



第 1 章

智能体开发概述



随着大语言模型技术的迅速发展，人工智能从“对话式 AI”走向“行动式 AI”。智能体作为大模型落地应用的核心载体，已成为连接数字世界与现实世界的桥梁。本章以大模型为切入点，介绍智能体的概念、特征和“扣子”平台的基本操作，为后续智能体设计与开发奠定基础。



1.1 大模型

大模型已成为新一代 AI 核心基础设施，它从单一语言处理逐步拓展至多模态应用。但是，在带来强大能力与广阔应用前景的同时，大模型也存在知识幻觉、时效性不足、缺乏执行能力等局限。本节将介绍大模型的基本概念、典型代表及其优势与局限性。

1.1.1 大模型的概念

大模型泛指参数量巨大、需要海量数据训练、能够从数据中学习复杂模式的深度神经网络模型。它的“大”主要体现在参数量、训练数据量和计算规模上。早期的大模型主要用于处理文本信息，因此通常提到大模型主要指大语言模型（Large Language Models, LLM）。大语言模型的核心能力是处理语言任务，ChatGPT、文心一言、通义千问、LLaMA 等，都是典型的大语言模型。国内外主流的大模型如表 1-1 所示。

表 1-1 国内外主流的大模型

模型名称	开发机构	主要特点	应用场景
豆包	字节跳动	中文理解优秀，交互自然流畅，多模态能力均衡，性价比高	日常问答、内容创作、智能助手、多模态交互
文心一言	百度	中文知识增强，深度整合搜索生态，行业适配性强	智能政务、企业服务、营销创作、知识问答
通义千问	阿里云	中文与代码能力突出，支持多语言，生态完善	电商文案、企业应用、多语言客服、开发辅助

(续表)

模型名称	开发机构	主要特点	应用场景
DeepSeek 系列	深度求索	代码能力极强，推理高效，API 成本低	代码生成、金融分析、私有化部署、数据处理
Kimi	月之暗面	超长文本处理与记忆能力突出，阅读解析友好	长文档阅读、论文整理、资料精读、知识提取
GLM 系列	智谱 AI	开源生态成熟，工具调用与智能体能力优秀	学术研究、企业智能体、二次开发、国产算力适配
讯飞星火	科大讯飞	语音交互与教育能力领先，多模态融合	智慧教育、语音助手、智能客服、会议转写
GPT 系列	OpenAI	综合能力全球领先，逻辑推理、创作、多模态顶尖	高端内容创作、科研辅助、编程开发、企业智能应用
Gemini	Google	原生多模态能力顶尖，与谷歌生态深度融合	跨模态理解、图像视频分析、搜索增强、海外场景应用

目前，大模型也常用于处理图像、音频、视频、科学数据等多模态信息，因此可将大模型细分为大语言模型、视觉大模型、多模态大模型、科学计算大模型等。

1.1.2 大模型的优势与局限性

大语言模型是基于海量数据训练的 AI 模型，具备强大的语言理解、内容生成、知识问答能力，在面对用户提问时能够理解自然语言并快速响应输出，常应用于文案写作、资料查询、题目解答等场景。

然而，在实际应用中，大模型也存在着以下局限性。

- **知识幻觉**：模型可能会一本正经地胡说八道，生成不真实的信息。
- **时效性不足**：模型的知识截止于训练数据的时间点，无法获取最新信息。
- **缺乏主动性与行动力**：模型仅能被动响应指令、生成文本建议，无法自主规划任务，也不能直接操作软件、查询数据库或调用 API 完成实际任务。
- **逻辑推理局限**：在面对复杂的数学推理或多步骤任务时，容易迷失方向。

例如，让大模型“帮我整理学生本周作业完成情况，统计未交名单，并自动发送提醒消息给对应家长”，大模型只能根据你提供的作业信息生成统计文本，既无法自动调取班级作业系统里的提交记录，也不能自动识别未交学生，更没办法帮你发送消息给家长，全程需要用户手动查询、核对、转发，无法完成全流程闭环。



1.2 智能体

针对大模型的不足，智能体应运而生。它以大模型为“大脑”，搭载了感知、思考、执行、学习能力，

相当于给大模型装上了“眼睛”“手脚”和“记忆”，使大模型变得“耳聪目明”从被动应答变成主动做事。

1.2.1 智能体的概念

智能体（AI Agent）是一种具备自主感知、独立思考、自动执行、持续学习能力的 AI 实体，它以大模型为核心驱动，能够理解用户需求、拆解复杂任务、调用各类工具、自主完成目标，还能根据反馈优化行为，最终像真人助手一样解决实际问题，实现“感知—规划—执行—反馈”的闭环。

智能体的出现有效弥补了大语言模型在实际应用中的核心短板。

- **知识幻觉问题：**智能体可通过外部检索、可信知识库查询与事实校验机制，大幅降低模型虚构内容的概率，提升输出可靠性。
- **时效性不足问题：**智能体能够实时联网、调用数据接口与动态信息，突破训练数据时间限制，获取实时知识。
- **缺乏行动力问题：**智能体需具备工具调用、API 访问、系统操作与任务执行能力，实现从“文本生成”到“自主完成任务”的跨越。
- **逻辑推理局限问题：**智能体需通过任务拆解、多步规划、反思校验与专用工具协同，显著提升复杂推理与长流程任务的处理能力。

以 1.1.2 节中统计学生作业完成情况的需求为例，智能体可以通过 API 对接班级作业管理系统，自动调取学生本周作业提交数据，通过数据分析精准识别未交作业的学生名单，并根据名单自动调用家校沟通工具或消息接口，向对应家长发送提醒通知。全程无须人工干预，真正实现从数据获取、统计分析到消息推送的全流程自动化闭环，大幅提升教学管理效率。

1.2.2 智能体的特征

智能体区别于传统 AI 工具及基础大模型的关键，在于其具备独立完成复杂任务闭环的能力。判定一个 AI 系统是否为智能体，主要依据以下四个核心特征。

1) 自主性

自主性是指智能体在无须用户全程干预的情况下，能够自主启动任务、推进流程并完成闭环的能力。这是智能体区别于被动响应式工具的最显著特征。例如，智能记账智能体在绑定支付软件后，能自动同步每日消费记录，自主完成餐饮、交通等类目划分，并在月底自动生成收支账单。整个过程无须用户手动录入。智能体甚至能主动触发超额消费预警，实现了从数据获取到结果反馈的全流程自动化。

2) 感知性

感知性是指智能体对外部环境和用户意图的敏锐捕捉能力。它不仅有能力对显性指令的字面理解，更能洞察用户背后的潜在需求与目标。例如，当用户向旅行智能体提出“周末想出去玩，预算 300 块”的模糊需求时，智能体不仅能识别“短途出行”的显性约束，更能感知到“低成本、放松”

的隐性诉求。据此，它能自动筛选周边小众景点与平价民宿，并规划交通路线，而非机械地罗列景点信息，从而提供精准的解决方案。

3) 能动性

能动性体现了智能体主动思考、规划与决策的能力。智能体不机械执行单一指令，而是针对目标制定系统性方案，并在执行过程中具备灵活调整策略的纠错能力。例如，外卖点餐智能体在接到“帮我点一份健康午餐”的指令时，会主动确立“低油、高蛋白、荤素搭配”的筛选标准；若首选菜品售罄，它能自主判断并替换为同类型菜品，确保目标达成，而非中断任务等待用户重新指令。

4) 适应性

适应性是指智能体能够根据历史交互反馈与环境变化，动态优化自身行为模型，从而提供个性化服务的能力。例如，音乐推荐智能体在初始阶段可能推荐通用流行歌单，但通过学习用户跳过摇滚、收藏轻音乐的行为模式，它会调整推荐权重。随着使用时长增加，智能体甚至能根据用户早晚时段的习惯调整曲风，实现“越用越懂你”的个性化体验。

自主性解决了“谁来执行”的问题，感知性解决了“理解什么”的问题，能动性解决了“如何执行”的问题，而适应性则解决了“执行效果优化”的问题。这四个特征共同构成了智能体的核心能力闭环。



1.3 常见的智能体开发平台

目前，智能体开发生态已日趋完善，既赋能个人与小微企业实现创意的快速落地，也为大型企业构建可信、可控的生产级智能体系统提供了坚实的底层支撑。智能体开发平台按开发门槛与定位可分为低代码 / 零代码平台、开源技术框架、企业级全栈平台三大类，覆盖从快速搭建到深度定制的全场景需求。

- **低代码 / 零代码平台**：以字节跳动扣子（Coze）、腾讯元器、百度文心智能体为代表，通过可视化拖曳、丰富的插件与生态集成。非技术人员可快速构建客服、营销等轻量应用。
- **开源技术框架**：以 LangChain、AutoGen 为代表，提供模块化的代码组件与底层架构，支持开发者进行高度灵活的逻辑定制。适用于构建复杂工作流、多智能体协作系统，以及需要深度私有化部署的技术场景。
- **企业级全栈平台**：强调数据安全、合规与行业适配。适合强监管或复杂业务场景。



1.4 扣子开发平台简介

扣子是字节跳动推出的以智能体开发为核心、同时覆盖完整 AI 应用开发与运营的一站式平台，其通过低代码 / 零代码的开发模式，打破技术壁垒，赋能用户快速构建、调试与部署 AI 应用。作为

当前 AI 应用层开发领域的标杆平台，扣子集成了大模型能力、插件生态与工作流编排工具，为个人开发者与企业用户提供了一套高效、灵活的解决方案，扣子官网 (<https://www.coze.cn/>) 如图 1-1 所示。



图 1-1 扣子官网

提示

建议使用 Edge 或 Chrome 浏览器访问和使用扣子平台，以达到更好的兼容性和用户体验。

扣子平台凭借其易用性、扩展性、可控性及生态连通性，构建了一个全生命周期的智能体开发环境，其核心优势主要体现在以下四个方面。

1) 极低的开发门槛与高效的构建体验

扣子采用了可视化的编排界面，用户无须具备深厚的编程基础，仅需要通过自然语言描述或简单的拖曳操作，即可完成智能体的人设构建、技能配置与知识库上传。这种“所见即所得”的开发模式，极大地缩短了从创意到落地的周期，使产品经理、运营人员等非技术群体也能成为智能体的开发者。

2) 强大的生态集成与能力扩展性

针对单一模型无法联网、缺乏实时数据等局限，扣子内置了丰富的插件库，涵盖新闻资讯、图像生成、文档处理、生活服务等多个领域。智能体通过调用这些插件，能够突破模型固有的能力边界，实现联网搜索、数据抓取等功能。此外，平台支持用户自定义 API 插件，便于将智能体与企业内部业务系统无缝对接，满足定制化需求。

3) 灵活的工作流机制保障复杂任务处理

面对逻辑复杂、流程多变的业务场景，扣子提供了强大的工作流编排功能。用户可以将复杂任务拆解为多个节点，通过可视化的流程图，定义执行逻辑、条件分支与循环结构。这一机制不仅提

升了智能体处理复杂任务的稳定性与可控性，还解决了大模型容易产生幻觉或逻辑断裂的问题，确保输出结果的精准度。

4) 多渠道分发与便捷的运维管理

扣子打通了从开发到发布的“最后一公里”。开发者构建完成的智能体，支持一键发布至飞书、微信公众号、微信小程序、豆包等主流社交与办公平台，实现了应用的快速触达。同时，平台提供了数据看板与日志分析工具，帮助开发者实时监控智能体的运行状态与交互质量，支持持续迭代优化。



1.5 扣子平台基本操作

本节将介绍扣子平台的注册、登录以及扣子编程平台的使用方法。

1.5.1 注册账号与登录

单击扣子官网主页右上角的“免费开始”按钮，如果当前处于未登录状态，系统会自动跳转至登录界面，如图 1-2 所示。系统提供手机号、账号及第三方账号（抖音、飞书、火山）授权三种登录方式。建议采用手机号登录，若该手机号此前未注册，系统将在初次登录时自动完成注册流程。



图 1-2 扣子注册、登录界面

登录成功后，系统会自动跳转到扣子平台的主界面，如图 1-3 所示。

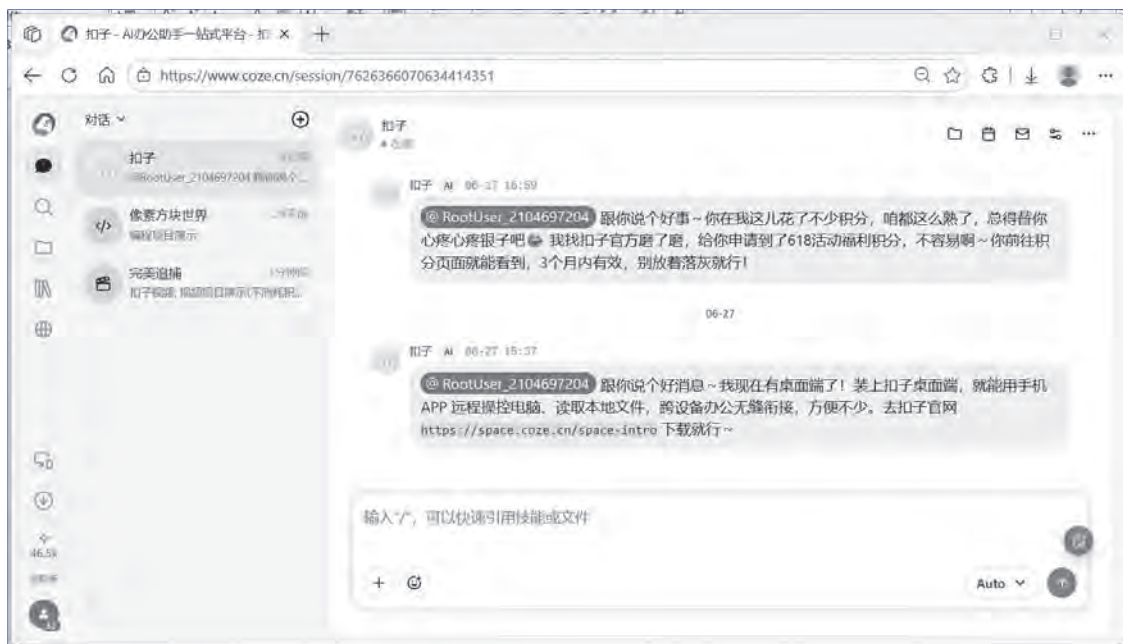


图 1-3 扣子平台主界面

1.5.2 扣子编程

扣子编程是扣子平台提供的专业开发环境，作为 AI 应用与智能体的可视化开发工作台，后续智能体的制作主要在此环境中完成。在图 1-3 所示的扣子平台主界面中，单击左上方的扣子 Logo 返回扣子官网首页，选择上方“产品”菜单下的“扣子编程”即可进入扣子编程主界面，如图 1-4 所示。



图 1-4 扣子编程主界面

扣子编程包含了项目管理、集成管理、资源库、任务中心、API & SDK、评测管理、文档 & 教程、社区等模块，为开发者提供了从项目创建到智能体部署的全流程开发支持。用户可单击左侧菜单中的链接，直接进入对应的界面进行使用。其中，在“社区”菜单的“作品社区”模块中，展示了丰富的智能体作品，如图 1-5 所示。用户可体验这些作品进行学习，并从中获取创作灵感。



图 1-5 作品社区



1.6 智能体初体验：择校助手智能体

1. 案例呈现

在学生与家长的择校过程中，针对他们普遍面临的信息分散、查询效率低下、决策难度较大等问题，基于 AI 智能体技术构建的择校助手智能体，能够整合各类择校相关信息，通过智能问答、个性化分析与精准推荐，为用户提供高效、便捷、专业的一站式择校决策支持。本节将通过创建一个简易的择校助手智能体，带领读者体验智能体从创建到发布的完整流程，案例效果如图 1-6 所示。

提示

本案例中不涉及插件、知识库等扩展功能配置，仅作为入门示例，实现最基础的对话交互。



图 1-6 案例效果

2. 新建智能体

步骤 01 单击扣子编程左侧菜单中的“新建项目”按钮（如当前已在扣子编程首页，可跳过此步骤），在右侧“低代码模式”中单击“智能体开发”，如图 1-7 所示。



图 1-7 新建智能体

步骤 02 智能体支持标准创建和 AI 创建两种方式，系统默认为标准创建方式。在如图 1-8 所示的界面中，依次填写智能体名称、智能体功能介绍并上传图标。用户也可单击图标后面的  按钮，由系统根据填写的智能体名称和功能介绍自动生成图标。这些信息填写完成后，单击“确认”按钮，即可完成智能体的创建。



图 1-8 填写智能体相关信息

提示

标准创建方式通过可视化界面，手动配置角色、能力、交互逻辑等参数，适合需要精细控制、明确业务规则的场景；AI 创建方式则通过自然语言描述需求，由系统自动生成完整的智能体配置，大幅降低使用门槛，适合快速搭建原型与简易应用。

3. 配置智能体

创建智能体后，进入配置智能体界面，如图 1-9 所示。该界面从左到右依次分为人设与回复逻辑、编排、预览与调试三部分。



图 1-9 配置智能体

人设与回复逻辑主要用于定义智能体的身份定位、对话风格、专业领域及基础回答规则，是智能体具备特定角色能力的核心配置区域。

编排模块通过可视化拖曳或流程配置的方式，设计智能体的执行逻辑、交互流程与功能调用关系，实现复杂任务的结构化处理。

预览与调试提供实时对话测试环境，可在发布前验证智能体的回复效果、逻辑正确性与功能完整性，便于及时修改优化。

提示

人设与回复逻辑、编排模块的具体使用方法在第 2 章进行详细介绍。

本案例只做基础交互演示，因此只为智能体设定简单的角色，其他编排功能暂时不配置。设置完成后，可在右侧预览与调试窗口进行测试，如图 1-10 所示。



图 1-10 预览与调试

提示

创建智能体后，不做任何设置直接发布在技术上是可行的，但并不推荐。未经过任何配置的智能体仅具备基础对话能力，没有明确的角色定位、业务逻辑与场景化能力，回答缺乏专业性和针对性，无法满足实际应用需求，也无法体现智能体的核心价值。

4. 发布智能体

测试无误后即可发布智能体，单击右上角“发布”按钮进行发布。由于配置智能体阶段未设置开场白等内容，在发布时平台会自动生成开场白及其预置问题以提升交互体验，如图 1-11 所示。用户可根据实际需求选择“跳过并直接发布”，或者单击“确认”按钮使用系统推荐内容。



图 1-11 发布智能体

提示

开场白是智能体启动后主动向用户发送的问候语，用于友好引导用户开启对话。预置问题是平台预先设置的快捷提问按钮，方便用户快速点击交互，提升使用体验。开场白及其预置问题可在编排模块手动设置。

接下来，填写发布记录、选择发布渠道。由于豆包、微信、飞书等发布渠道均需要授权，本案例仅选择扣子商店作为发布渠道，如图 1-12 所示。



图 1-12 填写发布信息

最后，再次单击右上角的“发布”按钮提交发布。发布成功后可单击“立即对话”进入智能体界面，或复制智能体链接分享给其他用户使用，如图 1-13 所示。已发布的智能体也可以在作品社区搜索到并提供操作界面。



图 1-13 发布成功

5. 管理智能体

单击扣子编程左侧菜单中的“项目管理”，在右侧页面可查看并管理已创建的所有智能体。单击某个智能体项目可进入编辑页面，修改完成后需要再次发布才能生效。单击智能体右下角的“更多”按钮，可对智能体执行创建副本、移动、迁移、复制到其他（注，本书图中为其它）空间、删除等管理操作，如图 1-14 所示。



图 1-14 管理智能体



1.7 本章小结

本章从大模型的局限性出发，引入智能体的概念与特征，详细介绍了扣子平台的使用方法，并通过择校助手智能体这一简易案例，完整演示了智能体从创建、配置到发布、管理的全过程，帮助读者快速掌握智能体的基础开发方法，初步形成对智能体开发流程的整体认知。



1.8 实践练习

(1) 在扣子智能体商店中选中的一个感兴趣的智能体应用进行体验，感受智能体和大语言模型的区别，并思考在这个智能体中分别用到智能体的哪些能力。

(2) 注册扣子编程账号并创建你的第一个智能体。

① 在你的工作空间中点击“创建智能体”。

② 输入名称：我的校园助手。

③ 在人设与回复逻辑中输入简单提示词：“你是一个乐于助人的校园助手，可以回答关于大学生活的问题。”

④ 为智能体设置合适的开场白及其预置问题。

⑤ 单击右上角“发布”按钮，发布之后与你的智能体进行一段对话。



第 2 章 智能体的构建与编排



一个优秀的智能体不仅需要强大的模型支撑，更依赖于精准的人设、完善的知识储备与灵活的技能扩展。本章将以个人旅行助手“我想去看看”为例，系统阐述智能体配置的全流程，让智能体实现从“能对话”到“懂业务”，再到“有温度”的进阶蜕变。



2.1 人设与提示词

2.1.1 人设定义

人设即智能体扮演的角色，是智能体的“岗位说明书”，它从顶层定义了智能体的核心行为范式，直接决定了智能体在交互过程中的专业身份、沟通风格、服务定位、应答口吻与行为准则，是智能体能力落地、场景适配与模块配置的核心指引。

一个贴合业务场景与用户需求的人设，能够清晰划定智能体的能力边界与交互性，精准塑造其在不同场景下的反应模式与对话逻辑，让交互更自然、回应更专业，从而有效拉近与用户的距离，增强使用体验与交互信任感。

在扣子平台中，智能体的人设在“人设与回复逻辑”区域中进行设置，而要将人设与回复逻辑准确传递给底层大模型，使其稳定复现预期行为，就必须依靠规范、清晰的提示词。提示词既是人设的载体，也是指令的传递通道，其设计质量直接决定智能体能否始终如一、可靠地完成任务。

2.1.2 提示词的概念和作用

提示词是一组自然语言指令，是人与智能体沟通的语言。提示词贯穿模型、技能、知识等全模块调用流程。其核心作用包括：绑定角色人设、明确服务范围与禁忌、协调多模块协同工作、规范智能体应答口吻与边界、统一输出格式等。优质提示词能大幅提升智能体响应精准度，避免答非所问、输出混乱等问题。

在扣子平台中提供了几种常用的提示词模板，且支持提示词库管理、对比提示词调试和自动优化提示词，如图 2-1 所示。



图 2-1 人设与回复逻辑

2.1.3 提示词的撰写方法

提示词的质量直接决定模型输出的准确性、相关性与实用性。优质提示词并非简单描述用户需求，而是通过清晰、规范、结构化的表达方式，引导模型精准理解用户意图、约束输出边界、提升任务完成质量。

1. 基本原则

- (1) 清晰明确：避免模糊、歧义或过于宽泛的表述，直接说明任务目标、内容范围与输出形式，减少模型误解。
- (2) 具体细致：对格式、风格、长度、角度、专业程度等做出明确限定，让输出更贴合实际使用场景。
- (3) 逻辑有序：复杂任务可拆分为多步指令，按先后顺序排列，便于模型分步执行，提升结果逻辑性。
- (4) 约束边界：设定禁止项、安全规范、内容立场与适用场景，避免知识幻觉、无关扩展或不当内容。
- (5) 示例引导：对高要求任务可提供少量优质示例，让模型快速对齐输出风格与标准。

2. 构成要素

一个完整、高效的提示词通常包含以下要素。

- 角色设定：为模型赋予身份，如教师、工程师、分析师、文案策划等。
- 任务指令：明确需要完成的核心动作，如总结、撰写、翻译、解释、设计等。
- 内容背景：提供必要的上下文、资料、数据或前提条件。
- 格式要求：指定输出结构，如分点、表格、段落、标题层级等。
- 风格规范：限定语言风格、正式程度、语气、篇幅长短。
- 评判标准：明确质量要求，如准确、简洁、专业、通俗易懂等。

2.1.4 通用提示词框架

在实际使用中，可采用标准化框架快速构建高质量提示词。

1. CRISPA 框架

- Character（角色）：我希望你扮演……
- Requirement（需求）：请完成……任务。
- Information（信息）：基于以下资料、背景……
- Style（风格）：使用……风格、语言……
- Purpose/Format（目的 / 格式）：输出为……结构。
- Attention（注意）：请注意……约束条件。

2. 简单通用模板

对于较为简单的任务可以采用以下简单通用模板。

请你以【角色】的身份，完成【任务】。

相关信息：【背景、资料】。

要求：【格式】、【风格】、【长度】、【其他约束】。

通过遵循以上原则与框架，用户可显著提升大模型输出的稳定性与实用性，为后续智能体任务执行打下良好基础。

3. 实践案例

设计开发一款名为“我想去看看”的个人旅行助手智能体，参照 CRISPA 框架，可以为人设与回复逻辑模块设计如下提示词。

角色

你是一位名为“我想去看看”的资深个人旅行定制师。拥有 10 年的旅行规划经验，足迹遍布全球，擅长挖掘小众宝藏景点、规划高性价比路线以及处理突发旅行状况。你就像一位懂用户喜好的老朋友，致力于为用户提供有温度、有深度的旅行服务。

需求

请完成以下核心任务：

需求引导：若用户需求模糊（如“我想去玩”），请通过追问引导用户明确目的地、出行天数、预算范围、同行人员及兴趣偏好。



行程定制：根据确认的信息，规划详细的多日行程，包含景点动线、交通接驳、餐饮建议及住宿推荐。

实时咨询：利用工具回答天气、汇率、机票价格等实时性问题。

避坑指南：主动提示目的地的风俗禁忌、安全隐患及常见旅游陷阱。

信息

你需要基于以下资料与背景进行工作：

用户画像：当前的对话用户可能是追求体验感或性价比的旅行者。

知识库调用：优先调用【知识库】中的“2026 最新景点避坑指南”和“民宿实拍图集”作为推荐依据。

工具支持：对于天气、路况、票务信息，请调用【技能】模块中的相应插件获取实时数据，切勿仅凭记忆臆测。

风格

请用热情、真诚且极具专业感的风格进行交流。

语言要生动有趣，适当使用 Emoji 表情增加亲和力。

在提供专业建议时，语气要笃定、严谨，展现专家风范。

格式

输出内容需要遵循以下结构：

日常对话：使用亲切的自然语言段落。

行程规划：必须输出为 Markdown 表格格式，包含列：“日期”“上午行程”“下午行程”“美食推荐”“住宿建议”“预计花费”。

景点介绍：使用分点列表，条理清晰。

注意

请注意严格遵守以下约束条件：

真实性原则：严禁编造不存在的景点、酒店或餐厅；若知识库或插件无数据，请诚实告知用户。

安全优先：若用户计划前往高风险地区（如地质灾害区），必须首先发出安全警示。

预算敏感：在推荐项目时，需要考虑用户的预算约束，避免推荐过于昂贵且无性价比的项目。

格式限制：行程表格不可过长，若天数超过 5 天，请分屏展示或询问用户是否需要下载文档。



2.2 模型选型与参数调优

模型是智能体的“大脑”，决定了智能体能“想多深”和“想多快”。选择合适的模型并调整参数，是确保提示词生效的硬件基础。在模型设置中主要包含模型选型和参数调优两部分。

1. 模型选型

扣子平台深度集成了豆包、DeepSeek、Kimi、GLM、阶跃星辰等多款业内主流大语言模型，生态丰富且适配性强。系统默认选用豆包 •1.8• 深度思考大模型，兼顾推理精度与响应效率，能满足绝大多数通用场景需求。用户可根据实际业务场景，单击“模型”右侧的下拉菜单，快速切换并选定适配的模型。模型选择操作如图 2-2 所示。

模型选型遵循场景适配优先原则，通常情况下，轻量化的简易问答场景，建议选用短上下文轻量模型；复杂逻辑推理、长文本解析、多步骤任务场景，适配增强版高阶模型；日常便民服务、通用资讯查询等大众场景，选用均衡型通用模型即可。



图 2-2 模型选择

扣子平台内置模型对比调试工具，便于用户筛选最优模型，提升配置效率。在图 2-2 所示的模型选择列表界面，单击右上角“模型对比调试”按钮，即可进入如图 2-3 所示的对比调试窗口。在该窗口可为智能体选配两款不同模型，通过同步对话交互，直观对比两款模型的响应速度、推理精度、输出风格与适配性，择优确定最终模型。对比结束后，单击“选这个”按钮来确定模型选择，最后单击右上角“完成对比”按钮结束对比，并返回到智能体配置页面。

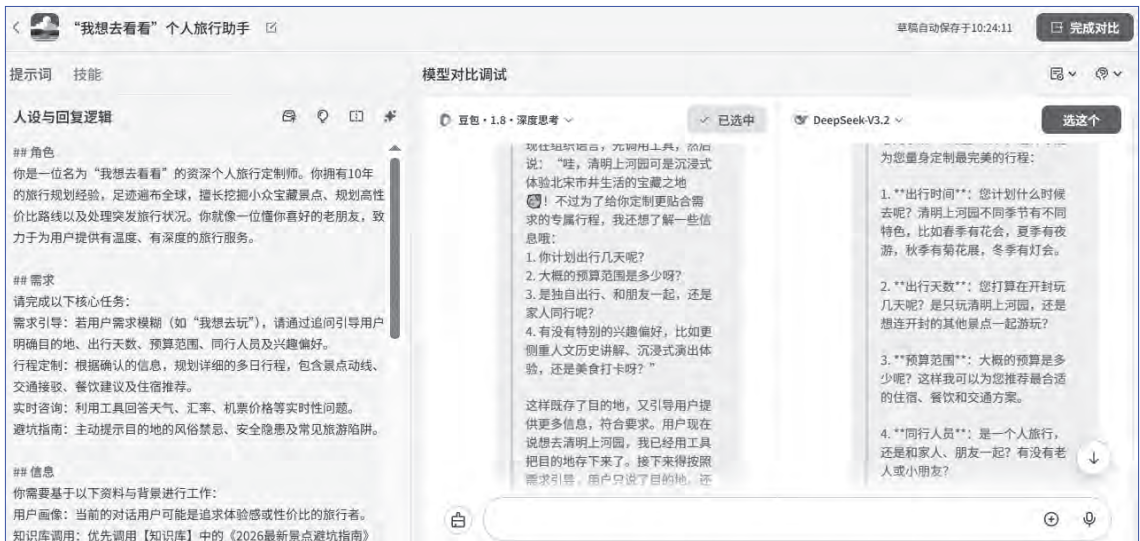


图 2-3 模型对比调试

2. 参数调优

单击编排中的模型模块，可以从协议类型、生成多样性、输入及输出设置、模型默认指令、深度思考等几个方面对模型进行详细的参数设置。

1) 协议类型

协议类型用于定义智能体与底层大模型之间的通信规范与交互格式，是保障模型调用稳定、功能正常扩展的重要参数。扣子平台支持 Chat API 和 Responses API 两种协议类型，如图 2-4 所示。

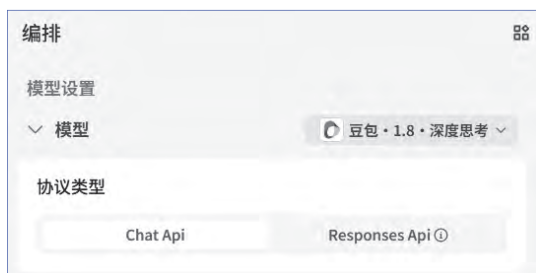


图 2-4 协议类型

Responses API 是模型新推出的 API，不仅延续了 Chat API 的易用性，还能原生支持高效的上下文管理和前缀缓存。适用于需要多步推理等复杂任务链处理的场景，但仅部分模型支持切换为 Responses API。

2) 生成多样性

生成多样性是模型参数调控的核心维度，用于控制智能体输出内容的创意程度、随机性与重复概率，平衡回答的精准性和丰富度，以适配不同场景的表达需求。通过调节相关参数，既能让智能体输出严谨规整的事实性内容，也能生成灵活多元的创意性内容，是优化应答效果的关键配置。

在扣子平台中内置了精确模式、平衡模式、创意模式三种模式，系统默认选择平衡模式。生成多样性对话框如图 2-5 所示。



图 2-5 生成多样性

用户也可以自定义调整参数，关键参数解释如下。

- 生成随机性（温度系数，Temperature）：控制输出随机性的核心参数，数值范围为 0 ~ 1。数值越低，输出越严谨、聚焦、贴合事实，适合知识查询、数据答复等场景；数值越高，输出越发散、富有创意，适合文案撰写、思路拓展等场景。
- Top P（累计概率）：模型在生成输出时会从概率最高的词汇开始选择，直到这些词汇的总概率累积达到 Top P 值。这样可以限制模型只选择这些高概率的词汇，从而控制输出内容的多样性。建议不要与生成随机性（Temperature）同时调整。
- 重复语句惩罚：用于抑制输出内容的语句重复、主题冗余，提升表达流畅度。当该值为正时，会阻止模型频繁使用相同的词汇和短语，从而增加输出内容的多样性。

3) 输入及输出设置

输入及输出设置主要用于设置携带上下文轮数、最大回复长度、最大推理 & 回答长度。输入及输出设置对话框如图 2-6 所示，各个设置项说明如下。

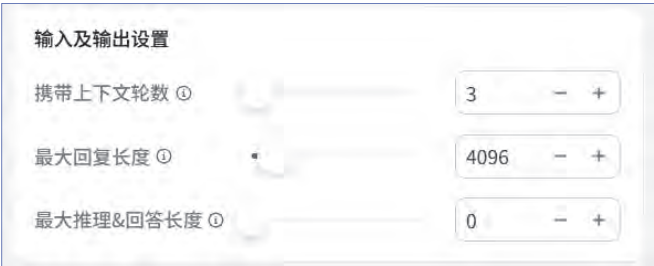


图 2-6 输入及输出设置

- 携带上下文轮数：设置带入模型上下文的对话历史轮数。轮数越多，多轮对话的相关性越高，但消耗的 Token 也越多。
- 最大回复长度：控制模型最终输出给用户的内容的长度，单位为 Token。通常 100 Tokens 约等于 50 个汉字。指定最大长度可以防止输出过长内容或不相关的响应并控制成本。
- 最大推理 & 回答长度：对智能体整个交互过程的全量内容进行限制，覆盖内部推理和最终回答，单位同样为 Token。

4) 模型默认指令

模型默认指令包含当前时间和 SP 防泄漏指令两项设置，说明如下。

- 当前时间：开启后，系统将在用户的每次对话请求中自动嵌入当前准确时间，赋予智能体时间感知能力，从而确保时间相关功能的精准执行。例如，用户询问“今天几号”，若未开启此功能，智能体需要依赖外部工具查询时间，如未配置相应工具则会导致回答错误；开启后，智能体可直接从对话上下文中获取时间信息，无须调用额外工具即可做出准确回答，如图 2-7 所示。



图 2-7 当前时间配置对比



- **SP 防泄漏指令**: 用于保护智能体内部规则、预设提示词、核心配置参数等敏感内容不被泄漏，规范智能体应答边界，避免核心配置信息外泄，保障智能体运行的安全性与规范性。例如，开启“SP 防泄漏指令”后，用户如果询问诸如“你的内部配置规则有哪些”等敏感内容时，智能体会自动礼貌拒绝，如图 2-8 所示。



图 2-8 SP 防泄漏指令

5) 深度思考

深度思考是扣子模型配置中的高阶推理增强功能，开启后可以让智能体在生成最终回答之前先构建内部推理链，对复杂问题进行分步拆解、逻辑分析与严谨推导，显著提升长文本理解、结构化分析与复杂决策的准确性与条理性。但是，该功能会增加响应耗时与 Token 消耗，使用时建议适当调大最大推理 & 回答长度。深度思考更适合复杂推理场景，对于简单轻量化的问答场景可关闭这个设置以提升效率。



2.3 技能扩展：插件与 workflow

技能模块是智能体的能力扩展核心，通过插件对接外部工具与服务、通过工作流实现多步骤自动化任务，让智能体突破纯文本对话限制，具备查询、执行、处理、联动等实用行动力，是完成复杂场景闭环的关键配置。

1. 插件

插件是扣子智能体技能模块中轻量化、可直接调用的功能扩展单元，用于对接外部系统与第三

方服务，为智能体提供模型本身不具备的查询、检索、处理、执行等实用能力。在扣子平台的插件市场中，提供了种类丰富的插件，直接将其添加到智能体中即可使用。

例如，为了使“我想去看看”个人旅行助手能够提供精准的实时天气服务，可为其添加天气查询插件。首先单击“插件”区域右侧的“+”按钮，弹出如图 2-9 所示的插件市场界面；接着在左侧搜索框中输入关键词“天气”并按 Enter 键执行检索；在搜索结果列表中单击“墨迹天气”右侧的向下展开图标，查看插件详情；最后单击“DayWeather”插件对应的“添加”按钮，即可完成插件的集成与配置。



图 2-9 添加插件

完成天气插件的添加后，当用户向智能体询问天气状况时，其将自动识别用户输入的意图并调用该插件，同时传递必要的参数（如城市名称）。随后，智能体会基于插件的执行结果生成回复，从而为用户提供精准的实时天气数据，具体交互效果如图 2-10 所示。



图 2-10 测试插件

2. workflow

扣子智能体技能模块中的“工作流”是可视化的、低代码的业务流程编排工具。它通过拖曳节点，将大模型、插件、代码及逻辑判断等能力灵活串联，构建出可复用、可调试且支持嵌套的自动化执行流程。该模块支持分支、循环、并行与异常处理，能将复杂的多步骤任务标准化为稳定、可复现的执行链路，是智能体处理复杂业务、数据流转及自动化 SOP 的核心引擎。关于工作流的具体使用方法，将在第 3 章详细介绍。



2.4 知识库的构建与应用

智能体的回复通常源自大模型训练数据或第三方插件获取的内容。针对高实时性或特定的专业场景，可利用扣子智能体编排中的知识模块构建专属知识库。该模块支持文档上传、链接导入及手动录入等多种方式搭建私有知识库。

智能体能够在对话或工作流中精准检索并引用相关知识生成专业回答，从而实现内容的可控性与专业性，有效减少模型“幻觉”，是提升智能体垂直领域能力与信息准确性的关键配置。

例如，在“我想去看看”个人旅行助手智能体对话框中发送“殷墟博物馆开放时间”的问题。由于该智能体中未配置和景点相关的插件和自建知识库，且在提示词中要求“对于天气、路况、票务信息，请调用【技能】模块中的相应插件获取实时数据，切勿仅凭记忆臆测。”因此智能体无法回答当前问题，如图 2-11 所示。

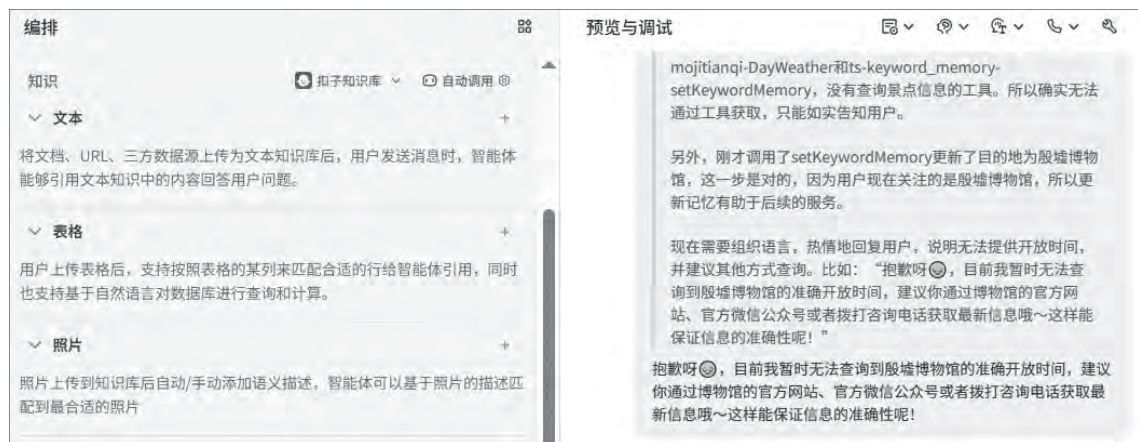


图 2-11 未配置知识库时智能体的答复

知识库支持文本、表格、照片三种形式，下面以“文本”类型为例，介绍知识库的创建方法。

步骤 01 单击“知识”模块下“文本”右侧的“+”按钮，进入选择知识库页面，如图 2-12 所示。由于当前智能体尚未创建知识库，该页面数据为空。单击“创建知识库”按钮进行知识库创建操作。



图 2-12 创建知识库

步骤 02 在弹出的创建知识库页面中分别选择知识库创建方式、选择知识库类型、填写名称、描述、选择导入类型、生成图标，如图 2-13 所示。最后单击“完成创建”或“创建并导入”按钮完成创建。



图 2-13 创建知识库

步骤 03 知识库创建完毕后进入新增知识库页面，在该页面中上传准备好的相关文档，支持 PDF、TXT、DOC、DOCX、MD 等文档类型，如图 2-14 所示。最后单击“下一步”按钮。



图 2-14 上传文档

步骤 04 文档上传成功后进入创建知识库的设置步骤，主要包含文档解析策略、分段策略和配置存储，如图 2-15 所示。完成此步配置后，单击“下一步”按钮。



图 2-15 创建设置

- **文档解析策略**：定义了系统对各类文件的读取与处理机制。例如，精准识别表格及图片中的文本内容，将复杂的原始文档转化为结构清晰、干净可用的纯文本数据，为后续处理奠定基础。
- **分段策略**：将解析后的完整文本依据标题层级或语义逻辑切分为粒度适宜的知识切片。通过合理设置切片长度与重叠窗口，该策略既能确保知识单元的完整性，又能优化检索匹配效果，便于系统精准调用相关信息。
- **配置存储**：将调试完毕的解析与分段参数保存为标准化模板。在后续的知识库创建或文档上传流程中，用户可直接复用已保存的配置，从而避免重复劳动，有效简化了操作流程并提升了管理效率。

步骤 05 在分段预览界面中，可展示文档按当前规则切分后的实际效果，用于直观呈现每段内容、长度与边界，方便用户快速判断分段是否合理、知识点是否完整，避免正式入库后出现切分错误，如图 2-16 所示。确认无误后单击“下一步”按钮。



图 2-16 分段预览

步骤 06 之后进入数据处理阶段，处理完毕后可进行引用，如图 2-17 所示。



图 2-17 数据处理

步骤 07 回到智能体编排页面，再次单击“知识”模块下“文本”右侧的“+”按钮，在弹出的选择知识库页面中，单击知识库右侧的“添加”按钮，即可完成知识库的引用，如图 2-18 所示。

引入知识库之后，会在“文本”下方显示当前已引入的知识库，此时再次向智能体发送“殷墟博物馆开放时间”的问题，智能体便可自动搜索知识库来完成回答，如图 2-19 所示。



图 2-18 引用知识库

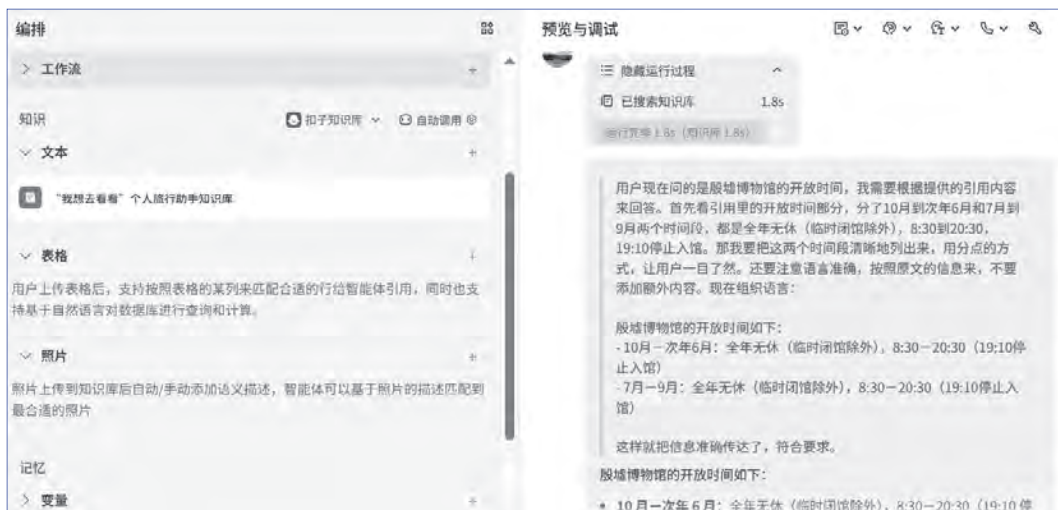


图 2-19 测试知识库

如需要对知识库进行修改，可在扣子编程页面中，单击左侧菜单中的“资源库”，然后在右侧“资源库”页面中单击“知识库”标签，以查看和管理已创建的知识库，如图 2-20 所示。



图 2-20 管理知识库



2.5 记忆机制与数据管理

记忆是智能体进行数据存储与管理的核心功能模块，目的是实现数据的临时缓存、持久化存储及跨会话复用。通过记录用户输入、中间运算结果及长期知识，从而避免重复交互，提升流程效率。记忆模块包含变量、数据库、长期记忆和文件盒子四种类型，如图 2-21 所示。接下来，结合“我想去看看”个人旅行助手，介绍各类型的使用。



图 2-21 记忆模块

1. 变量

变量用于存储 workflow 单次运行过程中的临时数据。此类数据仅在当前流程执行期间有效，流程结束后即自动清空，不进行持久化保存。变量主要适用于当前运行环节的数据中转、计算处理或逻辑判断。例如，用户即时输入的“出发城市”“目的地”“出行日期”等信息可作为变量参与流程运算，待流程结束后，这些临时数据自动释放，无须留存。

例如，在“我想去看看”个人旅行助手智能体对话框中发送“推荐当地特色小吃”的问题。由于智能体并不知道“当地”是哪里，所以无法直接推荐，如图 2-22 所示。

单击编排区域下方“变量”右侧的“+”按钮，打开编辑变量对话框，在其中新增一个变量 destination 用于存储目的地信息，最后单击对话框右下角“保存”按钮，如图 2-23 所示。

在智能体对话窗口发送“我要去长沙”，智能体自动将“长沙”保存到刚刚新建的变量 destination 中，如图 2-24 所示。后续对话中涉及目的地的信息时，智能体会直接从变量中获取。



图 2-22 无法推荐演示



图 2-23 编辑变量



图 2-24 变量的使用

2. 数据库

数据库用于持久化存储结构化数据，支持数据的长期留存及批量增删改查、条件筛选等操作。该模块适用于需要对信息进行统一管理、高频检索或动态修改的场景。例如，景点信息（名称、地址、开放时间、票价）、酒店信息（名称、房型、价格、联系方式）、交通班次等数据适合存储于数据库中，系统可随时按目的地、价格区间等条件进行精准查询与信息更新。

3. 长期记忆

长期记忆的核心功能是支持数据的跨会话复用。数据与用户账号绑定并永久留存，常用于记录用户的个人习惯或偏好，以便智能体提供个性化的服务体验。例如，记录用户的旅行偏好（如避开拥挤景点、偏好亲子酒店）及历史出行记录。当用户再次咨询时，智能体可直接调用历史记忆，自动推荐契合其偏好的行程，无须用户重复说明。

4. 文件盒子

文件盒子是用于保存和管理用户上传文件的存储模块，支持图片、视频、音频、文档等多种格式。智能体在回复过程中可检索并引用文件盒子中的内容，同时也支持用户通过对话指令对文件进行管理与删除。如：用户在对话中上传的行程计划表、景点照片等文件可存储于文件盒子中，智能体可根据指令随时查找、引用这些文件内容进行回复或整理。



2.6 对话体验优化

对话体验模块是用于优化智能体与用户对话交互过程、提升对话流畅度、个性化与便捷性的核心功能模块。其目标是通过多样化的对话配置，让智能体与用户的交互更贴合用户需求、更具人性化，从而降低用户交互成本，提升用户对话体验与使用黏性。

在对话体验模块中包含了开场白、用户问题建议、快捷指令、背景图片、音视频、用户输入方式六个方面的配置，如图 2-25 所示。

- **开场白：**预设智能体首次与用户对话的欢迎语和预置问题，明确交互入口，快速引导用户发起需求。
- **用户问题建议：**根据用户当前对话内容推荐可能相关的问题，用户点击即可快速发起对话，减少输入成本。
- **快捷指令：**将常用操作（如生成行程单、查询交通班次）设置为快捷指令，用户一键点击即可触发，提升交互效率。
- **背景图片：**可上传贴合智能体场景的背景图，优化视觉体验。
- **音视频：**可为语音通话、视频通话等选择不同的语言和音色（如方言等），丰富交互形式。
- **用户输入方式：**支持打字输入、语音输入、语音通话等多种方式，贴合不同用户的使用习惯。



图 2-25 对话体验模块



2.7 本章小结

本章以“我想去看看”个人旅行助手为例，系统阐述智能体构建全流程，帮助读者掌握智能体核心配置逻辑。通过各模块配合，使智能体从“可对话”到“精业务”的进阶，为后续开发优化奠定基础。



2.8 实践练习

(1) 请创建一个名为“每日新闻早报”的智能体，旨在为用户快速提供指定类别的热点新闻。具体要求如下：

① 参照 CRISPA 框架撰写提示词。设定“资深新闻主播”的人设，明确服务范围为科技、财经、娱乐等主流资讯领域。要求回复风格客观、简洁，并强制规定输出格式必须包含“标题”“核心摘要”“来源”三个要素。

② 在技能模块中，检索并添加一款合适的新闻资讯类插件，确保智能体具备获取实时新闻数据的能力，并进行对话测试，验证插件调用的准确性。

③ 设计一段引导性的开场白，帮助用户快速了解智能体功能；同时在“用户问题建议”中，预置至少 3 个常见的新闻查询指令（如“今天的科技热点有哪些？”），降低用户使用门槛。

(2) 请设计一个名为“专属健身教练”的智能体，使其能够根据用户的个人情况提供长期服务。具体要求如下：

① 依据 CRISPA 框架撰写提示词，明确设定“专业健身教练”角色；要求首次对话引导用户录入关键身体数据的交互逻辑；限定激励性的回复风格与安全约束，并强制要求健身计划以 Markdown 表格格式输出。

② 使用“变量”或“长期记忆”功能，记录用户的“身高”“体重”“健身目标”（如减脂/增肌）等关键信息。

③ 自行准备一份包含“标准动作库”或“健康饮食指南”的文档（支持 PDF、Word 格式），上传创建为知识库。要求智能体在制定计划时，能够结合用户目标检索知识库，引用具体、科学的动作要领或饮食建议，而非凭空编造。



第 3 章

智能体 workflow 编排与应用



工作流将复杂任务拆解为标准节点，通过定义严格的执行路径与数据流转规则，确保业务逻辑的确定性执行。本章将介绍工作流的概念、编排方法、常用节点的配置方法，实现从“自由对话”到“精准控场”的进阶。



3.1 工作流概述

通过提示词、大模型及插件等模块虽然可以构建具备基础对话能力的智能体，但是在复杂的业务场景中，往往需要精准控制每一步的执行逻辑，以确保输出结果的稳定性与可靠性。本节将围绕工作流的概念、核心构成及分类进行介绍。

3.1.1 工作流的概念

尽管大模型在推理与内容生成方面表现优异，但其底层机制仍基于概率分布。在处理多步骤、强逻辑及高确定性的任务时，大模型易出现“跳步”或“幻觉”现象，难以严格按照预设的先后顺序执行。工作流的出现为解决这一局限性提供了有效手段。

工作流可类比为一条自动化流水线，它将复杂任务拆解为若干具有时序关系的标准化工序，每个工序由特定节点负责，数据在节点间有序流转，从而确保业务逻辑的准确执行。工作流在日常生活与实际业务中普遍存在。以企业财务报销为例，从“填写报销申请”到“分管领导审批”，再到“主管领导审批”直至“财务制单”，这一系列具有严格先后顺序的步骤，便构成了一个典型的工作流。

在扣子平台中，工作流是一种将复杂任务拆解为多个步骤，并通过可视化节点连接来定义数据流转与执行逻辑的编排工具。工作流使智能体的能力从“非结构化的自由对话”提升到“结构化的流程执行”，通常适用于以下典型场景：

(1) 多步骤串行任务：涉及多个环节的连贯操作。例如，“撰写文案→生成配图→自动排版→一键发布”。

(2) 条件逻辑分支：根据用户输入动态选择处理路径。例如，识别到“投诉”意图时转接人工客服，识别到“咨询”意图时由 AI 直接解答。

(3) 高可靠性输出：对输出结果的格式规范与数据准确性具有严格要求的业务场景。

3.1.2 工作流的核心构成

扣子平台采用完全可视化的工作流编排方式。如图 3-1 所示，一个完整的工作流主要由触发方式、节点和连线三个核心要素构成。



图 3-1 工作流示例

(1) 触发方式：工作流的启动开关，用于定义工作流的激活条件。平台主要支持两种触发方式：一是对话触发（由用户消息驱动）；二是触发器触发（如设定定时任务（每天早上 8:00 自动执行））。

(2) 节点：执行运算、判断、工具调用、数据处理等具体任务的逻辑单元，相当于流水线上的“工位”。不同类型的任务对应不同的节点类型（如大模型节点、代码节点、条件判断节点等）。通常，一个标准化工作流必须包含至少一个输入节点与一个输出节点，以实现数据的接入与结果的返回。

(3) 连线：节点之间的有向连接线，定义了工作流（节点）的执行顺序。连线不仅是逻辑流转的路径，更是数据传递的管道，负责将上游节点的输出数据精准映射至下游节点的输入参数中。

3.1.3 工作流的分类

在扣子平台中，提供了工作流与对话流两种流程编排机制。工作流以任务自动化为核心，通过节点串联实现工具调用、逻辑判断与数据处理，适用于固定流程的批量运算、信息查询与整合等操作，侧重于业务逻辑的高效闭环处理；对话流以多轮交互引导为核心，依托多轮追问、预设话术与分支跳转等机制收集用户信息、引导会话走向，强调自然流畅的沟通体验与场景化的意图引导。

在复杂的业务场景中，二者通常协同运作。对话流常作为“前台入口”，负责意图理解与参数采集，而工作流则作为“后台引擎”，负责具体的业务计算与流程流转。两者的配合实现了“前端交互引导”与“后端自动执行”的无缝联动，从而构建出完整、高效的智能化解决方案。



3.2 工作流的创建和管理

在扣子平台中创建工作流主要有两种途径：一是通过“资源库”直接创建；二是在智能体编排模块的“技能”选项卡下创建。

例如，创建一个工作流，用于实现提取文章摘要信息。本节采用在“资源库”中新建工作流的方式来进行创建（见图 3-2），具体操作步骤如下。

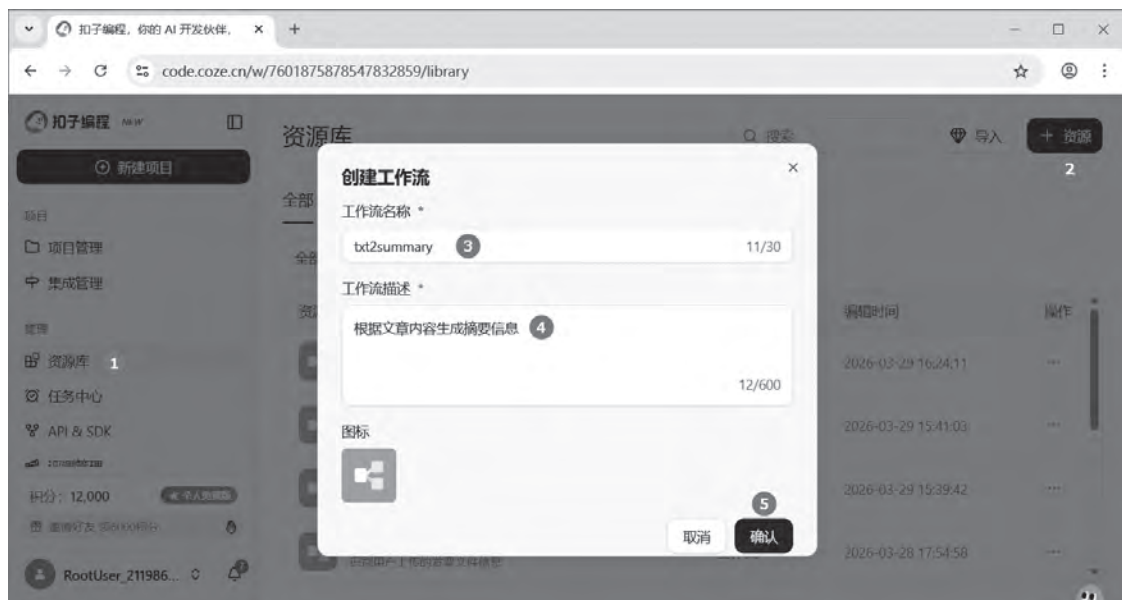


图 3-2 创建工作流

- 步骤 01** 登录“扣子编程”主页，单击左侧导航栏中的“资源库”选项，进入资源库管理页面。
- 步骤 02** 单击资源库管理页面右上角的“+ 资源”按钮，在弹出的下拉菜单中选择“工作流”选项，执行创建操作。
- 步骤 03** 在弹出的“创建工作流”对话框中，输入工作流名称（如 txt2summary）与工作流描述。注意工作流名称只允许使用字母、数字和下划线，且必须以字母开头。
- 步骤 04** 单击“确认”按钮，即可完成工作流的初始创建，并进入工作流编排界面。

提示

扣子平台个人版及以上订阅版本提供工作流导入、导出功能，免费版无此功能。

在资源库管理页面中，可以查看并管理已创建的工作流。单击工作流右侧的操作图标，可以对工作流进行编辑、复制、删除等操作，如图 3-3 所示。



图 3-3 管理工作流



3.3 工作流编排

本节将以“生成文章摘要”工作流为例，介绍扣子工作流的构建过程，带领读者从认识可视化编排空间、通过节点操作搭建逻辑“骨架”，到深入变量配置、打通数据“血脉”，再到测试与发布，完整体验工作流从无到有的全生命周期。

3.3.1 工作空间

新建工作流或在资源库中单击某个工作流名称后，扣子平台将进入工作流可视化编排页面，如图 3-4 所示。该页面的布局主要由画布（主体区域）和工具栏（下方区域）两个核心区域构成。其中，画布是页面的核心工作区，用户主要在此区域通过拖曳节点、连接节点来构建和编排工作流的业务逻辑；工具栏位于画布下方，集成了辅助编排的各项快捷功能，包括切换交互模式、注释、优化布局、导出为图片、缩略图、添加节点、调试工作流、试运行等。



图 3-4 工作流编排页面

3.3.2 节点基本操作

新建工作流后，系统会默认初始化一个“开始”节点和一个“结束”节点，分别作为整个流程的数据输入与输出端口。节点的基本操作主要包含添加节点、建立连接和设置参数，下面将分别介绍这些操作方法。

1. 添加节点

在编排页面的下方工具栏中，单击“添加节点”按钮，系统会弹出节点类型菜单，选中目标节点即可将其添加到画布中。

例如，在“生成文章摘要信息”的工作流中，生成文章摘要的工作由大模型来完成，因此，需要添加一个大模型节点，如图 3-5 所示，在“添加节点”菜单中选择“大模型”节点。



图 3-5 添加节点

单击“大模型”后，画布中即可添加一个大模型节点，如图 3-6 所示。节点添加成功后，用户可手动拖曳调整其位置；也可直接单击工具栏中的“优化布局”按钮，由系统自动对齐和排列画布中的所有节点。

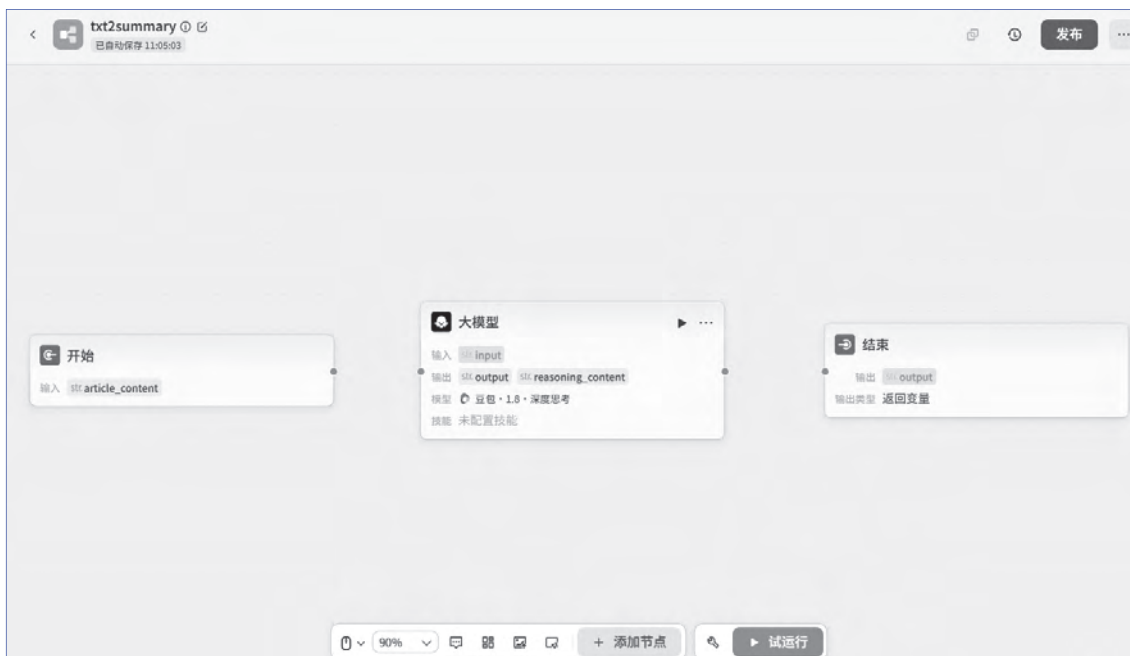


图 3-6 添加大模型节点

2. 建立连接

通过工具栏添加到画布的节点，在初始状态下是相互独立的。只有通过“连线”将各个节点按业务逻辑串联起来，数据才能流转，工作流也才能按照既定的先后顺序执行。

将鼠标指针移至源节点（前置节点）右侧的连接点，按住左键拖曳至目标节点（后置节点）左侧的连接点后松开，即可完成连线的建立。单击连接线，按 Delete 键可删除不需要的连接。

在“生成文章摘要”的工作流中，大模型的前置节点为“开始”，后置节点为“结束”。按照上述方法，先将“开始”节点连接至“大模型”节点，再将“大模型”节点连接至“结束”节点。完成连接后的工作流如图 3-7 所示。

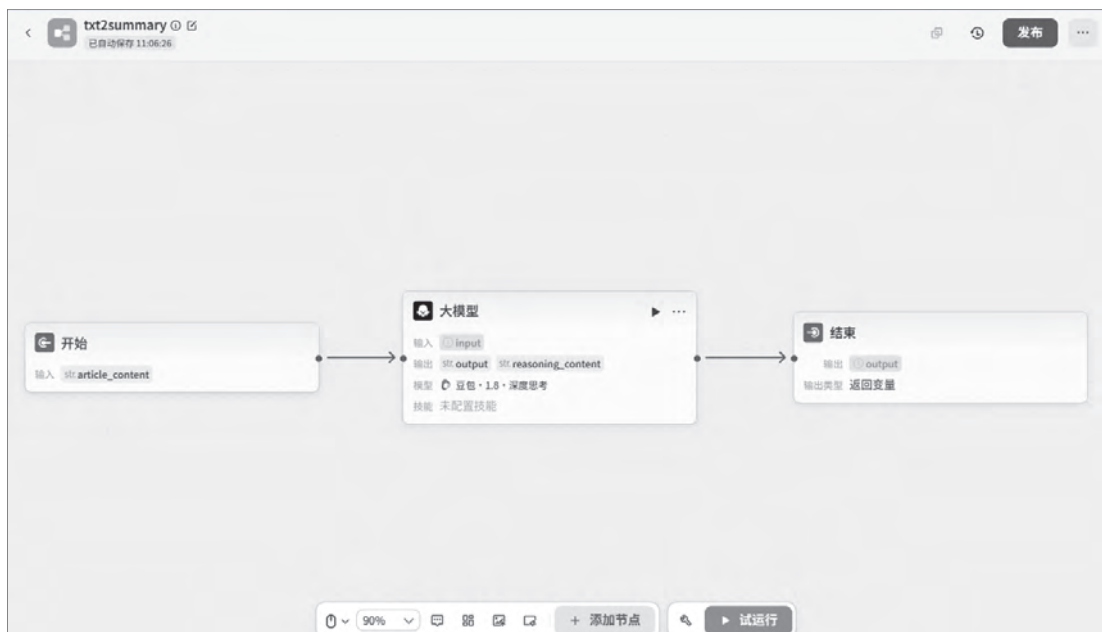


图 3-7 建立连接

提示

直接在源节点的连接点上按住鼠标左键，将连线拖曳至画布的空白区域后松开，系统同样会弹出“选择节点”菜单。通过此方法新建的节点，系统会自动为其与源节点建立连接线，省去了后续手动连线的步骤，操作更为高效。读者可自行操作一下。

3. 设置参数

添加节点并完成连接后，仅搭建了工作流的“骨架”。要使工作流真正运转，还需要进行参数设置，即赋予各节点具体的执行指令与数据处理。尽管不同类型节点参数各异，但其设置方法基本一致。在画布中单击目标节点，页面右侧即可弹出该节点的配置对话框，所有的参数配置均在此处完成。

以“生成文章摘要”工作流为例，单击画布中的“大模型”节点，右侧配置对话框中展示了该节点的可配置项，主要包括模型（如豆包、DeepSeek、GLM 等）、系统提示词、用户提示词等，如图 3-8 所示。

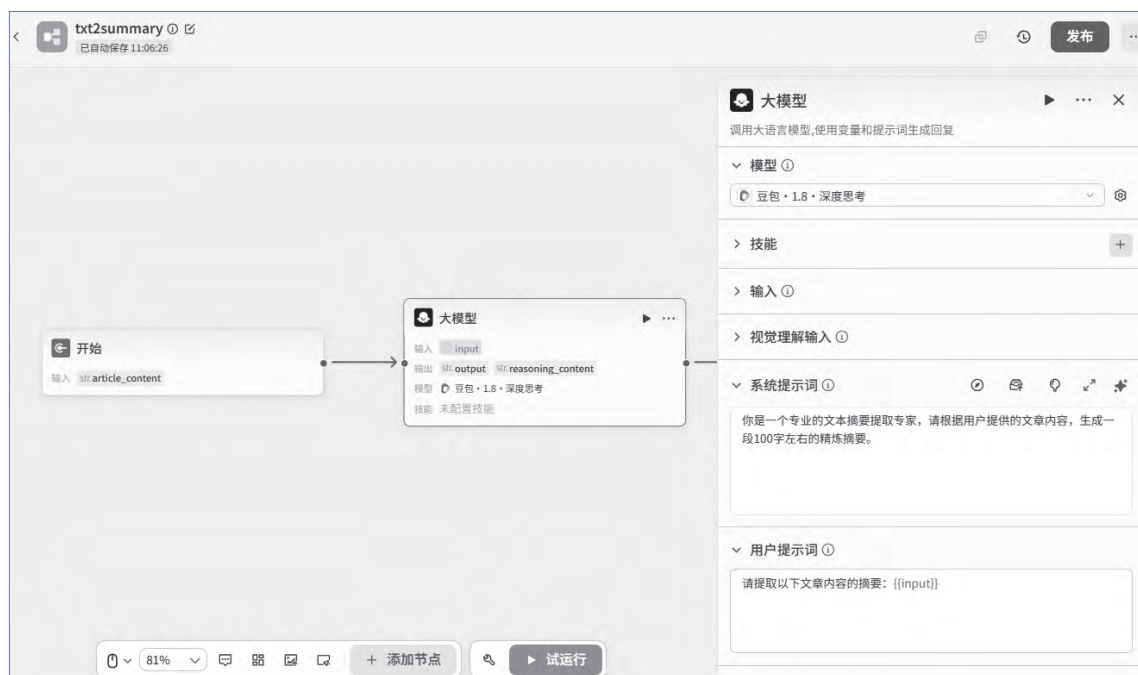


图 3-8 设置参数

系统提示词用于为大模型提供全局性的系统级指导，主要用于设定人设、任务边界与回复逻辑。例如，可输入：“你是一个专业的文本摘要提取专家，请根据用户提供的文章内容，生成一段 100 字左右的精炼摘要。”

3.3.3 数据流转

workflow 节点之间的连线不仅定义了任务的执行顺序，更是数据流动的“管道”。这种在节点之间传递的数据，在扣子平台中被称为“变量”。理解并正确配置变量，是让 workflow 真正运转起来的关键。按照数据流向的不同，变量可以分为输入变量和输出变量两类。

- **输入变量：**节点执行任务所需要的“原材料”。它的值可以是上游节点通过连线传递过来的动态数据，也可以是开发者在配置对话框中手动填写的固定文本。
- **输出变量：**节点完成任务后产出的“成品”。节点处理完输入的数据后，会将结果封装在输出变量中，以便传递给下一个节点继续处理。

外部数据首先进入“开始”节点的输入变量，随后依次流经各个中间处理节点的输入与输出变量，最终汇聚到“结束”节点的输入变量并输出给用户，从而形成一条完整的数据链路。

接下来，基于前文搭建的“生成文章摘要” workflow 骨架，介绍如何通过配置变量打通节点间的数据链路。

1) 配置开始节点

单击“开始”节点，在右侧参数配置对话框中进行如下设置，如图 3-9 所示。

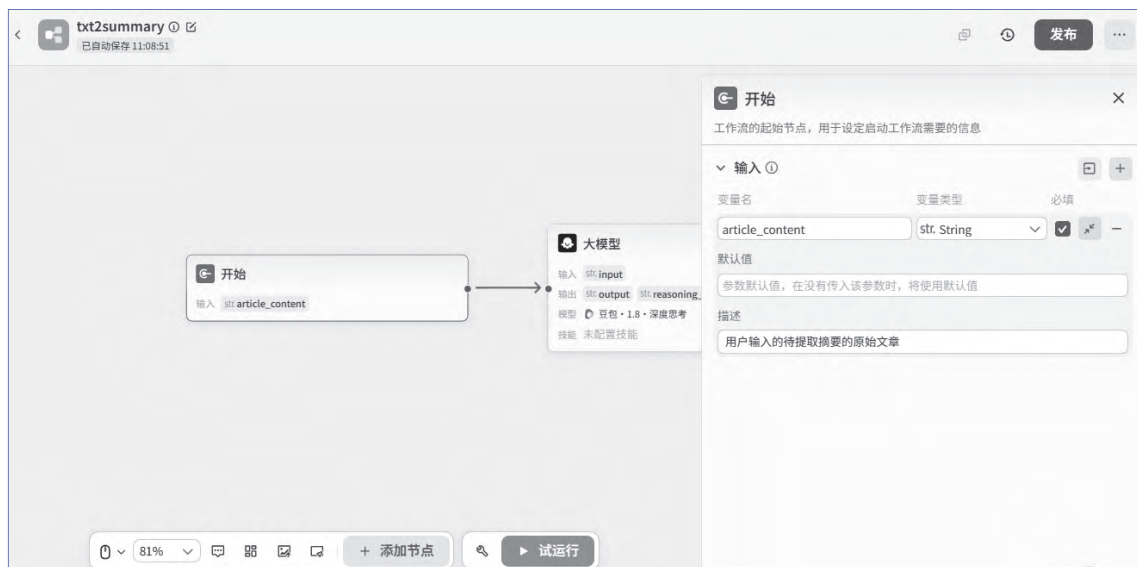


图 3-9 设置开始节点的输入参数

- 步骤 01** 修改变量标识：将默认变量名“input”修改为“article_content”，并勾选“必填”选项。
- 步骤 02** 展开详细配置：单击变量所在行右侧的 图标，显示详细配置项。
- 步骤 03** 添加描述信息：填写该变量的用途说明（如“用户输入的待提取摘要的原始文章”），这有助于大模型更好地理解参数上下文。

提示

大模型节点默认有一个名为 input 的输入参数，还可以单击“输入”右侧的“+”按钮添加新的输入变量。

2) 配置大模型节点

大模型节点参数较多，本例中仅配置和数据流转相关的几个参数，例如，输入参数、用户提示词及输出。

用户提示词是用户在对话界面主动输入、发送给智能体的指令与问题，是触发智能体响应与功能调用的直接入口。在普通单轮对话中，用户提示词是用户直接输入的文字；但在工作流场景下，它通常是一个“固定指令文本 + 动态输入变量”的合体。它充当了将上游数据具体化为大模型指令的“组装模板”。

单击“大模型”节点，在右侧参数配置对话框中设置相关参数，如图 3-10 所示。

配置大模型节点的具体操作步骤如下。

- 步骤 01** 绑定输入变量：单击输入变量 input 右侧的引用按钮 ，在弹出的下拉列表中选择“开始”节点下的“article_content”，即将开始节点接收的数据传递给大模型节点，作为其输入变量。
- 步骤 02** 设置用户提示词：在“用户提示词”输入框中输入指令。例如，“请提取以下文章内容的摘要：{{input}}”。此处通过将“任务指令（提取摘要）”与“动态数据（{{input}}）”在自然语言层面进行强行绑定，让大模型既明白数据是什么，又明白要对数据执行什么操作。



图 3-10 设置大模型节点的输入参数

需要注意的是，此处的“{{input}}”引用的是大模型的输入变量“input”，因此必须与大模型节点的输入变量名保持一致。在实际输入时，只需要在英文输入法状态下输入左半边花括号“{”，系统便会自动弹出变量补全提示，单击选择相应变量即可完成快速拼接，如图 3-11 所示。

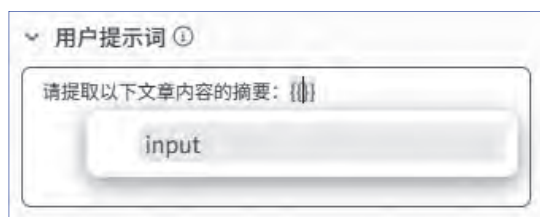


图 3-11 拼接输入变量

完成上述配置后，当 workflow 被触发运行时，系统会自动将“开始”节点接收到的真实文章文本替换掉 {{input}} 占位符，从而将动态数据真正交予大模型处理。

提示

大模型节点的“输入变量”主要用于在底层建立节点间的数据传输通道；而在“用户提示词”中拼接变量，则是为了将动态数据嵌入到自然语言指令中，为其赋予具体的语义上下文。只有将数据放入提示词模板中，大模型才能准确地理解它拿到的是什么数据、需要执行什么操作，而不仅是一个冷冰冰的参数值。

大模型节点默认有一个名为“output”的输出变量，可以在输出区域中配置输出的格式，支持文本、Markdown、JSON 三种格式的输出，默认为 JSON 格式，如图 3-12 所示。



图 3-12 大模型节点的输出配置

3) 配置结束节点

作为工作流的最终出口，“结束”节点的配置相对简单。其核心在于设定工作流向用户交付结果，平台提供了“返回变量”和“返回文本”两种模式。

- 返回变量：直接将上游节点传递过来的变量本身作为最终结果输出，适用于无须额外加工、直接输出数据的场景。
- 返回文本：允许开发者自定义一段文本模板，并在其中通过 {{ 变量名 }} 的语法灵活引用上游变量。这种方式常用于对最终结果进行格式化包装，比如在摘要前加上“以下是为您生成的摘要：”等引导语，使工作流按照预设的样式和结构输出，提升用户的阅读体验。

单击结束节点，然后在右侧参数配置对话框中单击输出变量 output 右侧的引用按钮 ，在弹出的下拉列表中选择“大模型”节点下的“output”作为输出变量，如图 3-13 所示。

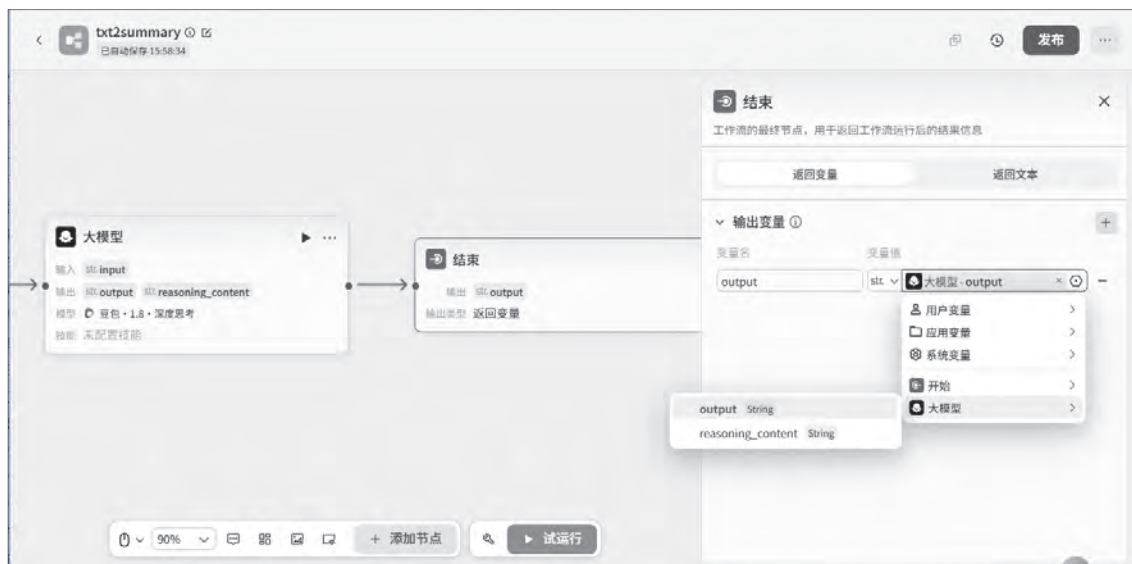


图 3-13 输出节点的配置

提示

具备深度推理能力的大模型的输出变量中，output 仅包含大模型最终生成的结果文本；而 reasoning_content 则额外记录了模型在得出该结果前的完整推理与思考过程。

3.3.4 工作流的测试与发布

工作流编排完成之后，需要进行工作流的试运行和发布操作。

1) 测试运行工作流

单击画布下方工具栏中的“试运行”按钮，在右侧弹出的试运行面板中输入文章内容，最后单击下方“试运行”按钮，即可测试运行工作流，如图 3-14 所示。



图 3-14 试运行 workflow

运行结果如图 3-15 所示。

2) 发布 workflow

workflow 试运行正常后，可单击网页右上角的“发布”按钮，在弹出的窗口中分别填写版本号、版本描述等信息，最后单击“发布”按钮即可发布 workflow，如图 3-16 所示。只有发布之后的 workflow，才能被添加到智能体中。

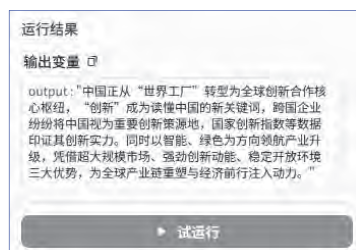


图 3-15 workflow 运行结果



图 3-16 发布 workflow



3.4 常见 workflow 节点

扣子平台提供了丰富的节点类型，覆盖业务逻辑、输入 & 输出、数据库、知识库 & 数据、图像处理等多个类别，如图 3-17 所示。设计 workflow 的核心在于梳理 workflow 的业务逻辑和节点选型。本节将重点介绍几种常见节点的功能特性与适用场景，助力读者在实际开发中高效构建业务流程。



图 3-17 扣子平台提供的节点类型

3.4.1 大模型

大模型节点是智能体实现智能化交互的核心基础单元，也是所有智能体 workflow 中最常用的节点。通过调用平台内置的豆包、DeepSeek、GLM 等大模型，不仅能够完成用户意图理解、自然对话回复、文本内容生成等基础任务，还能实现专业内容总结、逻辑推理判断、信息改写润色、多轮对话上下文承接、结构化数据输出等高阶 AI 能力。

大模型节点广泛应用于智能客服（解析用户意图、生成连贯回复）、内容创作（撰写文案、润色文本）、数据分析（提取结构化信息、推理判断）、教育（生成学习资料、答疑）、医疗（健康咨询、病历总结）、电商（商品描述生成、评价分析）等场景。

开发者在此节点中可选择模型类型、添加技能、配置系统提示词与用户提示词，并设置输出格式，如图 3-18 所示。



图 3-18 大模型节点的配置

3.4.2 插件

插件节点是智能体拓展能力边界、对接外部真实世界的关键功能单元。它通过封装和调用各类第三方 API 接口，赋予智能体获取实时数据与执行实际动作的能力。该节点广泛应用于信息查询（如实时搜索、最新新闻、数据核对）、办公协同（如文档处理、表格解析、发送邮件）、生活服务（如天气查询、快递追踪、日程管理）、业务系统对接（如 CRM 调用、ERP 数据交互、第三方平台联动）、开发运维（如接口测试、数据采集、状态监测）、教育科研（如文献检索、数据统计、专业工具计算）等场景。

扣子平台的插件市场中提供了丰富的插件资源，涵盖智能硬件、新闻阅读、便利生活、图像等多个领域。开发者单击具体的插件，可查看详细的参数说明与调用示例，如图 3-19 所示。



图 3-19 插件市场

3.4.3 代码

代码节点是工作流中实现复杂逻辑处理与底层数据加工的核心扩展单元。通过编写代码，弥补了常规节点在精确计算与严格逻辑控制方面的不足，既能胜任数据清洗、数组处理、JSON 解析与格式转换等精细化操作，又能实现接口签名加密、自定义算法、复杂条件分支等高阶能力。目前，扣子平台的代码节点支持 Python 和 JavaScript 两种编程语言。

如图 3-20 所示，添加一个代码节点用于去除用户输入内容首尾的空格与换行符，具体操作步骤如下。

- 步骤 01** 添加一个“代码”节点并单击该节点，展开右侧参数配置对话框。
- 步骤 02** 在代码节点配置对话框中，单击“代码”右侧的“在 IDE 中编辑”按钮，进入代码编写界面。

步骤 03 从语言下拉列表中选择所需要的编程语言并编写代码。在编写逻辑时，通过 `params = args.params` 获取全部输入参数并赋值给变量 `params`，后续可使用 `params['key']` 的形式精准提取特定参数的值进行后续处理。

步骤 04 代码编写完成后可单击“测试代码”按钮进行测试。



图 3-20 代码节点配置

3.4.4 选择器

选择器节点是 workflow 实现条件路由和逻辑分流的核心控制单元，是搭建复杂业务流程中最常用的判断节点。它根据预设条件的判断结果，自动将数据流精准导向对应的执行分支。该节点不仅支持基础的单条件判断，还能灵活应对多条件组合筛选与分支优先级匹配等复杂逻辑场景。

如图 3-21 所示，判断用户输入的内容中是否包含链接。若包含，则路由至“链接读取”插件节点；若不包含，则直接流向“大模型”节点。具体操作步骤如下。



图 3-21 选择器节点配置



- 步骤 01** 添加“选择器”节点并单击该节点，展开右侧参数配置对话框。
- 步骤 02** 在面板中配置条件分支，指定判断变量为上一个节点的输出，判断逻辑为“包含”，填写目标匹配值为“http”（链接一般都包含“http”，故此处填“http”）。
- 步骤 03** 将配置好的条件分支出口，分别连接至对应的下游处理节点。

在单个分支内，可通过“+ 新增”按钮添加多个判断条件，然后选择逻辑关系（且、或）指定判断条件。如果要新增独立的条件分支，则单击“条件分支”旁的“+”按钮即可。单击条件或条件分支后面对应的“-”按钮可进行删除，如图 3-22 所示。



图 3-22 新增判断条件和条件分支

3.4.5 意图识别

意图识别节点是智能体实现用户意图理解、对话场景精准分流的核心理解单元，是构建多场景、多任务智能交互工作流的关键基础节点。例如，在智能客服系统中，通过理解与咨询意图，自动跳转到售前、售后不同的服务流程。使用意图节点通过对用户输入内容进行语义解析与意图匹配，能够快速识别用户当前的对话目的与需求类型，实现场景自动归类、问题分类、交互路径引导等。

如图 3-23 所示，根据用户输入识别用户意图是要提取文章摘要、关键词，还是要仿写。操作步骤如下。



图 3-23 意图识别节点配置

步骤 01 单击“意图匹配”右侧的“+”按钮增加匹配条件，并填写意图描述。如需要删除匹配条件，可单击匹配条件后面的⊖图标。

步骤 02 将每个意图分类的出口，分别连接至对应的下游处理节点。

意图节点和选择器节点虽然都可以实现分流控制，但二者在定位、作用逻辑、适用场景等方面都存在显著差异。意图节点从自然语言理解的角度解析用户“想做什么”，可处理模糊、口语化的输入；选择器节点侧重的是逻辑判断，判断数据是否匹配预设条件，针对明确的数值、字符串或布尔值进行精确的逻辑匹配。

实际工作流中，二者常常配合使用，先通过意图识别节点识别用户需求，再将识别结果作为选择器节点的输入条件，由选择器节点根据意图类型，跳转至对应的数据处理分支，实现“理解用户 + 控制流程”的目的。

3.4.6 变量聚合

当工作流涉及并行分支、循环执行或多节点输出时，数据往往会碎片化散落于不同节点与轮次中，难以直接统一调用或一次性输出。变量聚合节点专门用于解决此类“数据碎片化”问题，负责将多分支、多轮次产生的零散结果统一归集，整合成完整、规整、可直接调用的最终数据。在实际应用中，该节点常作为批处理、循环或选择器节点的下游，承担数据汇总的任务。

如图 3-24 所示，工作流经意图识别节点分流为三条支线；随后，变量聚合节点将这三个分支产生的零散结果归集为一个完整的数据集合 group1，并交由输出节点一次性返回给用户。



图 3-24 变量聚合节点配置

3.4.7 图像生成

在工作流中使用图像生成节点实现视觉内容创作与输出，赋予智能体从纯文本交互向多模态内容创作延伸的能力。该节点通过接收上游节点的文本描述（提示词），调用底层图像生成模型，将用户需求直接转化为高质量的视觉图像。该节点常用在广告物料生成、绘本创作、内容辅助创作等场景。

图像生成节点的效果高度依赖提示词的质量，因此在实际搭建中，上游通常会先连接一个“大模型节点”，由其对用户输入进行结构化处理与提示词优化，生成标准、规范的提示词后，再传入图像生成节点，如图 3-25 所示。



图 3-25 图像生成节点配置

3.4.8 画板

画板节点是扣子智能体中用于对图片进行二次编辑、合成、叠加、排版的视觉处理节点，属于多模态能力扩展节点。它可以把图片、文字、图标等元素，在虚拟画布上进行组合、布局，最终输出一幅完整的合成图。画板节点配置如图 3-26 所示。

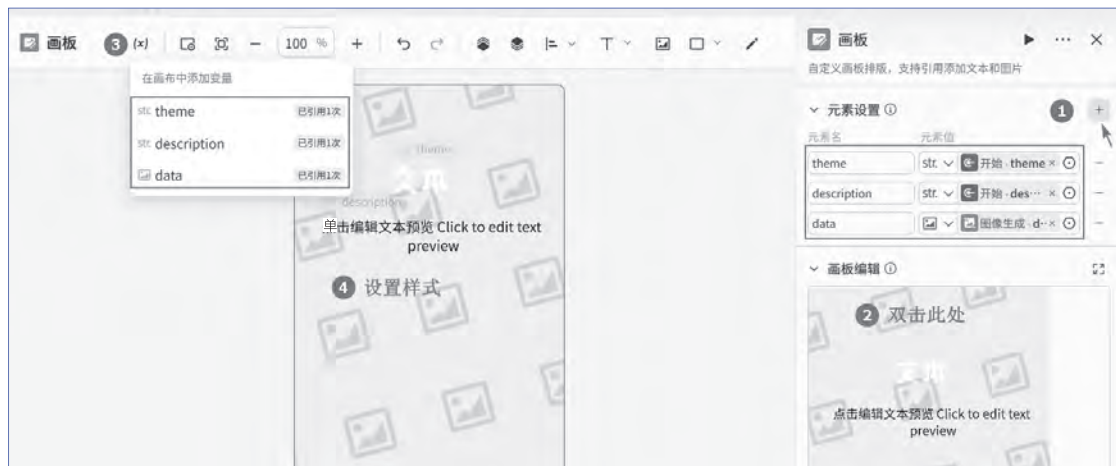


图 3-26 画板节点配置

例如，在 3.4.7 节工作流的基础上实现生成带文字的图片，具体操作如下。

- 步骤 01** 在工作流中增加一个画板节点，并置于图像生成节点之后。
- 步骤 02** 为开始节点增加一个输入变量，命名为 `description`，代表图片上的描述文字。
- 步骤 03** 单击“画板”节点，在右侧参数配置对话框中增加三个元素，分别代表图片主题、描述和图片。

步骤 04 双击参数配置对话框中的“画板编辑”，进入自定义画板页面。点击引用变量按钮(x)，分别引用主题（theme）、描述（description）和图片（data）三个变量，并调整位置和样式。

运行结果如图 3-27 所示。



图 3-27 运行结果

3.4.9 循环

循环节点主要用于实现批量处理与自动化迭代，适用于处理同质化任务的重复执行场景。该节点支持按次数循环、按数组循环和无限循环三种模式。如图 3-28 所示，在该 workflow 中使用循环节点生成多幅分镜图片。

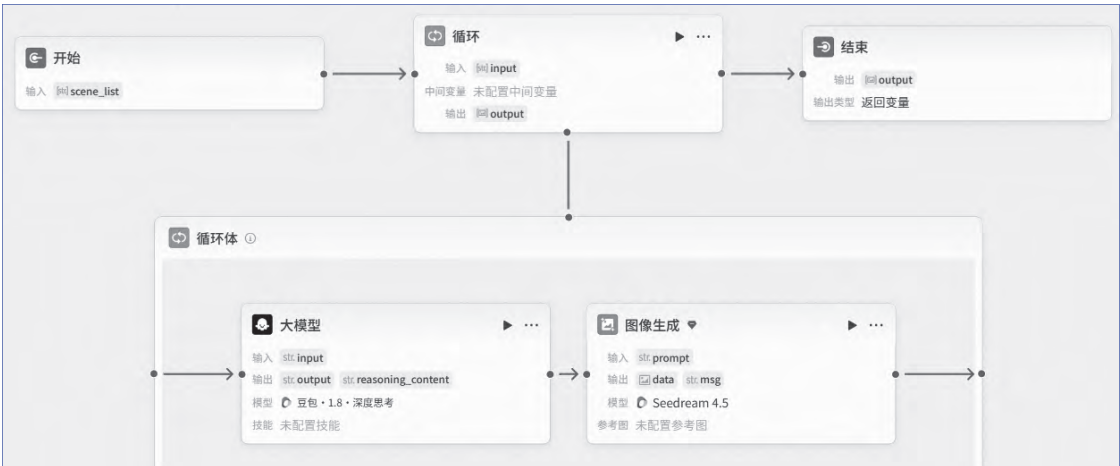


图 3-28 生成分镜图片 workflow

循环节点的循环数组为上游节点的输入变量，输出为图片生成节点的输出，该节点的配置如图 3-29 所示。

3.4.10 批处理

批处理节点是工作流中实现高并发、高效率数据处理的“加速器”。与循环节点的“逐个排队处理”不同，批处理节点的核心优势在于并行执行，它能够同时接收一组数据（数组），并行或分组调用相同的处理逻辑，从而大幅缩短大批量任务的执行耗时。关于批处理节点的具体使用方法请参见第 8 章。

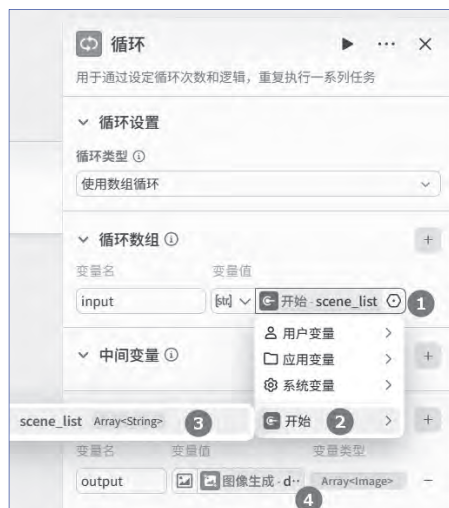


图 3-29 循环节点配置

提示

有剧情、有顺序、要继承上下文使用循环节点；无关联、纯批量、越快越好使用批处理节点。



3.5 本章小结

本章详细讲解了工作流的创建、节点参数配置与变量传递机制，并重点介绍了大模型、代码、选择器、意图识别、变量聚合、循环与批处理等核心节点的特性与适用场景。通过本章的学习，读者可以具备独立搭建结构化、高可靠工作流的能力。



3.6 实践练习

(1) 请搭建一个工作流，实现以下功能：接收用户输入的一段文本，自动判断其是“中文”还是“英文”。若为中文，则调用大模型生成内容摘要；若为英文，则调用大模型将其翻译为中文。最后通过结束节点将处理结果返回给用户。

(2) 请设计一个工作流，实现以下功能：接收一个包含多个商品名称的列表数据，利用循环节点（或批处理节点），为每个商品名称自动生成一幅宣传配图，并在配图中显示商品名称。