# 思维链与推理微调

张恒睿 2601210115

Zero-shot CoT / CoT / Auto-CoT / Fine-tune-CoT / DSS / SCoTD / PoT / MAmmoTH / s1 / LIMO

# 常用 Benchmark 速览

| Benchmark | 主要考察什么 | 答案形式 |
| --- | --- | --- |
| GSM8K | 小学数学应用题，需要多步算术推理 | 数值 |
| MATH500 | 竞赛数学，涵盖代数、几何等<br>MATH 测试集中的 500 题子集 | 开放答案 |
| AIME | 美国数学邀请赛<br>本报告采用 AIME24（30 题） | 整数<br>000–999 |

来源：CoT 与 s1 的任务定义，MATH500 官方数据说明。

# Zero-shot CoT：Let's think step by step

Joe 每分钟出拳 25 次，一场 5 回合，每回合 3 分钟。共出拳多少次?

**推理提示**

Q: [问题]

A: Let's think step by step.

第 1 次调用 →

**生成推理**

每回合：
3 × 25 = 75 拳

全场：
5 × 75 = 375 拳

保留完整上下文 →

**答案提取提示**

[原提示＋生成的推理]

Therefore, the answer (arabic numerals) is

第 2 次调用 →

**最终答案**

375

**同一模型，两次调用**

参考文献：[10] Kojima et al. Large Language Models are Zero-Shot Reasoners. NeurIPS 2022

# Zero-shot CoT 的效果

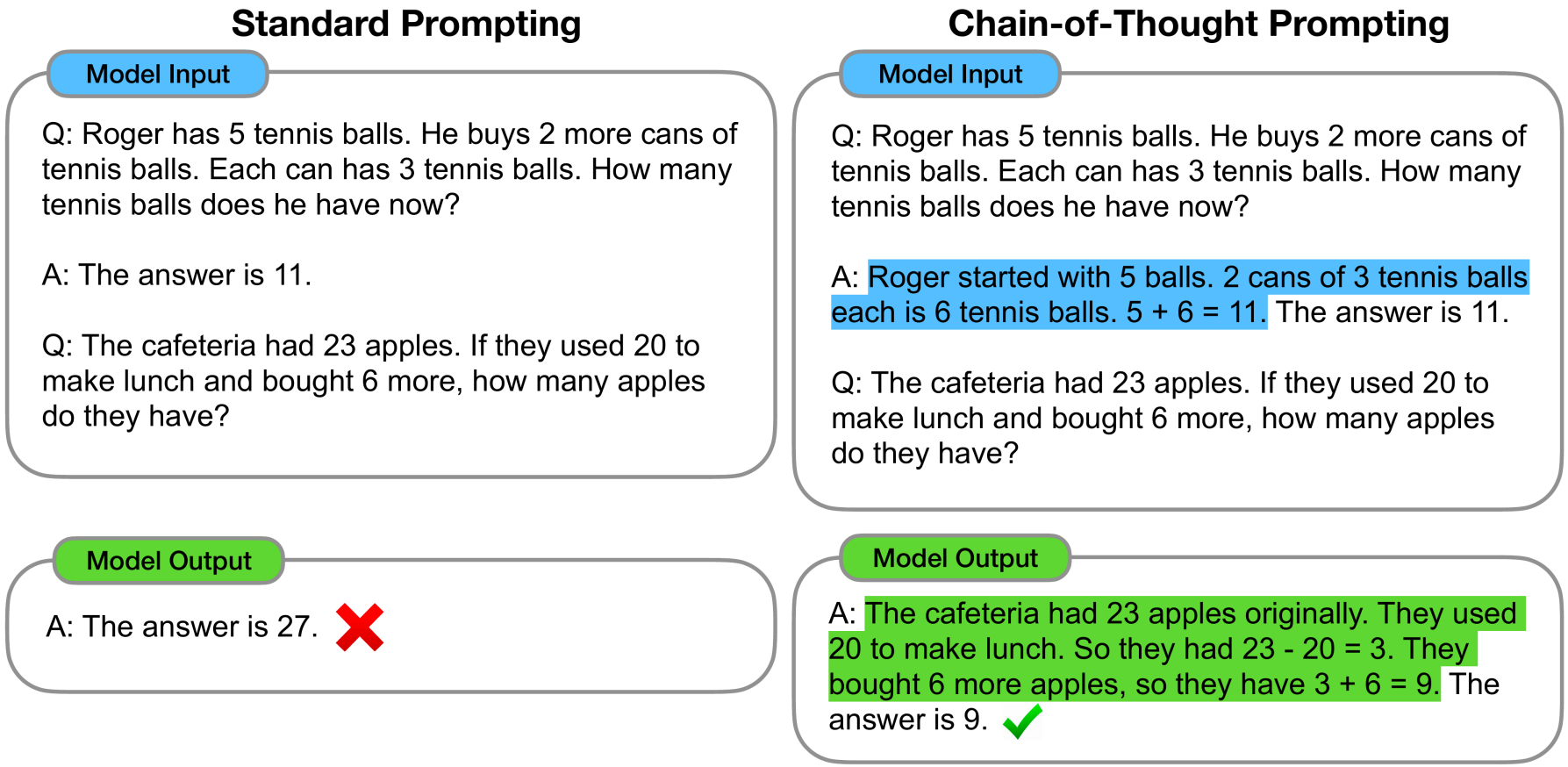
同一 text-davinci-002 模型，两种方式均不提供示范。

| 提示方式 | 输入形式 | MultiArith | GSM8K |
| --- | --- | --- | --- |
| Zero-shot<br>直接作答 | 问题＋答案提示 | 17.7 | 10.4 |
| **Zero-shot CoT** | **问题＋Let's think step by step.<br>→ 生成推理 → 提取答案** | **78.7** | **40.7** |

原表 2 的零样本设置 · 最终答案准确率（%）

参考文献：[10] Kojima et al. Large Language Models are Zero-Shot Reasoners. NeurIPS 2022

# CoT 提示中的中间步骤




原图 1：标准提示与包含中间推理的少样本提示

参考文献：[1] Wei et al. Chain-of-Thought Prompting. NeurIPS 2022

# 模型规模与 CoT 提示收益

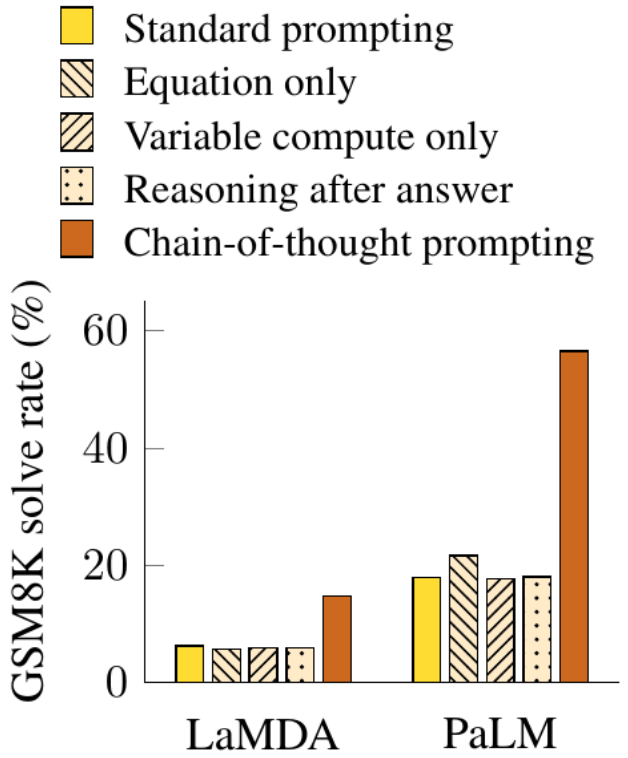
| PaLM | 标准提示 | CoT |
|---|---|---|
| 8B | 4.9 | 4.1 |
| 62B | 9.6 | 29.9 |
| **540B** | **17.9** | **56.9** |

GSM8K 准确率（%）· 原文 Table 2 节选

- PaLM 540B 的 GSM8K 准确率从 17.9% 提高到 56.9%。
- 较小模型的提升有限，8B 模型在此设置下略有下降。

参考文献：[1] Wei et al. Chain-of-Thought Prompting. NeurIPS 2022

# 中间步骤的内容与顺序



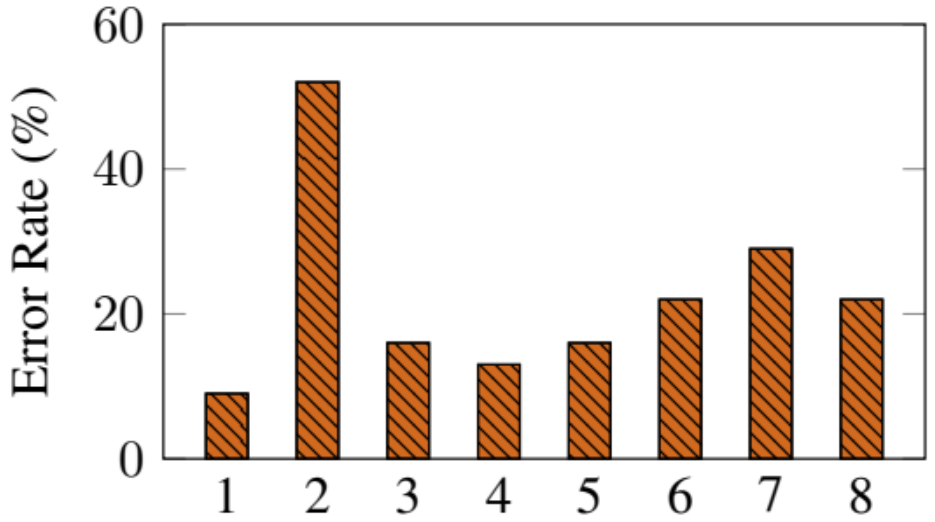

原图 5 · GSM8K · LaMDA 137B / PaLM 540B

| 图中标签 | 示范回答形式 |
| --- | --- |
| Standard prompting | 直接给答案<br>“答案是 11。” |
| Equation only | 先列方程，再给答案<br>“5+2×3=11。答案是 11。” |
| Variable compute only | 先输出点号占位，再给答案<br>“..... 答案是 11。” |
| Reasoning after answer | 先给答案，再解释<br>“答案是 11。2 罐共 6 个，5+6=11。” |
| Chain-of-thought prompting | 先推理，再给答案<br>“2 罐共 6 个，5+6=11。答案是 11。” |

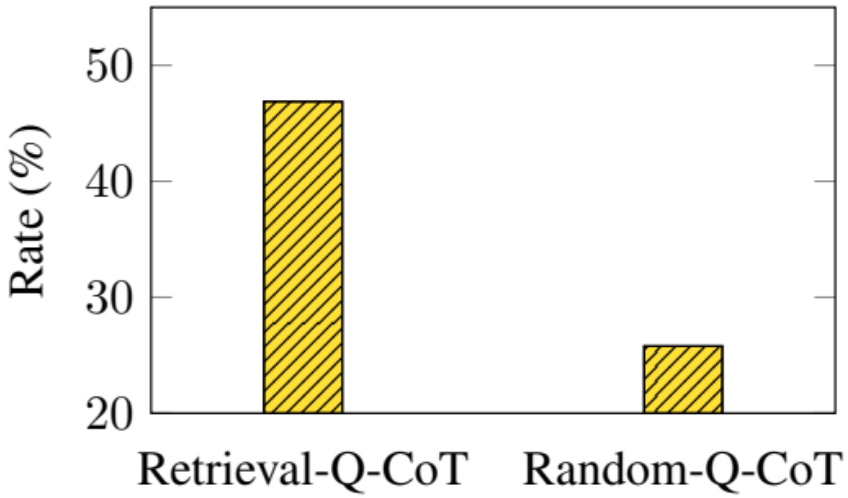
同题示例：原有 5 个球，再买 2 罐，每罐 3 个

参考文献：[1] Wei et al. Chain-of-Thought Prompting. NeurIPS 2022

# 如何选择CoT示例




原图 3：MultiArith 的 600 题，8 个语义簇




原图 2：相似题示范与随机题示范

① 同一数据集，不同问题簇表现不同。
第 2 簇错误率约 52%，明显高于其他簇。

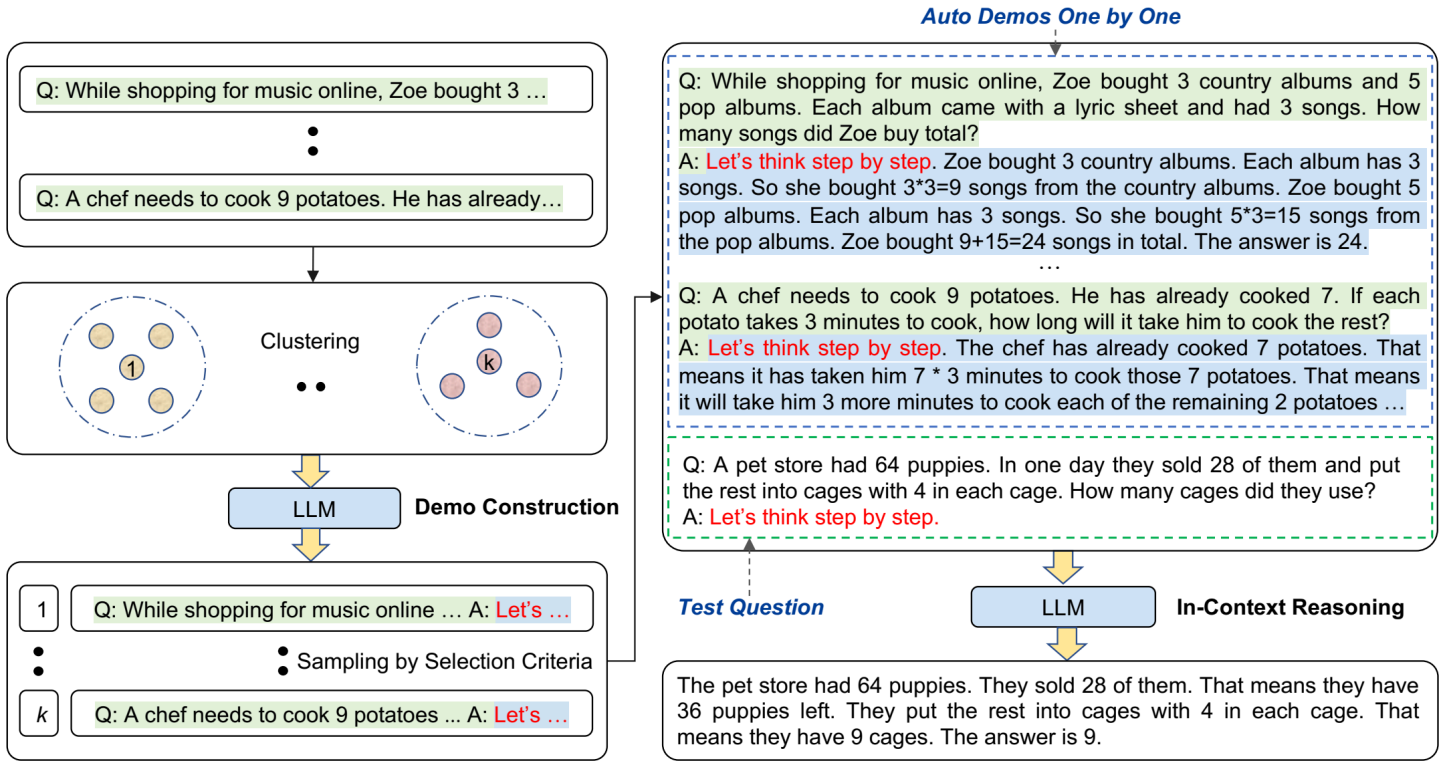
② 原先答错的 128 题，各给 8 条示范。
仍错：相似示范 46.9%，随机示范 25.8%。

- 选题动机：跨簇选择代表题，构造多样化推理示范。

参考文献：[2] Zhang et al. Automatic Chain of Thought Prompting. ICLR 2023

# Auto-CoT 的基本思路

聚类选代表题，自动生成推理与答案，再组成多样化的少样本提示。




原图 4：问题聚类、自动示范构造与上下文推理

参考文献：[2] Zhang et al. Automatic Chain of Thought Prompting. ICLR 2023

# 自动示范与人工示范的结果

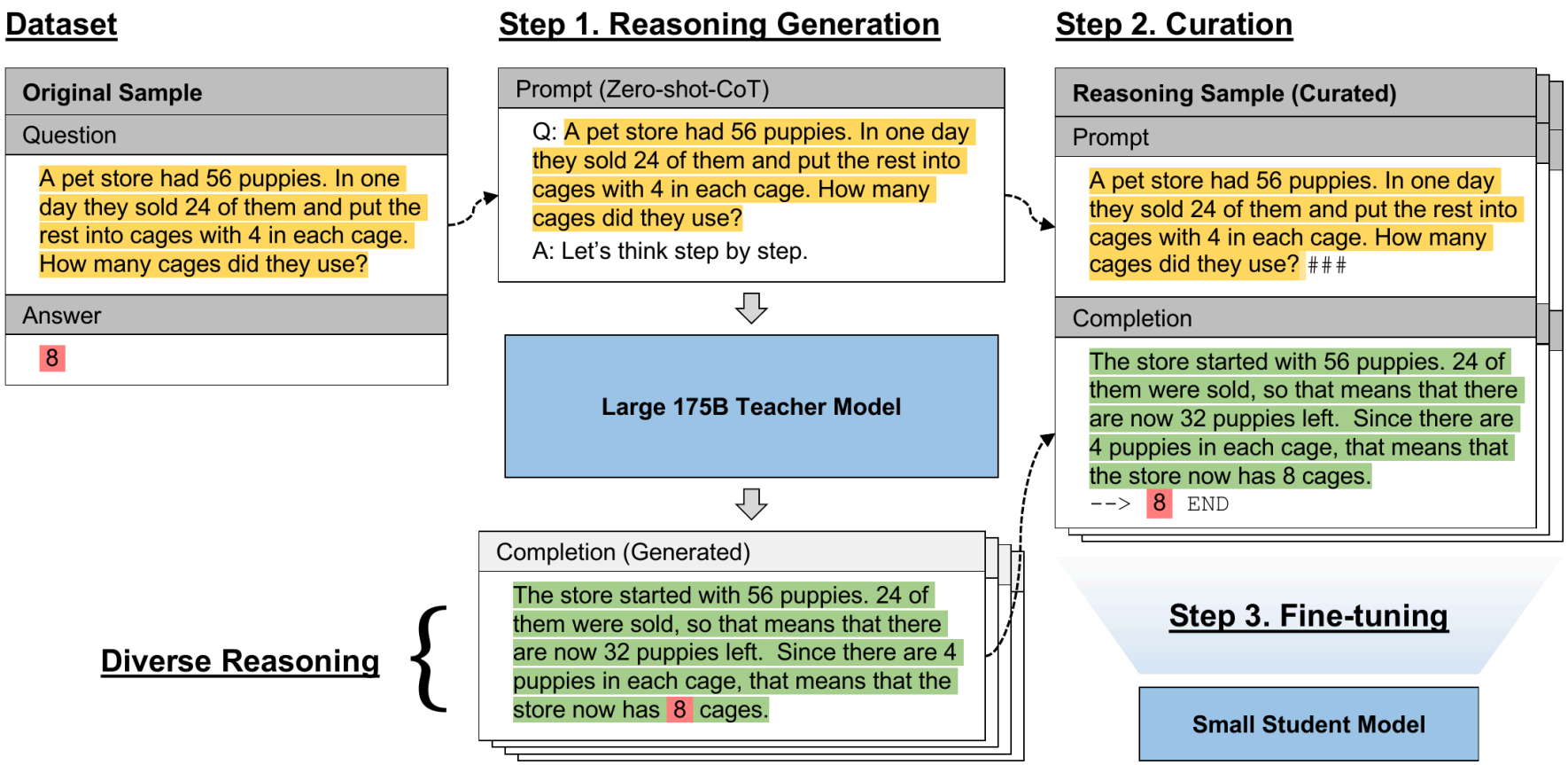
格式例子：原有 5 个球，再买 2 罐，每罐 3 个，共多少个?

| 提示方式 | 提示示例（简化） | GSM8K | SVAMP | CSQA | Coin Flip |
|---|---|---|---|---|---|
| 直接回答（有示范） | 给问答示范，不写推理。<br>示范答案如：“11。” | 15.6 | 65.7 | 79.5 | 57.2 |
| Zero-Shot-CoT | 不给示范，只在问题后加<br>`Let's think step by step.` | 40.7 | 63.7 | 64.6 | 91.4 |
| Manual-CoT | 人工写：“2 罐共 6 个，<br>加上原有 5 个，答案是 11。” | 46.9 | 68.9 | 73.5 | 97.2 |
| **Auto-CoT** | **各簇选题，由模型生成<br>“问题＋推理＋答案”示范。** | **47.9** | **69.5** | **74.4** | **99.9** |

text-davinci-002 · 准确率（%）· Table 3 四项任务节选

参考文献：[2] Zhang et al. Automatic Chain of Thought Prompting. ICLR 2023

# 教师轨迹作为学生的训练目标




原图 2：教师生成轨迹、按答案筛选、微调学生

参考文献：[3] Ho et al. Large Language Models Are Reasoning Teachers. ACL 2023

# 多条教师轨迹继续改善学生

| 6.7B 学生 | 训练与测试方式 | MultiArith | SVAMP | 日期理解 |
|---|---|---|---|---|
| Few-shot-CoT<br>无微调 | 测试时给推理示范。<br>不更新学生参数，生成理由与答案。 | 10.00 | 11.33 | 12.84 |
| 仅答案微调 | 用“问题＋标准答案”微调。<br>测试只预测答案。 | 15.00 | 20.67 | 14.41 |
| Fine-tune-CoT | 教师每题贪心生成一条，筛答案。<br>学生学习输出“理由＋答案”。 | 33.33 | 12.67 | 60.36 |
| **Fine-tune-CoT<br>多轨迹** | **教师每题采样 D 次，再筛正确轨迹。<br>用多条“理由＋答案”微调学生。** | **53.33** | **30.33** | **83.78** |

原表 1 · 准确率（%）· 教师采样次数 D：64 / 64 / 8，筛选后保留数可更少

参考文献：[3] Ho et al. Large Language Models Are Reasoning Teachers. ACL 2023

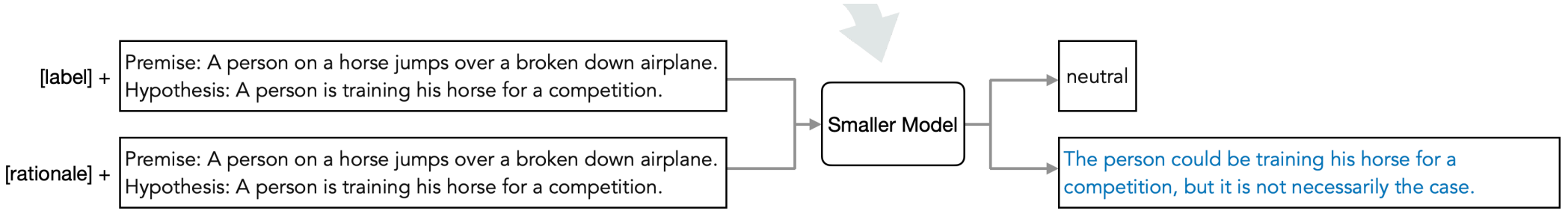
# 轨迹数量与筛选规则

| 选择方式 | 选择规则 | QuaRel | OpenBookQA | CSQA |
|---|---|---|---|---|
| 随机选 5 条 | 从答案正确的候选轨迹中，随机抽取。 | 74.8 | 61.4 | 64.5 |
| 教师似然最高 5 条 | 教师越容易生成的轨迹优先。按每 token 平均对数概率排序。 | 79.1 | 61.6 | 65.1 |
| 多样性选 5 条 | 同题轨迹按语义做层次聚类。分成 5 簇，每簇随机选一条。 | 80.6 | 62.6 | 64.8 |
| 完整采样集 | 每题采样 30 次并筛去错误答案。保留全部剩余轨迹，不再选 5 条。 | 83.8 | 67.0 | 67.0 |

OPT-1.3B · 准确率（%）· 原图 7；前三行约每题 5 条，完整集使用更多训练材料

参考文献：[5] Li et al. Symbolic Chain-of-Thought Distillation. ACL 2023

# 理由与标签的多任务训练




原图 2 下半：任务前缀不同，同一学生分别预测标签与理由

$$L = L_{label} + \lambda L_{rationale}$$

- 同一学生分别学习标签预测与理由生成。
- 加权两个任务的损失，测试时可直接预测标签。

参考文献：[4] Hsieh et al. Distilling Step-by-Step! Findings of ACL 2023

# 多任务目标改善标签预测

| 训练目标 | e-SNLI | ANLI | CQA | SVAMP |
|---|---|---|---|---|
| 仅标签 | 88.38 | 43.58 | 62.19 | 62.63 |
| 理由与标签拼成单序列 | 88.88 | 43.50 | 61.37 | 63.00 |
| **理由、标签分成两个任务** | **89.51** | **49.58** | **63.29** | **65.50** |

220M T5-Base · 100% 人工标注数据 · 准确率（%）· Table 2

参考文献：[4] Hsieh et al. Distilling Step-by-Step! Findings of ACL 2023

# PoT 的提示与程序执行

```
# Question: Toulouse has twice as many sheep as Charleston.
Charleston has 4 times as many sheep as Seattle. How many
sheep do Toulouse, Charleston, and Seattle have together if
Seattle has 20 sheep?
# Answer this question by implementing a solver() function.
def solver():
    # Let's write a Python program step by step, and then return
the answer
    # Firstly, we need define the following variable:
```

原图 3 右半：零样本输入提示

```
Seattle = 20
Charleston = Seattle * 4
Toulouse = Charleston * 2
# Then, we need to calculate the sum of the three
sum = Seattle + Charleston + Toulouse
# Finally, we need to return the answer
return sum
```

原图 3 右半：模型生成的程序片段

- 模型理解数量关系并生成程序，Python 执行得到 260。

参考文献：[6] Chen et al. Program of Thoughts Prompting. TMLR 2023

# PoT 的数值推理结果

| 提示方式 | GSM8K | AQuA | SVAMP | TabMWP | FinQA |
|---|---|---|---|---|---|
| CoT | 63.1 | 45.3 | 76.4 | 65.2 | 40.4 |
| CoT + 计算器 | 65.4 | 45.3 | 77.0 | 65.8 | — |
| **PoT** | **71.6** | **54.1** | **85.2** | **73.2** | **64.5** |

原表 2 节选 · Codex / code-davinci-002 · 少样本 · 贪心解码 · 准确率（%）

- CoT + 计算器：先生成文字推理，再事后重算其中的算式。
- 例子：把“2 × 3 = 7”修正为“2 × 3 = 6”。
- GSM8K：CoT 63.1%，加计算器 65.4%，PoT 71.6%。

参考文献：[6] Chen et al. Program of Thoughts Prompting. TMLR 2023

# 变量语义与分步计算

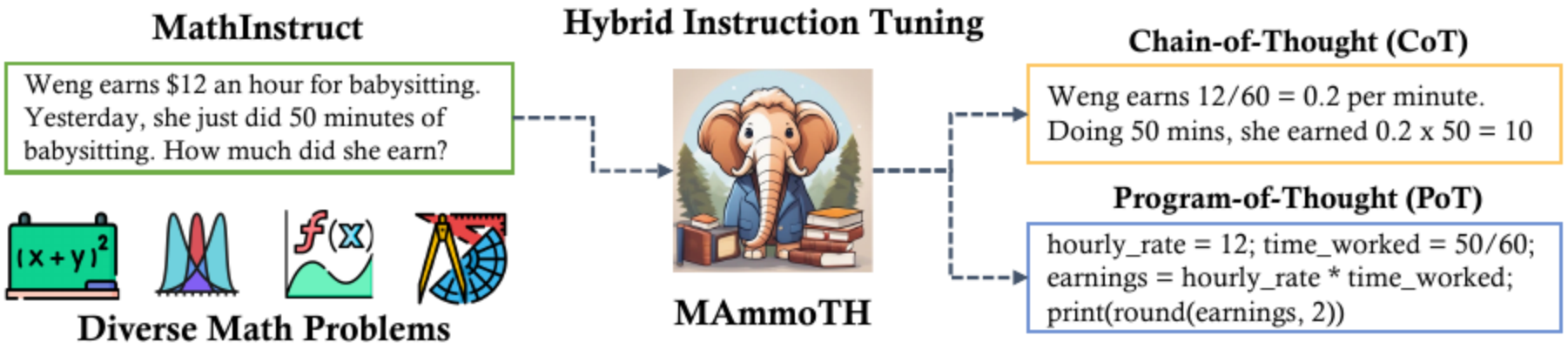
同一小题：原有 5 个球，再买 2 罐，每罐 3 个。三段程序均输出 11。

| 程序生成方式 | 输出程序示例 | GSM8K | SVAMP | FinQA |
|---|---|---|---|---|
| PoT | `initial=5`<br>`added=2*3`<br>`print(initial+added)` | 71.6 | 85.2 | 64.5 |
| 去除语义绑定 | `a=5`<br>`b=2*3`<br>`print(a+b)` | 60.2 | 83.8 | 61.6 |
| 直接生成最终方程 | `print(5+2*3)` | 45.8 | 81.9 | 58.9 |

数值来自原表 6 · 准确率（%）· 代码片段用于说明变量命名与分步结构

参考文献：[6] Chen et al. Program of Thoughts Prompting. TMLR 2023

# MAmmoTH 的混合指令微调

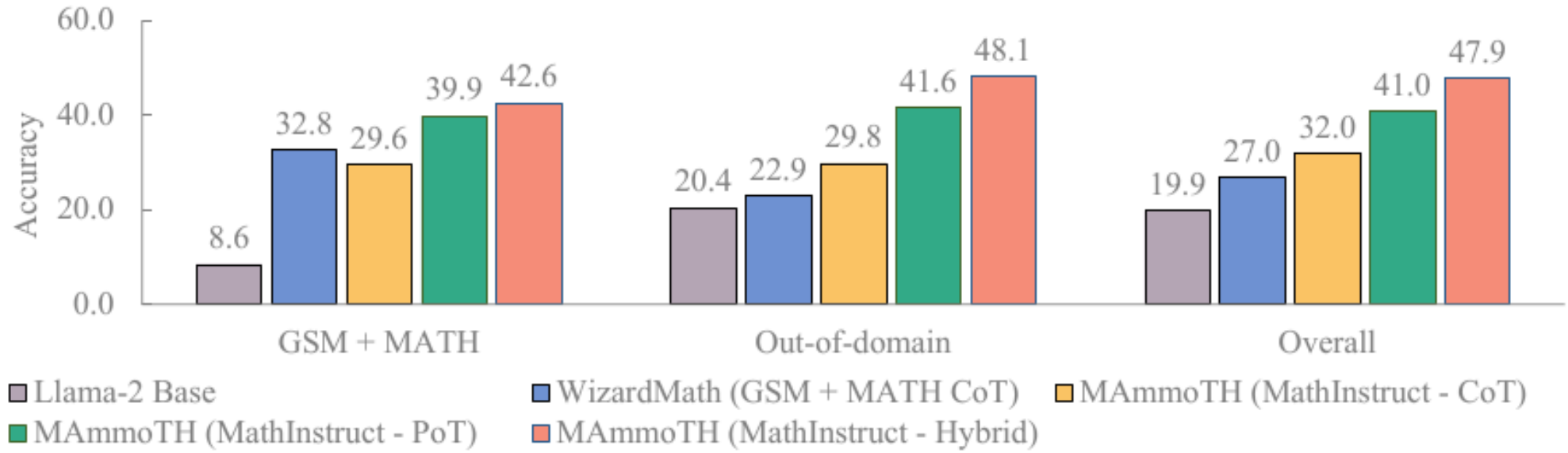



原图 1 上半：多领域问题的 CoT / PoT 混合监督

- 约 26 万条指令：72% CoT，28% PoT。
- 统一为指令—解答格式，对 Llama-2 / Code Llama 做普通 SFT。

参考文献：[7] Yue et al. MAmmoTH. ICLR 2024

# CoT 与 PoT 混合训练




7B Llama-2 · 准确率（%）；Overall 为九个数据集的平均，细分见表 9。

- 7B Llama-2：CoT 子集 32.0，PoT 子集 41.0，混合 47.9。
- 混合训练在九项任务平均分上最高，训练数据量与组成也同时变化。

参考文献：[7] Yue et al. MAmmoTH. ICLR 2024

# PoT 优先的混合解码

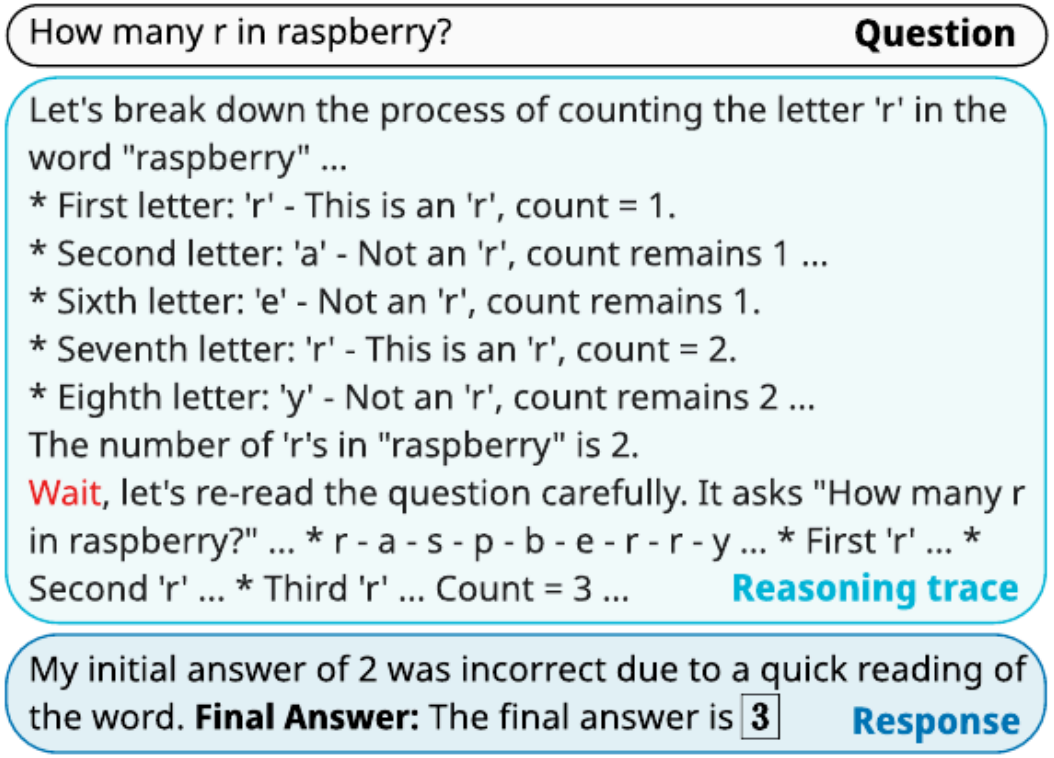
| MAmmoTH-7B 解码 | GSM8K | MATH | SimulEq | MMLU-Math | 九项平均 |
|---|---|---|---|---|---|
| CoT | 50.5 | 10.4 | 18.9 | 39.9 | 33.0 |
| PoT | 51.6 | 28.7 | 48.2 | 44.6 | 46.1 |
| **Hybrid** | **53.6** | **31.5** | **41.2** | **42.6** | **47.9** |

原表 10 节选 · 同一个已完成混合 SFT 的 MAmmoTH-7B · 准确率（%）

- 程序可执行时采用结果，执行失败时改用 CoT。
- 整体均值提高，但并非每个数据集都优于单用 PoT。

参考文献：[7] Yue et al. MAmmoTH. ICLR 2024

# s1 的训练与预算强制




原图 3：追加 Wait 后，模型重新检查字母计数

- 训练：用 1,000 条教师推理轨迹做普通 SFT。
- 推理：在结束边界追加 Wait，继续检查当前解答。

参考文献：[8] Muennighoff et al. s1: Simple test-time scaling. EMNLP 2025

# s1 的 SFT 收益与预算强制收益

| 模型 / 设置 | AIME24 | MATH500 | GPQA-D |
|---|---|---|---|
| Qwen2.5-32B-Instruct | 26.7 | 84.0 | 49.0 |
| **s1 · 无预算强制** | **50.0** | **92.6** | **56.6** |
| **s1 · 预算强制** | **56.7** | **93.0** | **59.6** |

原表 1 节选 · 准确率（%）· BF 为主表汇总，各指标的推理设置可不同

- 1K 轨迹 SFT 已带来主要提升。
- 预算强制进一步提高准确率，同时增加推理计算。

参考文献：[8] Muennighoff et al. s1: Simple test-time scaling. EMNLP 2025

# 精选 1K 与全量 59K 的训练比较

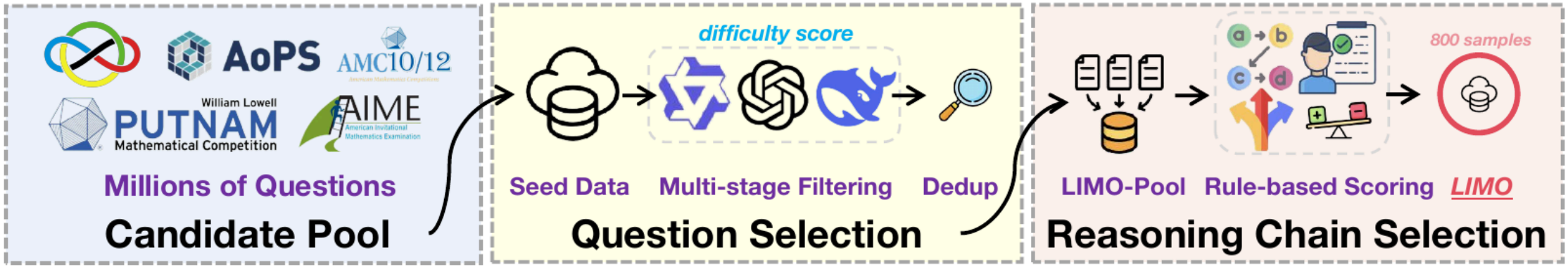
| 训练数据 | AIME24 | MATH500 | GPQA-D | H100 GPU · h |
|---|---|---|---|---|
| 随机 1K | 36.7 | 90.6 | 52.0 | — |
| **精选 s1K** | **50.0** | **93.0** | **57.6** | **7** |
| 全部 59K | 53.3 | 92.8 | 58.1 | 394 |

原表 2 与 §5.1 · BF：约 30K 思考上限 · 准确率（%）· GPU · h 仅计学生 SFT

- 全量训练消耗约 56 倍 H100 GPU 小时。
- 三项差值的 95% 置信区间均包含 0，观察到的增益较小。

参考文献：[8] Muennighoff et al. s1: Simple test-time scaling. EMNLP 2025

# LIMO 的 800 条精选轨迹




原图 2：候选问题、难度筛选与推理链评分

- 排除易题，保留强模型 32 次 rollout 中仅答对 1–3 次的问题。
- 多教师生成轨迹，按规则评分筛出 800 条，再做普通 SFT。

参考文献：[9] Ye et al. LIMO: Less is More for Reasoning. arXiv v3, 2025

# LIMO 的推理链评分规则

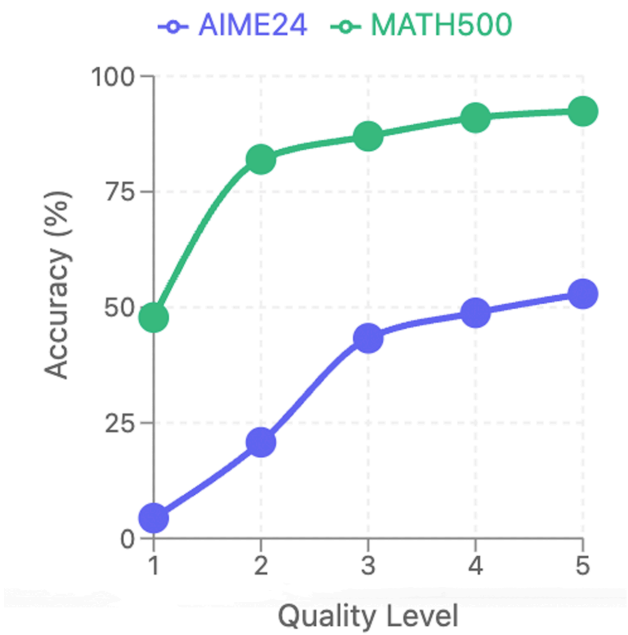
| 评分维度 | 权重 | 规则信号与原文例词 |
| --- | --- | --- |
| 充分展开的推理 | 30% | 用解答长度衡量推理展开程度。 |
| 自我验证 | 20% | 统计验证相关词，如 `check`、`verify`。 |
| 探索式方法 | 25% | 统计试探性表达，如 `perhaps`、`might`。 |
| 自适应粒度 | 25% | 统计连接短语，如 `therefore`、`since`。 |

LIMO v3 §3.1.2

- 每题选最高分解答，再对 2,125 个问答对排序，取前 800 条。

参考文献：[9] Ye et al. LIMO: Less is More for Reasoning. arXiv v3, 2025

# LIMO 的表现与轨迹质量消融




原图 3：固定 500 题的解答质量消融

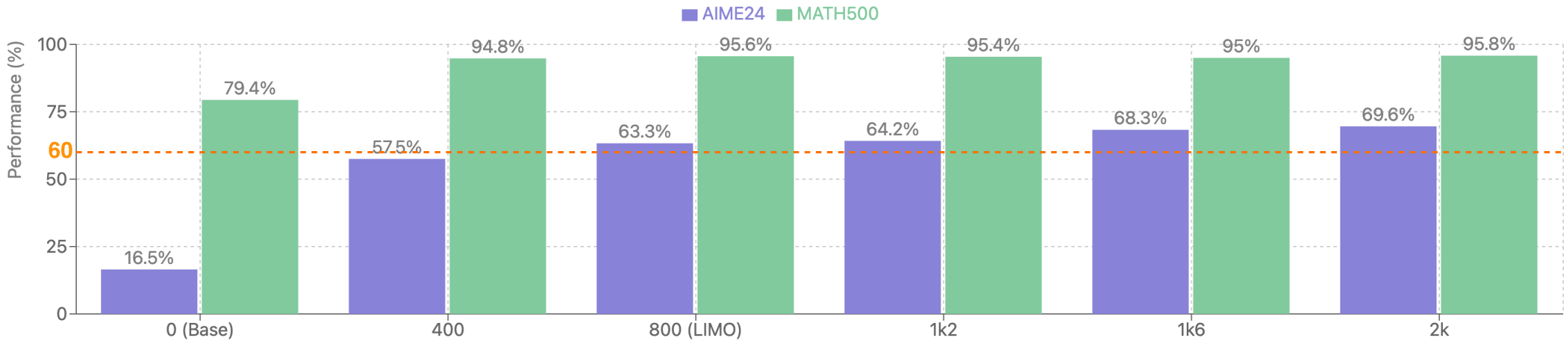
| 训练数据 | AIME24 | MATH500 |
|---|---|---|
| 原始基座 | 16.5 | 79.4 |
| OpenThoughts-114K | 50.2 | 80.6 |
| **LIMO • 800** | **63.3** | **95.6** |

OpenThoughts-114K：11.4 万条 DeepSeek-R1 推理示范。

- 800 条精选轨迹改善同一基座的数学表现。

参考文献：[9] Ye et al. LIMO: Less is More for Reasoning. arXiv v3, 2025

# 少量精选数据与继续扩充的收益




LIMO：800 → 2,000 条，AIME24 63.3 → 69.6；MATH500 95.6 → 95.8。

- LIMO 扩至 2,000 条，AIME24 从 63.3% 提高到 69.6%。
- 少量精选数据已很有效，增加数据仍可能继续改善。

参考文献：[8] Muennighoff et al. s1: Simple test-time scaling. EMNLP 2025
[9] Ye et al. LIMO: Less is More for Reasoning. arXiv v3, 2025

# 提示、训练目标与推理计算

| 四组工作 | 主要改变 | 测试时的行为 |
| --- | --- | --- |
| Zero-shot / CoT / Auto-CoT | 推理指令或含推理的示范 | 生成推理，再输出答案 |
| Teachers / DSS / SCoTD | 用教师理由或轨迹监督学生 | 生成推理或直接预测标签 |
| PoT / MAmmoTH | 程序提示与 CoT / PoT 混合 SFT | 生成并执行程序，必要时回退 |
| s1 / LIMO | 精选少量长推理轨迹做 SFT | 生成推理，s1 额外控制预算 |

依据上述工作的方法与实验归纳