WanWu Open API

文本问答API

文本生成应用无会话支持，适合用于翻译/文章写作/总结 AI 等等。

请求接口

Type	Instructions
方法	Http
请求URL	按照应用实际API根地址，例如：`http://localhost:8081/user/api/openapi/v1/rag/chat`
字符编码	UTF-8
请求类型	POST
鉴权方式(header参数)	`Authorization: Bearer {API Key}`
请求格式	`Content-Type: application/json`
响应格式	非流式：`Content-Type: application/json` 流 式：`Content-Type: text/event-stream`

请求参数

Parameter	Required	Type	Instructions
stream	否	bool	是否以流式接口的形式返回数据，默认为非流式。 true为流式，false为非流式。
query	是	string	用户提出的问题或提示语

响应参数

Parameter	Type	Instructions
code	int	状态码，用于表示请求成功或具体的错误类型
message	string	提示信息，通常用于提供关于code的详细解释或请求成功的确认信息
msg_id	string	提示信息ID
data	array	当前响应文本，包含了根据用户输入和知识库搜索得到的答案
history	array	包含之前对话历史的数组，用于上下文管理
finish	int	仅在流式返回中有该字段。表示流是否结束，0：未结束，1：正常结束，2：生成长度导致结束，3：异常结束，4：命中安全护栏结束

data

Parameter	Type	Instructions
output	string	当前响应文本内容片段
searchList	array	知识增强搜索结果

searchList

Parameter	Type	Instructions
kb_name	string	知识库名字
snippet	string	知识内容片段
title	string	文件标题

history

Parameter	Type	Instructions
query	string	请求文本
response	string	模型响应文本

调用示例

流式

```
curl -k --location 'http://localhost:8081/user/api/openapi/v1/rag/chat' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <API Key>' \
--data '{
    "stream": true,
    "query": "请一句话介绍元景万悟"
}'
```

非流式

```
curl -k --location 'http://localhost:8081/user/api/openapi/v1/rag/chat' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <API Key>' \
--data '{
    "stream": false,
    "query": "请一句话介绍元景万悟"
}'
```

响应示例

流式

```
data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "元景", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "万悟", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "是", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "联通", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "推出的", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "AI", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "工程化", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "平台", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "，", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "提供", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "模型", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "纳", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "管", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "、", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "工作", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "流", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "编排", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "、", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "知识", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "库", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "管理等", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "全套", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "功能", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "，", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "支持", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "企业" , "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "高效", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "构建", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "智能化", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "应用", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "并", "searchList": []}, "history": [], "finish":  0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "降低", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "AI", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "技术", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "门槛" , "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "。", "searchList": []}, "history": [], "finish": 0}


data: {"code": 0, "message": "success", "msg_id": "bf10ca0882a09ec6a4e1e26190a841b2", "data": {"output": "", "searchList": []}, "history": [], "finish": 1}
```

非流式

```
{
    "code": 0,
    "message": "success",
    "msg_id": "f89327a67dc05fec500f009b2c605401",
    "data": {
        "output": "元景万悟是联通推出的模块化AI工程化平台，提供从模型纳管到应用落地的完整工具链，支持多租户架构和企业级知识库等功能，帮助企业降低AI应用门槛并加速数字化转型。",
        "searchList": [
            {
                "kb_name": "元景万悟",
                "title": "README.pdf",
                "snippet": "库建设、复杂工作流编排等完整功能体系的AI工程化平台。平台采用模块化架构设计，支持灵活的功能扩 展和二次开发，在确保企业数据安全和隐私保护的同时，大幅降低了AI技术的应用门槛。无论是中小型企 业快速构建智能化应用，还是大型企业实现复杂业务场景的智能化改造，联通元景万悟Lite都能提供强有 力的技术支撑，助力企业加速数字化转型进程，实现降本增效和业务创新。 🔥 平台核心优势 ✓ 企业级工程化：提供从模型纳管到应用落地的完整工具链，解决LLM技术落地\"最后一公里\"问题 ✓ 开放开源生态：采用宽松友好的 Apache 2.0 License，支持开发者自由扩展与二次开发 ✓ 全栈技术支持：配备专业团队为生态伙伴提供 架构咨询、性能优化 全周期赋能 ✓ 多租户架构：提供多租户账号体系，满足用户成本控制、数据安全隔离、业务弹性扩展、行业定制\n化、快速上线及生态协同等核心需求 🚩 核心功能模块 1. 模型纳管（Model Hub） ▸ 支持 数百种专有/开源大模型（包括GPT、Claude、Llama等系列）的统一接入与生命周期管理 ▸ 深度适配 OpenAI API 标准 及 联通元景 生态模型，实现异构模型的无缝切换 ▸ 提供 多推理后端支持（vLLM、TGI等）与 自托管解决方案，满足不同规模企业的算力需求 2. 可视化工作流（Workflow Studio） ▸ 通过 低代码拖拽画布 快速构建复杂AI业务流程 ▸ 内置 条件分支、API、大模型、知识库、代码 等多种节点，支持端到端流程调试与性能分析 3. 企业级知识库、RAG Pipeline ▸ 提供 知识库创建→文档解析→向量化→检索→精排 的全流程知识管理能力，支持\npdf/docx/txt/xlsx/csv/pptx等 多种格式 文档，还支持网页资源的抓取和接入 ▸ 集成 多模态检索 、级联切分 与 自适应切分，显著提升问答准确率 4. 智能体开发框架（Agent Framework） ▸ 可基于 函数调用（Function Calling） 的Agent构建范式，支持工具扩展、私域知识库关联与多轮对\n话 ▸ 支持在线调试 5. 后端即服务（BaaS） ▸ 提供 RESTful API ，支持与企业现有系统（OA/CRM/ERP等）深度集成 ▸ 提供 细粒度权限控制，保障生产环境稳定运行
"
            },
            {
                "kb_name": "元景万悟",
                "title": "README.pdf",
                "snippet": "API + 应用程序导向 API + 应用程序导向 编程方法 ✅ ✅ 支持的LLMs ✅ ✅ RAG引擎 ✅ ✅ Agent ✅ ✅ 工作流 ✅ ✅ 可观测性 ✅ ✅ 本地部署 ✅ ❌ license友好 ✅ ❌ 多租户 🚀 快速开始 Docker安装 从源码安装 🎯 典型应用场景 智能客服：基于RAG+Agent实现高准确率的业务咨询与工单处理 知识管理：构建企业专属知识库，支持语义搜索与智能摘要生成 流程自动化：通过工作流引擎实现合同审核、报销审批等业务的AI辅助决 策 平台已成功应用于 金融、工业、政务 等多个行业，助力企业将LLM技术的理论价值转化为实际业务收 益。我们诚邀开发 者加入开源社区，共同推动AI技术的民主化进程。 ⚖️ 许可证 联通元景万悟Lite根据Apache License 2.0发布。"
            }
        ]
    },
    "history": [
        {
            "query": "请一句话介绍元景万悟",
            "response": "元景万悟是联通推出的模块化AI工程化平台，提供从模型纳管到应用落地的完整工具链，支持多租户架构和企业级知识库等功能，帮助企业降低AI应用门槛并加速数字化转型。"
        }
    ],
    "finish": 1
}
```

智能体创建对话API

智能体应用支持会话持久化，可将之前的聊天记录作为上下文进行回答，可适用于聊天/客服 AI 等。

请求接口

Type	Instructions
方法	Http
请求URL	按照应用实际API根地址，例如：`http://localhost:8081/user/api/openapi/v1/agent/conversation`
字符编码	UTF-8
请求类型	POST
鉴权方式(header参数)	`Authorization: Bearer {API Key}`
请求格式	`Content-Type: application/json`
响应格式	非流式：`Content-Type: application/json` 流 式：`Content-Type: text/event-stream`

请求参数

Parameter	Required	Type	Instructions
title	是	string	对话标题

响应参数

Parameter	Type	Instructions
code	int	状态码，用于表示请求成功或具体的错误类型
msg	string	提示信息，通常用于提供关于code的详细解释或请求成功的确认信息
data	array	当前响应文本，包含了根据用户输入和知识库搜索得到的答案

data

Parameter	Type	Instructions
conversation_id	string	当前响应文本片段ID

调用示例

```
curl -k --location
'http://localhost:8081/user/api/openapi/v1/agent/conversation' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <API Key>' \
--data '{
    "title": "你好，元景万悟"
}'
```

响应示例

```
{
    "code": 0,
    "data": {
        "conversation_id": "56"
    },
    "msg": ""
}
```

智能体对话API

智能体应用支持会话持久化，可将之前的聊天记录作为上下文进行回答，可适用于聊天/客服 AI 等。

请求接口

Type	Instructions
方法	Http
请求URL	按照应用实际API根地址，例如：`http://localhost:8081/user/api/openapi/v1/agent/chat`
字符编码	UTF-8
请求类型	POST
鉴权方式(header参数)	`Authorization: Bearer {API Key}`
请求格式	`Content-Type: application/json`
响应格式	非流式：`Content-Type: application/json` 流 式：`Content-Type: text/event-stream`

请求参数

Parameter	Required	Type	Instructions
conversation_id	是	string	历史响应文本片段ID
stream	否	bool	是否以流式接口的形式返回数据，默认为非流式。true为流式，false为非流式
query	是	string	用户提出的问题或提示语

响应参数

Parameter	Type	Instructions
code	int	状态码，用于表示请求成功或具体的错误类型
message	string	提示信息，通常用于提供关于code的详细解释或请求成功的确认信息
gen_file_url_list	array	模型生成输出的文件url列表
response	string	当前响应文本，包含了根据用户输入和知识库搜索得到的答案
search_list	array	知识增强搜索结果
history	array	包含之前对话历史的数组，用于上下文管理
usage	array	token使用量
finish	int	仅在流式返回中有该字段。表示流是否结束，0：未结束，1：正常结束，2：生成长度导致结束，3：异常结束，4：命中安全护栏结束

gen_file_url_list

Parameter	Type	Instructions
output_file_url	string	输出文件url

history

Parameter	Type	Instructions
query	string	请求文本
response	string	模型响应文本

search_list

Parameter	Type	Instructions
kb_name	string	知识库名字
snippet	string	知识内容片段
title	string	文件标题

usage

Parameter	Type	Instructions
prompt_tokens	int	输入提示词token数
completion_tokens	int	输出文本token数
total_tokens	int	输入加输出总的token数

调用示例

流式

```
curl -k --location 'http://localhost:8081/user/api/openapi/v1/agent/chat' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <API Key>' \
--data '{
    "stream": true,
    "conversation_id": "56",
    "query": "请一句话介绍元景万悟"
}'
```

非流式

```
curl -k --location 'http://localhost:8081/user/api/openapi/v1/agent/chat' \
--header 'Content-Type: application/json' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <API Key>' \
--data '{
    "stream": false,
    "conversation_id": "56",
    "query": "请一句话介绍元景万悟"
}'
```

响应示例

流式

```
data: {"code": 0, "message": "success", "response": "元景", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "万悟", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "是", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "联通", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "推出", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "的", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "AI", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "工程", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "化", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "平台", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "，", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "提供", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "从", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "模型", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "纳", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "管", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "到", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "应用", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "落", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "地的", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "完整", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "工具", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "链", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "，", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "支持", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "企业", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "级", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "AI", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "应用的", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total _tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "快速", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "构建", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0,  "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "与", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "智能化", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "改造", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "【", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "1", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "^", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "】", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "。", "gen_file_url_list": [], "history": [], "finish": 0, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}


data: {"code": 0, "message": "success", "response": "", "gen_file_url_list": [], "history": [], "finish": 1, "usage": {"prompt_tokens": 0, "completion_tokens": 0, "total_tokens": 0}, "search_list": [], "qa_type": [1]}
```

非流式

```
{
    "code": 0,
    "message": "success",
    "response": "元景万悟是联通推出的AI工程化平台，提供从模型纳管到应用落地的完整工具链，支持企业级知识库、RAG Pipeline、智能体开发等功能，助力企业数字化转型【1^】【2^】。",
    "gen_file_url_list": [

    ],
    "search_list": [
        {
            "kb_name": "元景万悟",
            "title": "README.pdf",
            "snippet": "库建设、复杂工作流编排等完整功能体系的AI工程化平台。平台采用模块化架构设计，支持灵活的功能扩 展和二次开发，在确保企业数据安全和隐私保护的同时，大幅降低了AI技术的应用门槛。无论是中小型企 业快速构建智能化应用，还是大型企业实现复杂业务场景的智能化改造，联通元景万悟Lite都能提供强有 力的技术支撑，助力企业加速数字化转型进程，实现降本增效和业务创新。 🔥 平台核心优势 ✓ 企业级工程化：提供从模型纳管到应用落地的完整工具链，解决LLM技术落地\"最后一公里\"问题 ✓ 开放开源生态：采用宽松友好的 Apache 2.0 License，支持开发者自由扩展与二次开发 ✓ 全栈技术支持：配备专业团队为生态伙伴提供 架构咨询、性能优化 全周期赋能 ✓ 多租户架构：提供多租户账号体系，满足用户成本控制、数据安全隔离、业务弹性扩展、行业定制\n化、快速上线及生态协同等核心需求 🚩 核心功能模块 1. 模型纳管（Model Hub） ▸ 支持 数百种专有/开源大模型（包括GPT、Claude、Llama等系列）的统一接入与生命周期管理 ▸ 深度适配 OpenAI API 标准 及 联通元景 生态模型，实现异构模型的无缝切换 ▸ 提供 多推理后端支持（vLLM、TGI等）与 自托管解决方案，满足不同规模企业的算力需求 2. 可视化工作流（Workflow Studio） ▸ 通过 低代码拖拽画布 快速构建复杂AI业务流程 ▸ 内置 条件分支、API、大模型、知识库、代码 等多种节点，支持端到端流程调试与性能分析 3. 企业级知识库、RAG Pipeline ▸ 提供 知识库创建→文档解析→向量化→检索→精排 的全流程知识管理能力，支持\npdf/docx/txt/xlsx/csv/pptx等 多种格式 文档，还支持网页资源的抓取和接入 ▸ 集成 多模态检索 、级联切分 与 自适应切分，显著提升问答准确率 4. 智能体开发框架（Agent Framework） ▸ 可基于 函数调用（Function Calling） 的Agent构建范式，支持工具扩展、私域知识库关联与多轮对\n话 ▸ 支持在线调试 5. 后端即服务（BaaS） ▸ 提供 RESTful API ，支持与企业现有系统（OA/CRM/ERP等）深度集成 ▸ 提供 细粒度权限控制，保障生产环境稳定运行
"
        },
        {
            "kb_name": "元景万悟",
            "title": "README.pdf",
            "snippet": "API + 应用程序导向 API + 应用程序导向 编程方法 ✅ ✅ 支持的LLMs ✅ ✅ RAG引擎 ✅ ✅ Agent ✅ ✅ 工作流 ✅ ✅ 可观测性 ✅ ✅ 本地部署 ✅ ❌ license友好 ✅ ❌ 多租户 🚀 快速开始 Docker安装 从源码安装 🎯 典型应用场景 智能客服：基于RAG+Agent实现高准确率的业务咨询与工单处理 知识管理：构建企业专属知识库，支持语义搜索与智能摘要生成 流程自动化：通过工作流引擎实现合同审核、报销审批等业务的AI辅助决策 平台已成功应用于 金融、工业、政务 等多个行业，助力企业将LLM技术的理论价值转化为实际业务收 益。我们诚邀开发者加入开源社区，共同推动AI技术的民主化进程。 ⚖️ 许可证 联通元景万悟Lite根据Apache License 2.0发布。"
        }
    ],
    "history": [
        {
            "query": "请一句话介绍元景万悟",
            "response": "元景万悟是联通推出的AI工程化平台，提供从模型纳管到应用落地的完整工具链，支持企业级知识库、RAG Pipeline、智能体开发等功能，助力企业数字化转型【1^】【2^】。"
        }
    ],
    "usage": {
        "completion_tokens": 0,
        "prompt_tokens": 0,
        "total_tokens": 0
    },
    "finish": 1
}
```