

- 项目名称
  - 目录
  - 简介
  - 环境要求
  - 安装步骤
    - 1. 创建 Conda 虚拟环境
    - 2. 安装依赖
    - 3. 使用 Docker
      - 3.1 拉取指定的 Docker 镜像
      - 3.2 创建名为 test 的容器
  - 运行项目
    - 1. 激活 Conda 环境
    - 2. 启动 Docker 容器
    - 2. 运行 run.py
  - 常见问题
    - 问题 1: 运行 run.py 时出错

## 项目名称

---

Official Repository of "GraphTeam: Facilitating Large Language Model-based Graph Analysis via Multi-Agent Collaboration".

## 目录

---

- 简介
- 环境要求
- 安装步骤
  - 1. 创建 Conda 虚拟环境
  - 2. 安装依赖
  - 3. 使用 Docker
- 运行项目
  - 1. 激活 Conda 环境
  - 2. 运行 `run.py`
- 常见问题

# 简介

---

Graphs are widely used for modeling relational data in real-world scenarios, such as social networks and urban computing. While large language models (LLMs) have achieved strong performance in many areas, existing LLM-based graph analysis approaches either integrate graph neural networks (GNNs) for specific machine learning tasks (e.g., node classification), limiting their transferability, or rely solely on LLMs' internal reasoning ability, resulting in suboptimal performance. To address these limitations, we take advantage of recent advances in LLM-based agents, which have shown capabilities of utilizing external knowledge or tools for problem solving. By simulating human problem-solving strategies such as analogy and collaboration, we propose a multi-agent system based on LLMs named GraphTeam, for graph analysis. GraphTeam consists of five LLM-based agents from three modules, and the agents with different specialties can collaborate with each other to address complex problems. Specifically, (1) input-output normalization module: the question agent extracts and refines four key arguments (e.g., graph type and output format) from the original question, facilitating the problem understanding, and the answer agent organizes the results to meet the output requirement; (2) external knowledge retrieval module: we first build a knowledge base consisting of relevant documentation and experience information, and then the search agent retrieves the most relevant entries from the knowledge base for each question. (3) problem-solving module: given the retrieved information from search agent, the coding agent uses established algorithms via programming to generate solutions, and in case the coding agent does not work, the reasoning agent will directly compute the results without programming. Extensive experiments on six graph analysis benchmarks demonstrate that GraphTeam achieves state-of-the-art performance with an average 25.85% improvement over the best baseline in terms of accuracy.

## 环境要求

---

- **操作系统:** 适用于 Windows、macOS 或 Linux
- **Conda:** 已安装
- **Docker:** 已安装并运行

# 安装步骤

---

## 1. 创建 Conda 虚拟环境

首先，使用 Conda 创建一个指定版本的 Python 虚拟环境。

```
conda create -n myenv python=3.10.14
```

激活虚拟环境：

```
conda activate myenv
```

## 2. 安装依赖

在虚拟环境激活的状态下，运行以下命令安装项目所需的依赖包：

```
pip install -r requirements.txt
```

## 3. 使用 Docker

Docker 用于在代码生成完成后执行代码。请按照以下步骤操作：

### 3.1 拉取指定的 Docker 镜像

```
docker pull chuqizhi72/execute_agent_environment:latest
```

### 3.2 创建名为 **test** 的容器

```
docker create --name test chuqizhi72/execute_agent_environment:latest
```

# 运行项目

---

## 1. 激活 Conda 环境

确保 Conda 虚拟环境已激活。如果尚未激活，请运行：

```
conda activate myenv
```

## 2. 启动 Docker 容器

确保 Docker 容器已启动。如果尚未启动，请运行：

```
docker start test  
docker exec -it test /bin/bash
```

## 2. 运行 `run.py`

在激活的虚拟环境中，导航到项目目录并运行 `run.py`：

```
python run.py
```

## 常见问题

---

### 问题 1: 运行 `run.py` 时出错

**解决方案:** 确保所有依赖已正确安装，并且虚拟环境和docker已激活。检查 `run.py` 中的路径和配置是否正确。