



**知识图谱问答系统使用说明**

**编者：3023244322蒋茜，3023244328马佳一**

**资料提供：3023244327邵玺冉，3023244338张婉毓**

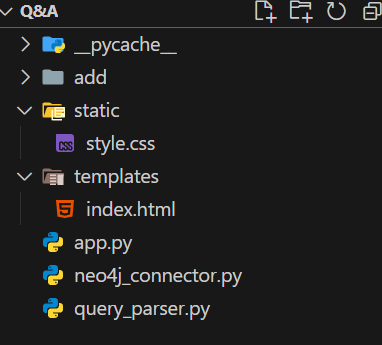
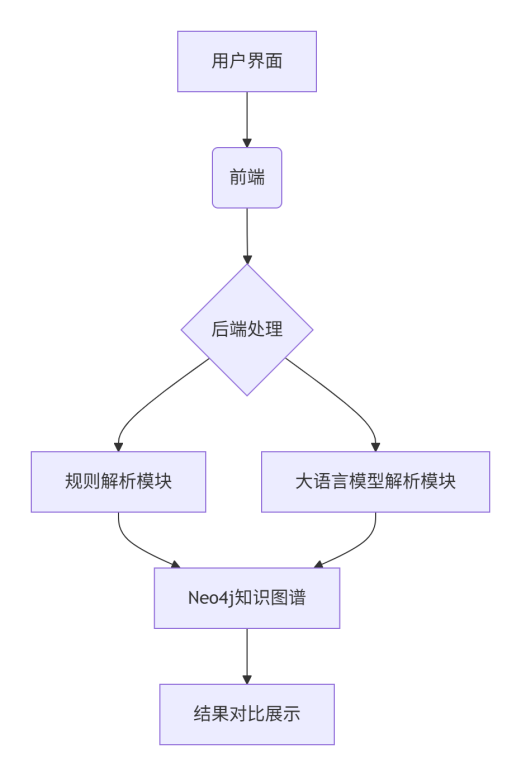
**（姓名不分先后，按首字母排序）**

**2025年 6 月 30 日**

## 1. 系统概述

这是一个基于neo4j知识图谱的智能问答系统，结合了传统的规则解析和大语言模型API连接，能够对领域知识问题进行智能回答，并展示**两种不同方式**的不同结果，下面以认知科学相关知识库进行实例展示。

## 2. 系统组成



## 3. 使用前准备

### 运行环境

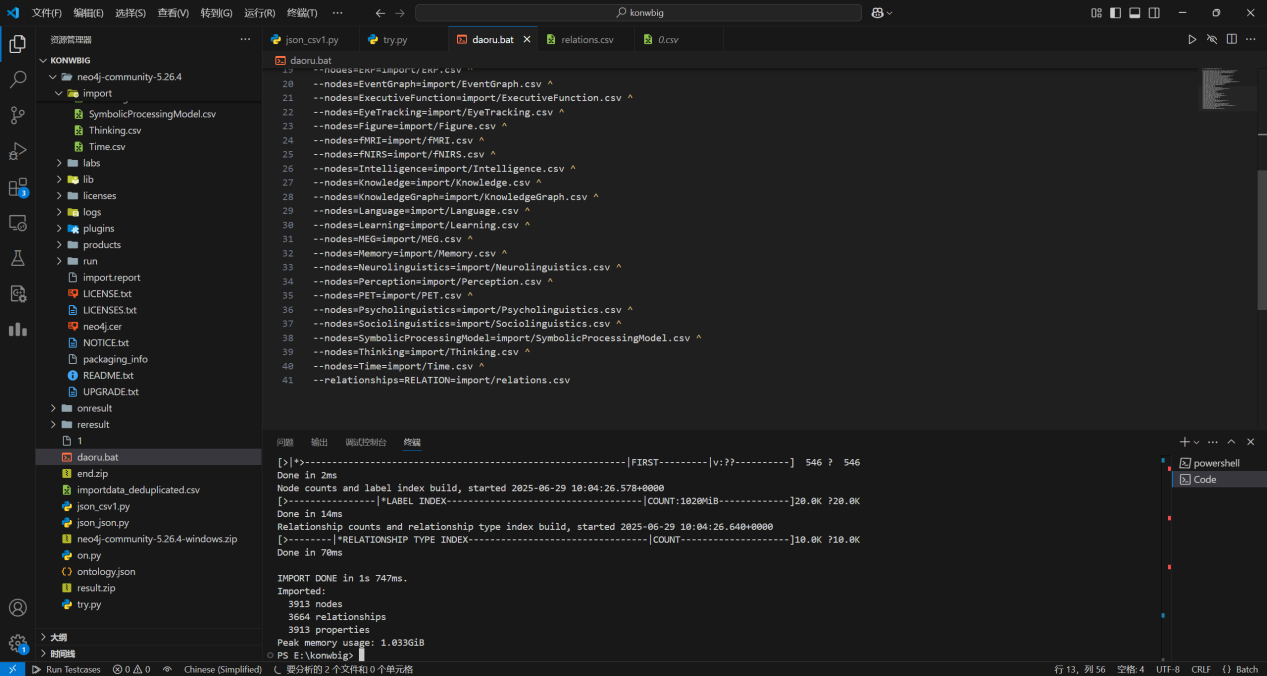
* Python 3.8+
* Flask框架
* Neo4j数据库(3.5+)
* 前端基础：HTML/CSS/JavaScript

### 安装步骤

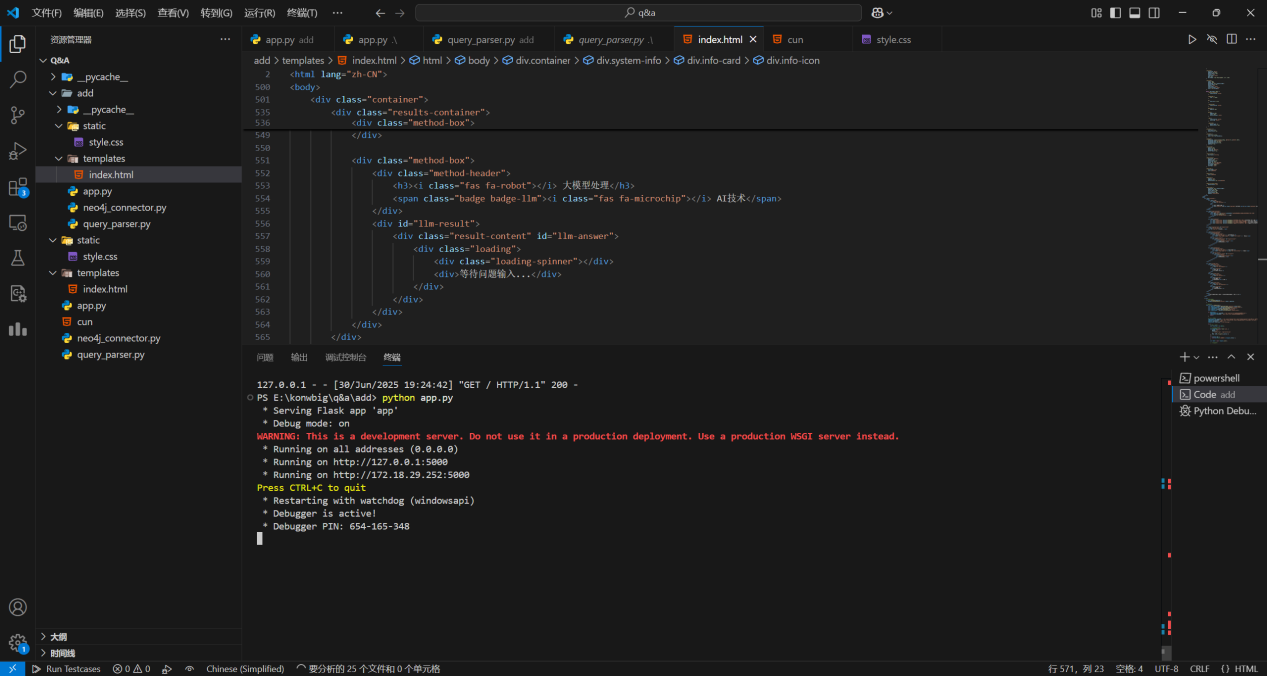
1. 安装依赖包：

pip install flask neo4j requests

1. 启动Neo4j数据库（默认端口7687）
2. 导入你所需要的数据

  
4. 运行系统：

python app.py



## 4. 使用流程

### 步骤1：访问系统

在浏览器中访问：http://localhost:5000

### 步骤2：输入问题

在输入框中输入自然语言问题，例如：

* "比尔盖茨的母亲是谁？"
* "认知科学相关的技术有什么？"

### 步骤3：点击"提问"按钮

系统会同时启动两种方法进行处理：

* 左侧：原有规则处理方法
* 右侧：大语言模型处理方法

### 步骤4：查看结果 结果包含：

* 生成的Cypher查询语句（可扩展查看完整语句）
* 查询执行结果
* 处理响应时间

## 5. 系统结果解读



### 5.1 结果区域介绍

**Cypher查询框**：展示系统生成的查询语句

**结果区域**：显示查询返回的结果

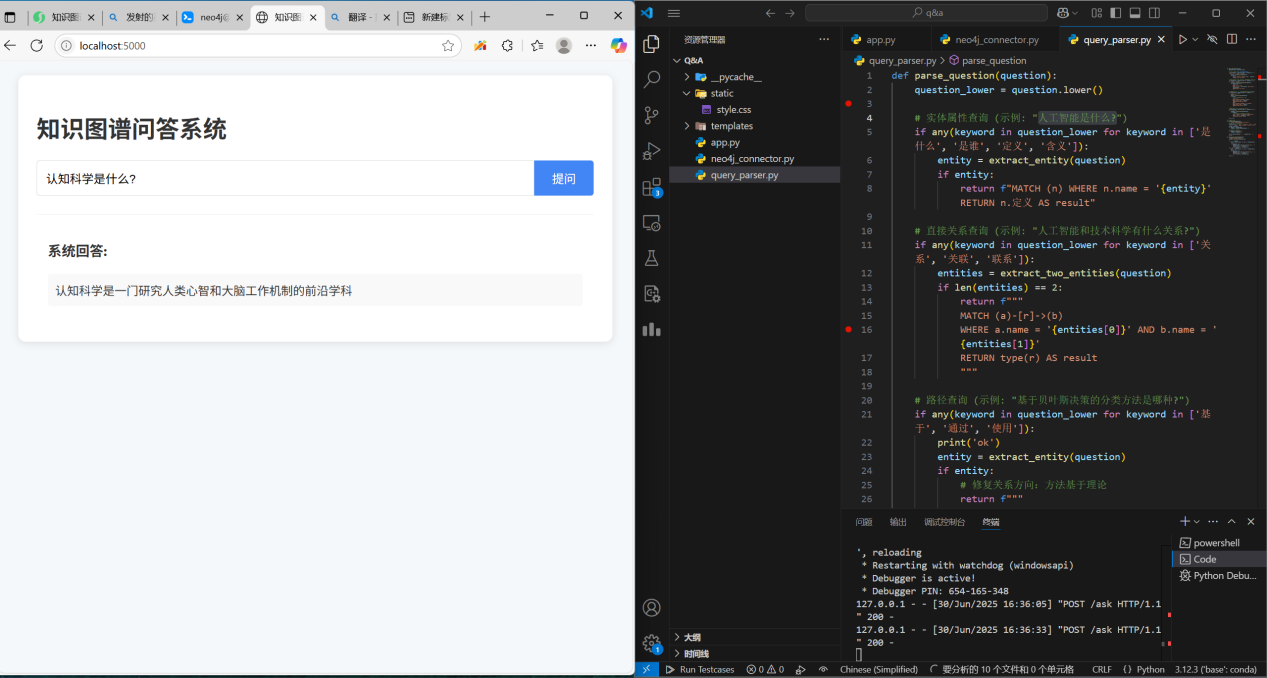
**时间统计**：显示整个处理耗时

### 5.2 不同结果类型解读

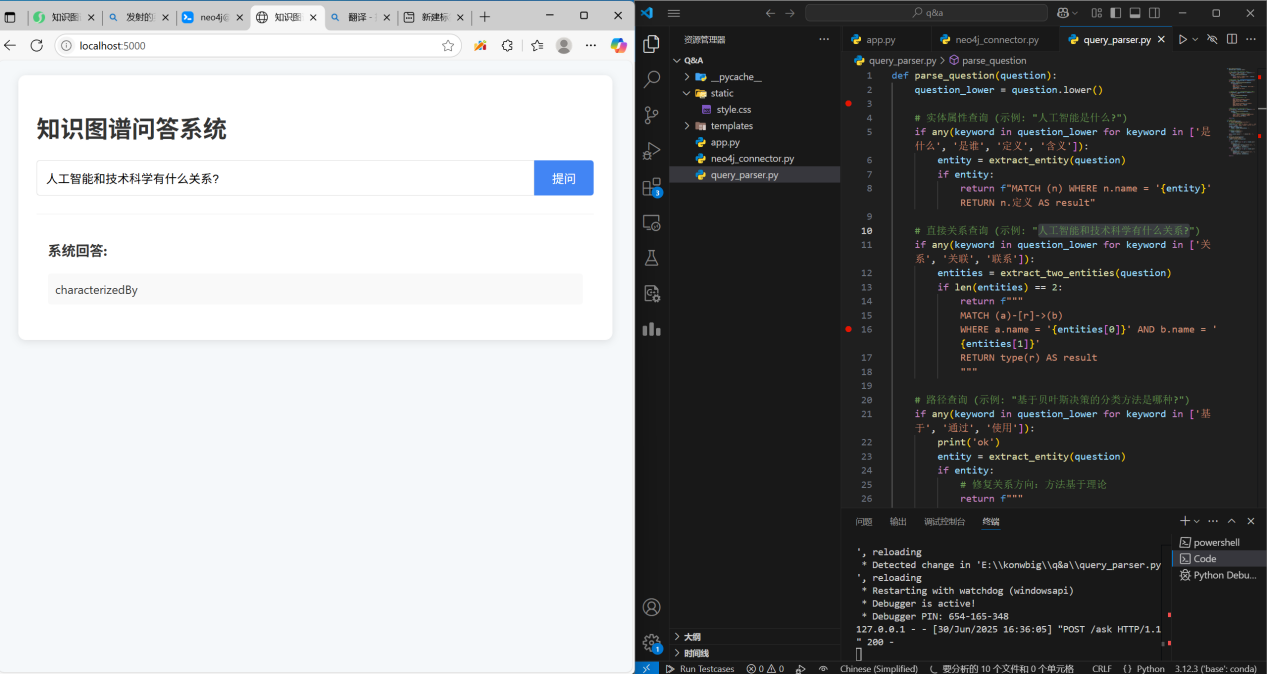
**标准查询结果**：

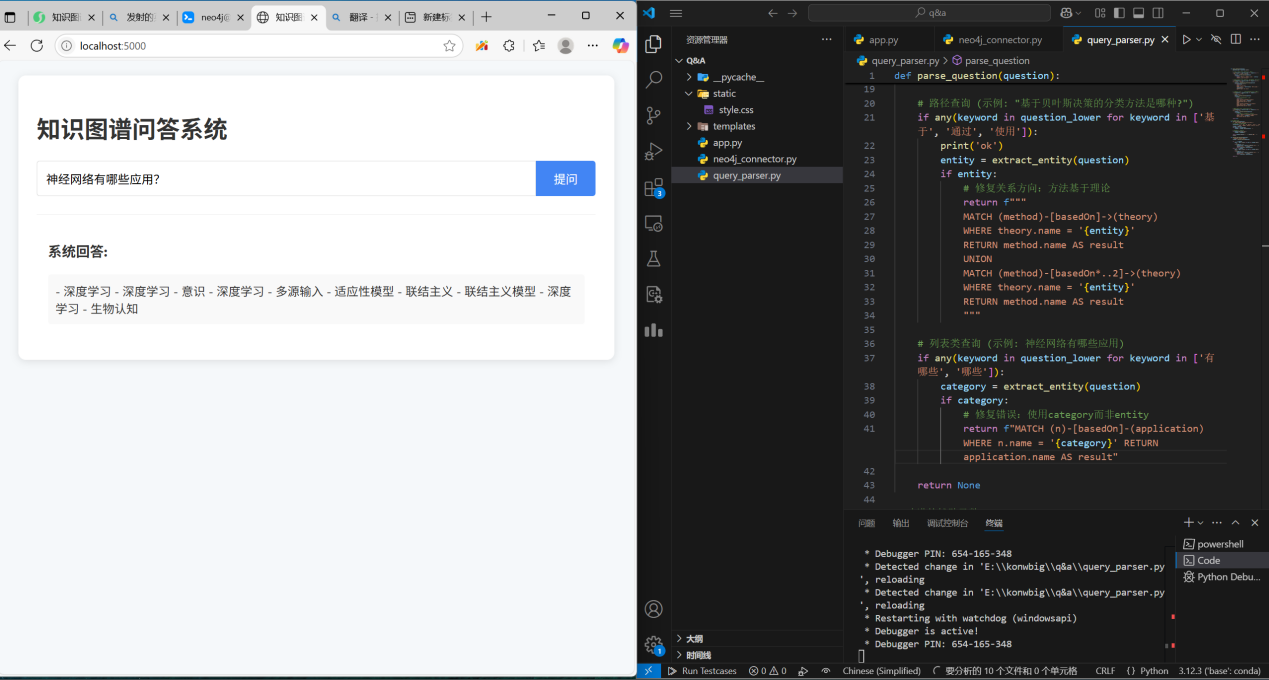
(下面展示的为1.0版本的标准查询）

1. 实体属性查询

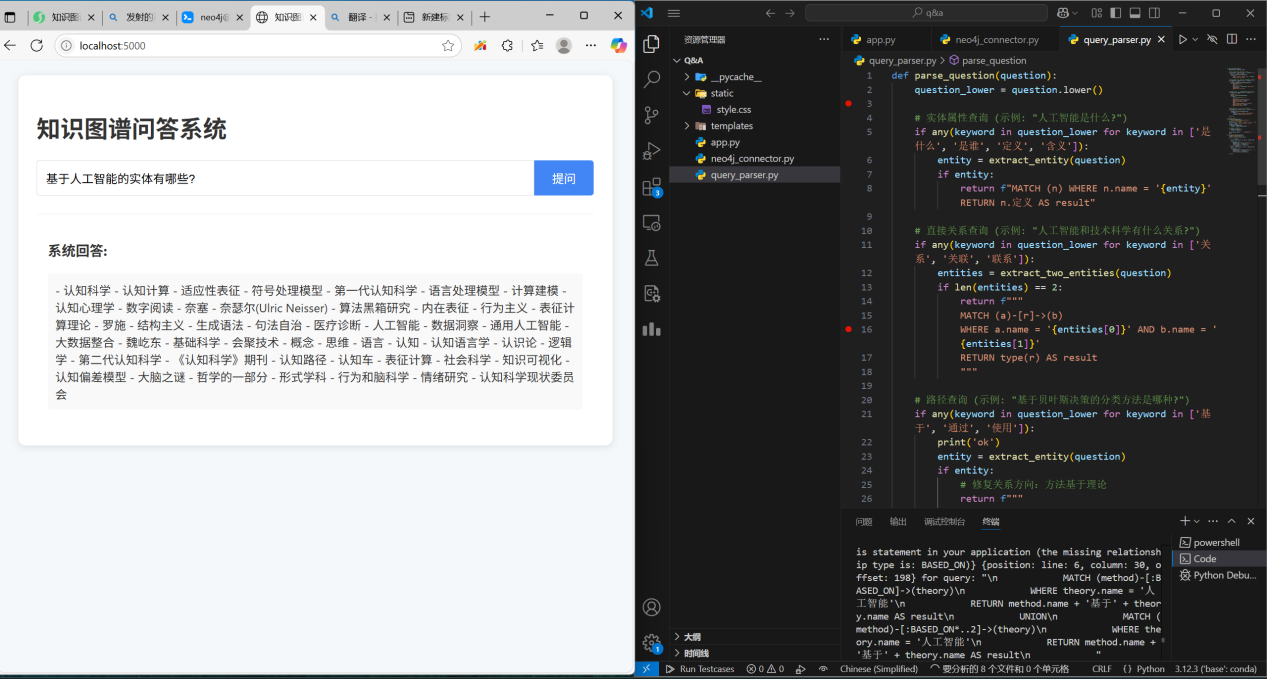


1. 直接关系查询





1. 简单路径查询



这些已经全部移植到2.0版本：





但是如图可见，大模型的api调用并不稳定，很容易出现调用失败

### 5.3 结果对比分析



如图所示，大模型可以处理规则引擎所不能处理的一些问题

二者之间的对比：

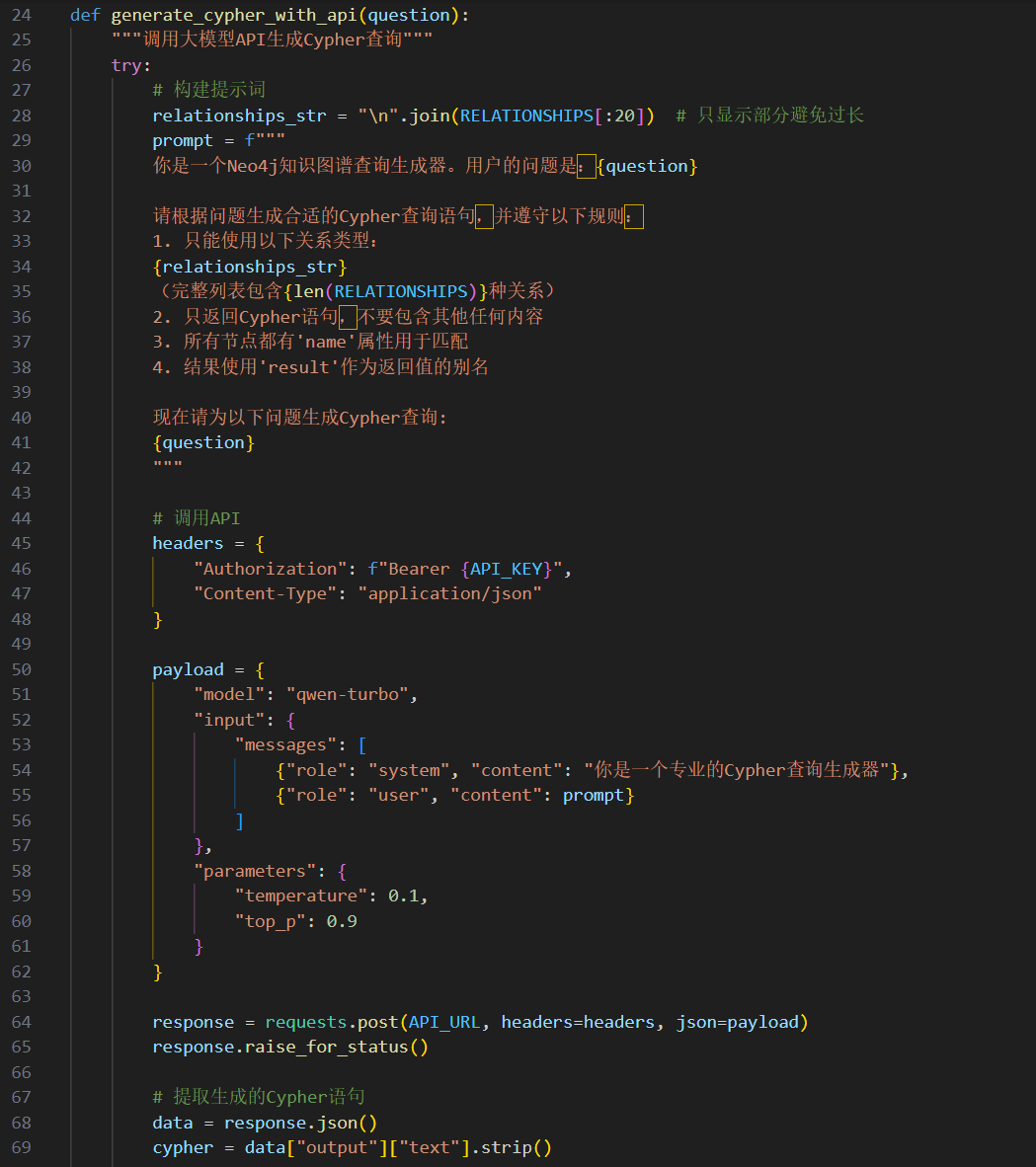
| **问题类型** | **规则处理方法** | **大模型处理方法** |
| --- | --- | --- |
| 基础查询 | 快速但有限 | 容易调用失败 |
| 复杂关系查询 | 可能失败 | 成功率更高 |
| 新颖问题 | 必然失败 | 尝试推理 |

## 6. 系统亮点

### 6.1 双引擎并行处理架构



### 6.2 大模型交互技术



## 7.项目总结

### 过程关键点：

**知识图谱建模**：设计合适的图数据库模式

**查询语言转换**：从自然语言到Cypher的转换

**API集成**：大语言模型服务的调用和结果处理

### 项目亮点：

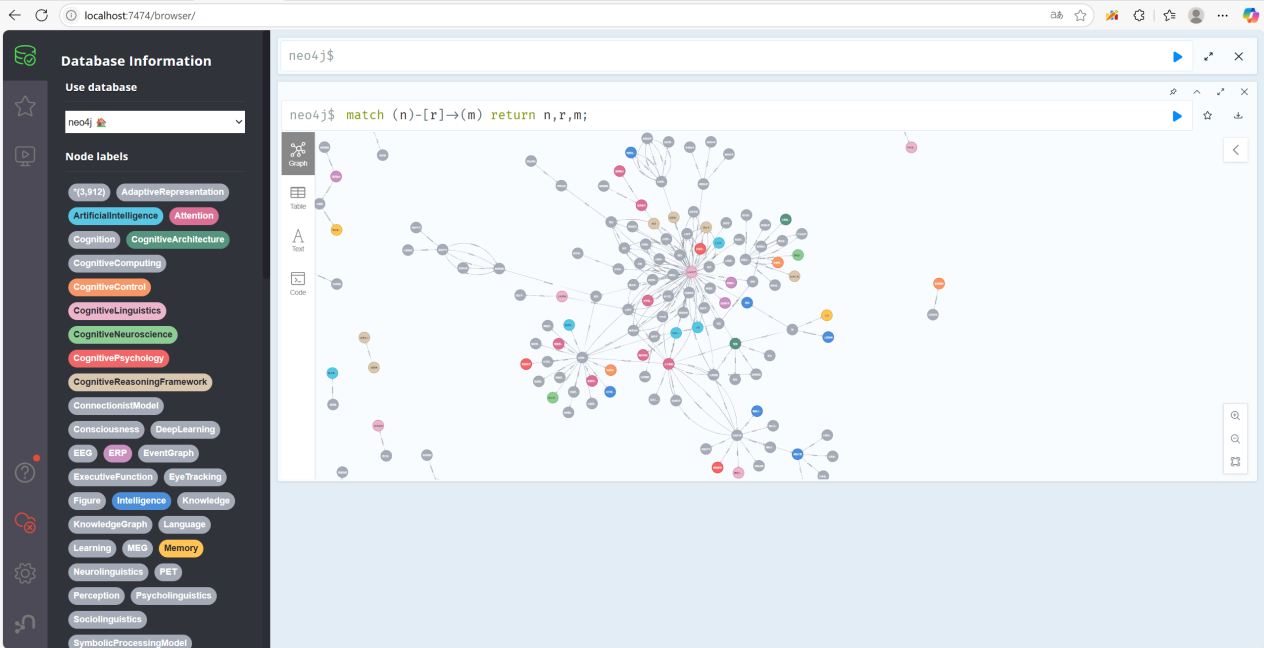
* 创新的双处理引擎架构
* 实时的处理时间对比
* 清晰的查询过程可视化
* 健壮的错误处理机制

## 8. 常见问题解答

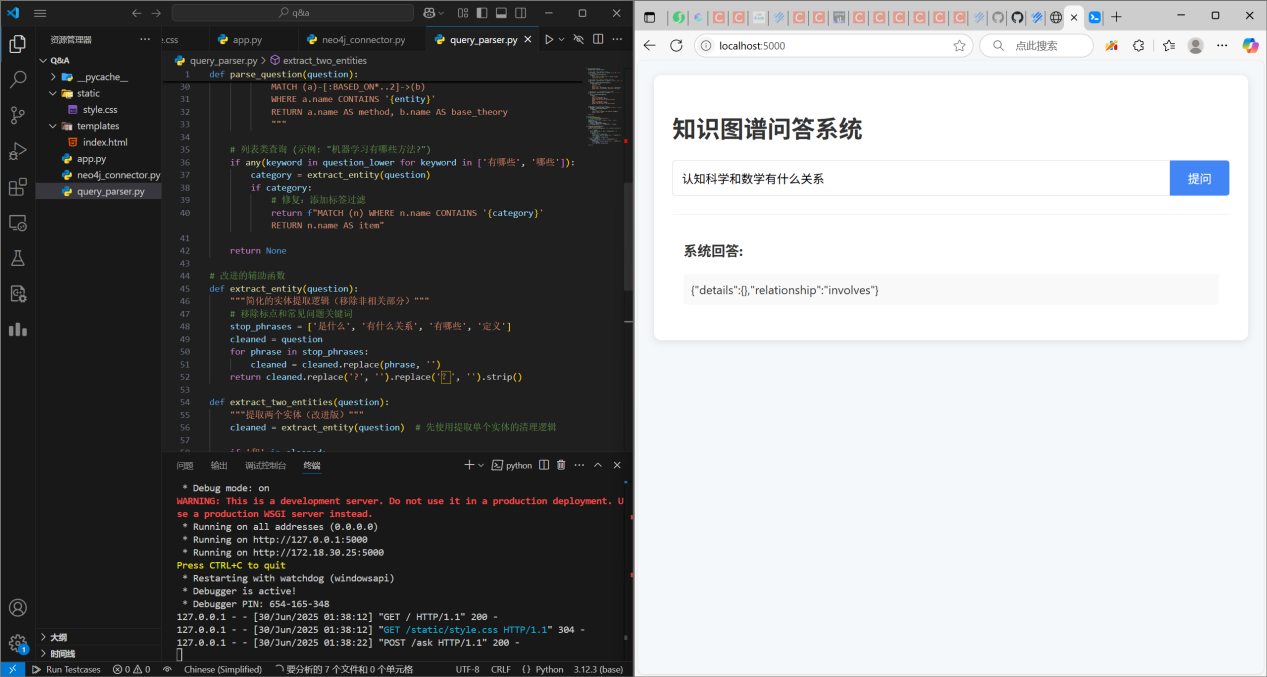
​**​Q: 系统返回"未生成Cypher查询"怎么办？​**​  
A: 尝试简化问题结构，或使用更标准的专业术语提问

​**​Q: 为什么大模型方法有时比规则方法更慢？​**​  
A: 大模型需要额外API调用时间，但处理复杂问题有优势

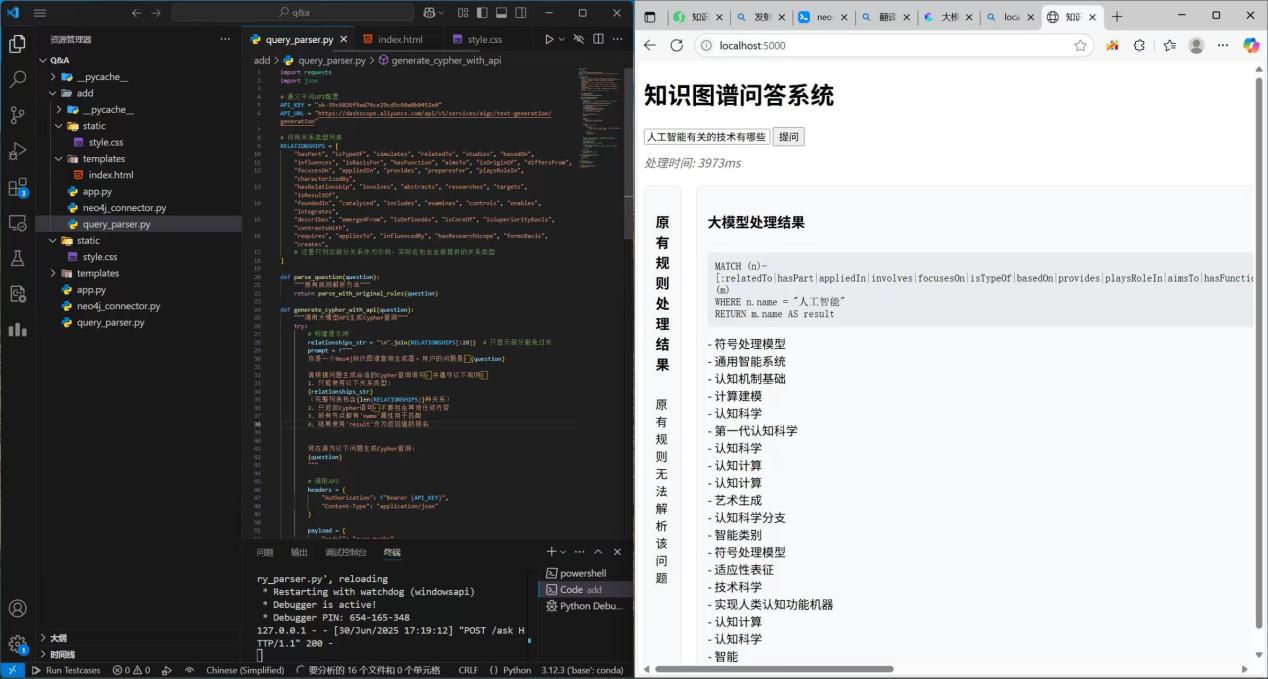
​**​Q: 如何扩展系统的知识库？​**​  
A: 通过Neo4j添加新的节点和关系即可



## 9. 结语

本知识图谱问答系统作为结课项目，完整实现了从前端到后端、从规则系统到人工智能的多种技术栈整合。我们经过了初步的1.0版本，只实现了简单的模版匹配：

再进行与大模型api的结合，实现了问答系统的创新性提升,2.0版本诞生：



最后我们进行了页面升级，更好地实现了与用户的交互：



3.0最终版就此出现

通过这个过程，我们对于知识工程的基本架构和流程有了更为清晰的掌握，并且动手实践了很多之前没有尝试过的技术栈，真的很累，但是也收获颇多。