火影忍者知识图谱

马标、赖丹玲、邓仪
指导老师：李直旭

苏州大学 计算机科学与技术学院

目录

1 摘要

针对于知识图谱构建这一任务，火影忍者的人物关系层次适中，较为符合作业要求，且火影忍者数据繁多，各大垂域网站皆有详细资料。因此，我们组选择火影忍者作为图谱主体。

本文的主要任务是将火影忍者人物归属关系做了细化分类，并尝试使用自动化的方法来实现知识图谱的构建。文章具体介绍文本数据细节，数据爬取方式，图谱构建与实体载入，并附上实例代码。

2 分工介绍

1818401014 赖丹玲

主要负责：实体的属性载入，问答库的爬取、清洗与整理

1809401010 马标

主要负责：原始实体数据的获取与补全，protege 的初步界面化

1827405085 邓仪

主要负责：多媒体数据的爬取、清洗、梳理，论文的部分完善

3 数据介绍

火影忍者涉及到的数据量十分庞大，包含各种图片、音频等资源。本节将按照图片，音频，问答库顺序分别对数据进行介绍。

3.1 图谱数据概况

3.1.1 各类知识的数量级介绍

数据类别	数量级
总实体个数	200+
总三元组个数	1000±
图片张数	200±
音频资料条数	50±
问答库中问答数量	80±

表 3.1.1 知识数量级

本次图谱的总实体数量即为火影忍者动画内所有人物数量与村落、部落数量之和，共计 200 个左右；总三元组个数包括每个人物的各类属性数量、人物及村落间关系数量、问答库及媒体资料数量之和。

3.1.2 概念树等各类知识的详细情况

我们收集了所有火影忍者中出现的人物的姓名和村落、部落名称，将他们作为实体。把人物按照村落形成类似族谱的关系树（如：某人物是某村落的成员，用 is_a_person_from 关系表示），并梳理村落间的关系（如某村落在某地，用 is_a_village_in 关系表示），同时梳理人物之间的关系（如两个人物间的父子关系，用‘父子’关系表示）。在将所有实体组建成概念树以后，通过百度百科爬取各个人物的基本属性，在进行数据清洗后我们最终选择性别、生日、身高、体重、查克拉属性、血型这六个标签作为每个人物的基本属性，并对所有实体的基本属性内容进行完善。

3.1.3 其他图谱重要信息和数据介绍

除去百度百科所爬取的基本信息，我们还收集了多媒体信息和问答库信息作为对知识图谱的信息补充。

在多媒体信息内容方面，我们主要搜集了图片和音频资源。在图片数据方面，我们通过对维基百科和萌娘百科的爬虫，获取了拥有单独百科词条的火影忍者人物的个人简介图片链接，总共爬取的大约七成的人物的个人介绍图片，每位人物的图片数量在 1-4 张不等；在音频资源方面，我们通过在网易云音乐以‘火影忍者’为关键词的歌单搜索，获得了榜单上排名最高的包括动漫背景音乐、主题曲、人物热血语录等在内的 50 条音频链接。

在问答库信息方面，我们主要通过知乎来爬取问答数据信息。通过以‘火影忍者’、‘火影忍者（书籍）’、‘火影忍者疾风传‘为话题，并按照’回答字数在 2000 字以内、支持数在 1000 次以上‘的标准爬取合适的问题及答案。

3.2 媒体数据的爬取

我们的媒体数据主要分为图片数据和音频数据。

图片数据是根据不同的人物名称在百科页面中爬取的人物简介中的人物图片。由于百度百科、搜狗百科、智库百科等国内大型常用百科的网页都设置了反爬虫机制，无法通过爬虫获取图片信息，我们选择了维基百科来进行图片爬虫。但维基百科中只有较著名的火影忍者人物拥有单独词条，占所有人物的比例只有三成左右，所以我们在进行第一次维基百科爬虫后统计了未能获得图片的人物，并通过萌娘百科进行第二次爬虫。在二轮爬虫过后，爬取到图片的人物数量达到七成左右。剩下的人物由于出场次数不高，在整部剧集中影响不大，就不再另外获取图片数据。

3.2.1 人物图片的爬取

在爬虫过程中，首先进入百科页面的 html 文件，找到人物图片所在位置和标签。如图 3.2.1 所示。

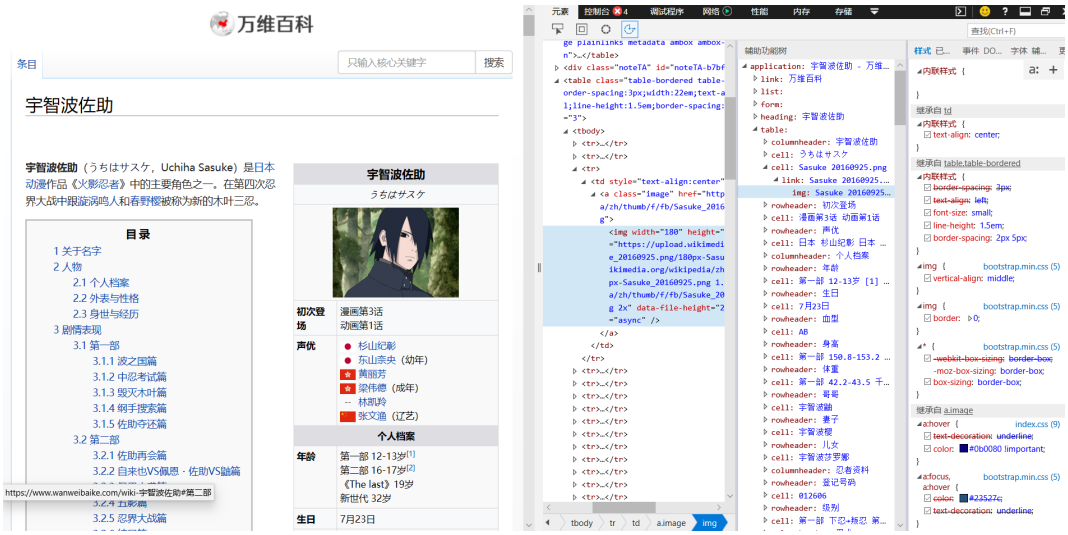

图 3.2.1 百科页面

根据所有人物名字列表，根据维基百科网址的生成格式，生成所有人物百科网页网址列表。（由于人物名字数量太多，放在同一列表输入会引起错误，所以把所有人物名称分为五个列表输入）。接着使用 BeautifulSoup 库解析网页并获取图片链接，在爬取每个人物的图片后判断是否为空，如果为空则代表第一次爬虫未能爬到图片，该人物名字加入第二轮爬虫的列表。实例代码如下：

第一轮爬虫结束后根据所得的二轮爬虫名单再进行而二轮萌娘百科爬虫，最后将所得信息导出到 json 文件。

在完成后检查所有爬取的链接，我们发现一些人物名称存在歧义，例如“赤土”一词在维基百科中的词条是指泥土的一种，所以图片链接出现错误。对于该类可能存在歧义的人物名称，我们进行了检查，并手动替换了图片链接。

3.2.2 音频数据的爬取

我们在网易云音乐中以火影忍者为关键词搜索歌单，选取了排行榜前五名的歌单，内容包括背景音乐、主题曲、热血语录等。通过 BeautifulSoup4 进行网页爬虫并获取歌单内歌曲地址，并导出至 json 文件中。

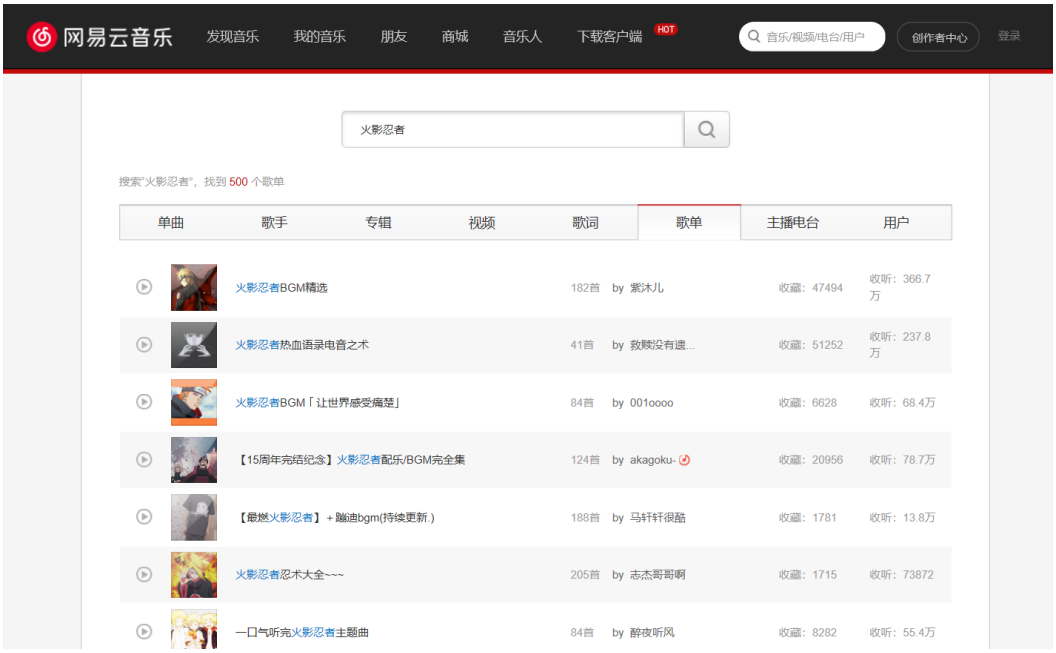

图 3.2.2 网易云音乐搜索结果图示

3.3 问答库的爬取

这里进行一次补充说明：在第一次汇报时，老师曾经跟我们说过，可以去寻找一些相关的垂域网站，但是在网上搜索之后发现好像没有，只有火影忍者游戏的官网，但是漫画和游戏的内容还是有一些区别，所以我们没有使用。此次爬取过程当中，我们主要是在知乎进行爬取。知乎的爬虫比较简单，几乎没有反爬机制，简单研究后就能上手。经过考虑后，我们在知乎选取了三个话题：火影忍者、火影忍者（书籍）、火影忍者疾风传。和之前爬取实体信息一样，我们从网上寻找了一些现有的代码，在此基础上进行了一些修改，来进行问答库的爬取。通过修改程序的部分参数，获取我们想要的答案：我们只需要字数在 2000 以内，支持数在 1000 以上的回答，因为我们认为只有这些回答具有一定的意义。

```
1         if int(item["target"]["voteup_count"]) < 1000:
2             continue
3         if len(item['target']['content']) > 2000:
4             continue
```

code3.3.1 筛选 2000 以内 1000 以上回答的代码

获取完成后，我们开始尝试将这些问答写入 json 文件。我们先将这些问答写入 txt 文件中，格式

为一行问题，一行答案，一行分隔。

```
1  Q:16岁了，写轮眼还没开眼，这正常吗？
2  A:<p>首先...你姓什么？</p><p>你要明白一个事实:</p><
3
4  Q:《火影忍者》中有哪些细思恐极的细节？
5  A:日向宁次说过，人的命运从出生起就定了。鸣人不信，还
6
7  Q:16岁了，写轮眼还没开眼，这正常吗？
8  A:<p>泻药。</p><p>人在妙木山，刚下飞雷神。</p><p>首
9
```

图 3.3.1 问答库图示

因为问题比较多，如果一次性生成可能会有一些辛苦，我们考虑每次录入 30 个，以减轻压力，提高容错度（不用重新生成），对应的 json 文件打开方式为‘a’，即可以追加写入的方式。

当读入一个完整的题目（包括问题与回答）后，为了提高可读性，我们先利用函数去掉其中的部分字符。首先看问题，看这个问题适不适合读入问答库，一些明显关联度不大的问题直接输入 n 否决，光看题目看不出的可以输入 u 再查看一下问题的答案后进行确定。如果确认这个问题适合进入问答库，就要为他选出标签，并写进 json 文件中。

4 图谱构建

本节将介绍图谱构建的过程，包括对初始数据处理的一些步骤，部分代码实现，问答库和媒体库的爬取及挂载等。

4.1 实体属性载入

通过之前爬取的数据，我们可以看到，百度百科当中爬取的信息已经有实体的部分属性，也就是百度百科页面上对人物概况的一个简述。在此基础上，我们可以尝试对它们进行整理与加工，再使用，这样就能减少工作量。

中文名	大蛇丸	体　重	57.3kg（第二部）
外文名	おろちまる	所　属	火之国·木叶隐村
	Orochimaru	忍者登记号码	002300
其他名称	“灾厄之风”	忍校毕业年龄	6岁
配　音	松本和香子	曾所属组织	木叶、晓、音
	山口真弓（少年）	食物喜好情况	喜欢苹果、鸡蛋；讨厌冷的东西
	伪装草忍：山口由里子	喜欢的话	破坏、混沌
	女体：小岛幸子/陈欣（香港）、蘇强文（代配）	想挑战的对手	三代火影猿飞日斩（原带队老师）
登场作品	《火影忍者》系列及衍生作品	原队友	自来也、纲手、油女龙马
生　日	10月27日（天蝎座）	已知弟子	御手洗红豆，宇智波佐助
年　龄	54岁（第二部）→70岁（博人传）	部　下	音忍五人众、香燐、重吾等
性　别	男（本体及灵魂）→不定（由容器的性别决定）	儿　子	Log、巳月
血　型	B型	通灵术	蛇类、罗生门、秽土转生
身　高	179.4cm→172cm（第二部）	查克拉属性	火、风、雷、土、水、阴、阳
		特　性	通灵
		武　器	草薙剑·空之太刀

图 4.1.1 百度百科大蛇丸的信息

{'newLemmaId': '531127', 'lemmaId': '28245', 'title': '大蛇丸（日本动漫《火影忍者》中角色）', 'name': '大蛇丸', 'tags': [], 'description': '大蛇丸，日本漫画《火影忍者》及其衍生作品中的主要角色，原火之国木叶隐村的“三忍”之一，与自来也、纲手同为第三代火影猿飞日斩的弟子。具有极其强大的实力和不死身。擅长研究忍术并渴望得到写轮眼。本身野心极大，由于目睹了太多人的死亡、知道生命是脆弱的而误入歧途，他认为人体中蕴含着一生都无法使用的力量，因此他想获得长生不老从而学习所有忍术，掌握世间的真理。其野心被多次粉碎，在佐助与鼬一战中被鼬的十拳剑封印。后在第四次忍界大战中，从御手洗红豆和药师兜的身上看见了药师兜的失败，彻底醒悟。之后被佐助复活，与四位火影和鹰小队前往战场支援忍者联军。忍界大战结束后，被允许在不伤害他人性命的前提下继续实验，如今制造了人造人巳杯、巳月， 并留在音忍村中生活。', 'view_number': 0, '所属': [['火之国·木叶隐村', '-1', '']], '曾所属组织': [['晓', '2976952', '']], '武器': [['草薙剑·空之太刀', '-1', '']], '外文名': [['おろちまる Orochimaru', '-1', '']], '想挑战的对手': [['三代火影猿飞日斩（原带队老师）', '-1', '']], '体重': [['57.3kg（第二部）', '-1', '']], '忍校毕业年龄': [['6岁', '-1', '']], '中文名': [['大蛇丸', '-1', '']], '喜欢的话': [['破坏、混沌', '-1', '']], '通灵术': [['蛇类、罗生门、秽土转生', '-1', '']], '食物喜好情况': [['喜欢苹果、鸡蛋；讨厌冷的东西', '-1', '']], '生日': [['10月27日（天蝎座）', '-1', '']], '性别': [['男（本体及灵魂）→不定（由容器的性别决定）', '-1', '']], '登场作品': [['《火影忍者》系列及衍生作品', '-1', '']], '其他名称': [['“灾厄之风”', '-1', '']], '血型': [['B型', '-1', '']], '配音': [['松本和香子山口真弓（少年）伪装草忍：山口由里子女体：小岛幸子/陈欣（香港）、蘇强文（代配）', '-1', '']], '忍者登记号码': [['002300', '-1', '']], '查克拉属性': [['火、风、雷、土、水、阴、阳', '-1', '']], '身高': [['179.4cm→172cm（第二部）', '-1', '']], '年龄': [['54岁（第二部）→70岁（博人传）', '-1', '']], '儿子': [['Log', '91326', ''], ['巳月', '3106243', '']], '部下': [['香燐', '23780512', ''], ['重吾', '4122417', ''], ['音忍五人众', '3839900', '']], '原队友': [['油女龙马', '22870927', ''], ['纲手', '807113', ''], ['自来也', '7041', '']], '已知弟子': [['宇智波佐助', '401898', ''], ['御手洗红豆', '23777635', '']], '特性': [['通灵', '5414820', '']], 'url': 'https://baike.baidu.com/item/%E5%A4%A7%E8%9B%87%E4%B8%B8/531127'}

图 4.1.2 爬取后的文档

图 4.1.1 展示的是大蛇丸的信息，图 4.1.2 展示了爬取后的文档。然而，并不是所有的人物都有这么丰富的信息。在观察了内容后，我们最终选取了以下几个标签作为属性的主要内容：性别、生日、身高、体重、查克拉属性、血型。

由于创建实体的代码和加载实体属性的代码并不是同一份代码，且并不是同一个同学负责完成，我们采取的方法并不是直接生成 owl 文件。我们考虑在爬取信息获得的 json 文件提炼出这一部分需要的信息，并且将它转化为文件所需要的格式，再复制进去。因为实体的数目不大，所以整体工作量也并不大。代码的内容如下：

```
import json
f=open('data.json','r',encoding='utf-8')
l=[]
for i in f.readlines():
    dic=json.loads(i)
    l.append(dic)
```

code4.1.1 提取有用信息

其次，我们考虑将 l 列表中这些已有的信息进行整合，得到一个字典 d。字典 d 的 keys 表示实体名称，values 表示实体所指向的这些属性。通过观察 owl 文件内部属性，我们写出了以下的代码。其中，为了方便复制粘贴，我们对 keys 的字符进行了切片。

```
name=[],zf=[]
for i in l:
    name.append(i['name'])
    s=''
    if
    '''
    '''
    '''
    s+='……'
    zf.append(s[8:-1:])
d=dict(zip(name,zf))
```

code4.1.2 切片操作

实验运行时，用户输入实体姓名，程序输出实体对应的属性，可以直接输入。当实体没有这些属性时，程序直接输出’Not Found’以提示用户。最后输入数字 0 时，程序结束。

```
# coding=utf-8
s=input()
while s!='0':
    if s in d.keys():print(d[s])
    else:print("Not Found!")
    s=input()
```

code4.1.2 异常处理

我们可以做这样的输入尝试，会得到相应的结果：

```
大蛇丸
<untitled-ontogogy-25:性别>男（本体及灵魂）→不定（由容器的性别决定）<untitled-ontogogy-25:性别>
        <untitled-ontogogy-25:生日>10月27日（天蝎座）<untitled-ontogogy-25:生日>
        <untitled-ontogogy-25:查克拉属性>火、风、雷、土、水、阴、阳<untitled-ontogogy-25:查克拉属性>
        <untitled-ontogogy-25:身高>179.4cm→172cm（第二部）<untitled-ontogogy-25:身高>
        <untitled-ontogogy-25:体重>57.3kg（第二部）<untitled-ontogogy-25:体重>
        <untitled-ontogogy-25:血型>B型<untitled-ontogogy-25:血型>
漩涡鸣人
<untitled-ontogogy-25:性别>男<untitled-ontogogy-25:性别>
        <untitled-ontogogy-25:生日>10月10日（天秤座）<untitled-ontogogy-25:生日>
        <untitled-ontogogy-25:查克拉属性>火、风、雷、土、水、阳<untitled-ontogogy-25:查克拉属性>
        <untitled-ontogogy-25:身高>147.3-149.5cm（第一部）→166cm（第二部）→180cm(剧场版10）<untitled-ontogogy-25:身高>
        <untitled-ontogogy-25:体重>40.1-40.6kg（第一部）→50.9kg（第二部）→60kg（剧场版10）<untitled-ontogogy-25:体重>
        <untitled-ontogogy-25:血型>B型<untitled-ontogogy-25:血型>
0

Process finished with exit code 0
```

图 4.1.3

运行程序时利用 pycharm 的界面，可以对着 owl 文件中实体的名字进行复制粘贴，整体是非常快的。虽然这样仍然不是全自动化，但比起全人工，效率还是高了很多。

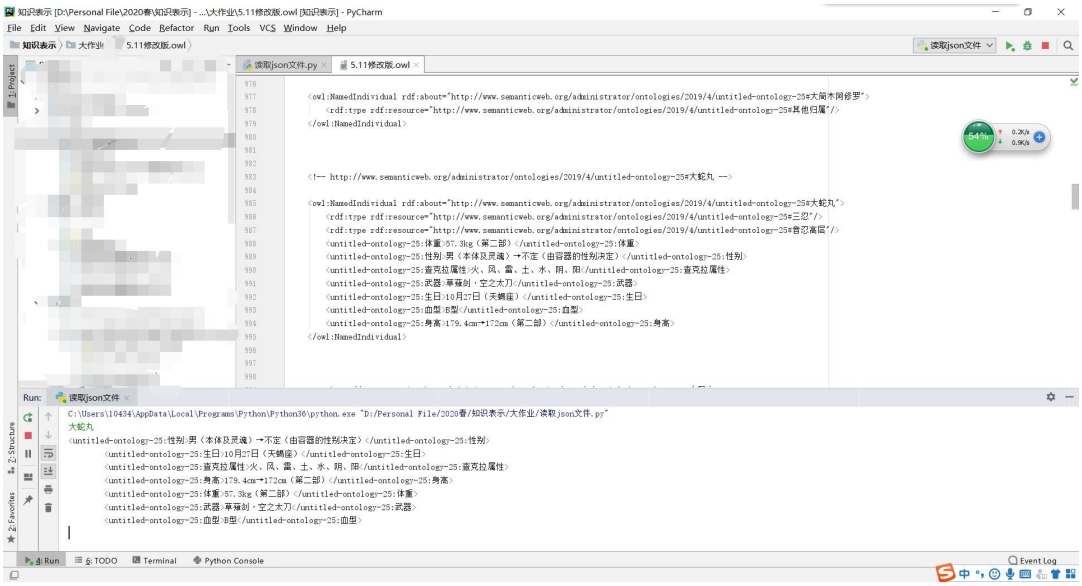

图 4.1.4

4.2 json 文件生成

采用代码自动生成最终的 json 文件，示例文件图片如下：

图 4.2.1 作业 json 文件展示

5 心得体会

赖丹玲：这次我主要负责实体的属性载入，问答库的爬取、清洗与整理。在学习的过程中我了解了知识图谱的知识，其中针对我负责的地方，我深刻地感受到了机器自动化的重要性，我只是实现了部分自动化，比起那些纯手工的同学，已经很有效地提高了效率，如果可以实现全自动化，相信构建可视化的知识图谱就会变得轻松，不再困难。希望以后可以丰富目前我所写的两个程序与修改的一个程序，在别的问题上可以利用所学去实现别的主题的知识图谱。但是同时，我也发现此次我实验中做的不够好的地方：由于选题问题，获得数据的来源被迫单一；有一些多词条页面处理起来麻烦，还没有想好处理的方法；爬取的问题无厘头的很多……以后我也会努力学习知识，争取改进这些不足，像学长那样写出实用的程序，让学弟学妹可以直接利用，减少他们的工作量，把时间和关键放在知识图谱的学习上。

马标：这次我主要负责初始实体的爬取，缺省数据的完善，可视化与中期汇报，文章校验与排版。在本次合作的过程中，由于前期不了解知识量，以为数据量几百个比较轻松，就只想着大部分都人工处理。但在处理的过程中发现，这种枯燥且重复率极高的工作，人工处理是十分浪费资源的。比如形成实体转化 owl 的 txt 文件，这种人工处理工作量稍大，机器处理又十分方便的工作，还是更加适合运用编程处理。因此，在前期手工实现数据补全和完善，手动 protege 挂载之后，在处理后期 json 文件的过程中，我便去了解如何基于 python 实现 json 文件的读写，并实现自动读取属性值，自动挂载属性。但是对于数据爬取那一块，由于时间紧张，这里并没有实现大型爬虫来读取全网数据，且爬取出来的数据杂乱无章的很多，没办法集中处理。后期也会针对这些问题做出相关的学习和弥补。

邓仪：通过本次的知识图谱构建任务，我了解到了图谱的基本构建原理，同时也认识到了知识图谱的重要性和其中的重难点。知识图谱的构建并非只是简单的实体之间的关系整理，它涉及到了很多其他的相关专业知识，包括自然语言处理、爬虫技术、数据处理等许多方面。我们构建的图谱的实体规模在 200 个左右，但真正专业有效的图谱的数据规模是非常庞大的，这对数据的处理能力非常高，同时也需要将我们的工作更加自动化，才能做到效率与准确度的结合。这其实对我们的代码能力有着很高的要求。同时，在我负责的工作（多媒体数据的爬取、整理）中，我发现许多在互联网上获得的数据在简单的批量操作中会存在许多问题，如我在爬取图片的过程中遇到的人物姓名的歧义问题，这些都是数据获取中的细节问题，要让构成的知识图谱更加地完善、准确，我们必须注重并处理好细节问题。同时，在这次的任务中我的两位队友对于图谱前期的选题、框架搭建等问题上都给予了我们的队伍很大的帮助，希望自己以后对于知识图谱能有更全面、更深刻的理解，在面对新的邻域和知识时能够有更强的学习能力，并提升自己对于问题的处理能力和代码能力。

6 结语

此次大作业，我们针对火影忍者这一话题在网上进行数据的搜集、筛选与清洗，并结合在知识表示课上所学到的技术，制作了一个火影忍者知识图谱。在文章和附录文件中，我们附上了相关联的问答库数据、媒体数据。总体来说，作业的完成度较高。美中不足是较多部分是自己手写代码进行半自动化处理，而非全自动化处理，如果能够在以后的学习生活中摸索这个方向，相信会有更大的进步。

我们的直接成果是一份知识图谱，但我们更重要的是收获了眼界，能够做到积极思考，积极学习。通过此次的大作业实践，我们深刻的感受到了只有实践才能出真知，只有自己亲自动手去做，我们才会意识到实现程序自动化处理和手动化处理之间原来有这么大的差别，才能知道知识图谱对于问题的处理、智能的交互有这么大的作用，能够带来这么多的好处。现在知识在人工智能中的地位越来越重要，而我们做得还不够好，相信在不断的学习实践探索当中，我们能够作为浩瀚星海中的个体，勇敢的发展技术，探索人工智能宇宙未知的部分。

附录

知乎爬虫代码源码 https://zhuanlan.zhihu.com/p/87336715
知乎火影忍者话题页 https://www.zhihu.com/topic/19555130/questions
知乎火影忍者疾风传话题页 https://www.zhihu.com/topic/19680979/questions
知乎火影忍者（书籍）话题页 https://www.zhihu.com/topic/20718094/questions
网易云音乐火影忍者专辑页 https://music.163.com//search/m/?s=%E7%81%AB%E5%BD%B1%E5%BF%8D%E8%80%85type=10