RAG Pipeline API 接口文档-【环境准备&API接口说明】

环境准备

1. 安装必要的依赖：

```
# 默认llm环境已经安装，可不需要
pip install fastapi uvicorn -i https://pypi.tuna.tsinghua.edu.cn/simple
```

2. 启动服务：

```
uvicorn api:app --reload --host 0.0.0.0 --port 8000 --workers 4
```

3. 访问API文档：

- 启动服务后访问：http://localhost:8000/docs
- 可以在线测试API功能

API接口说明

1 向量检索RAG接口

接口信息：

- 请求方法：POST

- 接口路径：`/rag_pipeline`
- 接口描述：基于向量检索的RAG对话系统

请求参数：

```
{
    "query": "string",              // 必填，用户的问题
    "context_query": "string",      // 可选，用于检索上下文的查询 默认与用户问题相同
    "k": 3,                         // 可选，返回的相似文档数量
    "context_query_type": "query", // 可选，检索类型：query/vector/doc
    "stream": true,                 // 可选，是否使用流式响应
    "temperature": 0.1              // 可选，生成温度
}
```

响应格式：

- 非流式响应：

```
{
    "response": "AI助手的回答",
    "context": ["相关上下文1", "相关上下文2"]
}
```

- 流式响应

```
data: {"type": "context", "data": ["1: ", "", ""]}
data: {"type": "response", "data": "请假"}
data: {"type": "response", "data": "流程"}
```

2 知识图谱RAG接口

接口信息：

- 请求方法：POST
- 接口路径：`/graph_rag`
- 接口描述：基于知识图谱的RAG对话系统

请求参数：

```
{
    "query": "string",          // 必填，用户的问题
    "exclude_content": true,    // 可选，是否排除详细内容
    "stream": true,            // 可选，是否使用流式响应
    "temperature": 0.1         // 可选，生成温度
}
```

响应格式：

- 非流式响应：

```
{
    "response": "AI助手的回答",
    "context": ["相关上下文1", "相关上下文2"]
}
```

- 流式响应

```
data: {"type": "context", "data": ["1: ", "", ""]}
data: {"type": "response", "data": "请假"}
data: {"type": "response", "data": "流程"}
```

3 Agent路由接口

接口信息：

- 请求方法：POST
- 接口路径：`/agent`
- 接口描述：智能路由的Agent系统，可以根据问题类型自动选择合适的工具进行回答

请求参数：

```
{
    "query": "string",          // 必填，用户的问题
    "history": [                // 可选，对话历史，默认为空
        ["用户问题1", "AI回答1"],
        ["用户问题2", "AI回答2"]
    ]
}
```

响应格式：

- 非流式响应：

```
{
    "response": "AI助手的回答"
}
```

- 流式响应

```
data: {"type": "context", "data": ["1: ", "", ""]}
data: {"type": "response", "data": "请假"}
data: {"type": "response", "data": "流程"}
```

下一节：RAG Pipeline API 接口文档-【使用示例】

下一节