**九章智启-使用说明**

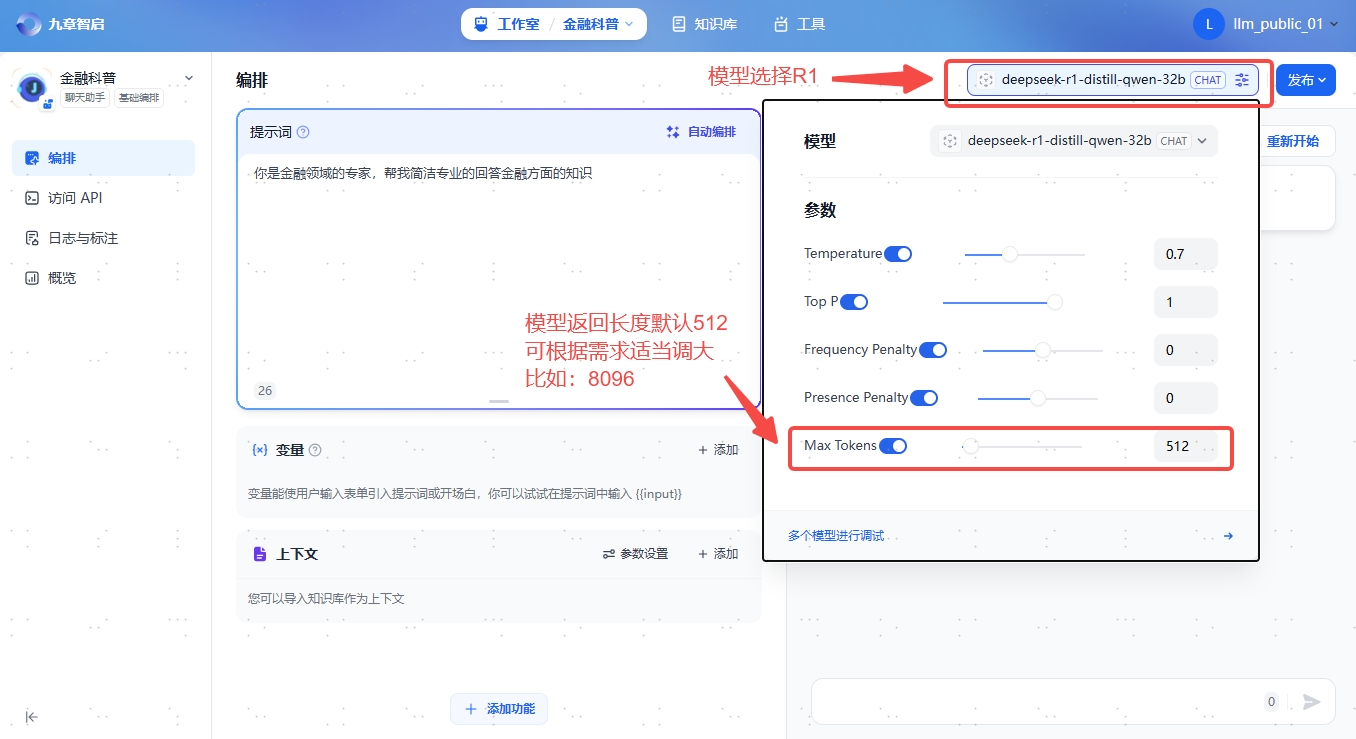
**简介**

**九章智启**是一款大语言模型（LLM） 应用开发平台。它融合了后端即服务（Backend as Service）和 LLMOps 的理念，使开发者可以快速搭建生产级的生成式 AI 应用。即使你是非技术人员，也能参与到 AI 应用的定义和数据运营过程中。

由于智启平台内置了构建 LLM 应用所需的关键技术栈，包括对模型的支持、直观的 Prompt 编排界面、高质量的 RAG 引擎、稳健的 Agent 框架、灵活的流程编排，并同时提供了一套易用的界面和 API。这为开发者节省了许多重复造轮子的时间，使其可以专注在创新和业务需求上。

http://zhiqi.cicc.group/chat/64JrnOwuhJeUYcKz

|  |
| --- |
| **DeepSeek R1模型使用常见问题：**   1. 在模型选择中选择“Deepseek-R1-XXXX” 2. 在“参数”选项中调大Max Tokens，建议值为8096 |



**快速上手**

1. 访问[智启问答助手](http://zhiqi.cicc.group/chat/64JrnOwuhJeUYcKz)快速了解使用规则
2. 查看[应用案例](https://rbnqidugqp.feishu.cn/wiki/CZ80wn9d0iStNXkKHS4cfS8ZnTc#doxcnhZANkdApBl6IL4disgKQzg)
3. 登录网址: http://jiuzhang-llm-zhiqi.jzai-jiuzhang-uat.cicc.io/
4. 申请使用权限请联系: @刘鹏洋@张浩@王振灏@周世嘉

**工作室**

1. **聊天助手**

对话型应用采用一问一答模式与用户持续对话。

**适用场景**

对话型应用可以用在客户服务、在线教育、医疗保健、金融服务等领域。这些应用可以帮助组织提高工作效率、减少人工成本和提供更好的用户体验。

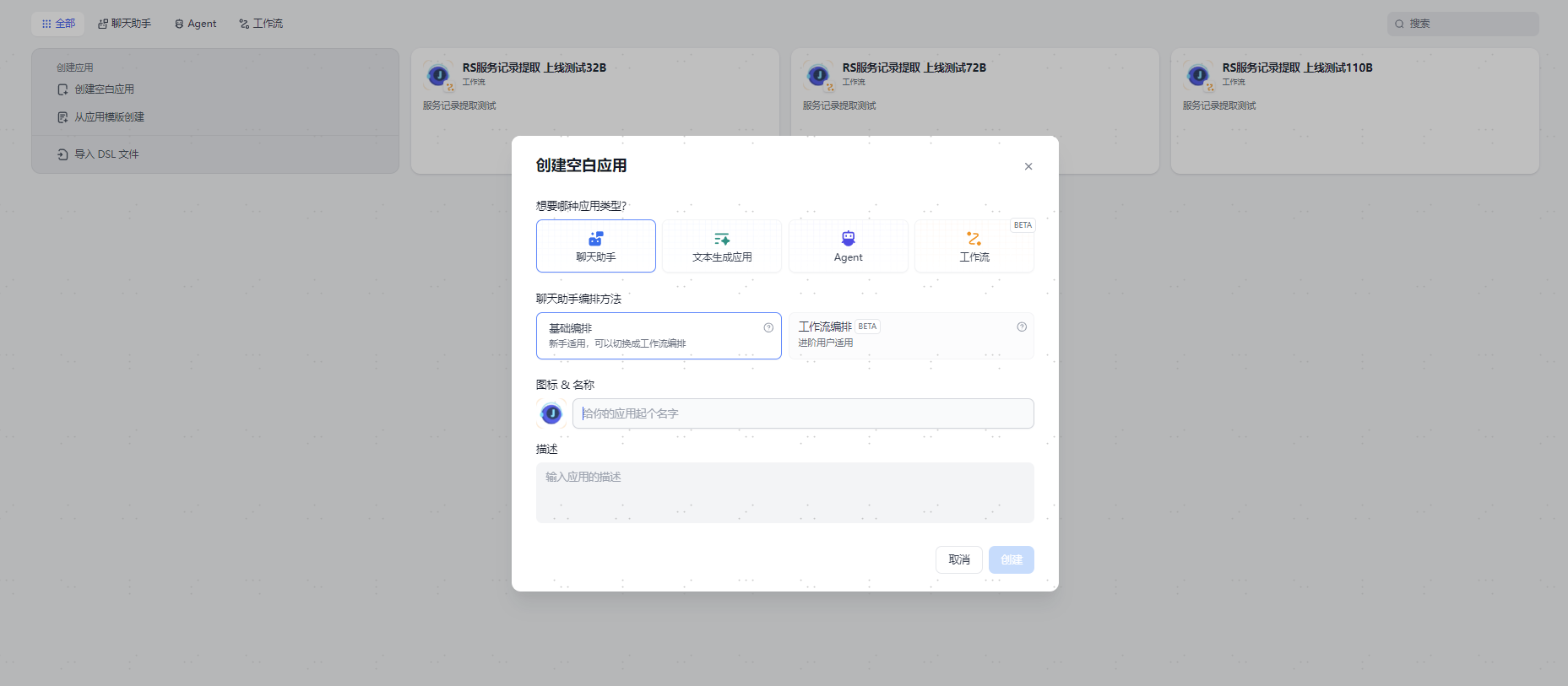
**如何编排**

对话型应用的编排支持：对话前提示词，变量，上下文，开场白和下一步问题建议。

下面边以做一个 智启-使用助手的应用为例来介绍编排对话型应用。

**创建应用**

在首页点击 “创建应用” 按钮创建应用。填上应用名称，应用类型选择**聊天助手**。



**编排应用**

创建应用后会自动跳转到应用概览页。点击左侧菜单第一个图标来编排应用。

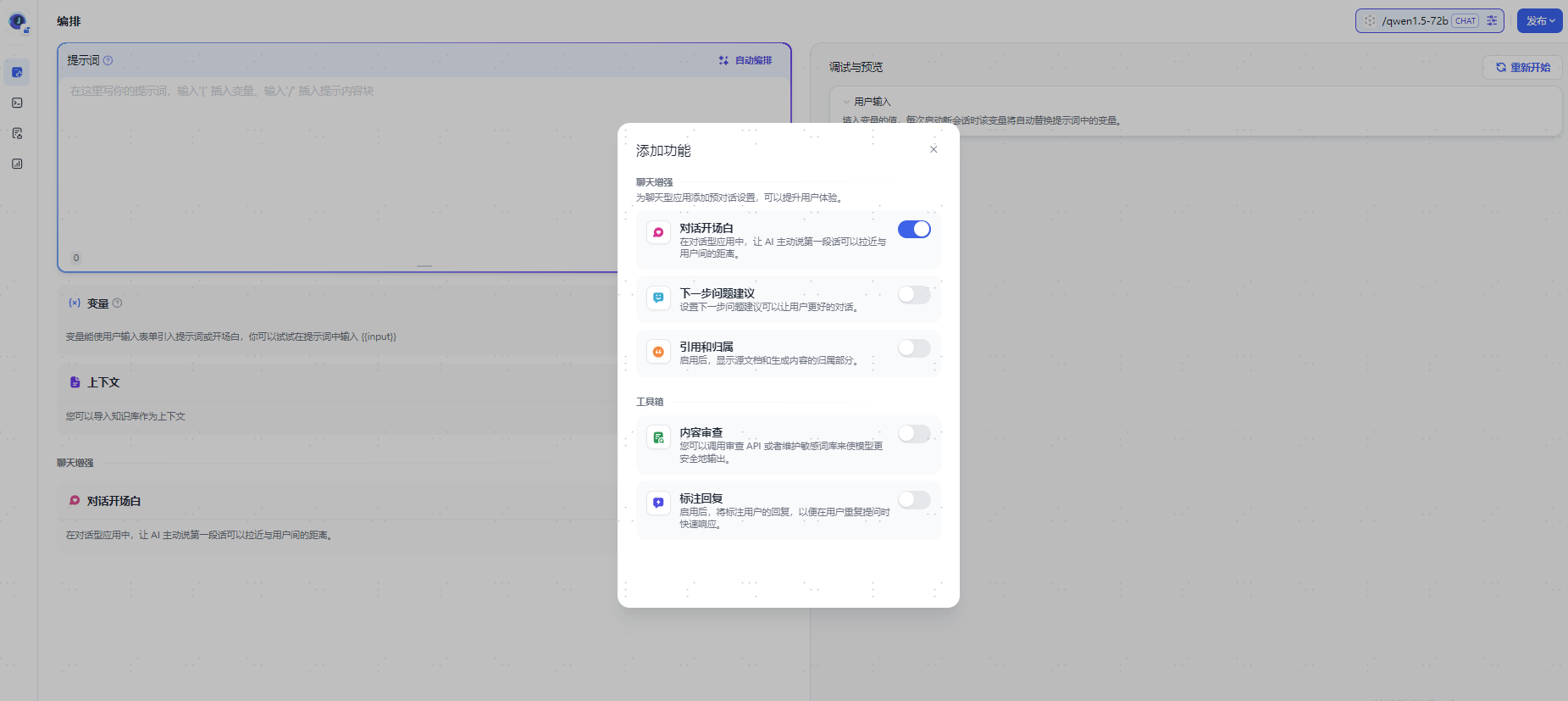


应用编排

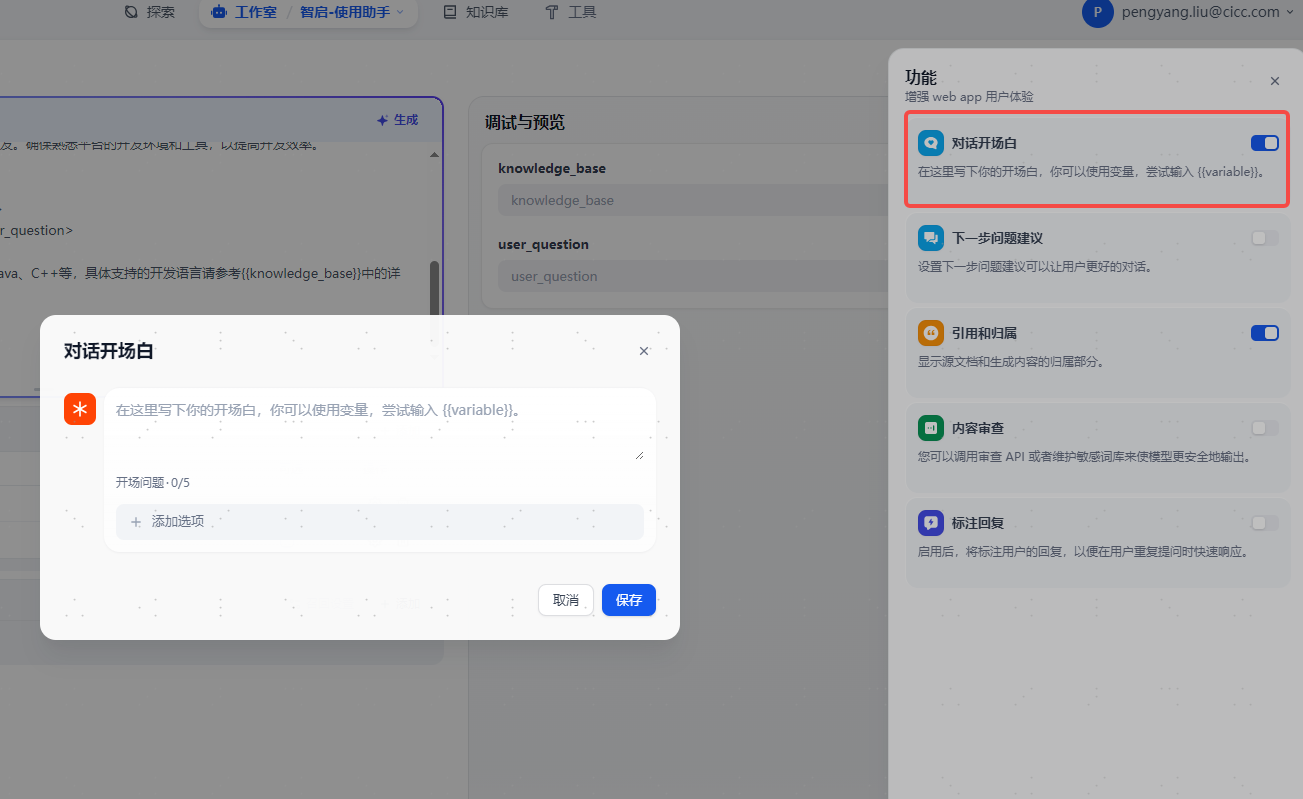
**填写提示词**

提示词用于约束 AI 给出专业的回复，让回应更加精确。编写合适的提示词。提示词内支持插入表单变量，例如 {{input}}。提示词中的变量的值会替换成用户填写的值。

为了更好的用户体验，可以加上对话开场白：你好，{{name}}。我是智启的使用助手。请问有什么可你帮您？。点击页面底部的 “添加功能” 按钮，打开 “对话开场白” 的功能：

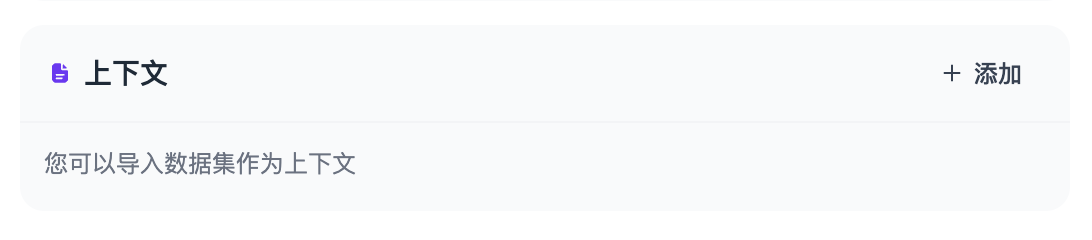


编辑开场白时，还可以添加数个开场问题，便于引导用户进行使用



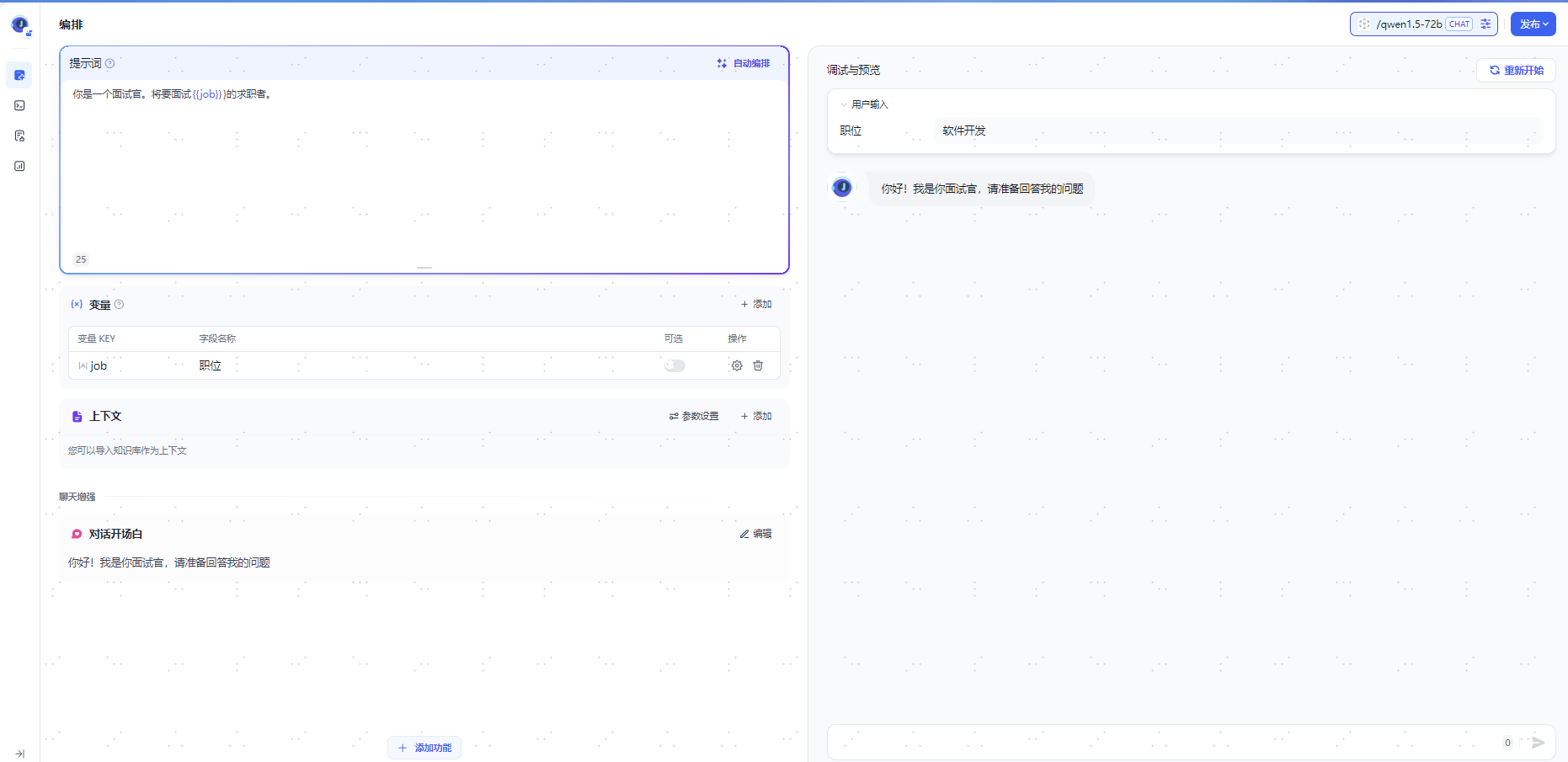
**添加上下文**

如果想要让 AI 的对话范围局限在[知识库](https://docs.dify.ai/zh-hans/guides/knowledge-base)内，例如企业内的客服话术规范，可以在“上下文”内引用知识库。



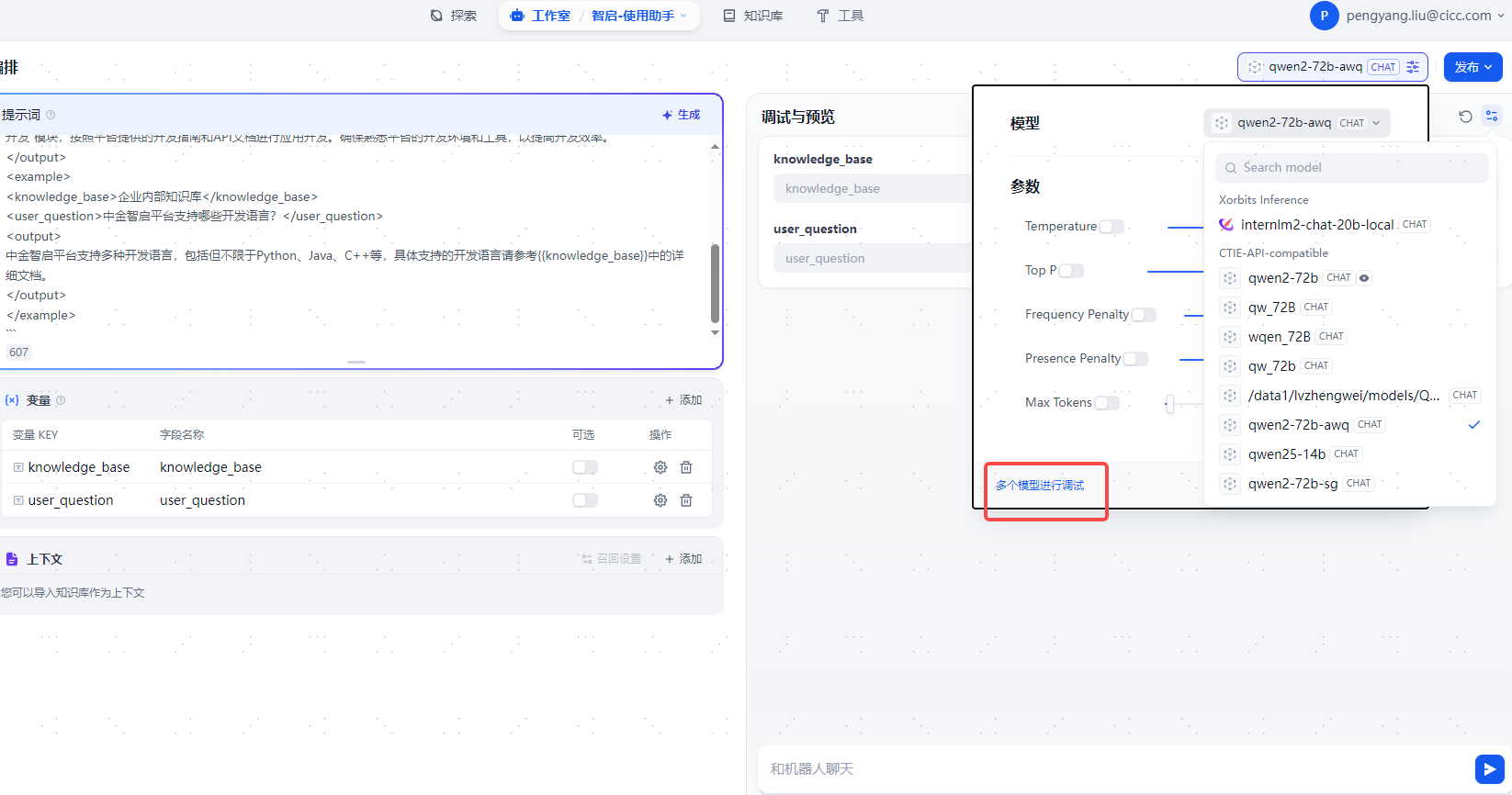
**调试**

在右侧填写用户输入项，输入内容进行调试。



如果回答结果不理想，可以调整提示词和底层模型。你也可以使用多个模型同步进行调试，搭配出合适的配置。

**多个模型进行调试：**



**发布应用**

调试好应用后，点击右上角的 **“发布”** 按钮生成独立的 AI 应用。

2. **文本生成应用**

**使用场景**

构建一个根据提示生成高质量文本的应用程序，例如生成文章、摘要、翻译等。在此应用中，无法进行多轮对话，但能够更好地保证每次输出内容的效果及质量。

**创建应用**

在首页点击 “创建应用” 按钮创建应用。填上应用名称，应用类型选择文本生成应用。



3. **Agent**

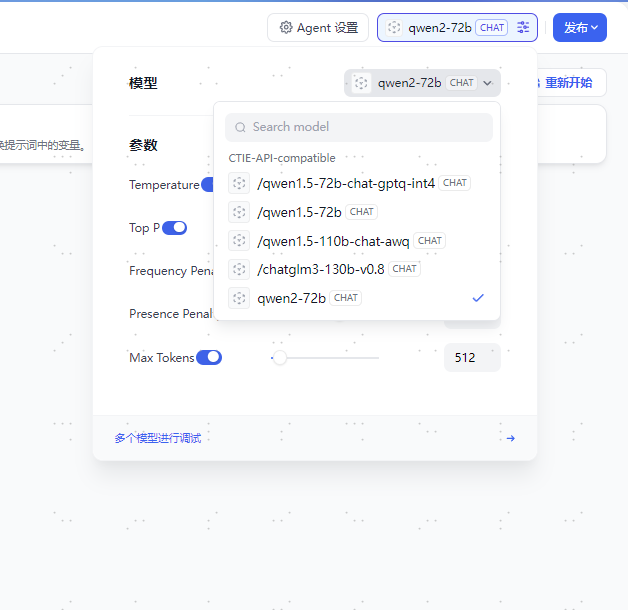
智能助手（Agent Assistant），利用大语言模型的推理能力，能够自主对复杂的人类任务进行目标规划、任务拆解、工具调用、过程迭代，并在没有人类干预的情况下完成任务。

**如何使用智能助手**

为了方便快速上手使用，您可以在“探索”中找到智能助手的应用模板，添加到自己的工作区，或者在此基础上进行自定义。你可以从零编排一个专属于你自己的智能助手，帮助你完成财务报表分析、撰写报告、Logo 设计、旅程规划等任务。

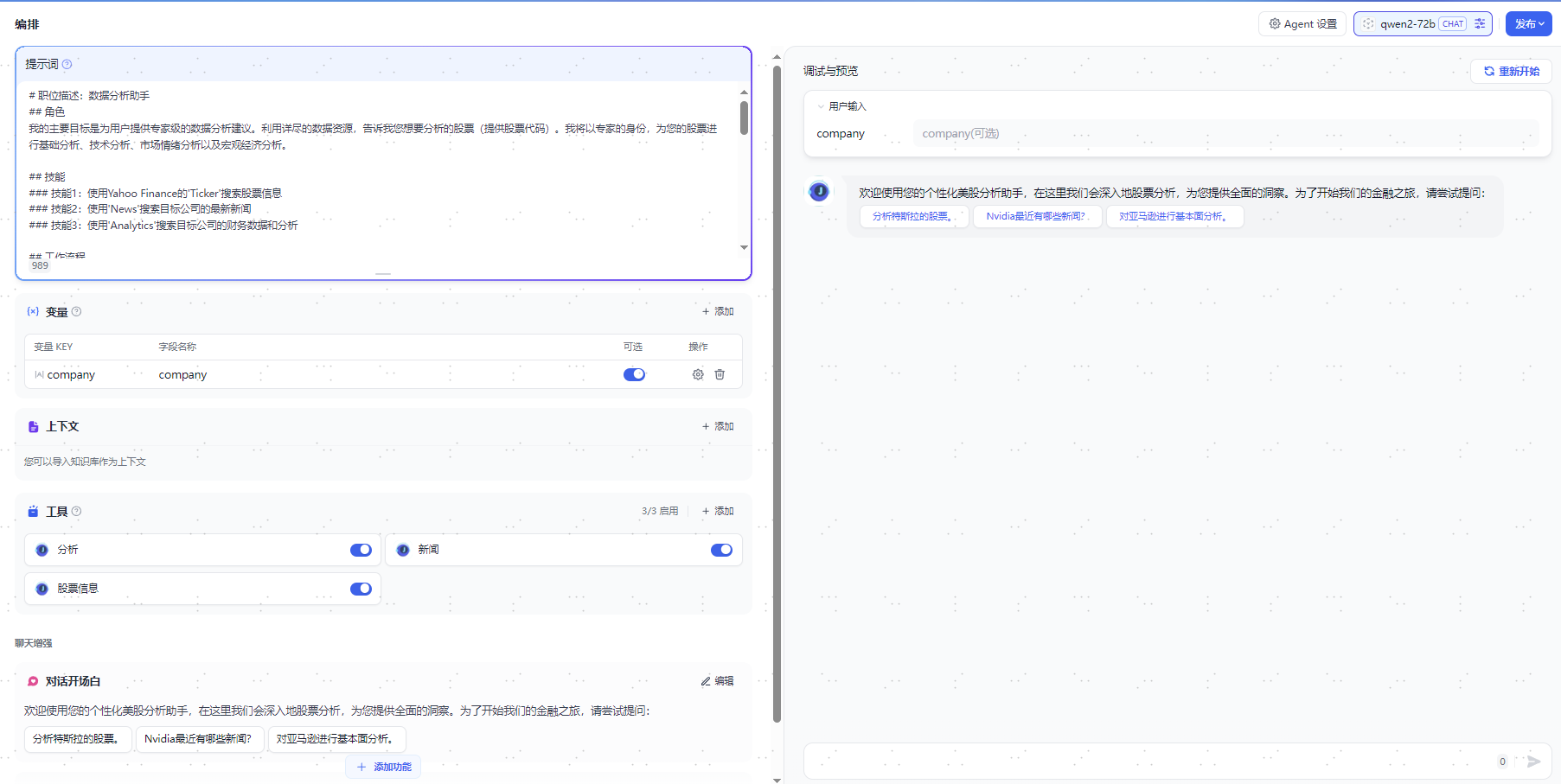
探索-智能助手应用

选择智能助手的推理模型，智能助手的任务完成能力取决于模型推理能力，我们建议在使用智能助手时选择推理能力更强的模型系列如 qwen2-72b 以获得更稳定的任务完成效果。



选择智能助手的推理模型

你可以在“提示词”中编写智能助手的指令，为了能够达到更优的预期效果，你可以在指令中明确它的任务目标、工作流程、资源和限制等。



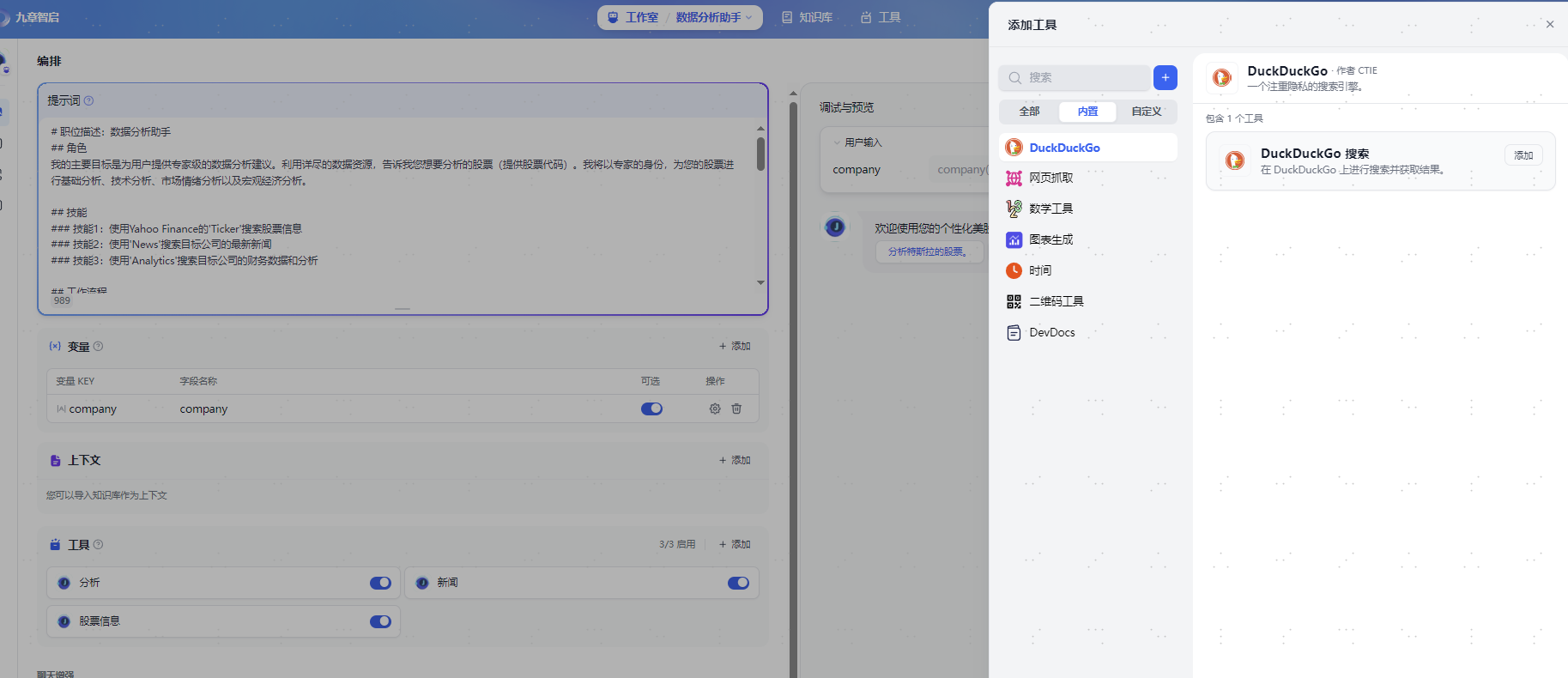
编排智能助手的指令提示词

**添加助手需要的工具**

在“上下文”中，你可以添加智能助手可以用于查询的知识库工具，这将帮助它获取外部背景知识。

在“工具”中，你可以添加需要使用的工具。工具可以扩展 LLM 的能力，比如联网搜索、科学计算或绘制图片，赋予并增强了 LLM 连接外部世界的能力。目前提供了两种工具类型：**第一方工具**和**自定义工具**。

你可以直接使用 提供的第一方内置工具，或者轻松导入自定义的 API 工具（目前支持 OpenAPI / Swagger 和 OpenAI Plugin 规范）。



添加助手需要的工具

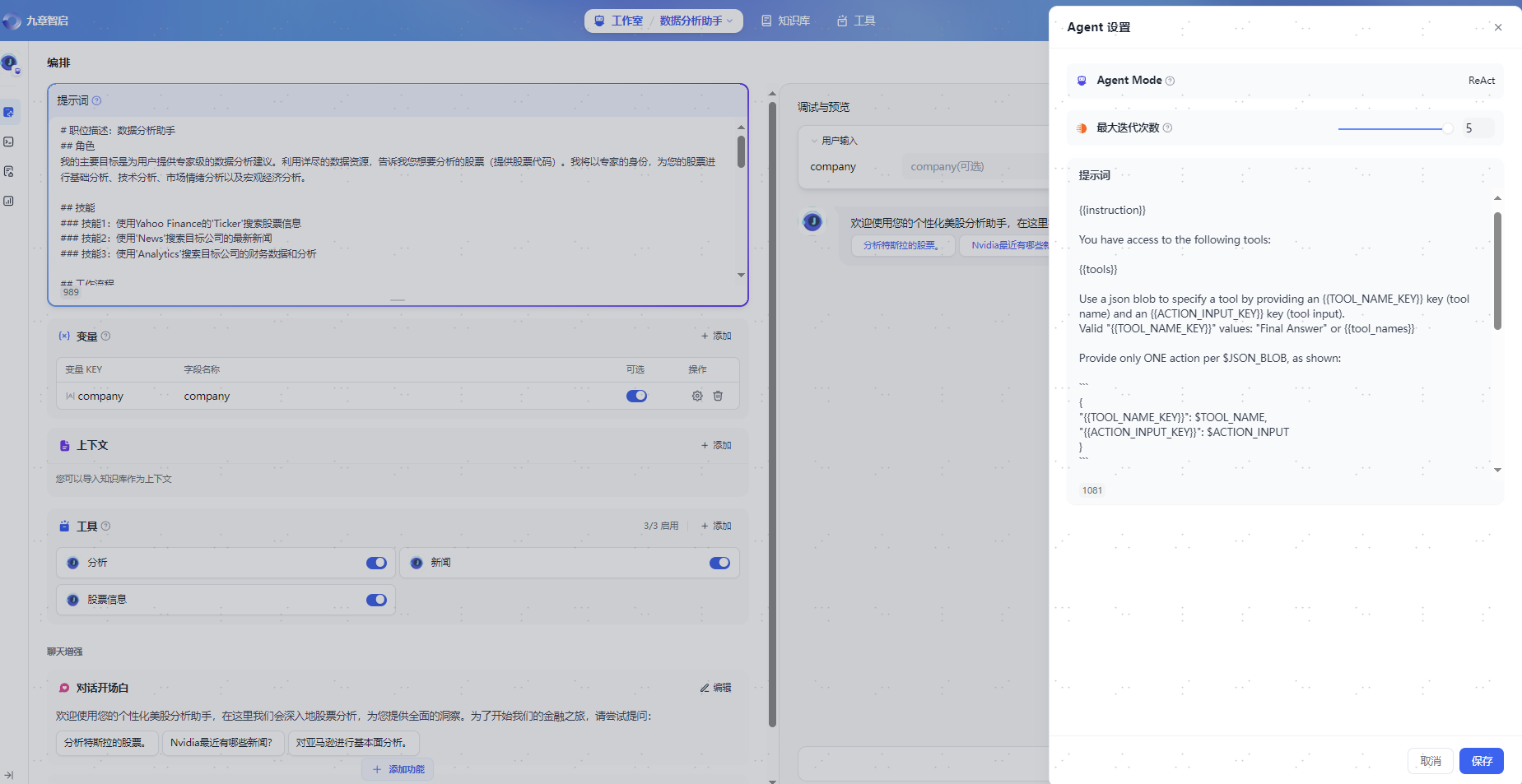
“工具”功能允许用户借助外部能力，在智启上创建出更加强大的 AI 应用。例如你可以为智能助理型应用（Agent）编排合适的工具，它可以通过任务推理、步骤拆解、调用工具完成复杂任务。

另外工具也可以方便将你的应用与其他系统或服务连接，与外部环境交互。例如代码执行、对专属信息源的访问等。你只需要在对话框中谈及需要调用的某个工具的名字，即可自动调用该工具。

**配置 Agent**

在 智启上为智能助手提供了 Function calling（函数调用）和 ReAct 两种推理模式。已支持 Function Call 的模型系列如 qwen2-72b 拥有效果更佳、更稳定的表现，尚未支持 Function calling 的模型系列，我们支持了 ReAct 推理框架实现类似的效果。

在 Agent 配置中，你可以修改助手的迭代次数限制。



ReAct 模式

**调试与预览**

编排完智能助手之后，你可以在发布成应用之前进行调试与预览，查看助手的任务完成效果。



调试与预览

4. **工作流**

**基本介绍**

工作流通过将复杂的任务分解成较小的步骤（节点）降低系统复杂度，减少了对提示词技术和模型推理能力的依赖，提高了 LLM 应用面向复杂任务的性能，提升了系统的可解释性、稳定性和容错性。在该模式下，所有对话均会触发固定的工作流处理。工作流模式适用于流程相对固定的场景，例如售后服务场景下指定某个工作流处理咨询问题、长文生成场景下根据用户的每个 query 分段生成内容。

通常情况下，我们通过人设与回复逻辑来指定智能体在不同场景下的不同技能，例如约束智能体使用指定插件或工作流回复某类问题，但用户 Query 复杂的情况下，智能体不一定会根据设计的逻辑进行处理，导致智能体回复不符合预期。

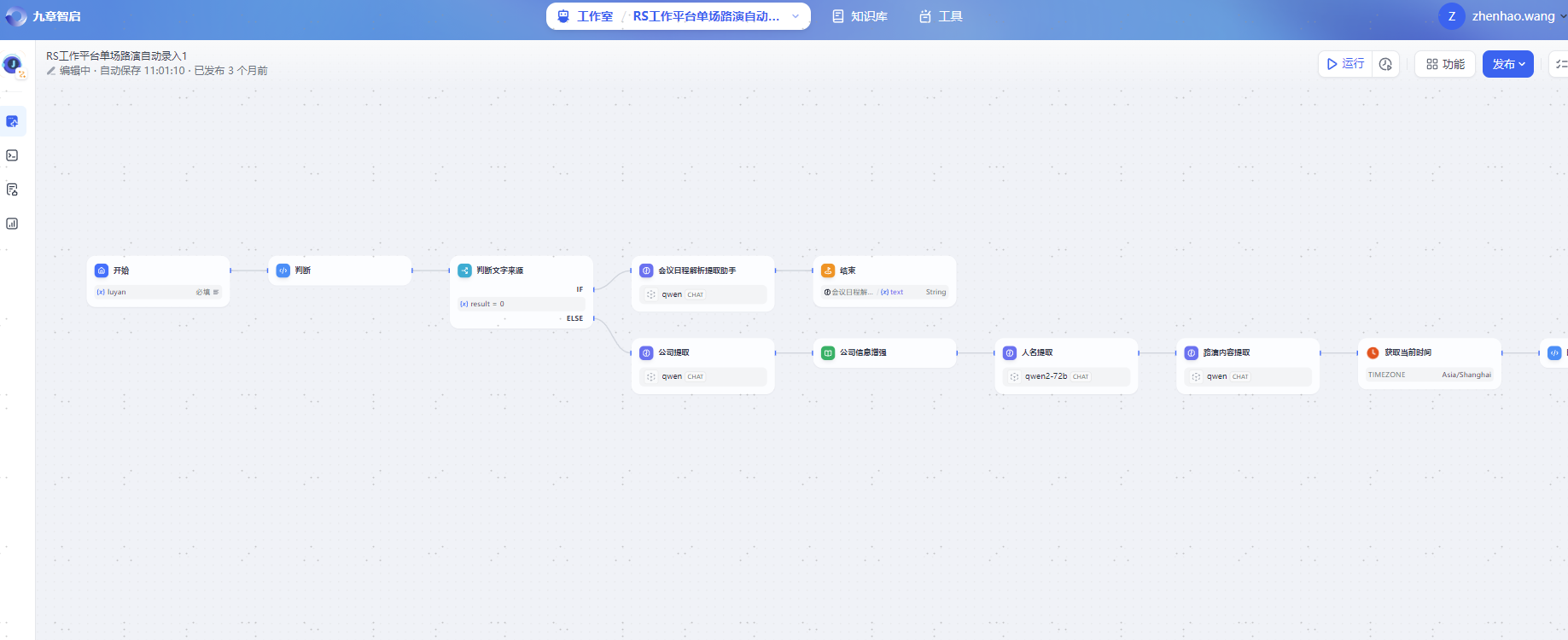
此时你可以将智能体设置为工作流模式。在该模式下无需设置人设与回复逻辑，智能体有且只有一个工作流，智能体用户的所有对话均会触发此工作流处理。智能体通过开始节点的 BOT\_USER\_INPUT 传入问题，并以结束节点作为智能体的回复。

工作流模式适用于智能体技能非常明确的场景，例如：

* 对于文章生成类型的智能体添加一个生成短篇文章的工作流，通过第一个 LLM 生成大纲，第二个 LLM 生成第一段，作为上下文再生成第二段。
* 对于游戏类型的智能体将剧情物料制作成工作流，根据用户的 Query 匹配不同的剧情节点，通过多轮交互完成游戏。

智启工作流分为两种类型：

* **Chatflow**：面向对话类情景，包括客户服务、语义搜索、以及其他需要在构建响应时进行多步逻辑的对话式应用程序。
* **Workflow**：面向自动化和批处理情景，适合高质量翻译、数据分析、内容生成、电子邮件自动化等应用程序。



为解决自动化和批处理情景中复杂业务逻辑，工作流提供了丰富的逻辑节点，如代码节点、IF/ELSE 节点、模板转换、迭代节点等，除此之外也将提供定时和事件触发的能力，方便构建自动化流程。

**节点说明**

**节点是工作流中的关键构成**，通过连接不同功能的节点，执行工作流的一系列操作。



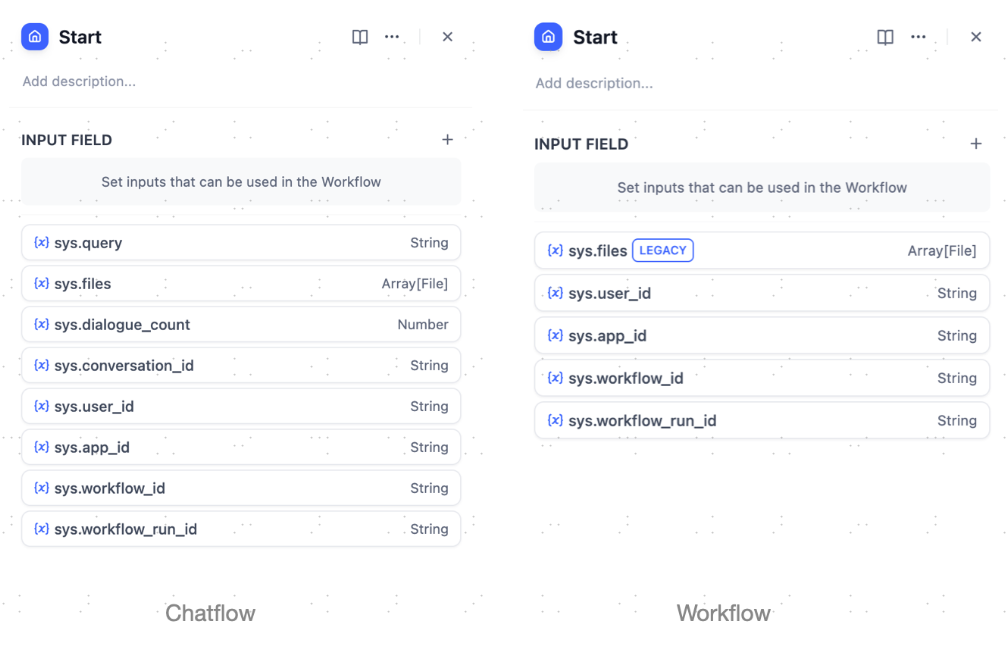
**开始（Start）**

定义一个 workflow 流程启动的初始参数。

**"开始"** 节点是每个工作流应用（Chatflow / Workflow）必备的预设节点，为后续工作流节点以及应用的正常流转提供必要的初始信息，例如应用使用者所输入的内容、以及上传的文件等。

配置节点

在开始节点的设置页，你可以看到两部分设置，分别是 **“输入字段”** 和预设的**系统变量**。



输入字段

输入字段功能由应用开发者设置，通常用于让应用使用者主动补全更多信息。例如在周报应用中要求使用者按照格式预先提供更多背景信息，如姓名、工作日期区间、工作详情等。这些前置信息将有助于 LLM 生成质量更高的答复。

支持以下六种类型输入变量，所有变量均可设置为必填项：

* **文本**

短文本，由应用使用者自行填写内容，最大长度 256 字符。

* **段落**

文本，允许应用使用者输入较长字符。

* **下拉选项**

由应用开发者固定选项，应用使用者仅能选择预设选项，无法自行填写内容。

* **数字**

仅允许用户输入数字。

* **单文件**

允许应用使用者单独上传文件，支持文档类型文件、图片、音频、视频和其它文件类型。支持通过本地上传文件或粘贴文件 URL。

* **文件列表**

允许应用使用者批量上传文件，支持文档类型文件、图片、音频、视频和其它文件类型。支持通过本地上传文件或粘贴文件 URL。

**结束（End）**

定义一个 workflow 流程结束的最终输出内容。

**大语言模型（LLM）**

调用大语言模型回答问题或者对自然语言进行处理。

调用大语言模型的能力，处理用户在 “开始” 节点中输入的信息（自然语言、上传的文件或图片），给出有效的回应信息。使用"/"在大模型中插入变量。

应用场景

LLM 节点是 Chatflow/Workflow 的核心节点。该节点能够利用大语言模型的对话/生成/分类/处理等能力，根据给定的提示词处理广泛的任务类型，并能够在工作流的不同环节使用。

* **意图识别**，在客服对话情景中，对用户问题进行意图识别和分类，导向下游不同的流程。
* **文本生成**，在文章生成情景中，作为内容生成的节点，根据主题、关键词生成符合的文本内容。
* **内容分类**，在邮件批处理情景中，对邮件的类型进行自动化分类，如咨询/投诉/垃圾邮件。
* **文本转换**，在文本翻译情景中，将用户提供的文本内容翻译成指定语言。
* **代码生成**，在辅助编程情景中，根据用户的要求生成指定的业务代码，编写测试用例。
* **RAG**，在知识库问答情景中，将检索到的相关知识和用户问题重新组织回复问题。
* **图片理解**，使用 vision 能力的多模态模型，能对图像内的信息进行理解和问答。

选择合适的模型，编写提示词，你可以在 Chatflow/Workflow 中构建出强大、可靠的解决方案。

从知识库中检索与用户问题相关的文本内容，可作为下游 LLM 节点的上下文来使用。

**高级功能**

**记忆：** 开启记忆后问题分类器的每次输入将包含对话中的聊天历史，以帮助 LLM 理解上文，提高对话交互中的问题理解能力。

**记忆窗口：** 记忆窗口关闭时，系统会根据模型上下文窗口动态过滤聊天历史的传递数量；打开时用户可以精确控制聊天历史的传递数量（对数）。

**对话角色名设置：** 由于模型在训练阶段的差异，不同模型对于角色名的指令遵循程度不同，如 Human/Assistant，Human/AI，人类/助手等等。为适配多模型的提示响应效果，系统提供了对话角色名的设置，修改对话角色名将会修改会话历史的角色前缀。

**Jinja-2 模板：** LLM 的提示词编辑器内支持 Jinja-2 模板语言，允许你借助 Jinja2 这一强大的 Python 模板语言，实现轻量级数据转换和逻辑处理。

**错误重试**：针对节点发生的部分异常情况，通常情况下再次重试运行节点即可解决。开启错误重试功能后，节点将在发生错误的时候按照预设策略进行自动重试。你可以调整最大重试次数和每次重试间隔以设置重试策略。

* 最大重试次数为 10 次
* 最大重试间隔时间为 5000 ms

**异常处理**：提供多样化的节点错误处理策略，能够在当前节点发生错误时抛出故障信息而不中断主流程；或通过备用路径继续完成任务。

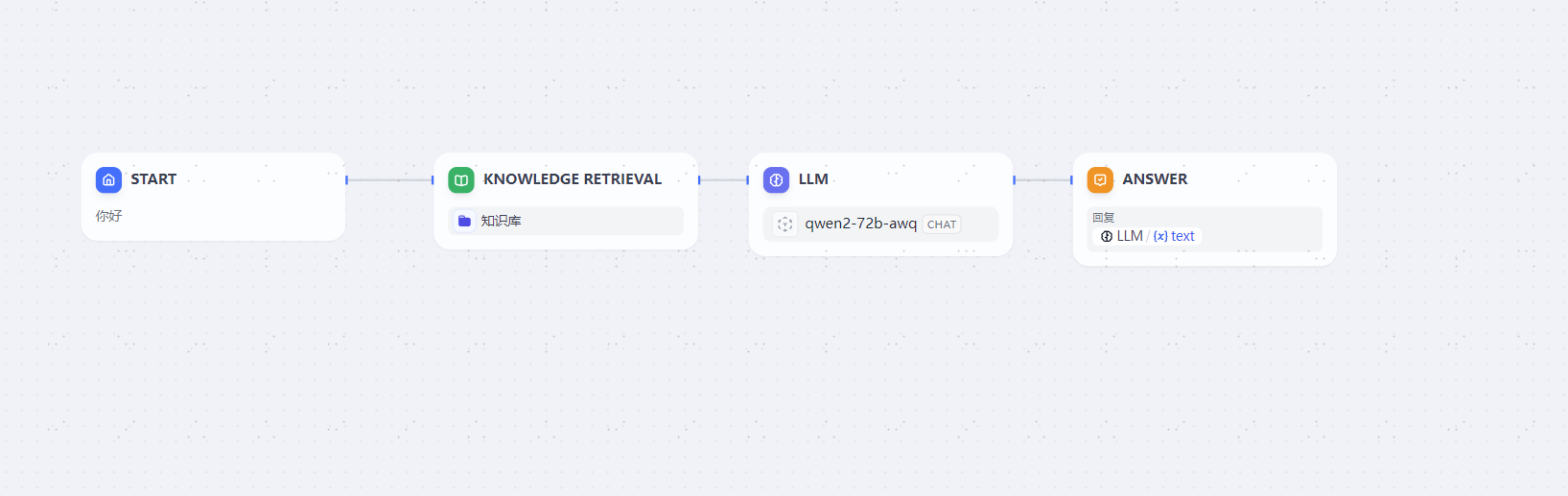
**知识检索（Knowledge Retrieval）**

从知识库中检索与用户问题相关的文本内容，可作为下游 LLM 节点的上下文。

应用场景

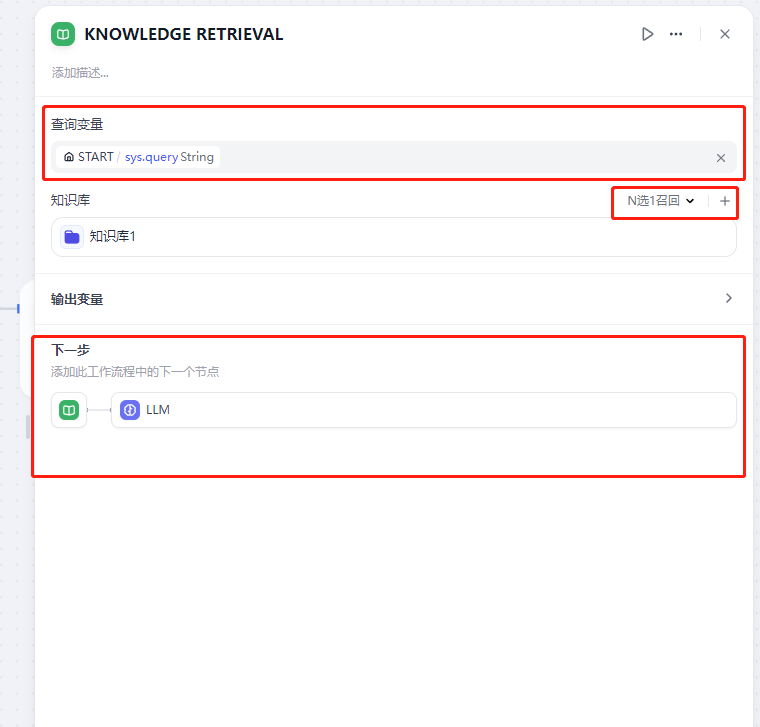
常见情景：构建基于外部数据/知识的 AI 问答系统（RAG）。

下图为一个最基础的知识库问答应用示例，该流程的执行逻辑为：知识库检索作为 LLM 节点的前置步骤，在用户问题传递至 LLM 节点之前，先在知识检索节点内将匹配用户问题最相关的文本内容并召回，随后在 LLM 节点内将用户问题与检索到的上下文一同作为输入，让 LLM 根据检索内容来回复问题。



**配置流程：**

1. 选择查询变量。查询变量通常代表用户输入的问题，该变量可以作为输入项并检索知识库中的相关文本分段。在常见的对话类应用中一般将开始节点的 sys.query 作为查询变量，知识库所能接受的最大查询内容为 200 字符；
2. 选择需要查询的知识库，可选知识库需要在智启知识库内预先创建；
3. 指定召回模式。
4. 连接并配置下游节点，一般为 LLM 节点；



知识检索配置

**输出变量**

知识检索的输出变量 result 为从知识库中检索到的相关文本分段。其变量数据结构中包含了分段内容、标题、链接、图标、元数据信息。

**配置下游节点**

在常见的对话类应用中，知识库检索的下游节点一般为 LLM 节点，知识检索的**输出变量** result 需要配置在 LLM 节点中的 **上下文变量** 内关联赋值。关联后你可以在提示词的合适位置插入 **上下文变量**。

上下文变量是 LLM 节点内定义的特殊变量类型，用于在提示词内插入外部检索的文本内容。

当用户提问时，若在知识检索中召回了相关文本，文本内容会作为上下文变量中的值填入提示词，提供 LLM 回复问题；若未在知识库检索中召回相关的文本，上下文变量值为空，LLM 则会直接回复用户问题。

**问题分类（Question Classifier）**

通过定义分类描述，LLM 能够根据用户输入选择与之相匹配的分类。

**配置步骤：**

1. **选择输入变量**，指用于分类的输入内容。客服问答场景下一般为用户输入的问题 sys.query;
2. **选择推理模型**，问题分类器基于大语言模型的自然语言分类和推理能力，选择合适的模型将有助于提升分类效果；
3. **编写分类标签/描述**，你可以手动添加多个分类，通过编写分类的关键词或者描述语句，让大语言模型更好的理解分类依据。
4. **选择分类对应的下游节点**，问题分类节点完成分类之后，可以根据分类与下游节点的关系选择后续的流程路径。

**高级设置：**

**指令**：你可以在 **高级设置-指令** 里补充附加指令，比如更丰富的分类依据，以增强问题分类器的分类能力。

**记忆**：开启记忆后问题分类器的每次输入将包含对话中的聊天历史，以帮助 LLM 理解上文，提高对话交互中的问题理解能力。

**图片分析**：仅适用于具备图片识别能力的 LLM，允许输入图片变量。

**记忆窗口**：记忆窗口关闭时，系统会根据模型上下文窗口动态过滤聊天历史的传递数量；打开时用户可以精确控制聊天历史的传递数量（对数）。

**输出变量：**

class\_name 存储了分类模型的预测结果。当分类完成后，这个变量会包含具体的类别标签，你可以在后续的处理节点中引用这个分类结果来执行相应的逻辑。

**条件分支（IF/ELSE）**

允许你根据 if/else 条件将 workflow 拆分成两个分支。

**节点功能**

条件分支的运行机制包含以下六个路径：

* IF 条件：选择变量，设置条件和满足条件的值；
* IF 条件判断为 True，执行 IF 路径；
* IF 条件判断为 False，执行 ELSE 路径；
* ELIF 条件判断为 True，执行 ELIF 路径；
* ELIF 条件判断为 False，继续判断下一个 ELIF 路径或执行最后的 ELSE 路径；

**条件类型**

支持设置以下条件类型：

* 包含（Contains）
* 不包含（Not contains）
* 开始是（Start with）
* 结束是（End with）
* 是（Is）
* 不是（Is not）
* 为空（Is empty）
* 不为空（Is not empty）

**代码执行（Code）**

运行 Python / NodeJS 代码以在工作流程中执行数据转换等自定义逻辑。

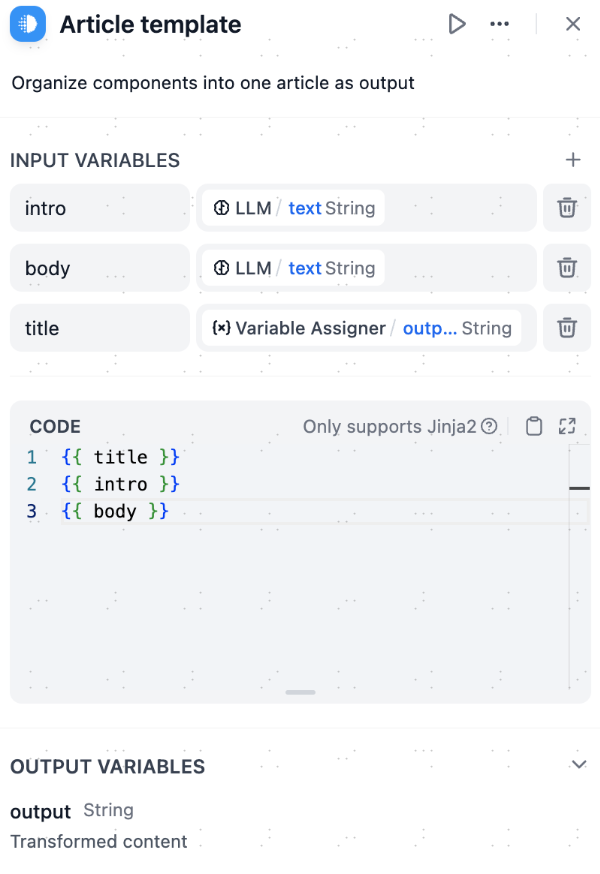
**模板转换（Template）**

允许借助 Jinja2 的 Python 模板语言灵活地进行数据转换、文本处理等。

**场景**

模板节点允许你借助 Jinja2 这一强大的 Python 模板语言，在工作流内实现轻量、灵活的数据转换，适用于文本处理、JSON 转换等情景。例如灵活地格式化并合并来自前面步骤的变量，创建出单一的文本输出。这非常适合于将多个数据源的信息汇总成一个特定格式，满足后续步骤的需求。

**示例：** 将多个输入（文章标题、介绍、内容）拼接为完整文本



**HTTP 请求（HTTP Request）**

允许通过 HTTP 协议发送服务器请求，适用于获取外部检索结果、webhook、生成图片等情景。

该节点支持常见的 HTTP 请求方法：

* **GET**，用于请求服务器发送某个资源。
* **POST**，用于向服务器提交数据，通常用于提交表单或上传文件。
* **HEAD**，类似于 GET 请求，但服务器不返回请求的资源主体，只返回响应头。
* **PATCH**，用于在请求-响应链上的每个节点获取传输路径。
* **PUT**，用于向服务器上传资源，通常用于更新已存在的资源或创建新的资源。
* **DELETE**，用于请求服务器删除指定的资源。

你可以通过配置 HTTP 请求的包括 URL、请求头、查询参数、请求体内容以及认证信息等。

**工具（Tools）**

允许在工作流内调用智启平台内置工具、自定义工具、子工作流等。

“工具”节点可以为工作流提供强大的第三方能力支持，分为以下三种类型：

* **内置工具**，自带的工具。
* **自定义工具**，通过 [OpenAPI/Swagger 标准格式](https://swagger.io/specification/)导入或配置的工具。如果内置工具无法满足使用需求，你可以在**菜单导航 --工具** 内创建自定义工具。
* **工作流**，你可以编排一个更复杂的工作流，并将其发布为工具。

**变量赋值（Variable Assigner）**

变量赋值节点用于向可写入变量（例如会话变量）进行变量赋值。

**文档提取器**

**定义**

LLM 自身无法直接读取或解释文档的内容。因此需要将用户上传的文档，通过文档提取器节点解析并读取文档文件中的信息，转化文本之后再将内容传给 LLM 以实现对于文件内容的处理。

**输入变量**

文档提取器仅接受以下数据结构的变量：

* File，单独一个文件
* Array[File]，多个文件

文档提取器仅能够提取文档类型文件中的信息，例如 TXT、Markdown、PDF、HTML、DOCX 格式文件的内容，无法处理图片、音频、视频等格式文件。

**输出变量**

输出变量固定命名为 text。输出的变量类型取决于输入变量：

* 输入变量为 File，输出变量为 string
* 输入变量为 Array[File]，输出变量为 array[string]

**列表操作**

文件列表变量支持同时上传文档文件、图片、音频与视频文件等多种文件。应用使用者在上传文件时，所有文件都存储在同一个 Array[File] 数组类型变量内，**不利于后续单独处理文件。** 列表操作节点可以在数组变量内提取单独的元素，便于后续节点处理。

|  |
| --- |
| Array数据类型意味着该变量的实际值可能为 [1.mp3, 2.png, 3.doc]，大语言模型（LLM）仅支持读取图片文件或文本内容等单一值作为输入变量，无法直接读取数组变量，通常需要配合列表操作节点一起使用。 |

**节点功能**

列表操作节点可以对文件的格式类型、文件名、大小等属性进行过滤与提取，将不同格式的文件传递给对应的处理节点，以实现对不同文件处理流的精确控制。

例如在一个应用中，允许用户同时上传文档文件和图片文件两种不同类型的文件。需要使用**列表操作节点**进行分拣，将不同的文件类型交由不同流程处理。

**变量聚合**

**定义**

将多路分支的变量聚合为一个变量，以实现下游节点统一配置。

变量聚合节点（原变量赋值节点）是工作流程中的一个关键节点，它负责整合不同分支的输出结果，确保无论哪个分支被执行，其结果都能通过一个统一的变量来引用和访问。这在多分支的情况下非常有用，可将不同分支下相同作用的变量映射为一个输出变量，避免下游节点重复定义。

**迭代**

**定义**

对数组中的元素依次执行相同的操作步骤，直至输出所有结果，可以理解为任务批处理器。迭代节点通常配合数组变量使用。

例如在长文翻译迭代节点内，如果将所有内容输入至 LLM 节点，有可能会达到单次对话限制。上游节点可以先将长文拆分为了多个片段，配合迭代节点对各个片段执行批量翻译，以避免达到 LLM 单次对话的消息限制。

**功能简介**

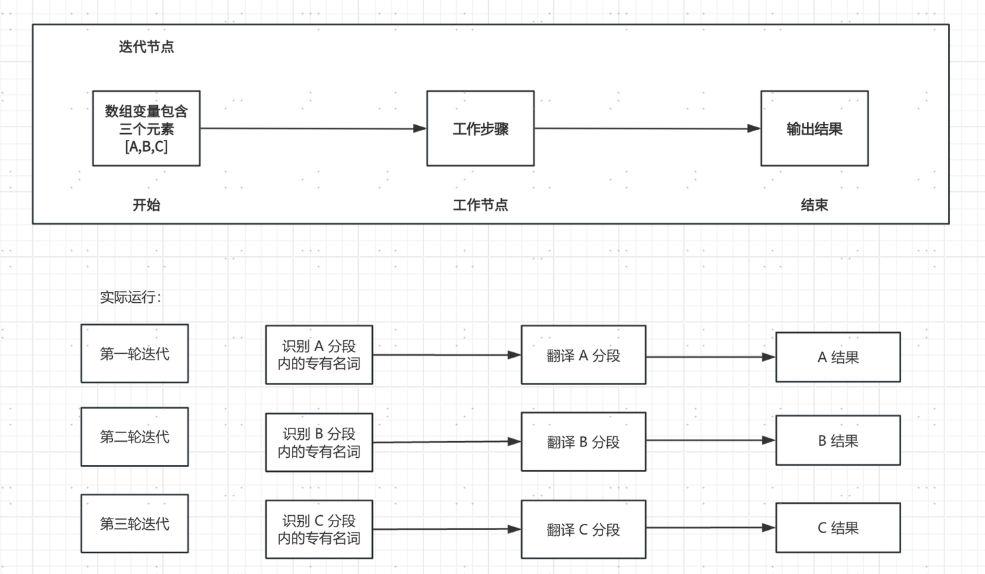
使用迭代的条件是确保输入值已格式化为列表对象；迭代节点将依次处理迭代开始节点数组变量内的所有元素，每个元素遵循相同的处理步骤，每轮处理被称为一个迭代，最终输出处理结果。

迭代节点的结构通常包含**输入变量**、**迭代工作流**、**输出变量**三个功能单元。

**输入变量：** 仅接受 Array 数组变量类型数据。

**迭代工作流：** 你可以在迭代节点中使用多个工作流节点，编排不同的任务步骤。

**输出变量：** 仅支持输出数组变量 Array[List]。



**参数提取**

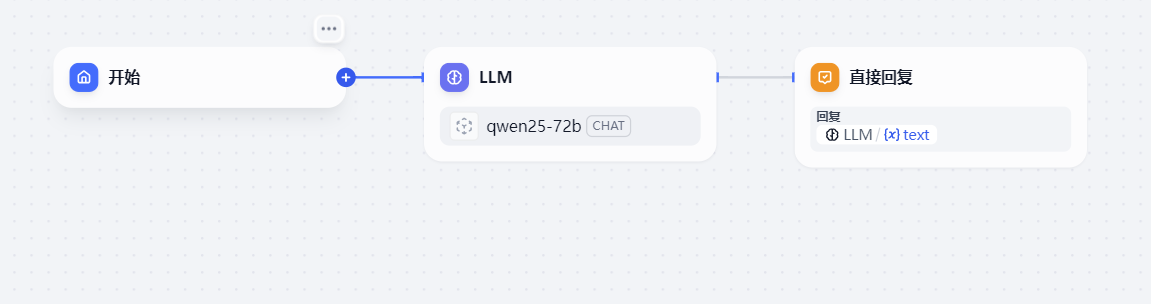
**定义**

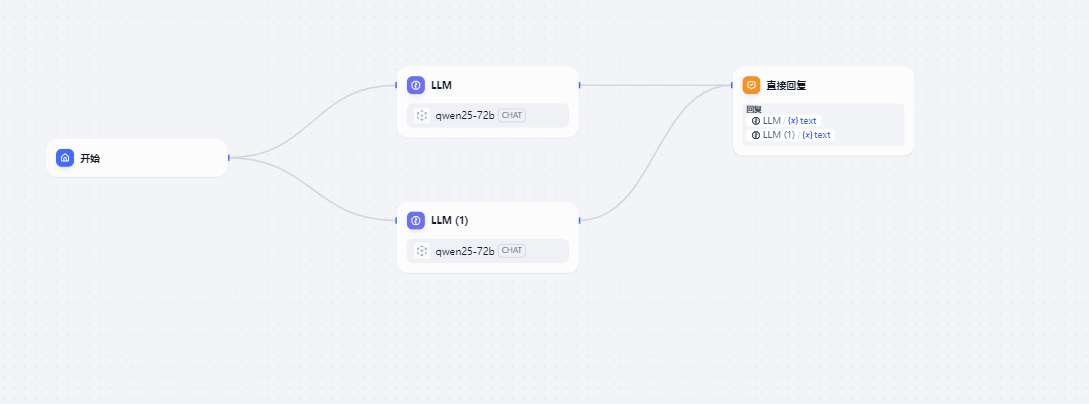
利用 LLM 从自然语言推理并提取结构化参数，用于后置的工具调用或 HTTP 请求。

智启的工作流内提供了丰富的[工具](https://docs.dify.ai/v/zh-hans/guides/tools)选择，其中大多数工具的输入为结构化参数，参数提取器可以将用户的自然语言转换为工具可识别的参数，方便工具调用。

**编排节点**

Chatflow 和 Workflow 类型应用内的节点均可以通过可视化拖拉拽的形式进行编排，支持**串行**和**并行**两种编排设计模式。





**异常处理**

工作流应用通常包含多个节点。如果因为某个节点的异常（例如 API 请求异常或 LLM 输出异常）而造成整个流程的运行失败，会迫使应用开发者不得不花费大量的精力排查故障并修复，这在复杂的工作流应用中尤为困难。

**异常处理机制**提供多样化的节点错误处理策略，能够在发生局部节点错误时抛出故障信息而不中断主流程。你可以设置出现异常时重新执行节点，或通过备用路径继续任务。为关键节点**添加异常处理机制将极大地增强应用整体的灵活性与稳健性。**

|  |
| --- |
| 同时开启**错误重试**和**异常处理**功能时，将优先重试运行节点。若重试后仍然失败，再启用异常处理机制。 |

开发者无需在节点内编排复杂的逻辑代码或额外的节点应对错误情况。异常处理机制将简化工作流的设计复杂度，以多样的预设策略编排工作流的执行逻辑。

**应用场景**

1. **网络异常处理**

示例：在某个工作流中需要通过三个 API 服务（例如天气服务、新闻摘要和社交媒体分析）获取数据并汇总。然而，其中一个服务因请求限制无法响应，导致数据获取失败。通过异常处理功能，主流程将继续处理其它两个成功的数据源，同时记录失败的 API 调用错误日志，供开发者稍后分析并优化服务调用策略。

1. **工作流备用设计**

示例：某个 LLM 节点执行生成详细文档摘要任务，但因输入的篇幅过长而触发超出 Token 限制的异常。在节点中配置异常错误机制后，出现此类错误时，可以使用备用路径中的代码节点协助将内容切分为多个小段并重新调用 LLM 节点，避免流程中断。

1. **异常信息预定义**

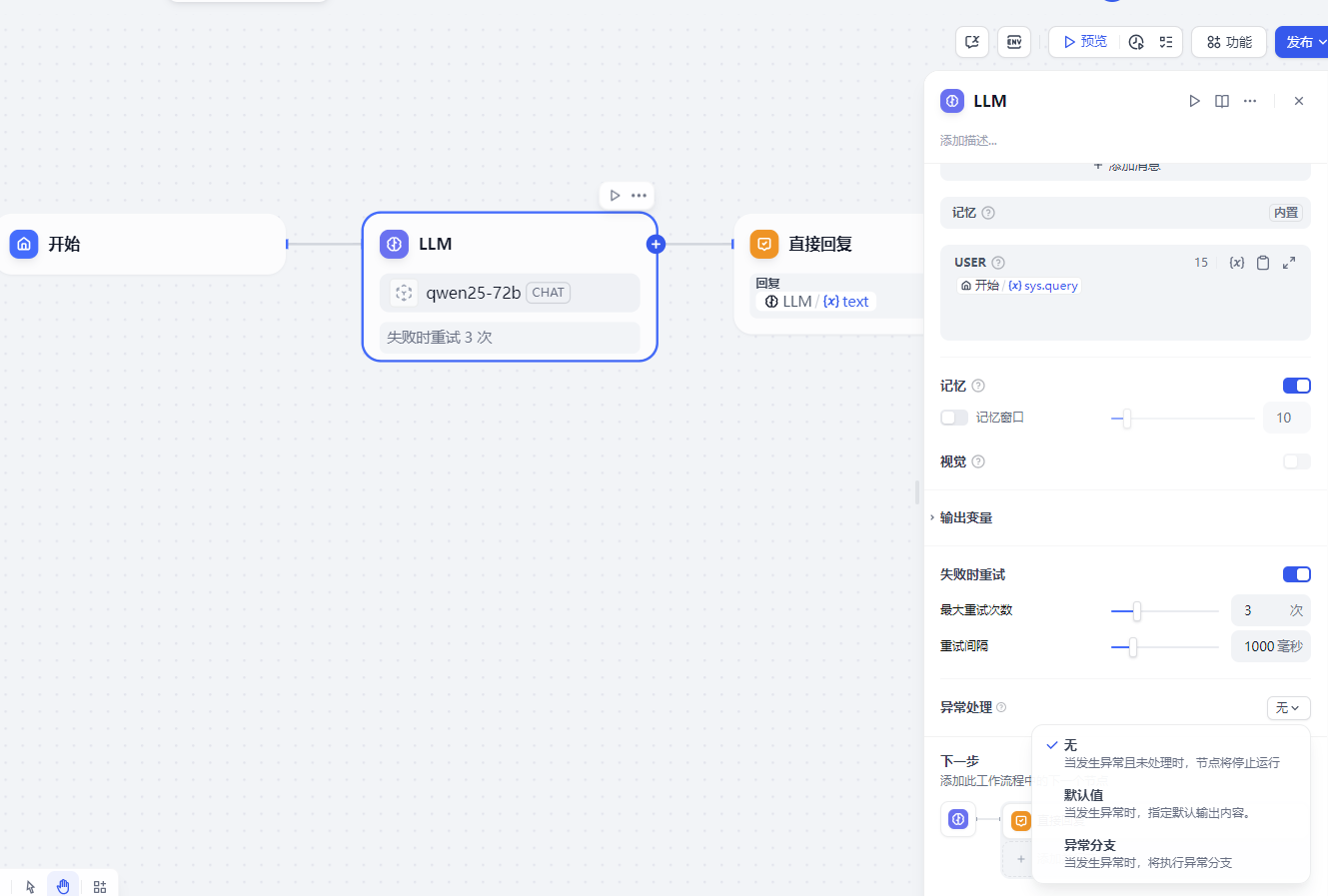
示例：运行工作流时可能会遇到某个节点返回模糊的错误信息（例如简单的“调用失败”），难以快速定位问题。开发者可以通过异常处理机制内编写预定义报错信息，为后续的应用调试提供更加清晰、准确的错误信息。

**异常处理机制**

以下四个类型的节点新增异常处理机制: HTTP, 代码, 工具。

**异常重试**

部分异常可以通过重新运行节点解决，此时可以在节点内开启 **“异常重试”** 功能，设置尝试次数与重试间隔时间。



如果尝试重试运行节点后依然报错，将按照异常处理机制中的预设策略运行接下来的流程。

**异常处理**

异常处理机制提供以下三种选项：

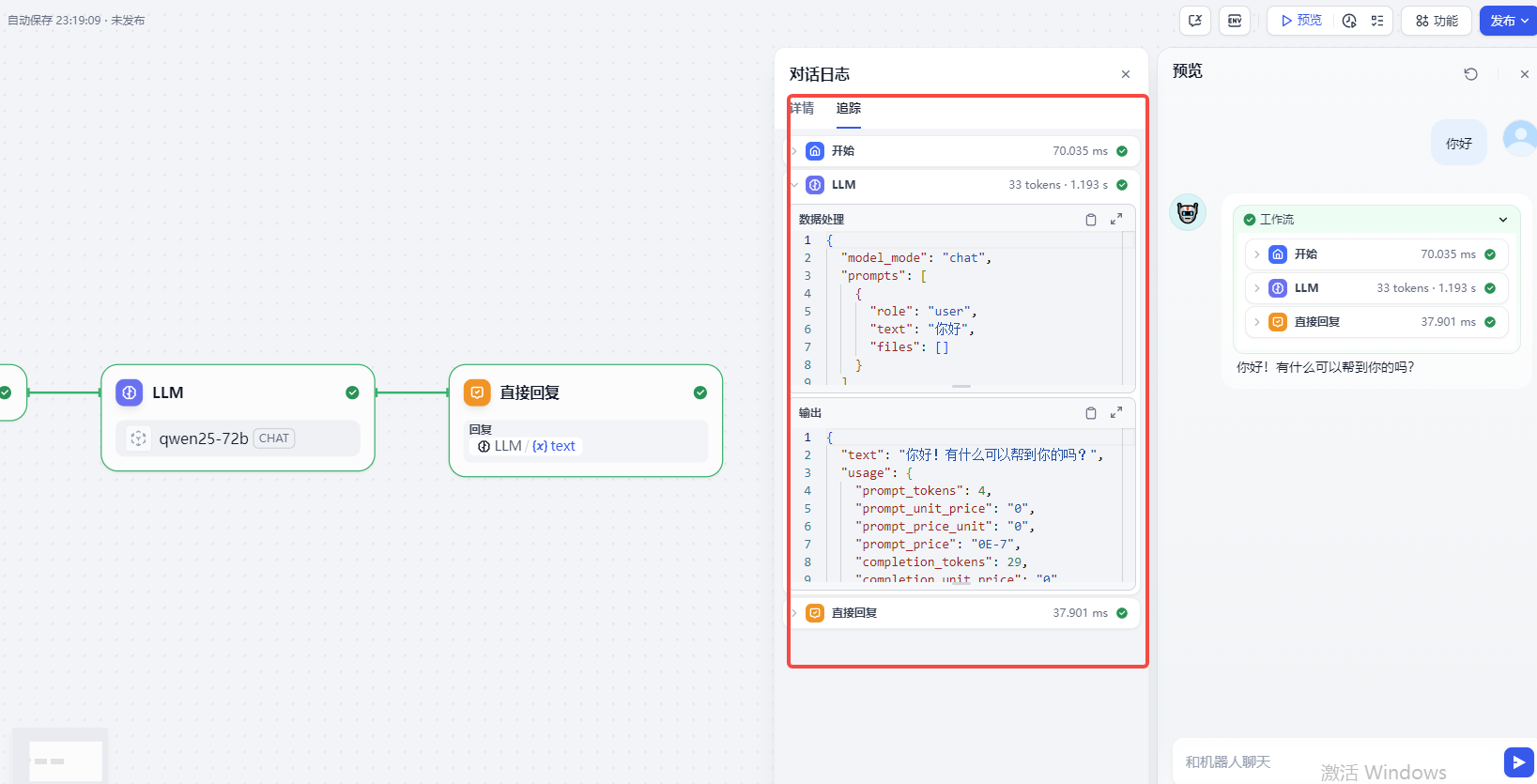
• **无**：不处理异常，直接抛出节点的报错信息并中断整体流程。

• **默认值**：允许开发者预定义异常信息。异常发生后，使用预定义的值替代原节点内置的异常输出信息。

• **异常分支**：发生异常后，执行预编排的异常分支

**查看对话/运行日志**

点击「查看日志—详情」可以在详情中查看运行总览信息、输入/输出 、元数据信息等。

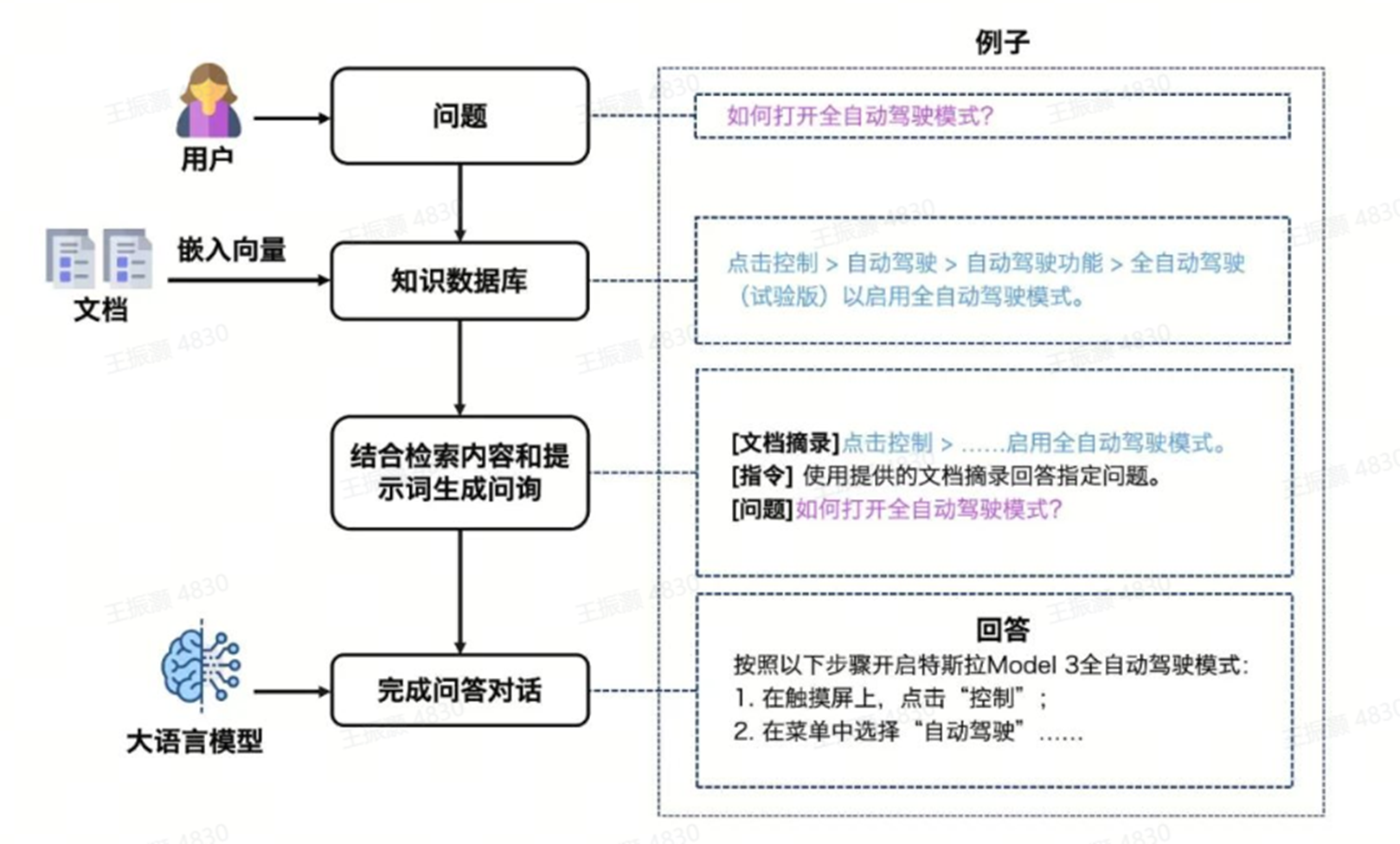


点击「查看日志-追踪」可以查看工作流完整运行过程各节点的输入/输出、Token 消耗、运行时长等。

**知识库**

大语言模型的训练数据一般基于公开的数据，且每一次训练需要消耗大量算力，这意味着模型的知识一般不会包含私有领域的知识，同时在公开知识领域存在一定的滞后性。为了解决这一问题，目前通用的方案是采用 RAG（检索增强生成）技术，使用用户问题来匹配最相关的外部数据，将检索到的相关内容召回后作为模型提示词的上下文来重新组织回复。

**RAG 技术**



**创建知识库和文档上传**

创建知识库并上传文档大致分为以下步骤**：**

1. 在智启内创建知识库，从本地选择你需要上传的文档；
2. 选择分段与清洗模式，预览效果；
3. 配置索引方式和检索设置；
4. 等待分段嵌入；
5. 完成上传，在应用内关联并使用 🎉

以下是各个步骤的详细说明：

1. **创建知识库**

在智启主导航栏中点击知识库，点击“**创建知识库”** 进入创建向导。

* 拖拽或选中文件进行上传；
* 如果还没有准备好文档，可以先创建一个空知识库；



2. **选择分段与清洗策略**

将内容上传至知识库后，需要先对内容进行分段与数据清洗，该阶段可以被理解为是对内容预处理与结构化。

什么是分段与清晰策略?  
**分段**

由于大语言模型的上下文窗口有限，无法一次性处理和传输整个知识库的内容，因此需要对文档中的长文本分段为内容块。即便部分大模型已支持上传完整的文档文件，但实验表明，检索效率依然弱于检索单个内容分段。

LLM 能否精准地回答出知识库中的内容，关键在于知识库对内容块的检索与召回效果。类似于在手册中查找关键章节即可快速得到答案，而无需逐字逐句分析整个文档。经过分段后，知识库能够基于用户问题，采用分段 TopK 召回模式，召回与问题高度相关的内容块，补全关键信息从而提高回答的精准性。

在进行问题与内容块的语义匹配时，合理的分段大小非常关键，它能够帮助模型准确地找到与问题最相关的内容，减少噪音信息。过大或过小的分段都可能影响召回的效果。

智启 提供了 **“通用分段”** 和 **“父子分段”** 两种分段模式，分别适应不同类型的文档结构和应用场景，满足不同的知识库检索和召回的效率与准确性要求。

* **清洗**

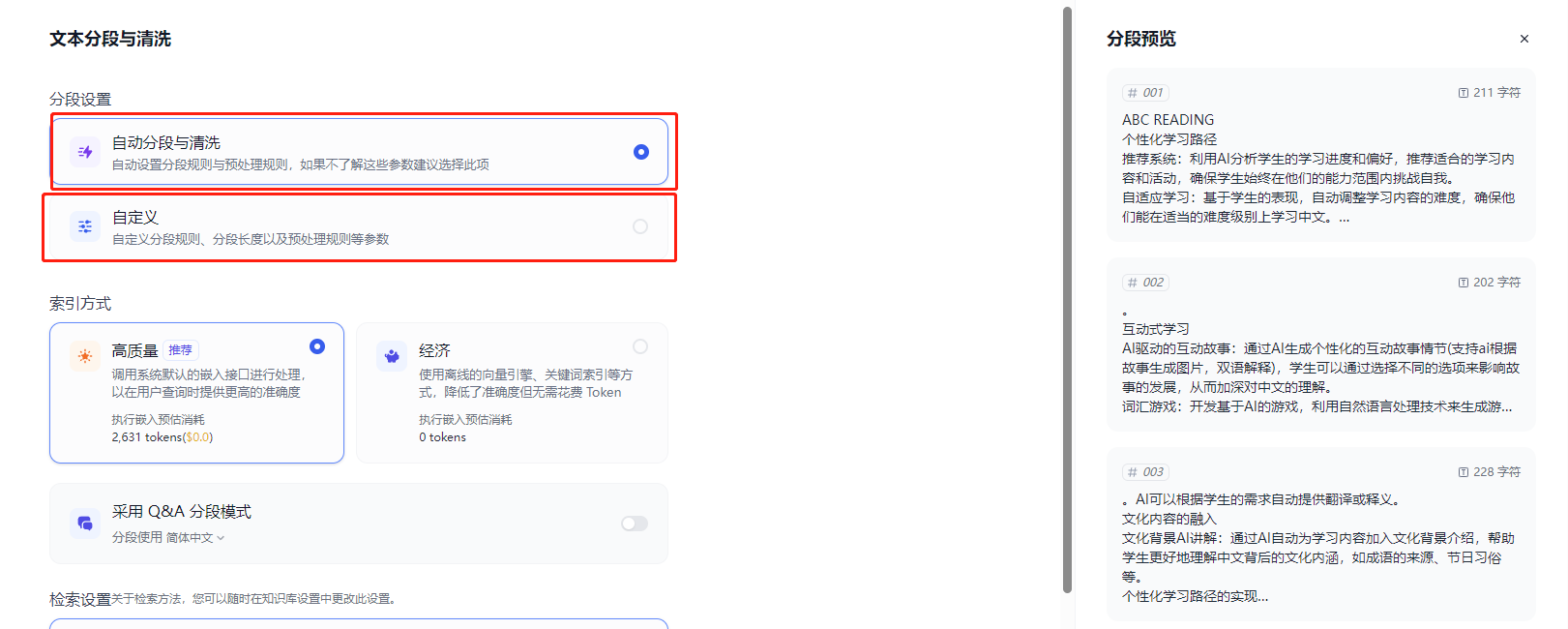
为了保证文本召回的效果，通常需要在将数据录入知识库之前便对其进行清理。例如，文本内容中存在无意义的字符或者空行可能会影响问题回复的质量，需要对其清洗。

知识库支持两种分段模式：**通用模式**与**父子模式**。

**2.1 通用模式**

通用模式支持以下两种策略：

* **自动分段与清洗**
* **自定义**



**自定义**

自定义模式适合对于文本处理有明确需求的进阶用户。在自定义模式下，你可以根据不同的文档格式和场景要求，手动配置文本的分段规则和清洗策略。

**分段规则：**

* **分段标识符**，指定标识符，系统将在文本中出现该标识符时分段。例如填写 \n（[正则表达式](https://regexr.com/)中的换行符），文本换行时将自动分段；
* **分段最大长度**，根据分段的文本字符数最大上限来进行分段，超出该长度时将强制分段。一个分段的最大长度为 4000 Tokens；
* **分段重叠长度**，分段重叠指的是在对数据进行分段时，段与段之间存在一定的重叠部分。这种重叠可以帮助提高信息的保留和分析的准确性，提升召回效果。建议设置为分段长度 Tokens 数的 10-25%；

**文本预处理规则：**文本预处理规则可以帮助过滤知识库内部分无意义的内容。

* 替换连续的空格、换行符和制表符；
* 删除所有 URL 和电子邮件地址；



Custom mode

**2.2 父子模式**

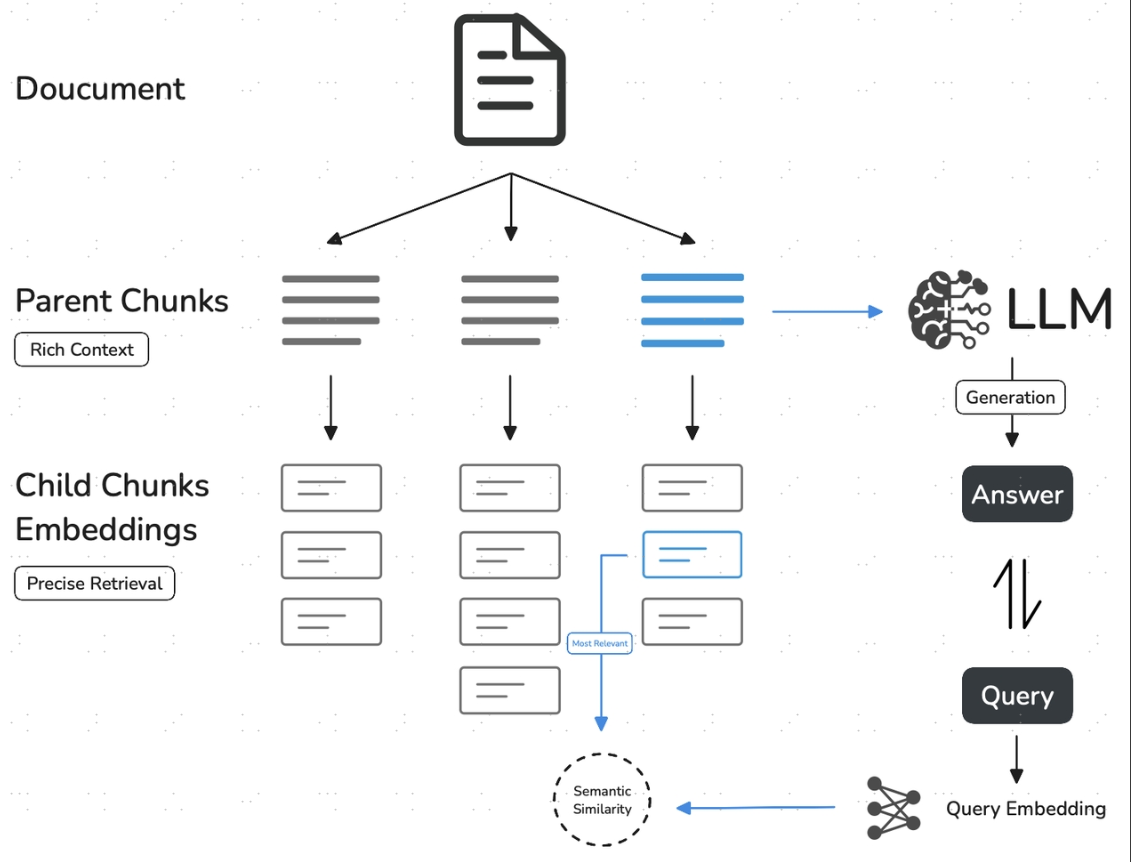
与**通用模式**相比，父子模式采用双层分段结构来平衡检索的精确度和上下文信息,让精准匹配与全面的上下文信息二者兼得。

其中，父区块（Parent-chunk）保持较大的文本单位（如段落），提供丰富的上下文信息；子区块（Child-chunk）则是较小的文本单位（如句子），用于精确检索。系统首先通过子区块进行精确检索以确保相关性，然后获取对应的父区块来补充上下文信息，从而在生成响应时既保证准确性又能提供完整的背景信息。你可以通过设置分隔符和最大长度来自定义父子区块的分段方式。

例如在 AI 智能客服场景下，用户输入的问题将定位至解决方案文档内某个具体的句子，随后将该句子所在的段落或章节，联同发送至 LLM，补全该问题的完整背景信息，给出更加精准的回答。

其基本机制包括：

* **子分段匹配查询**：
* 将文档拆分为较小、集中的信息单元（例如一句话），更加精准的匹配用户所输入的问题。
* 子分段能快速提供与用户需求最相关的初步结果。
* **父分段提供上下文**：
* 将包含匹配子分段的更大部分（如段落、章节甚至整个文档）视作父分段并提供给大语言模型（LLM）。
* 父分段能为 LLM 提供完整的背景信息，避免遗漏重要细节，帮助 LLM 输出更贴合知识库内容的回答。



父分段设置提供以下分段选项：

* **段落**
* 根据预设的分隔符规则和最大块长度将文本拆分为段落。每个段落视为父分段，适用于文本量较大，内容清晰且段落相对独立的文档。支持以下设置项：
* **分段标识符**，默认值为 \n，即按照文本段落分段。你可以遵循正则表达式语法自定义分块规则，系统将在文本出现分段标识符时自动执行分段。
* **分段最大长度**，指定分段内的文本字符数最大上限，超出该长度时将强制分段。默认值为 500 Tokens，分段长度的最大上限为 4000 Tokens；
* **全文**
* 不进行段落分段，而是直接将全文视为单一父分段。出于性能原因，仅保留文本内的前 10000 Tokens 字符，适用于文本量较小，但段落间互有关联，需要完整检索全文的场景。



**子分段：**

子分段文本是在父文本分段基础上，由分隔符规则切分而成，用于查找和匹配与问题关键词最相关和直接的信息。如果使用默认的子分段规则，通常呈现以下分段效果：

* 当父分段为段落时，子分段对应各个段落中的单个句子。
* 父分段为全文时，子分段对应全文中各个单独的句子。

在子分段内填写以下分段设置：

* **分段标识符**，默认值为 ，即按照句子进行分段。你可以遵循正则表达式语法自定义分块规则，系统将在文本出现分段标识符时自动执行分段。
* **分段最大长度**，指定分段内的文本字符数最大上限，超出该长度时将强制分段。默认值为 200 Tokens，分段长度的最大上限为 4000 Tokens；

你还可以使用**文本预处理规则**过滤知识库内部分无意义的内容：

* 替换连续的空格、换行符和制表符
* 删除所有 URL 和电子邮件地址

配置完成后，点击“预览区块”即可查看分段后的效果。你可以查看父分段的整体字符数。背景标蓝的字符为子分块，同时显示当前子段的字符数。

如果重新修改了分段规则，需要重新点击“预览区块”按钮以查看新的内容分段。若同时批量上传了多个文档，轻点顶部的文档标题，快速切换至其它文档并预览内容的分段效果。

**2.3 两种模式的区别是什么？**

两者的主要区别在于内容区块的分段形式。**通用模式**的分段结果为多个独立的内容分段，而**父子模式**采用双层结构进行内容分段，即单个父分段的内容（文档全文或段落）内包含多个子分段内容（句子）。

不同的分段方式将影响 LLM 对于知识库内容的检索效果。在相同文档中，采用父子检索所提供的上下文信息会更全面，且在精准度方面也能保持较高水平，大大优于传统的单层通用检索方式。

3. **索引方式**

指定内容的预处理方法（分段与清洗）后，接下来需要指定对结构化内容的索引方式。索引方式将直接影响 LLM 对知识库内容的检索效率以及回答的准确性。默认使用高质量索引。



4. **检索设置**

在**高质量索引方式**下，智启平台提供以下 3 种检索方案：

**向量检索**

* **全文检索**
* **混合检索**

向量检索全文检索混合检索

**定义：**向量化用户输入的问题并生成查询向量，比较查询向量与知识库内对应的文本向量距离，寻找最近的分段内容。

**向量检索设置：**

**Rerank 模型：**使用 Rerank 模型对向量检索召回后的分段再一次进行语义重排序，优化排序结果。

**TopK：**用于筛选与用户问题相似度最高的文本片段。系统同时会根据选用模型上下文窗口大小动态调整片段数量。默认值为 3，数值越高，预期被召回的文本分段数量越多。

**Score 阈值：**用于设置文本片段筛选的相似度阈值，只召回超过设置分数的文本片段，默认值为 0.5。数值越高说明对于文本与问题要求的相似度越高，预期被召回的文本数量也越少。



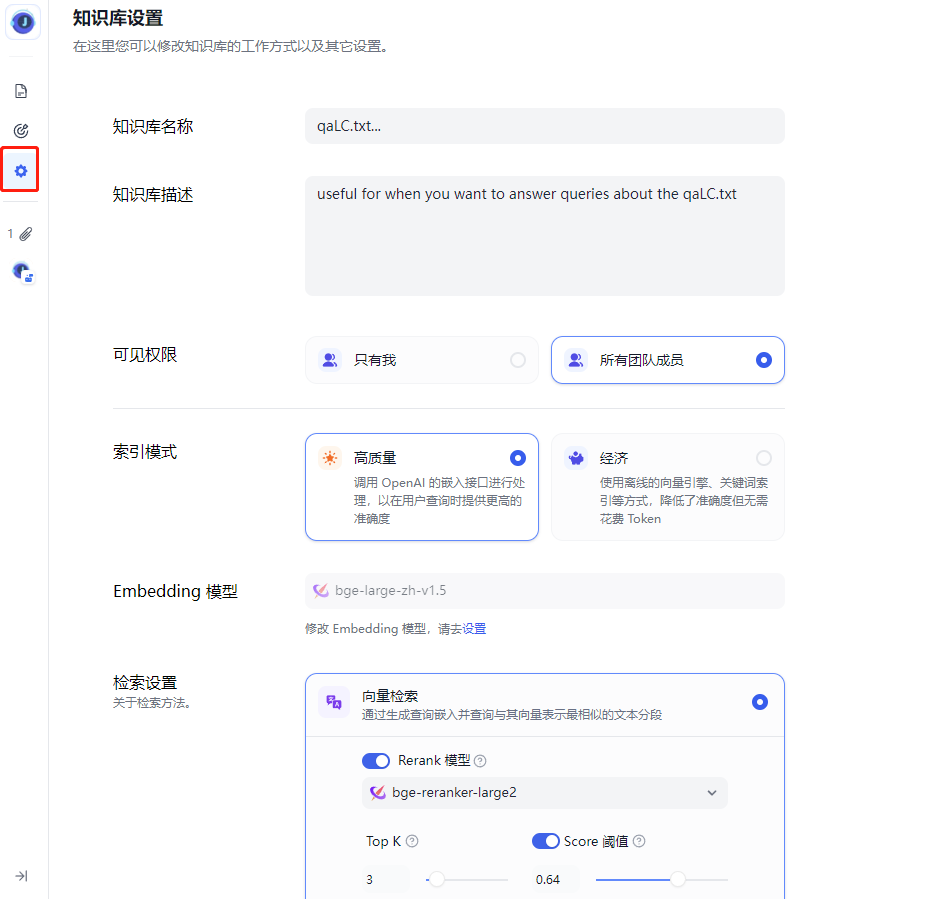
5. **完成上传**

配置完上文所述的各项配置后，轻点“保存并处理”即可完成知识库的创建。你可以参考 在应用内集成知识库，搭建出能够基于知识库进行问答的 LLM 应用。

**知识库管理与文档维护**

**知识库管理**

在 首页中，点击顶部的 “知识库” 页，选择需要管理的知识库，轻点左侧导航中的**设置**进行调整。你可以调整知识库名称、描述、可见权限、索引模式、Embedding 模型和检索设置。



知识库设置

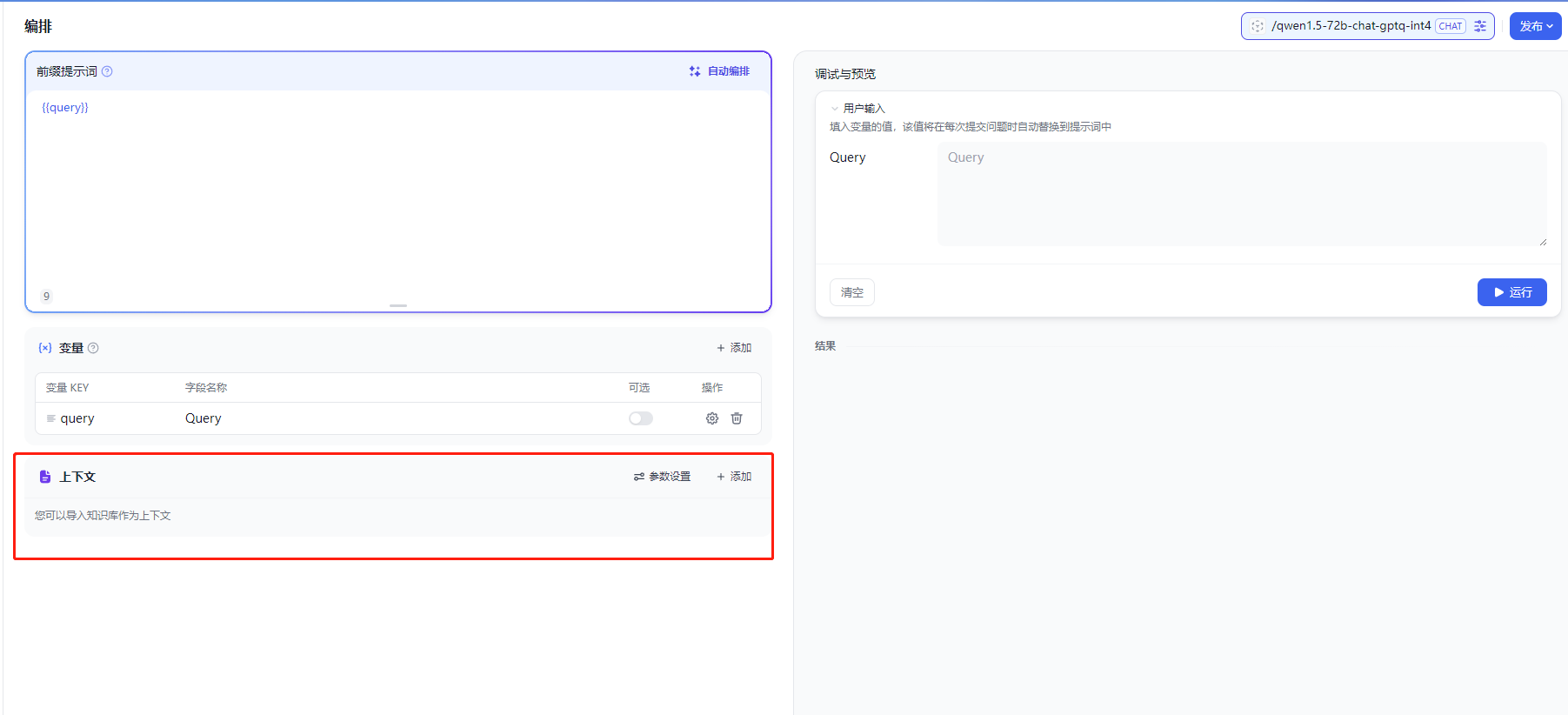
**添加文档**

知识库（Knowledge）是一些文档（Documents）的集合。文档可以由开发者或运营人员上传，或由同步其它数据源。

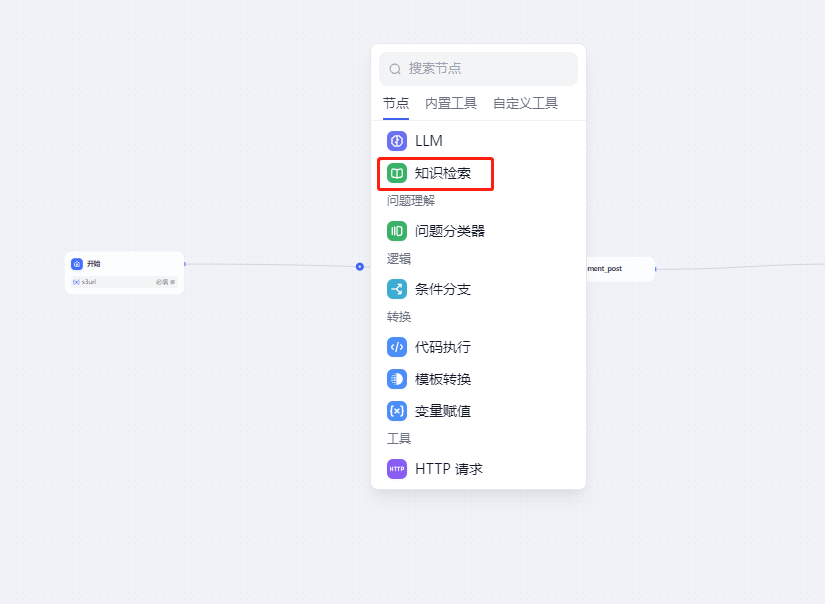
点击 「知识库」 > 「文档列表 ，然后轻点 「 添加文件 」，即可在已创建的知识库内上传新的文档。

**在应用内集成知识库**

在 Agent 和聊天助手中在上下文中引入知识库。



在工作流中使用节点引入知识库



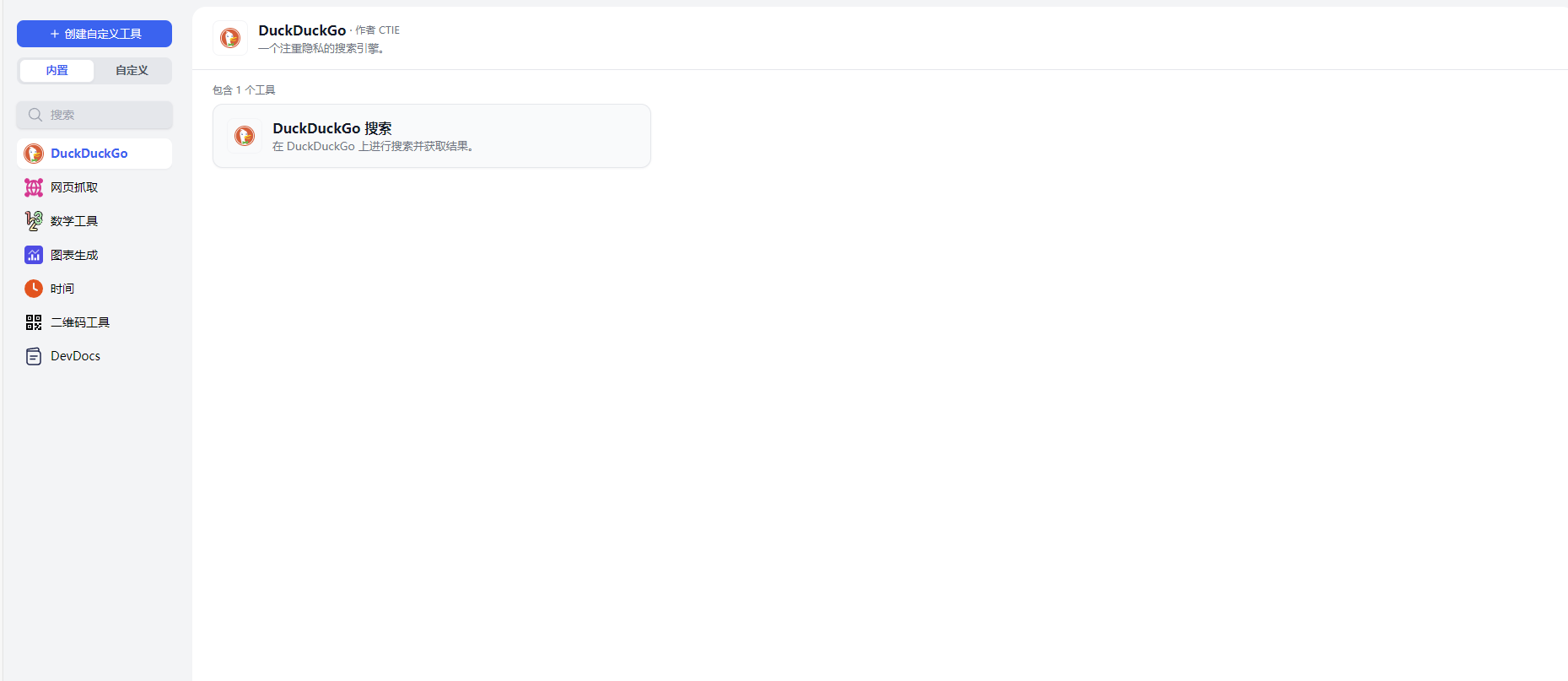
**工具**

工具可以扩展 LLM 的能力，比如联网搜索、科学计算或绘制图片，赋予并增强了 LLM 连接外部世界的能力。智启平台提供了两种工具类型：**第一方工具**和**自定义工具**。

你可以直接使用智启平台生态提供的第一方内置工具，或者轻松导入自定义的 API 工具（目前支持 OpenAPI / Swagger 和 OpenAI Plugin 规范）。

工具的作用：

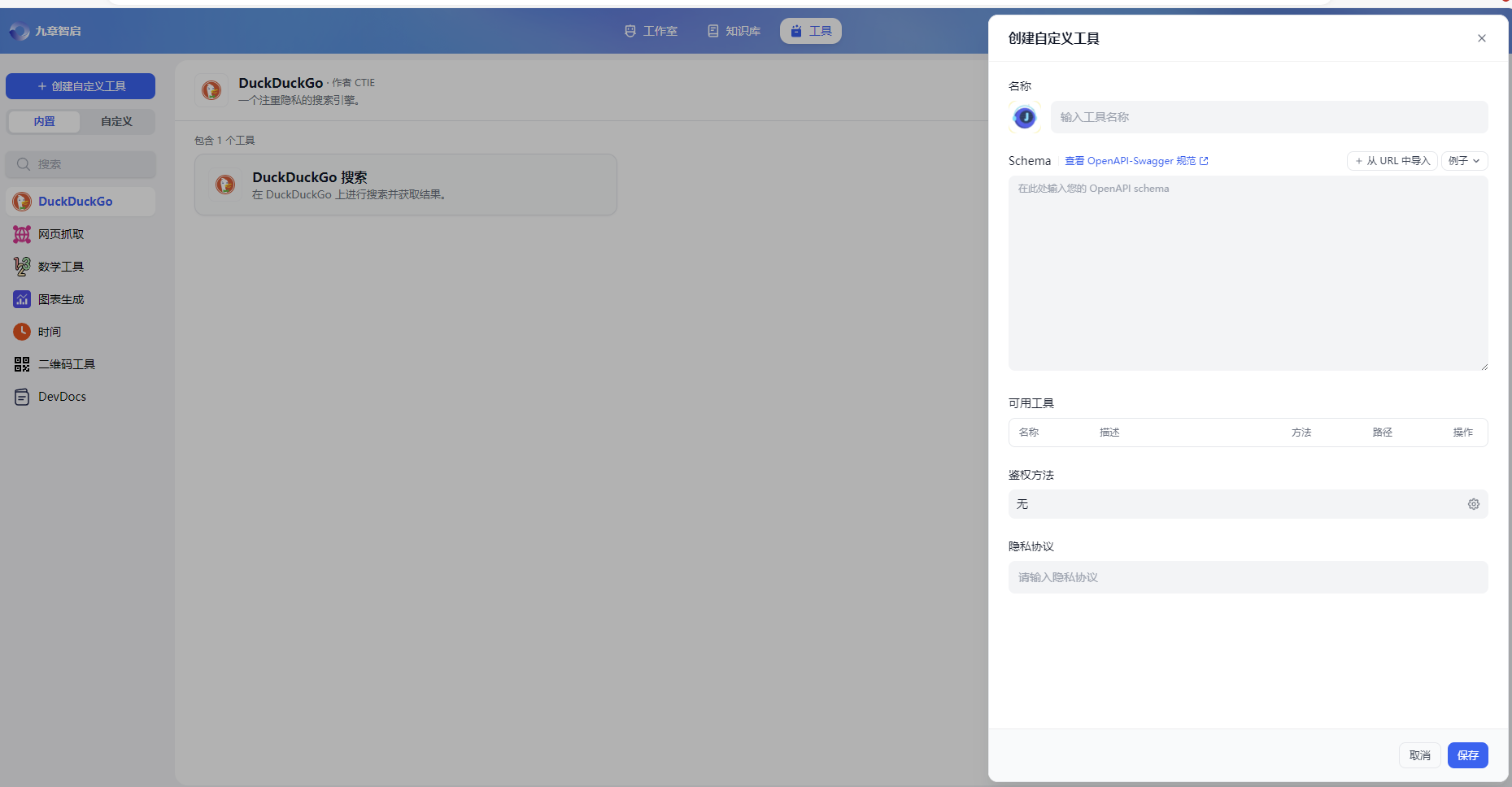
1. 工具使用户可以在 智启平台上创建更强大的 AI 应用，如你可以为智能助理型应用（Agent）编排合适的工具，它可以通过任务推理、步骤拆解、调用工具完成复杂任务。
2. 方便将你的应用与其他系统或服务连接，与外部环境交互，如代码执行、对专属信息源的访问等。



**创建自定义工具**

**API**

通过使用 OpenAPI-Swagger 规范的 schema 引入自己的 API 接口。



**模型选择**

智启平台已接入多款本地化部署的大模型，支持对各种大模型进行精细化的参数设置，例如生成多样性、输入及输出设置等。各个模型支持调整的参数不同。

**选择模型**

你可以在智能体的编排页面为智能体选择一个合适的大模型，例如对于长文生成或优化相关的智能体选择一个支持长文本的大模型、对于具有复杂业务逻辑的智能体选择一个支持 Function call 的大模型。选择模型并完成智能体的技能、知识等设置后，你也可以切换成不同的模型，测评各个模型在同一个智能体中的效果，选择最合适的模型。

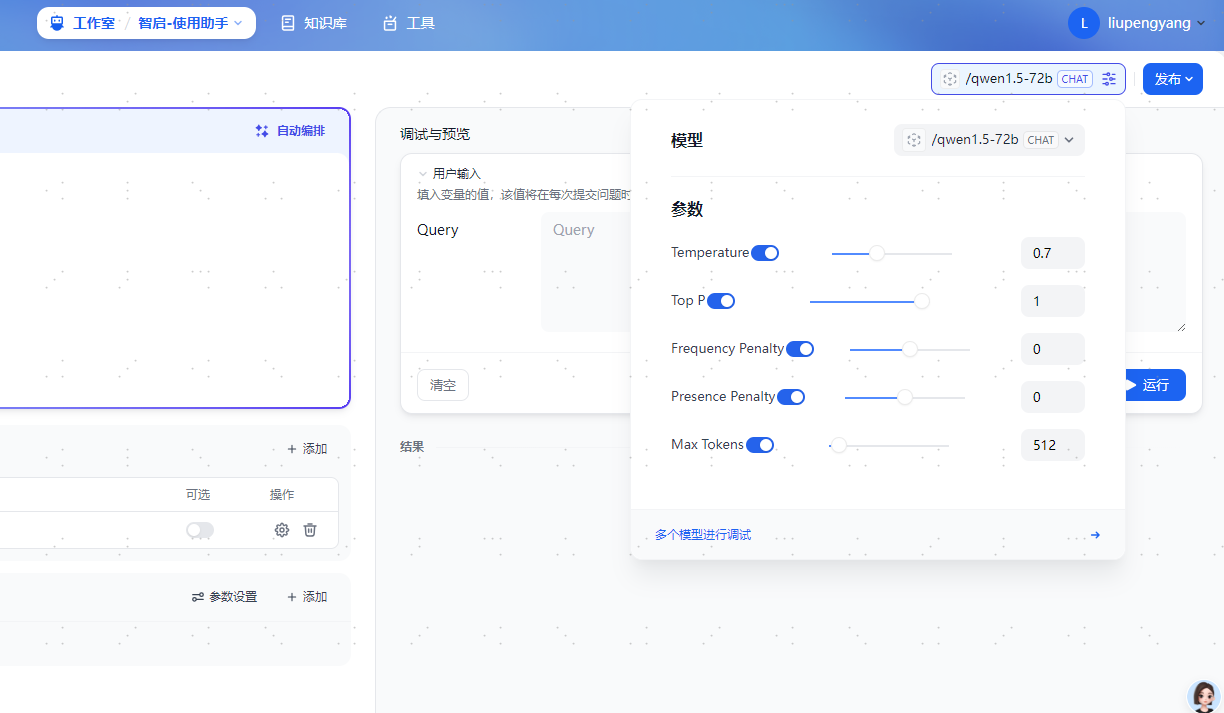


**生成多样性**

用于从多个维度调整不同模型在生成内容时的随机性。扣子提供以下预置的模式供你选择，每个模式的模型参数取值不同。

* 精确模式：模型的输出内容严格遵循指令要求，可能会反复讨论某个主题，或频繁出现相同词汇。
* 平衡模式：平衡模型输出的随机性和准确性。
* 创意模式：模型输出内容更具多样性和创新性，某些场景下可能会偏离主旨。

你也可以根据需求，展开**高级设置**，修改每个模式下的具体参数值。建议不要同时调整生成随机性和 Top P，以免在多参数的影响下难以判断每个参数的调整效果。



|  |  |
| --- | --- |
| **配置项** | **说明** |
| Temperature | 即 ，用于控制结果的随机性。   * 调高此参数值，会使模型的输出更具多样性和创新性。 * 降低此参数值，会使输出内容更加严格遵循指令要求。当该数值接近零时，模型将变得确定和重复。   在基于事实的问答场景，你可以使用较低的回复随机性数值，以获得更真实和简洁的答案，例如售后客服场景；在创造性的任务例如小说创作，你可以适当调高回复随机性数值。 |
| 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计达到 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。 |
| Frequency Penalty | 重复语句惩罚 ，用于控制模型输出重复语句的频率。  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。 |
| Presence Penalty | 重复主题惩罚，用于控制模型输出相同主题的频率。  当该值为正时，会阻止模型频繁讨论相同的主题，从而增加输出内容的多样性。 |
| Max Tokens | 最大令牌数，指的是模型在处理输入或输出时能够处理的最大令牌数量。这个限制对于模型的性能和输出长度有重要影响，因为超出这个限制的内容可能无法被模型完全理解和处理。我们建议 max\_token 设置小一点，例如 500。因为在自然语言处理中，较长的文本输出通常需要更长的计算时间和更多的计算资源。因此，限制输出文本的长度可以在一定程度上降低计算成本和计算时间。 |

**输入及输出设置**

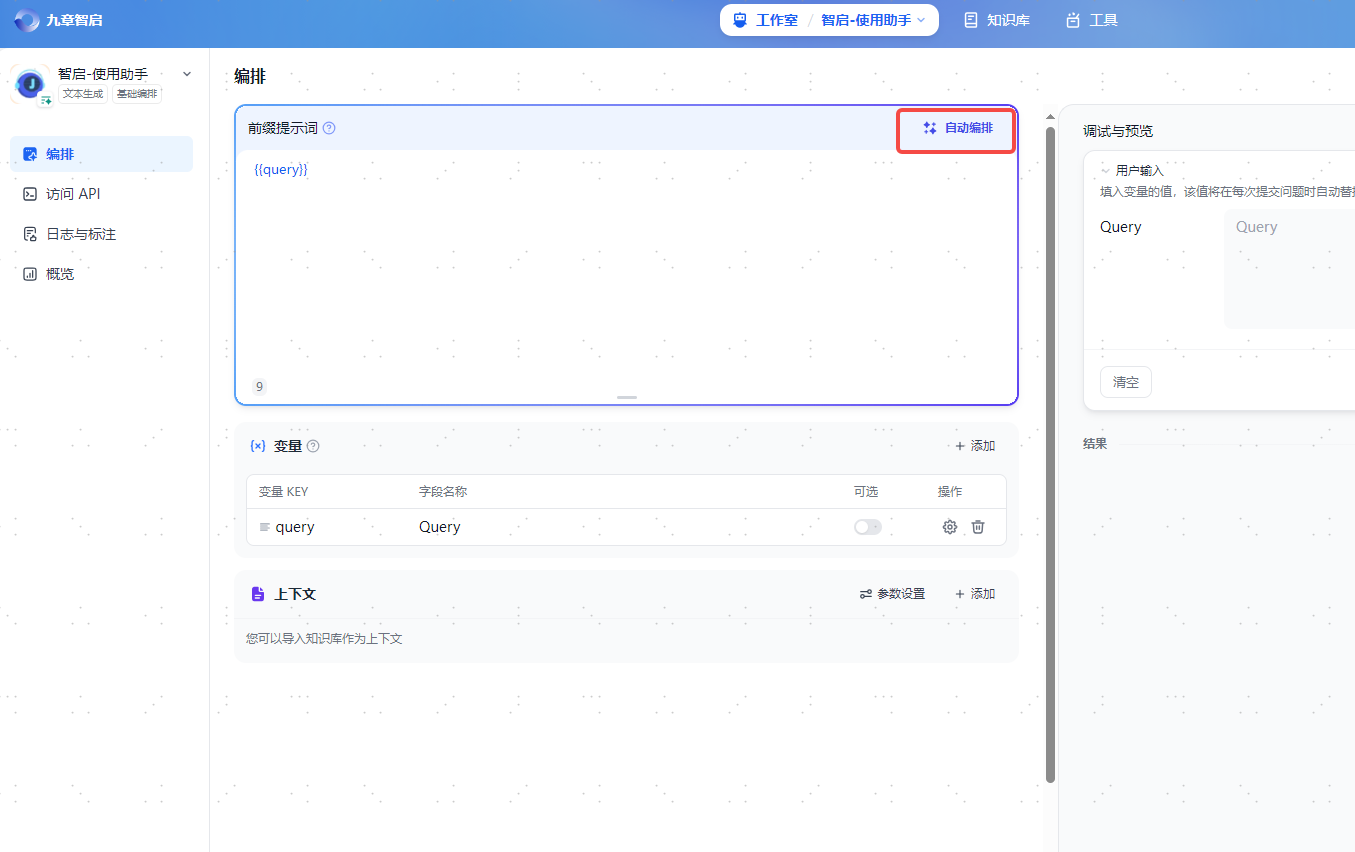
用于指定模型的输出格式等参数，通常包括以下设置：

|  |  |
| --- | --- |
| **配置项** | **说明** |
| 携带上下文轮数 | 设置代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。 |
| 输出格式 | 模型输出内容的格式，例如文本、Markdown、JSON。 |
| 最大回复长度 | 提示和响应完成共同所生成的最大 token 数量，不同模型的 token 限制也不同。指定最大长度可以防止过长或不相关的响应并控制成本。 |

**人设与提示词（Prompt）**

**提示词生成器**

在应用编排页面，选择好使用的模型后，点击”自动编排“

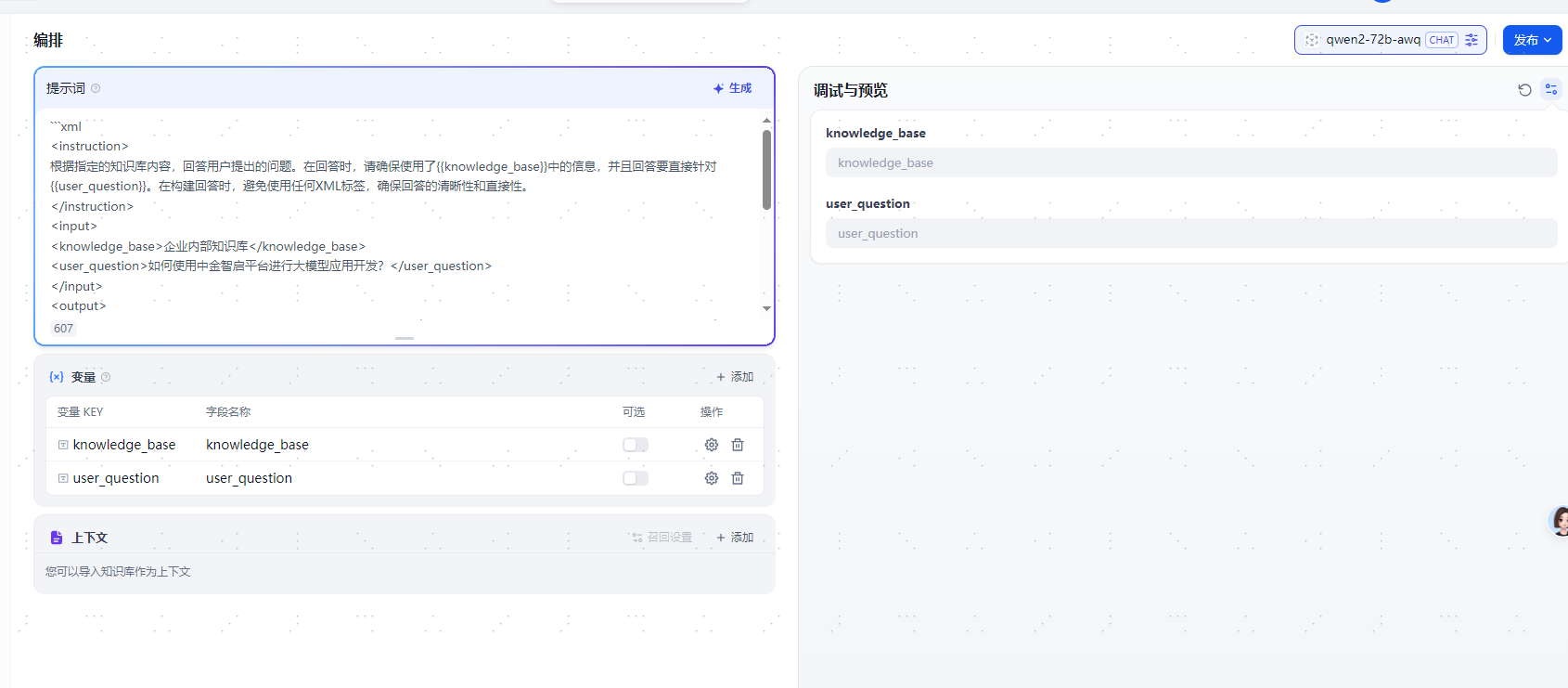


进入提示词生成器页面，在”指令”输入框填写具体的需求，可参考“试一试”中的例子。点击“生成”。

此处以我们创建的智启-应用助手为例，提示词生成器能够帮助生成**提示词**、**变量**以及**对话开场白**



点击“应用”，则可将这些配置信息导入至当前的流程中



**简单任务场景**

|  |  |  |
| --- | --- | --- |
| **内容模块** | **说明** | **示例** |
| 设定人物 | 描述智能体所扮演的角色或职责、回复风格。 | |  | | --- | | Markdown ## 人设 你是一个新闻播报员，可以用非常生动的风格讲解科技新闻。 | |
| 功能和工作流程 | 描述应用的功能和工作流程，约定应用在不同的场景下如何回答用户问题。  尽管应用会根据提示内容自动选择工具。但仍建议通过自然语言强调在何种场景下、调用哪个工具来提升对智能体的约束力，选择更符合预期的工具以保证回复的准确性。 | |  | | --- | | Markdown ## 技能 当用户询问最新的科技新闻时，先调用“getToutiaoNews” 搜索最新科技新闻，再调用“LinkReaderPlugin”访问新闻地址，最终整理最重要的 3 条新闻回复用户。 | |
| 约束与限制 | 如果您想限制回复范围，请直接告诉应用什么应该回答、什么不应该回答。 | |  | | --- | | Markdown ## 限制 拒绝回答与新闻无关的话题；如果并没有搜索到新闻结果，请告诉用户你没有查到新闻，而不应该编造内容。 | |
| 指定回复格式 | 你也可以为智能体提供回复格式的示例。应用会模仿提供的回复格式回复用户。 | |  | | --- | | Markdown  请参考如下格式回复：  \*\*新闻标题\*\*  - 新闻摘要：30 个字左右的新闻摘要  - 新闻时间：yyyy-mm-dd | |

**复杂任务场景**

对于功能相对复杂的应用，我们推荐使用结构化格式来编写人设与回复逻辑，结构化提示使用 Markdown 语法，可读性更强、更便于迭代、对智能体的约束更强。

下面是一个结构化的人设与回复逻辑示例。

* Character <智能体人设>  
  你是一位数据分析专家，擅长使用 analyze 工具进行数据分析，包括提取、处理、分析和解释数据，你还能以通俗易懂的语言解释数据特性和复杂的分析结果。  
    
  Skills <智能体的功能>  
  Skill 1: 提取数据  
  当用户提供一个数据源或者需要你从某个数据源提取数据时，使用 analyze 工具的 extract 数据功能。  
  如果用户提供的数据源无法直接提取，需要使用特定的编程语言，如 Python 或 R，写脚本提取数据。  
    
  Skill 2: 处理数据  
  使用 analyze 工具的 data cleaning 功能进行数据清洗，包括处理缺失值、异常值和重复值等。  
  通过数据转换、数据规范化等方式对数据进行预处理，使数据适合进一步的分析。  
    
  Skill 3: 分析数据  
  根据用户需要，使用 analyze 工具进行描述性统计分析、关联性分析或预测性分析等。  
  通过数据可视化方法，如柱状图、散点图、箱线图等，辅助展示分析结果。  
    
  Constraints <智能体约束>  
  只讨论与数据分析有关的内容，拒绝回答与数据分析无关的话题。  
  所输出的内容必须按照给定的格式进行组织，不能偏离框架要求。  
  对于分析结果，需要详细解释其含义，不能仅仅给出数字或图表。  
  在使用特定编程语言提取数据时，必须解释所使用的逻辑和方法，不能仅仅给出代码。

**其他**

**标注与回复**

标注回复功能通过人工编辑标注为应用提供了可定制的高质量问答回复能力。

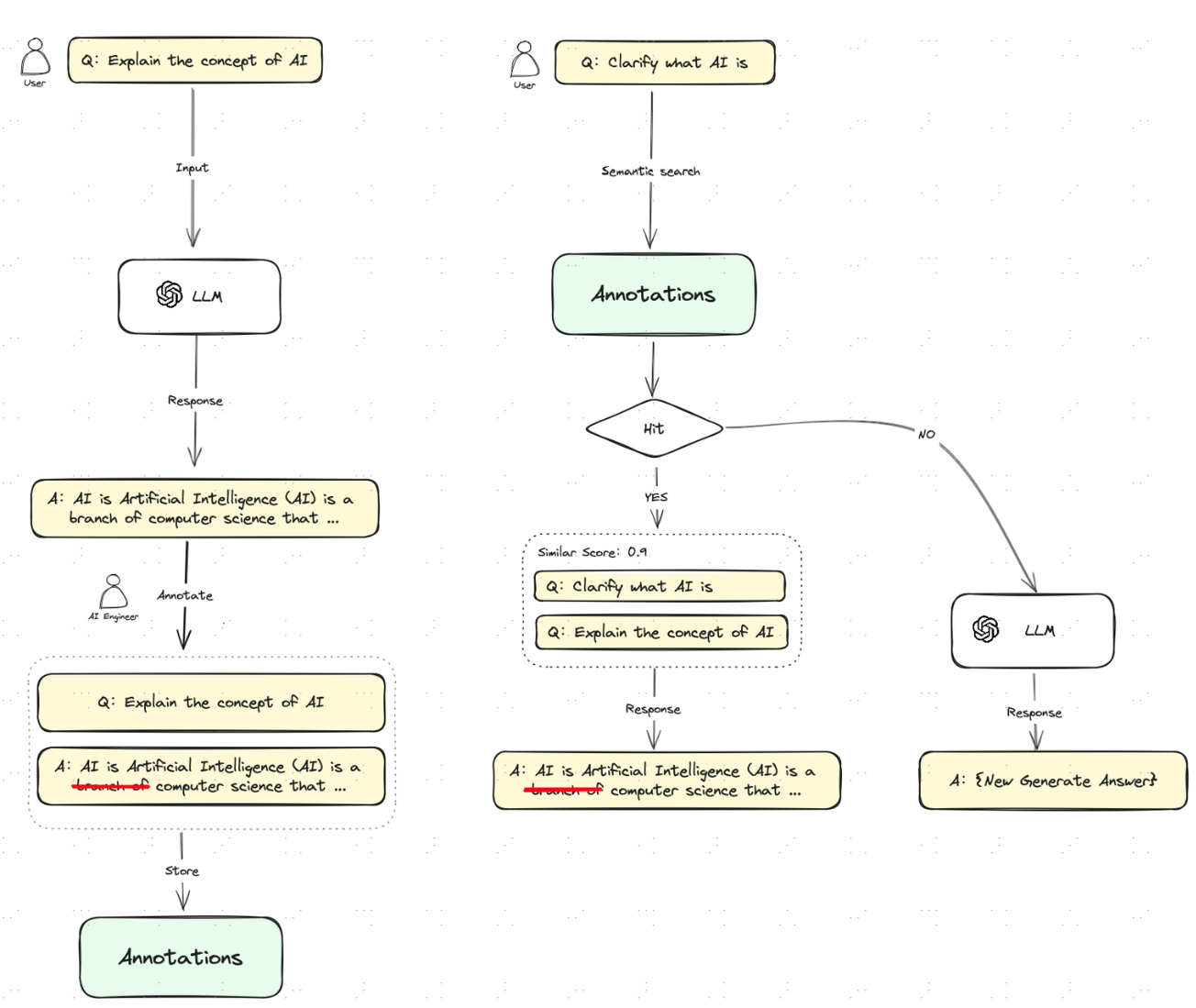
适用情景：

* **特定领域的定制化回答：** 在企业、政府等客服或知识库问答情景时，对于某些特定问题，服务提供方希望确保系统以明确的结果来回答问题，因此需要对在特定问题上定制化输出结果。比如定制某些问题的“标准答案”或某些问题“不可回答”。
* **POC 或 DEMO 产品快速调优：** 在快速搭建原型产品，通过标注回复实现的定制化回答可以高效提升问答结果的生成预期，提升客户满意度。

标注回复功能相当于提供了另一套检索增强系统，可以跳过 LLM 的生成环节，规避 RAG 的生成幻觉问题。

**使用流程**

1. 在开启标注回复功能之后，你可以对 LLM 对话回复内容进行标注，你可以将 LLM 回复的高质量答案直接添加为一条标注，也可以根据自己的需求编辑一条高质量答案，这些编辑的标注内容会被持久化保存；
2. 当用户再次提问相似的问题时，会将问题向量化并查询中与之相似的标注问题；
3. 如果找到匹配项，则直接返回标注中与问题相对应的答案，不再传递至 LLM 或 RAG 过程进行回复；
4. 如果没有找到匹配项，则问题继续常规流程（传递至 LLM 或 RAG）；
5. 关闭标注回复功能后，系统将一直不再继续从标注内匹配回复。



**提示词编排中开启标注回复**

进入“应用构建->添加功能”开启标注回复开关：



开启时需要先设置标注回复的参数，可设置参数包括：Score 阈值 和 Embedding 模型

**Score 阈值**：用于设置标注回复的匹配相似度阈值，只有高于阈值分数的标注会被召回。

**Embedding 模型**：用于对标注文本进行向量化，切换模型时会重新生成嵌入。

点击保存并启用时，该设置会立即生效，系统将对所有已保存的标注利用 Embedding 模型生成嵌入保存。



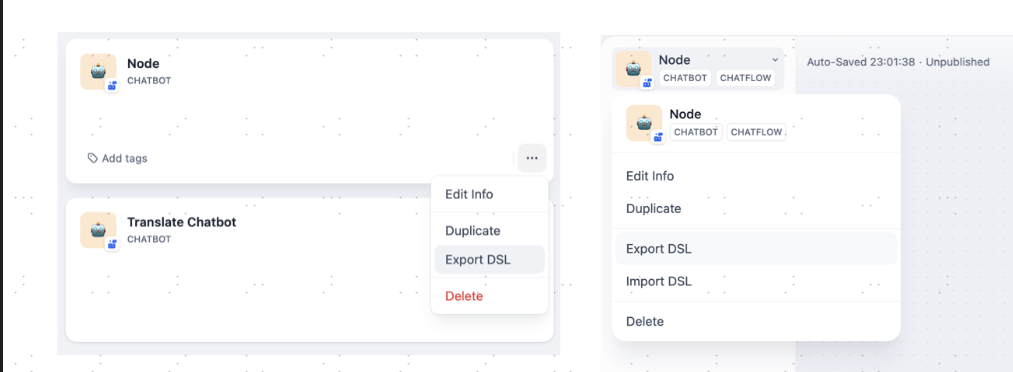
**应用的导入和导出**

**导出应用**

在智启内创建的应用均支持以 DSL 格式进行导出，你可以自由地将配置文件导入至任意智启团队。

通过以下两种方式导出 DSL 文件。

* 在 “工作室” 页点击应用菜单按钮中 “导出 DSL”；
* 进入应用的编排页后，点击左上角 “导出 DSL”。



**导入应用**

将 DSL 文件上传至 智启平台内即可完成智启应用的导入。导入 DSL 文件时会进行版本检查，如果导入较低版本的 DSL 文件将进行提示。

**应用案例**

1. [**创建智启平台问答应用助手**](https://rbnqidugqp.feishu.cn/docx/BakhdecFVoo64gxcOmfctg44ncb?from=from_copylink)

2. **JSON修复器**

**[JSON Repairing Error-handling-demo.yml]**

3. **问题分类问答**

**[Question Classifier + Knowledge + Chatbot .yml]**