课题名称：基于知识图谱的电影自动问答系统

小组名称：Beatles

小组成员：张礼明 谢梁杰 钟朋恒 潘旖茵

目 录

1 引言

1.1 系统简介

Beatles 电影知识自动问答系统，是面向大众的基于知识图谱的电影知识自动问答系统。该系统通过整合爬取的电影信息，构建基于电影领域的知识图谱，搭建基于安卓的自动问答平台，为用户提供准确、全面的电影知识输出。

1.2 系统研究背景与意义

随着结构化数据源的剧增，互联网正从仅包含网页和网页之间超链接的文档万维网向包含大量描述各种实体和实体之间丰富关系的数据万维网转变。在语义网的基础上，Google 于 2012 年率先提出了“知识图谱”的概念。知识图谱，旨在描述真实世界中存在的各种实体或概念。其中，每个实体或概念用一个全局唯一确定的 ID 来标识，称为它们的标识符。每个属性–值对用来刻画实体的内在特性，而关系用来连接两个实体，刻画它们之间的关联。从目前来看，知识图谱技术主要应用在搜索引擎和问答系统等方面。

随着知识图谱的提出，各大搜索引擎公司投入到知识图谱的构建中，具有代表的项目如：Google 的 Knowledge Graph，百度的知心和搜狗的知立方等。Google 将知识图谱技术应用于搜索引擎中，旨在将搜索结果进行智能化，达到任何一个关键词都能获得完整的知识体系，并从以下三个方面提高搜索质量：1.找到正确结果。由于一个关键词可能代表多重含义，所以知识图谱会将最全面的信息展现出来，让用户找到自己最想要的那种含义。2.最好的总结。有了知识图谱，Google 可以更加深入的理解用户所想要搜索的信息，并总结处理相关的内容和主题。3.更深、更广。由于“知识图谱”会给出搜索结果的完整知识体系，所以用户往往会发现很多不知道的东西（知识）。Google 提出的知识图谱本质上就是语义网

的进一步完善和发展，而为了提供更好的人机交互体验，基于知识图谱的自动问答系统为知识图谱的实现提供了一种新的思路。

基于知识图谱的自动问答系统能够针对更多复杂问题的句子进行回答，甚至可以进行一定的推力计算，并且由于是基于知识库的，所以系统具有良好的扩展性。现今，具有代表性的产品有：IBM 的 Waston、Google Now3 和 Siri4 等。随着知识图谱研究的不断推进，基于知识图谱的自动问答已经成为最为热门的研究领域之一。

目前，我国电影行业发展迅速，观影需求持续扩大。在过去，人们主要通过海报、预告片以及豆瓣平台等途径获取电影信息。通过这样的方式，用户在短时间得到的信息是十分有限和松散的，因此，对电影领域构建基于知识图谱的自动问答系统为突破解决这一问题指出了新的方向。基于电影领域的知识图谱，通过建立基于电影领域的知识库，以问答系统的形式，为用户提供更准确、更全面的电影信息，以期推动电影产业更好地发展。

1.3 系统开发的创新之处

（1）面向电影领域构建的知识图谱

通过爬取网上的电影数据，并对数据进行结构化处理，构建面向电影领域的知识图谱，能够更好、更准确、更全面地向用户展示电影信息。知识图谱将搜索结果进行智能化，达到任何一个关键词都能获得完整的电影知识体系。从目前来看，知识图谱技术主要面向开放领域开展研究，成效一般。因此，面向电影等受限领域构建知识图谱，将知识库范围大幅度减小，有利于提高知识搜索的准确性，就实际应用而言，更具研究价值。

（2）构建基于电影领域的问答系统

传统的知识搜索缺乏友好的人机交互接口，用户需输入问题，并通过不断修正问题和筛选结果，才能找到满意的答案。问答系统克服了这一单调繁琐的工作模式，在用户提问的基础上，系统通过对用户提问的引导以及上下文问题之间的联系，直接返回问题答案。

2 系统总体设计

2.1 系统框架

项目系统主要分为三大模块,分别是数据获取与存储模块、自动问答处理模块、人机交互模块。数据获取与存储模块主要用于获取和整合网页上的数据信息;自动问答处理模块主要用于处理自然语言问题并返回问题答案;人机交互模块主要用于与用户问题对话交流。具体系统模块如下：

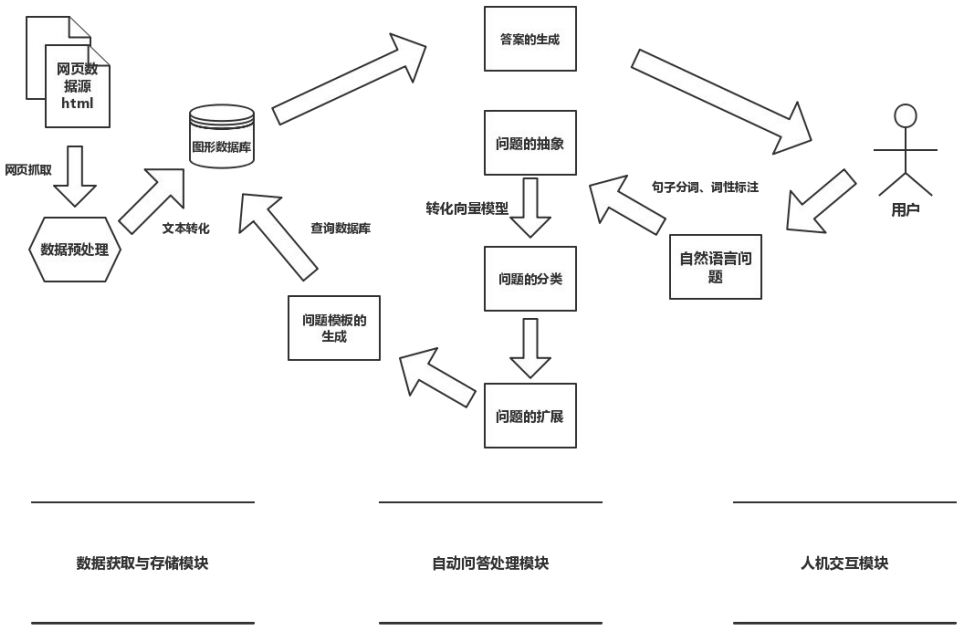


图 2-1 系统的结构模块图

2.2 数据获取与预处理

数据的获取主要锁定多个电影网站信息源，通过分布在不同电脑的爬虫文件，聚合了电影百度百科、豆瓣网、时光网、M1905、中国电影票房网等各大电影门户网站的电影信息。我们一共爬取了 997 部电影信息，爬取后利用正则表达式匹配得到相应的内容，其具体的网页链接如表 2-2-1 所示，爬取并预处理的结果如图 2-2-2 所示：

电影门户网站	网页链接
百度百科	http://baike.baidu.com/item/但丁密码
豆瓣电影	https://movie.douban.com/subject_search?search_text
时光网	http://service.channel.mtime.com/
M1905	http://www.1905.com/search/?q=但丁密码
中国电影票房网	http://www.cbooo.cn/

表 2-2-1 电影门户网站链接

电影:007：大破量子危机

基本介绍:
电影评分：6.4
电影票房：12075.1万
上映时间：2008-11-05(中国大陆)
国家地区： 英国 / 美国
片长：106 分钟
图片：007：大破量子危机/image/movieImg.jpg
关键词列表:
邦德 最终 维斯佩 组织 背叛

内容:
剧情解析:智破皇家赌场后痛失挚爱的詹姆斯·邦德（丹尼尔·克雷格饰），决定化悲愤为力量，全力追查真相，为韦斯帕报仇。他和上司M（朱迪·丹奇饰）收到情报，追查一个曾勒索韦斯帕的犯罪组织，并遇上了美艳的神秘女郎嘉美莲（欧嘉·柯瑞兰寇饰），在她的引领下调查冷血商人及犯罪组织首领多明尼克（马修·阿马立克饰）。与多明尼克有不同戴天之仇的嘉美莲，协助邦德共同展开报仇大计。多明尼克以环保组织作烟幕，电影剧照(18张)事实上操控着庞大的犯罪集团，正密谋在南美洲的玻利维亚发动政变，透过协助流亡独裁者麦德拉诺夺回政权，换取该国看似贫瘠但拥有世界上最重要资源的土地。邦德走遍奥地利、意大利及南美洲调查，同时逐步贴近韦斯帕之死的幕后黑手。在充满背叛、欺诈与杀机的谜局之下，邦德誓必要比中央情报局、恐怖份子，甚至M快一步，粉碎多明尼克的阴谋[2]；。
豆瓣评论url:https://movie.douban.com/subject/1946882/
影评评价:007：大破量子危机影片评价编辑配乐太糟 配角无彩《007：大破量子危机》的配乐十分糟糕，简直把影片变成了一部音乐喜剧；两位邦女郎奥尔加·库瑞兰柯和吉玛·阿特登的表演还算过得去，但没让观众留下深刻印象的地方。而法国男演员马修·阿马立克扮演的反派人物也没有精彩的过招。好在扮演“M夫人”的朱迪·丹奇的表演还是一如既往的老辣[17]；。（网易娱乐评）徒手杀敌 动作生猛在《007：大破量子危机》中，影片首映图片(15张)邦德除了一部功能强大的手机之外，邦德再无别的高科技武器装备，制服敌人主要靠双手。没有了高端武器，邦德只得空手对付敌人，真实而暴力的动作场面，得到了八成观众的欣赏，认为比较符合现动作片的潮流。“邦德”饰演者丹尼尔·克雷格比上一任邦德布鲁斯南更生猛、更有型，动作戏基本上是丹尼尔·克雷格亲身完成，而观众也是欣赏他西装上沾满血迹和尘土、刚刚打完一场搏斗的形象[18]；。（《广州日报》评）剧情精炼 缺少高潮片头长达8分钟的追车场面，保持了007系列电影紧张、刺激、惊险的所有元素，大场面的运用也表达出导演先发制人的控制欲，但是似曾相识的剪辑画面，一遍遍反复出现时“晕”自然产生。《007：大破量子危机》就是《007：皇家赌场》的续集，整部影片完全成了邦德的成长秀，一个失去心爱女人的男人的复仇史，目的极为明确，几乎找不到高潮点，连一点点悬念都没有。当然不得不提到邦德捉摸不定的内心戏，在强烈纠葛的内心戏搅和中，观众只好不停开小差，以思考

图 2-2-2 电影预处理的结果

2.3 数据存储及查询

Neo4j 是一个无框架数据库,它将数据作为顶点和边存储,适合知识图谱的存储结构。在开始添加数据之前,不需要定义表和关系。一个结点可以具有任何属性,任何结点都可以和其他结点建立关系。Neo4j 的查询语言 Cypher 是一种对图形声明查询的语言,使用图形模式匹配作为主要的机制来处理图形数据选择。Neo4j 提供 Java 版的基本操作 API 接口，方便融合到整个系统当中。Neo4j 的可视化展现形式如图 2-3-1 所示：

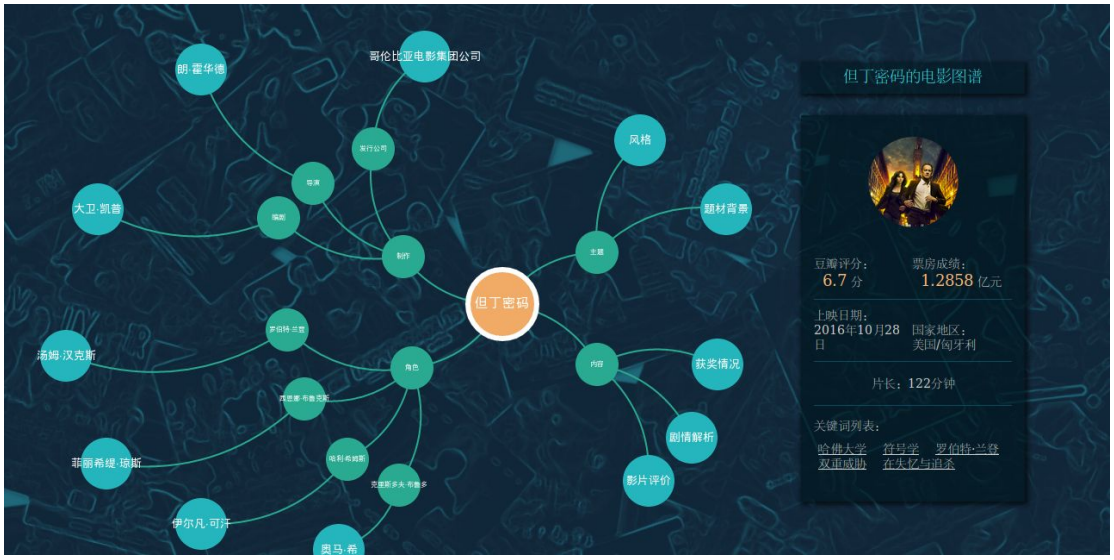


图 2-3-1 知识图谱的可视化展示

我们主要构建以电影为核心的知识图谱，以电影名称为根节点，以此延伸出电影的主题、内容、制作和角色，每一级节点又可以延伸至下一级的节点。图谱的深度为 3。根节点和叶子节点都有他的知识卡片。我们看到电影的根节点含有电影的摘要信息,例如豆瓣评分、票房成绩等等.另外，出品公司、导演都有他们各自的知识卡片。图 2-3-2 为知识图谱的主要结构图：

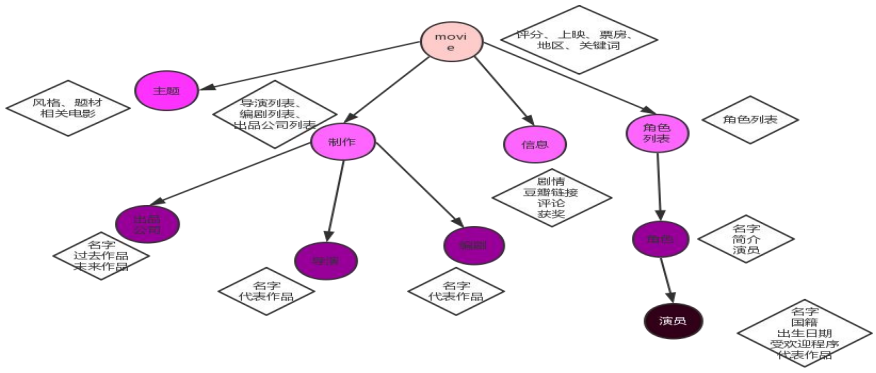


图 2-3-2 知识图谱的结构图

2.4 系统搭建及表现形式

自动问答系统的搭建主要分为服务器端与客户端。服务器端主要采用 Java+Tomcat。Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试 JSP 程序的首选。而客户端的表现形式主要采用移动安卓 app。项目的开发环境为基于 Linux 的 Ubuntu14.04 系统，开发工具为 Eclipse，开发语言则选用 Java。在开发过程中，应用到的其他框架有 Spark，Neo4j 等。

3 关键技术及算法设计

3.1 自动问答流程说明

自动问答的主要流程按问题的抽象与分类、问题的扩展与抽取、答案的生成展开。问题的抽象与分类主要用于确立问题的意图，问题的扩展与抽取主要是将问题扩展并转化为标准模板，以便抽取。答案的生成主要是将抽取的实体构建实体链，然后访问图形数据库，以求答案。

3.2 问题的抽象与分类

3.2.1 命名实体词典的构建

在自动问答的研究中，实体识别可以用于从用户的不同表达中找到用户关心的实体词，而实体词本身又包括大量的属性和语义信息，所以实体识别是用户问题处理的基础工作。一般意义的实体识别主要包括命名实体识别（Named Entity Recognition，简称 NER），又名叫“专名识别”，是指从文本识别具有特别意义的实体，主要包括人名、地名、机构名、专有名词等。目前有许多开源的中文命名实体工具包可以免费使用，但其巨大的局限性在于对专有领域的命名实体识别效果不理想，无法识别出如“我不是潘金莲”的专有电影术语。因此，在本次项目中，我们主要采以 Hanlp 提供的通用型命名实体工具包，并添加部分人工标注的命名实体，其添加的细则如表 3-1 所示、具体标注内容如图 3-2 所示：

命名实体类型	自定义词性
电影名称	nm
电影角色	nnt
电影演员	nr
电影出品公司	nis
电影工作人员	nnd
其他专有名词	nz

表 3-1 电影命名实体细则

不爱不散 nm 100
诡爱 nm 100
就是闹着玩的 nm 100
狂暴飞车 nm 100
国家要案 nm 100
麦兜当当伴我心 nm 100
天地逃生 nm 100
换爱七日 nm 100
痞子英雄之全面开战 nm 100

图 3-2 部分的标注内容

3.2.2 问题的抽象

问题的抽象主要是为问题的分类做前期的预处理工作。由于用户可能的问题会涉及到不同的电影名称、电影角色等，为了方便问题的分类，需要将特定的电影名称抽象到统一的概念，以下面的例子为例：

源问题：但丁密码中饰演罗伯特兰登的演员是谁?

抽象问题：nm 中饰演 nnt 的演员是谁?

如上面的例子，问题中涉及到专有的电影名称会转化为他的词性 nm。这样做的好处在于能让分类器减轻特征的选取工作量，也可以缩减训练集的规模。具体的转换如表 3-3 所示：

转化规则	源问题	抽象问题
电影名称--nm	但丁密码的风格是什么	nm 的风格是什么?
角色名称--nnt	但丁密码中饰演罗伯特兰登的演员是谁	nm 中饰演 nnt 的演员是谁?

表 3-3 问题的抽象规则

3.2.3 问题的分类

按知识图谱的内容，我们将问题分成以下不同的类型电影的基本属性、主题、制作、内容、角色等，具体的内容分类如表 3-4 所示：

问题类型	疑问词短语	例子
豆瓣评分	多少	电影的豆瓣评分是多少？ mαq
票房成绩	多少	电影的票房成绩是多少? 电影的票房成绩怎么样?
片长	多久/多长时间	电影的片长是多久? 电影的片长是多长时间?
国家地区	哪个国家/哪个地区/哪一个国家/哪一个地区	电影是哪个国家拍摄的?
上映时间	什么时候/什么时间	电影的上映时间是什么时候? 电影是什么时候上映?

表 3-4 问题分类的示例

对于电影的分类，我们主要利用了 spark 构建了贝叶斯分类器，通过人工标注的方法形成少量的训练集，图 3-5 为分类模型构建的流程图;

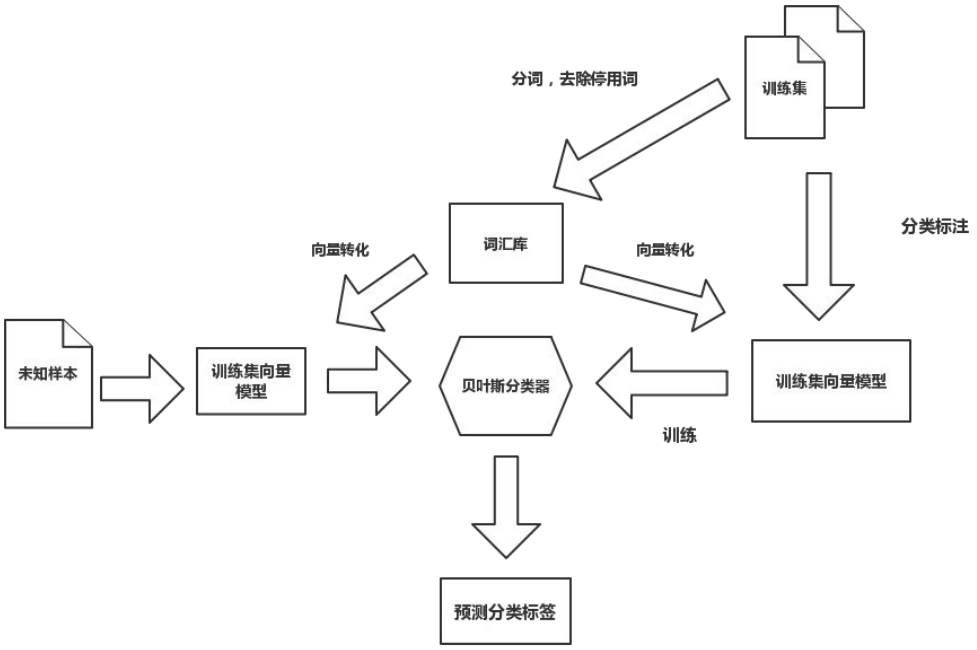


图 3-5 分类器模型图

3.3 问题的扩展与抽取

自然语言问句可能的表述有简单也有复杂的，但总体上都是描述主语与宾语的关系，而图是一个能够通过边来描述结点与结点之间关系的模型。语序图是一个有向图，是一个通过谓语作为连接、由主语指向宾语的有向图。将每个主语或宾语看作是一个实体，将谓语看作是属性关系。以下面的问句为例：

问句：但丁密码中饰演罗伯特兰登的人是谁?

转化为对应的语序图如下：

由于自然语言中经常出现表达不完整的情况，为了对问题的扩张而抽取出完整的语序图，我们构建了一系列的问题模板，通过分类器的标注，确定问题的意图，并映射对应的问题模板，在模板中形成对应的语序图。问题模板如表 3-6 所示：

问题的类型	问题模板
电影票房	nm 票房
电影风格	nm 主题 风格
电影角色	nm 角色列表 角色 nnt 简介
电影导演	nm 制作 导演列表

表 3-6 问题模板示例

3.4 答案的生成

答案生成的过程主要是将语序图转化为 Neo4j 的查询语句，然后在数据库中进行查询。但由于查询语言中键值查询是准确匹配的，而用户输入往往是模糊查询的。因此，本系统主要利用计算查询词与检索词之间的莱文斯坦相似度来完成链接对齐的工作。

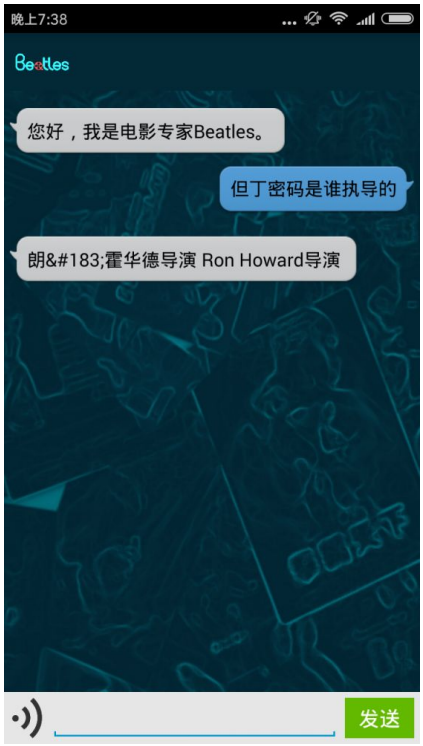
莱文斯坦距离(LD)用于衡量两个字符串之间的相似度。以下我们称这两个字符串分别为 s(原字符串) 和 t(目标字符串)。莱文斯坦距离被定义为"将字符串 s 变换为字符串 t 所需的删除、插入、替换操作的次数"。莱文斯坦距离越大，字符串的相似程度越低。

4 系统用例说明

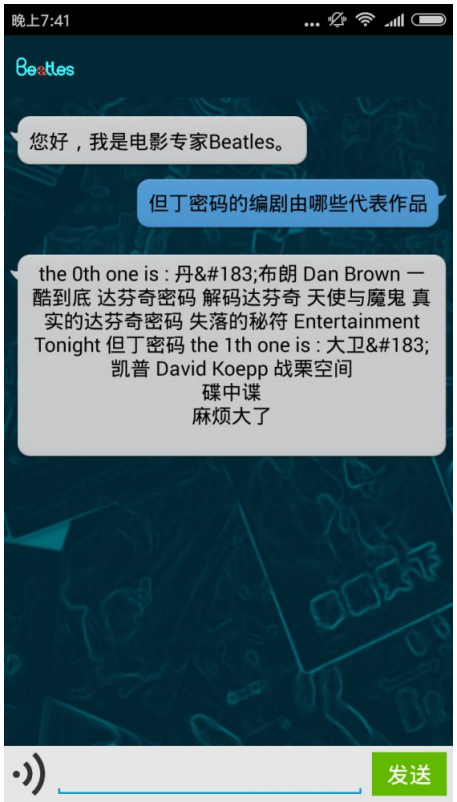
步骤一：点击 app 进入系统，首页是我们的 logo — Beatles

步骤二：进入系统，主页如下：

步骤三：选择手动输入或语音输入问题，效果如图：



步骤四：系统也能处理具有一定逻辑推理性的问题。如图：



步骤五：如果系统无法对用户提问给出合理答案，则返回“God knows”。

4 参考文献

[1]贺樑.基于中文知识图谱的电商领域问答算法设计与系统实现[D].华东师范大学.2016.
[2]朱敏.面向多领域大规模知识库的自然语言自动问答研究[D].西南交通大学.2015.
[4]宗成庆.统计自然语言处理[M].清华大学出版社.2008
[3]Holden Karau,Andy Konwinski[著],王道远[译].Spark 快速大数据分析[M].人民邮电出版社.2015

6 项目总结

6.1 项目的不足之处

1.用户的提问语句格式要求相对苛刻

由于受到技术、时间以及成本的限制，Beatles 电影自动问答系统能够处理的问题输入是相对有限的，对于符合输入规范的问题，系统能够提供相对满意的回复，否则，可能会产生错误的答案。

2.对新的电影知识没有有效的处理方案。

目前，Beatles 电影自动问答系统仅面向已知电影信息构建知识图谱，而对于未来产生的电影信息并没有有效的处理方案。电影信息每时每刻都在不断变化，随着时间的推移，系统必将无法很好地满足用户的搜索需求。对于新的电影知识，可以通过分析用户搜索日志，识别出新的实体以及联系，以完善现有的电影知识库，提高问答系统的服务质量。

3.将用户的提问孤立开。

目前，Beatles 电影知识问答系统仅针对用户当前问题产生答案。然而，在很多情况下，用户提问的问题是相互联系，承上启下的。因此，问答系统不应该将问题孤立开，相反，可以通过联系上下文的问题以及分析其中的内在联系，以更好地理解当前用户的提问意图，从而提高问答系统的准确性。

6.2 小组成员的感悟

张礼明：

这次大作业给我的感觉就是做了好久好久的样子，犹如经历一场漫长的战役。如今总算画上一个句号，做出来的效果也差强人意吧。总的来说，NLP 这个领域是十分的广阔且有趣，能给予我们的大作业无限的遐想。不过倘若没有足够的执行力坚持下去，即使想象再好，也终究是一个纸上谈兵的空话。在课程开始的时候，我便决定要让大作业往知识图谱的方向靠拢。正因为比较早确定方向，所以我有足够的时间去学习相应的内容，并反复地构思和磨合。不过虽说如此，消化知识的过程总是相当的缓慢，期间我也遇到不少的麻烦，尤其在思路的构建上。由于 NLP 的学习还未算深入，许多算法的可行性校验是一件非常痛苦的过程，例如专有名词的识别，是构建自定义词典库呢，还是自行训练命名实体识别模型。又如前期觉得依存句法对句子分析的逻辑性较强，然而应用在实际中却不见其然。这些都需要不断的学习与反思，从而总结出经验。

再说项目的细节上，我主要负责统筹项目的进展，算法模型的构建，以及知识图谱的存储。在实现项目功能的同时，我也锻炼了自己的思维方式和掌握不少的技能，例如学会使用 neo4j，深入 spark 的框架等等。很感谢这次大作业对我的训练，赐予了我一段宝贵的学习经验。

小组成员打分：

谢梁杰	钟朋恒	潘祷茵
97	98	95

谢梁杰：

这次大作业，我主要负责爬虫工作。由于有之前在实验室的项目经验，爬虫工作进行得相对顺利。虽然对我来说，这次大作业加深了我对自然语言处理的理论知识的理解，并将学习到如何将自然语言知识的理论与实际应用相结合。同时，我还接触到了知识图谱和问答系统领域的相关基础概念，了解到其工作原理。一开始，我完全处于懵逼状态。面对知识图谱一堆陌生的名词，我对项目处于抗拒的状态。经过不断的查找资料，最终慢慢地区分本体，知识库，语义网，语义网络等概念以及简单了解到知识图谱的工作流程，我开始对知识图谱产生了好奇兴趣。另外就是问答系统，看起来简单的功能却包含着大学问。如果把整个系统比作一个机器，知识图谱则好比机器运转的燃料，而问答系统则点燃燃料，使整个机器运转。问答系统的研究已经有了漫长的历史，现如今，知识图谱的火爆，为问答系统的研究带来了新的思路。我很希望在技术，时间和成本允许的条件下，对这两个领域能够进行更深的了解。以前，我感觉这些高深莫测的算法与实际应用风马牛不相及。在本次大作业中，我们开发了饿面向电影领域的自动问答系统，让我明白了理论贵在实践，以及学习到一些如何将理论与实践相结合的方式。同时，很感谢组长在大作业过程中对我的帮助和指点以及老师这一学期关于自然语言处理的精彩授课。

小组成员打分：

张礼明	钟朋恒	潘旖茵
98	96	96

钟朋恒：

本次大作业我主要负责安卓客户端和后端系统的搭建工作，理解了基于电影领域的自动问答系统的整个工作流程。通过自然语言处理课程的学习、实践与应用，结合知识图谱的工作原理，我认识到即使让计算机理解一个简单的问题，在背后也需要做大量的工作。对于这个基于电影领域的自动问答系统来说，从输入一个问题到产出答案，这就需要经历问题分类、实体抽取、实体关系抽取、链接对齐、构建问题模版、查询数据库等一系列的过程，在这过程中，涉及 libsvm 分类器预测、词典比较、构建依存关系树、构建同义词典、计算词语相关性等知识点，从输入到输出，这背后的实现实属不易。由于这个系统是基于一定领域的自动问答系统，而且在电影领域中，由于样本数量有限等因素，这个系统能处理的问题十分受限。而且在搭建整个系统的过程中，因为之前不怎么了解服务端系统的搭建，在搭建服务端的过程中也遇到了不少的问题，由此导致出现的一系列问题也十分具有挑战性。虽然困难，但在这过程中所学到的知识却是十分宝贵的，让我明白到自然语言处理技术的魔力所在，也明白到一个自动问答系统实现的困难，即使是一个简单的问题，也需要我们在背后做很多工作，才能让计算机很好地理解这个问题的意思，并返回对应的答案。经历了这次大作业之后，相信自然语言处理的知识会对以后的工作和学习增益不少。

小组成员打分：

张礼明	谢梁杰	潘旖茵
98	97	95

潘祷茵：

几经波折，大作业从初具规模到成熟竣工，我们组大神们功不可没。在大作业里面，我负责的部分是问题分布采集，模板抽取提炼，以及词典的编写分配。一部电影，因观众的兴趣广泛，所以涉及的问题也不少，但是依旧可以分类。举个例子，“nm 的编剧是谁”，“nm 是谁写的”，“谁写了 nm”，“谁是 nm 的编剧”都可以归类为同一个问“编剧”的问题。通过自然语言处理的学习，我收获良多。不仅仅是大作业的共同协作的过程，更多的是跟随老师的步伐在知识的海洋里面遨游的愉悦，以及日常学习专研的快感。自然语言处理虽然平时只是一个星期一个上午，但是平时作业时时有，能让人在温故而知新，不容易忘怀专业知识。我自己修了双学位，课程也是一个星期某几个晚上，发现如果自己没有提前去看，课后不复习，往往下一堂课就有点忘怀上一节课的内容。所以自然语言处理的学习的方法我也运用到了其他课程里面，使得大有裨益。

这个学期以来非常感谢老师的谆谆善诱，感恩团队大神的指导。大四的学生，忙实习，忙工作，忙考研，或者忙考公务员，大家的方向好，目标和重点都和以前都不太一样了，但是学无止境，学习总归是正确的，在学习上，时常遇到困惑，课堂听讲解决，或者课后麻烦小组大神们，总感觉自己问题不少，但是大家都不厌其烦，无限感恩。

小组成员打分：

张礼明	谢梁杰	钟朋恒
98	97	97