arXiv:2611.28970v1 [cs.DB] 21 Nov 2026

信函

数据智能体设计空间：来自 KDD Cup 2026 的洞见

Boyan Li[1], Zhuowen Liang[1],Yupeng Xie[1],Xiaotian Lin[1],Xinyu Liu[1],Yizhang Zhu[1],Yuyu Luo[1]✉

1. 数据科学与分析方向，信息枢纽，香港科技大学（广州），中国广州511453

收稿日期：yyyy年mm月dd日；录用日期：yyyy年mm月dd日电子邮箱：yuyuluo@hkust-gz.edu.cn。© 高等教育出版社 2026年

关键词
数据智能体；设计空间；分析工作流；大语言模型

■ 1 引言

数据智能体使用大语言模型（LLM）来解释分析请求、与数据源交互，并执行计算工作流 [1]。推理—行动交互 [2]和可执行代码操作 [3]是这些工作流的基础。证据表示决定智能体能够计算和检查的内容，而调度与状态管理则决定工作何时继续，以及哪些结果可以复用。这些依赖关系促使我们开展系统层面的分析。

KDD Cup 2026 复杂数据分析的数据智能体 竞赛 [4] 为这一研究提供了一个具体场景，其采用了统一的评估标准和资源约束。该竞赛的评测基准 *DataSpace*将分析问题与包含结构化数据、文档和视频的异构工作空间配对，要求智能体返回完整的表格结果，以便进行确定性评测 [5]。我们分析排名最高的十支队伍的最终代码提交，以识别其实现中共有的设计选择与权衡。

为组织这些证据，我们构建了一个数据智能体设计空间，包括用于分配和控制工作的编排层；用于任务建模、证据构建、候选方案计算和结果评估的功能层；以及用于保存工作产物及其来源的状态层。在这十份参赛提交方案中，我们识别出三种反复出现的设计模式：**(I1)** 根据任务进展分配有限的计算资源；**(I2)** 通过显式表示连接计算与评估；以及**(I3)** 保存可复用产物及其来源。我们利用这些洞见概述未来方向，即协同设计和评估这三个层次，重点关注完成度、正确性和证据复用之间的权衡。

■ 2 数据智能体设计空间

■ 2.1 任务与系统模型

任务形式化。 任务是使用可访问数据与有界资源：

$$\tau = (q, R, C, B), \quad \mathcal{E} = (D, \mathcal{U}),$$
$$y = \mathcal{A}(\tau; \mathcal{E}), \quad y \models C, \quad c \preceq B. \tag{1}$$

其中， q 是问题， R 标识了 D中的可访问来源，而 C规定了任务和输出要求。预算 B 对运行的资源使用向量 c 的各分量设定上限，包括耗时和模型调用。环境的服务 U 支持来源访问、模型推理、计算和存储。在 DataSpace 中， y 是一个序列化为 CSV 的完整表格，用于确定性外部评估 [5]。

系统形式化。 如图 1所示，我们将智能体建模为

$$\mathcal{A} = (O, \mathcal{F}, W_0, I), \qquad \mathcal{F} = \{T, G, S, V\}, \tag{2}$$

其中， O 负责协调 F 中的函数， W_0 是初始工作状态， I 由请求、环境和交付接口构成。在每个决策事件 t， O 根据当前状态 W_t 和剩余预算 B_t 来选择函数调用或提交一个候选。对于函数调用决策，执行过程如下

$$\begin{aligned} a_t &= O(W_t, B_t), \\ Z_t &= \mathrm{Run}(a_t, W_t; \mathcal{E}), \\ W_{t+1} &= \mathrm{Update}_O(W_t, Z_t). \end{aligned} \tag{3}$$

动作 a_t 指定 F 中的函数、输入工件和工作者范围； Z_t 包含返回工件和观察结果。更新策略将这些输出和运行时状态记录在工作状态中。调用可以重复执行、并行运行或委派执行。提交选定的候选项 $\hat{y}_t$ 会通过交付接口 $I_{\text{out}} \in I$终止此次运行：

$$a_t = \mathrm{commit}(\hat{y}_t) \quad \Longrightarrow \quad y = I_{\text{out}}(\hat{y}_t). \tag{4}$$

■ 2.2 层与设计选择

为推导出这些层，我们按照系统职责对观察到的机制进行了分组。调度和重试控制执行，提取和计算转换信息，而缓存则保留工件。

编排层（O）。工作流拓扑将调用组织成阶段、循环、委派工作或并行分支，而资源分配则为这些工作分配模型调用、上下文容量和时间。随着函数返回观察结果和评估，反馈决策

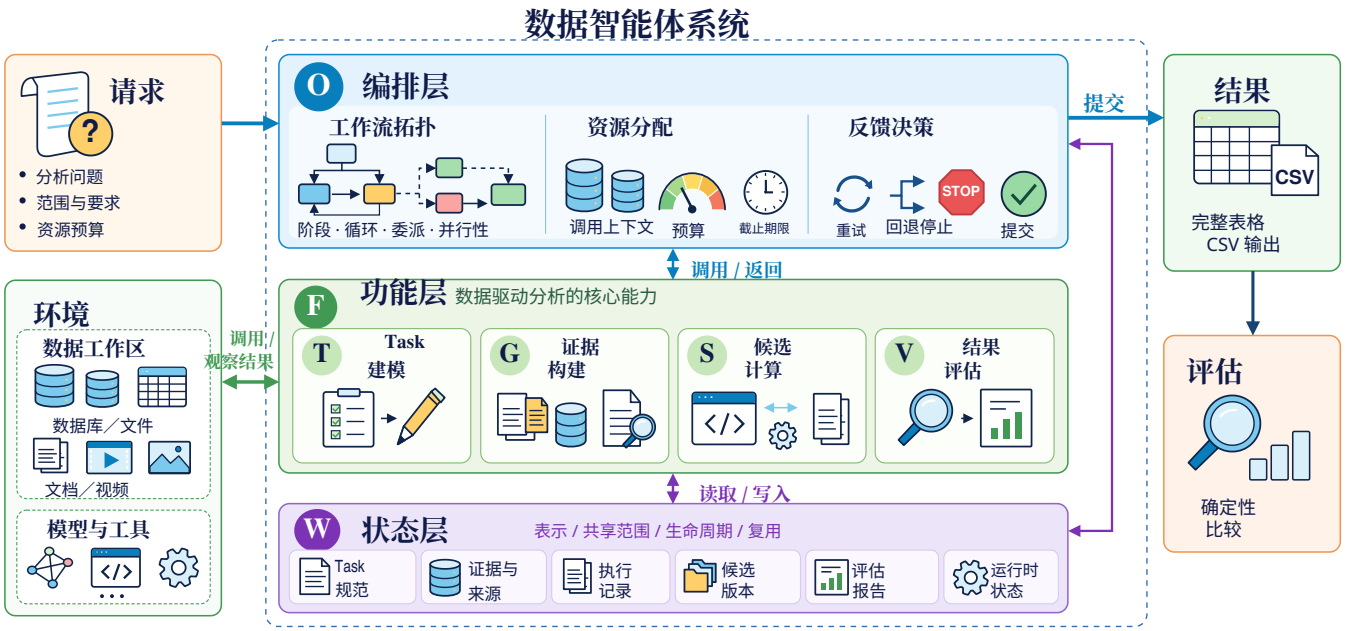

图 1 数据智能体的三层设计空间。编排通过调用和反馈组合四种并列的函数；工作状态存储这些函数产生的工件和运行时信息。各函数通过环境接口访问数据和工具，而编排则提交选定的候选方案以供外部评估。

会决定是重试、回退、停止，还是提交候选项。控制范围涵盖源文本块、工具操作、尝试次数和任务。

功能层（F）。任务建模（*T*）将请求和上下文解读为一份规范，其中包含范围、输出字段、行粒度和单位，并且这些内容的明确程度各不相同。为提供可用的表格、事实或规则，证据构建（*G*）会按选定的时间和粒度选择、提取、规范化并关联源内容。候选方案计算（*S*）利用规范和证据，通过选定的接口构建并执行解决方案，生成候选方案和执行记录。结果评估（*V*）依据适用标准检查这些工件及其支持性规范和证据，并返回评估报告，为后续编排决策提供依据。

状态层（*W* 。状态层通过保留工件及其运行时状态来衔接各次调用，包括规范、带有来源信息的证据、执行记录、候选方案版本和评估报告。这些内容在表示于对话上下文、解释器会话、文件或缓存中的方式，决定了它们如何被访问和操作。共享范围控制内容在不同工作器、尝试或任务间的可见性，而生命周期控制持久性。随后，复用策略决定哪些保留的工件可为后续调用提供信息，从而将过去的工作与后续计算和控制衔接起来。

■ 3 设计洞见

为识别反复出现的设计模式，我们对这一设计空间中的十份参赛提交方案进行比较。我们检查其代码、配置和提示词，并依据表格 1中的标准核对有歧义的情况。作为竞赛组织者，我们获准根据竞赛规则 [4]对这些参赛提交方案进行研究分析。

■ 3.1 自适应计算资源分配

在固定预算下，进一步完善方案会与完成任务所需的资源相竞争。十份参赛提交方案中有四份结合了阶段预算与有界修复（表格1）。排名第五的方案利用首轮耗时估算是否有足够预算再运行一轮，并为最终选择预留 420 秒。亚军方案的默认调度器以四个候选方案达成一致为目标，将尝试次数上限设为十次，并将未找到有效候选方案时的尝试次数限制为五次。这些策略使是否继续取决于已取得的进展和剩余资源。 **因此，计算应根据观察到的进展进行，同时保留完成任务所需的预算。**

■ 3.2 作为功能接口的表示

结构化证据将语义解释与精确计算联系起来，并使中间结果可供检查。十个参赛提交方案中，有七个将结构化证据与内容或覆盖率检查结合起来（表 1）。例如，冠军方案在汇总提取的文档表格时会检查实体覆盖率和行数。其提交的表格选择记录进一步说明了覆盖率的重要性。在从模型生成的求解器程序中提取出的参考集合非空的 33 项任务中，每项任务的平均参考表格召回率为 89.39%，而选中的候选表格所占比例中位数为 6.25%。有五项任务遗漏了参考集合中列出的表格；其中两种情况下，被遗漏的表格在五轮中均未获得任何赞成票。这些遗漏表明，即便各方意见一致，参考表格的覆盖仍可能不完整。该实现通过工具保留对未选表格的访问权限，以便重新审视源内容选择。 **表示方式既决定智能体能够执行哪些计算，也决定它能够检测出哪些错误。**

表1 排名最高的十份最终提交中的共享设计机制。

Rank	编排 O			函数 $\mathcal{F}$			状态 W		
	Task 优先级	阶段 预算	有界 修复	输出 规范	结构化 证据	证据 检查	跨尝试 复用	来源 溯源	候选 保留
1	–	–	✓	–	✓	✓	✓	✓	–
2	✓	–	✓	✓	✓	–	✓	✓	✓
3	✓	–	✓	–	✓	✓	✓	✓	✓
4	–	–	✓	–	✓	–	–	–	–
5	–	✓	✓	–	✓	✓	✓	–	✓
6	–	✓	✓	–	✓	✓	✓	–	✓
7	✓	✓	✓	–	✓	✓	–	✓	–
8	✓	–	–	✓	✓	–	✓	✓	✓
9	–	–	✓	–	✓	✓	–	✓	✓
10	–	✓	✓	✓	✓	✓	–	✓	–

排名遵循官方最终排行榜 [4]。标准遵循列顺序。**编排：** 优先处理排队中的工作；分配特定阶段的时间/步骤；限制纠正尝试次数。**函数：** 在专用工件中规定特定任务的输出要求（由代码定义或提示指定）；结构化的文档/视频证据；检查提取内容/覆盖范围。**状态：**在多次尝试之间保留已准备的证据；将工件链接至来源或生成工具；保留完整候选以供选择/恢复。

■ 3.3 工件复用与溯源

十个参赛提交方案中，有六个会在多次尝试之间保留已准备的证据，其中四个还会保留来源信息（表 1）。亚军会在不同候选方案尝试之间复用提取的文档表格，避免重复进行源材料准备。排名第七的方案会记录生成工具及其参数，以便在提交时重新生成完整结果。这类记录保留了工件的获取方式，使其推导过程可供后续检查。当候选方案复用相同证据时，也会继承其中包含的任何错误。因此，必须结合共享证据依赖来解读候选方案一致性。**状态应同时保留可复用的工作及其来源信息，以便区分额外计算与额外证据。**

■ 4 讨论与未来机遇

(O1) 这三点启示促使我们围绕额外计算的价值，对各层进行联合设计。未来的智能体可以结合工作状态中记录的证据依赖和评估结果，引导资源分配。当疑似错误源自多个候选共用的证据时，智能体可以先重新审查来源，再生成更多候选。如果证据发生变化，记录的依赖关系可以指出哪些下游结果需要重新计算。这将证据构建的粒度与修复范围联系起来，使智能体能够解决根本问题，同时保留未受影响的工作。

(O2) 这些观察到的模式促使我们开展受控比较：固定模型、任务集和总预算，同时联合改变编排策略与证据共享方式。随后，向提取的表格中注入已知错误，可以检验智能体是否能够定位错误证据并修复受影响的答案。除最终正确性外，评估还应衡量任务完成情况，以及诊断与恢复所消耗的资源。这类比较将阐明复用何时有助于自适应分配，以及何时有必要重新检查共享证据。

■ 致谢

我们感谢参加 KDD Cup 2026 复杂数据分析数据智能体竞赛的各参赛团队，感谢他们提交的参赛方案，这些方案为本研究提供了基础。本研究得到中国国家自然科学基金（62402409）、广东省科学技术协会青年科技人才支持计划（SKXRC2025461）、广州市科学技术协会青年人才支持项目（QT-2025-001）、广州市基础与应用基础研究基金（2026A1515010269、2025A04J3935、2023A1515110545）、广州—香港科技大学（广州）联合资助计划（2025A03J3714）以及异构计算互联与调度软件栈研发项目（YF202400000003）的支持。

■利益冲突

所有作者均参与了 KDD Cup 2026 复杂数据分析数据智能体竞赛的组织工作。Boyan Li 和 Yuyu Luo 担任大会主席，其他作者担任委员会主席。

■ 参考文献

[1] Zhu Y、Wang L、Yang C、Lin X、Li B、Zhou W、Liu X、Peng Z、Luo T、Li Y、Chai C、Chen C、Di S、Fan J、Sun J、Tang N、Tsung F、Wang J、Wu C、Xu Y、Zhang S、Zhang Y、Zhou X、Li G、Luo Y. 数据智能体综述：新兴范式还是被夸大的炒作? CoRR，2025，abs/2510.23587[2] Yao S、Zhao J、Yu D、Du N、Shafran I、Narasimhan K、Cao Y. ReAct：协同语言模型中的推理与行动。载于：第十一届国际学习表征会议论文集。2023[3] Wang X、Chen Y、Yuan L、Zhang Y、Li Y、Peng H、Ji H. 可执行代码操作能激发更出色的大语言模型智能体。载于：第41届国际机器学习会议论文集。2024，50208–50232[4] KDD Cup 2026。KDD Cup 2026：面向复杂数据分析的数据智能体，2026。访问日期：2026年9月16日[5] Li B、Liang Z、Xie Y、Lin X、Luo T、Liu X、Zhu Y、Peng Z、Li Y、Zhang Z、Zhang J、Tang N、Li G、Luo Y. DataSpace：异构工作空间中可验证分析的数据智能体基准测试。arXiv预印本 arXiv:2608.03451，2026