项目方案

项 目 名 称：	《水浒传》人物关系问答系统
应 用 领 域：	文学、问答系统
组　　长：	XXX
小 组 成 员：	YYY
填 报 日 期：	X 年 X 月 X 日

目 录

《水浒传》人物关系问答系统

摘 要

当前，如何让计算机理解人类的自然语言，并运用人类的自然语言模拟语言交际过程，实现“人机对话”，已经成为人工智能的一个重要研究领域——自然语言处理。问答系统是集自然语言、知识表示、信息检索于一体的研究课题，它建立在文本检索的基础上，但又不同于传统的搜索引擎。传统的搜索引擎要求用户输入一些关键字的组合，且对于用户提交的查询只能定位，用户必须依靠自己去筛选出需要的有用信息；而问答系统允许用户以自然语言的形式输入一个问句，最终返回给用户的也是自然语言形式的简短而准确的答案。目前，国外已有很多学者进行英文问答系统的研究，甚至已经有相对成熟的英文问答系统，但是国内有关中文问答系统的研究还不够多，因为中文问答系统对相关领域的研究要求更高，如：中文词语之间没有空格；汉语的句法分析和语义理解更为困难等，这些都造成了中文问答系统的发展缓慢。

本文是对中文问答系统的探索，针对文学作品人物关系复杂，无法进行快速准确查询的问题，本文提出基于水浒传的人物关系问答系统，并进行了实例验证，采用分词、句法分析等自然语言处理技术，研究了文学作品水浒传中人物关系，实现了根据用户输入的人物名称快速返回其人物关系的功能，系统包括三个主要部分：人物关系检索、人物关系全貌和人物关系问答。对于用户提交的问题，首先利用哈工大的语言技术处理平台 LTP 进行分词，提取关键词；其次，对于已经预处理的数据建立图数据库，然后用分词提取出来的关键字进行 Neo4j 图数据库的查询，匹配相关信息，利用 Python Flask 建立前端展示页面，建立知识图谱展示。最后，本文对自然语言处理的问答系统的实现和试验结果进行了评价，还对问答系统未来发展方向进行了展望。

关键字：问答系统，Flask，水浒传，LTP，Neo4j

1 项目概述

1.1 项目背景

问答系统作为人工智能的一个分支，已有了漫长的发展历史。20 世纪 60 年代发展的问答系统，允许用户以自然语言的方式查询数据库中存储的信息。该时期最成功受到人们关注的两个问答系统是 BASEBALL[1]和 LUNAR。到了 20 世纪 90 年代，该时期的问答系统已经可以弥补传统搜索引擎针对用户提问返回一系列相关网页链接的缺陷， 最为著名的问答系统是 Start 系统，该系统根据用户所查询的信息是否存在于已有数据库中设定了两种处理模式，即当用户查询的内容已经存在于知识库中的情况下，系统可以直接将对应的答案返回给用户；如果知识库中没有存储对应的信息，则通过搜索引擎检索并处理后反馈给用户[2]。随着神经网络技术、深度学习技术等的发展进步，问答系统进入了以知识和知识自动化为中心的新阶段。21 世纪初期诞生的 Watson 问答系统，由美国 IBM 公司研发，通过存储有关影视、新闻等多个领域的海量资料，实现在较短时间内针对用户提问返回相应答案，并由此在知识竞赛中打败人脑一举成名，现已在多个领域广泛使用[3]。

与国际上问答系统的发展相比较，我国对其的研究起步较晚。近年来随着科学技术的飞速发展，我国各大高校和研究所也开始对其展开了深入的研究，如复旦大学[20]和中科院都参加了 QA Track 的竞赛、上海交通大学开发的智能答疑系统以及中科院计算所研究[22]的知识问答系统等。2005 年百度公司推出百度知道，这是一个交互式的问答系统， 具体实现时主要是将用户查询的问题与数据库中已存在的问题进行比较，若相同则立即返回答案，若不同则根据相似度计算返回与之相似的若干问题及答案供用户参考。整体来看，中文的语法以及语义复杂性等多因素给研究带来了不少挑战，因此针对中文的语句相似度研究、文本理解等知识问答系统逐渐成为研究的热点，且有很大的发展空间。

1.2 项目意义

近年来网络小说读者越来越多，一大批网络文学作品层出不穷，类似的小说软件如书旗、番茄小说、微信读书等，包含了大量的文学作品，文学作品类型与题材丰富，然而，一部文学作品的字数通常是超过百万的，当读者阅读文学作品时，有时很难快速确定书中具体的人物关系，这就使得用户仅仅通过自己阅读是很难准确地捕捉到作品中具体的人物关系。如果使用传统的搜索引擎对文学作品中的人物关系进行查询，得到的结果往往都是相对应的大量文字片段的网页链接，无法得到简洁准确的答案[18]。由此，能够弥补上述缺陷的人物关系问答系统逐渐受到广泛关注，它不仅允许用户以自然语言的方式进行提问，还能够实现针对用户提问返回相应简洁准确答案句的功能，在一定程度上提高了用户的查询效率。针对文学作品人物关系复杂，无法进行快速准确查询的问题，本文提出基于水浒传的人物关系问答系统，并进行了实例验证，并采用分词、句法分析等自然语言处理技术，研究了文学作品水浒传中人物关系，实现了根据用户输入的人物名称快速返回其人物关系的功能。

1.3 应用场景展望

近年来网络文学蓬勃发展，文学作品的数量和题材层出不穷。一部文学作品的字数通常是超过百万的，这就使得用户仅仅通过自己阅读是很难准确地捕捉到作品中具体的人物关系。如果使用传统的搜索引擎对文学作品中的人物关系进行查询，得到的结果往往都是相对应的大量文字片段的网页链接，无法得到简洁准确的答案[18]。由此，能够弥补上述缺陷的问答系统逐渐受到广泛关注，它不仅允许用户以自然语言的方式进行提问，还能够实现针对用户提问返回相应简洁准确答案句的功能，在一定程度上提高了用户的查询效率。此外，关于中文问答系统的研究还不够成熟，中文语法及语义的复杂性给问答系统研究带来了不小挑战，因此，针对中文的语句相似度研究、文本检索、知识推理等问答系统的应用前景广阔，且有很大发展空间。

2 项目内容

2.1 项目总体功能框架

基于 neo4j 的水浒传人物关系的可视化和问答系统包括以下功能模块：人物关系查询、人物关系全貌查询以及人物关系问答。这三个模块分别实现了人物关系的可视化和人物关系的问答系统，并且以图数据库的形式展示出来，有利于人们更加直观、清晰的查看水浒传人物关系，项目具体功能框架如下图所示。

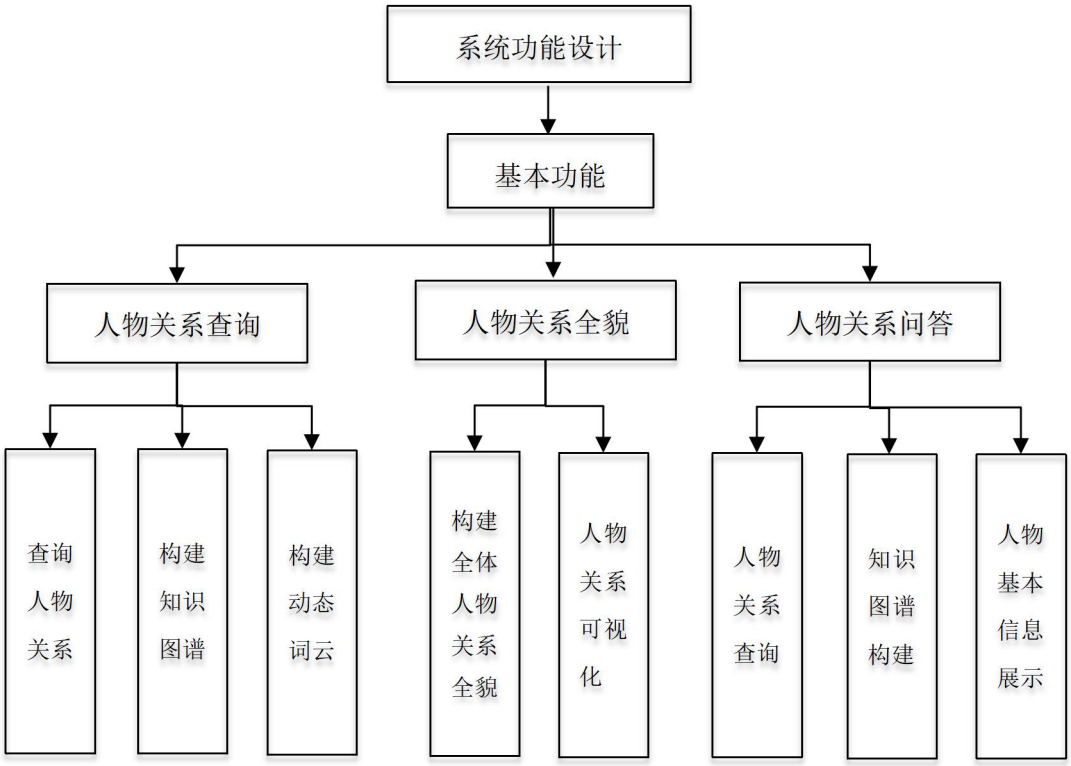

图 2.1 功能设计图

2.2 功能分析

2.2.1 界面设计

界面设计主要包括 index 主界面、all_relation 全部人物关系界面、search 查询界面和 KGQA 人物关系问答界面。主界面显示网站首页，点击开启探索可以进入查询界

面，查询界面由侧边栏、查询搜索框、动态词云、知识图谱展示界面这几部分构成；人物关系全貌界面由知识图谱及侧边栏构成，全貌占据整个页面，这种方式可以将显示范围最大化，视觉呈现更为舒适。KGQA 界面由搜索框，图数据展示和人物基本信息展示这几个部分组成，可以实现人物关系的简单查询。

2.2.2 数据库设计

在整体的系统架构处理流程中，考虑到字段属性和数据格式，综合分析实际需求，以及前后端交互存在的紧密联系，选择图数据库 neo4j 进行数据存储，Neo4j[15]是一个高性能的,NOSQL 图形数据库，它将结构化数据存储在网络上而不是表中。它是一个嵌入式的、基于磁盘的、具备完全的事务特性的 Java 持久化引擎，但是它将结构化数据存储在网络(从数学角度叫做图)上而不是表中。Neo4j 也可以被看作是一个高性能的图引擎，该引擎具有成熟数据库的所有特性。程序员工作在一个面向对象的、灵活的网络结构下而不是严格、静态的表中——但是他们可以享受到具备完全的事务特性、企业级的数据库的所有好处。

2.2.3 结构设计

水浒传人物关系问答系统实现对人物关系的查询，在技术设计层面，前端采用超文本标记语言 HTML 编写界面框架，使用预处理器 template 实现界面的样式设计，应用脚本语言 JavaScript 完成整个界面的动态呈现,通过 Flask 中 app 接口实现人物关系显示、人物关系问答交互等相关功能。平台层基于前后端分离的思想，后端使用 Werkzeug 工具箱开发的 Python Flask 框架搭建，访问端口 localhost:5000，结构设计如图所示。

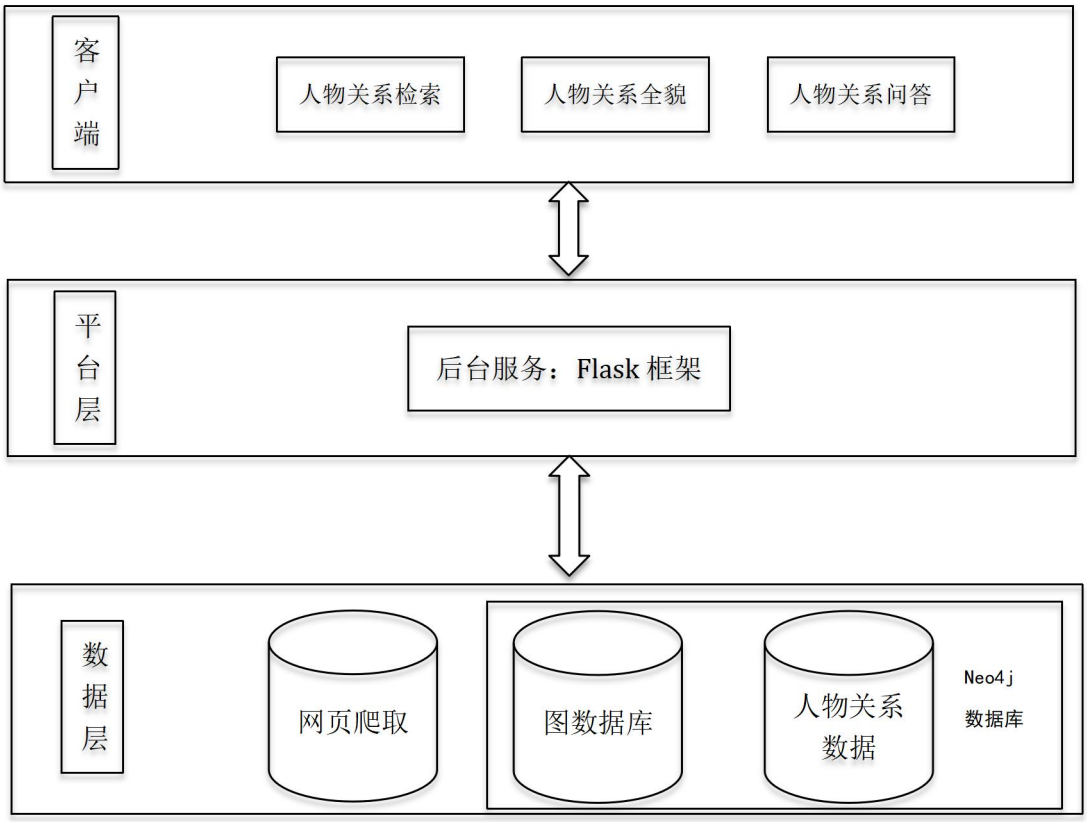
客户端
人物关系检索
人物关系全貌
人物关系问答
平台层
后台服务：Flask 框架
数据层
网页爬取
图数据库
人物关系数据
Neo4j
数据库

图 2.2 结构设计图

3 技术路线与方法

3.1 项目技术路线

该项目首先从互联网网页获取数据，进行文本预处理，将获取的数据整理成三元组数据，之后，将数据导入图数据库 Neo4j 构建知识图谱，构建知识图谱查询模块；其次，根据处理后的三元组数据进行网络爬虫，爬取相关人物信息（如中文名、别名、图像、所属地等），将爬取的图片保存为 Images，将爬取的人物信息保存为.json 文件；最后，利用 Flask 框架进行后台处理模块的构建以及前端页面的展示。其中，关于问答模块，使用了由哈工大研发的 LTP 语言技术处理平台。对输入的句子进行切分，从而查找人物关系信息。该项目的整体流程如下图所示：

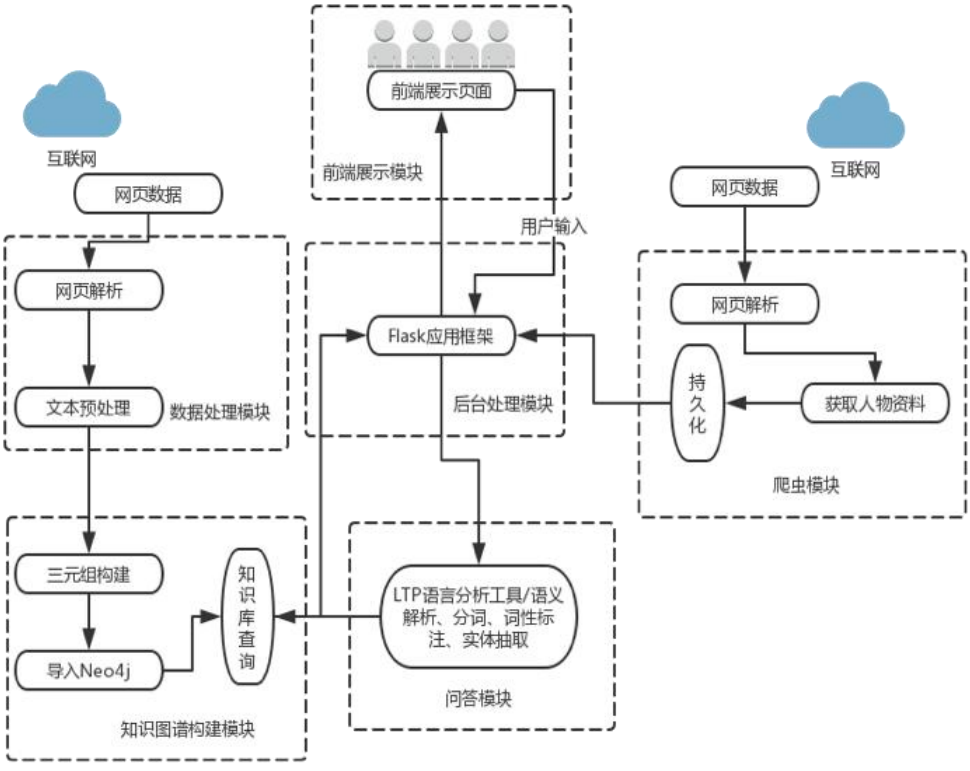

图 3.1 项目整体流程图

3.2 采用技术和方法

3.2.1 LTP 介绍

语言技术平台（Language Technology Platform，LTP）[14]提供了一系列中文自然语言处理工具，用户可以使用这些工具对于中文文本进行分词、词性标注、命名实体识别、依存句法分析、语义角色标注等等工作，该平台功能丰富、处理过程高效、精准。从应用角度来看，LTP 为用户提供了下列组件：

- 针对单一自然语言处理任务，生成统计机器学习模型的工具
- 针对单一自然语言处理任务，调用模型进行分析的编程接口
- 系统可调用的，用于中文语言处理的模型文件
- 针对单一自然语言处理任务，基于云端的编程接口

LTP 是哈工大社会计算与信息检索研究中心历时十年开发的一整套中文语言处理系统。LTP 制定了基于 XML 的语言处理结果表示，并在此基础上提供了一整套自底向上的丰富而且高效的中文语言处理模块 （包括词法、句法、语义等 6 项中文处理核心技术），以及基于动态链接库（Dynamic Link Library，DLL）的应用程序接口，可视化工具，并且能够以网络服务（Web Service）的形式进行使用。

LTP 系统框架图如下所示：

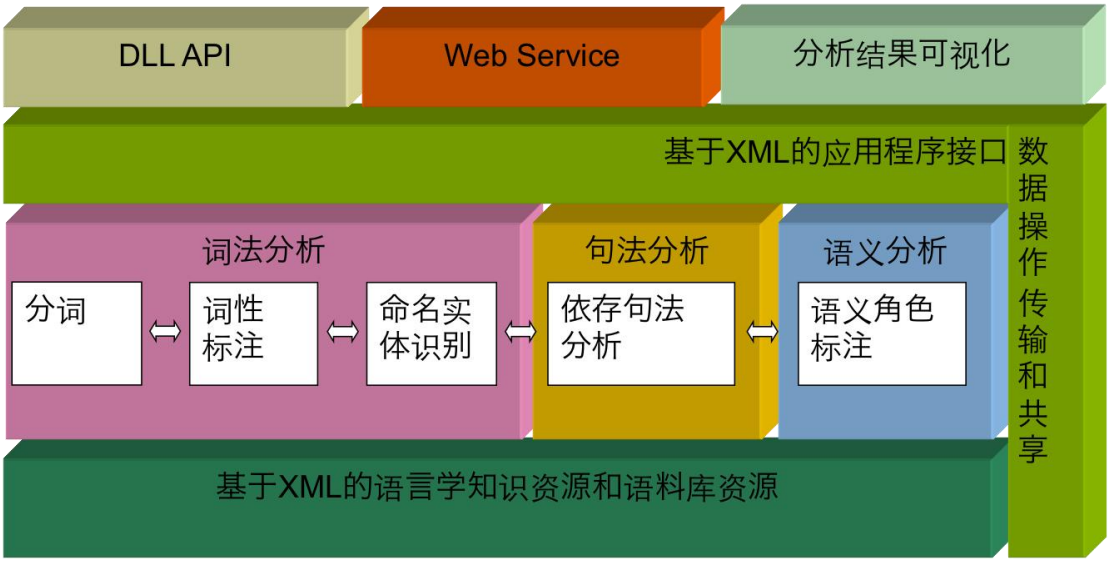

图 3.2 LTP 系统框架

LTP 提供了 6 项中文处理技术，由底层到高层依次为：词法分析（包括分词、词

性标注和命名实体识别)、句法分析（依存句法分析）和语义分析（词义消歧和语义角色标注），这些技术均在国际评测中取得优异成绩[2] 。

对于中文信息处理的各单项技术，目前主流的都是基于统计的方法，所采用算法、训练数据以及所选择的特征对于一个基于统计的自然语言处理系统都起到至关重要的作用，其中任何一项的改进，都会推动某项技术的进步。因此，对于 LTP 中的各项技术 ，我们都试图从算法、数据和特征等方面加以改进 ，在保证分析效率的前提下，有多项技术达到目前已知的最好水平。下面分别加以介绍。

1）分词（Word Segmentation）

中文分词将一个汉字序列切分成词的序列，是中文信息处理最基础的技术之一。其中歧义（包括组合型歧义和交集型歧义）和未登录词是困扰分词系统的主要问题。自 Nianwen Xue 首次提出将分词问题看作序列标注问题以来[3]，各种基于统计的序列标注模型，如条件随机域（Conditional Random Field，CRF）[4]等，便被应用于中文分词，其不但能够很好的解决分词歧义问题，而且能够解决部分未登录词问题，因此该方法成为目前的主流方法。LTP 也采用了基于 CRF 的分词方法。

2）词性标注（POS Tagging）

词性标注指对于句子中的每个词都指派一个合适的词性，如名词、动词、形容词等。词性标注是典型的序列标注问题，早期采用如隐马尔科夫模型[24]等生成模型（Generative Model）加以解决。然而，这类方法需要较强的独立假设，因此最终系统的准确率并不高。以最大熵马尔科夫模型（Maximum Entropy Markov Models， MEMM）为代表的判别模型（Discriminative Model）可以利用更丰富的特征，而且不需要假设这些特征是独立的，很好的解决了生成模型所面临的问题，使得词性标注准确率有了大幅度的提升。在 LTP 中，使用了准确率更高的支持向量机作为最基本的分类器，进一步提升了词性标注的准确率。与此同时，针对数据稀疏问题，特别是分词阶段识别的未登录词，我们首次引入了汉字特有的偏旁部首特征，进一步提高了词性标注泛化能力。

3）命名实体识别（NE， Named Entity Recognition）

命名实体识别是指文本中出现的专有名称和有意义的时间或数量短语，主要包括人名、地名、机构名、时间、数量等。NE 识别的任务就是将这些名称和短语识别出来并加以归类。目前主要有两类方法：基于规则的方法和基于统计的方法。对于规律性

比较强的命名实体，规则的编写高效而准确，如时间表达式等。而基于统计的方法常被应用于规律性不强的命名实体，如地名、机构名等。通常基于统计的命名实体识别被看作是序列标注问题，常用的机器学习算法包括隐马尔科夫模型，最大熵马尔科夫模型，条件随机场等。

LTP 采用了统计和规则相结合的方法，统计模型采用 MEMM，能够识别人名、地名、机构名、时间、日期、数量和专有名词 7 类实体。然而，该方法仍然依赖于大规模的训练语料，人工标注成本较高。为此，LTP 中加入了一种借助英文命名实体识别系统从双语料中自动生成大规模中文命名实体识别训练语料方法[5]，扩展了系统的覆盖范围，提高了识别能力。

4）词义消歧（Word Sense Disambiguation）

一词多义是自然语言固有的特征，也是语言应用中十分普遍的现象。汉语多义词（歧义词）在词典中只占总词语量的 10%左右，大约 8000 多个多义词。比例虽然低，但是歧义词多为常用词，在语言应用中出现的频率很高。根据对大规模语料库的统计数据发现，汉语歧义词在语料中出现的频率很高。根据对大规模语料库的统计数据发现，汉语歧义词在语料库中出现的频度达到 42%左右。如何确定歧义词的词义是进行自然语言处理领域不可回避的问题。基于统计的词义消歧技术是当前词义消歧研究领域的主流方法，但该方法需要有词义标记的训练语料，而获得规模足够大的高质量标注语料，需要代价高昂的人力物力，而且数据的一致性也很难保证。如果语料规模偏小，数据稀疏问题就会十分严重。所以目前学术界多是针对个别多义词，人工标注较多的样本进行词义消歧的实验。然而该方法很难应用于全部多义词的消歧。为了能够标注更大规模的语料库，LTP 提出了一种利用双验证码进行语料库标注的方法，该方法基于人体计算思维，巧妙的利用互联网背后用户的知识，在其自然使用网络的状态下，自动的获取词义消歧语料库。词义消歧的另外一个问题就是小概率词义的训练数据难以获得，多义词的词义分布很多情况下非常不均衡，为了解决这个问题，刘挺等人[25]提出了等价伪词的方法，解决了数据不均衡的问题。有了大规模词义消歧语料库，刘挺等人采用支持向量机作为分类器，基于多种特征，实现了词义消歧系统，并在 2007 年 SemEval Task 11 词义消歧评测任务中获得来第一名。

5）依存句法分析（Dependency Parser）

依存句法分析将句子由一个线性序列转化为一颗结构化的依存分析树，通过依存

弧上的关系标记反应句子中词汇之间的句法关系。与短语结构相比，依存结构具有形式简洁、易于标注、便于应用等优点，逐渐受到学术界和工业界的重视。目前主要有基于转移和基于图两种依存句法分析反法，其中基于图的方法由于进行的是全局最优解的查找，获得了更高的准确率，因此，在 LTP 中，也采用了基于图的方法，并且使用了高阶的特征，以获得更高的准确率。与通常的采用动态规划算法进行解码的句法分析器不同，LTP 采用了基于柱状搜索的解码算法，以及基于标点的两阶段句法分析方法，在不损失分析精度的情况下，较大的提高了句法分析的效率，使得句法分析能够满足一般的互联网信息处理应用对处理速度的需求。

6）语义角色标注（Semantic Role Labeling）

语义角色标注是目前浅层语义分析的一种主要实现方式，其具有问题定义清晰，便于人工标注和评测等优点。该方法不对整个句子进行详细的语义分析，而只是标注自然语言短语为给定谓语的语义角色，如施事、受事、时间、地点等。通常，人们将语义角色标注问题看成是分类问题。也就是说，可以使用各种分类算法逐一判断一个语言单元（词、短语或句法成分）是否是语义角色，然后预测其属于何种具体的语义角色。对于分类器输出的结果，还需要根据语义角色标注的多种约束条件进行一些后处理操作，形成最终的语义角色标注结果。数据稀疏仍然是困扰语义角色标注的主要问题之一，如何充分利用泛化能力更强的特征，是目前亟待解决的问题。基于 Kernel 方法是解决这一问题的较好途径，例如，对于句法特征较为稀疏的问题，可以使用 Convolution Tree Kernel，泛华路径、位置等特征。LTP 中的语义角色标注采用最大熵分类器识别谓词和语义角色，在解码阶段采用基于整数线性规划（ILP， Internet Linear Programming）的方法。该方法可以较为方便的融合多种语义角色标注所具有的约束信息，最终进一步提高了系统的精度。

最后，下表中给出了 LTP 中各项技术的具体性能指标。

表 3.1 LTP 各项技术性能指标

训练数据	测试数据	性能	效率
1998 年 1-5 月份《人民日报》	6 月《人民日报》	F 值 97.4%	185KB/s
1998 年 2-6 月份《人民日报》	1 月份《人民日报》	准确率为 97.80%，未登录词（未出现在训练树中的词）准确率 85.48%	56.3KB/s
1998 年 1 月份《人民日报》	6 月份前 10000 句	F 值 92.3%	14.4KB/s

表 3.1 LTP 各项技术性能指标(续)

训练数据	测试数据	性能	效率
哈工大社会计算与信息检索研究中心自行标注的依存树库 9000 句	1000 句	依存关系准确率（LAS）为 73.91%,依存弧准确率(UAS)为 78.23%	0.2KB/s
哈工大社会计算与信息检索研究中心标注的全文词义消歧语料库 9000 句	1000 句	多义词准确率 91.29%，全部词 94.34%	7.2KB/s
Chinese PropBank 2.0（22277 句）	2556 句	F 值 77.2%	1.3KB/s

3.2.2 Neo4j 介绍

在现代社会中，对信息的管理已经变得越来越重要，如交通信息、文献检索、金融信息等，都需要处理大量的数据。数据库技术已成为信息系统的核心和基础。在数据库技术发展过程中，出现过众多的数据模型，比较常用的有 3 种，分别为层次模型、图模型和关系模型。关系模型建立在严格的数学基础上，具有较高的数据独立性和安全性，使用简单。关系数据库是目前应用最为广泛的数据技术。但是随着数据规模的膨胀与数据复杂性的增加，关系模型已经无法满足领域需 要，以社交网络为例，采用关系数据库将导致数据冗余，并且不能适应社交数据的动态性，也不能很好地支持类似“好友的好友”这样的多层复杂查询。针对数据 间内在关系复杂且动态变化的问题，人们再次将目光转向图形数据库，图形数据库能够有效的存储、管理、更新数据及其内在关系，并能高效执行多层复杂操作。在实际应用中选择何种数据库与应用需求紧密相关，下面将以图形数据库 Ne04j[10]为例从产品成熟度、模型安 全、可扩展三个方面与关系数据库加以比较，并为实际应用中的数据库选择提供一些建议和帮助。

图形数据库[8]就是将数据存储在图(Graph)结构中。如图所示是一个简单的有向无环图。其中，节点表示一个实体。例如人或商品。边表示点与点之间的连接关系，可以是有方向和无向的。如用户 A 买了商品 B 表示 A→B；如果用户 A 与用户 C 相互都认识，这种关系就是双向的，表示为 A←→C。属性表示点和边所附带的属性。例如用户姓名、年龄等。需要注意的是每个点或边的属性是动态可变的。

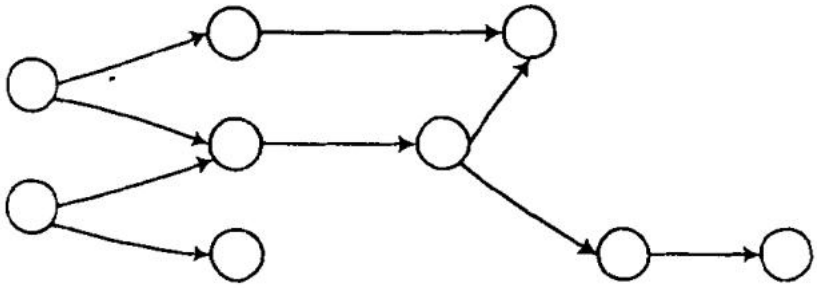

图 3.3 一个 DAG 例子

Neo4j 是一个高性能的,NOSQL 图形数据库，它将结构化数据存储在网络上而不是表中。它是一个嵌入式的、基于磁盘的、具备完全的事务特性的 Java 持久化引擎，但是它将结构化数据存储在网络(从数学角度叫做图)上而不是表中。Neo4j 也可以被看作是一个高性能的图引擎，该引擎具有成熟数据库的所有特性。程序员工作在一个面向对象的、灵活的网络结构下而不是严格、静态的表中——但是他们可以享受到具备完全的事务特性、企业级的数据库的所有好处。

一般的数据库系统主要涉及四种操作：增、删、查、 改(CRUD)，图的查找和搜索可以通过图的遍历和相关 图论算法完成，图数据库支持复杂的查询，而关系数据库则需要大量的连接操作(Join Operations)不仅费时且复杂，图数据库对于连接操作只需要一个起始节点，然后利用其免索引邻接(Index. Free Adjacency)特性实现无关于数据规模的查询性能，图数据库借助于灵活的图存 储结构特别适合路径查询和模式发现.对于根据历史数据预测未来趋势有巨大潜力.关系数据库利用结构化查询语言(SQL)进行查询，图数据库也有相应的查询语言如 Cypher、SPARQL 和基于路径的 Gremlin 等. 其中 Cypher 是一种声明式、类 SQL、灵活且表达力强的查询语言，且应用较为广泛。 总体来说，相比于关系数据库静态、刚性和不灵活的本质使得改变 Schemas 满足不断变化的业务模型非常困难，图数据库因为其无模式特点，使得它更能适应领域变化以及天生的可添加性，使得我们可以添加新的节点、属性、关系. 甚至子图而不影响现有业务逻辑因而扩展性高，对于具有复杂关联关系的数据处理也十分高效，利用图的多关系可以在语义上更直接表达多维时空数据，因此图数据库在社会学如社交网络、推荐系统、地理空间(通讯、物流、公路交通、路径选择等)、数据管理、 网络、授权管理等领域有着广泛应用。

3.2.3 Flask 介绍

Flask 是一个轻量级的可定制框架，使用 Python 语言编写，较其他同类型框架更为灵活、轻便、安全且容易上手。它可以很好地结合 MVC 模式进行开发，开发人员分工合作，小型团队在短时间内就可以完成功能丰富的中小型网站或 Web 服务的实现。另外，Flask 还有很强的定制性，用户可以根据自己的需求来添加相应的功能，在保持核心功能简单的同时实现功能的丰富与扩展，其强大的插件库可以让用户实现个性化的网站定制，开发出功能强大的网站。

Flask 是目前十分流行的 web 框架，采用 Python 编程语言来实现相关功能。它被称为微框架(microframework)，“微”并不是意味着把整个 Web 应用放入到一个 Python 文件，微框架中的“微”是指 Flask 旨在保持代码简洁且易于扩展，Flask 框架的主要特征是核心构成比较简单，但具有很强的扩展性和兼容性，程序员可以使用 Python 语言快速实现一个网站或 Web 服务。一般情况下，它不会指定数据库和模板引擎等对象，用户可以根据需要自己选择各种数据库[27]。Flask 自身不会提供表单验证功能，在项目实施过程中可以自由配置，从而为应用程序开发提供数据库抽象层基础组件，支持进行表单数据合法性验证、文件上传处理、用户身份认证和数据库集成等功能。Flask 主要包括 Werkzeug 和 Jinja2 两个核心函数库，它们分别负责业务处理和安全方面的功能，这些基础函数为 web 项目开发过程提供了丰富的基础组件。

Flask 的基本模式为在程序里将一个视图函数分配给一个 URL,每当用户访问这个 URL 时，系统就会执行给该 URL 分配好的视图函数，获取函数的返回值并将其显示到浏览器上，其工作过程见图。

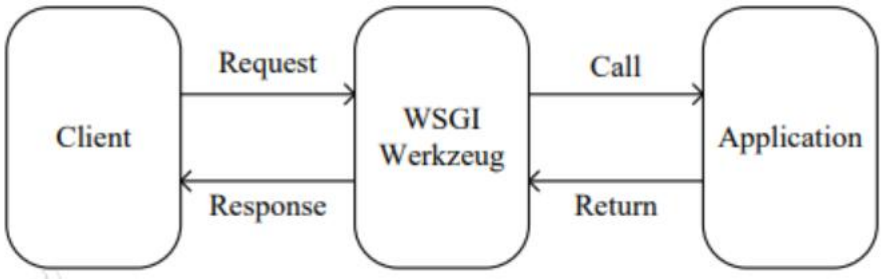

图 3.4 Flask 框架工作过程图

IT 运维的基本点为安全、稳定、高效，运维自动化的目的就是为了提高运维效率，Flask 开发快捷的特点正好符合运维的高效性需求。在项目迭代开发的过程中，所需

要实现的运维功能以及扩展会逐渐增多，针对这一特点更是需要使用易扩展的 Flask 框架。另外，由于每个公司对运维的需求不同，所要实现的功能也必须有针对性地来设计，Flask 可以很好地完成这个任务。

4 项目功能设计

4.1 功能介绍

该项目使用的系统环境如下：windows10，python3.6，编译工具为 PyCharm，图数据库 Neo4j 的版本为 neo4j-community-4.4.11，由哈工大研制的语言技术处理平台 LTP 版本为 ltp_data_v3.4.0，Flask 版本为 1.0，py2neo 版本为 2020.1.0，pyltp 版本为 0.2.1，bs4 版本为 0.0.0。人物关系数据是由人工合成的，人物图片信息 Image 和基本信息 data.json 是从网页上进行爬取的。

4.1.1 人物关系检索

首先是人物关系检索页面，输入水浒传人物名字（如武松）可以查询出与武松有关系的人物关系图谱，可以点击相关图谱查看具体关系，也可点击上方横条，筛选人物（“天罡”、“地煞”、“其他”等），每一种颜色表示一种所属关系，如林冲、宋江属于天罡。此外，也可点击右侧词云，即可出现相关任务关系图谱。

图 4.1 人物关系检索图

4.1.2 人物关系全貌

在左侧导航栏中，点击水浒传人物关系全貌，可以呈现出所有人物的关系图谱，如下图所示。

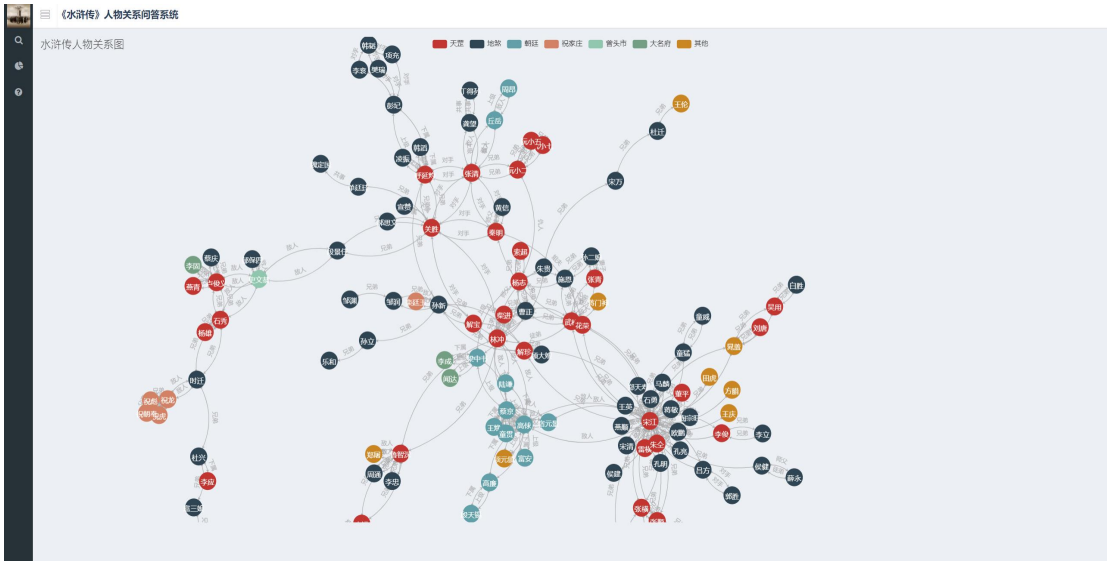

图 4.2 人物关系全貌图

4.1.3 人物关系问答

在左侧导航栏中，点击人物关系问答，可以进行特定人物关系的查询，如查询关胜的对手是谁，可以呈现出关胜的人物关系图谱。此外，点击相关人物节点，右侧边栏会呈现该人物的图片及相关信息，但是由于网络爬虫的人物图片不一定正确，或者爬虫时未找到相关人物图片，会导致问答系统不能正确处理信息，问答系统主要依赖于 LTP 当中的分词功能，当用户输入一个问题时，LTP 会将该句子进行切分，找出关键词以及需要查询的人物关系，再从图数据库 Neo4j 中，匹配相关的人物信息。下图所示是人物关系问答系统。

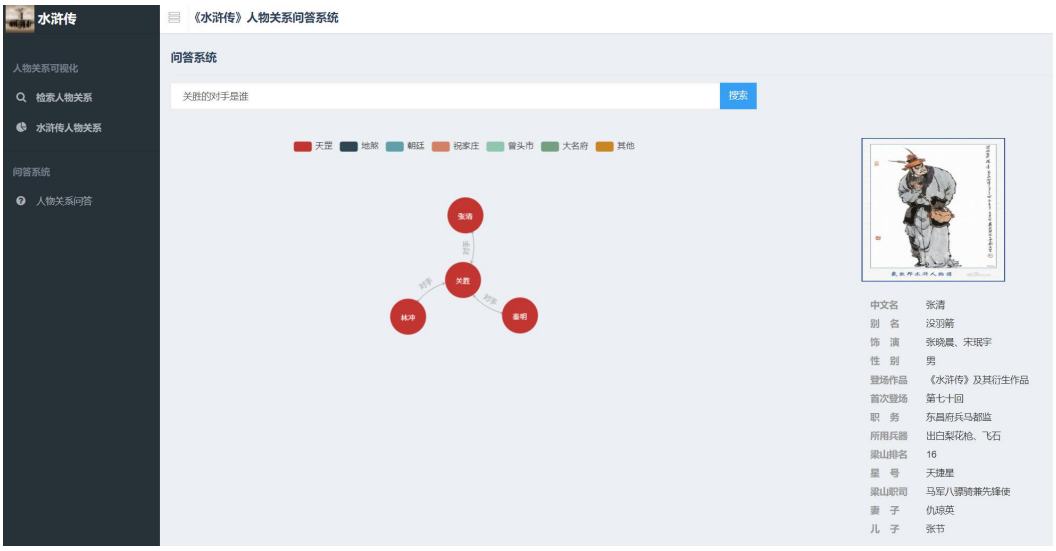

图 4.3 人物关系问答

4.2 使用步骤

该项目的具体安装部署如下步骤所示：

- 安装所需的库，执行 pip install -r requirement.txt
- 先下载好 neo4j 图数据库，并配好环境（注意 neo4j 需要 jdk11）。修改 neo_db 目录下的配置文件 config.py，设置图数据库的账号和密码
- 切换到 neo_db 目录下，执行 python create_graph.py 建立知识图谱
- 下载好 ltp 模型。
- 在 KGQA 目录下，修改 ltp.py 里的 ltp 模型文件的存放目录
- 运行 python app.py，浏览器打开 localhost:5000 即可查看

5 应用前景分析

目前的问答系统，还不能像人类一样能自如地回答用户提出的各种问题。问 答系统的思维和推理能力还比较欠缺，这是问答系统现在普遍存在的问题。尽管 问答系统现在离我们理想的目标还比较远，自动问答技术还处于刚刚起步阶段， 但是在最近几年，随着网络和信息技术的快速发展，同时人们想更快地获取信息 的愿望，都大大促进了问答系统的快速发展。 问答系统的研究与实现是建立在文本检索的基础上的，但是问答系统又不同于传统的搜索引擎，问答系统相当于一种智能的、新型的中文搜索引擎，因为传统的搜索引擎要求用户输入一些关键字的组合，而且其输出的是根据用户提交的查询字串搜索到的相关文献或网页，用户需要进一步筛选自己需要的信息；问答系统则允许用户以自然语言的形式对系统进行提问，其最终的目标是：准确地以简单的自然语言形式表达的句子或者以摘要的形式把正确的答案提交给用户。问答系统作为自然语言处理的一个重要的应用领域，是非常值得研究的。 虽然国外已经有很多研究机构和公司在英语问答系统的研究上取得了一些成果，但是由于问答系统是集自然语言处理、知识表示、人机交互、多媒体处理和智能 学习系统等多领域为一体的智能系统，它的发展将大大取决于这些领域的共同进步，所以目前对它的研究还十分有限。国内对中文问答系统的研究不是很成熟。 因为中文问答系统对相关领域的研究要求更高，例如：中文词语之间没有空格； 汉语的句法分析和语义理解更为困难等，这些都造成了中文问答系统的发展缓慢。 问答系统的实现主要有两个层次:一是依靠信息检索的相关技术来检索相关的文本段落，早期的问答系统主要是借助于简单的命名实体以及句子的匹配来进行答案的抽取的[22]；问答系统的另一个层次是较深层次的应用，也就是利用自然语言处理的技术，对问题和文本中的句子进行语法和语义上的分析，再结合信息检索等技术来进行问题答案的生成，这是目前大多数问答系统所采用的方法。本系统就是在第二个层次上的应用尝试。

本系统将自然语言处理技术[23]和图数据库链接起来，实现知识问答，运用自然语言的分词技术，进行信息检索，再以知识图谱的形式展示出来。这种图谱形式的问答查询是未来应用中不可缺少的部分。

6 创新点与局限性

目前，国内关于问答系统的研究还处于初始阶段，关于中文问答系统的研究还不够多，因为中文问答系统对相关领域的研究要求更高，中文语法及语义的复杂性给问答系统研究带来了不小挑战。在本文中，基于水浒传进行人物关系查询中，如何进行问句的分词是一个难点，如对问句“武松的好友是谁？”进行切分，需要分出关键词“武松”“好友”，对于用户输入问答不同的关键字，自然语言处理平台 LTP 有时识别不出人名或人物关系，无法正确切分句子。

本文主要是对中文问答系统的研究实践，采用中文自然语言处理平台 LTP、以及图数据库 Neo4j 实现人物关系问答，运用自然语言的分词、句法分析等技术，进行信息检索，再以知识图谱的形式展示出来。这种图谱形式的问答查询是未来应用中不可缺少的部分。

本项目的局限性如下：① relation.txt 中存储的是人物关系数据，人物关系数据的大小影响着问答系统最终展示给用户图谱的大小，人物关系数据越多，生成的图谱就越复杂，人物关系越少，生成的实体关系就越少。其次，该数据文件是由人工合成的，数据量较少，该项目的一个可拓展的方向就是如何从网页爬取人物关系数据，并且生成一个三元组文件。② 关于问答页面右侧的人物信息展示，是依照 relation.txt 的数据集中的第一列人名进行网络爬取，爬取对应人名的图片及简介，该做法会导致爬取的人物信息不是水浒传人物信息，导致结果不匹配，这是本项目的第二个局限性。 ③ 问答系统中，输入问句进行分词、句法分析会出现分词结果不准确，切分词错乱，导致无法在数据库中找到创建的图谱，也是本项目的一个局限性。

7 总结

针对文学作品人物关系复杂，无法进行快速准确查询的问题，本文提出基于水浒传的人物关系问答系统，并进行了实例验证。本文采用分词、句法分析等自然语言处理技术，研究了文学作品水浒传中人物关系，实现了根据用户输入的人物名称快速返回其人物关系的功能。

参考文献

[1] Minaee S， Liu Z. Automatic question-answering using a deep similarity neural network: IEEE, 10.1109/GlobalSIP.2017.8309095[P]. 2017.

[2] Pathak S， Mishra N. Context aware restricted tourism domain question answering system[C]// 2016 2nd International Conference on Next Generation Computing Technologies (NGCT). IEEE, 2016.

[3] Abdallah A， Kasem M， Hamada M, et al. Automated Question Answer medical model based on Deep Learning Technology:10.1145/3410352.3410744[P]. 2020.

[4] Rajpurkar P， Jia R， Liang P. Know What You Don't Know: Unanswerable Questions for SQuAD:, 10.18653/v1/P18-2124[P]. 2018.

[5] Nianwen Xue. Chinese Word Segmentation as Character Tagging[J]. International Journal of Computational Linguistics and Chinese Language Processing, 2003, 8(1):29-47.

[6] Lafferty J， Mccallum A， Pereira F. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data[J]. proceedings of icml, 2002.

[7] Ruiji, Fu, Bing, et al. Generating Chinese named entity data from parallel corpora[J]. Frontiers of Computer Science, 2017, 8(4):629-641.

[8] He W, Liu K, Liu J, et al. DuReader: a Chinese Machine Reading Comprehension Dataset from Real-world Applications[C]// Workshop on Machine Reading for Question Answering. 2017.

[9] Sutskever I， Vinyals O， Le Q V. Sequence to Sequence Learning with Neural Networks[J]. Advances in neural information processing systems, 2014.

[10]Chen Y C， Bansal M. Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting[J]. 2018.

[11]Luo D， Su J， Yu S. A BERT-based Approach with Relation-aware Attention for Knowledge Base Question Answering[C]// 2020 International Joint Conference on Neural Networks (IJCNN). IEEE, 2020.

[12]Bahdanau D， Cho K， Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. Computer Science, 2014.

[13]Wang Y， Zhang R， Xu C, et al. The APVA-TURBO Approach To Question Answering in Knowledge Base[C]// International Conference on Computational Linguistics. Association for Computational Linguistics, 2018.

[14]Ndayikengurukiye D , Mignotte M . Salient Object Detection by LTP Texture Characterization on Opposing Color Pairs under SLICO Superpixel Constraint[J]. 2022.

[15]Srk A , Dgc B , Hd C , et al. Method to transfer Chinese hamster ovary (CHO) batch shake flask experiments to large-scale, computer-controlled fed-batch bioreactors - ScienceDirect[J]. Methods in Enzymology, 2021.

[16]Hobson Lane, Cole Howard, Hannes Max Hapke et al. Natural Language Processing in action [M]. 北京, 人民邮电出版社. 2020.10.

[17]Partner J , Vukotic A , Watt N . Neo4j in Action[J]. Pearson Schweiz Ag, 2014.

[18]Mckinney W . Data Structures for Statistical Computing in Python[J]. proc.python sci.conf, 2010.

[19]梅家驹，竺一鸣，高蕴琦等. 同义词词林[M]. 上海，上海辞书出版社. 1983.

[20]王余蓝. 图形数据库 NEO4J 与关系据库的比较研究[J]. 现代电子技术,2012,35(20):77-79. DOI:10.3969/j.issn.1004-373X.2012.20.023.

[21]廖理.基于 Neo4j 图数据库的时空数据存储[J]. 信息安全与技术,2017,6(8):43-44,56. DOI:10.3969/j.issn.1674-9456.2015.08.015.

[22]王慧慧. 基于自然语言处理的问答系统研究[D]. 四川:电子科技大学,2016. DOI:10.7666/d.D308857.

[23]李思彤,冀美琪,夏欣雨,殷复莲.基于文学作品的人物关系问答系统设计与实现[J].软件,2019,40(9):139-143

[24]徐海洲. 自动问答系统中问句相似度计算方法研究[D]. 华东交通大学, 2014.

[25]黄萱菁. 复旦大学媒体计算与 Web 智能实验室信息检索和自然语言处理(IRNLP)组 [EB/OL]. http://www.yssnlp.com/ yssnlp2004/report/Fudan-Huangx-uanjing.pdf/.

[26]王颖. 基于自适应标签抽取的客服微博自动应答系统[D]. 北京邮电大学, 2014.

[27] 曹存根. NKI-21 世纪的科技热点[J]. 计算机世界报, 1998, 5(2): 1-3.

[28] 李正华. 汉语依存句法分析关键技术研究[D].哈尔滨工业大学,2018.

[29] 刘挺, 车万翔, 李正华. 语言技术平台[J]. 中文信息学报, 2011, 25(6):53-63.

[30]黄浩. 浅述利用 Python+Flask+ECharts 设计实现医疗数据可视化大屏展示[J]. 数字技术与应用, 2022(009):040.

[31]拉杰森 阿鲁姆甘, 拉贾林加帕 尚穆加马尼. Python 自然语言处理实战, 北京：人民邮电出版社， 2020. 10