

直接行为优化： 释放轻量级 LLM 的潜力

Hongming Yang^{1,3*}, Shi Lin^{1,3*}, Jun Shao^{1†},
Changting Lin^{2,4}, Donghai Zhu^{1†}, Meng Han^{2,4}, Qinglei Kong⁵

¹Zhejiang Gongshang University, Hangzhou, China

²Zhejiang University, Hangzhou, China

³Binjiang Institute of Zhejiang University, Hangzhou, China

⁴GenTel.io, Hangzhou, China ⁵Harbin Institute of Technology, Shenzhen, China

{ tzoceanyang, linshizjgsu, chn.junshao, linchangting } @gmail.com

zhudonghai@zjgsu.edu.cn, mhan@zju.edu.cn, kongqinglei@hit.edu.cn

Abstract

轻量级大型语言模型 (LwLLMs) 是经过参数简化和优化的模型，旨在可以在消费级硬件上高效运行，提供在资源效率、成本效益和数据隐私等方面的显著优势。然而，这些模型通常在推理和推断能力上受到限制，这限制了它们在复杂任务上的表现并限制了其实际应用性。此外，现有的提示优化方法通常依赖于大量的人工努力或最先进的 LLMs 的元认知能力，使它们对 LwLLMs 的效果较低。为了应对这些挑战，我们引入了 DeBoP，一种从链式思维 (CoT) 提示技术中原创的新的 Direct Behavior Optimization Paradigm。与 CoT 提示不同，DeBoP 是一种自动优化方法，直接关注 LwLLMs 行为的优化。具体来说，DeBoP 将复杂提示的优化转化为使用无梯度蒙特卡罗树搜索的离散、可量化执行序列的优化。我们在七个具有挑战性的任务上评估了 DeBoP，这些任务中最先进的 LLMs 表现卓越，而 LwLLMs 却表现欠佳。实验结果表明，DeBoP 显著优于最近的多数任务上的提示优化方法。特别是，DeBoP 优化的 LwLLMs 在大多数任务上超越了 GPT-3.5，同时与其他自动提示优化方法相比，将计算时间减少了大约 60%。¹

1 引言

大型语言模型 (LLMs) 已经革新了自然语言处理领域，在复杂文本分析 (?)、高级数学推理 (?) 以及创意写作 (?) 等各种任务中取得了显著的性能。然而，这些进步伴随着显著的计算需求，例如通过新的 LLaMA 模型，包含 405B 个参数 (?)。这一挑战推动了大量研究工作来开发高效的替代方案，从而引发了轻量级 LLMs (LwLLMs) 的出现，这些模型针对消费级硬件进行了优化 (?)。值得注意的是，具有 3B 参数模型的 LwLLMs 可以在像 NVIDIA

@GeForce RTX 4060 (8GB VRAM) 这样的主流 GPU 上运行。此外，LwLLMs 在安全性和合规性方面优于通过 API 访问的模型 (?)，因为它们完全在机构的内部网络内运行。这显著减轻了通过公共网络泄露敏感数据的风险 (??)。

尽管轻量级大语言模型 (LwLLMs) 在资源效率和加强安全性方面表现出色，它们在复杂任务的高级推理和多步骤推理方面常常面临困难 (?)。解决这一问题的一个有希望的方法是提示优化，这大致可分为手动和自动技术。手动方法虽然有效，但需要大量的人力投入，使其在资源受限的环境中不切实际。另一方面，自动方法具有更广泛的适用性，但通常依赖于大语言模型的元认知能力 (?)，比如迭代自我反思 (??)、规划 (??) 和详细推理过程追踪 (??)，而这些是轻量级大语言模型难以高效处理的。更重要的是，这些方法中的层次推理结构可能导致快速错误传播和积累 (如图 1 所示)，这最终可能使性能下降到甚至低于更简单的方法水平，比如直接提示 (参见表 1 以了解详情)。

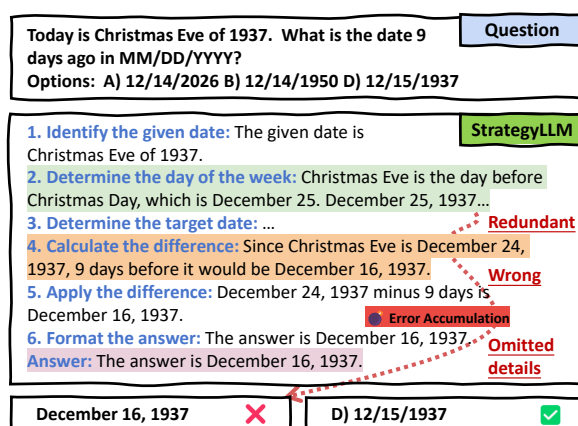


Figure 1: 一个玩具示例展示了 StrategyLLM (?) 在 LwLLMs 上如何失败。StrategyLLM 使用的策略在蓝色中被强调。由于 LwLLMs 有限的推理能力，redundant 推理、wrong 推理和 omitted details 的累积导致了最终答案的不正确。

为了解决这些挑战，我们在本文中引入了

*Equal contribution

†Corresponding author

¹DeBoP 的官方实现可在以下网址找到: <https://github.com/VastOcean-Yang/DeBoP.git>。这项工作已被 ACL 2025 接受。

DeBoP, 一种新颖的 **D**irect **B**ehavior **O**ptimization **P**aradigm, 它直接优化 LwLLMs 的行为。受表 1 中实验结果的启发, DeBoP 采用了基本的 Chain-of-Thought (CoT) 提示 (?), 这涉及一系列的示例。此外, DeBoP 将 LwLLMs 的行为优化重新表述为优化示例中的解决方案。具体来说, 它将解决方案解耦为由一系列关键步骤及其相应执行构成的结构化计划, 然后通过应用无梯度的蒙特卡洛树搜索 (MCTS) 来优化它们。这消除了现有优化方法所需的隐式元认知机制。此外, DeBoP 具有不依赖外部 LLM 接口且不需要耗时的手动提示工程的优势, 进一步突显了其适用性。

本文的贡献可以总结如下: 1) 我们提出了一种针对 LwLLMs 的自动化行为优化方法, 该方法显著扩大了 LLMs 在资源和安全敏感环境中的适用性。2) 我们基于无梯度的蒙特卡罗树搜索 (MCTS) 框架设计了一种外部优化器, 有效地量化提示设计, 并且无需高级元认知能力。3) 我们实验证明, 在多项具有挑战性的任务中, DeBoP 的有效性已被验证, 其中最新的 LLMs 通常表现优异 (?), 仅 LLaMA3-8B 和 DeBoP 的结合可以在大多数任务中超越 GPT-3.5, 同时与其他自动化提示优化方法相比, 减少了大约 60% 的 LwLLMs 时间消耗。

2 方法论

如前所述, DeBoP 是从 CoT 提示 (CoT Prompting) 中衍生出来的, 它采用了一系列的演示。在每个演示中, 一个问题与其对应的解决方案配对, 这代表了轻量级大语言模型 (LwLLM) 的行为。DeBoP 专注于自动优化这些解决方案, 并将每个解决方案分解为关键步骤计划及其相应的执行。因此, DeBoP 解决的是关键步骤计划和执行的生成、解决方案的优化, 以及 LwLLM 执行它们的指令。因此, DeBoP 包含四个阶段: PLANNING、COLLECTING、MCTS 和 TEACHING。相应的高层次描述如图 2 所示, 各个阶段的详细描述在第 ?? 节到第 ?? 节中提供。

2.1 目标和框架概述

我们的目标是自动在行为空间 $\mathcal{B}$ 中找到最佳行为 b^* , 该行为最大化其在 N 输入输出对的开发集上的整体性能, 记作 $\mathcal{S}_{\text{dev}} = \{(x_i, y_i)\}_{i=1}^N$ 。在我们的公式中, 行为空间 $\mathcal{B}$ 被定义为 LwLLM 在不同结构化演示的指导下可能展示的所有可执行行为轨迹的集合。

每种行为 $b \in \mathcal{B}$ 对应于 LwLLM 在响应任务时采取的一整套行动或决策轨迹, 该任务通过结构化示例 d_b 实例化。由于动作序列和示例

配置的组合性质, 空间 $\mathcal{B}$ 大到无法直接构造。因此, 我们将 $\mathcal{B}$ 视为一个隐式可搜索的空间, 通过引导探索可以识别出有前途的行为。

基于 DeBoP, 我们将原始优化问题重新表述为寻找一种示范 d_b , 以诱导出高效的行为 b 。行为的有效性通过一个函数来评估:

$$\text{Eval}(\text{LwLLM}(d_b), \mathcal{S}_{\text{dev}}),$$

, 该函数测量在 d_b 的引导下, LwLLM 在开发集上的输出质量。

形式上, 我们旨在识别:

$$b^* = \arg \max_{b \in \mathcal{B}} \text{Eval}(\text{LwLLM}(d_b), \mathcal{S}_{\text{dev}}).$$

。为了获得如图 2 所示的最优行为 b^* , 我们的框架分四个阶段进行: (1) PLANNING 生成一组可行的初始关键步骤计划 $\mathcal{S}_{\text{plan}}$, 其中每个计划都用标准化的 JSON 格式编码, 以减轻 LwLLMs 层次推理的复杂性。(2) COLLECTING 在开发集 $\mathcal{S}_{\text{dev}}$ 上执行并评估这些计划, 选择表现优异的示范形成种子集 $\mathcal{S}_{\text{demo}}$ 以进行迭代改进。(3) MCTS 使用强化学习, 具体来说是无梯度的蒙特卡罗树搜索算法, 构建示范搜索森林以探索和优化示范的解决方案。(4) 最后, TEACHING 将找到的最佳示范节点应用于通过教学机制指导 LwLLM, 从而诱导出最优行为 b^* 。

更多的算法和实现细节将在以下几节中提供。

在这个阶段, DeBoP 旨在生成一组可行的关键步骤计划 $\mathcal{S}_{\text{plan}} = \{p_1, p_2, \dots, p_k\}$, 其中 p_i 的形式为

{ "<Key Step 1>": " ", "<Key Step2>": " ", ... }

这些关键步骤计划的生成算法包括两个阶段。

在第一阶段中, 我们使用一个通用的元提示来指导 LwLLM 从一些领域内的示例中综合生成特定任务的指导方针。这些指导方针是简明的、指导性的摘要, 描述了如何将任务分解为可推广的推理步骤 (例如, 对于 DU 任务, “提取参考日期” 或 “减去时间间隔”)。值得注意的是, 尽管这些指导方针是针对特定任务量身定制的, 但用于生成它们的元提示是与模型对齐但与任务无关的, 因此可以在各种领域中重复使用。

在第二阶段, 生成的指导原则被转换成一种结构化、标准化的 JSON 格式的计划, 其中每一步都被表示为一个映射到空字符串的键。这种明确的结构通过减少歧义和认知负担简化了 LwLLMs 的后续推理。由于篇幅限制, 详细的模板在附录 ?? 中提供。

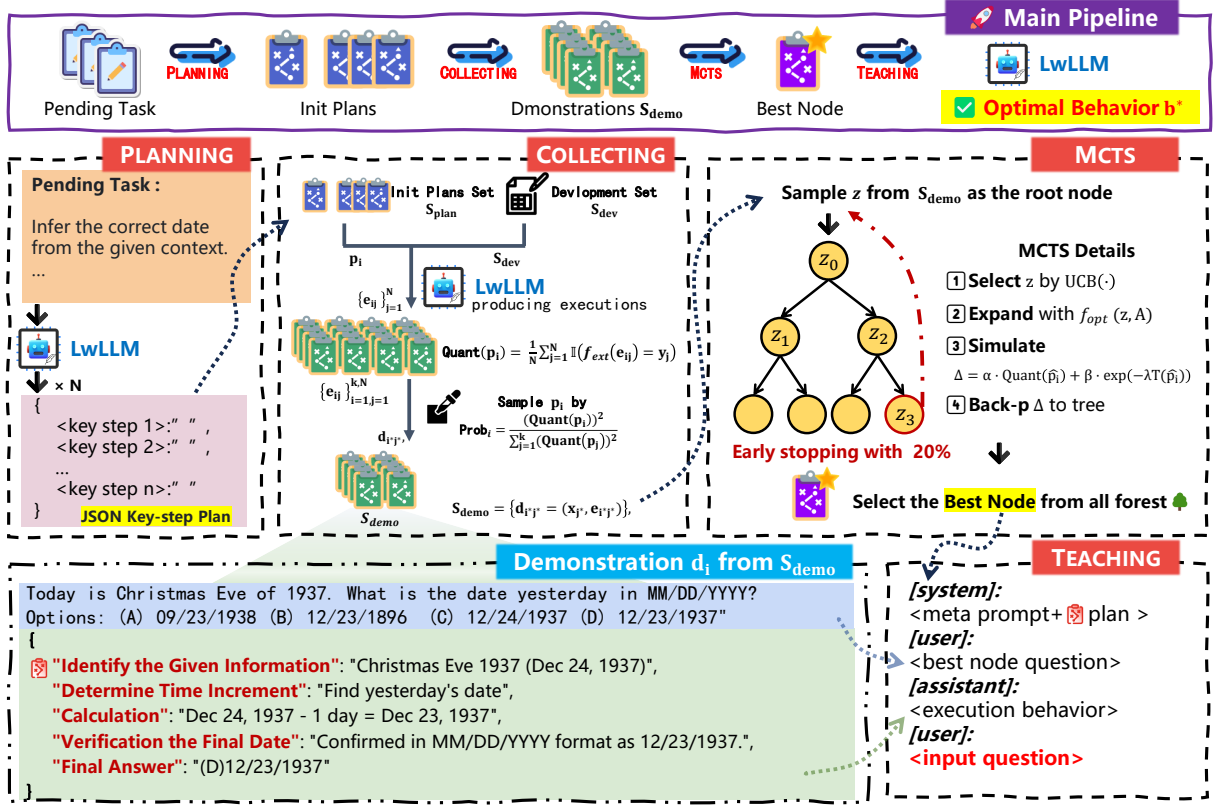


Figure 2: DeBoP 框架概述。如左下角所示，演示作为我们优化的主要目标，包括一个问题，一个计划，以及相应的执行行为。“待处理任务”指需要针对 LwLLM 优化其性能的下游任务。

在从 PLANNING 获得关键步骤计划集 $\mathcal{S}_{\text{plan}}$ 之后，DeBoP 通过以下两个阶段构建高性能演示集 $\mathcal{S}_{\text{demo}}$ 。在第一阶段，我们请求 LwLLM 完成 $\mathcal{S}_{\text{dev}}$ 中每个 x_j ($i = 1, \dots, N$) 上的每个关键步骤计划 p_i ($i = 1, \dots, k$) 的值，产生格式为 e_{ij} 的执行结果。随后，通过

$$\text{Quant}(p_i) = \frac{1}{N} \sum_{j=1}^N \mathbb{I}(f_{\text{ext}}(e_{ij}) = y_j),$$

来量化每个关键步骤计划 p_i 的性能，其中 f_{ext} 是从 e_{ij} 提取最终预测的启发式函数，而当 $\star$ 为 True 时， $\mathbb{I}(\star)$ 输出 1，否则输出 0。

在第二阶段中，我们通过对性能指标进行非线性转换来确定每个计划 p_i 的选择概率，使用

$$\text{Prob}_i = \frac{(\text{Quant}(p_i))^\alpha}{\sum_{j=1}^k (\text{Quant}(p_j))^\alpha},$$

其中在我们的实验中 $\alpha = 2$ 是²。一旦选择了一个关键步骤计划 p_{i^*} ($i^* \in \{1, \dots, k\}$)，我们随机从 $\{1, \dots, N\}$ 中选择一个 j^* 使得 $f_{\text{ext}}(e_{i^*j^*}) = y_{j^*}$ 成立。这就产生了一个选定的

²我们使用 $\alpha = 2$ 来稍微加强选择压力，以支持表现更好的计划

示例 $d_{i^*j^*} = (p_{i^*}, x_{j^*}, e_{i^*j^*})$ 。最终，所需的集合 $\mathcal{S}_{\text{demo}}$ 构建如下：

$$\mathcal{S}_{\text{demo}} = \{d_{i^*j^*} = (x_{j^*}, e_{i^*j^*})\}.$$

在此阶段，DeBoP 利用 $\mathcal{S}_{\text{demo}}$ 中的 d_i 的演示从 COLLECTING 中识别可能的最佳演示 d_{b^*} 。具体来说，可能的最佳演示 d_{b^*} 是从新构建的演示森林中选出的。在这个森林中，每个演示树的根节点对应于来自 $\mathcal{S}_{\text{demo}}$ 的一个随机演示 d_i ，而树的所有其他节点都是根节点的变体。这些节点的生成过程遵循四个阶段：选择、扩展、模拟和反向传播，所有这些都采用了 MCTS 框架。

对于正在构建中的给定演示树，我们首先根据上置信界 (UCB) 选择一个节点 z^* 进行扩展。特别地，我们使用以下公式以广度优先搜索类型的方法选择一个未扩展的节点 z^* ：

$$z^* = \arg \max_{z \in \text{children}(z_p^*)} \left(\frac{Q(z)}{N(z)} + c \sqrt{\frac{2 \ln N(z_p^*)}{N(z)}} \right)$$

其中 z_p^* 表示节点 z^* 的父节点，孩子 (z) 表示节点 z 的子节点集合， $Q(z)$ 是节点 z 的累积奖励， $N(z)$ 是节点 z 的访问次数， c 是探索系数。注意，如果一个节点没有足够的子节点，它被

视为未扩展，并且其奖励值被初始化为零，并在反向传播阶段进行更新。

一旦在选择阶段选择了一个未扩展的节点 z^* ，我们随机选择一种节点演化方法来应用于 z^* 以获得一个新节点 z 。然后，将这个新节点 z 添加到树中，作为所选节点 z^* 的子节点。基本的演化方法如下。

整合：请求 LwLLM 识别可以合并以简化推理过程、提高连贯性的关键步骤。

分解：请求 LwLLM 识别可以进一步分解以提高执行简便性和准确性的关键步骤。

详细说明：请求 LwLLM 扩展推理流程，以确保问题解决过程中的所有细节无遗漏。

剪枝：请求 LwLLM 在演示中删除最不重要的关键步骤及其相应的执行，以提高效率。

重采样：请求 LwLLM 重采样以生成新的示范。

简化：请求 LwLLM 简化和重组推理流程，以提高简洁性、一致性和逻辑清晰度。

上述过程可以表示为 f_{opt} ：

$$z \leftarrow f_{\text{opt}}(z^*, \mathcal{A}),$$

，其中 $\mathcal{A}$ 表示六个节点演化方法的集合。由于篇幅限制，每种方法的详细提示在附录 ?? 中提供。

在添加新节点 z 之后，我们请求 LwLLM 通过一种教学机制来学习相应的演示 $\hat{d} = (\hat{p}_i, \hat{x}_j, \hat{e}_{ij})$ （有关教学机制的详细信息，请参见第 ?? 节）。随后，我们运行一个类似于 COLLECTING 第一阶段的过程来获得 $\text{Quant}(\hat{p}_i)$ 。唯一的区别是，与 COLLECTING 中的过程相比，LwLLM 现在可以从演示 $\hat{d}$ 中获得额外的一次执行 $\hat{e}_{ij}$ 。

最后，我们计算 $\hat{p}_i$ 的 Δ 值为：

$$\Delta = \underbrace{\alpha \cdot \text{Quant}(\hat{p}_i)}_{\text{Execution Accuracy}} + \underbrace{\beta \cdot \exp(-\lambda T(\hat{p}_i))}_{\text{Time Efficiency}},$$

，其中 $T(\hat{p}_i)$ 是在 S_{dev} 中的每个 x_j 上使用关键步骤计划 $\hat{p}_i$ 时的平均执行时间，而 α 、 β 和 λ 是控制准确性和时间效率之间权衡的超参数。在我们的实验中，它们分别被设置为 1、1 和 0.5。

在此阶段，从模拟阶段获得的节点 z 的值 Δ 将被回溯到同一棵树的根节点，并且相应的访问计数也应相应更新。这些操作可以表示为：

$$Q(z') \leftarrow Q(z') + \Delta, N(z') \leftarrow N(z') + 1,$$

，其中 z' 是从节点 z 到相应根节点路径上的节点。

这四个阶段通常为预定义的迭代次数逐步执行，在此期间，示例演示会被逐步更新。然而，

为了提高计算效率，我们在每次迭代中加入了概率性的提前终止。在我们的实验中，提前终止的概率设为 20%。最终可能的最佳示例 d_{b^*} 是通过累积奖励最高的节点在构建的森林中确定的。

获得可能的最优示范 $d_{b^*} = (p'_i, x'_j, e'_{i,j})$ 后，我们将其整合到 LwLLM 的对话历史中，如图 2 的 TEACHING 部分所示。这个过程通过嵌入最优示范有效地“教导”LwLLM，引导其复制所期望的行为。通过这样做，我们确保 LwLLM 遵循已被证明有效的高质量执行模式，同时兼顾模型固有的推理限制。

3 实验

3.1 任务和数据集

语言理解和复杂推理是评估 LLM 自然语言处理能力的两个关键方面。因此，在我们的实验中，我们重点评估 DeBoP 在这两个领域的表现。选定用于评估的任务来自 BIG-Bench Hard (BBH) (?)，这是 BIG-Bench 任务 (?) 中的一个具有挑战性的子集。具体来说，选定的任务包括狡辩题 (SNK)、消歧问答 (DQA)、错位词语 (HB)、表格中的企鹅 (PIT)、数据理解 (DU)、逻辑演绎 (LD)、以及电影推荐 (MR)。选择这些任务的原因是因为它们共同测试语言理解（例如，SNK, DQA, HB）和复杂推理能力（例如，PIT, DU, LD, MR）。

为了突出 DeBoP 的优势，我们将其与两类提示优化方法进行比较：手动和自动。手动方法包括直接提示 (DP) 和连贯提示 (?)，而自动方法包括 StrategyLLM (?) 和 Self-Discover (?)。

DP 是最简单的方法，其中提示由一组问答对组成，没有任何中间推理步骤。CoT Prompting (?) 通过在提示中加入逐步推理来增强 DP，能够利用 LLM 的推理能力进行多步解决。StrategyLLM (?) 采用更高级的方法，通过使用四个不同的代理商——策略生成器、执行器、优化器和评估器——协作生成、评估和选择有前景的任务策略。类似地，Self-Discover (?) 使用包括选择、采用和实施在内的三个阶段流程，利用 LLM 的能力。更多细节可以在附录 A 中找到。

我们的实验使用 LLaMA3.2-3B (?) 和 LLaMA3-8B (?) 作为主要的 LwLLMs。此外，我们为 DP 和 CoT 提示使用了几乎相同的示例，关键区别在于 CoT 提示包含逐步解决方案。对于 DeBoP，我们为每项任务设置了五个关键步骤计划，并在 PLANNING 和 COLLECTING 阶段应用了温度采样 ($\text{temp} = 0.7$)。在 MCTS 阶段，我们采用了贪婪解码 ($\text{temp} = 0$)。

	PIT	DU	SNK	DQA	LD	HB	MR	Avg.
DP	52 / 44	32 / 42	49 / 55	42 / 30	51 / 45	76 / 70	61 / 52	51.9 / 49.4
CoT Prompting	62 / 69	59 / 47	68 / 56	63 / 49	75 / 66	95 / 71	66 / 51	69.7 / 58.1
StrategyLLM	32 / 27	41 / 28	69 / 51	63 / 53	57 / 38	78 / 49	60 / 22	57.1 / 38.6
Self-Discover	72 / 43	50 / 46	53 / 60	57 / 43	53 / 44	63 / 53	50 / 42	56.9 / 48.7
DeBoP (Ours)	83 / 67	84 / 74	76 / 74	70 / 56	87 / 74	82 / 59	74 / 59	79.4 / 66.1
GPT-3.5	79	85	67	63	80	86	74	76.3

Table 1: 模型准确性比较 (%) 涉及七个任务: PIT、DU、SNK、DQA、LD、HB 和 MR。表格中的大多数单元格以 “x / y” 格式显示结果, 其中 “x” 代表 LLaMA3-8B 的性能, “y” 代表 LLaMA3.2-3B 的性能。GPT-3.5 作为参考模型。每个任务中给定的 LwLLM 的最佳结果以黑体显示。

), 并将最大迭代次数限制为 50。

所有实验均在具有以下规格的计算系统上进行: NVIDIA® GeForce RTX 4090 24GB GDDR6X Blower Edition GPU, 英特尔® Xeon® Gold 6148 处理器 (20 核, 40 线程, 2.4 GHz 基础频率), Supermicro X11DPG-QT 双插槽 LGA3647 服务器主板, 以及 32GB 三星 DDR4-3200 注册 ECC (RDIMM) 内存。GPT3.5 的结果参考自?

为了平衡成本和效率, 我们在一个随机选择的 50 样本子集 (作为 S_{dev}) 和一个 100 样本子集 (作为 S_{test}) 上使用固定的随机种子 (42) 进行所有实验, 这些子集来自原始数据集。

3.2 结果

3.2.1 模型性能比较

表格 1 中的结果详细比较了 DeBoP 对比最近的提示优化方法在七个任务 (PIT, DU, SNK, DQA, LD, HB 和 MR) 中的准确性, 以及作为基线的 GPT-3.5。DeBoP 在大多数任务中始终优于其他方法。值得注意的是, 使用 LLaMA3-8B 的 DeBoP 的平均准确性在测试的所有方法中是最高的, 超过了 GPT-3.5 (76.3%) 以及其他优化策略如 CoT 提示、StrategyLLM 和 Self-Discover。这些结果表明, DeBoP 通过其关键步骤规划和通过 MCTS 优化, 与其他方法相比在多个任务中实现了优越的性能。同样值得注意的是, 虽然 CoT 提示在 HB 任务中取得了最高的准确性, 但这个成功主要归因于对提示进行了精心设计。

3.2.2 效率比较

表 2 给出了 LLaMA3-8B 在七项任务 (PIT, DU, SNK, DQA, LD, HB, MR) 上的效率比较 (以秒为单位), 以及各种提示优化方法的平均推理时间。手动提示优化方法通常比自动方法实现更高的效率。然而, 这些效率优势是以显著的手动工作作为代价的。相比之下, DeBoP

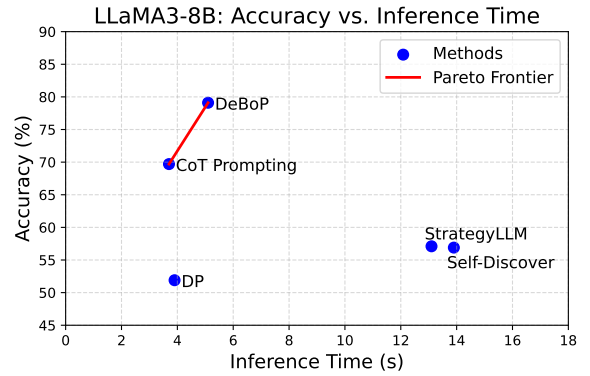


Figure 3: 帕累托前沿分析。

在保持自动化的同时展示了与手动方法相当的效率。例如, DeBoP 在 3.6 秒内完成了 DU 任务, 这接近于 CoT 提示 (3.4 秒) 并显著快于 DP (5.5 秒)。与其他自动化方法相比, DeBoP 将平均推理时间相对于 StrategyLLM 减少了大约 61%, 相对于 Self-Discover 减少了 63%。

图 3 通过一个帕累托前沿分析进一步说明了 DeBoP 的性能, 平衡准确性和推理时间。红线表示帕累托前沿, 在该前沿上, 改善一个指标 (无论是准确性还是推理时间) 都需要牺牲另一个。图中标注的 DeBoP 在保持相对较低推理时间 (约 5s) 的同时, 达到了较高的准确性 (约 80%), 使其成为效率与准确性的最佳权衡点。虽然 CoT Prompting 也位于帕累托前沿上, 但它以略快的推理时间 (3s) 达到了较低的准确性 (70%)。这表明在对准确性敏感的场景中, DeBoP 更为优选。

总之, DeBoP 证明其自身为一种高效的方法, 在精确度与推理时间之间取得了最佳平衡。它不仅在效率上超越了现有的自动方法, 还达到了可与手动提示优化方法相媲美或超越的准确性。

	PIT	DU	SNK	DQA	LD	HB	MR	Avg.
DP	6.5	5.5	2.4	2.0	3.7	4.1	3.2	3.9
CoT Prompting	4.3	3.4	1.4	3.4	3.9	2.4	6.9	3.7
StrategyLLM	5.5	11.3	13.2	11.5	14.9	11.9	23.5	13.1
Self-Discover	14.5	12.8	10.9	8.3	13.7	18.1	18.9	13.9
DeBoP (Ours)	7.3	3.6	4.2	4.4	7.5	4.5	4.4	5.1

Table 2: LLaMA3-8B 在不同方法中的效率比较（以秒为单位）。

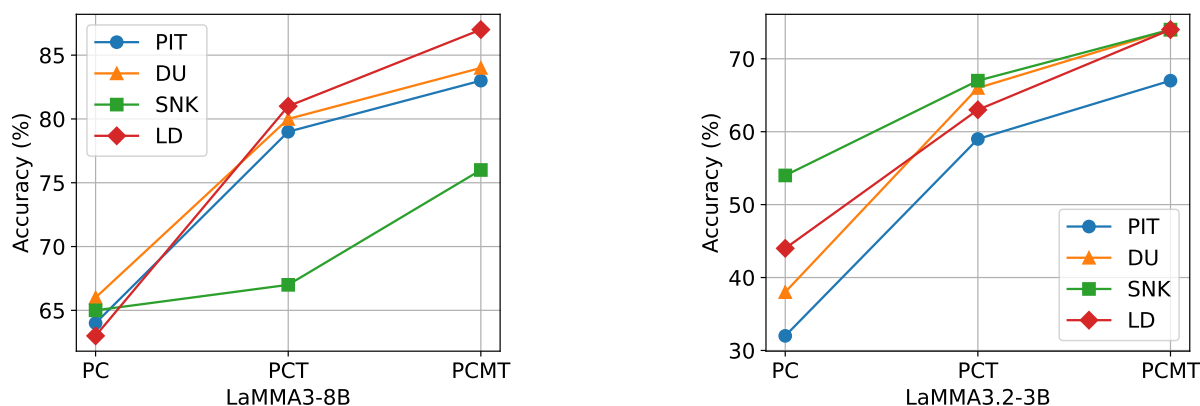


Figure 4: 对单个 DeBoP 阶段的消融研究。P、C、M 和 T 分别是 PLANNING、COLLECTING、MCTS 和 TEACHING 的首字母。由于 C 模块的必要性，我们最终为这些实验设计了 PC、PCT 和 PCMT 设置。

3.2.3 单独 DeBoP 阶段的消融研究

图 4 展示了不同 DeBoP 阶段 (PLANNING、COLLECTING、MCTS 和 TEACHING) 组合对两个 LLaMA 配置模型准确率的影响：LLaMA3-8B (左图) 和 LLaMA3.2-3B (右图)。图中比较了三种设置 (PC (PLANNING+ COLLECTING)、PCT (PLANNING+ COLLECTING+ MCTS+ TEACHING) 和 PCMT (PLANNING+ COLLECTING+ MCTS+ TEACHING)) 在不同任务 (PIT、DU、SNK 和 LD) 上的准确率。结果显示，PCMT 在所有任务中始终优于其他组合，每个任务均达到最高准确率。PCT 在所有任务上相较于 PC 有显著提升，但仍不及 PCMT。这突显了 MCTS 和 TEACHING 在提升 LwLLMs 复杂任务性能中的关键作用。消融研究表明，将这些阶段纳入 DeBoP 框架会显著提高所有任务的性能。

表格 ?? 中呈现的泛化分析为 DeBoP 在 PIT 和 DU 任务中不同模型配置的有效性提供了关键见解。LLaMA3-8B (记为 8B) 和 LLaMA3.2-3B (记为 3B) 在应用 DeBoP 时均显示出显著的准确性提升。具体而言，LLaMA3-8B 在使用自生成示例时达到了 83%/84% (PIT/DU) 的最高准确性。同样，LLaMA3.2-3B 在使用自己的示例时达到了 67%/74%。这些结果表明，与不使用 DeBoP 的基准表现（如第二列中所示的表现，如 8B 为 52%/32%，3B 为 44%/42%

）相比，DeBoP 显著增强了任务性能。

GPT-3.5 一直优于 LLaMA 模型，无论是否使用 DeBoP，这表明其卓越的泛化能力。值得注意的是，当应用 DeBoP 时，GPT-3.5 通过其自身的示范达到了 85%/91% 的最高整体准确率。

该分析证实，DeBoP 在提升模型在各项任务中的性能的同时，保持了良好的跨模型泛化能力。它还强调了利用大型语言模型自身生成的示例来提高较小模型准确性的潜力，为更高效和可扩展的提示优化方法铺平了道路。

3.2.4 单次示范与多次示范设置

Task	Method	Acc. (%)	Time (s)
PIT	DeBoP	83	7.3
PIT	DeBoP-3-shot	64	7.7
DU	DeBoP	84	3.6
DU	DeBoP-3-shot	66	5.4

Table 3: 增加示范次数的影响。

表 3 展示了在两个 BBH 任务——PIT 和 DU 中，使用多个示范与在 DeBoP 框架下的单个优化示范相比的影响。3-shot 上下文是通过基于单次示范性能选择三个正确执行的样本构建的。

表 3 显示, 引入额外的示例导致准确率显著下降 (在 PIT 上降低了 19 %, 在 DU 上降低了 18 %), 同时略微增加了推理延迟。我们将此归因于:

注意力分散: 较长的上下文加剧了 LwLLMs 在长程依赖性方面的限制。

延迟开销: 每增加一个演示将增加解码时间, 对于成本敏感的部署来说, 这不是一个小问题。

鉴于性能和效率的同步下降, 我们得出结论, 在 LwLLM 设置中, DeBoP 的多示例提示不是一个有利的方向。

4 相关工作

4.1 手动提示优化

影响深远的链式思维 (CoT) 提示方法 (?) 激发了众多创新策略来增强大型语言模型的问题解决能力。一种方法是使用编程语言来明确表达推理过程 (???), 而其他方法则通过复杂结构 (如森林或图) 来表示推理 (??)。一些方法将任务分解为更简单的组件, 这有助于模型专注于各个子问题 (????)。此外, 还开发了技术通过自动反馈机制来引入自我纠正 (?????), 并且一些方法结合了多种提示策略以进一步提高性能 (??)。尽管有这些进展, 但大多数这些方法依赖于人工标注的推理过程, 这限制了它们的普适性和灵活性。

4.2 自动提示优化

大型语言模型展示了令人印象深刻的能力, 但挑战仍然存在, 特别是在保持长期连贯性 (?) 和生成多步骤解决方案方面。为了解决这些限制, 出现了两种互补的策略: 自我反思和规划。

自我反思鼓励大型语言模型评估其自身输出, 识别潜在错误, 并反复改进其响应 (??)。这种内省方法不仅提高了生成内容的质量, 而且还使模型能够从自己的错误中学习, 从而随着时间的推移提高性能。

另一方面, 规划涉及将推理组织成一个结构化的步骤或行动序列, 目标是实现一个特定的目标 (??)。受传统规划算法的启发, 这种方法通过提供清晰的、逐步的路径帮助模型在复杂的决策空间中引导自己。这可以通过多种方式实现, 包括直接提示模型进行规划任务 (?), 将指令翻译成可执行的程序 (???), 或者使用基于搜索的技术, 如蒙特卡罗树搜索 (??)。

然而, 绝大多数方法依赖于最先进大型语言模型的元认知能力, 这限制了它们在应用于轻量级大型语言模型时的有效性。在本文中, 我们提出了 DeBoP, 一种新的直接行为优化范式, 可自主提升轻量级大型语言模型在特定任务中

的表现。我们在七个任务上评估了 DeBoP, 覆盖了语言理解和复杂推理。结果显示, DeBoP 持续表现出强大而高效的性能, 揭示了轻量级大型语言模型的新优化方法。此外, 我们深入的分析强调了对轻量级大型语言模型应用 DeBoP 时, 将模型能力与优化策略相契合的重要性。这些发现凸显了 DeBoP 作为一个有效框架的潜力, 能够提高轻量级大型语言模型的效率和准确性, 为未来在轻量级大型语言模型优化上的进一步进步铺平了道路。

5 局限性

虽然 DeBoP 通过无梯度的 MCTS 将复杂的提示优化转化为离散的执行优化, 从而显著提升了 LwLLMs 的性能, 但仍然存在一些限制。特别是, MCTS 扩展过程的广度和效率需要进一步发展以充分挖掘这种方法的潜力。此外, 鉴于 DeBoP 能够有效提高 LwLLMs 的性能, 它可能会被恶意行为者滥用。未来的工作将着重于改进 MCTS 扩展过程并实施安全措施以减轻潜在的滥用。

6

致谢 本研究得到了中国国家重点研发计划 (项目编号: 2023YFC3304902)、杭州创新团队 (项目编号: TD2022011)、国家自然科学基金 (项目编号: 62272413, 62372040, 62202127) 以及桐乡通用人工智能研究院 (项目编号: TAGI2-A-2024-0004) 的资助。

邵军也隶属于中国杭州的浙江省大数据与未来电子商务技术重点实验室。

在 PLANNING 阶段使用的提示在图 5、6 和 7 中有说明。具体来说, 图 5 展示了第一阶段, 而图 6 和 7 对应于第二阶段。

1 COLLECTING 中的提示

在图 8 中说明了在 COLLECTING 中使用的提示。

在 MCTS 中, 我们采用以下六种方法来扩展树节点: 合并、分解、精化、修剪、重采样、简化。每种方法的相应提示如图 9、10、11、12、13 和 14 所示。

A 基线方法的详细信息

七个 BBH 任务使用 DP 和 CoT 提示的提示在以下图中展示: 图 15 为 PIT, 图 16 为 DU, 图 17 为 SNK, 图 18 为 DQA, 图 19 为 LD, 图 20 为 HB, 以及图 21 为 MR。

如果一个示例包含解决方案 (以灰色显示), 则将其分类为 CoT 提示的示例; 否则, 它适用于 DP。

A.1 自我发现

自我发现的再现是指在 <https://github.com/catid/Self-Discover> 处的代码。

A.2 策略 LLM

的复现参考了 <https://github.com/gao-xiao-bai/StrategyLLM> 处的代码。

B 如何从演示中得出一种行为？

行为是基于输入任务和给定示例生成的 LwLLM 的实际响应。我们认为这种行为是由于输入与示例之间的交互决定的，并在 MCTS 和 TEACHING 阶段评估其有效性。

为了用一个简单的例子说明，在 DU 任务中，演示教模型如何解决问题：“今天是 3/5，这是简在 1973 年第二次看到流星雨的时间。24 小时后的日期是 MM/DD/YYYY？”这个任务被分解为以下步骤：

识别已知日期及其与目标日期的关系：在这种情况下，已知日期是 3/5，目标日期是 24 小时后。确定必要的调整：这里必须调整是将已知日期加 1 天。计算准确的日期：在 3/5 的基础上加 1 天，结果是 3/6。最终答案：03/06/1973。

当 LwLLM 遇到新的输入时，例如：“一周前是 2021 年 9 月 1 日。明天的日期是 MM/DD/YYYY 格式的多少？”它从演示中学习并生成自己的响应。它识别出已知日期为 09/01/2021，理解目标日期是一周前的日期之后的一天，进行必要的调整（加 8 天），并计算出准确日期为 09/09/2021。最终，LwLLM 根据演示和学习到的模式，选择正确的答案，即 09/02/2021。

B.1 为什么我们专注于行为层面的优化

指令级优化具有相同的高层目标——增强 LwLLM 推理——但通常假定在轻量级模型中不可靠的元认知能力（例如，抽象提示重写、反思性推理）。在这种约束下，像 STRATEGYLLM 这样的现有方法只能取得有限的增益。而通过无梯度 MCTS 直接优化示范行为，DeBoP 避开了这种依赖，同时仍然提升了推理质量。

我们的研究表明，对于 LwLLMs，精心策划的单样例行为优化比简单地增加演示次数或依赖指令重写在准确性和成本之间提供了更好的平衡。

C 关于蒸馏推理 LwLLMs 中演示影响的补充分析

最近的研究(?)报告称，基于演示的提示可能会损害前沿推理模型（如 DEEPSEEK R1）的性能。为了验证这一说法是否也适用于 LwLLM 领域，我们在两个 BBH 任务（PIT 和 DU）上进行了额外的实验，使用 DeepSeek-R1-LLaMA3-8B 和一个普通的 LLaMA3-8B。我们比较了零样本、三样本 CoT 和 DeBoP（如果可用）。

Task	Model	Method	Acc. (%)	Time (s)
PIT	DeepSeek-R1	3-shot CoT	89	15.4
	DeepSeek-R1	Zero-shot	81	17.9
	LLaMA3-8B	3-shot CoT	62	4.3
	LLaMA3-8B	Zero-shot	52	6.5
	LLaMA3-8B	DeBoP	83	7.3
DU	DeepSeek-R1	3-shot CoT	85	10.5
	DeepSeek-R1	Zero-shot	81	24.5
	LLaMA3-8B	3-shot CoT	59	3.4
	LLaMA3-8B	Zero-shot	32	5.5
	LLaMA3-8B	DeBoP	84	3.6

Table 4: 演示对 PIT 和 DU 的影响。

如表 4 所示，与负面发现？相反，DeepSeek R1 在这两个任务（+8% 和 +4%）上从 3-shot CoT 中明显受益，而 LLaMA3-8B 在配备我们行为优化的 DeBoP 演示时显示了最大的收益。这些结果表明，即使是具有更强上下文能力的现代蒸馏模型，精心设计的演示也能缩小行为空间，引导模型朝向更可靠的推理轨迹——这与 DeBoP 的核心理念相符。

尽管专为 DeepSeek-R1 量身定制的完整 DeBoP 流程尚待未来工作完成，但上述证据强化了行为层面优化超越基础 LwLLMs 的广泛适用性。

D 版权声明

我们根据 MIT 许可证发布我们的代码、数据和其他相关成果。该许可证允许在保留原始许可证通知的情况下使用、修改和分发，无论是学术还是商业用途。通过开放共享这些成果，我们旨在促进研究社区内的透明性、合作和可重复性。

PLANNING Prompt

[System Message]

There are a series of tasks of the same type. Please try to understand these tasks and give a general and instructive solution.

Your output should include:

- Include all necessary subtasks.
- All subtasks are easy to solve.
- Subtasks are in an appropriate order.
- Do not include specific information in the examples.
- Make sure the strategy is general and concise.
- Your output should finally start with "Therefore, we can get the general and instructive solution:", followed by your solution.

[User Message]

Please try to understand these tasks about mathematical reasoning and give a general and instructive solution. The examples of the problems is as follows:

- The roots of the equation $2x^2 - mx + n = 0$ sum to 6 and multiply to 10. What is the value of $m + n$?
- Solve for n : $0.03n + 0.08(20 + n) = 12.6$.
- Let C be the circle with equation $x^2 - 6y - 3 = -y^2 - 4x$. If (a, b) is the center of C and r is its radius, what is the value of $a + b + r$?

[Assistant Message]

I'm piecing together a plan to tackle three math problems, detailing the sequence of subtasks for each one, while avoiding specifics in the framework.

We have a set of algebraic and geometric problems, such as finding unknown values in equations or determining the center and radius of a circle. These problems can typically be approached by identifying the equation type (linear, quadratic, or geometric), applying known properties or standard forms, and then systematically solving for the unknowns.

Therefore, we can get the general and instructive solution:

- Identify the mathematical form of each problem (e.g., quadratic equation, linear equation, circle equation) and note which unknowns or parameters need to be found.
- Collect all given conditions (such as relationships involving the roots, values that must satisfy an equation, or geometric properties) and translate them into algebraic expressions.
- Combine or rearrange these expressions into a more standard form (for instance, factoring a quadratic, grouping like terms, completing the square) so the unknowns are clearly isolated.
- Solve the resulting equations step by step, ensuring each operation is straightforward (e.g., simple algebraic manipulations or known formulas).
- Use any final instructions (like summing certain values or substituting back into an expression) to obtain the required result.

[User Message]

Please try to understand these tasks about

{task definition},

and give a general and instructive solution. The examples of the problems is as follows:

{questions}

Figure 5: PLANNING 分两个阶段提示 (阶段 1)。

PLANNING Prompt

[User Message]

To solve this kind of problems about math.

We need to create a JSON structure that outlines a step-by-step solution based on the provided information. Each key in the JSON should represent a critical step required to solve the task, and the corresponding values should be empty strings.

Note

- Implement a reasoning structure for solvers to follow step-by-step and arrive at correct answers.
- Steps should be accurate and concise, and there should be no redundancy.

Please understand the provided guidelines:

Guidelines

- Identify the mathematical form of each problem (e.g., quadratic equation, linear equation, circle equation) and note which unknowns or parameters need to be found.
- Collect all given conditions (such as relationships involving the roots, values that must satisfy an equation, or geometric properties) and translate them into algebraic expressions.
- Combine or rearrange these expressions into a more standard form (for instance, factoring a quadratic, grouping like terms, completing the square) so the unknowns are clearly isolated.
- Solve the resulting equations step by step, ensuring each operation is straightforward (e.g., simple algebraic manipulations or known formulas).
- Use any final instructions (like summing certain values or substituting back into an expression) to obtain the required result.

Now please learn these guideline information and generate the corresponding JSON structure. The JSON structure should:

Note

- Include all necessary steps.
- Make all steps easy to solve.
- Have no redundant steps, that is, don't repeat the steps when there is already an answer.
- End with ****The final answer**** as the end of the json structure.

Figure 6: PLANNING 两阶段提示（第二阶段，第一部分）

它是

PLANNING Prompt

[Assistant Message]

Quick Summary of Guidelines

- Determine the parameters to be solved.
- Collect all the given conditions.
- Clarify the unknowns.
- Solve the equation step by step, ensuring that each step is clear, simple, and straightforward.
- Calculate the final answer.

Therefore, we can derive the step-by-step solution in the following JSON structure:

```
```json
{
 "Determine the parameters to be solved": "",
 "Collect all the given conditions": "",
 "Clarify the unknowns": "",
 ...
 "The final answer": ""
}
```
```

[User Message]

To solve this kind of problem about { task_definition }

We need to create a JSON structure that outlines a step-by-step solution based on the provided information. Each key in the JSON should represent a critical step required to solve the task, and the corresponding values should be empty strings.

Notes

- Implement a reasoning structure for solvers to follow step-by-step and arrive at correct answers.
- Steps should be accurate and concise, and there should be no redundancy.

Please review and understand the provided guidelines:

Guidelines

{guidelines}

Now please learn these guideline information and generate the corresponding JSON structure. The JSON structure should:

Notes

- Include all necessary steps.
- Ensure that all steps are easy to follow and solve.
- Have no redundant steps, meaning that steps should not be repeated when an answer already exists.
- End with The final answer as the last key in the JSON structure.

Figure 7: PLANNING 两阶段提示（阶段 2，第 2 部分）

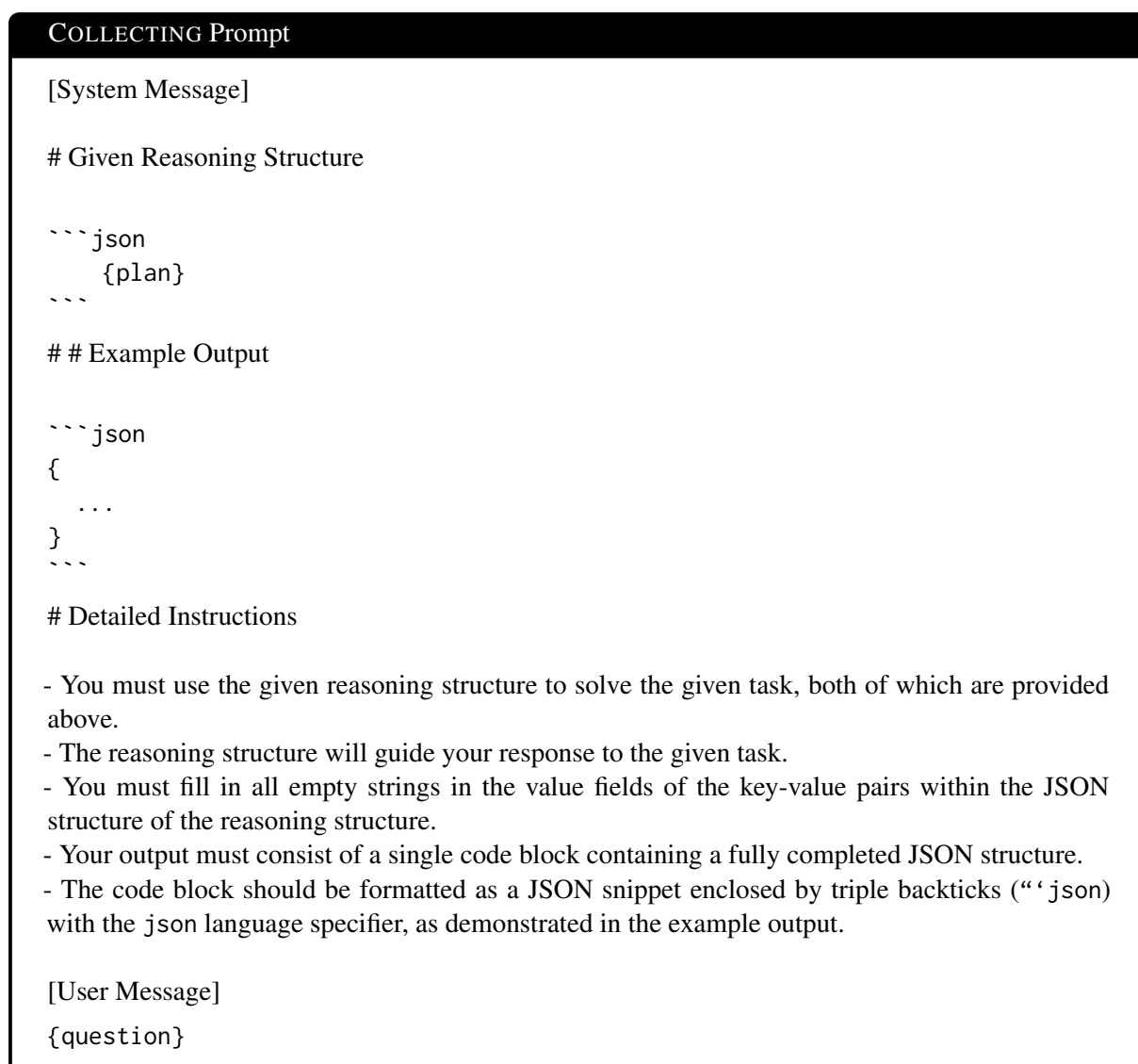


Figure 8: COLLECTING 相位提示。

MCTS Consolidation Prompt

For the given task:

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you determine whether any steps can be merged to make the reasoning process more logical? If so, please explain why and then present the optimized result in Markdown JSON format.

Notes

- If no steps can be merged without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 9: MCTS 合并提示。

MCTS Decomposition Prompt

For the given task:

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you determine whether any steps can be easily split to improve the clarity and accuracy of the reasoning process? If so, please explain why and then present the optimized result in Markdown JSON format.

Notes

- If no steps can be split without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 10: MCTS 分解提示。

MCTS Elaboration Prompt

For the given task:

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you provide a more detailed and expanded version of the reasoning flow (including both keys and values) to ensure that no critical aspects of the problem-solving process are overlooked? If so, please explain why and then present the optimized result in Markdown JSON format.

Notes

- If no steps can be expanded without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 11: MCTS 详细提示。

MCTS Pruning Prompt

For the given task:

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you determine whether one step (the least consequential one) can be removed without harming the overall reasoning, thereby making the process more concise and logical? If so, please explain why, and then present the optimized result in Markdown JSON format.

Notes

- If no step can be removed without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 12: MCTS 修剪提示。

MCTS Resampling Prompt

For the given task

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you determine whether any steps can be freely optimized to enhance the logical consistency and efficiency of the reasoning process? If so, please explain why and then present the optimized result in Markdown JSON format.

Notes

- If no steps can be freely optimized without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 13: MCTS 重采样提示。

MCTS Simplification Prompt

For the given task:

{task definition}

{question}

we already have the following JSON structured solution:

{demonstration}

Could you simplify and refine the reasoning flow (including both keys and values) to improve conciseness, coherence, and logical structure? If so, please explain why and then present the optimized result in Markdown JSON format.

Notes

- If no steps can be abbreviated without compromising the reasoning, state this explicitly and provide the original JSON structure.
- In this JSON structure, each key represents a piece of guidance for the solution, and each value details the specific reasoning process according to that guidance.

Figure 14: MCTS 简化提示。

DP & CoT Prompt of Penguin in a Table

Question: Here is a table where the first line is a header and each subsequent line is a penguin:
name, age, height (cm), weight (kg)

Louis, 7, 50, 11

Bernard, 5, 80, 13

Vincent, 9, 60, 11

Gwen, 8, 70, 15

For example: the age of Louis is 7, the weight of Gwen is 15 kg, the height of Bernard is 80 cm.

We now add a penguin to the table:

James, 12, 90, 12

How many penguins are less than 8 years old?

Options: (A) 1 (B) 2 (C) 3 (D) 4 (E) 5

解答：这个问题关注的是年龄。我们知道以下信息：Louis 7 岁，Bernard 5 岁，Vincent 9 岁，Gwen 8 岁。现在，我们将 James 添加到这个表中：James 12 岁。年龄小于 8 岁的企鹅是 Louis 和 Bernard。有 2 只企鹅年龄小于 8 岁。因此，答案是 (B)。

Answer: (B)

Question: Here is a table where the first line is a header and each subsequent line is a penguin:
name, age, height (cm), weight (kg)

Louis, 7, 50, 11

Bernard, 5, 80, 13

Vincent, 9, 60, 11

Gwen, 8, 70, 15

For example: the age of Louis is 7, the weight of Gwen is 15 kg, the height of Bernard is 80 cm.

Which is the youngest penguin?

Options: (A) Louis (B) Bernard (C) Vincent (D) Gwen (E) James

解答：这个问题关注的是年龄。我们知道以下信息：路易斯 7 岁，伯纳德 5 岁，文森特 9 岁，格温 8 岁。根据表格，伯纳德（5 岁）是他们当中最年轻的。最年轻的企鹅是伯纳德。所以答案是 (B)。

Answer: (B)

Question: Here is a table where the first line is a header and each subsequent line is a penguin:
name, age, height (cm), weight (kg)

Louis, 7, 50, 11

Bernard, 5, 80, 13

Vincent, 9, 60, 11

Gwen, 8, 70, 15

For example: the age of Louis is 7, the weight of Gwen is 15 kg, the height of Bernard is 80 cm.

What is the name of the second penguin sorted by alphabetic order?

Options: (A) Louis (B) Bernard (C) Vincent (D) Gwen (E) James

解决方案：这个问题关注于名字。我们知道以下几点：表中的企鹅名字是 Louis、Bernard、Vincent 和 Gwen。当我们按字母顺序排列它们的名字时，我们得到 Bernard、Gwen、Louis、Vincent。按字母顺序排列的第二个企鹅的名字是 Gwen。按字母顺序排列的第二个企鹅的名字是 Gwen。所以答案是 (D)。

Answer: (D)

Question:

{question}

Figure 15: 企鹅表格中的 DP & CoT 提示

DP & CoT Prompt of Date Understanding

Question: Today is Christmas Eve of 1937. What is the date 10 days ago in MM/DD/YYYY?

Options: (A) 12/14/2026 (B) 12/14/1950 (C) 12/14/2007 (D) 12/14/1937 (E) 07/14/1938 (F) 12/14/1988

解答：如果今天是 1937 年的平安夜，那么今天的日期是 1937 年 12 月 24 日。10 天前的日期是 1937 年 12 月 14 日，即 12/14/1937。所以答案是 (D)。

Answer: (D)

Question: Tomorrow is 11/12/2019. What is the date one year ago from today in MM/DD/YYYY?

Options: (A) 09/04/2018 (B) 11/11/2018 (C) 08/25/2018 (D) 11/02/2018 (E) 11/04/2018

解答：如果明天是 2019 年 11 月 12 日，那么今天是 2019 年 11 月 11 日。今天的一年前是 2018 年 11 月 11 日。所以答案是 (B)。

Answer: (B)

Question: Jane and John married on Jan 2, 1958. It is their 5-year anniversary today. What is the date tomorrow in MM/DD/YYYY?

Options: (A) 01/11/1961 (B) 01/03/1963 (C) 01/18/1961 (D) 10/14/1960 (E) 01/03/1982 (F) 12/03/1960

解答：如果简和约翰在 1958 年 1 月 2 日结婚，并且今天是他们的结婚 5 周年纪念日，那么今天的日期是 1963 年 1 月 2 日。明天的日期是 1963 年 1 月 3 日，即 01/03/1963。所以答案是 (B)。

Answer: (B)

Question:

{question}

Figure 16: DP & CoT 提示关于日期理解。

DP & CoT Prompt of Snark

Question: Which statement is sarcastic?

Options: (A) Yes, because having interests and actively researching them is a huge waste (B) Yes, because having interests and actively researching them is a huge deal

解答：如果我们看 (A)，它指出有兴趣并积极研究它们是巨大的浪费，暗示这是无用的努力。然而，我们知道有兴趣并积极研究它们通常并不是浪费，而是对个人有益。这种对比在 (A) 中出现，表明它带有一丝讽刺和嘲讽。

如果我们看 (B)，它说有兴趣并积极研究它们是件大事，暗示这是个重要和有影响力的努力。这可以说是一个中立且正确的陈述。

综上，讽刺的选项是 (A)。所以答案是 (A)。

Answer: (A)

Question: Which statement is sarcastic?

Options: (A) No one is going to disagree with you on this. Avoiding ad hominem attacks really help your case (B) No one is going to disagree with you on this. Ad hominem attacks really help your case

解决方案：如果我们看看 (A)，它说避免人身攻击确实有助于你的论点，暗示人身攻击是有害的，因为人身攻击针对的是人而不是观点，所以确实避免它们通常是有用的和有帮助的。因此，(A) 是一个中立（有效且可接受）的陈述。

如果我们看看 (B)，它说人身攻击真的有助于你的论点，暗示人身攻击是一件积极的事情。然而，我们之前已经说明人身攻击通常没有用或不具建设性。因此，这句陈述中的发言者似乎是在反着说他们的意思；所以，(B) 中似乎带有讽刺和嘲弄的意味。

综上所述，讽刺的选项是 (B)。所以答案是 (B)。

Answer: (B)

Question: Which statement is sarcastic?

Options: (A) Consistency in the league's punishments? What do you think this is supposed to be, politics? (B) Consistency in the league's punishments? What do you think this is supposed to be, moral?

解决方案：如果我们看 (A)，它将联盟的处罚一致性比作政治中的一致性。因为政治或政治事务通常不被认为是一致或可靠的，这个句子显得讽刺。

如果我们看 (B)，它将联盟的处罚一致性比作道德中的一致性。在道德、伦理或法律背景下讨论联盟的处罚一致性是有道理的，并没有显得在任何事情上讽刺。

综合上述，讽刺的选项是 (A)。所以答案是 (A)。

Answer: (A)

Question:

{question}

Figure 17: 狡辩的 DP & CoT 提示。

DP & CoT Prompt of Disambiguation QA

Question: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The chief told the counselor that they took the day off.

Options: (A) The chief took the day off (B) The counselor took the day off (C) Ambiguous

解决方案: 在这里我们需要确定代词“they”可能指的是谁。“they”有两个可能的指代对象，即首领和顾问。动词“told”可能能够帮助我们确定哪个更有可能（如果有的话）。设 X 为首领，Y 为顾问。那么这句话的形式为“X 告诉 Y，(X 或 Y) 做了某事。”

让我们首先考虑 Y：“X 告诉 Y，Y 做了某事。”这种情况下没有什么意义，因为 Y 已经知道 Y 做了某事，因为这对他们自己来说是信息。

现在，考虑 X：“X 告诉 Y，X 做了某事。”这就有意义了，因为 X 将分享一些关于他们自己的信息，而 Y 之前可能并不知道这些信息。

因为在这种情况下，X 是首领，Y 是顾问，所以答案应该是首领。因此，答案是 (A)。

Answer: (A)

Question: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: The manager sent a message to the secretary, but he didn't reply yet.

Options: (A) The secretary didn't reply yet (B) The manager didn't reply yet (C) Ambiguous

解决方案: 在这里我们需要确定代词“he”可能指的是谁。“he”有两个可能的指代对象，即经理和秘书。动词“sent”和“reply”可能会帮助我们判断哪个更有可能（如果有的话）。设 X 为经理，Y 为秘书。句子形式则为“X 给 Y 发了一条消息，但 (X 或 Y) 还没有回复。”

首先考虑 Y：“X 给 Y 发了一条消息，但 Y 还没有回复。”这种情况是合理的，因为句子中存在隐含的因果关系。Y 是消息的接收者，但 Y 还没有回复 X。

现在考虑 X：“X 给 Y 发了一条消息，但 X 还没有回复。”这种情况是不合理的，因为 X 是消息的初发者，所以现在轮到 Y 回复 X。

因为在这个上下文中，X 是经理而 Y 是秘书，答案应该是秘书。所以答案是 (A)。

Answer: (A)

Question: In the following sentences, explain the antecedent of the pronoun (which thing the pronoun refers to), or state that it is ambiguous.

Sentence: Bailey will plan to meet the director at his office.

Options: (A) It will be Bailey's office (B) It will be the director's office (C) Ambiguous

解决方案: 在这里我们需要确定代词“他的”可能指的是谁。“他的”有两个可能的指代对象，即贝利和主任。动词短语“计划见面”可能可以帮助我们确定哪个更有可能（如果有的话）。设 X 为贝利，Y 为主任。这个句子就呈现为“X 计划在 (X 或 Y) 的办公室与 Y 见面。”

让我们首先考虑 Y：“X 计划在 Y 的办公室与 Y 见面。”这种情况是有道理的，因为 X 可能想在 Y 的办公室与 Y 会面。

现在，考虑 X：“X 计划在 X 的办公室与 Y 见面。”这种情况也有道理，因为 X 可能想在自己的办公室与 Y 会面。

因为 X 和 Y 同时都是可能的，所以我们得出结论，代词的先行词是模糊的。因此答案是 (C)。

Answer: (C)

Question:

{question}

Figure 18: DP & CoT 消歧问答提示。

DP & CoT Prompt of Logical Deduction

Question: The following paragraphs each describe a set of three objects arranged in a fixed order. The statements are logically consistent within each paragraph.

In a golf tournament, there were three golfers: Amy, Eli, and Eve. Eve finished above Amy. Eli finished below Amy.

Options: (A) Amy finished last (B) Eli finished last (C) Eve finished last

解答: (1) 伊芙成绩高于艾米: “(上)? 伊芙? 艾米? (下)”。

(2) 伊莱成绩低于艾米: “(上)? 艾米? 伊莱? (下)”。

(3) 结合 (1) 和 (2), 我们得到以下顺序: “(上) 伊芙 艾米 伊莱 (下)”。

根据这个顺序, 最后一名 (在这个列表最后的那个人) 是伊莱。

伊莱最后一名。因此答案是 (B)。

Answer: (B)

Question: The following paragraphs each describe a set of three objects arranged in a fixed order. The statements are logically consistent within each paragraph.

On a shelf, there are three books: a white book, a green book, and an orange book. The green book is to the right of the white book. The orange book is the rightmost.

Options: (A) The white book is the leftmost (B) The green book is the leftmost (C) The orange book is the leftmost

解决方案: (1) 绿色书在白色书的右边: “(左)? 白色? 绿色? (右)”。

(2) 橙色书是最右边的: “(左)? 白色? 绿色 橙色 (右)”。

(3) 结合 (1) 和 (2), 我们得到以下顺序: “(左) 白色 绿色 橙色 (右)”。

根据这个顺序, 最左边的书是白色书。

白色书是最左边的。所以答案是 (A)。

Answer: (A)

Question: The following paragraphs each describe a set of three objects arranged in a fixed order. The statements are logically consistent within each paragraph.

On a shelf, there are three books: a red book, a gray book, and a white book. The white book is to the left of the gray book. The red book is the second from the left.

Options: (A) The red book is the leftmost (B) The gray book is the leftmost (C) The white book is the leftmost

解答: (1) 白色书在灰色书的左边: “(左)? 白色? 灰色? (右)”。(2) 红色书从左向右数是第二本: “(左)? 白色 红色 灰色? (右)”。(3) 结合 (1) 和 (2), 我们得到以下排序: “(左) 白色 红色 灰色 (右)”。根据这个排序, 最左边的书是白色书。白色书在最左边。所以答案是 (C)。

Answer: (C)

Question:

{question}

Figure 19: DP & CoT 逻辑推理提示。

DP & CoT Prompt of Hyperbaton

Question: Which sentence has the correct adjective order?

Options: (A) rubber terrible ship (B) terrible rubber ship

解决方案：当名词前面有一个以上的形容词时，这些形容词需要遵循以下顺序：[1. 观点] [2. 尺寸] [3. 年龄] [4. 形状] [5. 颜色] [6. 产地] [7. 材料] [8. 目的] 名词。

选项 (A): “rubber terrible ship”。(1) “rubber” 归入材料类别。(2) “terrible” 归入观点类别。选项 (A) 的形容词顺序是：[7. 材料] [1. 观点] (或用数字表示为 7 1)。由于 $7 < 1$ 是不正确的，所以 (A) 的顺序不正确。

选项 (B): “terrible rubber ship”。选项 (B) 的形容词顺序是：[1. 观点] [7. 材料] (或用数字表示为 1 7)。由于 $1 < 7$ 是正确的，所以 (B) 的顺序是正确的。所以答案是 (B)。

Answer: (B)

Question: Which sentence has the correct adjective order?

Options: (A) repulsive small Brazilian exercise ship (B) Brazilian repulsive exercise small ship

解决方案：当在名词之前有多个形容词时，这些形容词需要遵循以下的顺序：[1. 看法] [2. 大小] [3. 年龄] [4. 形状] [5. 颜色] [6. 来源] [7. 材质] [8. 目的] 名词。

选项 (A): “repulsive small Brazilian exercise ship”。(1) “repulsive” 属于看法类别。(2) “small” 属于大小类别。(3) “Brazilian” 属于来源类别。(4) “exercise” 属于目的类别。选项 (A) 的形容词顺序如下：[1. 看法] [2. 大小] [6. 来源] [8. 目的] (用数字表示，则为 1 2 6 8)。由于 $1 < 2 < 6 < 8$ 是正确的，所以选项 (A) 的顺序正确。

选项 (B): “Brazilian repulsive exercise small ship”。选项 (B) 的形容词顺序如下：[6. 来源] [1. 看法] [8. 目的] [2. 大小] (用数字表示，则为 6 1 8 2)。由于 $6 < 1 < 8 < 2$ 是不正确的，所以选项 (B) 的顺序不正确。因此，答案是 (A)。

Answer: (A)

Question: Which sentence has the correct adjective order?

Options: (A) blue gold wonderful square shoe (B) wonderful square blue gold shoe

解决方案：当一个名词前有多个形容词时，形容词需要遵循以下顺序：[1. 看法] [2. 大小] [3. 年代] [4. 形状] [5. 颜色] [6. 来源] [7. 材料] [8. 目的] 名词。选项 (A): “蓝色金色美妙正方形鞋”。(1) “蓝色” 属于颜色类别。(2) “金色” 属于材料类别。(3) “美妙” 属于看法类别。(4) “正方形” 属于形状类别。选项 (A) 的形容词顺序是 [5. 颜色] [7. 材料] [1. 看法] [4. 形状] (或以数字表示为 5 7 1 4)。由于 $5 < 7 < 1 < 4$ 是不正确的，所以 (A) 的顺序不正确。选项 (B): “美妙正方形蓝色金色鞋”。选项 (B) 的形容词顺序是：[1. 看法] [4. 形状] [5. 颜色] [7. 材料] (或以数字表示为 1 4 5 7)。因为 $1 < 4 < 5 < 7$ 是正确的，所以 (B) 的顺序是正确的。因此答案是 (B)。

Answer: (B)

Question:

{question}

Figure 20: DP & 倒装法提示词链

DP & CoT Prompt of Movie Recommendation

Question: Find a movie similar to Star Wars Episode IV - A New Hope, Indiana Jones and the Last Crusade, Star Wars Episode V - The Empire Strikes Back, The Big Lebowski.

Options: (A) Tetsuo (B) the Ironman (C) The Princess Bride (D) The Barkley Marathons The Race That Eats Its Young (E) Bug

解决方案: - 《星球大战 IV: 新希望》(动作, 冒险, 幻想; 1977 年)

- 《夺宝奇兵之圣战奇兵》(动作, 冒险; 1989 年)

- 《星球大战 V: 帝国反击战》(动作, 冒险, 幻想; 1980 年)

- 《大角色》(动作, 戏剧, 喜剧; 1998 年)

这些都是 2000 年以前制作的著名经典美国电影。在所有选项中, 唯一与这些电影类似的似乎是《公主新娘》(1987 年)。所以答案是 (C)。

Answer: (C)

Question: Find a movie similar to Twister, The Silence of the Lambs, Independence Day, Braveheart.

Options: (A) They Shoot Horses (B) Don't They (C) Forrest Gump (D) The Salton Sea (E) Extreme Days

解决方案: - 《龙卷风》(动作, 冒险, 惊悚; 1996)

- 《沉默的羔羊》(犯罪, 剧情, 惊悚; 1991)

- 《独立日》(动作, 科幻, 剧情; 1996)

- 《勇敢的心》(传记, 剧情, 史诗; 1995)

这些都是 1990 年代左右制作的著名好莱坞电影。在所有选项中, 唯一一部与这些电影相似的似乎是《阿甘正传》(喜剧, 剧情, 爱情; 1994)。所以答案是 (C)。

Answer: (C)

Question: Find a movie similar to Minority Report, Total Recall, Inside Out, Forrest Gump.

Options: (A) Phenomena (B) Lilting (C) Catwoman (D) Edge of Tomorrow

解决方案: - Minority Report (动作, 犯罪, 悬疑; 2002)

- Total Recall (动作, 冒险, 科幻; 2012)

- Inside Out (动画, 家庭, 喜剧; 2015)

- Forrest Gump (喜剧, 剧情, 浪漫; 1994)

这些都是过去几十年制作的著名电影。在所有选项中, 唯一看起来类似这些电影的是 Edge of Tomorrow (动作, 冒险, 犯罪, 悬疑; 2014), 因为它也是一部科幻电影, 并由汤姆·克鲁斯主演。所以答案是 (D)。

Answer: (D)

Question:

{question}

Figure 21: 电影推荐的 DP & CoT 提示。