计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2021,30(10):59−67 [doi: 10.15888/j.cnki.csa.008218]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

面向多源异质数据源的学科知识图谱构建方法①

李家瑞, 李华昱, 闫　阳

(中国石油大学 (华东) 计算机科学与技术学院, 青岛 266580)
通讯作者: 李华昱, E-mail: lhyzj@upc.edu.cn

摘　要: 针对以分散形式存储学科信息导致资源难以统计的问题, 基于计算机学科领域本体模型, 融合多源异质的学科数据构建高校计算机学科知识图谱. 首先通过网络爬虫等技术从相关网站和已有文档中获取领域知识, 并基于 BERT 模型对数据进行清洗; 然后利用 Word2Vec 判断人物研究方向之间的相似度, 解决实体对齐问题; 最终将数据导入 Neo4j 图数据库中实现知识的存储. 根据构建好的知识图谱建立计算机学科可视化系统, 能够提供信息检索与图形显示等多种功能, 实现计算机学科基础数据的快捷查询和资源统计, 以期促进后续的学科评估工作更加高效地完成.

关键词: 知识图谱; 计算机学科; 图数据库; 可视化系统

引用格式: 李家瑞,李华昱,闫阳.面向多源异质数据源的学科知识图谱构建方法.计算机系统应用,2021,30(10):59–67. http://www.c-s-a.org.cn/1003-3254/8218.html

Construction of Discipline Knowledge Graph for Multi-Source Heterogeneous Data Sources

LI Jia-Rui, LI Hua-Yu, YAN Yang

(College of Computer Science and Technology, China University of Petroleum, Qingdao 266580, China)

Abstract: It is difficult to count the discipline information stored in a scattered form. With regard to this problem, based on the domain ontology model of computer discipline, the computer discipline knowledge graph in universities is constructed by integrating the multi-source and heterogeneous data. First, domain knowledge is acquired from relevant websites and existing documents through Web crawlers and other tools, and the data are cleaned on the basis of the BERT model. Then, Word2Vec is used to judge the similarity between the research directions of characters, so as to solve the problem about entity alignment. Finally, the data are imported into the Neo4j graph database to realize the storage of knowledge. According to the knowledge graph, the visualization system of computer discipline is established, which can fulfil information retrieval, graphic display, and other functions and realize quick query and resource statistics of computer discipline data. It is expected to facilitate the follow-up discipline evaluation work and make it more efficient.

Key words: Knowledge Graph (KG); computer discipline; graph database; visualization system

高校之间的竞争主要以学科竞争为基础, 学科的实力在某种程度上可以代表院校的水平. 学科评估能够有效且全面地了解学科的建设现况, 通过对学科的正确评估, 寻找建设中存在的问题, 从而进一步明确该学科的前进方向, 实现更好的发展[1]. 由于学科建设的成果涉及很多方面的内容, 以分散的文档、网络资源等形式对学科相关信息进行存储和显示, 不能够全面地展示各项数据之间的关联, 同时会使信息统计和潜在关系的挖掘较为困难, 不利于后续评估工作的开展.

① 基金项目: 国家自然科学基金 (61572522); 中国石油大学 (华东) 研究生创新工程 (YCX2021128)
Foundation item: National Natural Science Foundation of China (61572522); Postgraduate Innovation Project of China University of Petroleum (YCX2021128)
收稿时间: 2021-01-11; 修改时间: 2021-02-23; 采用时间: 2021-04-06

知识图谱 (knowledge graph) 作为大数据时代下一种新型高效的知识组织方式, 能够基于图对多源异构数据进行知识融合与关联[2]. 本文将知识图谱技术应用至高校计算机学科领域, 首先通过网络爬虫和规则映射的方法, 从资源丰富的知网、高校官网、学科评估文件等数据源中获取计算机学科相关的领域知识. 针对可能出现的杂质数据问题, 使用微调后的 BERT (Bidirectional Encoder Representations from Transformers) 模型进行数据的分类, 过滤异类数据. 对于可能存在的人物实体重名问题, 提出一套利用 Word2Vec 进行相似度判断的实体对齐方法, 解决知识融合时的实例冲突问题. 最终将知识导入 Neo4j 图数据库中完成知识图谱的存储, 并基于此知识图谱建立起计算机学科可视化系统, 实现对各类数据的信息查询、关系展示等多种功能, 为上述问题提供了较好的解决思路.

本文的组织结构如下: 第 1 节介绍知识图谱的相关知识及本文的研究思路; 第 2 节介绍计算机学科本体的构建; 第 3 节介绍知识图谱构建的相关内容, 主要包括知识获取、知识融合和知识存储等过程; 第 4 节介绍计算机学科可视化系统的实现与性能评估; 第 5 节为总结与展望.

1 相关知识及研究思路

知识图谱的概念是由 Google 在 2012 年首先提出的, 目的是改善搜索引擎返回结果的质量, 提升用户搜索体验. 根据覆盖面的不同, 可以将知识图谱划分为通用知识图谱和领域知识图谱. 其中通用知识图谱的覆盖面更广, 涵盖了现实世界中的许多常识性知识, 较为知名的大规模通用知识图谱有 DBpedia、Wikidata、Freebase 等, 这些知识图谱的规模都很庞大, 但对抽取知识的质量要求并不严格, 包含各个领域的知识结构也较为简单, 所以在应用于特定领域时表现不是很好. 领域知识图谱则是面向具体的领域构建, 对该领域内知识的准确度和深度等都有着非常严格的要求, 能够为目标领域的上层应用提供很好的支持. 知识图谱目前已经在医疗、电商、法律等领域有了较多的应用, 比如通过基于知识图谱的聊天机器人, 让用户自主了解有关医疗保健和药物方面的知识[3]; 基于构建的盗窃案件法律文书知识图谱, 设计推理规则以提供相似案件量刑参考[4].

知识图谱模型以图论中的图结构 $G=(V, E)$ 为基础, 其中, V 是顶点集, E 是边集. 知识图谱可以被认知为由一条条事实知识构成, 知识可由三元组 (h, r, t) 的形式表示, 其中, h 代表头实体, t 代表尾实体, r 是两个实体之间的关系. 在构建知识图谱时, 主要有自上而下和自下而上两种构建方式. 自上而下方式指直接从较高质量的数据集中抽取相关的本体和模式信息; 而自下而上是指从采集到的大量数据中提取出资源模式, 然后选择其中置信度高的作为后续知识图谱构建的基础[5]. 对于一些较为成熟、知识体系完备的领域, 通常可以采用自上而下的构建方式, 即先对 schema 本体进行定义, 再使用有监督、半监督和无监督等方法抽取知识, 最后结合知识融合、知识推理等机制使得构建出的领域知识图谱更加完善.

知识图谱的一般构建流程为: 首先确定知识表示模型; 然后根据数据的不同来源, 选择不同的技术手段获取知识, 并导入至知识图谱数据库中; 接着综合利用知识融合、知识推理和知识挖掘等技术对构建出的知识图谱进行规模和质量上的提升; 最后根据目标场景的不同需求设计有效的知识访问与呈现途径, 如人机交互问答、图谱可视化分析、相似推荐等.

本文对计算机学科知识图谱的总体构建流程如图 1 所示. 首先对计算机学科领域本体进行建模, 定义概念之间的语义关系. 对于不同的数据来源, 设计相应的表格映射与网络爬虫算法, 并结合抽取规则得到领域数据, 然后利用基于 BERT 的分类方法对数据进行清洗过滤, 实现知识的获取. 在知识融合的过程中, 通过训练好的 Word2Vec 词向量模型判断词相似度, 进而完成实体的对齐. 最后将融合整理后的数据导入至 Neo4j 图数据库中进行知识存储. 基于上述构建好的知识图谱, 本文搭建了计算机学科的可视化系统, 可以提供基础信息查询、关键词检索、递进式检索和语义搜索等多种功能, 同时以图形化的方式展示结果, 便于用户完成实体关系查找和资源统计等工作.

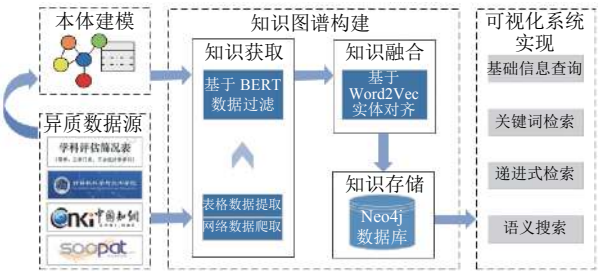


图 1 总体构建流程

2 计算机学科本体构建

本体定义了知识图谱的类集、关系集、属性集等, 主要强调概念之间的关系, 是对知识图谱模式层的管理. 通过构建本体模型, 可以对实体、关系以及实体属性等进行约束规范, 作为后续知识抽取与组织的指导[6]. 本文以中国石油大学 (华东) 计算机科学与技术专业第四轮学科评估简况表为主要知识源, 结合具体的计算机学科领域相关网站, 使用 OWL 语言作为本体描述语言, 通过 Protégé本体开发工具, 完成高校计算机学科本体的构建.

计算机学科本体中包含的概念及其构成的关系结构通过 Protégé中的 OntoGraf 工具展示如图 2 所示. 在此本体模型中, 主要包含了教师、校友、在读本校生、在外留学生、院校机构、国家级项目、省部级项目、期刊论文、会议论文和专利等 10 个类, 且子类概念之间通过多种关系相互关联. 本体中将概念之间的关联关系表示为语义关系, 在 Protégé中也被称为对象属性, 包括通用语义关系和自定义语义关系[6]. 本文构建的本体中包含了多种自定义语义关系, 相关的概念及详细说明如表 1 所示.

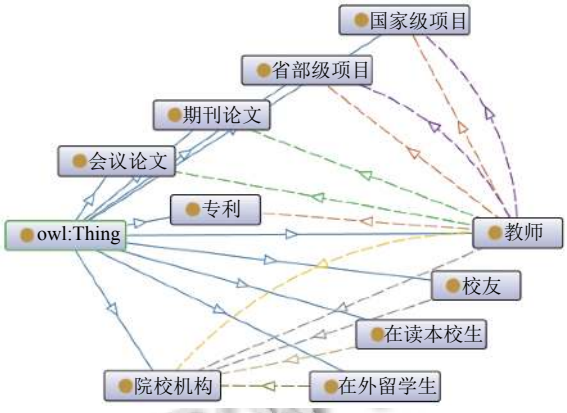


图 2 学科知识图谱本体模型

3 学科知识图谱构建

3.1 知识获取

3.1.1 数据来源

在知识图谱构建的过程中, 数据是极其重要的底层支持, 只有获取到大量研究领域中的数据, 才能够建立一个质量较好的知识图谱. 一般用于构建知识图谱的知识来源可以是结构化数据、半结构化数据、非结构化数据、物联网传感器和人工众包等[7]. 通过调查发现, 高校计算机学科领域内的相关数据主要分布在电子文档以及各种网站中, 比如学科评估文件、高校官网、国家知识基础设施等网站, 这些数据源都分别涵盖了不同类别的学科领域数据, 包括教师信息、论文、专利、科研项目等. 因此本文主要从表 2 所示的来源中获取领域知识.

表 1 自定义语义关系表

关联概念	语义关系	被关联概念	语义关系说明
教师	TeachIn	院校机构	任教于
教师	GraduatedFrom	院校机构	毕业于
校友	GraduatedFrom	院校机构	毕业于
在读本校生	StudyIn	院校机构	就读于
在外留学生	StudyIn	院校机构	就读于
教师	AuthorIn	期刊论文	发表论文
教师	AuthorIn	会议论文	发表论文
教师	Lead	国家级项目	主持项目
教师	Lead	省部级项目	主持项目
教师	ParticipateIn	国家级项目	参与项目
教师	ParticipateIn	省部级项目	参与项目
教师	InventorIn	专利	发明专利

表 2 知识图谱数据来源

数据源	包含数据类别
计算机学科评估简况表	教师、校友、在读本校生、在外留学生、院校机构、期刊论文、会议论文、国家级项目、省部级项目
高校官网	教师、院校机构
中国知网	期刊论文、会议论文
SooPAT	专利

3.1.2 数据提取过程

对于以表格文档形式存储的类结构化数据, 例如高校计算机学科评估简况表, 可以采用基于映射的信息抽取方法, 即先将待提取的表头字段与上文构建的学科本体中的数据属性之间建立一一映射关系, 然后使用本体定义的词汇描述提取出的结构化信息, 从而防止属性名之间同义异名问题的发生, 完成对目标表格单元中数据的提取.

对于存储于互联网网页中的数据, 由于不同网页的内容组织结构具有较大差异, 所以在爬取数据时, 需要根据不同的目标网站制定针对性的爬虫方法. 常用的网络爬虫有 Requests、Selenium 等, 不同爬虫的实现原理也存在差异: Requests 通过初始 URL 下载网页, 再结合网页解析库解析其中包含的标签内容, 获取新的 URL 依次进行爬取[8]; 而 Selenium 则是通过模拟用户的操作行为, 比如点击按钮、输入文本等方式, 直接运行在浏览器中, 实现网页间的正确跳转[9]. 不同的实

现原理也决定了每种类型爬虫的优缺点以及各自的适用场景: Requests 爬取的速度快, 但当跳转页面的 URL 无法获取时会导致爬取中断, 因此适合于目标 URL 可以获得的情况; 当目标 URL 不可直接获得时, 可以采用 Selenium 进行页面跳转, 但其存在的缺点是需要等待浏览器打开加载, 所以爬取效率远不如 Requests.

本文提出了一种网络爬虫算法, 能够根据网页组织形式的不同, 灵活调用以上两种工具, 在完成目标数据获取的同时, 又尽可能地提高爬取效率. 具体的爬虫工作流程如图 3 所示.

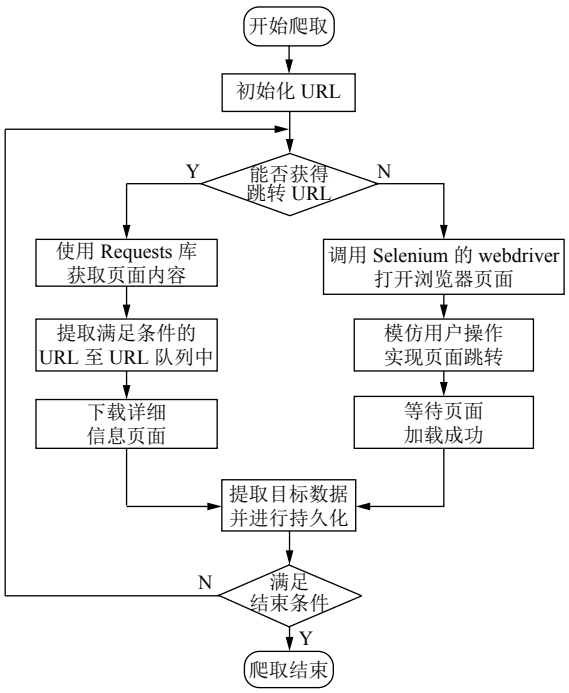


图 3 爬虫工作流程

算法在爬取开始后需要首先对网页中跳转 URL 的组织情况进行判断. 比如对于高校的官方网站, 其师资列表页面内一般包含教师详细信息页面的 URL, 因此可以通过以下步骤爬取: (1) 从师资列表页面 URL 开始, 通过 Requests 库获取页面内容; (2) 按照定义好的页面抽取规则, 取出教师详细信息页面 URL 并放入待抓取 URL 队列中, 若 URL 不完整, 则根据相似页面的 URL 构造对缺失字段进行补充; (3) 根据待抓取 URL 队列下载详细信息页面, 从中提取目标数据, 并保存至数据存储文件中; (4) 整个过程循环执行, 直到队列中的所有 URL 爬取完毕[8]. 而对于中国知网等一些不能直接获得跳转页面 URL 的网站, 可以选择 Selenium 工具爬取, 实现流程为: (1) 配置 URL 地址及相关参数, 调用 Selenium 的 webdriver 打开浏览器页面; (2) 等待页面加载完成, 定位搜索框与按钮元素, 完成搜索条件输入后, 模拟用户点击按钮进行跳转; (3) 页面加载成功后, 使用 XPath 提取目标数据, 并进行数据持久化操作; (4) 重复以上过程, 直至满足爬取数量或所有页面爬取完毕[9].

3.1.3 数据清洗

考虑到在进行数据爬取时会出现杂质数据的问题, 例如定位至错误的 HTML 标签, 或由于解析出错导致文本缺失等, 因此有必要在存储数据前进行数据清洗操作. 本文通过实验比较 TextCNN 和 BERT 两种模型对相关学科数据分类的结果, 设计出一种分类策略实现对爬虫数据的清洗过程.

文本分类模型 TextCNN 是由 Kim 等在 2014 年提出的, 其目的是对卷积神经网络 CNN 进行变形, 然后引入至文本分类的任务中[10]. TextCNN 的网络结构分为 4 层, 包括嵌入层、卷积层、最大池化层和全连接层, 通过输入待分类文本的词向量矩阵, 经过卷积和池化操作后, 输出该文本对应每个类别的概率分布[11,12]. BERT 主要基于双向 Transformer 编码器结构实现, 同时利用遮蔽语言模型 (MLM) 和下一句预测 (NSP) 两个无监督任务进行联合预训练, 使其经过特定的微调操作后即可迁移到下游自然语言处理任务中, 比如内容检测[13]、命名实体识别[14]、文本分类[15,16] 等.

为了确定使用哪种分类模型对学科数据的清洗效果更好, 以清洗论文专利类数据为例进行对比实验. 实验数据集中以包括论文专利的科研成果类数据为分类的正样本, 以非论文专利类数据作为负样本, 总共包含约 13 000 条数据, 取其中的 80% 作为训练集、20% 作为测试集对两类模型进行训练和测试.

对于 TextCNN 模型的嵌入层, 首先对文本数据进行分词处理, 然后使用基于 26 GB (800 多万条) 百度百科词条、13 GB (400 多万条) 搜狐新闻和 229 GB 小说合并的训练语料进行训练所得到的 Word2Vec 词向量模型[17], 对每个文本分词词语生成其 128 维嵌入表示, 整合后构成词向量矩阵作为输入数据. 在卷积层中, 设置卷积核尺寸为 [3, 4, 5], 每个尺寸的卷积核个数为 64, 提取输入矩阵不同的 feature map 特征. 最大池化层选择 1-max pooling 方式, 抽取 feature map 向量中的最大值, 即捕获其中最重要的特征. 最后将经卷积池化获取的特征传至 Softmax 层, 得到文本的分类标签结果.

本文设置 TextCNN 模型训练的批次大小值为 64, 测试过程中不同迭代步数的准确率变化如图 4 所示.

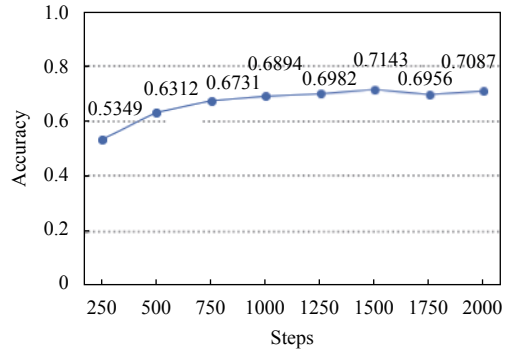


图 4 TextCNN 模型测试结果

对于 BERT 模型, 本文采用在中文维基百科上进行预训练后得到的 Bert-base-Chinese 模型作为基准模型, 模型总共包含 12 层, 隐层为 768 维, 使用 12 头模式, 共 1 亿多个参数; 在微调模型时使用与 TextCNN 模型相同的数据集, 设置学习率为 $2e^{-5}$, 批量学习的 batchsize 为 32, Epoch 循环次数为 5 次, 最终得到的测试准确率如图 5 所示.

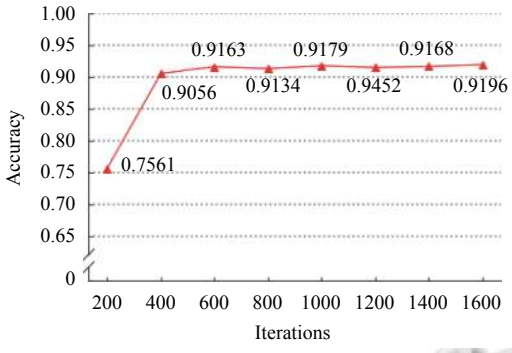


图 5 BERT 模型测试结果

对比实验结果可以发现, 在此场景中 BERT 的测试准确率能够达到 0.91 左右, 高于 TextCNN 模型. 因此本文选择基于 BERT 模型的方法, 以 Bert-base-Chinese 作为基准模型进行微调操作, 再对爬取到的相关学科数据按类分别进行清洗, 清洗前后的各类数据量统计如表 3 所示.

3.2 知识融合

在对不同来源的知识进行融合时, 容易出现实例异构问题, 即同名实体可能指向不同对象, 而不同名实体可能指向相同对象. 因此需要通过实体对齐技术, 确定不同信息来源中的两个实体是否指向现实世界中的同一个对象, 若是则在实体间构建相应的对齐关系, 完成知识的融合. 通过从知网、SooPAT 等数据源中采集高校计算机学科领域的相关数据, 在构建知识图谱的过程中会出现人物方面的歧义问题. 比如高校教师在不同的时间节点发表论文、发明专利等科研成果, 却由于工作调动等情况被判定为不同的人物实体; 或者同一高校内的重名教师被错误指向为同一实体, 从而造成科研成果信息的错误统计. 因此为了构建准确的高校计算机学科知识图谱, 需要设计出一种适合的实体对齐算法来解决上述问题. 本文采用的实体对齐算法如算法 1 所示.

算法1. entityAlignment()

输入: 待对齐重名实体集合*UnalignedSet*

输出: 判断结果*flag*

```
for 实体对EP in UnalignedSet do
  if Compare(EP.basicInformation) = False then
    flag = 非同一实体
  else
    关键词集合KWL += EP. 关键词
    词相似度WS = Cos(Word2Vec(KWL))
    if WS > 自定义阈值T then
      flag = 同一实体
    else
      flag = 非同一实体
end for
```

表 3 清洗前后各类数据量统计

数据类别	清洗前	清洗后
论文	3297	3208
专利	1189	1150
教师	83	83
院校机构	31	31

算法首先从多数据源中提取出重名人物得到待对齐实体集合; 然后, 通过人物的基本信息进行初步筛选, 基本信息包括性别、民族、出生年月等这些不易改变的属性信息; 最后, 根据人物发表论文或申请专利中的关键词集合, 使用 Word2Vec 获得对应词向量并计算词向量间的余弦相似度[18], 若相似度超过自定义阈值, 则可认为二者研究方向相同, 指代同一实体.

针对如何确定相似度阈值的问题, 本文设计了以下实验进行研究. 首先选取部分高校教师的论文信息作为原始数据, 每位教师随机选取 3 篇论文的关键词组成其研究方向关键词集合, 假设某位教师研究方向关键词集合的长度为 m, 则集合可以表示为:

$$\{K_{s1}, K_{s2}, \cdots, K_{sm}\} \tag{1}$$

然后将该位教师余下的论文分别与该集合组成对

比测试组, 假设余下的某篇论文包含的关键词个数为 n, 则对比的关键词集合为:

$$\{K_{t1}, K_{t2}, \cdots, K_{tn}\} \tag{2}$$

之后使用 Word2Vec 模型得到关键词集合对应的词向量, 研究方向关键词集合的词向量表示为:

$$\{V_{s1}, V_{s2}, \cdots, V_{sm}\} \tag{3}$$

对比的关键词集合的词向量表示为:

$$\{V_{t1}, V_{t2}, \cdots, V_{tn}\} \tag{4}$$

最后计算两个关键词集合之间词向量余弦值的平均值, 将其作为该篇论文与对应教师研究方向之间的相似度:

$$Similarity = \frac{\sum_{j=1}^{n}\sum_{k=1}^{m}\cos\left(V_{sk}, V_{tj}\right)}{n \times m} \tag{5}$$

两个词向量之间的余弦函数 cos(·) 定义为:

$$\cos(V_s, V_t) = \frac{V_s \cdot V_t}{\|V_s\| \times \|V_t\|} = \frac{\sum_{i=1}^{L} V_s^i \times V_t^i}{\sqrt{\sum_{i=1}^{L}\left(V_s^i\right)^2}\sqrt{\sum_{i=1}^{L}\left(V_t^i\right)^2}} \tag{6}$$

其中, L 为通过 Word2Vec 得到的词向量的维度, V^i 为词向量的第 i 个分量.

本文随机抽取了共 2400 组测试数据, 最终观察到关键词相似度的数值分布如图 6 所示. 从图 6 中可以看出, 相同研究方向的论文关键词相似度都在 0.5 以上, 因此本文在实体对齐算法中设置相似度阈值为 0.5.

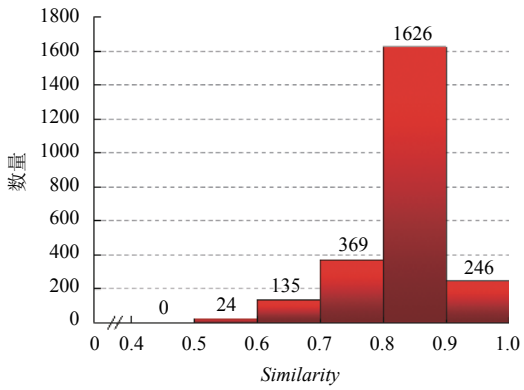


图 6 关键词相似度分布

为了验证算法的可行性, 本文选取了数位重名但研究方向不同的教师, 从知网上爬取其发表的论文信息, 取同一教师的论文关键词集合作为正例数据, 取不同教师的论文关键词集合作为反例数据, 构成了包含 800 余条数据的测试数据集. 然后从中随机抽取 200、400、600、800 条数据, 与人工标注的结果进行准确率的分析计算. 实验结果如表 4 所示, 4 次随机测试的准确率均在 90% 以上, 说明基于 Word2Vec 的人物实体对齐方法识别出的错误数据较少, 可以在高校学科领域的知识融合场景中使用.

表 4 人物实体对齐测试结果

数据条数	正确分类数	准确率 (%)
200	189	94.50
400	367	91.75
600	560	93.33
800	741	92.63

3.3 知识存储

经过清洗对齐处理后的数据, 其内容和格式已经满足学科知识图谱构建的要求, 下一步的工作就是把这些数据导入到底层数据库中. Neo4j 作为一种高性能的非关系型图数据库, 将数据存储在一个超大型网络上, 非常适用于对基于图结构的知识图谱进行存储[19]. 本文通过使用 Python 支持的 Py2Neo 第三方库提供的操作函数, 将各类数据以节点和边等形式导入 Neo4j 中, 并且可以进行对应的增删改查等操作.

最终构建完成的学科知识图谱的数据规模统计如表 5 所示, 图谱中的各类知识形成了一幅庞大且错综复杂的多关系网络, 有助于后续各项功能的实现及性能的优化.

表 5 计算机学科知识图谱数据统计

元素	种类	总数量
实体	10	3504
关系	24	4337
属性	39	22573

4 可视化系统实现

本文基于上述知识图谱开发了一个高校计算机学科的可视化系统, 系统采用 B/S (Browser/Server) 前后端分离的结构模式进行实现, 通过 Python 的 Flask 框架搭建. 前端中使用 Echarts 工具实现数据的图形显示效果[20], 通过文本、力导向图等多种形式对学科领域知识进行可视化显示.

4.1 系统功能

本可视化系统的功能主要包括基础信息查询、关键词检索、递进式检索和语义搜索等, 可以从实体、

属性、关系等多个维度完成知识的搜索与展示.

4.1.1 基础信息查询

基础信息查询功能的目的是统计与被查询实体有关的所有实体和关联关系, 然后以力导向图的形式将实体关系通过图形界面表示出来. 同时使用符合图数据库存储结构的推荐算法, 选出部分与被查询实体相似度最高的同类实体, 作为用户可能感兴趣的推荐信息. 功能实现过程的算法如算法 2 所示.

算法2. basicInfoQuery()

输入: 用户输入的查询字符串*instr*

输出: 查询结果的图形化显示

1. initialize *json_data* //变量初始化

/*直联查询模块*/

2. *Path1* = "(*n* {name:*instr*})-[*r*]-(*m*)" //构造匹配路径

3. *r*, *m* = Neo4j. Cypher(*Path1*) //查找直联实体与关系

4. *json_data* += jsonify(*r*,*m*) //得到JSON格式数据

/*相似推荐模块*/

5. *Path2* = "(*qe*:*Stype*)-[*r*1]-(*e*)-[*r*2]-(*me*:*Stype*)"

6. *me* = Neo4j. Cypher(*Path2*) //查找二度关联实体

7. *me_p* = pathAmount(*me*) //统计各实体关联路径数量

8. *se* = *me*.topK(*me_p*) //选择*k*个路径数量最多的实体

9. *json_data* += jsonify(*se*)

/*结果显示*/

10. Echarts.paint(*json_data*) //使用Echarts绘制图形

此功能主要包含直联查询和相似推荐两个数据处理模块. 在直联查询模块中, 先根据用户的输入构造相应的匹配路径[21], 然后通过 Cypher 语句从 Neo4j 图数据库中查找所有与其有关的实体和其间关系. 在相似推荐模块中, 首先构造多跳匹配路径"(*qe*:*Stype*)-[*r*1]-(*e*)-[*r*2]-(*me*:*Stype*)", 其中 *qe* 指被查询实体, *me* 指匹配到的实体, *Stype* 表示两者为同一数据类型, *r*1、*r*2 和 *e* 代表不做特定要求的关系和实体; 之后统计出匹配到的所有实体和对应的路径条数, 按数量由多到少进行排序, 选择其中的 Top-*k* 个实体作为相似推荐 (本文所取的 *k* 值为 3, 即最多推荐 3 个相似实体). 最后由得到的数据属性值确定节点和连线的类型和标签值, 传入 Echarts 的绘图函数完成图形的绘制与显示.

图 7 所示为输入"中国石油大学 (华东)"的信息查询结果, 界面中包含与此实体直接相连的各类实体节点以及其间关系的说明, 同时也为用户推荐出最相关的同类实体"中国科学院计算技术研究所""中国海洋大学"和"南开大学". 力导向图支持放大、缩小以及图形的移动, 当点击界面上方的类别标签时, 能够对该类所有的实体节点进行隐藏或再现, 便于用户观察和统计. 左键点击节点时, 可以跳转至该实体的详细属性页面, 图 8 所示为"大数据环境下的油气开采创新方法研究与应用示范"项目的详细属性显示.

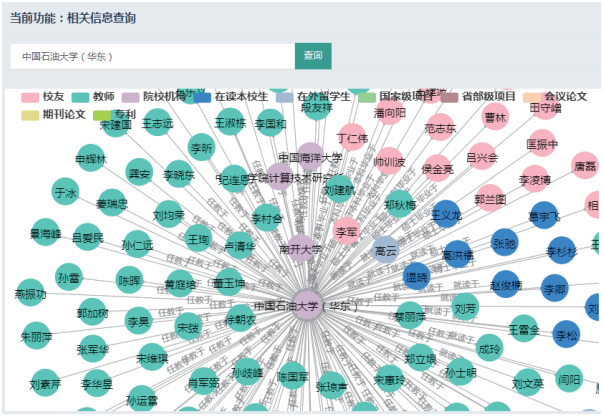


图 7 "中国石油大学 (华东)"查询结果

图 8 详细属性显示界面

4.1.2 关键词检索

关键词检索功能会显示所有与输入关键词相关的实体节点, 同时支持多关键词检索的任务. 系统首先利用哈工大 LTP 语言处理工具对用户输入的关键词进行词性标注, 包括人物、时间、名词等词性类型, 然后根据词性分布构造相应的正则表达式, 从知识图谱中查找符合条件的实体.

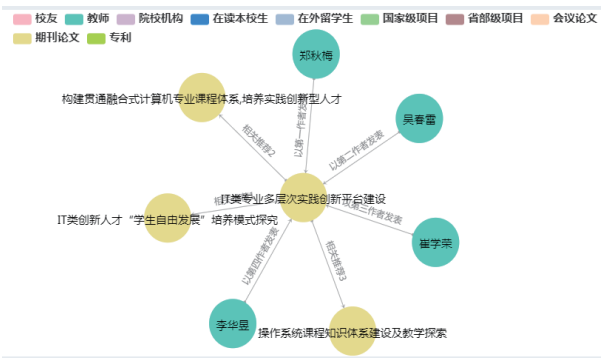
例如, 当输入多个关键词为"神经网络""识别""2019 年"时, LTP 词性标注模块将它们分别标注为"n""v""nt", 对应构造出的正则表达式即为"(?=.*[神][经][网][络]).*""(?=.*[识][别]).*" "(?=.*[2][0][1][9]).*". 之后将这些正则表达式作为属性字段组成 Cypher 语句进行检索, 返回满足条件的实体, 最终结果如图 9 所示.

图 9 多关键词检索结果

4.1.3 递进式检索

右键点击图中任一节点, 即可进入以该实体为中心的下一层知识图谱, 实现知识图谱的递进式查询. 例如当点击图 10(a) 中的节点“IT 类专业多层次实践创新平台建设”后, 系统页面弹出确认提示框, 用户确认后可跳转至以该节点为中心的信息展示界面, 递进式检索结果如图 10(b) 所示.

(a) 递进式检索确认页面



(b) 递进式检索结果

图 10 递进式检索功能展示

4.1.4 语义搜索

语义搜索通过挖掘输入问题背后的语义, 使搜索的结果更加符合用户的真实意图. 在进行语义搜索的过程中, 首先使用 LTP 分词工具对输入问题和定义好的问题模板进行分词, 通常分词工具对于专有领域中的实体和概念可能会出现过度划分等现象[22], 对后续搜索造成困扰, 因此本文通过增加自定义词典的方式来达到准确分词的目的. 然后, 分别将每一个问题模板和输入问题结合进行 One-Hot 编码, 得到此模板和输入问题的词向量表示, 通过计算两者词向量之间的余弦相似度, 选择相似性最高的作为该输入问题的类型. 最后将根据模板和问题关键词查询到的相关数据返回至前端界面进行整合显示.

图 11 为问题“郑秋梅老师和黄庭培老师共同发表的期刊论文情况”的搜索结果, 界面中通过图形绘制直观展示了两位老师共同发表的期刊论文和合作关系, 实现了语义搜索的目标.

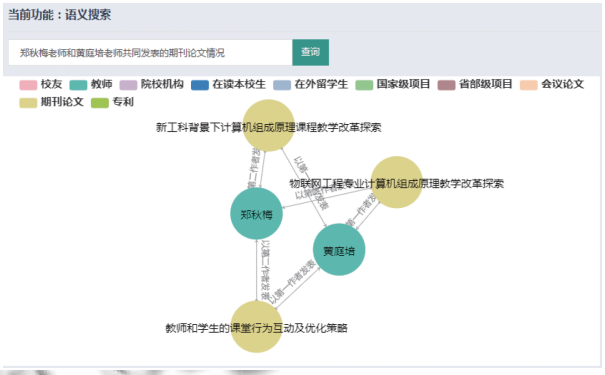


图 11 语义搜索结果

4.2 系统性能评估

为了验证本系统的性能能否满足用户实际使用的要求, 在学科知识图谱可视化系统构建完成后, 邀请了数十位高校计算机学科的相关人员进行测试. 测试过程根据测试方法的不同分为两组: 一组采用 α 测试方法, 即给予参与测试者一定的系统使用说明与操作指导, 主要是对可视化系统的可靠性进行验证; 另一组采用 β 测试方法, 让参与测试者在无任何指导和帮助的情况下自主探索系统的功能, 主要是对系统的健壮性和易用性进行验证. 最终 α 测试组的总体反馈为本系统的数据覆盖范围较为广泛, 交互友好, 可靠性强; β 测试组的总体反馈为系统易于上手和操作, 且无异常情况产生. 所有参与测试者的平均满意度达到 91.67%,

证明本可视化系统能够达到课题的研究目标要求.

同时本系统也作为辅助工具参与了第 5 轮学科评估材料的准备工作. 其中, 相关专家主要使用本系统对学科评估材料中的部分数据进行对比验证, 以及时发现材料中的错误内容. 这种工作模式不仅能够增加了评估材料的准确度, 而且加快了材料准备的速度, 提升了工作效率, 使得本系统在实际的应用场景中也取得了令人满意的效果.

5 结束语

本文就高校计算机学科领域进行研究, 给出了一套完整的领域知识图谱构建方案, 并通过实验结果证明了该方案的可用性. 针对多源异质的领域数据, 设计基于规则映射与改进网络爬虫相结合的数据获取方法, 然后使用 fine-tuning 后的 BERT 分类模型对数据进行清洗过滤. 对于不同来源知识的融合问题, 提出一种基于 Word2Vec 的实体对齐方法, 有效解决融合过程中的数据冲突问题. 最后将知识导入 Neo4j 图数据库进行存储, 并基于此知识图谱完成了计算机学科可视化系统的实现, 为以后的学科评估工作提供方便快捷的资源查询与关系展示等应用服务. 由于计算机学科的数据来源中还包括一些非结构化的数据, 后续工作中将完善有关非结构化文本的知识抽取方法, 使构建的学科知识图谱更加全面.

参考文献

1 黎晓玲. 教育部学科评估指标变迁及启示. 大学教育, 2020, (5): 1–3. [doi: 10.3969/j.issn.2095-3437.2020.05.001]

2 李涛, 王次臣, 李华康. 知识图谱的发展与构建. 南京理工大学学报, 2017, 41(1): 22–34. [doi: 10.14177/j.cnki.32-1397n.2017.41.01.004]

3 Barisevičius G, Coste M, Geleta D, *et al*. Supporting digital healthcare services using semantic Web technologies. Proceedings of the 17th International Semantic Web Conference. Cham: Springer, 2018. 291–306. [doi: 10.1007/978-3-030-00668-6_18]

4 乔钢柱, 冯婷婷, 张国晨. 基于知识图谱的盗窃案件法律文书智能推理研究. 计算机系统应用, 2019, 28(7): 206–213. [doi: 10.15888/j.cnki.csa.006974]

5 刘峤, 李杨, 段宏, 等. 知识图谱构建技术综述. 计算机研究与发展, 2016, 53(3): 582–600. [doi: 10.7544/issn1000-1239.2016.20148228]

6 章勇, 吕俊白. 基于 Protege 的本体建模研究综述. 福建电脑, 2011, 27(1): 43–45. [doi: 10.3969/j.issn.1673-2782.2011.01.021]

7 杨玉基, 许斌, 胡家威, 等. 一种准确而高效的领域知识图谱构建方法. 软件学报, 2018, 29(10): 2931–2947. [doi: 10.13328/j.cnki.jos.005552]

8 谢克武. 大数据环境下基于 Python 的网络爬虫技术. 电子制作, 2017, (9): 44–45. [doi: 10.16589/j.cnki.cn11-3571/tn.2017.09.017]

9 Fang T, Han T, Zhang C, *et al*. Research and construction of the online pesticide information center and discovery platform based on Web crawler. Procedia Computer Science, 2020, 166: 9–14. [doi: 10.1016/j.procs.2020.02.004]

10 Kim Y. Convolutional neural networks for sentence classification. arXiv: 1408.5882, 2014. [doi: 10.3115/v1/D14-1181]

11 刘春磊, 武佳琪, 檀亚宁. 基于 TextCNN 的用户评论情感极性判别. 电子世界, 2019, (3): 48, 50. [doi: 10.19353/j.cnki.dzsj.2019.03.020]

12 余传明, 王曼怡, 林虹君, 等. 基于深度学习的词汇表示模型对比研究. 数据分析与知识发现, 2020, 4(8): 28–40.

13 Jwa H, Oh D, Park K, *et al*. exBAKE: Automatic fake news detection model based on Bidirectional Encoder Representations from Transformers (BERT). Applied Sciences, 2019, 9(19): 4062. [doi: 10.3390/app9194062]

14 Li XY, Zhang H, Zhou XH. Chinese clinical named entity recognition with variant neural structures based on BERT methods. Journal of Biomedical Informatics, 2020, 107(5): 103422. [doi: 10.1016/j.jbi.2020.103422]

15 Lee JS, Hsiang J. Patent classification by fine-tuning BERT language model. World Patent Information, 2020, 61: 101965. [doi: 10.1016/j.wpi.2020.101965]

16 赵旸, 张智雄, 刘欢, 等. 基于 BERT 模型的中文医学文献分类研究. 数据分析与知识发现, 2020, 4(8): 41–49.

17 Sharma AK, Chaurasia S, Srivastava DK. Sentimental short sentences classification by using CNN deep learning model with fine tuned Word2Vec. Procedia Computer Science, 2020, 167: 1139–1147. [doi: 10.1016/j.procs.2020.03.416]

18 罗钰敏, 刘丹, 尹凯, 等. 加权平均 Word2Vec 实体对齐方法. 计算机工程与设计, 2019, 40(7): 1927–1933. [doi: 10.16208/j.issn1000-7024.2019.07.021]

19 Sun YH, Sarwat M. A spatially-pruned vertex expansion operator in the Neo4j graph database system. Geoinformatica, 2019, 23(3): 397–423. [doi: 10.1007/s10707-019-00361-2]

20 崔蓬. ECharts 在数据可视化中的应用. 软件工程, 2019, 22(6): 42–46. [doi: 10.19644/j.cnki.issn2096-1472.2019.06.011]

21 王鑫, 傅强, 王林, 等. 知识图谱可视化查询技术综述. 计算机工程, 2020, 46(6): 1–11. [doi: 10.19678/j.issn.1000-3428.0057669]

22 唐琳, 郭崇慧, 陈静锋. 中文分词技术研究综述. 数据分析与知识发现, 2020, 4(2): 1–17. [doi: 10.11925/infotech.2096-3467.2019.1059]