# ChatGraph - 企业知识图谱智能问答系统

ChatGraph 是一个基于图数据库（Neo4j）与大语言模型（LLM）的对话式图谱查询系统，旨在通过自然语言交互实现对图数据库的查询和分析。

## 🌟 项目概述

### 核心功能

* **自然语言转Cypher查询**：通过LLM将用户输入的自然语言转换为图数据库可执行的Cypher语句
* **图谱数据展示**：执行Cypher语句后以可视化方式展示图谱数据
* **智能问答交互**：基于Gradio构建的Web界面，支持流式输出和图形化展示

### 目标用户

* 数据分析师
* 图数据库使用者
* AI研究人员

### 解决的核心问题

降低图数据库使用门槛，使用户可以通过自然语言进行图谱查询，而无需掌握Cypher等专业查询语言。

## 🏗️ 系统架构

┌─────────────────┐ ┌──────────────────┐ ┌────────────────────┐  
│ Gradio前端 │───▶│ GraphAgent │───▶│ Neo4j图数据库 │  
│ (用户交互界面) │ │ (自然语言处理) │ │ (数据存储) │  
└─────────────────┘ └──────────────────┘ └────────────────────┘  
 │  
 ▼  
 ┌──────────────────┐  
 │ FAISS向量库 │  
 │ (知识库检索) │  
 └──────────────────┘

### 主要组件说明

1. **Gradio前端**：提供用户友好的Web界面，支持自然语言输入和结果展示
2. **GraphAgent**：核心处理模块，负责自然语言理解、Cypher生成和执行
3. **Neo4j图数据库**：存储企业关系网络数据，包括公司、人员及其关系
4. **FAISS向量库**：存储图数据库Schema和Cypher示例，支持语义检索

## 🛠️ 技术栈

* **后端语言**：Python 3.x
* **图数据库**：Neo4j
* **AI框架**：LangChain
* **大语言模型**：DeepSeek/Qwen（通过DashScope API）
* **向量检索**：FAISS
* **前端框架**：Gradio
* **数据生成**：Faker

## 📊 数据模型

系统包含以下节点和关系类型：

### 节点类型

1. **Company（公司）**
   * name：公司名称
   * credit\_code：统一社会信用代码
   * company\_id：公司ID
   * reg\_date：注册日期
   * reg\_capital：注册资本
2. **Person（人员）**
   * name：姓名
   * person\_id：人员ID
   * age：年龄
   * gender：性别

### 关系类型

1. **COMPANY\_INVESTMENT（企业投资）**：公司→公司
2. **PERSON\_INVESTMENT（人员投资）**：人员→公司
3. **COM\_BRANCH（分支机构）**：公司→公司
4. **PER\_JOB（任职）**：人员→公司
5. **PER\_LEGAL\_PERSON（法人）**：人员→公司

## 🚀 快速开始

### 环境准备

1. 安装Python 3.x
2. 安装Neo4j并启动服务
3. 安装依赖：

* pip install -r requirements.txt

### 配置

在[config.py](file:///Users/chenjiarui/PythonProject/ChatGraph/config.py)中配置Neo4j连接信息：

NEO4J\_URI = "bolt://localhost:7687"  
NEO4J\_USER = "neo4j"  
NEO4J\_PASSWORD = "your\_password"

### 数据准备

1. 生成模拟数据：

* python data\_generator.py --companies 150 --persons 400

1. 导入数据到Neo4j：

* python import\_data.py --companies 150 --persons 400 --clear

### 启动应用

python gradio\_app.py

访问 http://localhost:7860 使用系统。

## 🧪 使用示例

### 支持的查询类型

* 人员投资关系查询：“张三投资了哪些公司？”
* 公司法定代表人查询：“阿里巴巴的法定代表人是谁？”
* 分支机构查询：“腾讯有哪些分公司？”
* 任职关系查询：“李四在哪些公司担任高管？”
* 复杂路径查询：“A公司三度以内的投资关系涉及哪些公司？”

### 系统特色

1. **智能Cypher生成**：基于用户问题自动生成准确的Cypher查询语句
2. **错误自动修复**：Cypher执行出错时自动尝试修复
3. **关系图可视化**：自动生成Mermaid格式的关系网络图
4. **对话历史管理**：支持对话历史记录和导出
5. **流式输出**：支持逐步输出处理结果，提升用户体验

## 📁 项目结构

ChatGraph/  
├── config.py # 配置文件  
├── data\_generator.py # 模拟数据生成器  
├── import\_data.py # 数据导入脚本  
├── db\_connector.py # 数据库连接器  
├── graph\_agent.py # 图查询代理核心  
├── gradio\_app.py # Gradio前端应用  
├── test\_graph\_agent.py # 测试脚本  
├── database\_schema.md # 数据库结构说明  
├── cypher\_examples.md # Cypher查询示例  
├── mermaid\_examples.md # 关系图示例  
└── requirements.txt # 依赖列表

## 🧠 核心模块详解

### GraphAgent（核心处理模块）

* 使用LangChain框架集成大语言模型
* 利用FAISS向量库存储和检索知识库信息
* 实现自然语言到Cypher的转换
* 支持Cypher执行错误自动修复
* 提供流式输出功能

### 数据生成与导入

* 使用Faker库生成逼真的中文企业/人员数据
* 支持自定义数据规模（公司数量、人员数量）
* 保证数据合理性和一致性

### 前端界面

* 基于Gradio构建的现代化Web界面
* 支持流式输出，实时显示处理进度
* 自动生成并渲染关系网络图
* 提供对话历史导出功能

## 🔧 开发与测试

### 运行测试

python test\_graph\_agent.py

### 自定义配置

可通过命令行参数调整数据生成规模：

python data\_generator.py --companies 300 --persons 800 --seed 42

## 📈 应用场景

1. **企业关系分析**：分析企业间的投资、合作、竞争关系
2. **风险评估**：通过关系网络评估企业风险传导路径
3. **商业情报**：挖掘企业人员和组织的复杂关系
4. **数据探索**：通过自然语言快速探索图数据库中的信息

## 📄 许可证

本项目仅供学习和研究使用。

## 🤝 贡献

欢迎提交Issue和Pull Request来改进这个项目。