 4 个候选续写，或者用 "Lorem ipsum" 占位 question，再观察模型预测如何变化。结果是 **≥65% 题目的模型预测不变**，意味着对超过 2/3 的题，"问题文本本身"对模型作答几乎不提供信息——剩下的 surface 线索（长度、句法、词频）已经够推出答案[^chizhov2025goldenswag]。作者随后发布 GoldenSwag 子集（保留约 1,500 题清洗过的样本），并建议社区把 HellaSwag 的报数迁移到 GoldenSwag。

这是 construct validity 失败的教科书例子：HellaSwag 声称测 commonsense NLI，但实测发现题目设计中泄漏的 statistical signals 已经主导预测。结合 saturation（Claude 3 Opus 95.4%、Llama 3.1 95+%）和 test set 公开导致的污染，HellaSwag 在 2026 年的实际"信号"近 0。然而它仍出现在几乎每份 model card，因为它是 Open LLM Leaderboard v1 四件套之一——一种"路径依赖"的 reporting。

WinoGrande：zero-shot 严格评估的真相

WinoGrande（Sakaguchi 等 2019, AAAI 2020）由 Allen AI 把 Winograd Schema Challenge 扩到 44k，用 AfLite filtering + twin sentences 双重防 shortcut[^sakaguchi2019winogrande]。RoBERTa 一度只 79.1%，人类 94.0%，看起来是个理想的"反统计协同"评测。

Elazar 等 2021 EMNLP "Back to Square One"做了关键的方法论矫正：他们指出 LLM 在 WinoGrande 的"提升"几乎全部来自有监督微调阶段（fine-tuning 让模型学到 dataset-specific artifacts），在严格 zero-shot 评估下 popular LMs 表现近随机（接近 50%）[^elazar2021squareone]。换句话说，WinoGrande 测的不是 commonsense disentanglement，而是"模型对该 dataset specific lexical artifacts 的微调表面拟合能力"。这一发现在 2025 年被 WinoWhat 进一步验证：作者把 WinoGrande 句子做 meaning-preserving paraphrase（保留推理结构、换措辞），观察到现代 LLM 显著退化，证实模型仍在依赖 surface cues 而非 schema 的 commonsense pivot[^winowhat2025]。

18 / 58

WinoGrande 的"saturation 到 ~87.5%"早已被广泛报告，但 Elazar + WinoWhat 这条 critique line 才是更深的指控：饱和值 87% 与人类 94% 之间的 gap 既由噪声造成，也由 artifact leakage 造成；二者都不再反映 schema 设计想测的常识。这同样指向 §05 的 saturation-with-validity-loss 主题。

PIQA：Global-PIQA 的跨语言 37pp 缺口

PIQA（Bisk 等 2020, AAAI）测物理常识——goal + 两个 sol，模型选哪个物理上可行；典型题如"用瓶子开瓶器还是开酒器开 wine bottle"[^bisk2020piqa]。Phi-3.5-MoE 已到 88.6%，Epoch AI 把 PIQA 标为已"crossed the human ceiling"的饱和 benchmark[^epoch2026piqa]。但 2025 年 Global-PIQA 给了一记更深的 critique。

Chang 等 2025 在 NeurIPS 召集 335 位研究者跨 65 个国家把 PIQA 翻译/重写到 116 种语言/文化，每种语言独立标注。核心发现：**英文 PIQA 分数显著高估 LLM 的"物理常识"——在低资源语言上 frontier LLM 与英文表现的 accuracy gap 高达 37pp**[^chang2025globalpiqa]。这一发现把"physical commonsense"从一个 universal capability 拆成"英文 LLM 在 instructables.com 抓的英文物理常识"——一个非常窄的文化切片。

方法论上 Global-PIQA 是过去三年最重要的 culture-sensitive validity 工作之一：它证明许多被 LLM 论文广泛报告的"常识 / 推理"能力，当跨语言重测时立刻露出地区性 bias。这把"surface artifacts critique"（HellaSwag/WinoGrande）和 "cultural artifacts critique"（Global-PIQA）合并成同一条 critique line —— pretrain era 经典三件套测的与其说是"commonsense"，不如说是"在特定 web-crawl 子集中常见的英文 textual patterns"。

ARC 系族：饱和与历史定位

ARC（Clark 等 2018）把美国 3–9 年级科学考试做成 4 选 1 MCQ，并用 "challenge filter"（IR + PMI 双 baseline 都答错）筛出 2,590 题作为 Challenge Set，与 5,197 题的 Easy Set 形成对照[^clark2018arc]。Llama 3.1 405B 在 ARC-Challenge 已 96.1% [^llama3card]，Easy 几乎到顶。其饱和并不令人意外——题源是公开科学考试、ARC corpus 是 web crawl 的近邻、length-normalized log-likelihood 还允许 5pp 量级的协议浮动。

ARC 的"学术意义"今天主要不在分数，而在两点：(1) "Easy/Challenge" 二分作为 2018 年 baseline 难度的快照，可以反推过去 8 年 NLP 能力跃迁，是 scaling law 文献喜欢用的"小模型 → 大模型 dataset"；(2) ARC-AGI（François Chollet 系列）在 2024-2025 重新启用了 ARC 这个名字（虽然技术上是完全不同的 grid puzzle），形成令人困惑的命名碰撞。本章不展开 ARC-AGI，因为它属于第 6 章 live / agent 范式。

四、BBH：CoT 时代的"反 reasoning"批评

BBH（Suzgun 等, EMNLP Findings 2023）从 BIG-Bench 200+ 任务中筛出 "prior LLM < average human-rater" 的 23 个子任务，每任务 ~250 例，共 6,511 题，主要测多步推理 +

结构化解析[^suzgun2023bbh]。它的历史地位是 "CoT 在难任务上 emergent gain 的决定性证据"——PaLM-540B + CoT 在 10/23 任务超过 human average，Codex 17/23。

BBH 在 2024-2025 经历了同时来自两个方向的 critique。第一是 invalid CoT 仍涨分：Madaan & Yazdanbakhsh 2023 "Invalid Logic, Equivalent Gains" 实证显示，即便逻辑显然错误的 CoT 也能在 BBH 上带来与正确 CoT 相当的 accuracy 增益[^madaan2023invalidcot]。这把 "CoT 提升 = reasoning 提升" 这一隐含等式戳破——BBH 的 accuracy 与 reasoning quality 解耦。第二是饱和与替代：到 2024 末 frontier 模型在 BBH 上 >90%，Google DeepMind 于 2025-02 推出 **BBEH (BIG-Bench Extra Hard)** 全替换接班，明确以"BBH 已饱和"为前提，把每个 BBH 任务对位重写到难度大幅提高的版本[^kazemi2025bbeh]。BBEH 论文是 2025+ benchmark 重写浪潮的早期范例：不是"扩大题量"，而是"对位重写以保留 capability dimension 而拉升难度"。

社区 issue tracker 还报告 date_understanding / geometric_shapes 题的 ground-truth 标错（Issue #14, #15, 2025-12 报）[^bbh-issues]，加上 BIG-Bench 原始数据在 GitHub 长期公开、任务粒度极不均（boolean_expressions 50 题、hyperbaton 250 题，算术平均不加权），BBH 的统计稳健性也被质疑。其历史角色——"CoT emergent gain 第一证据"——已经稳固进入文献，但作为 frontier benchmark 的实际效用已让位给 BBEH。

五、TruthfulQA：inverse scaling 反转与 truthfulness ≠ honesty

TruthfulQA（Lin, Hilton, Evans 2021）的设计是"测模型是否会复述人类常见错觉"——817 题刻意构造为"互联网上常见错误答案"形式，三种评测 mode（MC1 / MC2 / Generation + GPT-judge）[^lin2021truthfulqa]。最反直觉的发现是 inverse scaling：GPT-3 / T5 系列上参数越大 truthful% 反而下降，因为模型更"流利"地模仿了 internet falsehoods。这个论文在 RLHF / Alignment 早期文献中地位很高，是 "fine-tuning != imitation" 的源头之一。

但 TruthfulQA 的 inverse scaling 已被 post-RLHF 时代反转——frontier 模型的 MC2 接近 80%，原始 "scale = less truthful" 结论失效（事实上现在是"scale + RLHF = more truthful"）[^lin2021truthfulqa]。这本身不是 critique，而是 benchmark 寿命的自然现象。真正的 critique 来自三个 2025+ 工作：

第一是 **MASK Benchmark (Ren 等, 2025-03)**："Disentangling Honesty From Accuracy in AI Systems"。Ren 等明确把 truthfulness（说真话）与 honesty（不说谎）拆开：frontier 模型在 TruthfulQA 上拿高分，但同时"在压力下高频说谎"——即模型知道事实，但当压力 prompt 时选择性输出虚假信息[^ren2025mask]。MASK 用 1,000+ scenario probe 这两个维度，发现 TruthfulQA 高分模型在 MASK 上 honesty 维度的得分常远低于 truthfulness 维度。这意味着 **TruthfulQA 测的是 "are you accurate on common misconceptions" 而不是 "do you have a propensity to deceive"**。

第二是 turntrout 2024 揭示的 gaming heuristics：简单的 "I have no comment" / refusal 策略可以显著拉高 truthful%，因为 metric 不强制 informative[^turntrout2024]。论文的 Truthful × Informative 双指标在 leaderboard 上常被合并为 truthful%-only，给 refusal-heavy 模型不公平的优势。第三是 safetywashing critique：2025 综述把 TruthfulQA 列为"safetywashing 候

选"——把 TruthfulQA 高分等同于 safe 模型，但实际它只测特定 misconception，不测 deception、sycophancy、harm[^safetywashing2025]。

TruthfulQA 的故事综合了 inverse scaling 反转、metric gaming、construct validity ambiguity 三种 critique，是把"safety benchmark = single number ≠ honesty"讲透的最好案例，会在 §07 的安全评估部分被反复引用。

六、SIQA 与 ANLI：附录级 critique

SIQA (Sap 等 2019, EMNLP) 测 social commonsense——给场景 + 问题，三选一 [^sap2019socialiqa]；ANLI (Nie 等 2020, ACL) 是 adversarial NLI 三轮 human-and-model-in-the-loop[^nie2020anli]。两者在 §03 的角色相对附录化，因为：

- **SIQA**：Mousavi 等 2025 "Garbage In, Reasoning Out?" 系统标注 SIQA dev set，发现 **>28% 样本含数据缺陷**（4% structural、18% semantic、7% pragmatic）[^mousavi2025garbage]；Lyu 等 2021 同样证明 distractors 仍有 lexical leak。SIQA 是 SocialIQa-Garbage critique 的代表，但因不是 Open LLM Leaderboard 主指标，影响主要限于专题文献。
- **ANLI**："Lost in Inference" (2024) 等指出"adversarial 是对 BERT/RoBERTa 而言"——现代 LLM 直接刷过 R2/R3，许多"adversarial"样本不再是 hard cases；classical NLI 与现代 LLM 评估脱节[^lost2024nli]。ANLI license 是 CC-BY-NC（不允许商用 pretrain），仓库已于 2023-10 归档，事实上 benchmark 寿命也走向尾声。

两者共同提示一个深层模式：**当 benchmark 维护方停止 push 新版本、社区独立 critique 工作累积超过 30% 缺陷率证据时，benchmark 进入"legacy 维护"阶段**——仍可作为 small model 的 sanity check，但不再是 frontier signal。

七、小结：knowledge-era critique 的三层结构

把上述案例归纳成一个"critique stack"，对 §05 的饱和-污染辩论有承上启下的意义：

第一层 — annotation quality：MMLU-Redux 6.49%、HellaSwag 36% (Surge AI)、SIQA 28% (Mousavi)、AGIEval / MMLU-Pro 的 issue tracker 报告——这一层证明几乎所有 pretrain era 经典 benchmark 的原始标注质量在严格审计下都不及格。

第二层 — construct validity：GoldenSwag 的 ≥65% answer-only predictability、WinoGrande 的 paraphrase degradation、Global-PIQA 的 37pp cross-lingual gap、Invalid-CoT-still-gains 的 BBH critique、MASK 的 truthfulness ≠ honesty 拆解——这一层证明即便排除标注错，"benchmark 声称测的能力" 与"实际能预测的信号"之间存在显著 misalignment。

第三层 — data contamination + saturation：Sainz 2023 NLP contamination 综述、Llama 3 自报 52% NQ 渗漏、Lewis 2020 的 60-70% test-train overlap、frontier 模型在 ARC/HellaSwag/PIQA/BBH 上 >90% 饱和——这一层是 §05 的主战场，本章只作 cross-reference。

21 / 58

三层 critique 共同推动了 2024–2026 的"reasoning-era 转向"：MMLU → MMLU-Pro 是题量驱动尝试，BBH → BBEH 是难度驱动重写，HellaSwag → GoldenSwag 是清洗驱动子集，TruthfulQA → MASK 是维度拆解，PIQA → Global-PIQA 是跨文化扩展。每一种 successor 都对应一种 critique 的工程回应，但没有任何一种成为新的"通用 base eval"。换句话说，pretrain era 那种"单 benchmark = 单 capability 数字"的 reporting 范式正在自然解体——这正是第 4 章数学/代码/STEM benchmark 的设计研究背景，也是 §05 / §07 的辩论起点。

引用

[^hendrycks2021mmlu]: Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., & Steinhardt, J. (2021). *Measuring Massive Multitask Language Understanding*. ICLR 2021. arXiv:2009.03300.

[^sainz2023nlp]: Sainz, O. et al. (2023). *NLP Evaluation in Trouble: On the Need to Measure LLM Data Contamination for Each Benchmark*. EMNLP 2023 Findings. arXiv:2310.18018.

[^gema2024mmluredux]: Gema, A. P. et al. (2024). *Are We Done with MMLU?* arXiv:2406.04127.

[^holtzman2021surface]: Holtzman, A. et al. (2021). *Surface Form Competition: Why the Highest Probability Answer Isn't Always Right*. EMNLP 2021.

[^wang2024mmlupro]: Wang, Y., Ma, X., Zhang, G. et al. (2024). *MMLU-Pro: A More Robust and Challenging Multi-Task Language Understanding Benchmark*. NeurIPS 2024 D&B Track Spotlight. arXiv:2406.01574.

[^mmlupro-issues]: TIGER-AI-Lab. *MMLU-Pro GitHub Issues* (#69, #76, #79). https://github.com/TIGER-AI-Lab/MMLU-Pro/issues.

[^zhong2023agieval]: Zhong, W., Cui, R., Guo, Y. et al. (2023). *AGIEval: A Human-Centric Benchmark for Evaluating Foundation Models*. arXiv:2304.06364.

[^agieval-issues]: ruixiangcui. *AGIEval GitHub Issues* (#7, #11, #20, #29). https://github.com/ruixiangcui/AGIEval/issues.

[^joshi2017triviaqa]: Joshi, M., Choi, E., Weld, D. S., & Zettlemoyer, L. (2017). *TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension*. ACL 2017. arXiv:1705.03551.

[^kwiatkowski2019nq]: Kwiatkowski, T. et al. (2019). *Natural Questions: A Benchmark for Question Answering Research*. TACL 7. https://aclanthology.org/Q19-1026/.

[^lewis2020testtrain]: Lewis, P., Stenetorp, P., & Riedel, S. (2020). *Question and Answer Test-Train Overlap in Open-Domain Question Answering Datasets*. arXiv:2008.02637.

[^dubey2024llama3]: Dubey, A. et al. (2024). *The Llama 3 Herd of Models*. arXiv:2407.21783.

[^zellers2019hellaswag]: Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., & Choi, Y. (2019). *HellaSwag: Can a Machine Really Finish Your Sentence?* ACL 2019. arXiv:1905.07830.

[^surge2024hellabad]: Surge AI. (2024). *HellaSwag or HellaBad? 36% of this popular LLM benchmark contains errors*. https://surgehq.ai/blog/hellaswag-or-hellabad-36-of-this-popular-llm-benchmark-contains-errors.

[^chizhov2025goldenswag]: Chizhov, P., Nee, M., Langlais, P.-C., & Yamshchikov, I. P. (2025). *What the HellaSwag? On the Validity of Common-Sense Reasoning Benchmarks*. arXiv:2504.07825.

[^sakaguchi2019winogrande]: Sakaguchi, K., Le Bras, R., Bhagavatula, C., & Choi, Y. (2019). *WinoGrande: An Adversarial Winograd Schema Challenge at Scale*. AAAI 2020. arXiv:1907.10641.

[^elazar2021squareone]: Elazar, Y., Zhang, H., Goldberg, Y., & Roth, D. (2021). *Back to Square One: Artifact Detection, Training and Commonsense Disentanglement in the Winograd Schema*. EMNLP 2021. arXiv:2104.08161.

[^winowhat2025]: WinoWhat Authors. (2025). *WinoWhat: A Parallel Corpus of Paraphrased WinoGrande Sentences with Common Sense Categorization*. arXiv:2503.23779.

[^bisk2020piqa]: Bisk, Y., Zellers, R., Le Bras, R., Gao, J., & Choi, Y. (2020). *PIQA: Reasoning about Physical Commonsense in Natural Language*. AAAI 2020. arXiv:1911.11641.

[^chang2025globalpiqa]: Chang, T. A. et al. (2025). *Global PIQA: Evaluating Physical Commonsense Reasoning Across 100+ Languages and Cultures*. arXiv:2510.24081.

[^epoch2026piqa]: Epoch AI. (2026). *PIQA Benchmark Tracker*. https://epoch.ai/benchmarks/piqa.

[^clark2018arc]: Clark, P. et al. (2018). *Think You Have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge*. arXiv:1803.05457.

[^llama3card]: Meta. (2024). *Llama 3.1 Model Card*. https://github.com/meta-llama/llama-models/blob/main/models/llama3_1/MODEL_CARD.md.

[^suzgun2023bbh]: Suzgun, M., Scales, N., Schärli, N. et al. (2023). *Challenging BIG-Bench Tasks and Whether Chain-of-Thought Can Solve Them*. EMNLP Findings 2023. arXiv:2210.09261.

[^madaan2023invalidcot]: Madaan, A., & Yazdanbakhsh, A. (2023). *Invalid Logic, Equivalent Gains: The Bizarreness of Reasoning in Language Model Prompting*. arXiv:2307.10573.

[^kazemi2025bbeh]: Kazemi, M. et al. (2025). *BIG-Bench Extra Hard*. arXiv:2502.19187.

[^bbh-issues]: suzgunmirac. *BIG-Bench-Hard Issues* (#14, #15). https://github.com/suzgunmirac/BIG-Bench-Hard/issues.

23 / 58

[^lin2021truthfulqa]: Lin, S., Hilton, J., & Evans, O. (2022). *TruthfulQA: Measuring How Models Mimic Human Falsehoods*. ACL 2022. arXiv:2109.07958.

[^ren2025mask]: Ren, R. et al. (2025). *The MASK Benchmark: Disentangling Honesty From Accuracy in AI Systems*. arXiv:2503.03750.

[^turntrout2024]: turntrout. (2024). *Gaming TruthfulQA: Simple Heuristics Expose Dataset Weaknesses*. https://turntrout.com/original-truthfulqa-weaknesses.

[^safetywashing2025]: Safety-bench survey authors. (2025). *Critique of Safety Benchmarks Including TruthfulQA*. arXiv:2502.09387.

[^sap2019socialiqa]: Sap, M., Rashkin, H., Chen, D., Le Bras, R., & Choi, Y. (2019). *Social IQa: Commonsense Reasoning about Social Interactions*. EMNLP 2019. arXiv:1904.09728.

[^nie2020anli]: Nie, Y., Williams, A., Dinan, E., Bansal, M., Weston, J., & Kiela, D. (2020). *Adversarial NLI: A New Benchmark for Natural Language Understanding*. ACL 2020. arXiv:1910.14599.

[^mousavi2025garbage]: Mousavi, S. M., Cecchinato, E., Horníková, L., & Riccardi, G. (2025). *Garbage In, Reasoning Out? Why Benchmark Scores are Unreliable and What to Do About It*. arXiv:2506.23864.

[^lost2024nli]: Lost-in-Inference Authors. (2024). *Lost in Inference: Rediscovering the Role of Natural Language Inference for Large Language Models*. arXiv:2411.14103.

引子：从 GSM8K 到 ArXivLean 的设计演化

数学 / 代码 / STEM 三类 benchmark 是 LLM 评测在 2021 年后最活跃的赛道：题源边界清晰（grade-school math、competition problems、Python function completion、real GitHub PRs）、grader 客观（数值 EM、SymPy normalize、Docker execution、unit tests），让"benchmark = capability score"这一假设具有比 §03 知识 / 推理类更强的可信度。但同样因为这三个特点，pretrain corpus 中"数学解题 + 代码 PR + 习题答案"的天然存量也最大，contamination 在这里的破坏力最强；同时 frontier 模型本身在数学 / 代码上进步最快，benchmark 的"半衰期"最短。

本章按"题源演化"组织：先沿 GSM8K → MATH → MATH-500 → AIME → FrontierMath 串联难度递增轴，再讨论 GPQA 的"google-proof"承诺、HumanEval+/EvalPlus 的测试用例增强、LiveCodeBench 的 rolling 协议、SWE-bench-Verified 的"已死的学术意义"，最后给 MathArena Platform、LemmaBench、Soohak、EternalMath、Riemann-Bench 五个 2026 新 benchmark 一段集中讨论，归纳出"verifier 即 benchmark"这一研究趋势。本章的批评焦点是 benchmark 设计哲学本身——题源选择、grader 设计、governance 模式——而非工程操作，工程操作请回看 Part I 的对应章节。

24 / 58

一、数学难度递增轴：GSM8K → MATH → MATH-500 → AIME → FrontierMath

GSM8K：CoT 范式的"演示舞台"

GSM8K（Cobbe 等, OpenAI, 2021）的核心贡献其实有两个：8.5K 道小学应用题作为评测，以及"训练 verifier 给候选答案打分 + 排序"这一 verifier-based scaling 范式——后者是 ORM/PRM、process supervision、RLHF reasoning 的源头[^cobbe2021gsm8k]。GSM8K 设计上有三个有意为之的特征：(1) 题目可读但需 2–8 步运算；(2) 答案为整数便于 EM 自动判分；(3) 人类可达 100%（实际数据 ~95-98%）。chain-of-thought prompting（Wei 等 2022）的标志性证据就是在 GSM8K 上得到的，由此奠定了 GSM8K 在 2022-2024 几乎每篇 LLM 论文必报数字的地位。

但 2024-2025 GSM8K 进入"saturation + memorization"双重质疑。第一个关键 critique 是 Mirzadeh 等（Apple, ICLR 2025）的 **GSM-Symbolic**：把 GSM8K 题目里的姓名、物品、数值参数化重抽，frontier 模型 accuracy 下降，**插入单个不相关 clause 即可造成最多 65% 性能下降**[^mirzadeh2024gsmsymbolic]。这个结果直接挑战了"GSM8K 高分 = 数学推理"的等式——若模型真在做多步推理，则替换无关名词不应有任何影响。第二个是 Zhang 等 (Scale AI, NeurIPS 2024) 的 **GSM1k**：完全平行的新数据集，发现多个模型家族在 GSM1k 上分数比 GSM8K 低最多 8pp，是系统性 memorization 的实证证据[^zhang2024gsm1k]。

GSM8K 官方 repo 已于 2026-04 归档（read-only），多数前沿 math leaderboard（codesota / BenchLM）于 2026-05 把 GSM8K 从加权指标中移除。GSM8K 的剩余角色是 small / mid-train 阶段的 ablation sanity check，以及"reasoning vs memorization"批评的引用源头。

MATH 与 MATH-500：竞赛级 + 子集化的 PRM 评测平台

MATH（Hendrycks 等 2021）从 GSM8K 跳一阶难度——AMC / AIME 量级的 12,500 题，附 5M 题的 AMPS 辅助预训练集（OpenWebMath / FineMath / Proof-Pile 的精神祖先）[^hendrycks2021math]。论文核心论断："仅扩大规模无法解决竞赛数学，需要新的归纳偏置"。题目设计哲学三点：七学科均衡、5 级难度细粒度 ability profile、AMPS 把 "数学预训练" 概念系统化。

MATH-500（Lightman 等, OpenAI, 2023, "Let's Verify Step by Step"）是 PRM800K 训练 process reward model 后从 MATH test set 抽出的 500 题分层子集，目的是降低 PRM best-of-N 评估的计算成本[^lightman2023verifystep]。500 这个数字是经验工程权衡（小到 1-pass eval、大到统计稳定），OpenAI 内部 ablation 验证其代表性。社区随后把 MATH-500 当作 MATH 的轻量替代，并在 R1（DeepSeek）/ o1 时代成为 reasoning RL 训练曲线的标准 in-process metric。

MATH / MATH-500 共享相同的 critique stack：

第一，**functional perturbation**。Srivastava 等 2024 的 MATH() 把题目参数化，在前沿模型上观察到 58–80% 的 accuracy drop，与 GSM-Symbolic 同源

25 / 58

[^srivastava2024functional]。这意味着 MATH 高分相当一部分源于 surface form pattern matching 而非真推理。

第二，**training set contamination**。MATH 训练集 7,500 题已被广泛纳入 LLM 预训练 corpus。BenchmarkingAgents 2026 的 partial-prompt completion 攻击显示 Qwen2.5-Math-7B 在 MATH-500 部分前缀的补全任务上以 54.6% EM 重建出剩余 40%，是大规模记忆的实证[^benchmarkingagents2026]。

第三，**saturation + indistinguishability**。GPT-5 (high) 在 MATH-500 上已 99.4%[^artificialanalysis-math500]，o3 (high) 在 full MATH 上 98.1%[^codesota-math]，frontier 模型之间的差距已落入采样噪声。BenchLM math leaderboard 已把 MATH 标为"effectively saturated"，主要用作参考而不再加权[^benchlm-math]。

MATH-500 的剩余学术意义在于：它是"PRM-style process supervision 评测"的事实标准格式，新一代 process reward 模型（如 PRMBench、ProcessBench）几乎都在 MATH-500 上做对照，这一格式惯性会延续到 reasoning model 时代。

AIME 2024 / 2025：竞赛真题的短窗口寿命

AIME 2024 / 2025 不是独立的"研究 benchmark"，而是把 American Invitational Mathematics Examination 两届真题（AIME I + II × 2 = 60 题）借作竞赛级数学的标准短卷[^dekoninck2026matharena]。它的"被 LLM 化"路径很典型：

- **2023 末**：OpenAI o1 preview 演示 AIME 2024 ~74%（pass@1），首次让 AIME 进入 LLM 报告卡。
- **2024-2025**：每个 frontier model 必报 AIME 2024/2025，AIME 成为 "reasoning 范式效力" 的公允指标。
- **2025-2026**：AIME 2024 在 o1 训练截止之后但 GPT-5 / Gemini-3 之前，已被广泛纳入预训练 + RL；AIME 2025 给 2025 年初截止训练的模型留下了 ~6 个月"未污染窗口"，到 2026 时也被吸收。
- **2026-05**：MathArena 平台正式以 AIME 2026 / USAMO 2026 接班，AIME 2024/2025 从"frontier metric"转为"legacy metric"。

AIME 的方法论缺陷有三：(1) 60 题（30/届）pass@1 粒度仅 ~3.3pp，统计 noise 大；MathArena 建议至少 32 次采样取平均；(2) 题与解大量出现在 Discord / AoPS / Codeforces 论坛抓取的训练 corpus 中，污染基本不可避免；(3) AIME final-answer 与 USAMO / IMO proof-based 形成"final-answer vs proof"的 false dichotomy，前者更易污染、不测证明能力。MathArena 团队在 2026-05-12 "Farewell to Final-Answer" 博文中明确把 AIME 类 final-answer benchmark 定为"effectively solved by AI"，正式 deprecate[^matharena-farewell]。

AIME 2024/2025 的故事浓缩了"竞赛真题 benchmark 寿命"的一般规律：**竞赛公开当日起，污染窗口开始关闭；6–18 个月后基本进入主流预训练 corpus；只能作为短期 contamination diagnostic 而非长期 capability metric**。

26 / 58

FrontierMath：研究级 + governance 争议 + 自爆 1/3 fatal errors

FrontierMath（Epoch AI, Glazer 等, 2024-11）的设计目标是"占据 AI 数学家曲线最右端"——研究者数小时（Tier 1-3）甚至数周（Tier 4）才能解的题，所有题为 unpublished originals 以阻断训练污染[^glazer2024frontiermath]。论文发布时 SOTA ~2%，确立"绝对 ceiling"形象。但 FrontierMath 经历了 2024-12 至 2025-01 的"FrontierMath scandal"——这是 2024-2025 benchmark governance 讨论中最重要的案例。

事件主线：(1) OpenAI 资助了 FrontierMath 开发，并对大部分题目（除一小撮独立 hold-out）拥有访问权；(2) Epoch AI 未在 contributor 招募阶段披露 OpenAI 资助，仅靠口头承诺 OpenAI 不用题做训练；(3) OpenAI 同时在 o3 发布会上把 FrontierMath 25% 成绩作为里程碑宣传。社区指控包括题与解的潜在污染、benchmark independence 受损、contributor 知情同意不充分[^harris2025scandal][^lesswrong2025lessons]。Hacker News 与 Silicon Reckoner 的讨论催生了后续 Riemann-Bench / Soohak 等纯独立 governance benchmark 的设计动机[^hn2025frontier]。

更严重的打击发生在 2026-05-11：**Epoch AI 公告 AI-assisted review 发现 Tier 1-4 中约 1/3 题目存在"fatal errors"**，目前正在人工复核，并将基于修正后题集重新发布分数；现行 leaderboard 数字应被视为 provisional[^epoch-tier4-2026]。这是 benchmark 自爆错率最严重的一次——比 MMLU-Redux 的 6.49% 高一个数量级。原因部分是研究级数学题本身难以验证（即便专家双盲验证也可能漏掉边界情况），部分是 hold-out 协议导致 community-wide 错误反馈循环非常慢。

FrontierMath 的学术意义现在主要不在分数，而在 governance 案例本身：sponsor disclosure norms、hold-out set 治理、contributor 知情同意、benchmark independence——这些议题在 §07 未来辩论中会作为核心案例反复出现。从设计角度看，FrontierMath 暴露了"研究级 + 私有 + 单源资助"组合的脆弱性，直接催生 Soohak（多 contributor）、Riemann-Bench（最小作者集 + 完全私有）、MathArena（公开 + 月更）三种 governance 替代方案。

二、GPQA-Diamond：科学版的"google-proof"承诺真伪

GPQA（Rein 等, NYU + Cohere + Anthropic, 2023-11）是研究生水平、声称 "Google-proof" 的 STEM MCQ benchmark；Main 448 题，**Diamond 子集 198 题**——筛选标准是"两位 PhD 都答对且 ≤1/3 高水平非专家答对"[^rein2023gpqa]。论文核心证据有三：(1) 专家命题 + 专家验证保证高难度；(2) 非专家开放 web 30+ 分钟仅 34%，专家 65%（剔除自报错后 74%），实证 "Google-proof"；(3) MCQ 形式便于自动评测但仍难。GPT-4 baseline 39%、CoT 47%，远低于 PhD 65%，把 LLM 与 PhD 差距具体量化。OpenAI o1 (2024-09) / o3 (2024-12) 用 GPQA-Diamond 作"博士级推理"标杆，开启 reasoning models 概念。

进入 2025-2026 后 GPQA-Diamond 进入饱和——Gemini 3.1 Pro 在 94.1%，frontier 模型差距已 1-2pp 落入采样噪声[^epoch-gpqa]。这本身不是问题，但 Epoch AI 的 Burnham 2025 "GPQA Diamond: What's Left?" 给出了更深的方法论 critique：**Organic Chemistry 占 Diamond 难题 70%（实际题量仅 36%），存在严重领域不平衡**[^burnham2025gpqa]。意思是"Diamond 的难"在很大程度上由 organic chemistry 的几十题决定，前沿模型在其他子

27 / 58

领域（如 Bio、Inorganic Chem、Physics 高级题）的差距远小。这一发现把"GPQA 测的是 PhD-level reasoning"细分为"GPQA 测的是 organic chemistry + 部分 quantum mechanics 的 PhD-level reasoning"。Epoch 同时标注 ~10% 问题有效性可疑（如 stellar spectroscopy 过约束、bioinformatics 题主观）[^burnham2025gpqa]。

GPQA 还有一个 "google-proof 承诺" 的本身 critique：当 frontier 模型 >90% 时，LLM 已超过非专家 + Google 的 34% 上限 3 倍多——这意味着 "google-proof" 的原始定义（"非专家用 Google 也答不出"）对 LLM 不再有意义。需要 v2 或 held-out server，但 2026-05 仍无 v2。Diamond 仅 198 题，1-2pp 差距已在 binomial CI 噪声范围内[^intuitionlabs-gpqa]。

GPQA 的故事提示一个普遍现象：**MCQ-based 难题 benchmark 的寿命受限于 organic chemistry / quantum mechanics 这种"题源天然稀缺 + 答案唯一可验证"的子领域容量**。一旦该子领域被刷透，整个 benchmark 的剩余区分度迅速塌缩。这与第 §03 的 MMLU-Pro 路径很像（扩选项也只是延寿，不是解构造效度）。

三、代码：HumanEval/MBPP → EvalPlus → LiveCodeBench

HumanEval：从"unlikely to be in training data"到 saturation

HumanEval（Chen 等, OpenAI, Codex paper, 2021）的 164 道 Python function-completion 题是 execution-based code eval 的起点[^chen2021codex]。设计动机："unlikely to be in the training data"——OpenAI staff 手写以避免 LeetCode / Codeforces 类公开题库的污染，并采用 pass@k 这个数学上有 unbiased estimator 的指标。HumanEval 配 MBPP（Austin 等, Google, 2021）的 ~1,000 crowdsourced "basic Python problems" 共同成为 2021-2024 code-LM 的事实标准[^austin2021mbpp]。

但 HumanEval/MBPP 的两层问题在 2023-2024 被系统化暴露。第一层是 **HumanEval+ / EvalPlus** (Liu 等 2023)：HumanEval 原 test suites 太稀疏（164 题平均不到 10 个测试用例），EvalPlus 用 mutation testing + LLM 生成把测试规模扩 80×，发现 frontier 模型 pass@k 下降 19.3–28.9%，并 re-rank 了多个模型[^liu2023evalplus]。同理 MBPP+ 把每题 3 个测试扩 ~35×，下降 5-15pp。这意味着 HumanEval/MBPP 的"功能正确性"只是稀疏测试下的功能正确性——加几个 corner case 测试，大半"通过"的代码就崩了。EvalPlus 自此成为 code-LM 测试稳健性的事实标准格式。

第二层是 **contamination**。Riddell 等 2024 量化了 HumanEval / MBPP 与开源预训练 corpus 的 surface + semantic-level 重叠，发现 memorized subset 上分数显著高于 non-memorized subset，是污染主导分数的实证[^riddell2024contamination]。Jain 等 2024 (LiveCodeBench paper) 直接断言 HumanEval/MBPP "are no longer sufficient"——当 frontier 模型 pass@1 >95% 时，benchmark 不再能区分能力[^jain2024livecodebench]。

到 2026 年 HumanEval / MBPP 是"compliance baseline"——在 frontier model release card 中仍出现，但没有 ranking 意义。HumanEval 的历史角色是 execution-based eval 范式的奠基（Codex paper 的 pass@k unbiased estimator 至今仍是几乎所有 code benchmark 的 metric），EvalPlus 是它最重要的科学延伸。

28 / 58

LiveCodeBench：rolling-release 的范式贡献

LiveCodeBench（Jain 等, Berkeley, 2024-03）的核心设计创新不是题源（LeetCode/AtCoder/Codeforces 都不新鲜），而是 **release-date filtering protocol**——每题标 release date，评估时按模型的 training cutoff 过滤只跑 post-cutoff 题，从而把"contamination-free 评估"变成一个可机器执行的协议[^jain2024livecodebench]。这一协议被 SWE-bench Live、SWE-Rebench、MathArena、ArxivMath 全部继承，是 2024-2026 live benchmark 范式的祖先。LiveCodeBench 版本演进：v1 (May 2023–Mar 2024, 400 题) → v6 (–Apr 2025, 1,055 题)，月度刷新延续到 2026。

LiveCodeBench 的局限也同时来自这个 protocol。第一，源平台 (LeetCode 等) 本身泄漏——解决方案在 GitHub README、Discord、blog 上数日内出现，contamination-free 保证需要严格 cutoff filter。第二，competitive programming 题分布偏向 algorithmic puzzle，不反映 repo-level / agentic engineering；与 SWE-Bench-class 应配对使用。第三是 **LiveCodeBench Pro** (2025-06) 的发现：IOI/ICPC medalist 标注下，frontier 模型在 medium 题仅 53%，hard 题 0%[^livecodebench-pro-2025]。LCB-Pro 揭示 LCB 系统性"过度信用 implementation polish"——能写出能 compile + 部分通过的代码，但缺乏深度 algorithmic reasoning。

LiveCodeBench 是 §06 live benchmark 范式的关键例子，本章只点出它作为 HumanEval/MBPP 接班的代际跃迁。

SWE-bench-Verified：已死的学术意义

SWE-bench-Verified 的故事最戏剧化。原 SWE-bench (Jimenez 等, ICLR 2024) 把真实 GitHub issue + PR pair 包装为 repo-level engineering task，Claude 2 当时仅 1.96% [^jimenez2024swebench]。OpenAI 2024-08 release Verified 子集——500 题经 93 位软件工程师人工 audit，剔除 ambiguous spec / broken test / unsolvable issue。Verified 迅速成为 2024 H2 至 2025 的 de facto agent coding benchmark：Claude 3.5 Sonnet 40-50%、mid-2025 顶级 70%、Claude Opus 4.5 报 80.9%。

但 2025-2026 Verified 的 validity 完全崩塌：

第一，**SWE-Bench+** (Aleithan 等 2024) 系统审计 Verified 的 "passing" patches：**32.67% 涉及 cheating**（解决方案出现在 issue report / comments 中，模型直接抄）；**31.08% pass 由于 weak tests**（测试不严格，错误 patch 也能 pass）；过滤后 SWE-Agent+GPT-4 从 12.47% 掉到 3.97%[^aleithan2024swebenchplus]。

第二，**OpenAI 内部 audit**：每个 frontier 模型都能 reproduce verbatim gold patches，是 contamination 的直接证据；同时 59.4% "最难未解" 问题的 test case 本身有缺陷 [^morphllm2026]。这一 audit 公开后，OpenAI 在 2026 Q1 公开停止报告 Verified 分数。

第三，**Pro vs Verified 16-35pp 差距**：Scale AI 的 SWE-Bench Pro 上每个 frontier 模型分数都比 Verified 低 16-35pp，例如 Claude Opus 4.5 在 Verified 80.9% vs Pro 45.9%。这个差距被直接归因为 Verified 的 contamination[^scale2026pro]。

29 / 58

SWE-bench-Verified 在 2026 已被全行业 deprecate，但它的学术意义并未消失——**它是 "static repo-level benchmark 在 12-18 个月内被攻陷" 的最干净案例**。它的 contamination 路径与 MMLU 类 MCQ 不同：Verified 题的 GitHub issue + PR 在 web crawl 中以"issue body + comment + final commit + test"的完整形式出现，是 LLM 训练时最容易复用的"配对监督信号"。这一点直接驱动 SWE-Bench Pro 的设计——GPL / 私有 / proprietary 代码 + 多语言 + 长 horizon multi-file patch 的组合，物理上阻断训练 corpus 渗透。

SWE-Bench Pro：multi-language repo-level engineering

SWE-Bench Pro (Scale AI, 2026) 是 Verified 的 contamination-resistant 接班：1,865 任务跨 Python/Go/TypeScript/JavaScript 41 仓库，分 Public (731, GPL/copyleft) / Private (276, 创业公司专有代码) / Held-out (858, 季度刷新) 三档，任务平均 107 LOC × 4.1 files[^scale2026pro]。设计哲学四点：(1) GPL/copyleft license 通常被预训练 data filter 排除；(2) Private 集是 unreachable 的 proprietary code；(3) 实际工程难度（保留 ambiguous spec）；(4) 可复现 Docker testing。

Pro 的 Public vs Private subset 差距本身被作为 contamination diagnostic 信号——Claude Opus 4.1 在 Public 22.7% vs Private 17.8%，4.9pp delta 暗示训练 corpus 有一定 leakage 但远小于 Verified 时代。Pro 的 critique 集中在三点：(1) scaffolding sensitivity（mini-SWE-agent vs Claude Code vs Devin 等 swing 10+pp，agent harness 与 model 仍纠缠）；(2) 仅 4 语言，无 Rust/C++/Java；(3) Held-out 858 题不可外部 audit。这些是"contamination-resistant successor"必然要付的代价——透明度 vs 抗污染的 trade-off。

四、2026 数学新基准三足：MathArena / LemmaBench / Soohak

2026 年数学 benchmark 进入"benchmark-as-platform"时代，由三个独立项目分担"live research math"评测的不同维度。

MathArena Platform：multi-stream rolling

MathArena (ETH Zürich SRI Lab, Dekoninck 等, 2026-05) 把 benchmark 升级为持续维护的评测基础设施，5 大并行子流：(1) Final-Answer Competitions（AIME 2026 / HMMT / BRUMO / SMT / Apex）；(2) Proof-Based（USAMO 2026 / IMO / Putnam / IMC / Miklós Schweitzer）；(3) ArxivMath（每月 ~40 题，从 30 天 arXiv 数学新论文 abstract 中提炼 final-answer 题）；(4) ArXivLean（Lean 4 形式证明）；(5) BrokenArxiv（含错误陈述以测推理鲁棒性）[^dekoninck2026matharena]。设计动机有三：(a) 静态竞赛 saturation；(b) Petrov 等 2025 "Proof or Bluff?" 发现 frontier 模型在 AIME 2025 拿 90%+ 而在 USAMO 2025 proof 评分仅 ~5%——final-answer 与 proof 是两个能力维度需要分离评测 [^petrov2025proofbluff]；(c) arXiv 月更内容是天然 post-cutoff 题源。

MathArena 在 2026-05-12 发布"Farewell to Final-Answer"博文，主动宣布连自家 flagship Apex set 都已被 GPT-5.5 / Gemini 3.1 Pro 饱和，主推 proof / research / broken 三流

[^matharena-farewell]。这是"benchmark 维护方主动 deprecate 自家旗舰"的少见案例，是 §05 saturation 讨论的标志性事件。

MathArena 的方法论意义在于：它把"single static dataset → 单数字 leaderboard" 的 reporting 范式替换为"multi-stream platform → 能力维度 profile"。每月 ArxivMath 仅 ~40 题，置信区间宽（±5pp 量级），但多月平均稳定；proof judge 依赖 LLM-judge 双判 + 人工抽检，仍有判分主观性[^moonlight-matharena]。

LemmaBench：自然语言 proof + LLM-judge

LemmaBench (Peyronnet, Gloeckle, Hayat, 2026-02) 走的是 ArxivMath 的姊妹路线——每月从 arXiv 数学预印本抓 lemma，通过 LLM 管道把"散落于论文上下文的隐式假设"补齐为自包含命题，再让 LLM 生成 proof，由独立 LLM-judge 逐步判定 [^peyronnet2026lemmabench]。2026-02 iteration 涵盖 677 lemma（81 预印本，~60% 通过 self-contained filter）。SOTA 自报 ~10-15% pass@1。

LemmaBench 的关键设计选择是 **autoformalize-lite**——不要求 Lean 等形式语言，而是用 LLM 把 lemma 改写为"自包含"自然语言陈述。这是相比 ArXivLean / miniF2F 的轻量化方案，trade off 是 LLM-judge 引入主观性（44 条 spot-check 中 judge 与人类一致率 67-83%）。另一个隐患是 generator + judge 同源——GPT-5 generator + GPT-5 judge 报 15% vs Gemini 2.5 Pro 仅 7%，需小心 leaderboard 解读。LemmaBench 也继承了 "arXiv lemma 假设 all preprints are correct" 的 sycophancy 隐患，未单独标识可疑 lemma——这是 BrokenMath / BrokenArxiv / Soohak Refusal 几个项目所揭示的。

Soohak：64 数学家众包 + Refusal subset 创新

Soohak (Son, Kim, Arnett 等 64 位数学家, 2026-05) 由两个子集组成：**Soohak Challenge** (340 题，研究级)、**Soohak Refusal** (99 题，故意构造矛盾 / 缺失假设 / 无解的"病态题"，测模型是否盲信题目)，另加 Soohak-Mini (702 题, olympiad-graduate)。当前 SOTA：Gemini-3-Pro 30.4%（Challenge），GLM-5 49.5%（Refusal, 无模型 >50%）[^son2026soohak]。

Soohak 设计的方法论意义体现在两点：(1) **64 位数学家众包**回应 FrontierMath "单源资助方"风险；(2) **Refusal subset 把 sycophancy 从 probe 升级为 first-class evaluation**——99 题各有明确"病因"（矛盾陈述 / 缺信息 / 不存在解 / 错误存在性），ground truth 是模型应输出"this problem has issue X"或拒绝作答。Refusal subset 的灵感来自 BrokenMath (2025-10)[^brokenmath2025] 和 MathArena BrokenArxiv，但 Soohak 是第一个把 refusal 作为独立子集 + 独立 metric 的工作。

The Decoder 2026-05 的报道点出关键：**无模型在 Refusal 上 > 50%**，Qwen3 家族 < 3%——表明 frontier 模型对"病态题"的元认知防御接近 0[^thedecoder2026soohak]。这是 2026 年 §07 安全 / 元认知讨论的核心实证。

31 / 58

EternalMath 与 Riemann-Bench：补全光谱两极

EternalMath (Ma 等, 2026-01 v1 → 2026-05 v2, 782 题) 是另一种 live math benchmark——只挑论文中"可执行验证"的构造性 / 定量结果，把它们改写成参数化模板（题中常数可重抽），用 SymPy / 数值 oracle 自动生成 ground truth[^ma2026eternalmath]。失去 proof 维度但获得 scoring 客观性。EternalMath 还贡献了 9-category failure mode taxonomy（knowledge gap → hallucination chain、boundary neglect、calculation drift 等），错误"常以 compounding chain 形式发生"——一个 knowledge gap 触发后续 hallucination + calculation drift + boundary neglect，这一观察后续被 BrokenMath / Soohak 引用作为"over-confidence cascade"的实证。

Riemann-Bench (Garre, Knutsen, Mehta, Chen, 2026-04, 25 题私有) 走极端——4 人作者团队、全私有、每题由 Ivy League 教授 / IMO 奖牌得主撰写数周，所有 frontier 模型 < 10% [^garre2026riemannbench]。设计哲学："作者越少 → 内部串谋几率越低 → governance 越纯净"。与 Soohak (64 作者) 形成方法论对照——多 contributor 提升广度但代价是 contributor 间共识协议成本；最小作者集提升纯净度但失去代表性。Riemann-Bench 是 ceiling indicator ("AI 何时跨越 10%")，n=25 不能做细分排名。

五、"verifier 即 benchmark" 的研究趋势

把以上五个 2026 新基准（MathArena 多流、LemmaBench、Soohak、EternalMath、Riemann-Bench）放一起看，可以观察到一个共同的设计哲学转向：**benchmark 的核心不再是题集，而是 verifier**。具体来说，benchmark 设计者花在"题源 mining + 自动验证管道"上的工程量已显著超过"人工出题"——MathArena 的 arXiv abstract mining + LLM-judge 双判、LemmaBench 的 self-containment filter + step-wise LLM-judge、EternalMath 的 template instantiation + SymPy oracle、Soohak Refusal 的 ill-posed 标注 + judge 抽取，都是"verifier 即 benchmark"的实例。

这一趋势有四个直接含义：

1. **题源边际成本骤降**。当 verifier pipeline 工程完成后，每月可低成本添加数十题，benchmark 寿命从"静态 1-2 年"变为"理论上无限"。代价是 verifier 自身的可信度成为新的方法论瓶颈。
2. **judge / verifier 的可靠性成为关键 critique 焦点**。LemmaBench 自报 67-83% LLM-judge 一致率、MathArena proof judge 主观性、Soohak Refusal 的 LLM-judge 抽取——这些将主导 2026 中后期的方法论讨论，预计 ICLR/NeurIPS 2026 会出现专门"benchmark verifier audit"主题工作（appendix 章已观察到 PAM / JECS / Prescriptive Scaling 三个相关 framework）。
3. **process reward model (PRM) 与 benchmark verifier 的界限模糊**。GSM8K 的 verifier 范式、MATH-500 的 PRM 评测、LemmaBench 的 step-wise LLM-judge——它们共享同一套"为 reasoning chain 打分"基础设施，未来 benchmark 与 reward model 训练数据可能更紧密共生。

32 / 58

4. **governance / contributor consent / sponsor disclosure 上升为方法论一级议题**。FrontierMath scandal 之后，benchmark 论文必须公开 funding、access agreement、hold-out 协议；Soohak / Riemann-Bench / MathArena 已把这些写进论文的 prominent position，未来 NeurIPS D&B Track 评审将把 governance 作为 first-class criterion。

这五个 2026 新基准与第 §03 的知识 / 推理类 benchmark 相比，最大的不同不是难度，而是 **benchmark 设计本身从"出题"转向"出 pipeline"**。这是 LLM evaluation 方法论的代际跃迁。后续章节（§05 saturation/contamination、§06 live/agent paradigm、§07 future debate）将在这一基础上展开。

引用

[^cobbe2021gsm8k]: Cobbe, K. et al. (2021). *Training Verifiers to Solve Math Word Problems*. arXiv:2110.14168.

[^mirzadeh2024gsmsymbolic]: Mirzadeh, I. et al. (2024). *GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models*. ICLR 2025. arXiv:2410.05229.

[^zhang2024gsm1k]: Zhang, H. et al. (2024). *A Careful Examination of Large Language Model Performance on Grade School Arithmetic*. NeurIPS 2024 D&B. arXiv:2405.00332.

[^hendrycks2021math]: Hendrycks, D. et al. (2021). *Measuring Mathematical Problem Solving with the MATH Dataset*. NeurIPS 2021 D&B. arXiv:2103.03874.

[^lightman2023verifystep]: Lightman, H. et al. (2023). *Let's Verify Step by Step*. arXiv:2305.20050.

[^srivastava2024functional]: Srivastava, S. et al. (2024). *Functional Benchmarks for Robust Evaluation of Reasoning Performance*. arXiv:2402.19450.

[^benchmarkingagents2026]: BenchmarkingAgents. (2026). *What LLM Benchmarks Don't Measure*. https://benchmarkingagents.com/what-these-benchmarks-miss/.

[^artificialanalysis-math500]: Artificial Analysis. (2026). *MATH-500 evaluations*. https://artificialanalysis.ai/evaluations/math-500.

[^codesota-math]: CodeSOTA. (2026). *MATH benchmark leaderboard*. https://www.codesota.com/benchmark/math.

[^benchlm-math]: BenchLM.ai. (2026). *Math leaderboard*. https://benchlm.ai/math.

[^dekoninck2026matharena]: Dekoninck, J. et al. (2026). *Beyond Benchmarks: MathArena as an Evaluation Platform for Mathematics with LLMs*. arXiv:2605.00674.

[^matharena-farewell]: MathArena. (2026-05-12). *Farewell to Final-Answer Competition Problems as Frontier Benchmarks*. https://matharena.ai/no_final_answer/.

33 / 58

[^glazer2024frontiermath]: Glazer, E. et al. (2024). *FrontierMath: A Benchmark for Evaluating Advanced Mathematical Reasoning in AI*. arXiv:2411.04872.

[^harris2025scandal]: Harris, M. (2025). *The FrontierMath Scandal*. Silicon Reckoner. https://siliconreckoner.substack.com/p/the-frontier-math-scandal.

[^lesswrong2025lessons]: LessWrong. (2025). *Some Lessons from the OpenAI–FrontierMath Debacle*. https://www.lesswrong.com/posts/8ZgLYwBmB3vLavjKE/some-lessons-from-the-openai-frontiermath-debacle.

[^hn2025frontier]: Hacker News. (2025-01). *FrontierMath was funded by OpenAI*. https://news.ycombinator.com/item?id=42763231.

[^epoch-tier4-2026]: Epoch AI. (2026). *FrontierMath Tier 4 Benchmark Page*. https://epoch.ai/benchmarks/frontiermath-tier-4.

[^rein2023gpqa]: Rein, D. et al. (2023). *GPQA: A Graduate-Level Google-Proof Q&A Benchmark*. arXiv:2311.12022.

[^epoch-gpqa]: Epoch AI. (2026). *GPQA-Diamond Benchmark Tracker*. https://epoch.ai/benchmarks/gpqa-diamond.

[^burnham2025gpqa]: Burnham, G. (2025). *GPQA Diamond: What's Left?*. Epoch AI Gradient Updates. https://epoch.ai/gradient-updates/gpqa-diamond-whats-left.

[^intuitionlabs-gpqa]: Intuition Labs. (2026). *GPQA-Diamond Benchmark: Scores, Leaderboard & How AI Models Compare*. https://intuitionlabs.ai/articles/gpqa-diamond-ai-benchmark.

[^chen2021codex]: Chen, M. et al. (2021). *Evaluating Large Language Models Trained on Code*. arXiv:2107.03374.

[^austin2021mbpp]: Austin, J. et al. (2021). *Program Synthesis with Large Language Models*. arXiv:2108.07732.

[^liu2023evalplus]: Liu, J. et al. (2023). *Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models for Code Generation* (EvalPlus). arXiv:2305.01210.

[^riddell2024contamination]: Riddell, M. et al. (2024). *Quantifying Contamination in Evaluating Code Generation Capabilities of Language Models*. arXiv:2403.04811.

[^jain2024livecodebench]: Jain, N. et al. (2024). *LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code*. arXiv:2403.07974.

[^livecodebench-pro-2025]: LiveCodeBench Pro Team. (2025). *LiveCodeBench Pro: How Do Olympiad Medalists Judge LLMs in Competitive Programming?* arXiv:2506.11928.

[^jimenez2024swebench]: Jimenez, C. E. et al. (2024). *SWE-bench: Can Language Models Resolve Real-World GitHub Issues?* ICLR 2024. arXiv:2310.06770.

34 / 58

[^aleithan2024swebenchplus]: Aleithan, R. et al. (2024). *SWE-Bench+: Enhanced Coding Benchmark for LLMs*. arXiv:2410.06992.

[^morphllm2026]: Morph LLM. (2026). *SWE-Bench Pro vs Verified — Contamination Gap*. https://www.morphllm.com/swe-bench-pro.

[^scale2026pro]: Scale AI. (2026). *SWE-Bench Pro: A Stronger Coding-Agent Benchmark*. https://scale.com/blog/swe-bench-pro.

[^petrov2025proofbluff]: Petrov, I. et al. (2025). *Proof or Bluff? Evaluating LLMs on 2025 USA Math Olympiad*. arXiv:2503.21934.

[^moonlight-matharena]: Moonlight Review. (2026). *Literature Review of MathArena*. https://www.themoonlight.io/en/review/matharena-evaluating-llms-on-uncontaminated-math-competitions.

[^peyronnet2026lemmabench]: Peyronnet, A., Gloeckle, F., Hayat, A. (2026). *LemmaBench: A Live, Research-Level Benchmark to Evaluate LLM Capabilities in Mathematics*. arXiv:2602.24173.

[^son2026soohak]: Son, G., Kim, S., Arnett, C. et al. (2026). *Soohak: A Mathematician-Curated Benchmark for Evaluating Research-level Math Capabilities of LLMs*. arXiv:2605.09063.

[^brokenmath2025]: BrokenMath Authors. (2025). *BrokenMath: A Benchmark for Sycophancy in Theorem Proving with LLMs*. arXiv:2510.04721.

[^thedecoder2026soohak]: The Decoder. (2026-05). *New Math Benchmark Reveals AI Models Confidently Solve Problems That Have No Solution*. https://the-decoder.com/new-math-benchmark-reveals-ai-models-confidently-solve-problems-that-have-no-solution/.

[^ma2026eternalmath]: Ma, J. et al. (2026). *EternalMath: A Living Benchmark of Frontier Mathematics that Evolves with Human Discovery*. arXiv:2601.01400.

[^garre2026riemannbench]: Garre, S., Knutsen, E., Mehta, S., Chen, E. (2026). *Riemann-Bench: A Private Benchmark of Moonshot Mathematics Problems*. arXiv:2604.06802.

5.1 饱和现象：从单点崩溃到曲线整体右移

LLM 评测三十年的方法论史里，2023 → 2026 是第一次出现"饱和"成为**结构性问题**而非个别 benchmark 偶发的死亡。第 3、4 章已经从知识与推理两条线分别证明：MMLU 在 GPT-4 / Claude 3 时代封顶在 88–89% 区间（与 MMLU-Redux 6.49% 错题率高度耦合，事实上人类专家上限就在 90 上下）[^mmlu-redux]；HumanEval 在 EvalPlus 增 80× 测试后 pass@1 仍可达 99% [^evalplus]；GSM8K 在 2026-Q1 已被新一代代码 / 工程模型推到 99%+ 行业共识。把这三条线拼起来观察，**饱和并非"题难度低估"的偶发症状，而是 web-scale 公开 benchmark 的必然命运**。

35 / 58

但 2026 视角下饱和与 2023 视角已经发生了三处微妙变化，影响后续学术辩论的方向：

第一是**饱和的纵深**。HumanEval、MMLU、GSM8K 在不同时刻饱和但理由不同 — HumanEval 是 164 题样本规模决定其早死，MMLU 是错题与 prompt 敏感性把上限锁在 90 以下，GSM8K 是叙事数学的难度面（按 LiveBench 维度看是"language + arithmetic"的简单组合）已被 reasoning-era 模型轻松扫平。任何宣称"benchmark 饱和"的论断必须解构到底是**ceiling effect、annotation noise、还是真实能力外推**这三者中的哪一类。

第二是**饱和的曲线形态**。LiveBench 数据呈现一个反直觉但关键的现象：原本应当 "delay saturation" 的 contamination-limited rolling benchmark 自身也开始出现头部聚类 — 2026-05 leaderboard 上 o3-mini 0.846 / Qwen3-235B 0.771 / Kimi K2-Instruct 0.764 已经挤进 0.75–0.85 区间[^livebench-llm-stats]。社区开始用 "plateau" 而非 "saturation" 描述这个状态[^plateau-2026]，意思是"区分度仍在但每月新题贡献的难度增量边际递减"。这给 §6 讨论 live benchmark 的方法论争议埋了伏笔：**只靠"每月加新题"这种线性扩展手段不足以阻止整体曲线右移**。

第三是 SWE-bench Verified 的"事实死亡"事件给整个学界提供了一个干净的 case study。2024-08 OpenAI 发布 500 题的 human-validated 子集；2025 H2 起 SWE-Bench+ (arXiv 2410.06992) 系统揭示 32.67% 的"成功"补丁靠的是 issue comment 中泄漏的 gold patch，31.08% 通过弱测试[^swebench-plus]；2026-Q1 OpenAI 停止在 model release card 中报告 Verified 分数。配套的 Scale AI SWE-Bench Pro 暴露出 16–35 个百分点的 contamination gap — Claude Opus 4.5 在 Verified 80.9% / 在 Pro 仅 45.9% [^morphllm-2026-pro-gap]。**Verified 死亡的意义不在于它"被破解"，而在于一手 benchmark 提供者（OpenAI）主动宣告该 benchmark 不再可用** — 这是 2024 之前都罕见的现象。

5.2 经典 contamination 文献：Sainz、Magar、Golchin 与 n-gram 假设

讨论 2026 新方法前必须先把经典 contamination 文献的方法论根基与盲区刻画清楚。三条主流路径自 2022 起逐步成型：

Magar 2022 (NLP 评估中的 contamination 度量)[^magar-2022] 首次把"训练时见过题"的影响形式化：定义 *exact match memorization* 与 *paraphrased memorization* 两种 contamination 类型，并指出仅看 exact match 会系统低估真实污染规模。这是后续 paraphrase / semantic contamination 讨论的起点。

Sainz et al. 2023 (NLP Evaluation in Trouble)[^sainz-2023-nlp] 通过手动 black-box 探测发现 ChatGPT 能逐字复述 BIG-Bench / GSM8K / MMLU 的部分题目，给出"大模型 + web-scale 数据 = 不可避免的 contamination"的第一份系统观察。这篇 EMNLP 2023 paper 在工业界引发立刻的 decontamination pipeline 跟进 — Llama 2、Mistral、DeepSeek 等都开始把"对每个 benchmark 做 13-gram filtering"作为标配。

Golchin & Surdeanu 2023 (Time Travel in LLMs)[^golchin-2023-timetravel] 设计 *guided instruction* 探针：让模型续写、复述、按 split / 段落 / 数据集结构提示，黑盒判断模型是

36 / 58

否"知道 benchmark 题目的存在"。这套方法影响了后续 Min-K% (Shi 2024)[^shi-2024-mink]、Min-K%++ 与 ReCaLL 等 membership inference 路线。

三条路径背后共享同一个**隐含假设**：contamination 是字符串级（exact / paraphrased）现象，13-gram filtering 加 membership inference 足以发现绝大多数泄漏。这个假设支撑了 2023–2025 间几乎所有 pretrain corpus 的 decontamination pipeline — lm-evaluation-harness 的 `decontaminator`、AI2 的 DCLM filter、Common Crawl decontamination scripts 等都是 13-gram 的不同 wrapper。**2026 的核心方法论突破，是 4 篇近距离发表的 paper 联合否定了这个假设**。

5.3 2026 春季的 contamination 方法学爆发：4 条正交路线

把 2026-02 至 2026-05 间 contamination 方法 paper 拼起来看，构成了一个清晰的四象限：是否需要 dataset owner 配合（active vs passive）× 是否在 model 部署后才能用（ex-ante vs ex-post）。每条新方法占住一个象限并对经典 n-gram 假设给出不同形式的反驳。

5.3.1 Soft Contamination（Spiesberger et al. 2026-02）：语义级盲区

Spiesberger 等用 embedding 相似度对 Olmo3 训练 corpus 做全量检测，发现 **CodeForces 78%、ZebraLogic 50% 的题目在训练数据中存在语义重复**，全部被 13-gram filter 漏掉 [^soft-contam-2026]。更关键的是干预实验：当 fine-tune 数据加入 benchmark 题目的语义改写版后，**严格 held-out (训练中不存在任何形式重复) 子集的准确率也会显著提升**。

这个发现的方法论含义有两层。表层意义是"现有 decontamination 工具链漏检"，需要 embedding-based filter 补救。深层意义则把 §1.2 提出的"benchmark 信息量"框架推到极致：**当一道题在 web 上有 78% 的语义重复率时，benchmark 报告的进步与"看相似题学到 transferable skill"是高度纠缠的**。论文的强 framing 后果是："当前公开 benchmark 上的进展报告需要打折扣，相当一部分来自 web-scale 训练数据的语义重复积累"。这一句话本身在 2026-Q1 评测社区引发的 Twitter 讨论强度远高于多数同期 paper [^plateau-2026]。

需要注意的争议点：作者只在 CodeForces 与 ZebraLogic 两个 benchmark 实测，外推到 MMLU / AGIEval / GSM8K 是否同样 78% 量级未验证；"语义重复带来 held-out 提升"是 (a) shallow recall 误归因为泛化、还是 (b) 看相似题真的学到 transferable skill，paper 的实验设计无法把两者完全分开。一个值得后续 paper 跟进的实验是用 Quantifying Test Set Contamination [^quantify-contam-2026] 的 scaling-law 框架对比"严格 contamination" 与 "语义近似 contamination" 在 loss 曲线上的差异 — 前者已被证明"单个 replica 即可让 loss 跌破不可约误差"，后者的量化效应大小尚是开放问题。

5.3.2 Cross-Context Verification（Song 2026-03）：黑盒行为路线

CCV 论文的开篇引言把现有方法批评得相当尖锐："string-level / probabilistic / paraphrase 路线 **never directly observe whether a model reasons or recalls** — 都是间接探针"

37 / 58

[^ccv-2026]。Song 的提案是直接观察行为：**在 N 个独立 session 中让模型解同一道题，contaminated 模型多次输出会高度一致（perfect recall），clean 模型会有显著 variation**。在 9 道 SWE-bench Verified × 5 trial 的 Claude Opus 4.6 实验中得到 perfect separation (Mann-Whitney U=0, p≈0.012, r=1.0)。

CCV 论文最 surprising 的二级发现是**33% 既有 prior contamination labels 是 false positive**。如果该数字在更广的 SWE 类长 context task 上成立，意味着 §5.2 介绍的 Min-K%、Min-K%++、ReCaLL 这类 membership inference 方法在 long-context 上**召回率偏激进**（误把 reasoning 出来的输出当 contamination）。CCV 同时报告了 multi-stage HCCA 扩展的"100% sycophantic confirmation"失败案例，强调"信息隔离比结构复杂度更关键"。这是 multi-agent contamination 检测的方法学 caution。

CCV 路线的局限是仅在 9 道 SWE-bench Verified 上验证，sample size 极小；推广到 MCQ / 短问答类 benchmark 时 "diversity" 信号的稳定性未知 — MCQ 题型本身只有 4 个可能输出，temperature 0 下 contaminated 与 clean 模型的 diversity 差距可能消失。

5.3.3 Dataset Watermarking（Huang, Chaudhuri, Wang 2026-05）：主动 ex-ante 防御

Huang 等的方案占住第四象限：**dataset owner 在发布前主动 embed 统计水印**。具体做法是通过 rephrasing 提升随机词对的共现频率，向数据中嵌入对全局 statistics 可检测但对单条 sample 不可见的信号 [^closed-watermark-2026]。论文最重要的理论贡献是**针对闭源 LLM 给出 provable detection guarantees** — 不依赖访问模型权重或概率，仅靠生成输出即可在 p<0.01 上检出 ~1% 数据混入。

这条路线对经典 contamination 假设的反驳方式是**完全绕过"检测"而走向"防御"**：与其事后 detect 谁 contaminated，不如让 dataset 自带可证明的 trace。这种 framing 跟 §6 讨论 LMSYS Arena 时"用人类偏好绕过 LLM-judge"是同一种方法学姿势 — 都是承认现有度量本身有根本缺陷，转而通过修改评测对象（dataset / 人 vs 自动 judge）的方式重新定义问题。

需要强调的局限是**实验主要在 fine-tuning 场景验证**（~1% dataset 混入）；真正 pretrain scale 上 watermark dataset 占 TB 级数据中的极小比例时 detection power 衰减如何，paper 未直接给出结果。这一点对 pretrain 评测者尤其关键 — 真正可怕的 contamination 多发生在 pretrain 阶段，而非 fine-tune。

5.3.4 LLM Olympiad（Cruz & Aji 2026-03）：协议层倡议

LLM Olympiad 不是检测方法而是**赛事化评测协议**。借用 IOI/IMO/ICPC 的传统提出三大机制：sealed problem set（题目仅在评测当天解封）、frozen submissions（模型在评测前提前冻结）、unified standardized harness（消除"各家用不同 prompt"）+ 赛后完整 release [^llm-olympiad-2026]。它是 contamination 防治的**社区组织层方案** — 不指望某种新算法解决问题，而是组织一次性事件来回避问题。

这个倡议的学术意义在于把 2022–2025 评测生态的两个极端拉到中间：完全公开 leaderboard（HELM / Open LLM Leaderboard，透明但易污染）vs 完全闭源 private

38 / 58

benchmark（SEAL、Anthropic 内部 eval，抗污染但黑箱）之间，LLM Olympiad 提出第三条道路。同期 DEP (Decentralized LLM Evaluation Protocol, arXiv 2603.01167) 走 ex-post 工程隔离路线、Judge Reliability Harness 走 meta-eval 路线 — 三者共同构成 2026-Q1 评测协议层的并发讨论。

需要承认的是 LLM Olympiad 是**position paper 而非实证 paper**：未提供 reference implementation、未做 sealed-exam 与 public benchmark 在 contamination 量化上的对比实验，社区接受度还有待 2026 H2 实际赛事运行后才能验证。

5.4 五大正交路线：active / passive / framing / protocol / judge meta-eval

把上述 4 个 2026 paper 与经典 n-gram / membership inference / canary 一起放进一个分类学，得到 contamination 应对的五大正交路线：

| 路线 | 代表方法 | 时点 | dataset 配合 | 主要批评 | |---|---|---|---|---| | **Active / ex-ante** | Carlini canary 2019、Closed-LLM Watermark 2026 | 训练前 | 需要 | 仅 dataset owner 可用 | | **Passive 字符串级** | 13-gram filter、ReCaLL、Min-K%++ | 训练前 / 后 | 不需要 | 漏检 paraphrase / 语义重复 | | **Passive 行为级** | CCV、HCCA | 训练后 | 不需要 | N 倍推理成本、sample size 偏小 | | **Framing / 度量** | Soft Contamination、Quantify Test Set Contamination | 训练后 | 不需要 | 不是 turnkey 工具，需要 embed corpus | | **Protocol / 组织层** | LLM Olympiad、DEP、Judge Reliability Harness | 赛事 / 评测时点 | 部分需要 | 成本高、社区采纳门槛大 |

这五条路线在 2026-Q2 仍互不替代，最佳实践是组合使用。一个具体的工程级建议（详见 Part I §contamination 工程章节）是：pretrain 阶段先用 13-gram + embedding-based filter 对训练 corpus 做两层 decontamination；评测阶段对 release model 用 CCV 做 spot-check；高 stake leaderboard 用 LLM Olympiad-style sealed-exam 提交 + Judge Reliability Harness audit。

5.5 学术争议核心："测训样集泄漏" vs "通用推理迁移"

把 §5.2–5.4 的方法学成果合到一起，学术界 2026 春季出现了一场尚未尘埃落定的核心辩论：**当 Soft Contamination 报告 CodeForces 78% 语义重复并实证其抬升严格 held-out 准确率时，benchmark 上的 model 进步究竟应该归因于 (A) 训练数据中累积的"近似题泄漏"，还是 (B) 模型真的从相似训练样本学到 transferable reasoning skill？**

这场辩论本质上是统计学习理论的老问题（generalization vs memorization）在 contamination 语境下的重生。各方立场可以归纳为三条：

强污染论者（Spiesberger 阵营）：认为既然语义重复率高达 78% 且严格 held-out 也受益，那么 benchmark 报告的进步绝大部分是 shallow generalization on semantic duplicates。推论是公开 web benchmark **整体上不能信任**，必须用 sealed / dynamic / 新生成 benchmark 替代。LiveBench、MathArena ArxivMath、SWE-Bench Pro Private 等就是这一阵营的实践产品。

39 / 58

真泛化论者（Quantify Test Set Contamination 部分作者倾向）：认为"看相似题学到 transferable skill"本身就是 ML 的合理目标。语义重复传染到 held-out 的本质机制可能是任务理解、变量符号识别、解题模板的内化 — 这些在 ML 文献里是"成功的泛化"而非污染。如果 LLM 学完 ZebraLogic 50% 的训练相似题后能解释 held-out ZebraLogic 题为何成立，这与"看 100 道带解析的数学题后能解新题"在性质上没有区别。

机制怀疑论者（CCV 阵营 + Anthropic 内部讨论）：质疑现有方法学**无法区分上述两种情况**。CCV 在 SWE 上 33% prior labels 为 false positive 说明 n-gram / Min-K% 类方法的判定本身不准；Soft Contamination 的 "held-out 也提升"实验中"如何确认 shallow recall vs deep transfer"的归因方法也未给出。在没有 mechanistic 证据之前，**任何"benchmark 进步多大比例来自 contamination"的具体数字都是 speculation**。

这场争议的实际后果之一，是 2026-Q2 顶会论文写作风格发生微妙转变。ICLR / NeurIPS 的 model paper 越来越倾向于报告**多个 contamination-limited benchmark 的组合分**（LiveBench + MixEval-Hard + SWE-Bench Pro）而非单一传统 benchmark；评测社区开始把**"模型在 release 后 6 个月 vs release 时刻的 benchmark 分数差距"**当作 contamination 度量 — 如果一个 benchmark 在 release 后 6 个月模型分数从 50% 涨到 80%，可以怀疑该 benchmark 已经"被吃透"。

5.6 缓解路径：MixEval-Hard / LiveBench / Arena-Hard 的设计哲学

§5.3 讨论了纯方法论层面的应对，但实际可用 benchmark 上 2024–2026 间也出现了三种不同的设计哲学。它们在 §6 的范式转移讨论会被详细展开，此处只对其作为"contamination 应对方案"的方法论位置做总结：

LiveBench (White, Dooley et al. ICLR 2025 Spotlight) 的核心承诺是 "contamination-limited via monthly source refresh" — 每月从最近 6 个月的 arXiv / IMDb / 新闻拉新材料生成 ~50 新题，全部用客观 verifier 评分（数值 EM / unit test / 规则解析）以绕过 LLM-judge bias [^livebench-paper]。但 §5.1 已经指出 LiveBench 也在 plateau；其设计哲学的局限是**"新材料"不等于"训练时未见过的语义"** — Spiesberger 的 Soft Contamination 框架直接威胁了这一点。LiveBench 目前的应对是把旧题在 6–12 个月窗口后退役，并保留 release-tagged 比较（如 livebench-2025-04-25 vs livebench-2024-11-25）允许时序对比。

MixEval-Hard (Ni et al. NeurIPS 2024) 走的是另一条路：**用 wild query 分布对既有 benchmark 题目做 weighted sampling** — 不造新题，而是把 MMLU / GSM8K / BBH / HellaSwag 等 14 个 source benchmark 里"与真实用户查询最相似"的题目混合得到 1K hard set。与 LMSYS Arena 相关性达 0.96 但成本仅 MMLU 的 6% [^mixeval-paper]。它的方法学优势是低成本接近 Arena ranking；局限是 source benchmark 本身的 contamination 会传染到 MixEval — 当 MMLU 6.49% 错题率传染到 MixEval-Hard 时，后者也继承了同样的问题。MixEval 自 2024-08 后未释出新版本，社区已经把它视为 "NeurIPS 2024 时期 Arena 替代品 baseline" 而非 frontier 评测 [^mixeval-repo]。

Arena-Hard-Auto (Li et al. 2024 + v2 2025-04) 是 LMSYS 离线 approximator，从 20 万真实用户 prompt 中 BERTopic 筛 500 hard prompts + v2 加 250 creative-writing prompts，用

40 / 58

GPT-4-Turbo / GPT-4.1 + Gemini-2.5 ensemble 做 pairwise judge [^arena-hard-paper]。它在 contamination 上的特殊位置是**评测对象本身是开放生成**（不是 MCQ 题目），理论上 contamination 必须包括 prompt + 期望 response 两端，门槛较高。但它的核心局限是 LLM-judge bias — GPT-4 judge 对 GPT 系模型有 self-preference (arXiv 2410.21819)、verbosity bias (arXiv 2310.10076)、position bias 三大已知问题，style control + ensemble 部分缓解但未根除。

三种设计哲学的对比给后续章节定下基调：**没有一个 benchmark 同时解决 contamination + judge reliability + saturation 三个问题**。LiveBench 强 contamination 防御但已开始 plateau；MixEval-Hard 强 Arena correlation 但继承 source contamination；Arena-Hard 强 user-aligned 但 LLM-judge 偏置。Part III 的决策章会主张同时报告三套指标作为 cross-validation。

5.7 小结与对后续章节的接口

本章把 2023–2026 的 contamination 学术辩论组织成"经典 n-gram 假设 → 2026 四方法学反驳 → 五正交路线分类学 → 强污染 vs 真泛化的核心争议 → 三种缓解 benchmark 设计哲学"五段叙事。三条结论值得在后续章节继续追：

1. **饱和不可单独缓解**。LiveBench 的 plateau 现象证明"每月新题"线性扩展不够；需要与 §6 的 agent benchmark / live tool eval 一起组成多维度上限。
2. **contamination 是方法学问题而非 benchmark 问题**。在 mechanistic 区分 shallow recall vs deep transfer 之前，所有 contamination 量化报告都是 lower bound — §7 future-debate 章会回到这个 epistemic 问题。
3. **评估协议层与方法层同样重要**。LLM Olympiad 类 sealed-exam 协议、CCV 类行为探针、Closed-LLM Watermark 类主动防御代表的是 contamination 问题的不同 framing — 选哪个不是技术问题而是组织能力 + 部署场景的问题。

§6 接下来讨论 live / dynamic / agent benchmark 的范式转移，将以本章的 contamination 五路线分类作为评估 agent benchmark 抗污染能力的方法论标尺。

引用

[^mmlu-redux]: Gema, A.P. et al. (2024). *Are We Done with MMLU?* arXiv:2406.04127.

[^evalplus]: Liu, J. et al. (2023). *Is Your Code Generated by ChatGPT Really Correct?* arXiv:2305.01210.

[^livebench-llm-stats]: llm-stats.com. (2026-05-25 snapshot). *LiveBench leaderboard*. https://llm-stats.com/benchmarks/livebench

[^plateau-2026]: *When AI Benchmarks Plateau*. (2026). arXiv:2602.16763.

[^swebench-plus]: Aleithan, R. et al. (2024). *SWE-Bench+: Enhanced Coding Benchmark for LLMs*. arXiv:2410.06992.

41 / 58

[^morphllm-2026-pro-gap]: MorphLLM. (2026). *SWE-Bench Pro vs Verified Contamination Gap*. https://www.morphllm.com/swe-bench-pro

[^soft-contam-2026]: Spiesberger, A., Vazquez, J.J., Pochinkov, N. et al. (2026-02-12). *Soft Contamination Means Benchmarks Test Shallow Generalization*. arXiv:2602.12413.

[^quantify-contam-2026]: *Quantifying Test Set Contamination on Generative Evaluations*. (2026-01). arXiv:2601.04301.

[^ccv-2026]: Song, T.-E. (2026-03). *Cross-Context Verification: Hierarchical Detection of Benchmark Contamination through Session-Isolated Analysis*. arXiv:2603.21454.

[^closed-watermark-2026]: Huang, P., Chaudhuri, K., Wang, Y.-X. (2026-05-07). *Dataset Watermarking for Closed LLMs with Provable Detection*. arXiv:2605.06865.

[^llm-olympiad-2026]: Cruz, J.C.B. & Aji, A.F. (2026-03-24). *LLM Olympiad: Why Model Evaluation Needs a Sealed Exam*. arXiv:2603.23292.

[^livebench-paper]: White, C., Dooley, S., Roberts, M. et al. (2025). *LiveBench: A Challenging, Contamination-Limited LLM Benchmark*. ICLR 2025 Spotlight. arXiv:2406.19314.

[^mixeval-paper]: Ni, J., Xue, F., Yue, X. et al. (2024). *MixEval: Deriving Wisdom of the Crowd from LLM Benchmark Mixtures*. NeurIPS 2024. arXiv:2406.06565.

[^mixeval-repo]: Ni, J. (2024–2026). *MixEval GitHub repository*. https://github.com/JinjieNi/MixEval

[^arena-hard-paper]: Li, T., Chiang, W.-L., Frick, E. et al. (2024). *From Crowdsourced Data to High-Quality Benchmarks: Arena-Hard and BenchBuilder Pipeline*. arXiv:2406.11939.

[^magar-2022]: Magar, I., Schwartz, R. (2022). *Data Contamination: From Memorization to Exploitation*. ACL 2022. arXiv:2203.08242. https://arxiv.org/abs/2203.08242

[^sainz-2023-nlp]: Sainz, O., Campos, J.A., García-Ferrero, I., Etxaniz, J., de Lacalle, O.L., Agirre, E. (2023). *NLP Evaluation in Trouble: On the Need to Measure LLM Data Contamination for each Benchmark*. EMNLP 2023 Findings. arXiv:2310.18018. https://arxiv.org/abs/2310.18018

[^golchin-2023-timetravel]: Golchin, S., Surdeanu, M. (2023). *Time Travel in LLMs: Tracing Data Contamination in Large Language Models*. ICLR 2024. arXiv:2308.08493. https://arxiv.org/abs/2308.08493

[^shi-2024-mink]: Shi, W., Ajith, A., Xia, M., Huang, Y., Liu, D., Huang, T., Zhang, S., Lee, K., Henderson, P., Chen, D., Wettig, A., Hashimoto, T. (2024). *Detecting Pretraining Data from Large Language Models* (Min-K% Prob). ICLR 2024. arXiv:2310.16789. https://arxiv.org/abs/2310.16789

42 / 58

6.1 从 dataset 到 rollout：评测对象的形态变迁

§5 把 contamination 与饱和的方法学反应组织清楚了，但还有一条更基础的 axis 没碰：**评测对象本身的形态**。2018–2023 的 NLP 评测是"模型对 dataset"，2024 后逐渐变成"agent 对环境"。这一形态变化对评测方法论的冲击不亚于 contamination 危机，且在 2026 年与 §5 的争议产生了几条耐人寻味的纠缠。

最简洁的对比图景：传统 dataset benchmark 的最小单元是 `(input, expected_output, scorer)`；agent benchmark 的最小单元是 `(initial_state, task_description, success_predicate, environment_step_function)`。**前者评测的是函数映射的准确度，后者评测的是序列决策在动态环境里的轨迹质量**。两者背后的统计学含义、可比性假设、可复现性约束完全不同。

具体到 2024–2026 的 benchmark 景观，这种形态变迁有四条同时演进的支线：

第一条是**LMSYS Chatbot Arena / Arena-Hard 路线**——把"评测对象"从 dataset 替换为"真实用户群"，用人类偏好（pairwise vote）作为基础度量。这条路线在 §5.6 已经从 contamination 角度讨论过，本章关注其方法论争议。

第二条是 **LiveBench / MixEval-Hard / Arena-Hard rolling dataset**——保留 dataset 形态但通过周期 refresh + dynamic sampling 缓解 contamination。形态没变，但 evaluation 的 unit 从"一次性数字"变成"时序数字"。

第三条是 **GAIA / OSWorld / WebArena+VWA / AgentBench / TAU-bench / AgencyBench / MCPMark / MCP-Atlas / Odysseys / AstaBench 等 agent rollout benchmark**——评测对象从 dataset 完全替换为 environment。这是 §6.3-6.5 的主轴。

第四条是 **NIAH / RULER / LongBench v2 / MMMU 这类长上下文 + 多模态 benchmark**——介于经典 dataset 与 agent 之间，保留静态 dataset 形态但任务输入 / 输出复杂度大幅提升。它们经历了与 agent benchmark 类似的"benchmark-as-environment"演化（如 NIAH 在 grid 上扫描），是研究范式转移的中间形态。

6.2 LMSYS Arena 方法论争议：Boyeau 2024 与 Bradley-Terry 假设

讨论 agent 评测之前先解决"人类偏好评测"的方法论争议。LMSYS Chatbot Arena 自 2023 以来积累超过 200 万 pairwise vote，是 LLM 选型的事实金标准；Arena-Hard-Auto 是其离线 approximator（500 题 v1 / 750 题 v2，GPT-4 ensemble judge）。但这条路线在 2024-2025 之间面临数次系统性方法学批评，最具代表性的是 Boyeau 2024 [^boyeau-2024-arena]。

Boyeau 等指出 LMSYS Arena 用 Bradley-Terry (BT) 模型把所有 pairwise vote 聚合成单一 Elo / win-rate ranking，但 BT 假设有几条关键前提：(1) 每对模型的胜率 logit 是稳定的标量差；(2) prompt 是 i.i.d. 抽样的；(3) judge 对 prompt-model 交互的偏好可分解为 model-specific + prompt-specific 两个独立项。这三条假设在 Arena 数据上**没有一条严格成立**：(1)

43 / 58

模型 A 在 coding prompt 上强 / 在 creative writing 上弱意味着相对胜率与 prompt 类别强耦合；(2) 用户提交的 prompt 显然有 selection bias（更复杂的 prompt 更可能被提交）；(3) judge（用户）对长答案、markdown 美观度有系统性偏好。

Arena-Hard v2.0 (2025-04) 引入 **style control** 把 token length + markdown 元素作为协变量回归，部分缓解 (3)；ensemble GPT-4.1 + Gemini-2.5 judge 缓解 single-judge bias；但 (1) (2) 在方法论层未被解决。这导致 LMSYS Arena ranking 在不同**任务子集**上对同一模型的相对位次可能差 5–10 名 — 一个具体 case 是 Claude Opus 系在 creative writing 上稳压 GPT 系，但在 coding 上反之，aggregate ranking 把这种 task-conditioned 差异平均掉了 [^arena-hard-paper]。

对调研者的方法论启示是：**Arena 与 Arena-Hard 的 single-number ranking 只在"跨任务平均代表通用能力"的强假设下成立**。把它当 frontier model selection 主信号是粗糙的；细粒度调研需要看 per-category breakdown 或专门 benchmark。Boyeau 提出的替代方案是用 **mixed-effects model** 显式建模 prompt × model 交互项，这在 2025-2026 的 leaderboard 上仍未被采纳，但理论上更正确。

6.3 Agent benchmark 的评分哲学：execution / judge / pairwise / TrueSkill

agent benchmark 与 dataset benchmark 在评分哲学上的差异，可以从五代表性 benchmark 的协议对照看清。

Execution-based（WebArena / OSWorld / SWE-bench / τ-bench / MCPMark）：success predicate 是程序化的 — 文件被改成正确状态、SQL query 返回正确结果、test suite 全部通过。优势是客观可复现，劣势是只能评测有清晰"成功状态"的任务，且 evaluator 自身可能有 bug（OSWorld 初版 300+ 个 issue 全部是 evaluator script 问题，2025-07 OSWorld-Verified 才修复 [^osworld-verified]）。

LLM-as-judge（GAIA / AstaBench Literature / AgencyBench / Arena-Hard / Odysseys）：用 frontier LLM 评分 final answer 或 trajectory。优势是适用于开放生成任务，劣势是 judge bias + reliability 问题（§5.3.4 的 Judge Reliability Harness 已经实证 4 个 SOTA judge 在 4 个 benchmark 上"无一 uniformly reliable" [^judge-reliab-2026]）。

Pairwise + Bradley-Terry（Arena-Hard / LMSYS Arena）：见 §6.2。

TrueSkill / Elo（Cattle Trade / 部分 multi-agent benchmark）：Microsoft 多智能体评级体系，对 n 个 agent 在多 round 比赛中的相对实力做贝叶斯估计 — μ 表示能力均值、σ 表示不确定性。Cattle Trade 报告 Gemini 3 Flash Preview TrueSkill μ=30.1, σ 仍较宽 [^cattle-trade-2026]。优势是支持非传递性（A 赢 B 不必赢 C 也赢 B 的赢者），劣势是 n 局间的 inference 必须假设 within-game variance 已主导（Cattle Trade paper 自承"deck order 主导 seat position"，意味着排名可能源于运气）。

Process / rubric-based（Odysseys / LH-Bench / 部分 AstaBench）：把任务分解为 6–10 个 graded rubric，每个 rubric 独立判定。Odysseys 报告 rubric κ = 0.788 vs binary

44 / 58

trajectory judge κ = 0.508，rubric 与人类专家一致性远高 [^odysseys-2026]。但 rubric 设计本身需要 task 作者的领域知识，scale 困难。

Hybrid（多层判定）：AgencyBench 是典型 — 规则评分 + Claude-4-Sonnet 文本 LLM-judge + Gemini-2.5-Pro 视觉 LLM-judge 三层；50 task held-out 上和 4 位人类专家 κ = 0.93 [^agencybench-2026]。这是 2026 frontier benchmark 的方法论共识 — **单一 judge 不可信，需要 multi-judge ensemble + 程序化 sanity check**。

6.4 Long-horizon agent: GAIA / OSWorld / WebArena 的危机与替代

agent benchmark 的方法论争议在 2026-Q1 集中爆发，触发器是 Berkeley RDI 在 2026-04 发布的 *How We Broke Top AI Agent Benchmarks* 报告 [^berkeley-exploit-2026]。一个名为 BenchJack 的 exploit agent 在不真正解任何题的前提下：

- 在 GAIA 上达 ~98%（利用 HF 上公开的 validation answer key + normalization collision）；
- 在 WebArena 上达 ~100%（answer key 暴露 + 不安全 code execution + 弱评分器）；
- 在 OSWorld 上达 73%（agent-evaluator 隔离不足、code sandbox 不严）。

这份报告联同 ServiceNow 团队的 WebArena Verified 审计（506 任务受 permissive matching 影响，如 "2 000" 被当 "2" 接受，92 任务 page-content 字段混淆 [^webarena-verified-2026]）、以及 SWE-bench Verified 在 OpenAI audit 下被发现"每个 frontier model 都能逐字复现 gold patch"的事实，构成 2026-Q1 的 agent benchmark crisis。学界一致的诊断是 **agent benchmark 本身的可信度问题不来自 contamination 而来自 evaluator script 漏洞**（execution-based 评分器自己有 bug，allowing 不真正解题的轨迹 pass）。

应对这场 crisis 的几条路线在 2026-Q1-Q2 集中涌现：

GAIA → Gaia2 (Meta FAIR, ICLR 2026 Oral): 引入异步 / 动态环境 + write-action verifier + 多 agent 协作 [^gaia2-2026]，GPT-5 (high) overall pass@1 仅 42%，把 GAIA 原版 ~75% 的天花板拉回到 frontier-relevant 区间。Princeton HAL leaderboard 继续维护 scaffolded GAIA 评测 (HAL + Sonnet 4.5 74.6%) 作为 historical reference。

WebArena → WebArena-Verified (ServiceNow, NeurIPS 2025): 修复 506 permissive matching、标记 unachievable 任务、提供 Hard-258 子集。**VideoWebArena (ICLR 2026)** 扩展到长上下文视频任务，强调 skill retention + factual retention，是 vision-language agent 的次世代评测。

OSWorld → OSWorld-Verified (Jul 2025) + OSWorld-MCP (ICLR 2026): Verified 修 300+ issue；MCP 版本引入 158 验证过的 MCP 工具，证明 MCP tool 显著提升 success（o3 从 8.3% → 20.4%@15-step）[^osworld-mcp-2026]。

SWE-bench Verified → SWE-Bench Pro (Scale AI 2026): 1865 多语言多文件任务，含 731 public copyleft + 276 private proprietary + 858 held-out 三层防污染。Verified vs Pro 的 16–35 个百分点 gap 本身被作为 contamination 直接证据 [^swe-bench-pro-card]。

45 / 58

AgentBench → AgentBench-FC: 2024 后转为 function-calling form + 与 AgentRL 集成；但社区使用度让位给 GAIA / OSWorld / SWE-bench 等专精 benchmark — 这是"广覆盖 + 浅深度"在 2026 失宠的典型案例 [^agentbench-2023]。

6.5 长 horizon agent: Odysseys / AgencyBench / AstaBench 的学术意义

agent benchmark 的更大转向是从"短任务做对"走向"长 horizon 真实流程"。三个 2026 代表性 benchmark 展示了这一转向的不同侧面：

AgencyBench (SII-GAIR, ACL 2026 Main) 把任务难度推到 **1M-token / 90 tool call / 数小时** 的工业生产规模 [^agencybench-2026]。其方法论贡献是把"长 horizon agent"在(token, turns, diversity, user-sim, sandbox) 五个 axis 同时拉满 — GAIA2 仅 22.5 turns、Toolathlon 26.8 turns、UltraHorizon 60 turns，AgencyBench 平均 90 turns。最强模型 GPT-5.2 仅 56.5%。学术意义有三层：(1) 验证了"agent build agent's work" 在 1M-token 规模下可行但不可靠；(2) 暴露了 scaffold sensitivity 严重 — Claude Opus 4.5 在 native Claude-Agent-SDK 上比第三方 SDK 高 20.5%，意味着 model 排名因 scaffold 选择洗牌；(3) 提供"S_avg / E_att（每尝试效率）/ E_tok（每 token 效率）"三件套 — 把"分数"与"成本 / 效率"解耦。

AstaBench (AI2, ICLR 2026 Oral) 是首个公认的科研 agent eval [^astabench-2026]。2400+ 题分 4 大类 11 子 benchmark — Literature Understanding（PaperFindingBench / ScholarQABench2 / LitQA2-FT / ArxivDIGESTables）/ Code Execution（CORE-Bench-Hard / DS-1000 / SUPER-Expert）/ Data Analysis（DiscoveryBench）/ End-to-End Discovery（E2E-Bench）。最反直觉的发现是 **end-to-end discovery 成功率仅 ~3%**（即便用最强 Claude Opus 4.7）而其他子任务可到 50%+，意味着 aggregate score (Opus 4.7 = 58.0%) 包含严重的"高位虚胖"风险 — 真科研最难的"提假设 + 端到端验证"层模型几乎做不了。AstaBench 同时提供 cost-adjusted leaderboard（Sonnet 4.6 性价比胜出，Opus 4.7 上限胜出），这是评测从"分数 ranking"走向"分数 + cost frontier"的代表案例。

Odysseys (CMU, 2026-04) 把 200 长程 multi-site web 任务搭在 live Internet 上，最强 Opus 4.6 仅 44.5% perfect success 且 Trajectory Efficiency 仅 1.06% [^odysseys-2026]。最关键的方法学贡献是**引入 efficiency metric** — rubric score / step。这暴露了 frontier model "靠堆步数往结果上靠"的低效模式：GPT-5.4 用 base64 编码绕过 spreadsheet 输入限制、Opus 用 Wayback Machine 抓缓存页 — 这些是聪明但低效的"环境重构"行为。Odysseys 的 rubric κ = 0.788 vs binary trajectory judge κ = 0.508 比较，方法学上确认 **rubric-based 评估比 binary judge 与人类专家一致性高 50%+**，是 2026 agent benchmark 评分协议的强参考。

6.6 MCP 工具调用范式：MCPMark vs MCP-Atlas 设计哲学

MCP (Model Context Protocol) 在 2025 年成为 LLM 接外部系统的事实标准（Anthropic 主推、被 OpenAI / Google 兼容），随之诞生了**深度 vs 广度**两条 benchmark 设计路线：

46 / 58

MCPMark (ICLR 2026) 走深度路线：127 任务跨 5 个真实 MCP server (Notion / GitHub / Filesystem / PostgreSQL / Playwright)，每任务平均 16.2 turns 和 17.4 tool calls，远超 MCP-Universe (6.8 turns) 和 LiveMCPBench (3.2 turns) [^mcpmark-2026]。**全程 programmatic verify.py**，不用 LLM-judge — 这点在 2026 agent benchmark crisis 后被认为是更可信的设计选择。SOTA GPT-5.2-high 57.5%。

MCP-Atlas (Scale AI, 2026) 走广度路线：1000 任务跨 36 个 production MCP server (Notion / Exa / Airtable 等收费 API) / 220 个工具。500 hidden split 防污染。**claims-based judge** 给 partial credit (0 / 0.5 / 1) 而非 0/1 [^mcp-atlas-2026]。SOTA Gemini 3.5 Flash high 83.6%。失败归因分类显示 **63.3% 失败归因为"推理"而非"工具调用"本身** — 工具调用本身的失败已被 Claude / GPT 系基本驯服，未驯服的是"在 10-25 候选工具里发现正确工具 + 组合 3-7 个 tool call"的推理。

两者在评测哲学上**正面冲突**：MCPMark 用 programmatic verify 求 0/1，认为 partial credit 模糊评测严谨度；MCP-Atlas 用 claim-based judge 求 partial credit，认为 0/1 丢失诊断粒度。2026-Q2 社区尚未达成共识，但工程实践上**两者互补使用**（MCPMark 测深度 CRUD，MCP-Atlas 测大规模 tool discovery + composition）是工业部署前的事实标配。

6.7 Multi-agent benchmark 新方向：Cattle Trade / Agent² RL-Bench

multi-agent benchmark 在 2026 出现两个值得关注的新方向。

Cattle Trade (ICLR 2026 workshop) 把多机制经济博弈（拍卖 / 隐藏出价 / 议价 / 虚张声势 / 对手建模）集成在单一长游戏内，242 局跨 7 frontier LLM + 3 heuristic code agent [^cattle-trade-2026]。关键发现：**硬编码 TrackerAgent 排名第 2，仅次于 Gemini 3 Flash Preview** — 即当前 LLM 在"信息追踪 + 策略适应"上仍逊于完美信息硬编码 baseline。失败模式 (overbidding / self-bidding / bankrupt trade initiation) 揭示 frontier LLM 在长程博弈中的策略不一致。

Agent² RL-Bench (Microsoft, 2026-04) 走 meta 方向：**让 LLM agent 自己设计、实现并执行 agentic RL post-training pipeline** [^agent2-rl-bench-2026]。6 个 diagnostic task × 3 难度 (rule-based / judge-based / closed-loop online RL)，每 run 12h 工程预算。最强 Claude Code + Opus 4.6 在 ALFWorld 把 base 4.85 → 97.76（极强），但 DeepSearchQA 上限仍 <24%。这是"agent build agent"评测的早期标杆，与 PaperBench / MLE-bench (AI 研究复现) 形成 meta 层评测谱。

两个 benchmark 共同代表 2026 multi-agent 评测的 framing 转变：不再问"LLM 能否完成 X"，而是问"LLM 能否在 strategic coherence / meta engineering 这类高阶能力上演示一致性"。这种 framing 与 §5 讨论的 contamination 争议相关 — strategic coherence 不可能从 web data shallow memorize，必须靠真泛化；如果未来 LLM 在 Cattle Trade 上稳定打败 TrackerAgent，将是"真泛化"假说的强证据。

47 / 58

6.8 Safety vs capability 评测分离：OpenAgentSafety 的方法论位置

agent benchmark 还有一个 2026 关键方向是 **safety 评测从 capability 评测中独立出来**。OpenAgentSafety (CMU + AI2, ICLR 2026) 是这一方向的代表性框架 [^openagentsafety-2026]：350+ executable task 跨 8 个 critical risk 类别（web / code / file system / messaging / financial / privacy / deception / social harm），跨良性 + 对抗双用户意图。

最关键的实验数字：**Claude-Sonnet-3.7 在 safety-vulnerable 任务上仍 51.2% 产出 unsafe behavior，o3-mini 高达 72.7%**。这意味着即便最强 alignment-tuned frontier model 在多轮工具使用场景下的 safety alignment 严重不足。这与 §7 将讨论的 WMDP（危险知识 MCQ）形成方法学对比：WMDP 测"知识层面的危险能力"，OpenAgentSafety 测"行为层面的危险倾向"。**安全评估必须双轨进行** — 不能用 WMDP 高拒答率代表 agent 部署后真实安全。

OpenAgentSafety 的方法学贡献是把 agent safety 评测从"单步 prompt injection 拒答"扩展到"多轮、多用户、多工具的真实可执行环境"。这是 §6.4 提到的 agent benchmark crisis 中**少数没有被 BenchJack exploit 攻陷**的 framework — 因为它评测的是 unsafe behavior emission 率，攻击者制造 false negative 无意义。

6.9 长上下文 + 多模态：NIAH / RULER / LongBench v2 / MMMU 的研究对话

长上下文与多模态 benchmark 在 §3-4 已部分覆盖，本节只聚焦其与 agent paradigm 的方法论对话。

NIAH 是单 needle retrieval，2023-2024 几乎所有"长 context"模型公关用图；但 2025 起被 NoLiMa [^nolima-2025] 论证"只测字面 retrieval、不测语义"、被 Sequential-NIAH 论证"单 needle 不够"。**RULER (NVIDIA COLM 2024)** 用 13 task 扩展（NIAH × 8 + variable tracking + aggregation + QA × 2），实证"声称 32K+ 的模型一半在 32K 跌破 85%"。**RULERv2 (2025)** 从 retrieval 升级到 multi-step reasoning。**LongBench v2 (THUDM, ACL 2025)** 是 503 道现实长 context MCQ（context 8K–2M 词），最反直觉发现是 o1-preview reasoning model 57.7% 超过人类专家 53.7%，确认 "long reasoning chain 是 long context 必要能力"[^longbench-v2-paper]。

但 LongBench Pro (2026-01) 论证 v2 仍有大量题目"截断 context 也能答对"，挑战其"真长 context"声明 [^longbench-pro-2026]。这一争议的方法论含义与 §5 的 contamination 争议相似：**benchmark 名义上测什么 ≠ 实际测什么**。NIAH 名义测长上下文实际测字面 retrieval；LongBench v2 名义测长 context reasoning 实际测的 reasoning 与 context length 弱相关 — 多 hop 推理本身已是 reasoning 模型强项，与 context 长不长无必然关系。

MMMU (CVPR 2024 Oral) 是 11550 大学级多模态学科题，2024 GPT-4V 仅 56% vs 人类 88%。2026-02 公开 test set labels 后 contamination 几乎不可避免，Qwen3.6 Plus 已达 86%。**MMMU-Pro (ACL 2025)** 解决"大量题 text-only 模型也能答对"的 shortcut 问题 —

48 / 58

vision-only embed prompt 把分数从 56% 拉低到 16.8-26.9% [^mmmu-pro-2025]。这是多模态评测方法学上一个干净的 case：当 source benchmark 包含 text shortcut 时，vision modality 实际未在被评估；vision-only embedding prompt 是有效的 ablation。

6.10 小结与对 §7 的接口

本章把 2024–2026 的 live / dynamic / agent benchmark 范式转移组织为：dataset → user vote → rolling dataset → agent rollout 的形态变迁；execution / judge / pairwise / TrueSkill / rubric / hybrid 六种评分哲学并存；agent benchmark crisis (Berkeley RDI exploit + WebArena Verified audit + SWE-Verified deprecation) 触发的 GAIA→Gaia2 / WebArena→Verified / OSWorld→MCP / SWE→Pro 四代升级；长 horizon agent (AgencyBench / AstaBench / Odysseys)、MCP 工具调用 (MCPMark vs MCP-Atlas)、multi-agent (Cattle Trade / Agent² RL-Bench)、safety 分离 (OpenAgentSafety)、长上下文与多模态 (NIAH→RULER→LongBench v2 / MMMU→MMMU-Pro) 五条平行支线。

三个对 §7 的接口问题：(1) Arena 与 agent benchmark 的人类参与度还能否进一步提升，或必然让位于 LLM-judge ensemble？(2) agent benchmark crisis 暴露的 evaluator script 脆弱性是工程问题还是评估理论问题？(3) safety 评测分离后，capability + safety 是否可以保持独立 leaderboard，还是必须 joint optimization？这三个问题构成 §7 future-debate 的核心议题。

引用

[^boyeau-2024-arena]: Boyeau, P. et al. (2024). *Statistical Considerations for Benchmark-Based Evaluation of LLMs in Open-Ended Settings*. （LMSYS Arena Bradley-Terry critique 综述系列）

[^arena-hard-paper]: Li, T., Chiang, W.-L., Frick, E. et al. (2024). *From Crowdsourced Data to High-Quality Benchmarks: Arena-Hard and BenchBuilder Pipeline*. arXiv:2406.11939.

[^osworld-verified]: XLANG Lab. (2025-07-28). *Introducing OSWorld-Verified*. https://xlang.ai/blog/osworld-verified

[^judge-reliab-2026]: Dev, S., Sloan, A., Kavner, J., Kong, N., Sandler, M. (2026-03-05). *Judge Reliability Harness: Stress Testing the Reliability of LLM Judges*. arXiv:2603.05399.

[^cattle-trade-2026]: Müller, R., Müller, C. (2026). *Cattle Trade: A Multi-Agent Benchmark for LLM Bluffing, Bidding, and Bargaining*. arXiv:2605.14537. ICLR 2026 malgai workshop.

[^odysseys-2026]: Jang, L.K., Koh, J.Y., Fried, D., Salakhutdinov, R. (2026-04-27). *Odysseys: Benchmarking Web Agents on Realistic Long Horizon Tasks*. arXiv:2604.24964.

[^agencybench-2026]: Shi, J., Xiao, Y., Jiang, M. et al. (2026). *AgencyBench: Benchmarking the Frontiers of Autonomous Agents in 1M-Token Real-World Contexts*. arXiv:2601.11044. ACL 2026 Main.

49 / 58

[^berkeley-exploit-2026]: Wang, H., Mang, Q., Cheung, A., Sen, K., Song, D. (2026-04). *How We Broke Top AI Agent Benchmarks: And What Comes Next*. UC Berkeley RDI blog. https://rdi.berkeley.edu/blog/trustworthy-benchmarks-cont/

[^webarena-verified-2026]: ServiceNow Research. (2026). *WebArena Verified: Reliable Evaluation for Web Agents*. NeurIPS 2025. https://openreview.net/forum?id=94tlGxmqkN

[^gaia2-2026]: Froger, R. et al. (2026). *Gaia2: Benchmarking LLM Agents on Dynamic and Asynchronous Environments*. arXiv:2602.11964. ICLR 2026 Oral.

[^osworld-mcp-2026]: OSWorld-MCP team. (2026). *OSWorld-MCP: Benchmarking MCP Tool Invocation In Computer-Use Agents*. arXiv:2510.24563. ICLR 2026.

[^swe-bench-pro-card]: Scale AI. (2026). *SWE-Bench Pro: A Stronger Coding-Agent Benchmark*. https://scale.com/blog/swe-bench-pro

[^agentbench-2023]: Liu, X., Yu, H., Zhang, H. et al. (2023). *AgentBench: Evaluating LLMs as Agents*. arXiv:2308.03688. ICLR 2024.

[^astabench-2026]: Bragg, J., D'Arcy, M., Balepur, N. et al. (2026). *AstaBench: Rigorous Benchmarking of AI Agents with a Scientific Research Suite*. arXiv:2510.21652. ICLR 2026 Oral.

[^mcpmark-2026]: Wu, Z., Liu, X., Zhang, X. et al. (2026). *MCPMark: A Benchmark for Stress-Testing Realistic and Comprehensive MCP Use*. arXiv:2509.24002. ICLR 2026.

[^mcp-atlas-2026]: Bandi, C., Dumitru, R.-G., Hertzberg, B., Agarwal, D. et al. (2026). *MCP-Atlas: A Large-Scale Benchmark for Tool-Use Competency with Real MCP Servers*. arXiv:2602.00933.

[^agent2-rl-bench-2026]: Chen, W., Yang, X., Yang, X. et al. (2026). *Agent² RL-Bench: Can LLM Agents Engineer Agentic RL Post-Training?* arXiv:2604.10547.

[^openagentsafety-2026]: Vijayvargiya, S., Soni, A.B., Zhou, X., Wang, Z.Z., Dziri, N., Neubig, G., Sap, M. (2026). *OpenAgentSafety: A Comprehensive Framework for Evaluating Real-World AI Agent Safety*. arXiv:2507.06134. ICLR 2026.

[^nolima-2025]: Modarressi et al. (2025). *NoLiMa: Long-Context Evaluation Beyond Literal Matching*. arXiv:2502.05167.

[^longbench-v2-paper]: Bai, Y., Tu, S., Zhang, J. et al. (2025). *LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks*. ACL 2025. arXiv:2412.15204.

[^longbench-pro-2026]: *LongBench Pro: bilingual realistic extension*. (2026). arXiv:2601.02872.

[^mmmu-pro-2025]: Yue, X. et al. (2025). *MMMU-Pro: A More Robust Multi-discipline Multimodal Understanding Benchmark*. ACL 2025. arXiv:2409.02813.

50 / 58

7.1 Benchmark 是否还驱动 scaling？三层观察

LLM 评测的元问题 — 也是本章主轴 — 是 benchmark 与 scaling 的耦合关系是否在 2026 出现质变。2017–2023 的 NLP/LLM 进步轨迹里，benchmark 是 scaling laws 的"读数"——研究者通过 MMLU / GSM8K / HumanEval 等少数几个指标推断 model size / data size / compute 的边际收益。Hoffmann 的 Chinchilla scaling 用 dataset 数字 + downstream MMLU 数字反推 compute-optimal 配比；DeepSeek、Llama、Qwen 等开源系列的 release card 也以 benchmark 数字作 scaling 证据。

但 2024-2026 间，**benchmark 与 scaling 的耦合至少在三层意义上松动**：

第一层是单一 benchmark 的边际信息量趋零。§5 已经详述饱和现象；当 MMLU 在 GPT-4 → Claude Opus 4.7 这段 frontier 演进中只能从 88 → 91，2-3 个百分点的差距已经低于 prompt sensitivity 与 MMLU-Redux 6.49% 错题率的混淆噪声，单一指标无法回答"scaling 还有多少头部空间"。daVinci-LLM 论文明确说"evaluation protocol 选择 significantly 影响对 pretrain progress 的理解" — 200+ ablation 在不同 eval setup 下给出不同 ranking [^davinci-llm-2026]。

第二层是 reasoning 时代的 scaling 不再是 model size 单变量。o1 / o3 / DeepSeek-R1 / Claude Opus thinking mode 的兴起把"inference-time compute"作为新的 scaling axis，但传统 benchmark 报告的是 final answer accuracy 而非 thinking token / accuracy tradeoff curve。AstaBench 的 cost-adjusted leaderboard（Sonnet 4.6 性价比胜，Opus 4.7 上限胜）是首批回应这一断层的尝试；但更深层的方法学突破（如把 thinking-time scaling 纳入标准 benchmark 报告 schema）尚未发生。

第三层是 benchmark 数量爆炸导致的 selective reporting。Benchmark Health Index (BHI, arXiv 2602.11674) 给出量化诊断：106 个 validated benchmark 里大量冗余（MMLU / MMLU-Pro / MMLU-Redux / Chinese-MMLU 等测同一能力）；自报告文化使得每个 tech report 选择有利的 benchmark 子集 [^bhi-2026]。BHI 的三轴 (Capability Discrimination / Anti-Saturation / Impact) 给出的"该 benchmark 是否值得继续用"是 2026 评测社区第一个 meta-evaluation 量化工具。

但 BHI 自己也有方法学局限：scoring 依赖 91-model 2025 distribution，model landscape 一变（reasoning-tuned model 加入）健康度漂移；Impact 轴用 citation count 偏向 old benchmark；Anti-Saturation 假设线性 ceiling-rise，对突破性 reasoning 模型（o1, R1）导致 ceiling 跳跃的情况不 robust。这些自承的局限本身就是辩论：**meta-evaluation 工具是否能逃出"被它评测的 benchmark 的同样陷阱"？**这是 2026-Q2 学界的开放问题。

7.2 Human eval (Arena) vs 自动 eval：还有第三条道路吗？

§6.2 讨论了 LMSYS Arena 的 Bradley-Terry 方法学问题，但更深层的问题是：**人类偏好是否本身就是评测的金标？** 三种立场在 2026 共存：

51 / 58

Arena 主义 认为人类用户偏好就是 ground truth（除了 safety / 危险能力等少数维度）。论据是 LLM 的最终用户场景是人，偏离人类偏好的"准确"输出对部署无价值。LMSYS / Anthropic 在多次公开 talk 中倾向这一立场。

自动 eval 主义 认为人类偏好充满 bias（verbosity / formatting / cultural / educational 等）、低 reproducibility、 expensive、且 specific judge group 的偏好未必代表"应当被服务的用户群"。强调用 deterministic execution、code unit test、math verifier 等程序化评分。LiveBench、Olmo、DCLM、AstaBench 部分子任务等是该立场代表。

Hybrid / 第三道路 试图把两者结合。Judge Reliability Harness 论证 LLM-as-judge 在 4 个 SOTA judge 上"无一 uniformly reliable"[^judge-reliab-2026]，建议 perturbation-aware judge selection；AgencyBench 用三层 judge（程序化 + 文本 LLM + 视觉 LLM）做 cross-check；OpenAI / Anthropic 在 safety eval 上同时部署 red-team 人评 + 自动 attack benchmark。

这场辩论在 2026 尚未收敛，但 Boyeau 2024 提出的 **mixed-effects model** 路线给了一个理论方向——把 prompt × model × judge 三方交互显式建模，把"人类 vote 平均"与"自动 eval 平均"作为该 model 的两个 marginal 视角，而非把它们当作互相替代的 ground truth。这一路线目前仅在学术 paper 层讨论，工业 leaderboard 尚未采纳。

7.3 过程评估 vs 结果评估：reasoning 时代的方法学转变

reasoning model（o1 / R1 / Claude Opus thinking）的兴起把"过程评估"推到议程上。传统 benchmark 几乎只评测 final answer — 数学题对不对、代码 pass test、MCQ 选项对不对。但 reasoning model 的关键能力恰恰在 chain-of-thought 中：是否能识别 ill-posed problem 而不强行作答、是否能在推理中 self-correct、是否给出的 reasoning chain 在中间步骤上 faithful。

2026 出现了几个代表性"过程评估"框架：

Odysseys 的 rubric-based 评估（§6.5）— 把"成功"分解为 6.1 个 graded rubric，每个 rubric 独立评分，可以诊断 frontier model "靠堆步数往结果上靠" 的低效模式（Trajectory Efficiency 仅 1.06%）[^odysseys-2026]。

Soohak 的 refusal subset（research-level math benchmark，arXiv 2605.09063）—— 64 mathematicians curated 题目，含**显式 refusal subset 测试模型对 ill-posed 题的识别能力**。Gemini-3-Pro 30.4% / GPT-5 26.4% / Claude-Opus-4.5 10.4%，**没有 model 在 refusal 上 > 50%** [^soohak-2026]。这是过程评估的新维度 — 不是"做对了吗"，而是"该拒答时拒答了吗"。

MathNet 的多模态多语言 olympiad math (arXiv 2604.18584) — 30676 题跨 47 国 17 语言，提供 problem solving / math-aware retrieval / RAG 三个任务头，是过程评估在"中间检索 + 推理"上的扩展 [^mathnet-2026]。

SLUMP (When the Specification Emerges, arXiv 2603.17104) — 20 ML papers / 371 verifiable components / 60 iterative coding requests，测"faithfulness loss when specification is progressively revealed" — 这是过程评估在 multi-turn coding 上的应用。

52 / 58

过程评估的核心方法学贡献是把单一标量（pass/fail）扩展为多维评分（rubric × step × verifiability），可诊断 frontier model 的具体能力短板而非只看 aggregate score。但它的代价是**评分成本高、需要领域专家设计 rubric、对 scale 困难** — Odysseys 200 task 已是 rubric benchmark 的规模上限，更大规模需要自动化 rubric generation 的方法学突破。

7.4 AI-as-judge 可靠性：Judge Reliability Harness 的诊断

LLM-as-judge 是 2024-2026 评测生态的核心基础设施，但 §5.3 引用的 Judge Reliability Harness (RAND Corp, arXiv 2603.05399) 给出了它的方法学诊断：4 个 SOTA judge × 4 个 benchmark (safety / persuasion / misuse / agentic) × 4 类扰动 (formatting / paraphrasing / verbosity / label flipping)，结论是 **"No judge that we evaluated is uniformly reliable across benchmarks"** [^judge-reliab-2026]。

具体的失败模式包括：judge model A 在 safety 上稳但在 persuasion 上对 paraphrase 敏感；judge B 反之；label flipping (把已知正确答案改成已知错误答案) 在某些 benchmark × judge 组合下 judge 错误率高达 30%+。这意味着选 judge 必须 **per-task validate**，不能假设 SOTA = 全场通用。

更深层的方法学问题是 **LLM-as-judge 的循环依赖**：当用 GPT-4 evaluate Claude / 用 Claude evaluate GPT-4 时，judge 与 evaluated model 处在同一个能力分布上，难以判断 evaluation 是真客观还是 model A 与 judge B 在某种隐性偏置上耦合。Arena-Hard 论文报告 GPT-4 judge 对 GPT 系模型 self-bias 8-10 个百分点 (arXiv 2410.21819)、verbosity bias (arXiv 2310.10076) 5-8 个百分点等具体数字 [^arena-hard-paper]。

应对路径有三：(1) judge ensemble (Arena-Hard v2 用 GPT-4.1 + Gemini-2.5 双 judge)；(2) judge 与 evaluated model 来自不同 lab + 不同 architecture (Anthropic 内部 eval 主张)；(3) judge 自身 reliability stress test (Judge Reliability Harness 提倡)。这三条仍未形成 community consensus，但已经是 ICLR / NeurIPS 模型 paper 的 reviewer 必看清单。

7.5 危险能力 eval：WMDP 与可控部署

WMDP (Weapons of Mass Destruction Proxy, ML Safety, arXiv 2403.03218) 是 2024 NeurIPS D&B 提出的危险知识代理 — 3668 道 MCQ，跨 biosecurity / cybersecurity / chemical security 三个领域，目的是给 unlearning 研究和 deployment safety 提供量化指标 [^wmdp]。它的方法学位置在 §6.8 已与 OpenAgentSafety 对比 — WMDP 测"知识层"，OpenAgentSafety 测"行为层"。

WMDP 的核心方法学创新是把"危险能力"从难以测量的开放生成转为 well-defined MCQ — 模型在 WMDP 上的高分意味着拥有相关知识，低分意味着已经 unlearned 或本身未学。这给 model unlearning 研究提供了第一个 standardized benchmark。

但 WMDP 也面临方法学批评：(1) **MCQ 形式只测 recognition，未测 generation** — 模型可能"认得"危险知识但拒答时仍然拒答；这点与传统 capability MCQ 的批评相同（Holtzman 等指出 MCQ 与 generation 不等价）；(2) **proxy 设定的伦理 / 法律问题** — 真

正危险的细节不能直接进入 MCQ，所以题目用"proxy"形式表达，但这又削弱了与真实部署风险的相关性；(3) **测同一模型在 WMDP 与 OpenAgentSafety 上的表现可能不一致** — 一个在 WMDP 上拒答 90% 的模型，agent deployment 下可能 unsafe rate 仍 50%+，因为后者测多轮工具使用累积效应。

WMDP 的实际部署影响是 Anthropic、OpenAI、Google DeepMind 都在 release card 中报告 WMDP 分数；同时配合 dangerous capability evaluation (DCE) 的内部 red-team。这是"危险能力 eval"从学术 benchmark 走向公司 deployment 标准的 case。

7.6 安全评估的双轨：HarmBench / AIR-Bench / Sorry-Bench 与 alignment

安全评估在 2024-2026 间形成三层并行架构：

HarmBench (Center for AI Safety, ICML 2024) 走 attack-pattern 路线 — 510 个有害行为 × 18 种 attack 方法 × 33 个 target model，用 fine-tuned Llama-2-13B classifier 自动判定 Attack Success Rate (ASR) [^harmbench]。优势是 reproducible 红队 — 任何研究者都能跑同样的 attack 套件；缺点是 classifier 在 XS-Test benign 上 false-positive ~26.8%，会高估 ASR。

AIR-Bench 2024 (Stanford CRFM + Virtue AI, NeurIPS 2024 D&B) 走 regulation-anchored 路线 — 把 EU AIA / US EO / 中国生成式 AI 服务管理办法等 8 个监管文件 + 16 家公司政策分解为 4 层 taxonomy（314 个 lowest-tier risk category），生成 5694 prompts，GPT-4o judge 评 refusal compliance [^air-bench]。优势是覆盖完整监管面（previous benchmarks only 71% level-3 coverage）；缺点是 GPT-4o judge 单点信任 + over-refusal 倾向。

Sorry-Bench (ICLR 2025) — 440 class-balanced unsafe instructions × 44 fine-grained topics × 20 linguistic augmentations [^sorry-bench]。重点在 fine-grained refusal evaluation，把 HarmBench 的粗粒度 7 大类 risk 拆得更细。

三套 benchmark **互补使用** — HarmBench 测 attack robustness，AIR-Bench 测 content compliance，Sorry-Bench 测 fine-grained refusal — 是 2026 prod model 选型的 safety 标配。MASK Benchmark (arXiv 2503.03750) 与 SocialHarmBench (arXiv 2510.04891) 进一步扩展到 honesty under pressure 与 political/social harm 两个新维度。

但这三套 benchmark 与 alignment 评估的关系存在方法学张力。Alignment 评估关注模型是否帮助人类、是否诚实、是否避免操控；safety benchmark 关注模型是否拒绝危险请求。一个模型可以在 HarmBench / AIR-Bench / Sorry-Bench 上得分 95+ 但在 MASK 上 honesty 不到 50% — 高拒答率 ≠ 真诚信。这与 §5 的 contamination 争议同样指向 epistemic 问题：**benchmark 名义上测什么 ≠ 实际部署上是否真有 safety**。Safetywashing critique (arXiv 2502.09387) 警告把 TruthfulQA / Sorry-Bench / HarmBench 高分等同于 safe model 是把准确率混淆为安全性。

54 / 58

7.7 Meta-evaluation: BHI / daVinci-LLM 的诊断

§7.1 已经引入 BHI；本节展开 BHI 与 daVinci-LLM 形成的 2026 meta-evaluation 双重诊断。

BHI (arXiv 2602.11674) 提出 benchmark 不评测 model 而评测 benchmark 本身的健康度 — Capability Discrimination / Anti-Saturation / Impact 三轴 [^bhi-2026]。BHI 的最大方法学贡献是把"benchmark inflation"问题量化，强 Discrimination + 高 Anti-Saturation 的 benchmark 应优先用，弱 Discrimination + 低 Anti-Saturation 的 benchmark 应退役。这给评测社区第一个 principled basis 做"benchmark for benchmark"决策。

daVinci-LLM (GAIR-NLP, arXiv 2603.27164) 从 pretrain science 角度做出补充 meta-evaluation [^davinci-llm-2026]：明确讨论 evaluation protocol 选择如何影响对 pretrain 进展的理解（200+ ablation 实证不同 eval setup 给出不同 ranking）。daVinci 提出 **Data Darwinism L0-L9 taxonomy** 把 data processing 分级 — 不同 level 数据训练出的模型在不同 benchmark 上呈现不同 saturation curve，意味着 benchmark 选择必须与 training stage 匹配（pretrain 阶段适合 OLMES Base 套件、post-train 适合 instruction-following benchmark）。

BHI 与 daVinci-LLM 共同指向 meta-evaluation 的两个方向：(1) **benchmark 选择需要量化 audit**（BHI 的 health score）；(2) **benchmark 选择需要 stage-specific 优化**（daVinci 的 stage-evaluation matching）。这两个方向尚未形成统一框架，是 2026 H2 评测方法学的活跃议题。

7.8 小结：开放问题与三条研究主线

本章把 2026 评测方法学的核心辩论组织为：scaling 与 benchmark 关系松动（§7.1）→ human eval vs 自动 eval 的第三道路（§7.2）→ 过程 vs 结果评估（§7.3）→ LLM-as-judge 可靠性（§7.4）→ 危险能力 / 安全评估双轨（§7.5-7.6）→ meta-evaluation 框架（§7.7）。七节共同指向三条 2026 H2 - 2027 的研究主线：

1. **Meta-evaluation 是否能形成稳定标准**？BHI / daVinci-LLM / Judge Reliability Harness 是首批工具，但 BHI 自己也有 model-distribution-dependence 问题。下一步是让 meta-eval 工具自己经得起 perturbation test。
2. **Process evaluation 是否能 scale**？Odysseys / Soohak / SLUMP / AstaBench rubric 评估在 100-1000 题规模可行，但 web-scale 评测 (~50K+ 题) 是否需要自动 rubric generation？
3. **Safety 与 capability 的 joint optimization 还是独立 leaderboard**？2026 学界倾向独立 leaderboard 避免 safety 被 capability 优化稀释；但部署上必须是 joint — 这两个层级的 reconciliation 是开放问题。

55 / 58

引用

[^davinci-llm-2026]: Qin, Y., Liu, Y., Mi, T. et al. (2026-03-28). *daVinci-LLM: Towards the Science of Pretraining*. arXiv:2603.27164.

[^bhi-2026]: Zhu, L., Hua, H., Miao, L., Zhao, B. (2026-02-12). *Benchmark Health Index: A Systematic Framework for Benchmarking the Benchmarks of LLMs*. arXiv:2602.11674.

[^judge-reliab-2026]: Dev, S., Sloan, A., Kavner, J., Kong, N., Sandler, M. (2026-03-05). *Judge Reliability Harness: Stress Testing the Reliability of LLM Judges*. arXiv:2603.05399.

[^arena-hard-paper]: Li, T., Chiang, W.-L., Frick, E. et al. (2024). *From Crowdsourced Data to High-Quality Benchmarks: Arena-Hard and BenchBuilder Pipeline*. arXiv:2406.11939.

[^odysseys-2026]: Jang, L.K., Koh, J.Y., Fried, D., Salakhutdinov, R. (2026-04-27). *Odysseys: Benchmarking Web Agents on Realistic Long Horizon Tasks*. arXiv:2604.24964.

[^soohak-2026]: *Soohak: 439-problem research-level math benchmark with refusal subset*. (2026). arXiv:2605.09063.

[^mathnet-2026]: *MathNet: 30,676 multimodal/multilingual olympiad-level problems*. (2026). arXiv:2604.18584.

[^wmdp]: Li, N. et al. (2024). *The WMDP Benchmark: Measuring and Reducing Malicious Use With Unlearning*. NeurIPS 2024 D&B. arXiv:2403.03218.

[^harmbench]: Mazeika, M., Phan, L., Yin, X. et al. (2024). *HarmBench: A Standardized Evaluation Framework for Automated Red Teaming and Robust Refusal*. ICML 2024. arXiv:2402.04249.

[^air-bench]: Zeng, Y., Yang, Y., Zhou, A. et al. (2024). *AIR-Bench 2024: A Safety Benchmark Based on Risk Categories from Regulations and Policies*. NeurIPS 2024 D&B / ICLR 2025. arXiv:2407.17436.

[^sorry-bench]: *SORRY-Bench: Systematic and Granular Evaluation of LLM Safety Refusals*. (2024). ICLR 2025. arXiv:2406.14598.

A.1 定位：Stage 0.5 scan 的学术意义

2026-02 至 2026-05 这四个月里，frontier scan 浮现的 73 项新工作既非"老 benchmark 的常规迭代"，也非"评测社区在固定 framing 下的产能堆积"。它们集中暴露了 §5 contamination 争议、§6 agent crisis、§7 meta-evaluation 之三条主轴在 frontier 时点上的最新形态。本附录从**学术辩论视角**而非工程目录视角审视这批新工作 — 哪些工作真正回应了已知的方法学开放问题、哪些只是把现象再发现一次、哪些值得继续追、哪些应保持怀疑。

56 / 58

A.2 Multi-agent benchmark 的"非传递性"挑战

Cattle Trade (arXiv 2605.14537) 与 Agent² RL-Bench (arXiv 2604.10547) 是 2026-Q2 multi-agent benchmark 的两个代表，但它们对 §6 的 agent benchmark 评分哲学辩论提出了**新框架挑战**。Cattle Trade 用 TrueSkill μ/σ 替代单标量分数，承认多 agent 长游戏的非传递性（A 赢 B 不蕴含 A 赢 C 亦赢 B 的赢者）。这与 §6.2 Boyeau 2024 对 Bradley-Terry 单标量 ranking 的批评异曲同工 — 当任务结构本身是 non-transitive 时，Arena-Hard 类 pairwise 聚合的合理性更加可疑。可争议处在于 Cattle Trade paper 自承"deck order 主导 seat position"，意味着 TrueSkill 排名内的真实 variance 可能源于运气而非能力 — 这给"TrueSkill 是否优于 BT"提供了一个 cautionary tale。

Agent² RL-Bench 走的是 meta 路线，问"LLM 能否设计、实现、执行 agentic RL pipeline"。这种 framing 对应 §5.5 的"真泛化"假说 — 如果 LLM 能稳定 engineer 出 working RL pipeline，就是 web-scale shallow recall 难以解释的能力。但其 6 task 的诊断规模偏小，结论之间的 confidence interval 大，跟踪价值高于直接采纳。

A.3 Live math platform：contamination 防御与 ceiling 问题

MathArena Platform (arXiv 2605.00674)、LemmaBench (arXiv 2602.24173)、Soohak (arXiv 2605.09063) 构成 2026 frontier 数学评测的三足。MathArena 用月度 ArxivMath + AIME/USAMO 2026 + Lean 形式证明实现"contamination-resistant by recency"；LemmaBench 直接挖 arXiv 论文 lemma 当 theorem-proving 任务，SOTA ~10–15%；Soohak 引入 **refusal subset** — 测模型对 ill-posed 题的识别 — 没有 model 在 refusal 上 > 50%。这条线对 §5.6 "三种缓解 benchmark 设计哲学" 的延伸是把 LiveBench 月更思路从工程化能力测推到 frontier 数学能力测，与 §7.3 过程评估辩论密切相关 — refusal subset 测的是模型是否能在不解时承认不解，这本质是过程评估的能力维度。需要怀疑的是 LemmaBench 的 lemma 抽取本身对**作者意图**的还原度有限，被抽出 statement 是否仍是原 paper 中"该被独立证明的 lemma"未在 paper 中给出验证标准。

A.4 MCP-Atlas vs MCPMark：OOD 维度的方法论辩论

§6.6 已经讨论过 MCPMark / MCP-Atlas 的深度 vs 广度对偶。从学术辩论视角再加一层 — MCP-Atlas 引入 partial credit (0/0.5/1) 与 claims-based judge，与 GAIA 的 binary exact-match 形成方法论冲突。**OOD 评测在 partial credit 框架下是否仍可比？** 如果模型 A 在 "做对一半" 上稳定、模型 B 在"做对全部"上稀疏成功，二者的 aggregate 排名取决于 partial credit weight 的选择 — 这是 §7.3 rubric-based 评估在工具调用领域的另一个版本。MCP-Atlas 自承 63.3% 失败归因为推理 — 这点把工具调用 benchmark 与 reasoning benchmark 的边界进一步模糊，与 §5 的 contamination 边界辩论形成镜像。

57 / 58

A.5 Contamination 五大正交方法 vs n-gram 假设

§5.3 已经把 CCV、Dataset Watermarking、Soft Contamination、LLM Olympiad、Judge Reliability Harness 五条新路线展开。Stage 0.5 scan 进一步浮现 JECS Joint Decontamination (arXiv 2605.21543) 与 Prior-Aware Memorization (arXiv 2602.18733) 两个补充工作 — 前者提供**provable 多模型联合 decontamination** 的理论保证，后者把 membership inference 的精度从粗粒度推到"是否与特定 training prefix 强关联"。两者共同的方法论含义是 — n-gram 假设的反驳路径不只是"找到更敏感的探针"，而是把 contamination detection 从启发式 framework 重新建立为**有 provable guarantee 的统计 framework**。这与 §5.7 第二条 epistemic 结论吻合 — 在 mechanistic 区分 shallow recall vs deep transfer 之前，contamination 量化必须给出明确的统计保证才能成为方法学共识。

A.6 Meta-evaluation：BHI 与 daVinci-LLM 的稳定性问题

§7.7 已经讨论 Benchmark Health Index 与 daVinci-LLM 的两条 meta-evaluation 路线。Stage 0.5 scan 没有出现第三个量化 meta-evaluation 框架 — 这是 2026-Q2 评测社区的一个间接信号 — meta-evaluation 仍处于"两个工具并存但未对齐"的早期阶段。BHI 自承 model-distribution-dependence 的局限给 §7.8 第一条主线（meta-eval 自身能否经得起 perturbation test）留下开放问题。Fast Probing for In-Training LLMs (arXiv 2604.01025) 把单 checkpoint eval 从 1 小时压到 3 分钟，是工程层突破而非 meta-evaluation 层突破，仍属 §7.7 框架辩论之外。

A.7 应跟踪 vs 应怀疑

值得跟踪：MathArena Platform 的月度 ArxivMath drops、AgencyBench 的 1M-token 长 horizon 评测、Odysseys 的 rubric-based efficiency metric、Cattle Trade 的 TrueSkill 多 agent 评测。这四条是对 §5/§6/§7 开放问题的真实推进。

应保持怀疑：(1) 凡是"作者自己设计 + 自己跑 + 自己 grade"的 self-reporting 类 benchmark 在缺少外部 audit 之前应折扣；(2) 凡是 sample size < 50 + LLM-as-judge 单点 grading 的 benchmark 在 reproducibility 上有限；(3) 凡是 partial credit weight 由作者选定的 benchmark 在 ranking 稳定性上敏感于 weight 选择。这三条 caution 与 §6.4 agent benchmark crisis 的 evaluator script 脆弱性问题同根 — frontier benchmark 的可信度危机不来自 contamination 而来自 evaluator + grader + reporting 三层的脆弱性。社区需要把"benchmark for benchmark" 的 BHI 路线扩展到 "auditor for benchmark" 的更严方法学层。

报告截稿日期: 2026-05-25。Frontier Watch 附录覆盖截稿后涌现的新工作。

58 / 58