《诡秘之主》知识图谱

> **小组成员：**
>
> **李政　22321133**
>
> **周宇鑫 22321172**
>
> **赵晨希 22321146**
>
> **彭宇辰 12321063**

一、项目介绍

本项目构建了《诡秘之主》小说的知识图谱，数据主要来源于《诡秘之主》百度百科与《诡秘之主》小说原文。构建本图谱的工作主要分为以下几个部分：

1. 数据爬取与预处理
2. 关系抽取
3. 属性提取
4. 构建问答系统

二、数据爬取与预处理

2.1 百度百科数据爬取

我们针对网络小说《诡秘之主》进行了数据爬取工作，旨在构建一个包含人物、剧情等相关信息的训练集语料库。具体实现过程中，我们采用了requests_html库以实现对目标网页的访问，并利用该库获取网页的HTML内容。随后，我们运用了BeautifulSoup库对网页内容进行了深入的解析，以提取出所需的有效信息。

详细的爬取流程如下所述：首先，我们从百度百科针对《诡秘之主》小说的相关词条中提取出小说中出现的关键信息，如人物名称、序列名称以及势力名称等，并记录下这些词条的URL地址，以便于后续的网页访问操作。在获取到这些URL地址后，我们对每个URL进行了独立的访问，并从中提取出各个词条的详细信息。通常情况下，百度百科的词条内容包括简介部分和正文部分，我们需要对爬取到的数据进行适当的处理，以转化为结构化的数据格式，以便于后续的分析和应用。

爬取的数据如下所示：

```
# 苏茜（爱潜水的乌贼所著网络小说《诡秘之主》及其衍生作品中的女配角）

中文名：苏茜
外文名：Susie
别　　名：空想之狗
性　　别：女
登场作品：诡秘之主

## 角色形象

### 外貌

金毛大狗，脖子上悬挂着一副金边眼镜。帮奥黛丽出门传递情报或者帮奥黛丽携带神奇物品时身上会背着小包。

### 设定

在苏茜不知道怎么解释一件事，或者奥黛丽告诉她的事情无法被理解时，她会对奥黛丽说“毕竟我只是一条狗。”

## 角色经历

苏茜是一条金毛大狗，价值450镑，为霍尔伯爵购买几万镑猎犬的赠品，没有成为非凡生物前就能够自己打开房门，
也因此误服奥黛丽所调制的序列9观众魔药，成为非凡生物，并能够开口说话。后期因为奥黛丽的钞能力，随同奥黛
```

2.2 构建实体列表

我们针对网络小说《诡秘之主》的实体识别任务，采取了基于匹配的策略。鉴于网络小说中的实体名称通常与现实世界有较大的差异，同时百度百科提供了关于这些实体的全面介绍，且涉及的实体类型相对较少，我们决定利用这些已有资源来辅助实体识别。

具体方法是，首先从爬取的数据中提取出实体的名称和类型，以此构建一个实体列表。在此过程中，我们特别注意到在《诡秘之主》小说中，同一人物可能会有多个不同的称呼。例如，人物“克莱恩·莫雷蒂”可能会被称为"梅林·赫尔墨斯","愚者"或“周明瑞”。因此，在构建实体列表时，我们需要对这些不同的称呼进行统一处理。为了保证文本预处理的准确性，我们在处理原文文本时也会对这些不同的称呼进行替换和统一，以确保实体识别的一致性和准确性。

2.3 文本数据预处理

为了对网络小说《诡秘之主》进行有效的文本分析，我们首先对原文文本进行了预处理，旨在提高后续实体识别和信息提取的准确性。预处理的具体步骤包括：

1. 清理原文：我们首先移除了原文中与作者相关的内容，这些内容通常与小说的实际情节无关，可能会干扰后续的数据分析。

2. 名称统一：针对小说文本中人物的多种称呼，我们根据之前构建的实体列表，将不同的称呼统一为一个标准名称，以消除文本中的歧义。

3. 句子分割：我们按照标点符号（如句号、问号、感叹号等）对文本进行分句，从而将长文本切分为独立的句子单元，便于后续的处理。

4. 实体匹配：在分句后的文本中，我们根据实体列表进行匹配，仅保留包含两个及以上实体的句子。这些句子更有可能包含关键信息，对于构建知识图谱和实体关系网络具有重要价值。

5. 过滤短句：为了确保数据集的质量，我们去除了过于简短的句子，因为这些句子往往缺乏足够的信息量，不利于后续分析。

通过上述步骤，我们构建了一个经过预处理的原文数据集，为后续的实体识别和信息提取工作打下了坚实的基础。

数据集示例如下：

```
{
    "text": "奥黛丽·霍尔笑吟吟望向对面的“同伴”。阿尔杰·威尔逊微皱眉头，旋即舒展道：“倒吊人",
    "entities": [
        "奥黛丽·霍尔",
        "阿尔杰·威尔逊",
        "倒吊人"
    ],
    "types": [
        "人物",
        "人物",
        "序列"
    ]
},
```

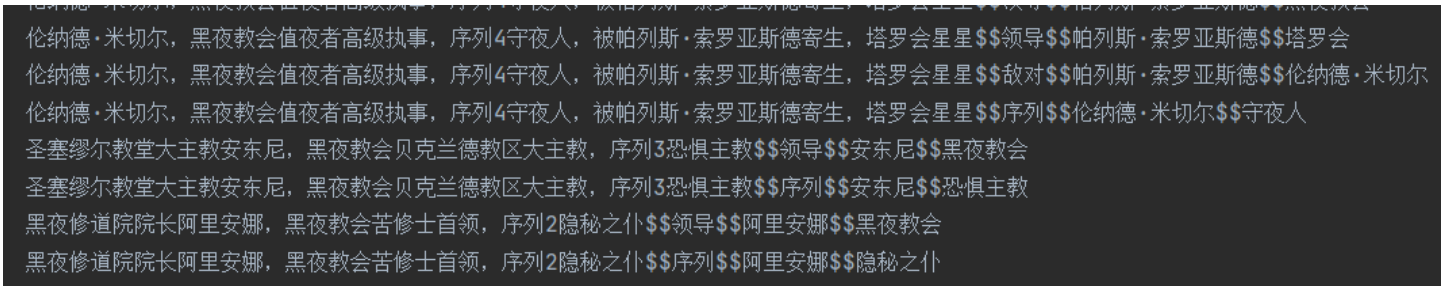
三、关系抽取

3.1 基于LLM的关系抽取

我们为人物、地区和序列间指定了如下关系：

```
relation_dict['人物-人物']=['爱慕','杀死','伙伴','敌对','老师','学生','家人','上级','下级']
relation_dict['人物-国家/势力']=['创建','领导','成员']
relation_dict['人物-序列']=['序列']
```

大语言模型有着强大的文本理解与处理能力。我们通过EasyInstruct这一大语言模型操作框架，使用大语言模型对小说数据集中的文本进行初步的关系标注。考虑到大语言模型的成本问题，我们使用了OpenAI的GPT-3.5 Turbo模型对小说原文数据集中5%的数据进行了关系的初步标注，结果展现为（原文，关系，头实体，尾实体）的四元组形式，示例如下：

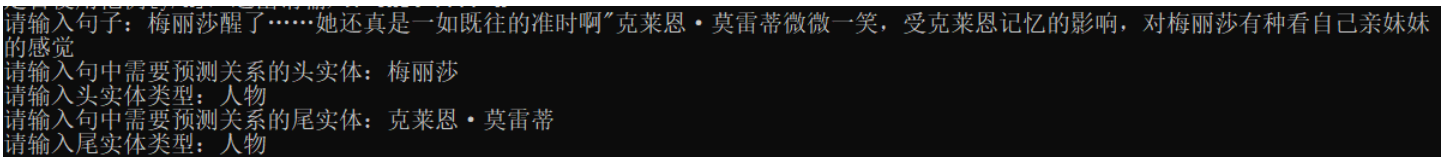


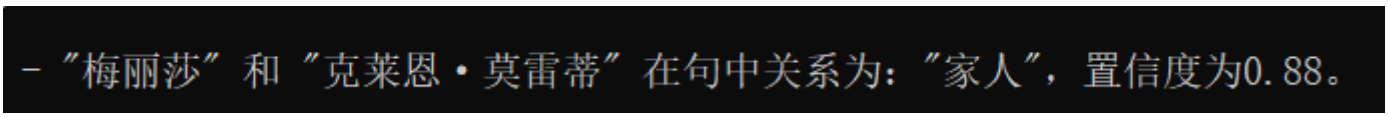
当然，由于提供的上下文有限，加上大语言模型存在的“幻觉”问题，其有的时候无法完全按照我们的指示回答问题，例如，返回一个不在限定关系中的结果，这个时候，就需要手动对生成的数据集进行数据清洗，剔除那些不符合要求的数据条目，以免影响后续的训练。

3.2 基于DeepKE的关系抽取

在使用LLM对数据进行初步矫正，以及人工的再次矫正后，我们就得到了一个可供DeepKE进行关系抽取训练的训练集。由于训练集规模较小（约1000条数据），我们使用Bert+LSTM网络模型，在预训练模型的基础上进行训练。由于使用的Bert模型最大支持512长度的输入，因此在训练之前，还需要对过长的文本进行剔除。

在Bert+LSTM模型上训练50个epoch后，得到最终的预测模型。预测效果如下所示：





在小说数据集上对所有实体对进行预测，剔除置信度过低的预测结果。预测数据存储为（关系，头实体，头实体类型，尾实体，尾实体类型）

```
伙伴$$克莱恩·莫雷蒂$$人物$$梅丽莎$$人物$$0.9988545179367065
伙伴$$梅丽莎$$人物$$克莱恩·莫雷蒂$$人物$$0.990852415561676
伙伴$$克莱恩·莫雷蒂$$人物$$梅丽莎$$人物$$0.9918385148048401
伙伴$$梅丽莎$$人物$$克莱恩·莫雷蒂$$人物$$0.9993785619735718
伙伴$$克莱恩·莫雷蒂$$人物$$梅丽莎$$人物$$0.9993185997009277
伙伴$$班森$$人物$$克莱恩·莫雷蒂$$人物$$0.9968462586402893
伙伴$$克莱恩·莫雷蒂$$人物$$班森$$人物$$0.998305082321167
伙伴$$蒸汽与机械之神$$人物$$让·马丹$$人物$$0.866363525390625
伙伴$$克莱恩·莫雷蒂$$人物$$让·马丹$$人物$$0.9986522793769836
伙伴$$让·马丹$$人物$$克莱恩·莫雷蒂$$人物$$0.9985748529434204
领导$$班森$$人物$$鲁恩王国$$国家/势力$$0.9564539194107056
伙伴$$克莱恩·莫雷蒂$$人物$$让·马丹$$人物$$0.9994906187057495
伙伴$$让·马丹$$人物$$克莱恩·莫雷蒂$$人物$$0.999488353729248
领导$$克莱恩·莫雷蒂$$人物$$鲁恩王国$$国家/势力$$0.9789272546768188
伙伴$$班森$$人物$$克莱恩·莫雷蒂$$人物$$0.988598108291626
伙伴$$克莱恩·莫雷蒂$$人物$$班森$$人物$$0.9944955706596375
上级$$班森$$人物$$克莱恩·莫雷蒂$$人物$$0.9623753428459167
上级$$克莱恩·莫雷蒂$$人物$$班森$$人物$$0.9083099961280823
领导$$克莱恩·莫雷蒂$$人物$$伦堡$$国家/势力$$0.9730715751647949
```

再次对数据进行手动清洗，并与之前标注过的训练集进行合并，即获得最终的预测数据。

四、属性提取

在属性提取部分，我们的目标是从网络小说《诡秘之主》中提取出人物和序列的相关属性。这些属性包括但不限于人物的别名、性别、晋升途径，以及国家的名称，首都等。为了实现这一目标，我们采取了以下步骤：

1. 结构化数据利用：我们首先从百度百科的简介部分提取出结构化的数据。由于百度百科的简介通常以表格形式呈现关键信息，这为我们提取属性提供了便利。

2. 异常属性筛选：在提取过程中，我们注意到一些属性可能是异常的，或者与小说的情节无关。为了确保数据的质量和相关性，我们对这些属性进行了筛选和剔除。

3. 无关属性过滤：除了异常属性，我们还需要去除那些与小说情节无关的属性，例如由小说改编的电视剧或其他衍生作品的信息。

4. 属性字典构建：经过筛选和过滤后，我们将提取出的属性以类似于字典的格式组织起来，每个属性作为一个键值对存在。

5. JSON文件写入：最后，我们将构建好的属性字典写入JSON文件中保存，以便于后续的数据分析和应用。

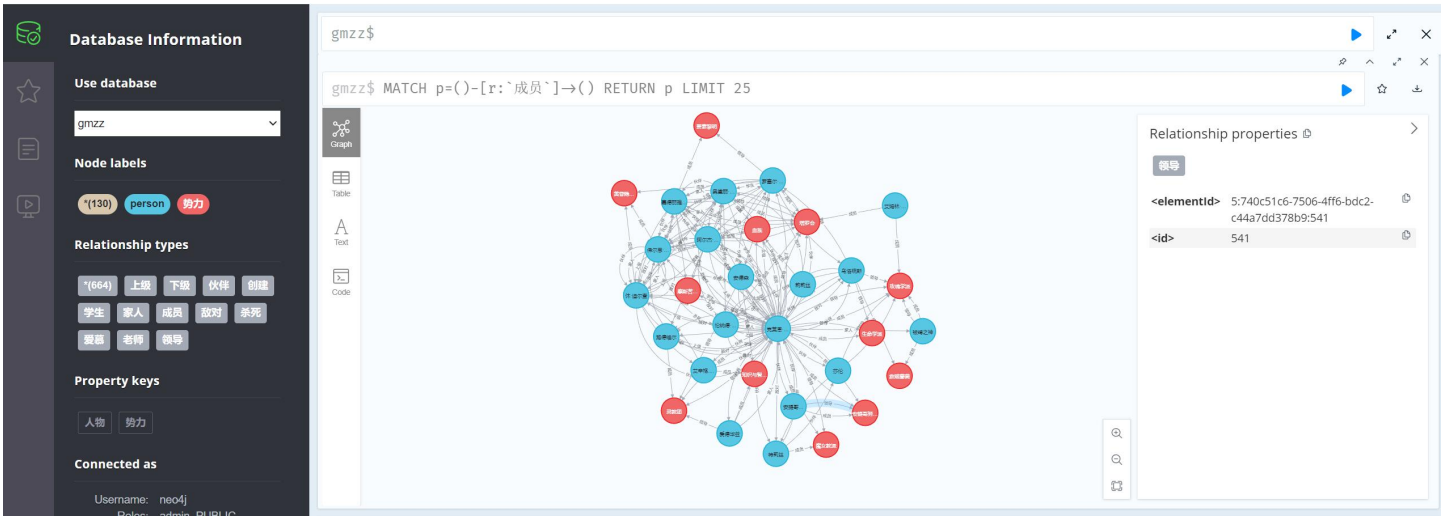
属性的内容和结构示例如下：

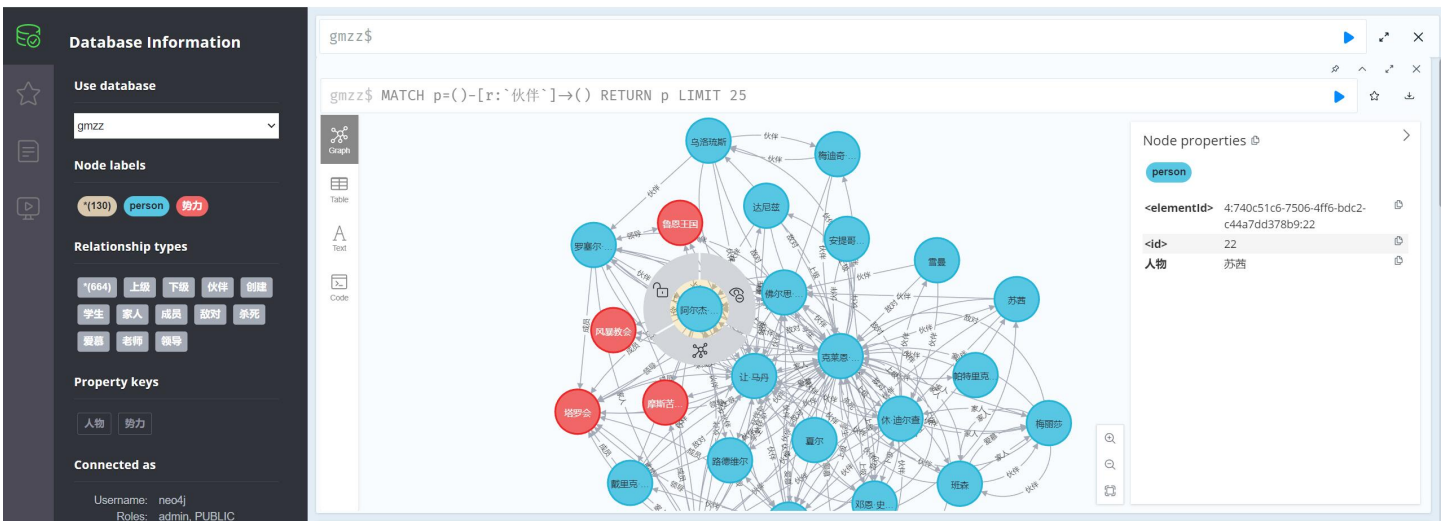
```
{
  "人物": [
    {
      "名称": "克莱恩·莫雷蒂",
      "别名": ["周明瑞", "愚者"],
      "性别": "男"
      "序列途径": "“愚者”途径"
    },
    {
      "名称": "阿尔杰·威尔逊",
      "别名": ["倒吊人","倒政委"],
      "性别": "男",
      "序列途径": "“暴君”途径"
    }
  ],
  "国家": [
    {
      "名称": "鲁恩",
      "首都": "贝克兰德",
      "官方语言": "鲁恩语",
      "主要宗教": "黑夜女神教会、风暴之主教会、蒸汽与机械之神教会、大地母神教会",
      "货币": "铜便士、银苏勒、金镑"
    }
  ]
}
```

以上就是我们在属性提取部分的工作内容和结果展示。通过这些步骤，我们能够有效地从原始文本中提取出有用的属性信息，并将其保存为结构化数据。

五、可视化和知识问答

将知识图谱导入neo4j进行可视化，部分图谱如下所示





与此同时，我们也实现了简单的问答系统

```
Q: 帕特里克·布雷恩是否是班森上级
A: 否
Q: 贝尔纳黛·古斯塔夫是否为要素黎明的成员
A: 是
Q: 奥黛丽·霍尔的伙伴是谁
A: [嘉德丽雅·古斯塔夫，海柔尔·马赫特，伦纳德·米切尔，让·马丹，戴里克·伯格，赫温·兰比斯，苏茜，休·迪尔查，佛尔思·沃尔，阿尔杰·威尔逊，克莱恩·莫雷蒂]
```