EvoRoute：经验驱动的自主路由大语言模型智能体系统

Guibin Zhang[1], Haiyang Yu[2],KaimingYang[1],Bingli Wu[2],
Fei Huang[2], Yongbin Li[2†], ShuichengYan[1†][†]新加坡国立大学,
[2]通义实验室, [†]通讯作者**主要联系人：**guibinz@outlook.com

arXiv:2601.02695v1 [cs.CL] 6 Jan 2026

摘要

由大语言模型（LLM）、工具和记忆模块组成的协同集成体驱动的复杂智能体人工智能系统，在复杂的多轮任务中展现出卓越能力。然而，这一成功背后却伴随着高昂的经济成本和严重的延迟，凸显出一个至关重要却尚未得到充分探索的权衡。我们将这一挑战形式化为**智能体系统三难困境**：实现最先进性能、尽量降低货币成本以及确保快速完成任务三者之间固有的张力。为破解这一三难困境，我们提出 `EvoRoute`，这是一种自进化模型路由范式，超越了静态、预定义的模型分配方式。借助不断扩展的过往经验知识库，`EvoRoute` 会在每一步动态选择帕累托最优的大语言模型骨干模型，在平衡准确率、效率和资源使用的同时，持续根据环境反馈优化自身的选择策略。在 GAIA 和 BrowseComp+ 等具有挑战性的智能体基准测试中进行的实验表明，将 `EvoRoute` 集成到现成的智能体系统后，不仅能够维持或提升系统性能，还能将执行成本降低多达 80%，并将延迟缩短逾 70%。

1 引言

随着大语言模型（LLM）驱动的智能体持续展现出日益先进的认知能力，涵盖感知（Driess 等人，2023年；Zheng 等人，2023年；Wei 等人，2024年）、规划（Zhu 等人，2024年；Erdogan 等人，2025年）、推理（Putta 等人，2024年；Masterman 等人，2024年）和动作（Li 等人，2024年；Yanget 等人，2024年），由 LLM 驱动的智能体人工智能系统已在一系列高度复杂的多轮任务中取得了显著性能，包括机器学习工程（Chan 等人，2025年）、多跳信息搜索（Mialon 等人，2023年）和报告

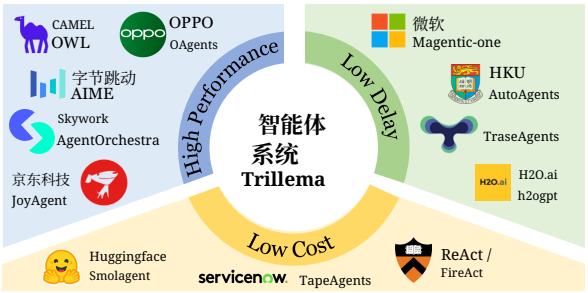


图1：智能体系统三难困境。现有的（深度研究）智能体系统各有所长，但没有任何一种系统能够同时满足这三项特征。

生成（Chen 等人，2025a），以及 GitHub问题修复（Jimenez 等人，2024年）。

超越单智能体框架，这些智能体系统的特点是多个异构模型（可能由不同机构开发，或针对不同领域进行专业化）协同运作，并结合复杂的工具调用工作流和记忆模块（Chen 等人，2025b；Chaudhry 等人，2025）。近期的代表性实例包括 Skywork AI 推出的 AgentOrchestra（Zhang 等人，2025c），该系统在 GAIA（70%+ （Mialon 等人，2023年）上取得了开源领域最先进（SOTA）的成绩；另一个实例是 ML-master（Liu 等人，2025年），其在 MLE-Bench（29。3% （Chan 等人，2025年）上取得了这一成绩。两者都使用了前沿模型（例如，OpenAI GPT-4.1（OpenAI）、Gemini- 2.5-Pro（Comanici 等人，2025年）。

尽管这类系统展现出令人瞩目的性能，但一些尚未充分探讨的评估维度也让人们对其实际部署产生了担忧。实证研究发现，某些通用智能体系统（例如 OWL- Workforce（Hu 等人，2025年）在 2025年6月于 GAIA 上取得开源领域的最先进成果，但其执行成本高昂，平均每项任务最高可达 3 美元。有时，即便是相对简单的查询，也可能需要执行长达 40分钟，最终却仍能得出正确答案。同样，

R&D Agent（Yang 等人，2025年）未能在 24 小时窗口内完成 MLE-Bench 规定的 75 项数据科学任务。总之，如图♣ **效率**(即任务完成速度)和♠ **成本**(即大语言模型调用与工具使用的累计开销)所示，复合智能体系统的这些指标往往不及其令人期待的♦ **性能**指标，见图1。

为明确形式化这一挑战，我们从经济学中的经典“不可能三角”理论（Aizenman 等人，2013年）和区块链领域的相关理论（Koutsoupakis，2021年）中汲取灵感。这些理论认为，系统无法同时实现全部三项理想属性（例如，区块链场景中的可扩展性、安全性和去中心化）。与此类似，我们提出**智能体系统三难困境**，以刻画复杂智能体系统中固有的权衡关系。具体而言，当这类系统作为通用人工智能助手运行时，往往面临三项核心目标之间的内在冲突：**性能**(即任务成功率和准确率）、**效率**(即完成任务所需的时间或步骤数）以及**成本**(即所消耗的计算资源和货币资源）。

真的无法设计出一种方法来克服这一三难困境，并实现三重最优的智能体系统吗？我们认为答案是肯定的。一个尤其有前景的方向是模型路由（Hu 等人，2024a），即，根据任务查询，从候选模型池中智能地选择最适合的大语言模型智能体。然而，现有的模型路由范式仍不足以打破这一三难困境。第一种是➤ **单模型路由**，即，分配一个模型处理整个任务（Feng 等人，2024年；Chen 等人，2024年），这种方式适用于原子任务，但难以应对复杂的智能体工作流；在这类工作流中，网页浏览、编程和摘要生成等子任务的需要不同的模型能力（Chen 等人，2025b；Frick 等人，2025年）。第二种范式➤ **多智能体路由**，在（Yue 等人，2025年；Chen 等人，2025b）中得到了形式化定义，它为多智能体系统中的不同智能体角色选择不同的模型。尽管这种方法与智能体系统的结构更加契合，但迄今为止，它仍局限于相对简单的单步推理场景（例如数学推理和代码生成），并且高度依赖大规模轨迹来建模智能体行为，因此不适用于动态的多轮智能体场景。通过轨迹对智能体行为建模，因此不适用于动态、多轮的智能体场景。

为应对这一三难困境，我们提出EvoRoute，这是一种三重最优、自我演化的模型路由范式，适用于通用智能体系统。具体而言，EvoRoute 不再局限于单一模型或固定的多智能体配置，而是在复杂工作流中以单个子任务为粒度运作。在执行每个步骤之前，它会动态选择最合适的大语言模型，具体方法如下：❶ **检索**：通过多方面检索，从演化知识库中找出历史上类似的子任务执行记录；❷ **筛选**：提炼出一个帕累托最优的候选模型集，即在成本、效率和性能等维度上均不被其他模型支配的模型；❸ **选择**：利用轻量级决策模型，根据丰富且具备上下文感知能力的统计证据做出最终选择。

EvoRoute 的“自我演进”特性通过双阶段运行设计得以实现：在优化阶段，系统进行基于树的探索，为给定任务采样多条轨迹，以主动构建并丰富其关于模型行为的知识库；相应地，在推理阶段，系统利用积累的经验，沿着单一且经过优化的执行路径推进，从而实现快速且具成本效益的任务完成。这种自适应、由经验驱动的方法使 EvoRoute能够持续完善其路由策略，逐步突破智能体系统三难困境的限制。

我们的贡献可总结如下：

❶ **问题形式化**。 我们将当前智能体系统中的关键瓶颈形式化为智能体系统三难困境，即性能、成本和效率之间固有的权衡，并提出一种可行的方法，通过实证缓解这一问题。

❷ **技术方案**。 我们提出 EvoRoute，一种新颖的自演化路由范式，通过细粒度模型选择打破这一三难困境；该范式结合多维检索与帕累托最优选择，实现资源感知模型路由。

❸ **实证验证**。 我们在包括 GAIA 和 BrowseComp+ 在内的五个具有挑战性的基准上开展了广泛实验，结果表明，EvoRoute 的表现最高可超越原生智能体系统 10.3%，同时仅产生 $\sim$ 20% 的成本，并实现近 $3\times$ 倍的执行速度。

2 相关工作

智能体人工智能系统 当代多智能体系统可根据其自动化程度大致分为三类：■ 手工构建型，即整个系统配置（例如，大语言模型骨干、提示策略和通信协议）均由人工指定，代表系统包括 AutoGen（Wu 等人，2023年）、AutoGPT（Richards 和等人，2023年）、CAMEL（Li 等人，2023年）和 ChatDev （ Qian 等人，2023 年）；■ 部分自动化型，即实现特定系统组件的自动化：例如，AutoAgent（Chen 等人，2023年）、LLMSelector（Chen 等人，2025b）和 MasRouter（Yue 等人，2025年）实现了智能体角色分配的自动化；DSPy（Khattab 等人，2023年）和 TextGrad（Yuksekgonul 等人，2024年）优化提示设计；GPTSwarm（Zhuge 等人，2024年）和 G-Designer（Zhang 等人，2024a）自适应地构建智能体间拓扑结构；■ 全自动化型，即系统内的所有模块均由系统自主设计和演化（Hu 等人，2024b；Zhang 等人，2024b、2025b；Wu 等人，2025年；Nie 等人，2025年；Gao 等人，2025年；Zhang 等人，2025a）。

大语言模型与智能体路由 通过智能路由利用多个模型，已成为现代人工智能和机器学习中的核心范式，旨在发挥模型能力的互补优势，以提升任务性能，同时有望优化计算成本（Srivatsa 等人，2024）。基于路由器的方法是本文的主要关注点，这类方法会学习将每个查询分配给最合适的模型（Hu 等人，2024a）。经典方法可分为：（I）基于神经网络的路由器，根据性能或成本信号进行训练，包括 LLM-Blender，它通过成对比较汇总由此选出的顶尖 k 个模型的输出（Jiang 等人，2023）；ZOOTER，它通过奖励引导、基于标签的标签增强来改进路由器训练（Lu 等人，2023）；以及其他方法（Ong 等人，2024；Feng 等人，2024；Zhang 等人，2025d）；（II）基于聚类的方法，包括 UniRoute（Jitkrittum 等人，2025）、BEST-Route（Ding 等人，2025）、Avengers（Zhang 等人，2025e），以及 Router-Bench 中引入的基线（Hu 等人，2024a）。尽管这些方法取得了成功，但它们主要关注单轮响应，并在相对简单的基准测试（例如 Chatbot Arena（Chiang 等人，2024）、MATH（Hendrycks 等人，2021））上进行评估；在深度研究任务等更复杂的场景中，它们尚未实现细粒度路由。

3 预备知识

本节将概述基于大语言模型的智能体人工智能系统及其运行工作流程，并在此基础上正式定义模型路由的目标。

记号。 我们考虑一个复杂的智能体人工智能系统 $\mathcal{M}$，旨在通过多步骤工作流解决用户提出的查询$\mathcal{Q}$。此类工作流通常由一组专用智能体或角色 $\mathcal{I} = \{1, 2, \ldots, N\}$ 组成，它们按照轮流制协议依次运行，调度器在每个时间步确定当前活动的智能体。工作流可能涉及一系列动作，从任务分解到执行特定子任务，形式化定义如下：

$$\mathcal{M} = \big\langle \mathcal{I}, \mathcal{L}, \phi, \mathcal{S}, \mathcal{T}, \mathcal{A}, \Psi, \mu, \mathcal{Q} \big\rangle, \qquad (1)$$

其中，$\mathcal{I} = \{1, 2, \ldots, N\}$ 表示智能体角色集合（例如，网页浏览器、编码员），$\mathcal{L}$ 表示可用大语言模型骨干的集合。每个智能体$i \in \mathcal{I}$ 通过映射$\phi : \mathcal{I} \to \mathcal{L}$静态分配一个大语言模型。系统状态保存在$\mathcal{S}$中，通常以共享内存或草稿板的形式实现。智能体工作流可能会调用一组外部工具 $\mathcal{T}$，例如代码解释器（Dong 等人，2023年）或网页搜索 API（Jimenez 等人，2024年）。完整的动作空间 $\mathcal{A}$ 既包括自然语言动作，也包括工具调用，形式化表示为$\mathcal{A} = \mathcal{A}_{\text{lang}} \cup \{\text{use_tool}(T, \text{args}) \mid T \in \mathcal{T}\}$。系统的状态转移动态由$\Psi(s_{t+1} \mid s_t, a_t)$决定，而调度器 $\mu(t) \in \mathcal{I}$在每个时间步 t选择当前活动的智能体。

在每一步 t，当前活跃的智能体 $i = \mu(t)$ 会生成一个动作 a_t。该动作由其策略 π_i 生成，而该策略由指定的大语言模型 $\phi(i) \in \mathcal{L}$ 实例化。策略会综合考虑当前状态 s_t、整体查询 $\mathcal{Q}$ 以及相关交互历史 $\mathcal{H}_t$：

$$a_t \sim \pi_i(s_t, \mathcal{H}_t, \mathcal{Q}),\ i = \mu(t),\ \pi_i \leftarrow \phi(i). \quad (2)$$

因此，系统的完整执行轨迹是一系列状态和动作：

$$\tau = (s_0, a_0, s_1, a_1, \ldots, s_T), \qquad (3)$$

其中 T 是终止步骤。查询 $\mathcal{Q}$ 的最终答案是根据整个执行轨迹 τ中汇总的信息综合得出的。

这些成熟的智能体系统有一个关键但往往未明示的特征：智能体到模型的映射 ϕ 是静态且由人类预先定义的。例如，网页浏览器智能体

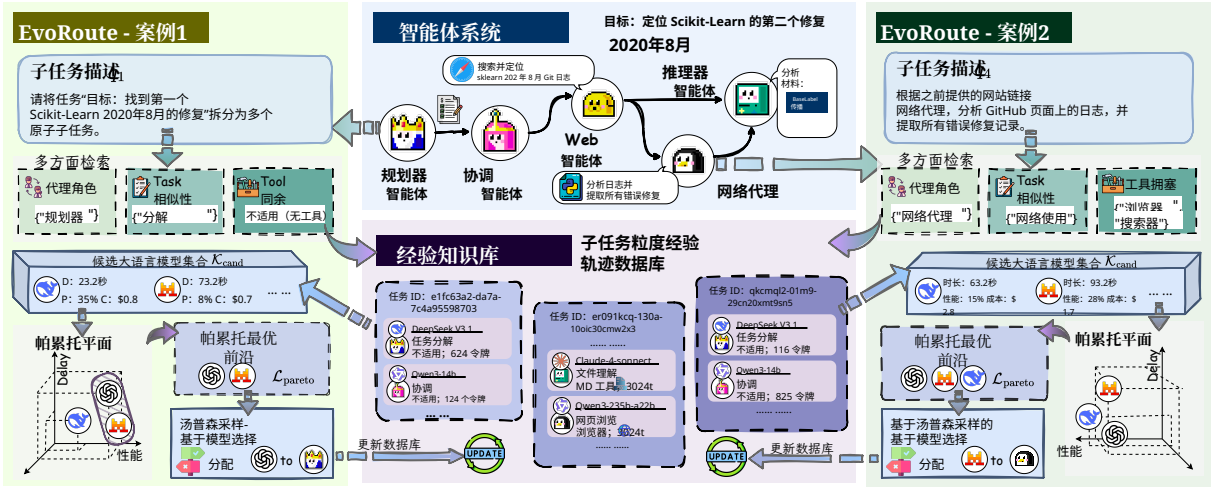


图 2：我们提出的 EvoRoute 概览。

在 OWL-Workforce 中被硬编码为使用 GPT-4o，而 o3-mini 则用于规划器智能体。其他主流框架也采用类似模式，包括微软的 Magentic-one（Fourney 等人，2024年）、Skywork AI 的 AgentOrchestra（Zhang 等人，2025c）、字节跳动的 AIME（Shi 等人，2025b）以及腾讯的 CK-Pro（Fang 等人，2025年）。这种僵化的设计虽然简单直接，却缺乏适应性，也未能考虑子任务在难度和性质上的差异。

目标表述。 本文致力于超越静态智能体到模型映射的局限，通过学习动态路由策略 ρ^*，以最优方式应对智能体系统三难困境。我们将其表述为以下多目标优化问题：

$$\rho^* = \underset{\rho}{\text{argmax}}(\mathbb{E}_{\tau\sim\rho}[\mathbb{P}(\tau)], -\mathbb{E}_{\tau\sim\rho}[\mathbb{C}(\tau)], -\mathbb{E}_{\tau\sim\rho}[\mathbb{D}(\tau)]), \tag{4}$$

其中，ρ 是动态路由策略，它在每个步骤 t 为当前智能体选择一个大语言模型 $l_t \in \mathcal{L}$。期望值 $\mathbb{E}_{\tau\sim\rho}$ 取自该策略下生成的执行轨迹 τ 的分布。对于每条轨迹，我们度量三个关键指标：♦ **性能** $\mathbb{P}(\tau)$，即任务成功分数；♠ **成本** $\mathbb{C}(\tau)$，即累计货币成本和计算开销；以及 ♣ **效率** $\mathbb{D}(\tau)$，即总实际执行时间。因此，追求 ρ^* 等价于寻找一个帕累托最优解，在最大化性能的同时，最小化成本和延迟。

4 方法论

4.1 自进化经验库

如图2所示，EvoRoute的自适应智能核心在于积累并利用过往经验的能力。这通过自进化经验库实现，记为 $\mathcal{K}$，它会细致记录步骤级的执行数据。我们采用一种探索策略来解决 $\mathcal{K}$的冷启动问题，详情见第A节。针对查询 $\mathcal{Q}$完成智能体工作流后，会得到一条轨迹 $\tau = (s_0, a_0, \ldots\ldots, s_T)$，随后对其进行回溯分析。对于每个步骤 $t \in \{0, \ldots, T-1\}$，我们都会提取并保存一条详细记录 $\mathcal{R}_t$，其中涵盖该特定动作的完整上下文和结果。形式上，每条记录 $\mathcal{R}_t$ 都是一个结构化元组，包含应对三难困境所需的多方面信息：

$$\mathcal{R}_t = \langle i_t, l_t, q_t, \mathbf{e}_t, T_t, c_t, d_t, \sigma_t, \mathbb{P}(\tau)\rangle, \tag{5}$$

其中 $i_t = \mu(t)$表示当前代理角色的标识符（例如编码员或协调员），而 $l_t \in \mathcal{L}$ 则指定了在步骤 t中为该角色选定的具体大语言模型骨干。元素 q_t 表示提供给智能体的自然语言子任务指令，其嵌入$\mathbf{e}_t = \text{Embed}(q_t)$用于语义相似性检索。 $T_t \subseteq \mathcal{T}$ 是动作 a_t中调用的工具子集；若未使用工具，则为 $T_t = \emptyset$。术语 c_t 和 d_t 分别表示在此步骤中产生的货币成本和实际耗时。 $\sigma_t \in \{0, 1\}$ 是执行成功的二值指示器（例如，工具调用是否无错误地执行），而 $\mathbb{P}(\tau)$是任务级成功率，用于将步骤 t 的贡献与全局任务结果关联起来。

每次任务执行完成后，知识库 $\mathcal{K}$ 都会通过追加所有新生成的记录来动态更新：

$$\mathcal{K} \leftarrow \mathcal{K} \cup \{\mathcal{R}_t\}_{t=0}^{T-1}. \tag{6}$$

这种细粒度、逐步记录的方式确保 $\mathcal{K}$逐渐发展为一个丰富的知识库，能够捕捉子任务特征、模型选择及其对性能、成本和效率产生的影响之间的细微关联。这些累积的实证知识构成了后续检索和选择阶段的基础。

4.2 多方面检索

建立知识库 $\mathcal{K}$ 后， **EvoRoute** 的第一个主动阶段是在出现新的子任务时，检索一组多样化且在历史上具有相似性的记录。给定步骤 t'中的新子任务，该子任务由活跃智能体 $i_{t'} = \mu($ $t')$及其指令 $q_{t'}$所表征，我们的目标是收集一个全面的候选集 $\mathcal{K}_{\text{cand}} \subseteq \mathcal{K}$。我们不采用严格的标准交集，而是汇总符合以下三个关键方面中至少一个的记录：代理角色、语义相似度或工具使用特征。这种析取策略确保为后续决策过程提供广泛且多样的证据池。形式上，最终候选集 $\mathcal{K}_{\text{cand}}$ 由三个独立检索的子集取并集构成：

$$\mathcal{K}_{\text{cand}} = \mathcal{K}_{\text{agent}} \cup \mathcal{K}_{\text{sem}} \cup \mathcal{K}_{\text{tool}}, \tag{7}$$

其中，每个子集都根据一个不同的相关性方面进行检索：

➤ **智能体角色匹配** 可识别由相同代理角色执行的所有历史步骤，从而捕捉功能等效的先例。

$$\mathcal{K}_{\text{agent}} = \{\mathcal{R}_t \in \mathcal{K} \mid i_t = i_{t'}\} \tag{8}$$

➤ **语义相似性检索** 会检索与当前子任务在语义上接近的子任务记录 $q_{t'}$。我们使用预训练的句子编码器 Embed(·) 和相似性度量指标（例如，余弦相似度），选取指令嵌入$\mathbf{e}_t$ 超过阈值 θ_{sim}的所有记录：

$$\mathcal{K}_{\text{sem}} = \{\mathcal{R}_t \in \mathcal{K} \mid \text{sim}(\text{Embed}(q_{t'}), \mathbf{e}_t) \geq \theta_{\text{sim}}\}, \tag{9}$$

其中 Embed(·) 通过 MiniLM 实现（Wang 等人，2020年），sim(·, ·)采用余弦相似度。

➤**工具一致性检索** 根据操作相似性收集记录，目标是检索可能需要类似工具交互的子任务。此方面专为以下指令而设计： $q_{t'}$预计会使用工具。我们首先采用一个轻量级预测函数 PredictTools$(q_{t'})$（其实现详见第B节），以分析指令并推断一组可能使用的工具，记为 T'_{pred}。如果历史记录 $\mathcal{R}_t$ 所调用的工具集合 T_t 与该预测集合存在非空交集，则检索该记录：

$$\mathcal{K}_{\text{tool}} = \{\mathcal{R}_t \in \mathcal{K} \mid T_t \cap \text{PredictTools}(q_{t'}) \neq \emptyset\}. \tag{10}$$

通过合并这三个集合， **EvoRoute**汇集了丰富且多维的先例集合， $\mathcal{K}_{\text{cand}}$。该集合构成后续筛选阶段的经验基础，我们将在此阶段推导特定于模型的统计数据，并确定帕累托最优前沿。

4.3 帕累托最优筛选与选择

检索到一组历史记录候选集 $\mathcal{K}_{\text{cand}}$后， **EvoRoute** 的最后阶段会将这些经验证据提炼为明确的动作。该过程首先找出检索记录中出现过的独特大语言模型，$\mathcal{L}_{\text{cand}} = \{l_t \mid \mathcal{R}_t \in \mathcal{K}_{\text{cand}}\}$。对于每个候选大语言模型$l \in \mathcal{L}_{\text{cand}}$，我们通过对相关记录子集 $\mathcal{K}_{\text{cand}}(l)$ $= \{\mathcal{R}_t \in \mathcal{K}_{\text{cand}} \mid l_t = l\}$取平均值，计算其汇总三难困境指标。ˆ ˆ ˆ由此得到统计画像（$\mathbb{P}(l)$ $\mathbb{C}(l)$ $\mathbb{D}(l)$），ˆ其中 $\mathbb{P}(l) = \sum_{\mathcal{R}_t \in \mathcal{K}_{\text{cand}}(l)} \mathbb{P}(\tau) \cdot \sigma_t / |\mathcal{K}_{\text{cand}}(l)|$，ˆ ˆ$\mathbb{C}(l) = \sum_{\mathcal{R}_t \in \mathcal{K}_{\text{cand}}($ $=)$ $\sum_{\mathcal{R}_t \in \mathcal{K}_{\text{cand}}\ l} d_t / |\mathcal{K}_{\text{cand}}^{\ l}\ l\ |$ $c_t / |\mathcal{K}_{\text{cand}}(l)|$，以及 $\mathbb{D}(l)\ _{()}( )$。随后，我们进行帕累托筛选。若存在另一个大语言模型 l'，在三个维度上均优于或等同于大语言模型 l（性能更高、成本更低、耗时更短），且至少有一个维度严格优于它，则认为 l 受支配。仅保留非支配模型后，我们便得到帕累托最优集， $\mathcal{L}_{\text{pareto}}$。

从 $\mathcal{L}_{\text{pareto}}$ 中进行纯粹贪婪的选择会抑制探索。为了解决这一问题，我们采用汤普森采样 (Russo 等人，2020年)。我们将三个三难困境指标分别视为服从正态分布的连续变量，并使用正态-逆伽马分布（NIG）共轭先验对其均值和方差的不确定性进行建模。

对于每个模型 $l \in \mathcal{L}_{\text{pareto}}$，我们根据 $\mathcal{K}_{\text{cand}}(l)$ 中的检索证据动态构建其后验分布。首先计算每个度量指标 $m \in \{\mathbb{P}, \mathbb{C}, \mathbb{D}\}$的样本统计量：计数 n_l、样本均值 $\bar{x}_{m,l}$ 和样本方差$s^2_{m,l}$。这些统计量用于参数化 NIG 后验分布 NIG($\mu_{m,l}$,$\nu_{m,l}$,$\alpha_{m,l}$,$\beta_{m,l}$)，其中 $\mu_{m,l} = \bar{x}_{m,l}$,$\nu_{m,l} = n_l$, $\alpha_{m,l} = n_l/2$ 和 $\beta_{m,l} = ($ $n_l - 1)s^2_{m,l}/2$，并假设先验为无信息先验。在决策时，我们从每个模型关于每个度量指标的完整后验分布中抽取一个样本，以

表1：在 GAIA 和 BrowseComp+ 基准测试上，与人工和基于路由的基线进行性能比较。对于 GAIA，结果按各个级别分别报告，这些级别由原始基准测试根据查询难度定义。

	设置	LLM	GAIA（所有级别）			级别 1			级别 2			级别 3			BrowseComp+		
			性能(%)	成本($)	延迟(h)	性能(%)	成本($)	延迟(h)	性能(%)	成本($)	延迟(h)	性能(%)	成本($)	延迟(h)	性能(%)	成本($)	延迟(h)
CK-Pro	人工	Qwen3-14b	10.91	4.16	24.67	16.77	1.06	6.23	10.34	2.58	14.90	0.00	0.52	3.54	6.50	12.50	6.80
		Claude-4	58.28	359.32	45.14	67.92	96.46	6.83	58.14	192.64	25.35	37.50	70.22	12.96	33.50	220.50	20.15
		GPT-4o	42.42	82.86	35.43	59.91	27.56	7.35	39.22	43.86	21.50	15.26	11.44	6.58	24.28	113.46	9.43
		GPT-4.1	48.48	61.84	23.04	56.29	16.96	7.46	49.73	35.26	10.87	26.77	9.62	4.71	31.44	185.01	18.69
	路由	PromptLLM	47.88	128.15	57.70	58.91	31.27	10.88	47.65	80.24	30.93	24.39	16.64	15.89	26.87	133.22	17.27
		GraphRouter	46.67	116.10	53.32	58.13	28.50	10.20	46.23	72.10	28.90	22.94	15.50	14.22	27.90	125.80	16.90
		MasRouter	53.50	120.15	52.95	67.92	27.80	9.80	49.74	77.40	28.15	23.51	14.95	15.00	28.50	128.90	17.10
	Ours	**EvoRoute**	63.19	85.40	38.75	83.02	26.50	6.10	59.30	33.60	18.30	33.33	25.30	14.35	38.72	79.30	10.33
Smolagent	人工	Qwen3-14b	11.52	0.82	8.60	15.35	0.30	3.00	4.73	0.42	1.20	27.39	0.10	4.40	5.80	1.95	4.25
		Claude-4	46.06	76.90	29.10	56.26	21.20	12.80	46.23	41.60	14.50	22.94	14.10	1.80	28.20	85.60	16.50
		GPT-4.1	39.39	18.10	7.90	55.00	5.70	3.20	36.30	8.90	2.60	16.01	3.50	2.10	25.50	58.20	11.20
	路由	PromptLLM	40.00	17.50	12.20	49.80	4.10	6.50	40.92	8.90	3.80	15.04	4.50	1.90	23.10	45.80	10.80
		GraphRouter	38.79	15.70	11.30	47.95	3.70	6.00	39.78	8.00	3.50	15.03	4.00	1.80	24.20	42.50	10.50
		MasRouter	40.61	17.00	12.30	49.82	4.00	6.80	42.07	8.50	3.70	15.04	4.50	1.80	24.80	44.10	10.90
	Ours	**EvoRoute**	56.36	15.60	8.90	73.91	3.70	2.90	53.73	6.20	4.80	27.04	5.70	1.20	32.50	25.50	9.80

生成其效用的随机实现。选择抽样效用最高的模型 l^*：

$$
\begin{aligned}
&\text{对于每个 } l \in \mathcal{L}_{\text{pareto}} \text{ 以及每个度量指标 } m \in \{\mathbb{P}, \mathbb{C}, \mathbb{D}\}: \\
&(\tilde{\mu}_{m,l}, \tilde{\sigma}^2_{m,l}) \sim \text{NIG}(\mu_{m,l}, \nu_{m,l}, \alpha_{m,l}, \beta_{m,l}) \\
&\tilde{x}_{m,l} \sim \mathcal{N}(\tilde{\mu}_{m,l}, \tilde{\sigma}^2_{m,l}), \\
&U'(l) = w_p \cdot \tilde{x}_{\mathbb{P},l} - w_c \cdot \tilde{x}_{\mathbb{C},l} - w_d \cdot \tilde{x}_{\mathbb{D},l} \\
&l^* = \underset{l \in \mathcal{L}_{\text{pareto}}}{\text{argmax}} \left(U'(l) \right),
\end{aligned}
\tag{11}
$$

其中，权重（w_p，w_c，w_d）体现了所需的三难权衡。

关键在于，这种选择并非流程的终点。由 l^* 驱动的智能体完成动作后，观测到的结果会被记录回知识库 $\mathcal{K}$。这就形成了反馈闭环，确保每项决策及其结果都能为系统不断提升的智慧添砖加瓦，从而实现 **EvoRoute** 的自我演进特性。

5 实验

5.1 实验设置

框架。 我们选取了三个具有代表性的智能体框架，并按架构复杂度递增排列： ❶ **ReAct** (Yao 等人， 2023年)，这是一种在“观察–动作–推理”之间迭代交替的单智能体架构； ❷ **Smolagents**[1]，，这是一种由管理智能体和工具智能体组成的双智能体框架；以及 ❸ **Cognitive Kernel-Pro** (Fang 等人，2025年)，由腾讯推出，于2025年8月在 GAIA 上取得开源领域的 SOTA，并采用了分层多智能体结构（例如，主智能体、网络代理和文件智能体。）

基线。 我们主要与两类基线进行比较： **(i) 手动设置**，包括各框架原先预定义的大语言模型配置，以及人工指定的设置（例如，统一使用 `gpt-4.1`或`Gemini-2.5-pro`）。人工分配的大语言模型可能因框架的兼容性要求而有所不同。 **(ii) 模型路由**，包括 PromptLLM（Feng 等人， 2024年）、MasRouter（Yue 等人， 2025年）和 GraphRouter（Feng 等人，2024年）[2]。

方法配置。 我们用不同价格和规模的模型填充大语言模型池 $\mathcal{L}$ ：{通义千问-14B、`GPT-4.1`、`GPT-4o`、`Claude-4-Sonnet`(`Claude-4`)、`Gemini-2.5-Flash`、`Gemini-2.5-Pro`}。模型价格列于第 C 节。

参数配置。 对于我们的多方面检索，我们将语义相似度阈值 θ_{sim} 设为 0.85，以平衡检索召回率和精确率。在选择阶段，我们为每个度量指标采用无信息 NIG 先验（$\nu_0 = 0$， $\alpha_0 = 0.5$，$\beta_0 = 0$）。最终选择由一个通过人工配置权重的效用函数指导

[1] https://github.com/huggingface/smolagents

[2] 我们的代码将在 https://github.com/bingreeky/evo-route。

表2：HotpotQA、DS-1000 和 DDXPlus 基准测试上的性能比较。

设置	方法/模型	DS-1000			HotpotQA			DDXPlus			Avg.		
		性能	成本	延迟	性能	成本	延迟	性能	成本	延迟	性能	成本	延迟
人工	GPT-4o	41.40	54.00	12.32	83.84	391.98	67.29	57.58	135.10	34.31	60.94	581.08	113.92
	GPT-4.1	29.40	49.59	12.01	87.00	213.22	78.77	55.30	89.82	21.95	57.23	352.63	112.73
	Gemini-2.5-Pro	54.30	68.33	16.17	87.40	343.43	79.18	81.10	109.85	27.65	74.27	521.61	123.00
	Qwen3-14b	38.20	12.68	10.86	81.40	22.96	35.52	41.26	14.16	31.67	53.62	49.80	78.05
路由	PromptLLM	52.00	26.66	11.09	85.20	47.02	62.46	60.07	127.05	26.88	65.76	200.73	100.43
	GraphRouter	52.80	24.50	11.50	86.10	64.80	61.80	62.50	119.50	27.50	67.13	208.80	100.80
	MasRouter	53.50	25.10	11.80	88.50	59.50	53.10	73.10	92.38	32.30	71.70	176.98	97.20
Ours	**EvoRoute**	56.50	23.50	11.20	87.80	49.10	60.50	79.50	65.80	20.53	74.60	138.40	92.23

$(w_p,\ w_c,\ w_d) = (1.0,\ 0.1,\ 0.05)$，该函数在优先考虑性能的同时，也会惩罚成本和延迟。

基准测试与评估。我们在五个基准测试上评估了我们的方法。其中两个是深度研究基准测试：GAIA（Mialon 等人，2023年）是一种广泛使用的通用智能体基准测试，用于评估文件读取、网页浏览和编程等能力，并根据难度分为三个层级；BrowseComp+（Chen 等人，2025c）是 OpenAI 的 BrowseComp（Wei 等人，2025年）的增强版本，修正了错误案例，并提供了更稳定的评估。其余三个基准来自 StreamBench（Wu 等人，2024年）：用于数据科学任务的 DS-1000（Lai 等人，2022年）、用于网络搜索的 HotpotQA（Yang 等人，2018年），以及用于医学推理的 DDXPlus（Tchango 等人，2022年）。上述数据集的统计信息见 D 节。

5.2 主要结果

表 1和 2 展示了`EvoRoute` 在以下三个关键维度上与不同基线的对比：**性能**、**成本**和**延迟**。图3 进一步以雷达图可视化的形式展示了 GAIA 不同难度等级下的结果。我们的主要发现总结如下。

现有智能体系统难以应对性能、效率与经济性三难困境。我们的实证分析表明，当代智能体系统在性能、成本和延迟之间存在根本性取舍。实现最先进性能通常需要付出高昂的成本并承受显著的延迟。例如，

使用强大的 `Gemini-2.5-Pro+ReAct` 实现了领先的 74。27% 平均性能，但成本高得令人难以承受，达到 $ 521。61，延迟长达 123 小时，如表2所示。同样，`Claude-4+CK-Pro` 在 GAIA 上取得了出色的 55。15% 准确率，但成本高达难以持续承受的 $359。32

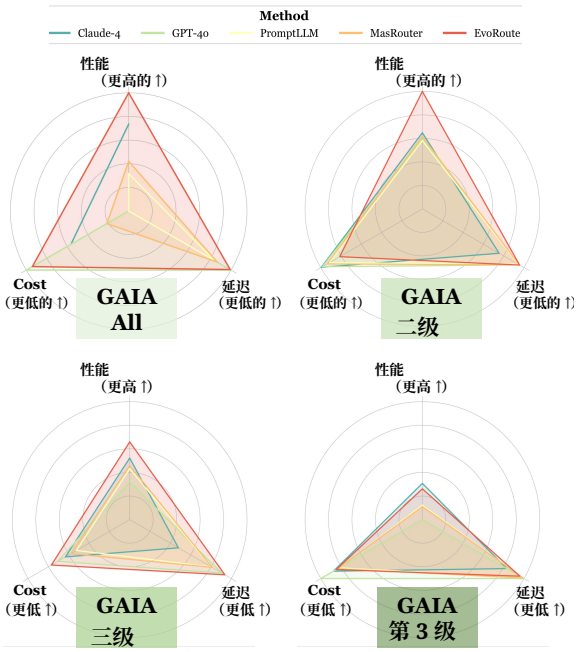


图3：对 GAIA 所有子集的三项关键度量指标进行比较分析：性能、成本和延迟。所有度量指标均经过全局归一化，成本/延迟的数值已反转，因此包围面积越大，表示经济性/效率越高。

（表1）。相反，选择 `Qwen3-14b+CK-Pro` 等经济型模型，可将成本大幅降至仅 $4。16，但代价是性能骤降至 10。91%。尽管 MasRouter 等现有最先进的路由方法能在一定程度上缓解这一问题（平均性能达到 71。70%，成本为$176。98），但带来的改进有限。这些方法的准确率仍落后于表现最佳的模型，无法从根本上打破这种权衡。

`EvoRoute` 有效缓解三难困境。相比之下，`EvoRoute` 展现出卓越的三难困境应对能力：在持续达到最先进性能的同时，大幅降低成本和延迟。与 CK-Pro 集成后，其表现超越了强大的 Claude-4 基线——

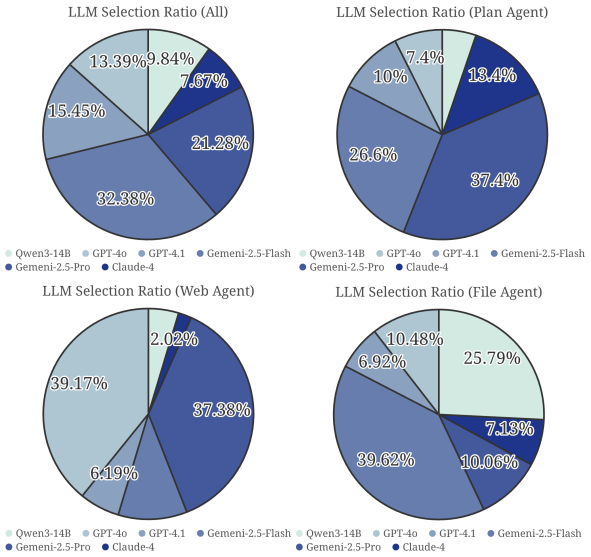


图4：CK- Pro 在不同智能体角色中的大语言模型选择分布。

在 GAIA 上的成绩（63.18% 对 58.28%），运营成本降低了逾 76%（85.40 美元对 359.32 美元），同时保持了相当的延迟。在 BrowseComp+上，它同样优于 Claude-4（38.72% 对 33.50%），成本不到其 36%（79.30 美元对 220.50 美元），延迟约为其一半（10.33 小时对 20.15 小时）。在所有测试配置中（表2），`EvoRoute`的平均性能最高（74.60%），略高于 Gemini-2.5-Pro（74.27%），而成本仅为其 26%（138.40 美元对 521.61 美元），延迟则缩短了 25%（92.23 小时对 123.00 小时）。这些结果有力地证明，`EvoRoute` 是解决智能体系统三难困境的有效方案。

5.3 框架分析

可视化。我们分析了`EvoRoute`在 CK-Pro+ GAIA 上不同智能体角色中的大语言模型选择情况（图4）。结果表明，模型分配与子任务复杂度相匹配：由规划智能体处理的认知要求较高的任务倾向于使用高能力模型（Gemini-2.5-Pro 占37.4%，Claude-4 占13.4%）；而文件智能体较为简单的操作性任务则主要使用高成本效益模型（Qwen3-14B 占25.79%），其中 Gemini-2.5-Pro 和 Claude-4 等高端模型的占比分别降至10.06%和7.13%。这表明，`EvoRoute` 会策略性地将最强大（也最昂贵）的模型保留用于关键推理，同时采用高效模型处理常规子任务。

消融研究 我们在 GAIA Level 1 上进行了消融研究，以验证核心组件（表 3）。我们评估了四种变体：**不使用 $\mathcal{K}$ 冷启动**（从空知识库开始）、**不使用多方面检索**（仅使用语义相似度）、**不使用汤普森采样**（采用贪心策略），以及**不使用帕累托过滤**（不进行模型剪枝）。缺少引导初始化的知识库所造成的影响最为严重，使性能骤降 13.21% ↓。停用多方面检索的影响几乎同样严重，导致性能下降 11.22%，并产生所有变体中最高的成本（$32.10）和延迟（6.95 小时）。此外，以贪心策略替代汤普森采样会使性能降低 6.52%，而省略帕累托过滤仅导致较小的 1.52% 降幅，这证实了有原则的探索和以效率为导向的剪枝对于系统整体效能都至关重要。

表3：四个变体的消融研究，每个变体禁用一个关键组件（在 GAIA Level 1 上测试）。

方法变体	性能（%）	成本（$）	延迟（小时）
`EvoRoute`（完整模型）	**83.02**	**26.50**	**6.10**
无 $\mathcal{K}$ 冷启动	69.81 (-13.21)	29.50	6.30
不使用多方面检索	71.70 (-11.22)	32.10	6.95
不使用汤普森采样	76.50 (-6.52)	29.80	6.40
不使用帕累托过滤	81.50 (-1.52)	28.20	6.35

简单任务上的架构开销。值得注意的是，在第5.2节中，ReAct 在较简单的任务（DS-1000、HotpotQA）上接受评估，而 Smolagent 和 CK-Pro 则用于 GAIA 和 BrowseComp+。这反映了一个实际观察：在简单任务上部署更复杂的框架可能会适得其反。例如，我们最初发现，`Gemini-2.5-Pro`+搭配 Smolagent 时，在 DS-1000 上的得分仅为31。7%，低于更简单的 38。2% ，后者由`Qwen3-14b`+搭配 ReAct 获得。轨迹分析表明，Smolagent 的功能（例如网络搜索和高级规划）会使原本只需一次大语言模型输入输出即可解决的任务变得过于复杂，从而带来额外开销。我们将此称为“架构过拟合”：结构复杂性和专用工具非但无益，反而会造成阻碍。这表明，最优的智能体架构应与任务的内在复杂度相匹配，而非一味追求最高级的复杂程度。

6 结论

在本研究中，我们提出了`EvoRoute`，这是一种动态模型路由框架，旨在系统性地解决智能体系统三难困境。我们在 GAIA 和 BrowseComp 等具有挑战性的基准测试上开展的实验+ 从实证角度验证了这一范式：在不牺牲任务成功率的前提下，货币成本最高降低 80%，延迟降低超过 70%。最终，`EvoRoute` 代-

表着朝着让强大的智能体人工智能系统在真实世界部署中变得更加实用、可扩展且具有经济可行性迈出的关键一步。

局限性与伦理问题

我们的评估聚焦于 CK-Pro 和 Smolagent，并未选择 OWL、Agent-Orchestra 或 AIME 等其他深度研究框架。鉴于深度研究系统层出不穷，且其架构范式大体相似——通常由中央协调器统筹多个专用子智能体——我们认为，实验已涵盖从简单到复杂、具有代表性的智能体系统谱系。此外，这些系统的令牌成本高昂（如表1所示），使得开展全面的大规模基准测试并不切实际。关于伦理考量，由于本研究仅依赖标准公开基准和广泛使用的开放框架，我们认为这项工作不存在明显的伦理风险。

贡献

G. Zhang 主要负责方法实现、稿件准备、可视化和实验分析。K. Yang 开展了相当一部分实验工作，也是核心贡献者之一。H. Yu 和 B. Wu 在整个项目过程中提供了大量指导，并参与了深入讨论。F. Huang、Y. Li 和 S. Yan 提供了资深层面的监督和战略指导。

参考文献

Joshua Aizenman、Menzie David Chinn 和 Hiro Ito。2013年。全球失衡时代的“不可能三角”假说：测量与检验。《国际经济学评论》，21(3)：447–458。

Jun Shern Chan、Neil Chowdhury、Oliver Jaffe、James Aung、Dane Sherburn、Evan Mays、Giulio Starace、Kevin Liu、Leon Maksin、Tejal Patwardhan、Lilian Weng 和 Aleksander Madry。2025年。MLE-Bench：评估机器学习智能体在机器学习工程中的表现。预印本，arXiv:2410.07095。

Gohar Irfan Chaudhry、Esha Choukse、Íñigo Goiri、Rodrigo Fonseca、Adam Belay 和 Ricardo Bianchini。2025年。迈向资源高效的复合人工智能系统。预印本，arXiv:2501.16634。

Guangyao Chen、Siwei Dong、Yu Shu、Ge Zhang、Jaward Sesay、Börje F. Karlsson、Jie Fu 和 Yemin Shi。2023年。Autoagents：自动智能体生成框架。*CoRR*，abs/2309.17288。

Kaiyuan Chen、Yixin Ren、Yang Liu、Xiaobo Hu、Hao- tong Tian、Tianbao Xie、Fangfu Liu、Haoye Zhang、Hongzhang Liu、Yuan Gong、Chen Sun、Han Hou、Hui Yang、James Pan、Jianan Lou、Jiayi Mao、Jizheng Liu、Jinpeng Li、Kangyi Liu 及其他 14 位作者。2025a。XBench：通过与职业相匹配的现实世界评估追踪智能体生产力的规模化增长。预印本，arXiv:2506.13651。

Lingjiao Chen、Jared Quincy Davis、Boris Hanin、Peter Bailis、Matei Zaharia、James Zou 和 Ion Stoica。2025年。优化复合人工智能系统的模型选择。预印本，arXiv:2502.14815。

Shuhao Chen、Weisen Jiang、Baijiong Lin、James T Kwok 和 Yu Zhang。2024年。RouterDC：通过双重对比学习进行查询驱动的路由器，用于组装大语言模型。*arXiv* 预印本 *arXiv:2409.19886*。

Zijian Chen、Xueguang Ma、Shengyao Zhuang、Ping Nie、Kai Zou、Andrew Liu、Joshua Green、Kshama Patel、Ruoxi Meng、Mingyi Su、Sahel Shari- fymoghaddam、Yanxi Li、Haoran Hong、Xinyu Shi、Xuye Liu、Nandan Thakur、Crystina Zhang、Luyu Gao、Wenhu Chen 和 Jimmy Lin。2025年。BrowseComp-Plus：更公平、更透明的深度研究智能体评测基准。预印本，arXiv:2508.06600。

Wei-Lin Chiang、Lianmin Zheng、Ying Sheng、Anastasios Nikolas Angelopoulos、Tianle Li、Dacheng Li、Hao Zhang、BanghuaZhu、MichaelJordan、JosephE. Gonzalez 和 Ion Stoica。2024年。Chatbot arena：一个通过人类偏好评估大语言模型的开放平台。预印本，arXiv:2403.04132。

Gheorghe Comanici、Eric Bieber、Mike Schaekermann、Ice Pasupat、Noveen Sachdeva、Inderjit Dhillon、Marcel Blistein、Ori Ram、Dan Zhang、Evan Rosen 和另外 1 位作者。2025年。Gemini 2.5：以高级推理、多模态、长上下文和新一代智能体能力推动前沿。*arXiv* 预印本 *arXiv:2507.06261*。

Dujian Ding、Ankur Mallick、Shaokun Zhang、Chi Wang、Daniel Madrigal、Mirian DelCarmen Hipolito Garcia、Menglin Xia、Laks V. S. Lakshmanan、Qingyun Wu 和 Victor Rühle。2025年。BEST-route：利用测试时最优计算的自适应大语言模型路由。发表于第四十二届国际机器学习大会。

Yihong Dong、XueJiang、ZhiJin 和 Ge Li。2023年。通过 ChatGPT 进行协作式代码生成。*arXiv* 电子预印本，arXiv:2304.07590。

Danny Driess、Fei Xia、Mehdi SM Sajjadi、Corey Lynch、Aakanksha Chowdhery、Ayzaan Wahid、Jonathan Tompson、Quan Vuong、Tianhe Yu、Wenlong Huang 等 1 人。2023年。PaLM-E：一种具身多模态语言模型。

Lutfi Eren Erdogan、NicholasLee、Sehoon Kim、Suhong Moon、Hiroki Furuta、Gopala Anumanchipalli、Kurt Keutzer 和 Amir Gholami。2025年。计划与行动：

提升智能体在长时程任务中的规划能力。*arXiv* 预印本 *arXiv:2503.09572*。

Tianqing Fang、Zhisong Zhang、Xiaoyang Wang、Rui Wang、Can Qin、Yuxuan Wan、Jun-Yu Ma、Ce Zhang、Jiaqi Chen、Xiyun Li、Hongming Zhang、Haitao Mi 和 Dong Yu。2025年。Cognitive kernel-pro：用于深度研究智能体和智能体基础模型训练的框架。预印本，arXiv:2508.00414。

Tao Feng、Yanzhen Shen 和 Jiaxuan You。2024年。Graphrouter：一种基于图的大语言模型选择路由器。预印本，arXiv:2410.03834。

Adam Fourney、Gagan Bansal、Hussein Mozannar、Cheng Tan、EduardoSalinas、Erkang、Zhu、Friederike Niedtner、Grace Proebsting、Griffin Bassman、Jack Gerrits、Jacob Alber、Peter Chang、Ricky Loynd、Robert West、Victor Dibia、Ahmed Awadallah、Ece Kamar、Rafah Hosn 和 Saleema Amershi。2024年。Magentic-one：用于解决复杂任务的通用多智能体系统。预印本，arXiv:2411.04468。

Evan Frick、Connor Chen、Joseph Tennyson、Tianle Li、Wei-Lin Chiang、Anastasios N. Angelopoulos 和 Ion Stoica。2025年。Prompt-to-leaderboard。预印本，arXiv:2502.14855。

Hongcheng Gao、Yue Liu、Yufei He、Longxu Dou、Chao Du、Zhijie Deng、Bryan Hooi、Min Lin 和 Tianyu Pang。2025年。Flowreasoner：强化查询级元智能体。预印本，arXiv：2504.15257。Dan Hendrycks、Collin Burns、Saurav Kadavath、Akul Arora、Steven Basart、Eric Tang、Dawn Song 和 Jacob Steinhardt。2021年。使用数学数据集衡量数学问题求解能力。*NeurIPS*。Mengkang Hu、Yuhang Zhou、Wendong Fan、Yuzhou Nie、Bowei Xia、Tao Sun、Ziyu Ye、Zhaoxuan Jin、Yingru Li、Qiguang Chen、Zeyu Zhang、Yifeng Wang、Qianshuo Ye、Bernard Ghanem、Ping Luo 和 Guo- hao Li。2025年。Owl：面向真实世界任务自动化中通用多智能体辅助的优化劳动力学习。预印本，arXiv：2505.23885。Qitian Jason Hu、Jacob Bieker、Xiuyu Li、Nan Jiang、Benjamin Keigwin、Gaurav Ranganath、Kurt Keutzer 和 Shriyash Kaustubh Upadhyay。2024a。Router- bench：多大语言模型路由系统的基准测试。预印本，arXiv：2403.12031。Shengran Hu、Cong Lu 和 Jeff Clune。2024b。智能体系统的自动化设计。*arXiv* 预印本 *arXiv*：*2408.08435*。Dongfu Jiang、Xiang Ren 和 Bill Yuchen Lin。2023年。LLM-blender：通过成对排序和生成式融合集成大语言模型。载于计算语言学协会第 *61* 届年会论文集（第 *1* 卷：长篇论文），第 14165–14178 页，多伦多，加拿大。计算语言学协会。

Carlos E. Jimenez、John Yang、Alexander Wettig、Shunyu Yao、Kexin Pei、Ofir Press 和 Karthik Narasimhan。2024年。SWE-bench：语言模型能否解决真实世界的 GitHub 问题？预印本，arXiv：2310.06770。

Wittawat Jitkrittum、Harikrishna Narasimhan、AnkitSingh Rawat、Jeevesh Juneja、Congchao Wang、Zifeng Wang、Alec Go、Chen-Yu Lee、Pradeep Shenoy、Rina Panigrahy、Aditya Krishna Menon 和 Sanjiv Kumar。2025年。用于高效大语言模型推理的通用模型路由。预印本，arXiv:2502.08773。

Omar Khattab、Arnav Singhvi、Paridhi Maheshwari、Zhiyuan Zhang、Keshav Santhanam、Sri Vard- hamanan、Saiful Haq、Ashutosh Sharma、Thomas T Joshi、Hanna Moazam 等 1 人。2023年。DSPy：将声明式语言模型调用编译为自我改进流水线。*arXiv* 预印本 *arXiv:2310.03714*。

DimitriosKoutsoupakis.2021. 有关不可能三角？可自由兑换的加密货币是否获得豁免？《经济研究杂志》，48(8):1480–1496。

Yuhang Lai、Chengxi Li、Yiming Wang、Tianyi Zhang、Ruiqi Zhong、Luke Zettlemoyer、Scott Wen tau Yih、DanielFried、Sida Wang 和 Tao Yu。2022年。DS-1000：用于数据科学代码生成的自然且可靠的基准测试。预印本，arXiv:2211.11501。

Guohao Li、Hasan Hammoud、Hani Itani、Dmitrii Khizbullin 和 Bernard Ghanem。2023年。CAMEL：用于探索大语言模型社会的“心智”的交互式智能体。载于*NeurIPS*。Manling Li、Shiyu Zhao、Qineng Wang、Kangrui Wang、Yu Zhou、Sanjana Srivastava、Cem Gokmen、Tony Lee、Erran Li Li、Ruohan Zhang 等 1 人。2024年。具身智能体接口：针对具身决策对大语言模型进行基准测试。神经信息处理系统进展，37：100428–100534。Zexi Liu、Yuzhu Cai、Xinyu Zhu、Yujie Zheng、Runkun Chen、Ying Wen、Yanfeng Wang、Weinan E 和 Siheng Chen。2025年。ML-Master：通过整合探索与推理迈向 AI-for-AI。预印本，arXiv:2506.16499。Keming Lu、Hongyi Yuan、Runji Lin、Junyang Lin、Zheng Yuan、Chang Zhou 和 Jingren Zhou。2023年。专家路由：高效的奖励引导大语言模型集成。预印本，arXiv:2311.08692。Tula Masterman、Sandi Besen、Mason Sawtell 和 Alex Chao。2024年。面向推理、规划和工具调用的新兴人工智能智能体架构概览：综述。*arXiv* 预印本 *arXiv:2404.11584*。Grégoire Mialon、Clémentine Fourrier、Thomas Wolf、Yann LeCun 和 Thomas Scialom。2023年。GAIA：通用人工智能助手的基准测试。载于第十二届国际学习表征会议。

Fan Nie , Lan Feng , Haotian Ye , Weixin Liang , Pan Lu , Huaxiu Yao、Alexandre Alahi 和 James Zou。2025年。弱者对强者：训练弱元代理以驾驭强大执行器。预印本，arXiv:2504.04785。Isaac Ong 、 Amjad Almahairi、Vincent Wu、Wei-Lin Chiang、Tianhao Wu、Joseph E. Gonzalez、M Waleed Kadous 和 Ion Stoica。2024年。RouteLLM：利用偏好数据学习大语言模型路由。 预印本，arXiv:2406.18665。OpenAI。 [链接]。

Pranav Putta、Edmund Mills、Naman Garg、Sumeet Motwani、Chelsea Finn、Divyansh Garg 和 Rafael Rafailov。2024年。Agent Q：自主人工智能代理的高级推理与学习。*arXiv* 预印本 *arXiv:2408.07199*。

Chen Qian、Xin Cong、Cheng Yang、Weize Chen、Yusheng Su、Juyuan Xu、Zhiyuan Liu 和 Maosong Sun。2023年。用于软件开发的交互式智能体。25 页，9 幅图，2 张表。

Toran Bruce Richards 等。2023年。Auto-GPT：一个自主的 GPT-4 实验。https://github.com/Significant-Gravitas/Auto-GPT。

Daniel Russo、Benjamin Van Roy、Abbas Kazerouni、Ian Osband 和 Zheng Wen。2020年。汤普森采样教程。预印本，arXiv:1707.02038。

Dingfeng Shi、Jingyi Cao、Qianben Chen、Weichen Sun、Weizhen Li、Hongxuan Lu、Fangchen Dong、Tianrui Qin、King Zhu、Minghao Liu、Jian Yang、Ge Zhang、Jiaheng Liu、Changwang Zhang、Jun Wang、Yuchen Eleanor Jiang 和 Wangchunshu Zhou。2025a。Taskcraft：自动生成智能体任务。预印本，arXiv:2506.10055。

Yexuan Shi、Mingyu Wang、Yunxiang Cao、Hongjie Lai、Junjian Lan、Xin Han、Yu Wang、Jie Geng、Zhenan Li、Zihao Xia、Xiang Chen、Chen Li、Jian Xu、Wenbo Duan 和 Yuanshuo Zhu。2025年。 AIME：迈向完全自主的多智能体框架。 预印本，arXiv:2507.11988。

KV Aditya Srivatsa、Kaushal Kumar Maurya 和 Eka- terina Kochmar。2024年。Harnessingthe po wer ofmultiple minds: Lessons learnedfrom llm routing。预印本，arXiv:2405.00467。Arsene Fansi Tchango、Rishab Goel、Zhi Wen、Julien Martel 和 Jou mana Ghosn。2022年。Ddxplus: A newdatasetforautomatic medical diagnosis。预印本，arXiv:2205.09148。Wenhui Wang、Furu Wei、Li Dong、Hangbo Bao、Nan Yang 和 Ming Zhou。2020年。MiniLM：面向预训练变换器任务无关压缩的深度自注意力蒸馏。神经信息处理系统进展，33:5776–5788。Jason Wei、Zhiqing Sun、Spencer Papay、Scott McK- inney、Jeffrey Han、Isa Fulford、Hyung Won Chung、Alex Tachard Passos、William Fedus 和 Amelia Glaese。2025年。Browsecomp：面向浏览智能体的简单却极具挑战性的基准测试。预印本，arXiv:2504.12516。Yuxi Wei、Zi Wang、Yifan Lu、Chenxin Xu、Changxing Liu、Hao Zhao、Siheng Chen 和 Yanfeng Wang。2024年。通过协作式大语言模型智能体实现自动驾驶的可编辑场景模拟。载于*IEEE/CVF*计算机视觉与模式识别会议论文集，第15077–15087页。Cheng-Kuang Wu、Zhi Rui Tam、Chieh-Yen Lin、Yun- Nung Chen 和 Hung yi Lee。2024年。Streambench：迈向语言智能体持续改进的基准测试。预印本，arXiv:2406.08747。

Qingyun Wu、Gagan Bansal、Jieyu Zhang、Yiran Wu、Shaokun Zhang、Erkang Zhu、Beibin Li、Li Jiang、Xiaoyun Zhang 和 Chi Wang。2023年。AutoGen：通过多智能体对话框架实现新一代大语言模型应用。

Shirley Wu、Parth Sarthi、Shiyu Zhao、Aaron Lee、Herumb Shandilya、Adrian Mladenic Grobelnik、Nurendra Choudhary、Eddie Huang、Karthik Subbian、Linjun Zhang 和其他 1 位作者。2025年。Optimas：利用全局对齐的局部奖励优化复合人工智能系统。*arXiv* 预印本 *arXiv:2507.03041*。

Xu Yang、Xiao Yang、Shikai Fang、Bowen Xian、Yuante Li、Jian Wang、Minrui Xu、Haoran Pan、Xinpeng Hong、Weiqing Liu 等 1 人。2025年。R&D Agent：通过大语言模型驱动的自动化研究、开发和演进，实现数据驱动的人工智能解决方案构建自动化。*arXiv* 预印本 *arXiv:2505.14738*。

Yijun Yang、Tianyi Zhou、Kanxue Li、Dapeng Tao、Lu- song Li、Li Shen、Xiaodong He、Jing Jiang 和 Yuhui Shi。2024年。由大语言模型通过并行文本世界训练的具身多模态智能体。发表于*IEEE/CVF*计算机视觉与模式识别会议论文集，第 26275–26285 页。

Zhilin Yang、Peng Qi、Saizheng Zhang、Yoshua Ben- gio、William W Cohen、Ruslan Salakhutdinov 和 Christopher D Manning。2018年。HotpotQA：一个用于多样化、可解释的多跳问答的数据集。*arXiv* 预印本 *arXiv:1809.09600*。

Shunyu Yao、Jeffrey Zhao、Dian Yu、Nan Du、Izhak Shafran、Karthik R Narasimhan 和 Yuan Cao。2023年。ReAct：在语言模型中协同推理与行动。载于第十一届国际学习表征会议。Yanwei Yue、Guibin Zhang、Boyang Liu、Guancheng Wan、Kun Wang、Dawei Cheng 和 Yiyan Qi。2025年。MasRouter：为多智能体系统学习大语言模型路由。*arXiv* 预印本 *arXiv:2502.11133*。Mert Yuksekgonul、Federico Bianchi、Joseph Boen、Sheng Liu、Zhi Huang、Carlos Guestrin 和 James Zou。2024年。TextGrad：通过文本实现自动“微分”。预印本，arXiv:2406.07496。

Guibin Zhang、Kaijie Chen、Guancheng Wan、Heng Chang、Hong Cheng、Kun Wang、Shuyue Hu 和 Lei Bai。2025a。Evoflow：即时演化多样化的智能体工作流。*arXiv* 预印本 *arXiv:2502.07373*。

Guibin Zhang、Luyang Niu、Junfeng Fang、Kun Wang、Lei Bai 和 Xiang Wang。2025b。通过智能体超网络进行多智能体架构搜索。*arXiv* 预印本 *arXiv:2502.04180*。

Guibin Zhang、Yanwei Yue、Xiangguo Sun、Guancheng Wan、Miao Yu、Junfeng Fang、Kun Wang、Tian- long Chen 和 Dawei Cheng。2024a。G-designer：通过图神经网络设计多智能体通信拓扑。*arXiv* 预印本 *arXiv:2410.11782*。

Jiayi Zhang、Jinyu Xiang、Zhaoyang Yu、Fengwei Teng、Xionghui Chen、Jiaqi Chen、Mingchen Zhuge、Xin Cheng、Sirui Hong、Jinlin Wang、Bingnan Zheng、Bang Liu、Yuyu Luo 和 Chenglin Wu。2024b。AFlow：自动化智能体工作流生成。*arXiv* 预印本。ArXiv:2410.10762。

Wentao Zhang、Ce Cui、YileiZhao、Rui Hu、Yang Liu、Yahui Zhou 和 Bo An。2025c。Agentorchestra：用于通用任务求解的分层多智能体框架。预印本，arXiv:2506.12508。

Yi-Kai Zhang、De-Chuan Zhan 和 Han-Jia Ye。2025d。能力指令微调：动态大语言模型路由的新范式。预印本，arXiv:2502.17282。

Yiqun Zhang、Hao Li、Chenxu Wang、Linyao Chen、Qiaosheng Zhang、Peng Ye、Shi Feng、Daling Wang、Zhen Wang、Xinrun Wang、Jia Xu、Lei Bai、Wanli Ouyang 和 ShuyueHu。2025e。复仇者联盟：联合小型语言模型挑战专有巨型模型的简单方案。预印本，arXiv:2505.19797。

Sipeng Zheng、Jiazheng Liu、Yicheng Feng 和 Zongqing Lu。2023年。Steve-eye：为开放世界中的具身智能体配备视觉感知。*arXiv*预印本 *arXiv:2310.13255*。

Yuqi Zhu、Shuofei Qiao、Yixin Ou、Shumin Deng、Shiwei Lyu、Yue Shen、Lei Liang、Jinjie Gu、Hua-jun Chen 和 Ningyu Zhang。2024年。Knowagent：面向基于大语言模型的智能体的知识增强规划。*arXiv* 预印本 *arXiv:2403.03101*。

Mingchen Zhuge、Wenyi Wang、Louis Kirsch、Francesco Faccio、Dmitrii Khizbullin 和 Jürgen Schmidhuber。2024年。GPTSwarm：作为可优化图的语言智能体。收录于第四十一届国际机器学习大会。

冷启动问题

任何基于经验的系统都面临一个根本性挑战，即“冷启动”问题：经验库 $\mathcal{K}$ 最初为空（白板状态），因此检索和路由机制无法发挥作用，因为它们没有可用于制定决策的历史数据。为解决这一问题，我们实施了一种专门的探索策略，旨在系统投入实际运行之前，用多样且富有信息量的初始经验填充 $\mathcal{K}$。

我们利用从 TaskCraft 数据集中精选的 50 个智能体任务（Shi 等人，2025a）。对于这 50 个任务中的每一个，我们都从头到尾运行 Smolagent 和 CK-Pro 框架。我们采用随机探索策略，以最大限度地提高所收集数据的多样性。具体而言，在任务执行的每个步骤 t 中，活动智能体 l_t 所使用的大语言模型骨干 $i_t = \mu(t)$ 均通过均匀随机抽样，从所有可用模型中选出 $\mathcal{L}$。形式化地说，选择过程如下：

$$l_t \sim \mathcal{U}(\mathcal{L}), \tag{12}$$

其中，$\mathcal{U}(\mathcal{L})$ 表示大语言模型骨干离散集合上的均匀分布。

完成 50 个任务中的每一个后，都会按照我们主要方法论中所述的方式处理完整执行轨迹 τ。提取完整的逐步记录集合 $\{\mathcal{R}_t\}_{t=0}^{T-1}$，并用于填充最初为空的经验库 $\mathcal{K}$。

我们进一步强调，冷启动过程的成本并不高：仅需 28.8 美元和约 3.6 小时即可完成，并生成约 480 条步骤级记录，从而提供信息足够丰富的先验。总体而言，这项初始化开销仍然不大。

B 工具预测函数

工具预测函数 PredictTools(q_t)采用两阶段混合策略，在保证预测准确率的同时尽量降低计算开销。首先，该函数会立即执行一次启发式检查，使用预定义词典将明确的触发关键词映射到相应工具（例如，“search”对应 web_search；“run”“plot”对应 code_interpreter）。这种方式能够以零延迟、零 API 成本处理大多数明确的情况。只有在初始启发式检查未能找到匹配项时，该函数才会启用更强大的备用方案：调用一次低成本且高效的大语言模型（实际使用 Qwen3-14b）。该模型以零样本方式接受提示，分析指令的语义，并从可用工具中识别所需工具。

表 4：我们实验中使用的大语言模型骨干库（$\mathcal{L}$）及其每百万令牌的相应价格。所选模型涵盖了多种能力和运营成本。

模型	输入价格（美元/百万）	输出价格（美元/百万）
Gemini-2.5-Flash	$0.3	$2.5
Qwen3-14B	$0.05	$0.22
Claude-4	$3.00	$15.00
GPT-4o	$2.50	$10.00
Gemini-2.5-Pro	$1.25	$10.00
GPT-4.1	$2.00	$8.00

C 模型价格

D 数据集详情

GAIA 基准（Mialon 等人，2023年）为通用人工智能助手提供了一套广泛的评估任务，包含 165 项任务，并系统地划分为三个难度等级：53 项基础任务（第 1 级）、86 项中级任务（第 2 级）和 26 项高级任务（第 3 级）。BrowseC-omp+ 基准测试包含 830 个评估实例，而 DS-1000 包含 1,000 项任务。HotpotQA 包含 7,405 个查询。DDXPlus 最初约含 130K 个样本；我们从中随机抽取 1,000 个实例用于评估。

E 人工智能助手的使用

我们使用了基于人工智能的工具（包括大语言模型），以支持稿件准备的各个阶段。这些工具用于语言润色、提升文本的清晰度和可读性、格式化参考文献，以及生成可视化图表。