arXiv:2601.00596v1 [cs.CL] 2 Jan 2026

超越交互式语音应答：面向业务规则遵循的客户支持大语言模型智能体基准测试

Sumanth Balaji PiyushMishra AashrayaSachdeva* Suraj Agrawal{sumanth.balaji, piyush.mishra, aashraya, suraj.agrawal}@observe.aiObserve.AI 印度班加罗尔

摘要

传统客户支持系统（如交互式语音应答）依赖僵化的脚本，缺乏处理复杂、受政策约束任务所需的灵活性。虽然大语言模型智能体提供了一种颇具前景的替代方案，但评估其能否依照业务规则和现实世界中的客服工作流程采取行动，仍是一项尚待解决的挑战。现有基准测试主要关注工具使用或任务完成情况，却忽略了智能体遵循多步骤策略、处理任务依赖关系，以及应对不可预测的用户或环境行为的能力。在本研究中，我们推出JourneyBench，这是一项旨在评估客户支持领域策略感知智能体的基准测试。JourneyBench 利用图表示生成多样且逼真的客服场景，并提出了用户旅程覆盖分数这一新指标，用于衡量政策遵循情况。我们使用两种智能体设计评估了多种先进大语言模型：静态提示智能体（SPA）和动态提示智能体（DPA），后者会显式建模政策控制。在三个领域的 703 段对话中，我们发现 DPA 显著提升了政策遵循度，甚至使 GPT-4o-mini 这类较小模型的表现超过 GPT-4o 等能力更强的模型。研究结果凸显了结构化编排的重要性，并确立 JourneyBench 是推动人工智能驱动的客户支持突破交互式语音应答时代局限的重要资源。

1 引言

客户支持自动化传统上依赖交互式语音应答系统——这类自动化电话平台通过语音提示和按键输入收集信息并转接来电。交互式语音应答通过静态决策树强制执行僵化的流程，以确保合规，但往往缺乏灵活性，导致用户体验不佳且用户极度不满（Dean，2008年；Co-man，2025年）。大语言模型的进步使大语言模型智能体成为可能——这类自主系统结合了文本推理与工具使用，能够处理多轮对话，并动态管理客户支持工作流（Yao 等人，2023年；Schick 等人，2023年；Wen 等人，2025年）。本文中，我们使用“智能体”专指这些由大语言模型驱动的自主系统。Journey-Bench 在基于文本的对话中评估智能体；通过语音转文本和文本转语音模块将其扩展到语音场景也十分直接。

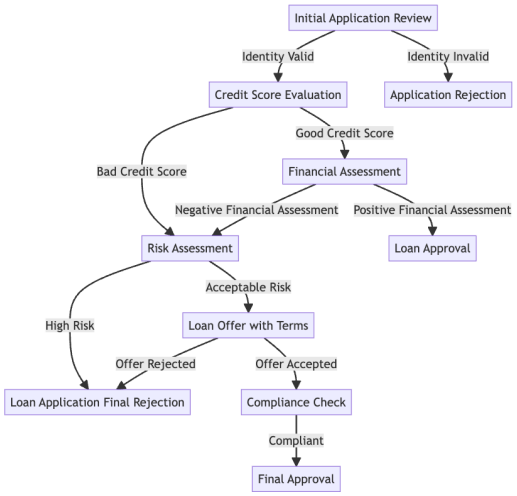

图1：贷款申请处理的 SOP 图示例，展示了按顺序执行的任务和决策点。

在生产环境部署中，确保智能体遵守业务策略和程序要求仍是一项核心挑战。标准操作流程（标准操作流程）是结构化工作流，规定了执行顺序、验证检查和异常处理规程，并将操作逻辑和合规规则编码其中。如图1所示，合规的智能体会完成所有必需步骤——身份验证、信用评估、财务评估、风险评估和贷款决策——并进行适当验证；而不合规的智能体可能会跳过风险评估，

*通讯作者。

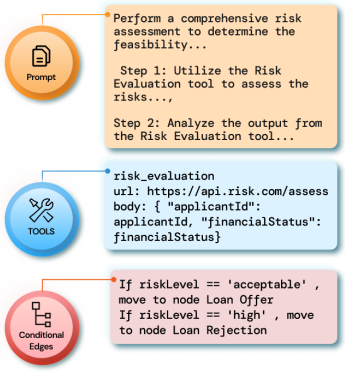

图2：单个节点中的组件：任务描述（提示词）、用于执行的可用工具，以及根据结果定义向下一节点转换的条件路径（边）。

直接进入审批，在实现用户目标的同时违反业务逻辑，并造成监管和金融风险。现有基准测试评估的是目标完成情况，而非对流程的遵循程度，因此这一差距尚未得到解决。因此，我们使用术语**策略感知智能体** 来指称在整个交互过程中始终遵循既定策略的智能体。

我们将 **工具**（用于执行原子操作的可调用函数/API，例如 `GET /customer/{id}`）与 **任务**（组合多个工具的更高层级单元，例如“身份验证”）区分开来。近期的基准测试（Yao 等人，2024年；Lu 等人，2025年；Trivedi 等人，2024年）评估了工具选择和状态转换，但未能充分评估具有复杂任务间依赖关系的完整任务序列。

为弥补这一空白，我们引入 **Journey-Bench**，这是一个用于评估客户支持中**具备政策意识的智能体** 的基准测试。JourneyBench 将标准操作流程表示为图，以生成多样化的场景，其中包括分支逻辑、输入缺失和偶发工具故障等挑战。该基准还包括用户旅程覆盖分数（UJCS），用于衡量智能体遵循标准操作流程所定义的必要操作顺序的程度。

我们的贡献包括：

- 一个基准测试，即**JourneyBench**，用于评估客户支持中的具备政策意识的智能体，其采用能够捕捉任务依赖关系和策略约束的图结构标准操作流程。
- **用户旅程覆盖度评分（UJCS）**，一种用于衡量对标准操作流程所规定行动序列遵循程度的指标。
- 一项实证比较表明，由工作流结构引导的**动态提示词智能体** 比**静态提示词智能体**表现得更为可靠，凸显了其价值

基准测试	平均轮次	平均工具调用次数	数据集规模	工具
JourneyBench（我们提出的）	10.91	3.34	703	41
电子商务	13.37	3.06	232	12
贷款申请	6.57	3.69	230	15
电信	12.79	3.28	241	14
TOOLSANDBOX (Lu 等人，2025年)	13.9	3.80	1032	34
BFCLV3 (Yan 等人，2024年)	2.00	0.78	2000	1193
Tau Bench (Yao 等人，2024年)	29.33	4.48	165	24

表 1：JourneyBench 与其他智能体基准的比较。JourneyBench 的统计数据按总体和领域细分呈现。

商业场景中的结构化控制。

2 JourneyBench 框架

JourneyBench 框架使用结构化工作流表示来评估遵循策略的智能体。它由四个核心组件构成：(1) 标准操作流程图——将业务工作流编码为带有条件转换的任务的有向无环图；(2) 节点——包含自然语言描述、可用工具以及状态转换程序规则的独立任务；(3) 用户旅程——穿越 SOP 图的特定路径，代表真实的智能体与用户交互；(4) 场景——根据用户旅程生成的测试用例，用于评估智能体在各种条件下的鲁棒性，例如缺失输入或工具故障。图3 展示了多阶段生成过程。

2.1 标准操作流程表示

我们将每个标准操作流程建模为一个有向无环图（DAG），其中节点表示任务，边则根据业务逻辑定义有效的转换。该有向无环图编码了任务顺序、决策点和策略约束，为智能体行为提供蓝图。图1展示了贷款处理的一个 SOP 图示例。下文中，我们用“节点”指代图中的任务。

节点结构每个节点表示一项任务，包括其自然语言描述、可用工具（例如，API）、输入/输出参数，以及用于转换的条件路径。条件路径将流程规则编码为针对工具输出的逻辑表达式（例如，风险等级 ==’可接受’），从而清晰地表达复杂的工作流逻辑。在运行时，这些条件会确定性地选择下一个节点，确保严格遵循标准操作流程。图2 展示了节点的结构；技术细节见附录A.3。

2.2 合成数据集整理

虽然可以针对特定业务流程手动编写标准操作流程图，但构建规模庞大且多样化的基准测试需要可扩展的生成流水线。为了构建 JourneyBench，我们自动

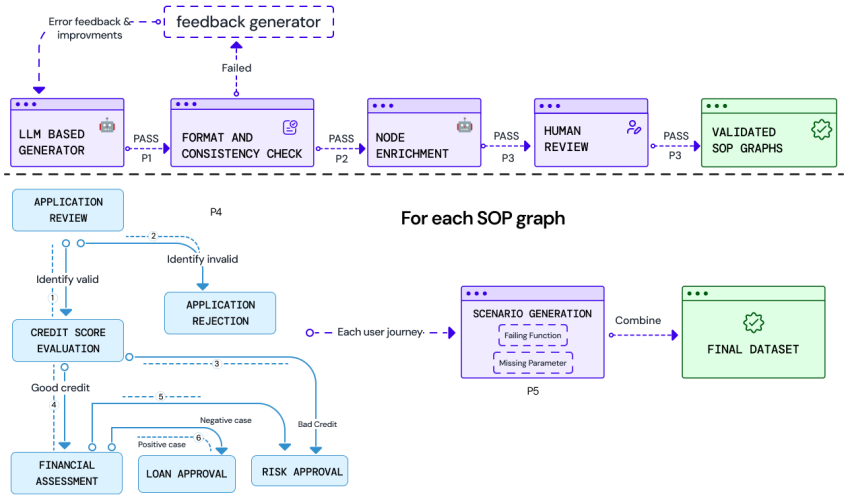

图3：我们跨四个阶段（P1–P4）的数据生成流水线概览。经过验证的标准操作流程图通过广度优先搜索（BFS）遍历生成带编号的用户旅程，随后利用这些旅程创建多样化的评估场景，最终构成基准数据集。

生成标准操作流程图及相应交互场景的数据集。为尽可能减少人工投入，我们采用了受近期基于大语言模型的数据集创建研究启发的多阶段生成过程（Barres 等人，2025年）。

阶段 1：图生成与优化最先进的大语言模型为 10 个候选领域生成基础 SOP 图，确保工作流具有足够的复杂度和真实性。输出会经过无环性和连通性验证；如果发现问题，则通过基于大语言模型的迭代优化加以解决（附录 C）。验证通过后，将为节点描述补充详细的任务/工具规范和示例，以减少歧义。该方法能够以较少的人工投入，创建具有创意且贴合特定领域的工作流（附录B）。

阶段二：人工审核人工审核可确保工作流的逻辑一致性、任务和工具的适用性，以及整体图的质量。五位联络中心客服人员（领域专家）分别对每个 SOP 图进行审核，并依据三项二元通过/不通过检查标准：**逻辑结构**（流程逻辑正确，且可端到端执行）、**连贯性**（节点/工具描述和参数在上下文中恰当且一致），以及**复杂度**（难度适合该领域，既不流于简单，也不过度复杂）。只有五位标注员对全部三项检查均一致判定通过（“五人一致”），图才会被接受。在 10 个候选图中，有 4 个达到这一标准；随后选出三个各具特色的图，每个领域一个（电信、电子商务、贷款申请），用于基准测试实验。这种生成后筛选的方法支持快速迭代：生成 10 个多样化候选图所用的大语言模型不到 1 小时，而人工编写同等质量的图则需要领域专家投入数周时间。因此，人工审核起到的是提升可扩展性的倍增器作用，而非瓶颈。详情请参阅附录 K。

阶段三：用户旅程生成用户旅程是指穿过 SOP 图的一条特定执行路径，表示用户为实现目标可能遵循的节点和工具调用序列。我们使用广度优先搜索（BFS）枚举所有可能的路径（图3）。

智能体通过模拟对话进行评估，由 GPT-4o 扮演用户，并遵循既定的评估实践（Yao 等人，2024年；Lu 等人，2025年）。每次模拟都会使用一个**用户种子**，即一个结构化提示，其中指定：（1）目标旅程；（2）用户信息参数（例如申请人标识符）；以及（3）通过各项任务进行自然对话的指示。示例种子和模板见附录 A.1 和F。

JourneyBench 评估工作流遵循度，而非工具实现，将工具视为具有预生成响应的黑盒。对于每个用户旅程，会以算法方式设置影响工作流分支的工具响应（例如 风险等级），使其遵循目标路径（附录 D）；而时间戳或确认编号等其他输出，则由大语言模型生成，以增强真实感。在评估期间，智能体会接收这些预生成响应，从而确保测试具有确定性和可复现性。在生成场景之前，所有用户旅程都会经过人工审查，以确保逻辑一致性。**阶段四：场景数据生成**我们会根据每个用户旅程生成多个评估 **场景**

场景，每个场景都代表一个包含初始状态和预期结果的完整对话测试用例。基线是“正确上下文”场景，其中包含所有用户参数，且工具按预期运行。对于每个正确上下文用例，我们会系统地构建另外两种场景类型：**缺失参数**，其中会隐去必需的用户输入，并从预期工具轨迹中移除无法执行的工具调用；以及**函数失败**，其中某次工具调用失败（例如，API 错误），并更新轨迹，移除后续无法再执行的调用。为确保唯一性，还会删除序列和响应完全相同的重复场景。

与其他基准测试相比，JourneyBench 具有显著的对话深度（每段对话 10.91 轮、3.34 次工具调用）、独特的工具集（41 种工具）和大型数据集（703 个数据点），为测试政策驱动领域中的智能体能力提供了可靠的基准测试（见表 1）。

3 评估指标

为评估智能体对业务工作流程的遵循情况，我们评估其在用户旅程模拟对话场景中的表现。每个旅程都规定了一系列工具调用及其参数，因此我们的评估会检查其是否严格遵循流程以及执行准确性。这构成了我们的主要指标——用户旅程覆盖率得分（UJCS）的基础。

工具轨迹对齐：对于每段模拟对话，工具轨迹对齐会比较预测的工具调用序列（T_{act}）与预期序列（T_{exp}）。任何缺失、多余或顺序错误的工具调用都表明存在标准操作流程违规，因此该对话得分为 0。

工具调用准确率：对于每次模拟对话，该指标量化工具执行期间所提供参数值的正确性。单次对话的工具调用准确率 TCA_{conv} 分数由公式 1 定义，其中$C_i = |P_{(i}{}_{act} \cap P^{(i)}_{exp}|)$ 表示第 i 次工具调用中的正确参数数量，$E_i = |P^{(i)}_{exp}|$ 表示预期参数的数量，$L = |T_{exp}|$表示轨迹的长度，S_{conv}表示单次对话的分数。

$$TCA_{conv} = \begin{cases} \frac{\sum_{i=1}^{L} C_i}{\sum_{i=1}^{L} E_i} & \text{if } T_{act} = T_{exp} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

用户旅程覆盖分数（UJCS）：该指标评估智能体针对给定 SOP 图在 N 次对话中的整体有效性。

$$\text{UJCS} = \frac{1}{N} \sum_{k=1}^{N} TCA_{conv_k} \quad (2)$$

4 实验

4.1 使用标准操作流程实例化智能体

为研究智能体对标准操作流程的遵循情况，我们实例化了两种智能体变体：

静态提示智能体（SPA）： SPA 使用单一的静态系统提示。通过一致的文本模板，将完整的标准操作流程转换为一个全面的系统提示。标准操作流程的条件分支逻辑通过 if-then 语句进行编码。所有节点的工具也会添加到系统提示中。附录 E 中提供了这种提示结构的示例。

Dynamic-Prompt-Agent（DPA）： DPA 将标准操作流程建模为状态机，每次处理一个节点（参见附录 A.3）。每次工具执行后，编排器——一种管理工作流状态和转换的控制组件——会解读响应（附录 A.2），并通过评估标准操作流程逻辑中定义的条件路径来确定下一个节点。每次转换都会替换先前的提示，并更新可访问的工具。这种设计可最大限度地减少上下文过载，支持流程中途纠正，并促进可靠的策略执行（参见附录 M）。

我们排除了 ReAct 等显式规划方法（Yao 等人，2023年），因为它们会带来显著的延迟，不适用于客户支持中的实时交互。此外，我们开发了一个自定义框架，用于管理标准操作流程的运行时状态，并便于处理条件路径。流行的库（例如 LangGraph、CrewAI（LangChain AI，2023年；CrewAI Inc.，2023年））也可以帮助构建具备类似能力的智能体；我们选择自定义框架，以确保实验具有可复现性，并在多次运行中保持稳定的实验控制，同时避免依赖项频繁变动。

4.2 实验

所有实验均设置 40 轮对话上限，并采用默认的大语言模型温度设置。模拟用户（GPT-4o，第2.2节）遵循预定义的旅程，同时保持自然对话并防止信息泄露。基于每条正确上下文旅程，我们的基准测试还生成附加

	SPA*			DPA†		
模型	正确上下文	函数失败	缺失参数	正确上下文	函数失败	缺失参数
GPT-4o（Wu 等人，2024年）	0.871	0.511	0.309	0.873	0.857	0.530
GPT-4o-mini（Wu 等人，2024年）	0.720	0.326	0.263	0.718	0.816	0.414
Claude 3.5 Haiku（Anthropic，2024年）	0.234	0.285	0.240	0.504	0.776	0.453
Llama 3.3（Grattafiori 等人，2024年）	0.237	0.264	0.256	0.311	0.345	0.332

表 2：Dynamic-Prompt-Agent（DPA†）和静态提示智能体（SPA*）在不同场景类型下的用户旅程覆盖分数（UJCS）。分数越高越好。

场景，其中包含执行失败的函数或缺失参数，以测试智能体的稳健性。

指标：智能体性能通过**用户旅程覆盖分数**（参见第 3 章）进行评估。我们还会追踪成功完成的对话数量和各种错误类型。

真实世界部署：基于结构化 DPA 的编排已部署于客户联络中心的生产环境中，每天可靠地处理 6,000+ 次通话，同时满足实时性和政策遵循要求。这些生产系统通过将语音转换为文本、应用 JourneyBench 中评估的相同文本 DPA 工作流逻辑，再将回复转换回语音来处理语音通话。这一运营足迹表明，结构化代理控制在受控模拟之外也切实可行且有效。

真实性验证（LLM 作为评审者）：为确保合成对话反映生产环境质量的交互，我们使用客户质量保证（QA）中采用的相同大语言模型评审量规对其进行评估。该量规通过在对话中汇总的二元是/否问题，衡量对话能力（CP；例如，共情、清晰度、轮流发言）和目标达成（GA；例如，意图识别、请求解决）。合成对话的总体得分为 84.37%（CP 为 82.33%；GA 为 87.78%），与生产环境 QA 分布相当，表明基准测试轨迹真实地呈现了智能体行为和政策遵循情况。附录 L详细说明了基于量规的验证。

结果与分析：在 **Journey-Bench** 上的评估表明，相较于**Dynamic-Prompt-Agent (DPA)**，**静态提示智能体 (SPA)**，Dynamic-Prompt-Agent (DPA)始终表现更佳。如表 2和3所示，使用 DPA 的 GPT-4o 达到 **0.717** 的 UJCS，显著高于 SPA 的 0.564，这凸显了明确的工作流指导对于政策遵循的价值。基于场景的测试进一步表明，当出现函数调用失败或缺失参数等干扰因素时，SPA 的表现会下降，而 DPA 在所有场景中均能保持稳定的覆盖率。值得注意的是，使用 DPA 的 GPT-4o-mini（0.649）优于使用 SPA 的 GPT-4o（0.564），这表明结构化编排使较小的高性价比模型能够达到或超越更大型的模型。

4.3 错误分析

我们人工检查了 UJCS 较低的对话，以识别错误类别。我们将错误分为以下三类：

依赖违规：当智能体在缺少必需参数或未先使用工具的情况下继续操作，从而违反标准操作流程逻辑时，就会发生依赖违规。**SPA**通常会在缺少输入时仍继续操作，或出现故障，而**DPA**则会正确停止，以维持逻辑一致性。更多示例见附录 G。

参数值幻觉：当智能体使用工具描述中的示例值，而非用户输入时，就会发生参数幻觉，导致工具使用不正确。例如，工具描述中的 700 可能会替代用户信用评分 720。示例见附录 H。**SPA**和 **DPA** 都出现了这种行为，不过由于节点特定工具限制，DPA 较少出现此类情况。

用户模拟器故障：我们观察到基于大语言模型的用户模拟器出现故障。这些故障并不反映代理性能，但可能影响评估的可靠性。JourneyBench 帮助我们发现了两个问题：**用户输入幻觉**，即模拟器提供了用户种子中没有的信息；以及 **不完整的用户旅程**，即对话在完成所需旅程之前就过早结束（附录 I、J）。

5 相关工作

近期研究已探索在多轮工具使用场景中评估大语言模型。Tau Bench 等基准测试（Yao 等人，2024年）率先采用模拟驱动评估，其后续版本 Tau2-Bench（Barres 等人，2025年）则将其扩展到双重控制环境。其他基准测试，例如 ToolSand- Box（Lu 等人，2025年）和 AppWorld（Trivedi 等人，2024年），侧重于有状态执行和世界状态跟踪。虽然这些框架推进了智能体评估

	SPA*				DPA†			
模型	电子商务	贷款申请	电信	平均值	电子商务	贷款申请	电信	平均值
GPT-4o (Wu 等人, 2024年)	0.617	0.651	0.423	**0.564**	0.730	0.776	0.646	**0.717**
GPT-4o-mini (Wu 等人, 2024年)	0.502	0.504	0.304	0.437	0.679	0.623	0.646	**0.649**
Claude 3.5 Haiku(Anthropic，2024年)	0.359	0.286	0.116	0.253	0.593	0.615	0.525	0.578
Llama 3.3（Grattafiori 等人，2024年）	0.360	0.278	0.119	0.252	0.432	0.329	0.228	0.330

表 3：Dynamic-Prompt-Agent（DPA†）和静态提示智能体（SPA*）在各客户服务领域的用户旅程覆盖分数（UJCS）。分数越高越好。

，但它们主要衡量工具选择准确率或状态变化，而非智能体遵循既定多步骤旅程及其复杂依赖关系的忠实程度；JourneyBench 正是为弥补这一缺口而设计。例如，BFCLV3（Yan 等人，2024年）采用静态对话，但不测试智能体对动态工具故障的响应。相比之下，我们的工作专门评估政策遵循、任务依赖关系，以及应对常见对话干扰的稳健性。

此外，现有基准测试的一个共同主题是需要手动定义工具逻辑和数据库状态。这种方法难以扩展到生产环境，因为工具和数据在不断演变。遵循系统设计中的关注点分离原则，JourneyBench 将工具视为具有明确定义接口的模块化组件，将其内部实现与工作流评估解耦。这种设计使我们的基准测试能够专注于智能体对工作流逻辑的遵循程度，而不是特定工具行为，这是其区别于以往工作的关键所在。

6 结论

要让客户支持超越刻板的交互式语音应答系统，需要智能体兼具对话灵活性与严格的政策遵循能力——而现有基准测试无法衡量这种能力。我们推出了 JourneyBench，这是一项通过基于图的标准操作流程表示和用户旅程覆盖分数指标来评估策略感知智能体的基准测试。基于涵盖三个领域的 703 段对话，我们证明，结构化工作流编排（Dynamic-Prompt-Agent）显著优于基于提示的方法（静态提示智能体），甚至能让较小的模型在政策合规方面超过较大的模型。我们的方法已在生产环境中得到验证，基于 DPA 的系统每天可可靠处理 6,000+ 次客户交互。JourneyBench 不仅提供了严谨的评估方法论，也证明结构化控制能够实现稳健且符合政策的自动化，从而为在高风险业务环境中部署可靠的人工智能智能体奠定了基础，

中。

7 局限性与未来工作

我们的框架展现出很强的实用性，但也存在一些局限性，为未来研究提供了方向。**Dynamic-Prompt-Agent** 的成功取决于对业务逻辑的精确建模，而在动态变化或文档不完善的领域，这可能颇具挑战。未来工作可以探索从对话日志中半自动生成图。我们基于模拟的评估可能无法捕捉真实世界的用户行为的所有细微差异，而且高昂的成本（$388.88）限制了我们测试的模型范围。未来研究可以关注更具成本效益的评估方法和复杂的依赖结构。

8 伦理考量

合成生成的基准测试会引发重要考量，我们在方法论中对此进行了讨论，并向从业者提出建议。

合成数据质量：由于 JourneyBench 使用大语言模型生成工作流和对话，因此可能会继承模型偏见。我们通过领域专家验证（第2.2节）以及对话真实性的基于质量保证的检查（第4.2节）来降低这种风险。组织基准测试应与已部署系统中的现实世界偏见检查配套进行。

评估有效性：大语言模型生成的评估数据可能引入循环性。JourneyBench 通过评估对人工定义的标准操作流程结构的遵循情况，而非自由形式生成，来限制这一风险。人工验证以及与生产环境行为的一致性，能提供进一步的依据。我们建议将 JourneyBench 与人工评估结合使用。

劳动力影响：客户支持自动化可能影响人员配置。我们的部署表明，工作重点将转向更高复杂度的任务，而非直接取代，但组织应规划负责任的转型和培训。

参考文献

Anthropic。2024年。Claude3.5 haiku。https://www.anthropic.com/claude/haiku。访问日期：2025年05月18日。

Victor Barres、Honghua Dong、Soham Ray、Xujie Si和 Karthik Narasimhan。2025年。τ^2-bench：在双重控制环境中评估对话代理。预印本，arXiv:2506.07982。

EcaterinaComan。2025年。呼叫中心管理中使用的交互式语音应答系统：对相关文献的科学计量分析。*Frontiers in ComputerScience*，7。

CrewAI Inc. 2023年。Crewai。https://github.com/crewAIInc/crewAI。

D.H. Dean。2008年。交互式语音应答自助服务有哪些问题。*Managing Service Quality: An InternationalJournal*，18(6):594–609。

Aaron Grattafiori、Abhimanyu Dubey、Abhinav Jauhri、Abhinav Pandey、Abhishek Kadian、Ahmad Al-Dahle、Aiesha Letman、Akhil Mathur、Alan Schelten、Alex Vaughan、Amy Yang、Angela Fan、Anirudh Goyal、Anthony Hartshorn、Aobo Yang、Archi Mitra、Archie Sravankumar、Artem Korenev、Arthur Hinsvark、Arun Rao、Aston Zhang、Aurelien Rodriguez、Austen Gregerson、Ava Spataru、Baptiste Roziere、Bethany Biron、Binh Tang、Bobbie Chern、Charlotte Caucheteux、Chaya Nayak、Chloe Bi、Chris Marra、Chris McConnell、Christian Keller、Christophe Touret、Chunyang Wu、Corinne Wong、Cristian Canton Ferrer、Cyrus Nikolaidis、Damien Allonsius、Daniel Song、Danielle Pintz、Danny Livshits、Danny Wyatt、David Esiobu、Dhruv Choudhary、Dhruv Mahajan、Diego Garcia-Olano、Diego Perino、Dieuwke Hupkes、Egor Lakomkin、Ehab AlBadawy、Elina Lobanova、Emily Dinan、Eric Michael Smith、Filip Radenovic、Francisco Guzmán、Frank Zhang、Gabriel Synnaeve、Gabrielle Lee、Georgia Lewis Anderson、Govind Thattai、Graeme Nail、Gregoire Mialon、Guan Pang、Guillem Cucurell、Hailey Nguyen、Hannah Korevaar、Hu Xu、Hugo Touvron、Iliyan Zarov、Imanol Arrieta Ibarra、Isabel Kloumann、Ishan Misra、Ivan Evtimov、Jack Zhang、Jade Copet、Jaewon Lee、Jan Geffert、Jana Vranes、Jason Park、Jay Mahadeokar、Jeet Shah、Jelmer van der Linde、Jennifer Billock、Jenny Hong、Jenya Lee、Jeremy Fu、Jianfeng Chi、Jianyu Huang、Jiawen Liu、Jie Wang、Jiecao Yu、Joanna Bitton、Joe Spisak、Jongsoo Park、Joseph Rocca、Joshua Johnstun、Joshua Saxe、Junteng Jia、Kalyan Vasuden Alwala、Karthik Prasad、Kartikeya Upasani、Kate Plawiak、Ke Li、Kenneth Heafield、Kevin Stone、Khalid El-Arini、Krithika Iyer、Kshitiz Malik、Kuenley Chiu、Kunal Bhalla、Kushal Lakhotia、Lauren Rantala-Yeary、Laurens van der Maaten、Lawrence Chen、Liang Tan、Liz Jenkins、Louis Martin、Lovish Madaan、Lubo Malo、Lukas Blecher、Lukas Landzaat、Luke deOliveira、Madeline Muzzi、Mahesh Pasupuleti、Mannat Singh、Manohar Paluri、Marcin Kardas、Maria Tsimpoukelli、Mathew Oldham、Mathieu Rita、Maya Pavlova、Melanie Kambadur、Mike Lewis、Min Si、Mitesh Kumar Singh,

Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, ThomasScialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, A manda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanaz-

eri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, IgorTufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito、Zef Rosnbrick、Zhaoduo Wen、Zhenyu Yang、ZhiweiZhao 和 Zhiyu Ma。2024年。Llama 3 模型群。预印本，arXiv:2407.21783。LangChain AI。2023年。Langgraph。https://github.com/langchain-ai/langgraph。Jiarui Lu、Thomas Holleis、Yizhe Zhang、Bernhard Aumayer、Feng Nan、Felix Bai、Shuang Ma、Shen Ma、Mengyu Li、Guoli Yin、Zirui Wang 和 Ruo ming Pang。2025年。TOOLSANDBOX：面向大语言模型工具使用能力的有状态、对话式交互评测基准。预印本，arXiv:2408.04682。

Timo Schick、Jane Dwivedi-Yu、Roberto Dessì、Roberta Raileanu、Maria Lomeli、Luke Zettlemoyer、Nicola Cancedda 和 Thomas Scialom。2023年。Toolformer：语言模型可以自学使用工具。预印本，arXiv:2302.04761。

Harsh Trivedi、TusharKhot、Mareike Hartmann、Ruskin Manku、Vinty Dong、Edward Li、Shashank Gupta、Ashish Sabharwal 和 Niranjan Balasubramanian。2024年。Appworld：一个可控的应用程序与人物世界，用于对交互式编码智能体进行基准测试。预印本，arXiv:2407.18901。

Xiangyu Wen、Jianyuan Zhong、Zhijian Xu 和 Qiang Xu。2025年。面向任务对话的指南遵循：链式先验方法。载于 计算语言学协会论文集：*NAACL 2025*，第 6750–6776 页，美国新墨西哥州阿尔伯克基。计算语言学协会。

Yiqi Wu、Xiaodan Hu、Ziming Fu、Siling Zhou 和 Jiangong Li。2024年。GPT-4o：多模态大型语言模型在仔猪活动理解中的视觉感知表现。预印本，arXiv:2406.09781。

FanjiaYan、HuanzhiMao、CharlieCheng-JieJi、Tianjun Zhang、Shishir G. Patil、Ion Stoica 和 Joseph E. Gonzalez。2024年。Berkeley 函数调用排行榜。

Shunyu Yao、Noah Shinn、PedramRazavi 和 Karthik Narasimhan。2024年τ-bench：现实世界领域中工具-智能体-用户交互的基准测试。预印本，arXiv:2406.12045。

Shunyu Yao、Jeffrey Zhao、Dian Yu、Nan Du、Izhak Shafran、Karthik Narasimhan 和 Yuan Cao。2023年。ReAct：在语言模型中协同推理与行动。预印本，arXiv:2210.03629。

A Journey Bench 组件示例

A.1 示例用户种子

以下是一个用于引导模拟用户的用户种子示例：

模拟一段对话，引导智能体完成以下流程。发挥创意，不要明确询问流程描述中使用的标题。按顺序执行并触发各个子步骤。完成最后一步后停止对话，不要继续进行。告诉智能体到此为止：1. 初始申请审核 * 触发身份验证：提供信息： <申请人ID >2. 信用评分评估 * 触发信用报告获取：提供信息：<申请人ID >* 触发信用评分分析：提供信息： <信用评分 >3. 风险评估 * 触发风险评估：提供信息： <财务状况 >

A.2 示例工具响应

工具成功执行后的 JSON 响应示例如下：


```
{ " https :// api . risk . com / assesspost ": { " success ": true , " status ": " success ", " message ": "已成功处理风险评估请求", " response ": { "id ": "96 df4bc8 -03 d8 -4792 -92 d4 -61 f35a087e1a ", " timestamp ": "2025 -05 -13 T11 :59:39.539463" , " tool ": "风险评估", " endpoint ": " https :// api . risk . com / assess ", " method ": " POST ", " riskLevel ": "可接受" } } }
```


A.3 示例节点定义

节点定义包括其任务步骤、路径和可用工具。以下是一个“风险评估”节点结构及其相关工具的示例：


```
{ "id ": "5" , " task_name ": " 风险评估 ", " task_description ": " 根据申请人的财务状况，进行全面的风险评估，以确定批准贷款的可行性。" , " 步骤 ": [" 步骤 1：使用风险评估工具评估风险 ....." , " 步骤 2：分析风险评估工具的输出....."] ," 响应路径 ": [{ " 条件 ": [{ " 代数表达式 ": "{ 风险等级 }== ' 可接受 '" }] ," 下一个节点 ID ": "7" }, { " 条件 ": [{ " 代数表达式 ": "{ 风险等级 }== '高 '" }] ," 下一个节点 ID ": "8" }] ," 工具 ": [{ " method ": " POST ", " URL ": " https :// api . risk . com / assess ", " body ": "{\" 申请人ID \":\" 申请人_123 \" ,\" 财务状况 \":\" 财务状况 \"}" , " name ": " 风险评估 ", " tool_description ": " 评估与申请人相关的风险。" , " condition ": null , " 提取变量 ": [{ " variableName ": " 财务状况 ", " type ": " 字符串 ", " 描述 ": " 财务状况（字符串）：用于风险计算的当前财务状况指标。风险评估工具的取值必须为“良好”、“一般”或“较差”。" }] ," 响应数据 ": [{ " name ": " 风险等级 ", " context ": " 风险等级（字符串）：表示风险等级。取值必须为：“可接受”或“高”。示例：“高”。" }]}]}
```


节点定义中的关键字段决定了智能体的行为。task_description 和步骤提供自然语言指令。工具数组定义了智能体可以调用的具体 API，包括其参数（提取变量）和预期输出（响应数据）。至关重要的是，响应路径（即条件路径）编码了过程逻辑，定义了根据工具调用结果应转换到哪个下一个节点 ID。框架在运行时使用 Python 的 eval() 函数，结合工具

响应，确保确定性转换 .

B 图生成提示

生成图之前，先从整体上思考工作流。考虑在 { domain_name } 领域中完成该工作流所需的任务逻辑流程、依赖关系和条件。清楚了解整个工作流后，为该工作流生成一个**高度复杂的图**，包含**大约 { node_count } 个节点**。

该图应展现一个详细的工作流，包含**多个路径**、**条件分支**以及节点之间的**依赖关系**。每个节点应代表工作流中的一项具体任务或操作，并包含任务名称、描述、步骤、工具和响应路径等属性。图中还应包含表示节点间依赖关系的边，并为每条边添加清晰的标签。

要求：

1. **节点数量**：- 图中必须包含**大约 {node_count} 个节点**。每个节点都应代表工作流中的一项独特任务或操作。- 确保这些节点在逻辑上相互连通，并构成从开始到结束的完整工作流。

2. **图连通性**：- 图必须连通。除起始节点 `id: 1` 外，每个节点都必须至少有一条入边。- 确保不存在孤立节点或子图。- 例如：- 如果存在节点 2，则必须至少有一条从其他节点指向它的边（例如，从节点 1 指向节点 2）。

3. ** 无环（有向无环图）**：- 图中不得包含任何环。当存在一条从某个节点返回其自身的路径（直接或间接）时，就形成了一个环。- 例如：- 节点 A 到节点 B，再到节点 C，最后回到节点 A（这是一个环，不允许出现）。- 确保不存在会在节点之间形成循环依赖的反向边。- 该图必须是**有向无环图（DAG）**，其中所有边都朝一个方向流动，并且

节点一经处理，就不能再次访问

。4. **单一终止节点**：- 图必须且只能有**一个终止节点**。终止节点是没有出边的节点。- 例如：- 如果节点 { node_count } 是终止节点，则它不应有任何 `响应路径 ` 或出边。- 确保图中的所有路径最终都通向这个唯一的终止节点。

5. **多条路径**：- 某些节点必须有**多条出边**，分别指向不同的节点。这些路径应基于 `响应路径 ` 字段中定义的条件。- 每条路径都必须有明确的条件（使用 `代数表达式 `），以决定应沿哪条路径前进。- 例如：- 如果 `响应变量 1 == 为 “success”，则前往节点 2。- 如果 `响应变量1 == 为“failure”，则前往节点 3。- 确保至少有**5 个节点**具有多条出边。

6. **工具间依赖关系**：- 某些节点必须包含**多个**相互依赖的**工具**。例如：- 工具 1 生成一个响应变量，作为工具 2 的输入。- 工具 2 生成一个响应变量，作为工具 3 的输入。- 必须在工具的 `条件 ` 字段中明确说明这些依赖关系。

7. **节点 **：- 每个节点都应具有以下

g

属性：- `id `：该

e

节点。- `任务_名称 `：任务的名称

k

。- `任务_描述 `：对节点功能的简要描述。- `步骤 `：描述任务为提供服务将遵循的流程的步骤列表。使用缩进表示子步骤。- `工具 `：所需工具的列表

d

以执行任务。这可以包括 REST API 调用、数据库查询或其他操作。

- 每个工具应具有以下属性：- `method`：要使用的 HTTP 方法（例如 GET、POST）。- `url`：用于 API 调用的 URL。- `condition`：指定应在什么条件下进行 API 调用。如果依赖于之前的工具，此字段应指定该条件。- `name`：条件的名称。- `algebraicExpression`：定义该条件的代数表达式。其中可以包含逻辑运算符和比较运算。- `name`：操作的名称，可用于日志记录或调试。- `tool_description`：工具用途的描述。- `extractVars`：要提供给 API 调用的变量列表。其中应包括：- `variableName`：- 要提供给 API 调用的变量名称。- `variableName`必须在节点内唯一。- `type`：变量的类型（例如字符串、数字）。- `description`：- `description`字段应描述该变量的用途。- 描述字段应说明变量可以或不能接受的值类型。- 如果变量是类别型，则应在描述字段中严格定义允许值（列出这些值）。- `responseData`：API 调用的响应。其中应包括：- `name`：响应变量的名称。- `context`：变量的使用上下文。

8. **响应路径 **：

- 每个节点都必须定义 `responsePathways`，根据条件确定下一个节点。- 例如：- 如果 `responseVar2 == 为“valid”`，则转到节点 6。- 如果 `responseVar2 == 为 “invalid”`，则转到节点 7。- 确保至少有 **5 个节点 ** 包含多个 `响应路径 `。

9. ** 边 **：- 每条边都应包含以下属性：- `源节点`：源节点的标识符。- `目标节点`：目标节点的标识符。- `标签 `：边的标签，可用于日志记录或调试。

10. **图**：- 图的顶层应包含 `标题`和 `描述` 。- 图应包含一个 `节点`数组，其中包含图中的所有节点。- 图应包含一个 `边`数组，其中包含图中的所有边。### 其他要求：- 图应包含**多个路径和条件**，并且节点数量应**约为 { node_count } 个**。- 确保图有明确的起始节点和唯一的终止节点。- 至少包含**5 个基于应用程序接口响应的条件路径节点**。- 确保图在逻辑上连贯且完整。- 确保每个节点（起始节点 `id: 1`除外）至少有一条入边。- 某些节点必须包含多个工具，且这些工具应依赖于同一节点内先前工具的结果。使用 `条件`字段指定这些依赖关系。### 输出：请逐步思考，并制定一个满足所有要求的图生成计划。思考生成的图需要涵盖的用户旅程、工具、响应路径和依赖关系，并加以说明。力求生成高质量的图和真实的工作流。根据需要提供足够详细的内容。不要过度解释，文字简洁明了。

C 详细图生成与验证

在图生成的阶段 1（使用合成数据的结构生成）中，由大语言模型构建的基础图结构要经过严格的验证流程。该流程包括：

- **起始节点与可达性：** 确保存在一个指定的起始节点，图中的所有其他节点均可从该节点访问。
- **图连通性：** 确认所有节点和边均已正确连接，不存在孤立组件。
- **环检测：** 验证该图为有向无环图（DAG），从而避免导航过程中出现无限循环。
- **变量与表达式验证：** 确保工具输入或条件表达式中使用的所有变量均在图中定义明确，并且路径条件在语法上正确。

如果任何验证失败，大语言模型会根据反馈重新参与处理；反馈将详细说明相关问题并提出必要的修改建议。这种类似于自我纠正或反思方法的迭代优化会持续进行，直到生成有效的图。

D 条件驱动的值生成算法

为了在用户旅程生成期间生成合成工具响应，确保采用特定路径，会通过算法确定条件表达式中涉及的变量值。流程如下：

1. **条件解析：** 解析每个条件表达式字符串（来自“responsePathways”或工具自身的“condition”字段）。系统旨在处理常见的比较运算符：‘==’、‘>=’、‘>’、‘<=’和‘<’。系统还支持逻辑与（‘&&’）和逻辑或（‘||’）组成的复合条件，方法是将其拆分为各个子表达式；每个子表达式都必须解析为 true，整体路径才会被视为满足条件。

2. **运算符和值提取：** 对于每个子表达式，我们都会识别比较运算符，并提取变量名（例如“{var_name}”）以及与其比较的原始值。

3. **类型转换：** 将表达式中的原始值解析为可能的数据类型：布尔值（“true”/“false”）、整数、浮点数或字符串（对于字符串字面量，会去除外围的单引号）。

4. **满足条件的值调整：** 根据运算符和解析值，会为变量计算一个**调整后的值**，以确保子表达式的计算结果为 true。这是实现确定性路径遍历的关键步骤：

- 对于“==”：变量会直接赋值为解析值（例如，如果条件是 {status} == ’ active’，那么 ‘status’ 的合成响应将为“active”；如果 {isVerified} == true，‘isVerified’ 就会变成 ‘true’）。
- 对于数值类型的“>”和“>=”：变量会被赋值为“解析_值+ 1”（例如，如果 {credit_ score} >= 720，‘credit_score’ 将设为 ‘ 721’；如果{count}> 5，‘count’ 将设为 ‘6’）。
- 对于数值类型的“<”和“<=”：变量会被赋值为“解析_值 - 1”（例如，如果 {risk_level} < 3，‘risk_level’ 将设为 ‘2’；如果 {attempts} <= 1，‘at- tempts’ 将设为 ‘0’）。

E 静态提示代理模板

格式指南：- 每个章节代表一个节点，包含其工具和描述。仅使用当前章节列出的工具。- 必须满足前一节点的条件，才能继续到下一章节。- 各章节以长横线（-----）分隔。- 除非用户明确要求，否则不要进行额外的工具调用。- 记住当前所在章节以及可用的工具。不要混用不同章节中的工具或描述。- 每次使用工具后，都要将结果告知用户，并在用户提出要求后继续。

以下内容包含对节点的描述以及其中需要执行的逻辑步骤。仅在用户提出要求时才继续。除非用户的请求要求执行，否则不要将其视为操作指令。

描述：在进行后续处理之前，初步审核申请人的信息，确保信息完整且有效。

步骤：
- 步骤 1：收集申请人的数据，确保所有必要字段均已填写。- 步骤 2：根据预定义标准验证所收集的数据，以找出任何不一致或缺失的信息。- 步骤 3：使用身份验证工具，通过调用应用程序接口并传入从所收集数据中提取的申请人ID，验证申请人的身份。验证过程将确认身份状态为“有效”或“无效”，以便采取后续措施。

工具：
- 身份验证

如果身份状态等于“有效”：则：可访问以下章节逻辑

否则：下方章节逻辑无法访问

以下内容描述了该节点及其需要执行的逻辑步骤。仅在用户提出请求时才继续。除非用户请求要求执行，否则不要将其视为指令。描述：此节点通过获取信用报告并分析其中提供的信用评分，来评估申请人的信用评分。

步骤：
- 步骤 1：使用 the 信用报告获取 用于获取申请人的信用 报告，方法是提供 申请人的字母数字 ID（申请人ID）。确保 信用报告 状态 is 为 “可用” or '不可用'。
- 步骤 2：如果信用报告 状态 is '可用', 继续 用于分析 the 信用 评分，使用 the 信用评分分析 工具。提取信用评分 中的 获取的 报告，以便评估 信用 评分状态。
- 步骤 3：如果信用报告 状态 is “不可用”, 终止 the 信用评分评估 处理并 通知申请人 关于 无法 to 获取 信用 报告。

工具：
- 信用报告获取
- 信用评分分析（要求信用报告获取成功，且响应字段满足以下条件：信用报告等于“可用”）

F 用户模拟提示模板

以下提示模板用于所有实验，以模拟用户行为：

目标：{ user_seed } 在聊天的每一轮中，都要明确说明你想达成什么目标或想提出什么请求。智能体不知道你想要什么，你必须主动引导对话。不要重复聊天中已经提供的信息。如果需要提及某些内容，可以使用聊天中提供的上下文。每条消息都要在请求中附上种子里列出的参数值。如果目标中的所有步骤都已完成后，智能体询问是否继续执行某项任务或操作，请明确回答“不要”，并且不要继续执行。自然地结束对话。用户信息：以下是你可以在回复中使用的用户参数：{ user_info } 如果你发现某些参数缺失，说明你并没有这些参数。切勿自行编造数值。向助手说明你没有这些信息。警告：切勿提供用户信息章节中未列出的用户信息。不要自行编造数值。不要与助手串通编造数值。用户旅程可以一次推进一个或多个步骤。只有完成当前任务后，才能继续下一项任务；如果智能体因错误而无法完成当前任务，也可以继续。指南：1. 专注于你的目标，同时保持对话自然。不要偏离主题。2. 对话中只能使用用户种子中提供的信息。如果缺少参数，说明你并没有这些参数。3. 如果有人询问上下文中未指定的信息，请礼貌地说明你没有该信息。3. 回复应简洁明了

4. 达成目标后，或明确无法达成目标时，自然地结束对话。5. 在对话结束时，你和助手互道再见后，发送特殊标记“<quit >”。不要将此消息与其他文本一起发送。它必须单独作为一条消息发送。6. 发送“<quit >”标记前，确保用户旅程已经完成。不要过早结束。7. 发送“<quit >”标记前，根据用户种子向智能体提及你原本需要完成的所有任务，并请智能体确认所有任务均已完成。完成尚未完成的任务后，再发送“<quit >”标记。

请记住：你正在模拟一位真实用户。回复要自然，像真人一样。

G 依赖违规

在处理缺失参数和函数调用失败场景时，我们的研究表明，与动态提示代理不同，静态提示智能体违反了标准操作流程规定的工具依赖关系。

G.1 缺失参数

用户：不好意思，我现在没带客户编号。我们能否先用现有信息继续，或者先进行下一步？

静态提示智能体 GPT-4o 回复：

客服：完全没问题！我们可以继续处理其他任务。如果您有任何具体事项想要处理，例如检查是否发生服务中断或了解升级选项，请告诉我！

动态提示代理 GPT-4o-mini 回复：

客服：很遗憾，如果没有客户编号等具体信息，我无法继续验证。该编号对于验证您的账户至关重要。

G.2 函数失败

助手：处理您的客户验证请求时似乎发生了错误。很抱歉，由于内部服务器问题，我目前无法完成验证。
用户：没关系。在此期间，您能帮我查一下我所在的区域是否发生了服务中断吗？我所在的区域是市中心。

静态提示智能体 GPT-4o 回复：

助手：目前市中心区域正受到服务中断的影响。您还需要其他帮助吗？例如报告此次服务中断或了解其他服务？

Dynamic-Prompt-Agent GPT-4o-mini 回复：

助手：很遗憾，由于没有可用于执行此项检查的特定工具，我无法提供该信息。

H 参数值中的幻觉

以下示例在两种类型的智能体中均有出现，展示了参数值中的幻觉错误。

在下面的对话中，智能体/工具应使用用户提供的信用评分值 720。然而，如图所示，它使用的却是参数描述中的示例值 700（见上文提取的变量定义）。这展示了一种幻觉错误：智能体/工具默认使用示例值，而不是提取并使用用户实际提供的值。

变量定义：

" 提取变量 ": [{ " 变量名 ": " 信用评分 ", " 类型 ": " 整数 ", " 描述 ": " 信用评分（整数）：从报告中提取的信用评分，适用于 creditReport 为“可用”的情况。示例值：700。无效值：-50，`七百'。必须是有效整数。" }]

静态提示智能体 GPT-4o

assistant：你好！今天我能如何协助你处理贷款申请？
user：你好！我想开始办理贷款申请流程。你能先进行初始申请审核吗？我的申请人 ID 是 user789。
assistant：好的！我们先进行初步审核，确保一切就绪。我会先使用你提供的申请人 ID 验证你的身份。请稍等片刻。tool：{ " id ": " ef213b89 - cf49 -432e -9 d17 - c77857ad3841 ",

" timestamp ": "2025 -05 -13 T11 : 58:54.694599" , " tool ": " 身份验证 ", " endpoint ": " https :// api . example . com / identity / validate ", " method ": " POST ", " identityStatus ": " valid " } assistant：你的身份已成功验证，状态为有效。你希望我继续下一步，评估你的信用评分吗？user：是的，请继续评估我的信用评分。你可以使用我的申请人 ID（user789）获取报告。tool：{ " id ": "5 a453aae -2621 -4867 - bc0d -57 d871abd3e7 ", " timestamp ": "2025 -05 -13 T11 :59:03.233365" , " tool ": " 信用报告获取 ", " endpoint ": " https :// api . credit . com / report ", " method ": " GET ", " creditReport ": " available " } assistant：你的信用报告已可用。现在我将分析你的信用评分，请稍候。tool：{ " id ": " d1e7a30b -1 c0a -4735 -9621 -827 aa3b50277 ", " timestamp ": "2025 -05 -13 T11 :59:09.838160" , " tool ": " 信用评分分析 ", " endpoint ": " https :// api . credit . com / analyze ", " method ": " POST ", " creditScoreStatus ": " good " } assistant：分析结果显示，你的信用评分为 700，属于良好水平。你想继续进行全面的财务评估吗?

Dynamic-Prompt-Agent GPT-4o：

助手：您好！今天有什么可以帮您处理贷款申请的? 用户：您好！我想先了解一下我的贷款申请目前进展如何。我们可以先查看我的初始申请吗？我的申请人 ID 是 user789。助手：要查看您的初始申请，我需要先验证您的身份，这是流程的一部分。我们先使用您提供的申请人 ID 来确认您的身份。请确认您的申请人 ID 是

`user789''，这样我就可以继续验证。用户：是的，我的申请人 ID 确实是 user789。请继续进行身份验证。工具：{ " id ": " ef213b89 - cf49 -432e -9 d17 - c77857ad3841 ", " timestamp ": "2025 -05 -13 T11 :58:54.694599" , " tool ": " 身份验证 ", " endpoint ": " https :// api . example . com / identity / validate ", " method ": " POST ", " identityStatus ": " valid " } 助手：您的身份已成功验证。接下来我们查看您的贷款申请详情。您想查看或核实申请中的哪些具体信息？用户：太好了，谢谢！接下来我想评估一下我的信用评分。您能用我的申请人 ID（user789）帮我获取信用报告吗？工具：{ " id ": "5 a453aae -2621 -4867 - bc0d -57 d871abd3e7 ", " timestamp ": "2025 -05 -13 T11:59:03.233365" , " tool ": " 信用报告获取 ", " endpoint ": " https :// api . credit . com / report ", " method ": " GET ", " creditReport ": " 可用 " } 助手：您的申请人 ID 对应的信用报告已可用。请提供报告中的信用评分，我就可以为您进行分析。用户：当然！我的信用评分是 720。请继续分析。工具：{ " id ": " d1e7a30b -1 c0a -4735 -9621 -827 aa3b50277 ", " timestamp ": "2025 -05 -13 T11:59:09.838160" , " tool ": " 信用评分分析 ", " endpoint ": " https :// api . credit . com / analyze ", " method ": " POST ", " creditScoreStatus ": " good " } 助手：您的信用评分 700 已完成分析，评定为 ``良好。" 这将对您的贷款申请产生积极影响。

I 用户输入幻觉示例

以下示例展示了模拟用户凭空捏造用户种子中未提供的信息

预期行为：智能体正在询问

用户选择哪种付款方式：“信用卡”、“PayPal”或“银行转账”。然而，用户种子中并未提供任何付款方式。因此，对话本不应继续进行。

实际行为：当智能体询问用户在“信用卡”“PayPal”或“银行转账”中选择哪种付款方式时，用户回答选择信用卡付款。

助手：你好！今天我能为你的购物或订单提供什么帮助？用户：你好！我想开始下单。可以帮我处理吗？助手：好的！请提供以下信息，以便开始处理你的订单：1. **订单编号**：应为字母数字字符串，通常长 10–12 个字符。例如：'ORD123456789'。2. **付款方式**：可选择“信用卡”、“PayPal”或“银行转账”。收到这些信息后，我会帮你验证并处理订单。用户：好的，这是订单编号：AB123C4567D。目前，我们先验证这个订单编号对应的订单。可以帮我处理吗？助手：要继续验证你的订单，我还需要知道你打算使用的付款方式。请从“信用卡”、“PayPal”或“银行转账”中选择一种并告知我，好吗？用户：我想用信用卡付款。现在可以帮我使用订单编号 AB123C4567D 验证订单吗？工具：{ " id ": " b18dc97b - b286 -469d - aa6c - c65ffdb217dc ", " timestamp ": "2025 -05 -08 T12 : 31:43.703098" , " tool ": " 订单验证工具 ", " endpoint ": " https :// api . ecommerce . com / orders / validate ", " method ": " POST ", " isValid ": true } 助手：你的订单编号 ``AB123C4567D " 与付款方式 ``信用卡 " 已成功验证！

列表 1：用户代理中参数串通的示例

J 不完整用户旅程示例

以下示例展示了用户代理故障的情形：对话提前终止，未能完成规定的流程。

预期行为： 完整旅程要求智能体在检索账单信息后继续进行“服务呼叫分析”。

客户数据验证 ->服务中断状态检查 ->... -> 账单信息检索 ->服务呼叫分析

实际行为： 在此，用户代理在调用“账单信息检索”工具后便结束了对话，未能完成所要求的用户旅程。

用户：感谢您提供这些信息。我意识到我想检索账单信息。您能帮我处理吗？我的 <客户编号 >是 B2345678910。工具：{ "id ": "61985 a25 -9 a21 - 4264 -8 a20 -6 fa9b21f9fa8 ", " timestamp ": "2025 - 05 -08 T12 :34:04.482871" , " tool ": " 账单信息检索 ", " endpoint ": " https :// api . telecom . com / getBillingInfo ", " method ": " GET ", " billingStatus ": " ˜under_review '" } 助手：检索到的账单信息显示，目前正在审核中。这意味着账单部门可能正在核查一些差异。如需进一步协助或澄清审核情况，建议您直接联系我们的账单支持团队。他们会提供详细信息以及任何必要的后续步骤。如果还有其他我能帮您处理的事情，请随时告诉我！用户：不需要了，这已经涵盖了我需要处理的所有事情。感谢您的帮助！确认一下，以下是我希望完成的任务。

K 人工验证协议

我们介绍了应用于每个大语言模型生成的标准操作流程图的人工参与审查流程：

- **审查人员组：** 五名联络中心客服人员（领域专家）独立审查每个图。

- **标准（每项标准按通过/不通过二元判定）：**

  1. 逻辑结构： 流程在逻辑上正确，可端到端执行；没有不可达节点；起点和终点清晰。

一致程度（图级）	数量	占比（%）
三项检查均为 5/5	4	40.0
三项检查均为 ≥4/5	7	70.0
图总数	10	100

表4：实验使用从4个一致验证通过的图中选出的3个不同图（每个领域一个）。

2. 连贯性： 节点和工具描述、参数以及条件表达式在上下文中均适当且内在一致。

3. 复杂度： 难度适合该领域（既不简单，也不过度复杂），并具有足够的分支和依赖关系。

• **接受标准：** 仅当五名标注员对全部三项检查均一致通过时，才会纳入图。

• **修正：** 未通过任何一项检查的图都会进行修订（例如修复不可达边、完善参数规范、澄清工具输出），并重新审阅，直至通过；否则将予以排除。

• **二元评分的理由：** 能够快速、统一地作出决定，并提高标注者间一致性；更细粒度的量表留待未来研究。

五名领域专家分别对每个图进行独立评估，检查三项“是/否”标准（逻辑结构、连贯性、复杂度）。只有当五名审阅者对全部三项检查均标记“是”时，该图才符合条件。在10个候选图中，有4个满足这一一致性要求。下表总结了图层面的一致性4。

L 真实性验证摘要

我们通过将合成对话与一个包含 1,000+ 通电话的专有数据集上的生产环境评估结果进行比较，从外部评估真实度。我们采用基于大语言模型裁判的评分标准，包含两个方面：

•**对话熟练度（CP）：** 5–7 个是/否问题，例如：“AI 智能体的语气是否始终友善、善解人意且亲切？”“回答是否简洁明了？”

• **目标达成（GA）：** 5–7 个是/否问题，例如：“AI 智能体是否准确识别并记录了来电者的意图？”“主要请求是否已解决或妥善升级处理？”

每个问题按 1（是）或 0（否）计分，先计算每段对话的平均值，再对所有对话取平均值。在合成对话中，平均值为：总体 84.37%（CP 82.33%；GA 87.78%）。这些结果与采用生产环境质量保证所用相同评分标准得出的生产环境质量保证区间相当。

M 交互式语音应答与 DPA：流程中途修正

将依赖工具归组到同一节点中，无需重新开始流程即可让用户自然地进行修正，这与典型的树形交互式语音应答流程不同。以下摘录展示了 Dynamic-Prompt-Agent 如何在节点处理中途应对修正：

assistant：让我们评估一下您的资格。请告知您的收入类别：“受薪”、“自雇”或“失业”。user：受薪。assistant：已记录。正在收集您的收入详情。tool：{ " tool ": " 收入详情收集 ", " method ": " GET ", " saveStatus ": " 已保存 " } user：抱歉，更正一下。我是自雇人士。assistant：收到。正在更新并重新收集收入详情。tool：{ " tool ": " 收入详情收集 ", " method ": " GET ", " saveStatus ": " 已保存 " } assistant：正在验证收入。tool：{ " tool ": " 收入验证 ", " method ": " POST ", " incomeValidationResult ": " 有效 " } assistant：正在进行财务健康评估。tool：{ " tool ": " 财务健康评估 ", " method ": " POST ", " financial HealthStatus ": " 良好 " }