计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(11):79–90 [doi: 10.15888/j.cnki.csa.008799]
©中国科学院软件研究所版权所有.
E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于 LEBERT 的多模态领域知识图谱构建①

李华昱, 付亚凤, 闫　阳, 李家瑞

(中国石油大学 (华东) 计算机科学与技术学院, 青岛 266580)
通信作者: 付亚凤, E-mail: z20070061@s.upc.edu.cn

摘　要: 多模态知识图谱 (multi-modal knowledge graph, MMKG) 是近几年新兴的人工智能领域研究热点. 本文提供了一种多模态领域知识图谱的构建方法, 以解决计算机学科领域知识体系庞大分散的问题. 首先, 通过爬取计算机学科的相关多模态数据, 构建了一个系统化的多模态知识图谱. 但构建多模态知识图谱需要耗费大量的人力物力, 本文训练了基于 LEBERT 模型和关系抽取规则的实体-关系联合抽取模型, 最终实现了一个能够自动抽取关系三元组的多模态计算机学科领域知识图谱.

关键词: 多模态; 知识图谱; 领域; LEBERT; 关系抽取规则; Lexicon Adapter

引用格式: 李华昱,付亚凤,闫阳,李家瑞.基于 LEBERT 的多模态领域知识图谱构建.计算机系统应用,2022,31(11):79–90. http://www.c-s-a.org.cn/1003-3254/8799.html

Construction of Multi-modal Domain Knowledge Graph Based on LEBERT

LI Hua-Yu, FU Ya-Feng, YAN Yang, LI Jia-Rui

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

Abstract: Multi-modal knowledge graph (MMKG) is a new research hotspot in artificial intelligence in recent years. This study provides a construction method for multi-modal domain knowledge graphs to solve the problem that the domain knowledge system of computer science is large and decentralized. Specifically, a systematic MMKG is constructed by crawling the relevant multi-modal data of computer science. However, the construction of an MMKG needs a lot of manpower and material resources. In response, this study trains a model of joint extraction of entities and relations based on the LEBERT model and relation extraction rules and ultimately implements an MMKG of the computer science domain that can automatically extract relation triples.

Key words: multi-modal; knowledge graph (KG); domain; LEBERT; relationship extraction rules; Lexicon Adapter

知识图谱由谷歌公司提出. 2012 年 5 月 17 日, 谷歌发布知识图谱项目, 并宣布以此为基础构建下一代智能化搜索引擎[1]. 知识图谱将大量多源异构的文本数据规范化为关系三元组, 存储在知识图谱中, 并以网络的形式将知识体系展示给用户, 有效提升了搜索效率和质量. 知识图谱又分为通用知识图谱和领域知识图谱. 通用知识图谱强调广度, 以常识性知识为主, 适合作为搜索、推荐、问答等应用的知识支撑, 面向大众; 领域知识图谱强调深度, 通常面向某一领域, 用户多为领域专家[2]. 到目前为止, 包括医疗、军事、海洋、金融等在内的多个领域已逐步将知识图谱落实到实际应用中[3–6].

然而, 随着数字信息形式的多样化发展, 出现大量包含视频、图片、语音、文本等在内的多媒体数据, 人们越来越倾向于使用内容丰富的多媒体数据进行学习、展示、记录留存等[7], 传统的知识图谱已经无法完全满足用户的需求[2,8]. 多模态知识图谱是将多模态信息引入到知识图谱的一种技术[7], 它在研究文本关系三元组的基础上, 构建跨模态的实体以及语义关系, 极大

① 基金项目: 山东省自然科学基金面上项目 (ZR2020MF140); 中国石油大学 (华东) 研究生创新基金 (22CX04035A)
收稿时间: 2022-02-26; 修改时间: 2022-04-02; 采用时间: 2022-04-13; csa 在线出版时间: 2022-07-15

© 中国科学院软件研究所 http://www.c-s-a.org.cn

丰富了只包含文本信息的传统知识图谱.

计算机学科领域拥有完整的知识体系和丰富的科研成果, 但由于其表现形式的多样性, 不得不将它们分散存储在不同的数据库中. 这种分散式的存储对于知识呈现的完整性造成了一定程度的损失. 因此, 本文通过构建多模态知识图谱, 对计算机学科领域的知识进行系统化的梳理, 在传统关系三元组的基础上, 辅以图片信息, 形成多模态的计算机学科知识网络.

本文的主要贡献如下.

1) 构建了一个包含文本、图片两种模态的计算机学科领域的数据集, 并最终在该数据集的基础上完成了计算机学科领域多模态知识图谱的构建.

2) 在知识图谱构建的实体-关系联合抽取任务中, 使用了引入 Lexicon Adapter 的 LEBERT 模型, 并叠加 BiLSTM-CRF 模型, 在文本构建的数据集中取得了良好的效果.

3) 设计了一种基于 BERT 预训练语言模型和余弦相似度的实体消歧方法.

本文第 1 节是多模态知识图谱的相关工作, 第 2 节是计算机学科领域本体的构建, 第 3 节是多模态知识图谱的构建, 介绍了实体抽取、关系抽取、实体消歧及知识存储部分, 第 4 节是实验部分, 介绍了命名实体识别、关系抽取等相关实验以及可视化系统展示, 第 5 节是结束语.

1 相关工作

21 世纪初, 万维网诞生, 图片数据大量涌现, 多媒体数据的搜索问题逐渐显现. 由于图片搜索的准确性较低, 当时的专家学者大都采用了为图片进行文本标注并建立知识库存储的策略, 以提高搜索质量, 多模态知识图谱的早期构建方法由此发展起来. 多模态知识图谱发展过程如图 1 所示.

图 1 多模态知识图谱的发展

2004 年, 美国卡内基梅隆大学 von Ahn 等人[9] 提出的“labeling images with a computer game”首次将图片与文本结合, 以电脑游戏的形式为图片添加正确的文字标签. 2006 年, 万维网创始者 Bizer 等人[10] 提出 Linked Data. 2009 年, Hausenblas 等人[11] 提出 IM, 将 Linked Data 应用于多媒体数据的互连; 美国普林斯顿大学提出的 ImageNet[12] 通过互联网搜索和众包构建实现, 它是一个基于 WordNet结构的大规模图像本体. 2013 年, Chen 等人[13] 提出 NEIL, 使用半监督学习算法, 标记给定类别的实例, 它试图用最少的人力开发世界上最大的可视化结构化知识库, 截止 2013 年, NEIL 已经标记了 40 多万个视觉实例[13]. 2014 年正式发布的 Wikidata[14] 中也存在大量的多模态资源, 它提供了一个可由所有人共享的免费协作知识库, 已经成为维基媒体最活跃的项目之一. 2017 年, 智利大学 Ferrada 等人[15] 提出 IMGpedia, IMGpedia 是一个大型的链接数据集, 它从 Wikimedia Commons 数据集的图像中收集大量的可视化信息, 构建并生成了 1 500 万个视觉内容描述符, 图像之间有 4.5 亿个视觉相似关系[15]. 2019 年, Liu 等人[16]提出的 MMKG 是一个包含所有实体的数字特征和图像的 3 个知识图谱的集合, 大量实验验证了 MMKG 在同一链路预测任务中的实用性; 东南大学 Wang 等人[17] 提出的 Richpedia, 将图片作为单独的实体进行存储, 并设计了 3 个基于自然语言处理和语法分析的关系抽取规则, 以从图片中获取实体之间的关系. 上述为通用多模态知识图谱的发展历程.

© 中国科学院软件研究所 http://www.c-s-a.org.cn

随着通用多模态知识图谱的发展, 越来越多的领域专家开始尝试将多模态技术应用于领域知识图谱, 例如多模态教学知识图谱[2]、多模态医学知识图谱[18]等. 然而, 这些工作并不支持实体-关系三元组的自动识别、抽取等, 而是依靠大量人工完成. 这种多模态领域知识图谱的构建方法难以应用至其他领域. 为了减少构建多模态领域知识图谱的人工投入, 为其他领域提供一种通用性较强的多模态知识图谱构建方法, 本文以计算机学科领域为例, 提出了一种较为完整的多模态领域知识图谱构建流程, 包括领域本体构建、数据获取、实体-关系抽取、实体对齐、知识存储以及可视化系统实现等步骤.

2 学科本体构建

1993 年, Gruber 等人[19]将本体定义为“一种概念化的精确的规格说明”. 1998 年, Studer 等人[20]将本体的概念扩充为“共享概念模型的明确形式化规范说明”. 总之, 本体主要是用来描述某个领域内的概念和概念之间的关系, 使这些概念和关系具有明确、唯一的定义. 本体作为一种重要的知识库, 它包含丰富的语义信息, 可以为问答系统、信息检索、语义 Web、信息抽取等领域的研究及相关应用提供重要的支持[21]. 本体一般由概念、概念间的关系以及建立在关系之上的公理这三部分组成, 概念又称为类. 根据描述的目标范围, 本体可以分为通用本体和领域本体. 不同于通用本体, 领域本体具有显著的领域特性, 通常对某个特定领域建立相应的知识规范描述. 本体中具有丰富的概念和关联关系, 因此通常使用规范化的语言对本体进行描述, 常见的本体描述语言有: RDF、OWL、Loom 等[22,23], 本文使用 OWL 语言对计算机学科领域本体进行描述.

本文采用自顶向下的方法构建计算机学科领域本体, 如图 2 所示. 本体中包含人物、论文、专利、组织、地点、职务、职称、学术会议、计算机类期刊、时间等 10 个类. 其中, 专利类别包含实用新型专利、外观设计专利、发明专利等 3 个子类; 组织类别包含企业单位、科研机构、高等院校等 3 个子类; 职务类别包含公司员工、教师、期刊主编、学生等 4 个子类.

计算机学科领域本体中的概念具有丰富的关联关系, 如表 1 所示, 共定义了主编、举办地点、任职于、职务、职称、位于、发明、申请、申请日、公开日、发表、主办、刊载等 13 种关系.

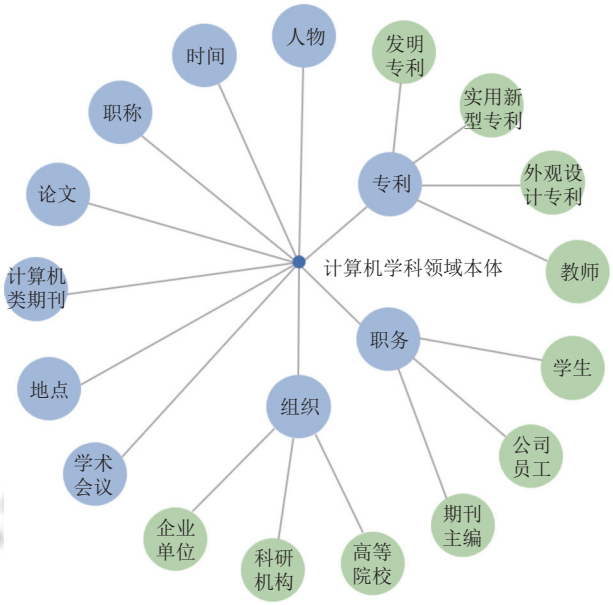

图 2 计算机学科领域本体

表 1 概念关联关系表

Domain	ObjectProperty	range
人物	主编	期刊
学术会议	举办地点	地点
人物	任职于	组织
人物	职务	职务
人物	职称	职称
组织	位于	地点
人物	发明	专利
组织	申请	专利
专利	申请日	时间
专利	公开日	时间
人物	发表	论文
组织	主办	学术会议
论文	刊载	期刊

以“<人物, 发明, 实用新型专利>”关系三元组为例, OWL 语言描述如下:

```
<owl:Class rdf:ID="人物"/>
<owl:Class rdf:ID="专利"/>
<owl:Class rdf:ID="实用新型专利">
<rdfs:subClassOf rdf:resource="#专利/">
</owl:的 Class>
<owl:ObjectProperty rdf:ID="发明">
<rdfs:domain rdf:resource="#人物/">
<rdfs:range rdf:resource="#实用新型专利/">
</owl:ObjectProperty>
```


© 中国科学院软件研究所 http://www.c-s-a.org.cn

本文旨在提供一种多模态学科领域知识图谱的构建方法, 本体中未能包含计算机学科领域中的全部概念和关联关系, 仅针对部分常见的概念进行领域本体的建模.

3 多模态领域知识图谱构建

本文中多模态领域知识图谱的构建步骤包括知识获取、知识抽取、实体链接、知识存储等. 领域知识图谱的构建需要大量语料, 本文采用 Python 网络爬虫对计算机学科领域数据进行爬取, 经过数据清洗、预处理后, 将数据划分为训练集、测试集和验证集, 用于后续知识图谱的构建. 知识抽取是从不同来源、不同结构的数据中进行知识提取, 通常包括实体抽取、关系抽取、属性抽取等, 知识抽取结果往往以关系三元组或属性三元组的形式存储到数据库中. 实体链接包括实体消歧和共指消解, 用于解决知识图谱中一词多义与多词同义的问题, 是知识图谱构建过程中必不可少的步骤. 多模态计算机学科领域知识图谱的构建流程如图 3 所示.

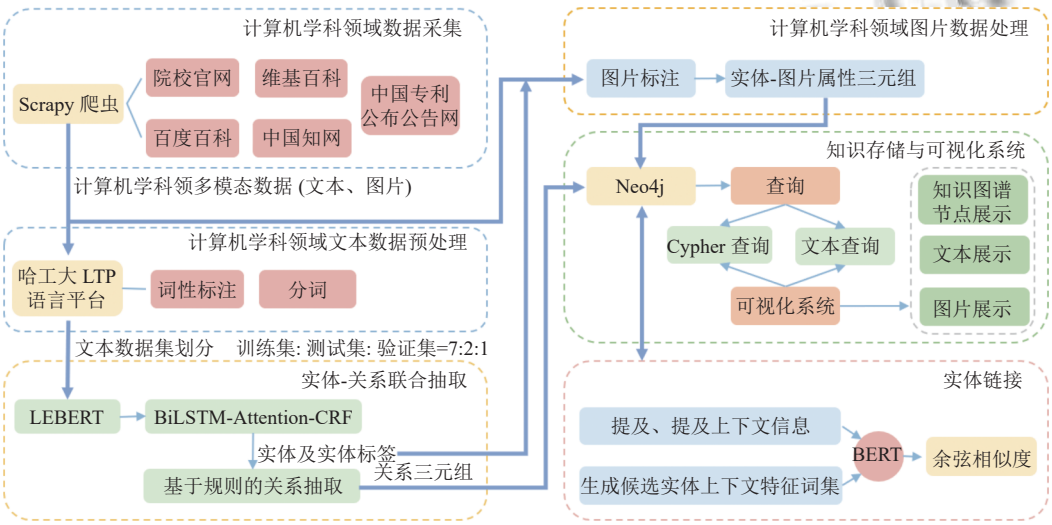

图 3 多模态计算机学科领域知识图谱构建流程图

3.1 知识获取与预处理

3.1.1 知识获取

计算机学科领域的多模态知识图谱数据来源主要为院校官方网站、百度百科、维基百科、中国知网、中国专利公布公告网等多个官方信息网站. 由于目标数据量较大, 数据的获取主要围绕北京航空航天大学、华中科技大学、南京大学、清华大学、山东大学、天津大学、中国石油大学 (北京)、中国石油大学 (华东) 等 8 所院校的相关领域专家和学生, 爬取的内容为人物、组织、论文、专利、期刊、会议, 以及与其相关的多模态属性信息. 文本数据的获取使用 Python 的 Scrapy 框架[24], 文本数据的爬取以知网论文为例, 爬取流程如图 4 所示. 本文将获取到的列表数据和详情页数据存储到 txt 文档中.

在定义好的计算机学科领域本体中, 人物、计算机类期刊、组织、学术会议等 4 个类别具有较为准确具体的图片信息, 因此在爬取这 4 个类别文本信息的同时需要获取相关的图片信息. 首先收集包含这 4 个类别图片信息的网页 URL, 将它们存储在 txt 文档中, 并使用爬虫进行爬取图片及图片周围的文本. 为了便于后续操作, 本文将文本中符合图片信息且为当前类别实例的字段放置在首项, 例如, 在爬取教师信息时, 教师的姓名字段作为 txt 文档每一行的第一项进行存储.

最终获取到的计算机学科数据, 包含半结构化的列表字段、非结构化的纯文本, 共 22 393 条, 相关图片 4 017 张, 详细信息见表 2.

此外, 为了尽量避免后续构建知识图谱时出现实体歧义问题, 在抽取数据时对数据中出现组织、职称类实体的简称以及缩写进行全称替换处理.

3.1.2 数据预处理

本文的数据预处理包括文本数据预处理和图片数据预处理.

文本数据预处理主要包括语料清洗、中文分词、词性标注等步骤. 语料清洗包括: 删除不相关数据和重

© 中国科学院软件研究所 http://www.c-s-a.org.cn

复数据;去除乱码与多余的符号. 文本数据清洗完成后, 使用哈尔滨工业大学社会计算与信息检索研究中心开发的语言技术平台 LTP 进行中文分词、词性标注等步骤. 最终将预处理后的文本数据按照训练集:测试集:验证集=7:2:1 的比例, 进行数据集划分, 并分别命名为“train.txt”“test.txt”“dev.txt”.

图片数据的预处理主要是为图片标注合适的文本标签, 即图片对应的实体名称, 方便后续为实体添加图片属性. 数据爬取时, 图片链接和与图片相关的字段信息按类别存储在不同的 txt 文件中, 本文选取每行的首项作为图片的文本标签, 由于该字段为计算机学科领域本体中已存在的类别的实例, 在后续为实体添加图片属性的操作中, 可直接将该字段和图片链接抽取为属性三元组的形式, 便于向数据库中存储.

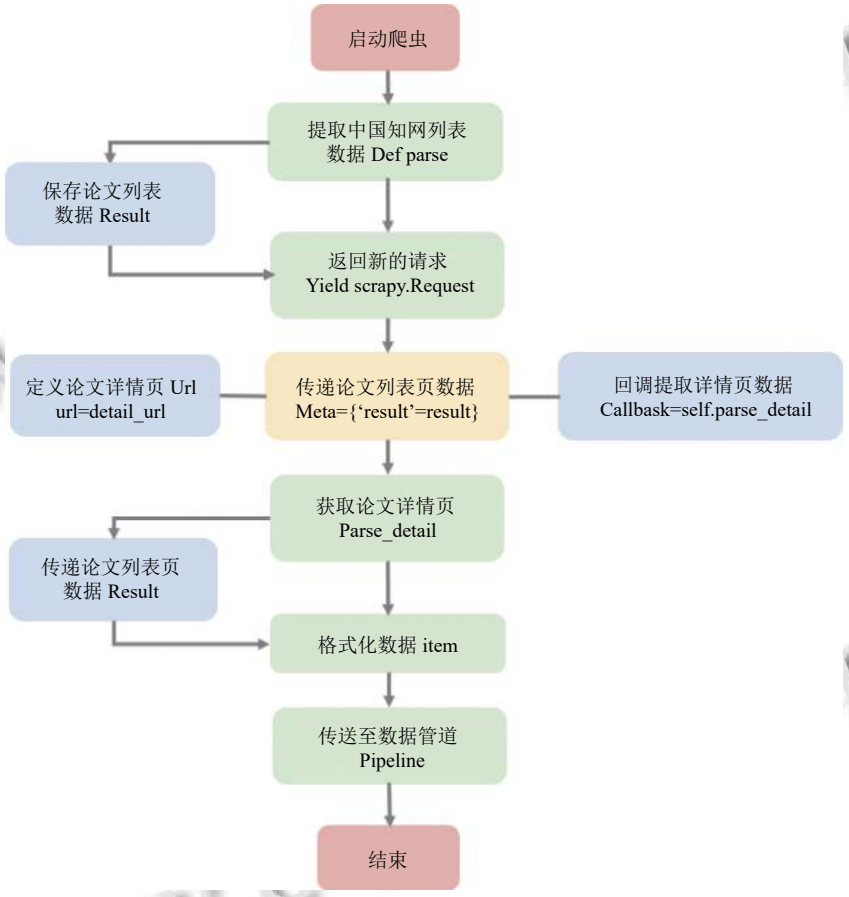

图 4 基于 Scrapy 框架的数据采集流程图

表 2 原始语料分类详情表

数据分类	文本数量	图片数量
计算机学科论文	5114	0
论文作者信息	7913	0
院校信息	2817	2817
计算机类专利	5349	0
计算机高等学院教师信息	754	754
计算机类期刊	188	188
学术会议	258	258

3.2 实体-关系联合抽取

本文将基于深度学习的命名实体识别模型和基于规则的关系抽取模型相结合, 进行句子级的实体-关系联合抽取. 命名实体识别作为自然语言处理的基本任务, 旨在从非结构化文本中识别具有特定含义的实体, 如人名、地名和组织[25]. 命名实体识别任务的输入是一个序列, 模型的输出是输入序列的标签序列. 关系定义为两个或多个实体之间的某种联系, 关系抽取任务的输入为一段文本, 输出通常是一个三元组 (实体 1, 关系, 实体 2). 实体-关系联合抽取的输出包括每个句子中的实体、实体类型, 以及从句子中抽取的关系三元组. 实体-关系联合抽取模型如图 5 所示.

© 中国科学院软件研究所 http://www.c-s-a.org.cn

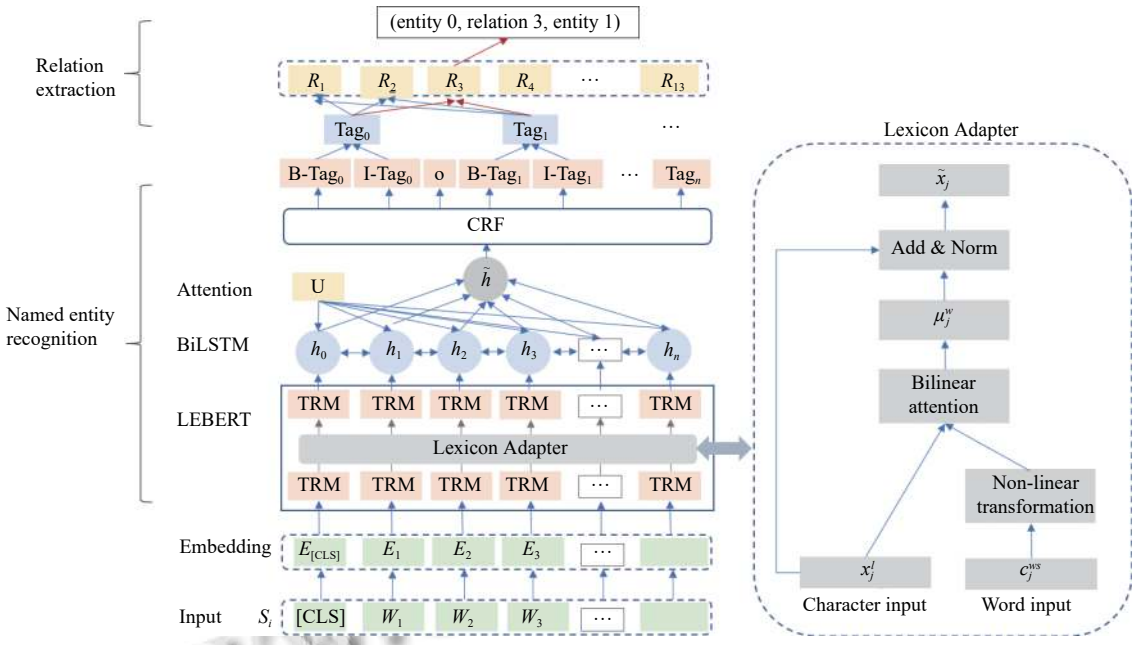

图 5 LEBERT-BiLSTM-Attention-CRF 框架图

该模型的初始输入为句子集合$S=\{s_1,s_2,\cdots,s_m\}$, 其中, s_i表示句子集中的第 i 个句子, m 表示句子的总个数; 每个句子由多个 token 组成, 即$s_i=\{w_1,w_2,\cdots,w_n\}$, w_j表示句子集中的第 j 个 token, n 表示一个句子中的 token 个数. 句子集合首先输入到 LEBERT 中得到每个句子的向量表示$e_i=\{E_{[CLS]},E_1,E_2,\cdots,E_n\}$. LEBERT 的输出向量作为 BiLSTM 的输入, BiLSTM 的隐藏层向量$h=\{h_0,h_1,h_2,\cdots,h_n\}$, 向量$h$与 Attention 机制的权重矩阵 U 进行加权计算后进行拼接, 拼接后的向量矩阵$\tilde{h}$输入 CRF 层进行解码, 最终得到每个 token 的对应标签 Tag. 得到实体及其相应的实体类别标签后, 根据预先定义的 13 种关系抽取规则, 将具有关联关系的实体对抽取为关系三元组的形式.

3.2.1 LEBERT 预训练语言模型

2018 年 10 月, Devlin 等人[26] 提出 BERT 模型, 该模型在 11 项自然语言处理任务中取得 SOTA. LEBERT 模型[27] 在 BERT 模型的某两层 Transformer[28] 之间, 加入了词典适配器 Lexicon Adapter, 以增强特征信息. 不同于 Sun 等人[29] 在 BERT 模型与其他模型之间引入特征词信息, LEBERT 模型在 BERT 模型内部的某两层 Transformer 之间引入特征词典适配器. 如图 6 所示, LEBERT 由 Transformer 编码器单元和词典适配器 Lexicon Adapter 组成.

1) Transformer

LEBERT 的基本单元由 Transformer 编码器组成, 如图 7 所示, Transformer 编码器由多头自注意力机制层和前馈神经网络构成. 嵌入向量输入编码器单元后, 首先进入多头自注意力机制层, 经过残差连接和归一化操作后, 输入到前馈神经网络中, 再经过一层残差连接[30] 和归一化操作后输入到下一个 Transformer 编码器单元中.

2) Lexicon Adapter

特征词典适配器 Lexicon Adapter 由特征词向量、双线注意力机制组成.

如图 6 所示, x_j^l是字符w_j在 BERT 模型中第l层 Transformer 的输出, c_j^{ws}为与字符w_j匹配的特征词向量. x_j^l与经过非线性变换的特征词向量c_j^{ws}通过双线性注意力机制层后计算得到增强的特征向量, 将计算得到的向量与x_j^l进行求和归一化操作后, 得到向量$\tilde{x}_j$, 最后将向量$\tilde{x}_j$输入到第$l+1$层 Transformer 中继续训练. LEBERT 将与训练语料相关的大量特征词直接融入 BERT 的训练过程, 有效增强了训练语料的特征向量. Liu 等人[27] 使用 CTB、MSRA 等多个数据集实验证明, 特征词的融入有效提高了 BERT 模型在命名实体识别任务中的准确率.

① 特征词词典构造

在本文构建的计算机学科领域本体中, 论文、计算机类期刊、学术会议、专利等概念包含大量具有鲜明计算机学科领域特点的实例. 因此, 为了增强 BERT

© 中国科学院软件研究所 http://www.c-s-a.org.cn

的训练效果, 特征词词典中的词组必须包含具有领域针对性的专业名词. 考虑到搜集的特征词是否能够有效强化语料特征的问题, 本文通过半自动化的方式, 从已爬取的计算机学科领域数据中提取相应的特征词放入特征词词典 D 中, 并通过 Word2Vec 模型转化为词向量, 共计 1 712 个特征词. 为了方便后续的字-词匹配操作, 将特征词以前缀树的形式存储, 记为 T.

② 字符-特征词匹配

给定计算机学科领域特征词前缀树 T 和一个包含 n 个字符的句子 $s_i=\{w_1,w_2,\cdots,w_n\}$. 首先遍历句子的所有字符子序列, 将它们与前缀树 T 进行匹配, 获得所有潜在可能配对的词, 例如输入句子“计算机网络”可以匹配到 “计算”“计算机”“计算机网络”“网络”等 4 个特征词, 将这 4 个特征词分别分配给它们包含的句子中的字符. 如图 8 所示, “计算”分配给“计”和“算”, “计算机”分配给“计”“算”和“机”, 计算机网络分配给“计”“算”“机”“网”“络”, “网络”分配给“网”“络”. 可以通过设置阈值限制每个字符被分配的特征词数量.

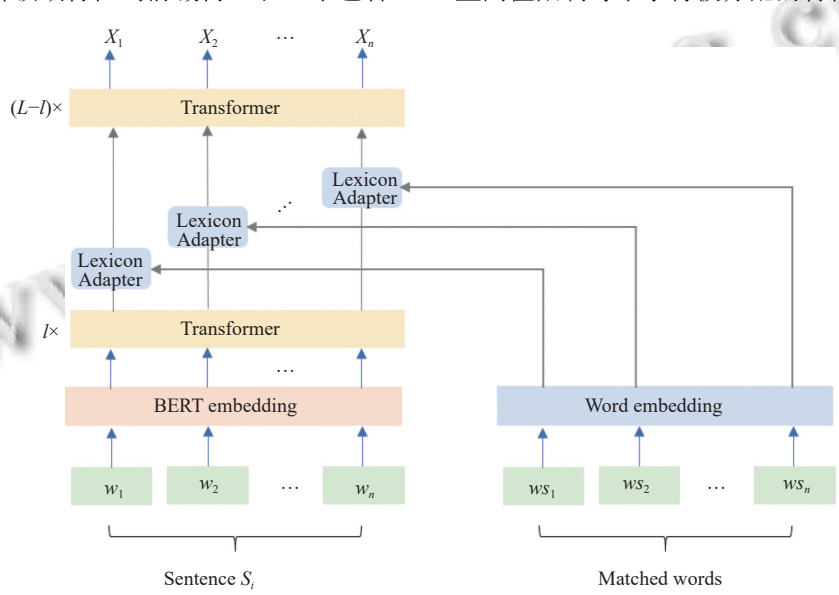

图 6 LEBERT 框架

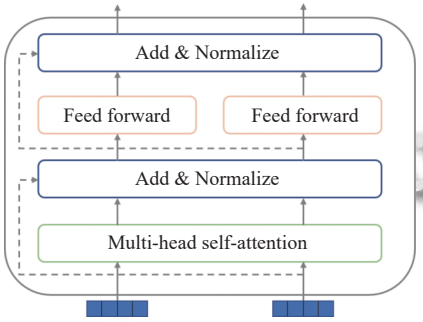

图 7 Transformer 编码器

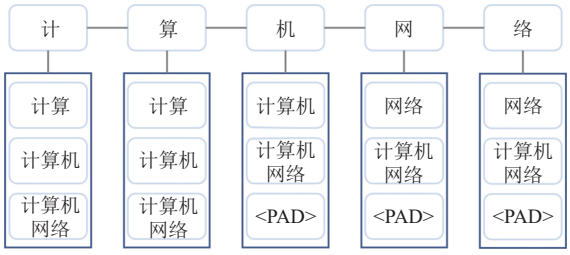

图 8 字符-特征词匹配示例

③ 双线性注意力机制

字符与特征词匹配成功后, 对于每一个位置的信息, 就由单个字符转变为字符及其对应的特征词组. 如图 8 所示, 为了处理两种不同类型的数据, 首先需要对特征词组向量 $c_j^{ws}=\{c_{j1}^w,c_{j2}^w,\cdots,c_{jp}^w\}$ 进行一次非线性转换. 非线性转换包含两层线性层和tanh激活函数, 如式 (1):

$$v_{jk}^w=W_2(\tanh(W_1c_{jk}^w+b_1))+b_2 \tag{1}$$

其中, W_1、W_2 为参数矩阵, b_1、b_2 为常量, c_{jk}^w 为字符 w_j 对应的特征词组中第 k 个特征词的词向量. 记 $v_j=(v_{j1}^w,v_{j2}^w,\cdots,v_{jp}^w)$. 为了确定特征词组中每个特征词的重要程度, 引入了双线性注意力机制, 如式 (2):

$$\partial_j=Softmax(x_j^lW_{\text{att}}v_j^{\text{T}}) \tag{2}$$

其中, W_{att} 为双线性注意力机制的权重矩阵. 得到权值矩阵 ∂_j 后, 将其与特征向量矩阵点乘, 作为双线性注意力机制层的最终输出, 最后与当前字符向量求和:

© 中国科学院软件研究所 http://www.c-s-a.org.cn

$$\mu_j^w = \sum_{k=1}^{p} \alpha_{jk} v_{jk}^w \tag{3}$$

$$\tilde{x}_j = x_j^l + \mu_j^w \tag{4}$$

向量$\tilde{x}_j$经过归一化操作后, 作为第$l+1$层 Transformer 的输入, 代替向量x_j^l继续训练.

3.2.2 BiLSTM-CRF 层

本文在 LEBERT 的基础上叠加 BiLSTM-CRF 模型进行命名实体识别任务. BiLSTM 模型将前向 LSTM 与后向 LSTM 的隐藏层向量拼接, 充分利用当前 token 的上下文特征信息, 以得到更加准确的预测结果. CRF 可以学习连续标签之间的约束, 以输出概率最大、整体最优的标签序列, 降低出现不合理标注的概率, 提高实体识别结果的准确率. BiLSTM 结构如图 9 所示, 由前向 LSTM 和后向 LSTM 组成, 前向 LSTM 的隐藏层的输出向量$h_R = \{h_{R1}, h_{R2}, \cdots, h_{Rn}\}$和后向 LSTM 的隐藏层的输出向量$h_L = \{h_{L1}, h_{L2}, \cdots, h_{Ln}\}$拼接为$h = \{h_1, h_2, \cdots, h_n\}$, 其中, $h_t = [h_{Rt} \oplus h_{Lt}]$, h_t代表 t 时刻 BiLSTM 的隐藏层向量.

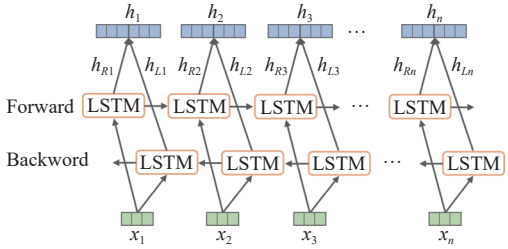

图 9 BiLSTM 框架

3.2.3 基于规则的关系抽取

由于本文采集的数据包含关系较少且关系词单一, 所以采用基于规则模板的方法, 使用计算机学科领域本体中类间的关系作为关系三元组中的关系词进行关系抽取. 关系抽取步骤如下.

1) 根据计算机学科领域本体中的概念和概念间的关联关系定义 13 个规则模板, 如表 3 所示.

2) 本文为实体-关系联合抽取, 若在一句话中出现具有对应规则的实体对, 则将实体对与对应关系组成关系三元组并存储到 Neo4j 数据库中.

3) 若一句话中出现 3 个及以上的实体, 创建数组并以实体类型命名, 暂时存储实体; 首先对所有实体类型进行两两组合, 若两种实体类型具有对应的规则, 则将对应数组中的实体进行组合, 否则不进行组合. 组合公式如式 (5):

$$C_{n_1}^1 C_{n_2}^1 = \frac{1}{n_1!(1-n_1)!} \cdot \frac{1}{n_2!(1-n_2)!} \tag{5}$$

其中, n_1是第 1 种实体类型的实体数量, n_2是第 2 种实体类型的实体数量. 本文数据集中的句子不包含复杂结构的句式, 因此不会出现具有对应关系的实体对不存在关系的情况.

表 3 关系三元组抽取规则定义表

关系类型	规则
主编	主编: (PER, PRD)
举办地点	举办地点: (MEETING, LOC)
任职于	任职于: (PER, ORG)
职务	职务: (PER, JOB)
职称	职称: (PER, POS)
位于	位于: (ORG, LOC)
发明	发明: (PER, PATENT)
申请	申请: (ORG, PATENT)
申请日	申请日: (PATENT, TIME)
公开日	公开日: (PATENT, TIME)
发表	发表: (PER, PAPER)
主办	主办: (ORG, MEETING)
刊载	刊载: (PAPER, PRD)

3.3 实体链接

实体链接的目的是将新实体与知识库中对应的实体进行链接, 补充知识图谱的内容, 用以解决实体歧义性和多样性问题. 实体的歧义性和多样性主要表现在两个方面: 一词多义、多词同义. 一词多义指一个词语可以指代多个实体, 在本文使用的数据集中, 人物类实体最容易出现一词多义问题, 例如不同的人使用相同的人名; 多词同义指多个词语指代同一个实体, 例如“石大”“中石大”等均可指代组织类实体中国石油大学(华东). 本文数据爬取自官方网站, 实体名称较统一, 多词同义问题较少, 主要针对一词多义问题进行实体链接. 本文基于 BERT 预训练语言模型和余弦相似度设计的实体链接算法如下.

算法 1. BCOSEL()

输入: 实体 x, 上下文信息 $text\_x$
输出: 0 或 1

1. if 数据库中存在实体 e, 且 $e.name$=$x.name$ then
2. 生成词集 e_SET={$e.name$, 实体 e 的相邻关系, 实体 e 的属性}
3. X_1 = BERT($text\_x$) //用 BERT 模型生成向量
4. X_2 = BERT(e_SET)
5. $Similarity$=cos(X_1,X_2) // 计算余弦相似度
6. if $Similarity$≥相似度阈值 then
   为实体 e 添加实体 x 的关系和属性

© 中国科学院软件研究所 http://www.c-s-a.org.cn

7. 返回 1
8. else
9. 向数据库中添加实体 x
10. 返回 0
11. else
12. 向数据库中添加实体 x
13. 返回 0

余弦相似度计算公式如式 (6) 所示:

$$Similarity = \cos(A,B) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} \quad (6)$$

若经过实体链接后, 发现知识库中已有该实体, 则将其新的属性信息添加到知识库中; 若知识库中没有该实体, 则将该实体及其属性信息一起添加到知识库中.

3.4 知识存储

知识图谱以图的形式展现实体、关系以及实体的属性, 本文使用的 Neo4j 数据库是一个高性能的 NoSQL 图形数据库, 以网络的形式存储结构化数据, 且具有较好的可视化界面, 可以较好地利用知识图谱的图形结构信息. 本文使用 Cypher 语言将关系三元组和属性三元组导入 Neo4j 数据库中. 在进行关系三元组存储操作时, Neo4j 数据库会自动为每一个实体设置唯一标识的 ID, 在整个数据库中, 节点的 ID 值是递增的和唯一的. Neo4j 中实体添加、关系创建以及属性添加等操作的相关 Cypher 语句如表 4 所示.

表 4 Neo4j 操作语句表

操作	Cypher语句
创建实体	CREATE (n:实体类型{name:'实体名称'}) RETURN n
创建关系	MATCH (a:实体类型), (b:实体类型) WHERE a.name = '实体名称' AND b.title = '实体名称' CREATE (a) - [r:关系名称] -> (b) RETURN r
添加实体图片属性	MATCH (n) WHERE n.name = ××××× SET n.image = '图片链接' RETURN n

4 实验分析

4.1 实体-关系联合抽取

4.1.1 语料标注

本文需要大量的标注语料进行命名实体识别模型的训练工作. 训练语料的标注由人工完成. 常用于命名实体识别的标注策略有 BIO、BIOE、BIOES、BMEWO 等模式. 本文采用 BIO 模式, 将每个字符标注为“B-X”“I-X”或“O”. 其中, “B-X”表示此字符所在的片段属于 X 类型并且此元素在该实体的开头位置, “I-X”表示此字符所在的片段属于 X 类型并且此元素在该实体的中间位置, “O”表示不属于任何类型. 本文根据计算机学科领域本体中定义的概念设置实体类型及标签, 标注的实体类型及其对应标签见表 5, 使用空行作为句子间隔, 共计 10 类实体, 21 种标签.

表 5 BIO 实体标签

概念	实体类别	标签
人物	PER	B-PER、I-PER
地点	LOC	B-LOC、I-LOC
组织	ORG	B-ORG、I-ORG
职称	POS	B-POS、I-POS
职务	JOB	B-JOB、I-JOB
计算机类期刊	PRD	B-PRD、I-PRD
学术会议	MEETING	B-MEETING、MEETING
论文	PAPER	B-PAPER、I-PAPER
专利	PATENT	B-PATENT、I-PATENT
时间	TIME	B-TIME、I-TIME
其他	其他	O

训练集标注 754 213 字符, 测试集标注 212 341 字符, 验证集标注 103 885 字符, 累计 1 070 439 行. 共标注 34 996 个实体.

4.1.2 模型训练与评估

1) 实验环境与参数设置

命名实体识别模型试验环境如表 6 所示.

表 6 实体-关系联合抽取实验环境说明表

配置	版本
操作系统	Ubuntu 18.04
CPU	Intel (R) Xeon (R) Silver 4210 CPU @ 2.20 GHz
GPU	NVIDIA GeForce RTX 2080 Ti (11 GB)
Python	3.7
TensorFlow	2.3.1
Transformer	3.4.0
Torch	1.6.0+cu92

© 中国科学院软件研究所 http://www.c-s-a.org.cn

本文使用的主要参数包括: LEBERT 基于用于中文自然语言处理任务的 BERT 模型, 共包含 12 层 Transformer, lstm_dim 设置为 128, 最大序列长度设置为 128, 学习率设置为 1E–5, batch_size 设置为 12, dropout_rate 设置为 0.5, clip 设置为 0.5.

2) 评估指标

命名实体识别领域中常用精确率 (*Precision*)、召回率 (*Recall*)、*F*1 值 (*F*1) 作为评价模型识别性能的指标, 计算公式如下:

$$Precision = \frac{TP}{TP+FP} \times 100\% \tag{7}$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \tag{8}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \times 100\% \tag{9}$$

其中, *TP* 表示正类判定为正类, *FP* 表示负类判定为正类, *FN* 表示正类判定为负类.

3) 实验结果

① 命名实体识别

基于 LEBERT-BiLSTM-Attention-CRF 的命名实体识别任务中, 10 类实体的精确率均值达到 94.72%、召回率均值达到 94.92%、*F*1 均值达到 95.02%, 如表 7 所示.

为了验证实验模型的效果, 选取 LEBERT-BiLSTM-CRF、BERT-BiLSTM-Attention-CRF 在相同的环境下进行对比实验, 数据集为经过人工标注的计算机学科领域的文本数据集. 实验结果如表 8 所示. LEBERT-BiLSTM-Attention-CRF 的 *F*1 值最高.

表 7 命名实体识别实验结果 (%)

实体类别	*Precision*	*Recall*	*F*1
PER	97.81	97.93	97.87
LOC	91.92	92.13	92.01
ORG	95.44	95.26	95.35
POS	96.41	97.22	98.81
JOB	99.23	98.11	98.67
PRD	95.46	96.68	96.07
MEETING	90.52	91.66	91.09
PAPER	93.17	93.53	93.35
PATENT	92.94	92.36	92.65
TIME	94.33	94.33	94.33
平均值	94.72	94.92	95.02

为了验证 LEBERT 与 BERT 在本数据集中的特征提取效果, 选取 LEBERT-BiLSTM-Attention-CRF、BERT-BiLSTM-Attention-CRF 进行对比实验. 实验结果表明, 在本数据集中, LEBERT 的特征提取效果明显好于 BERT, LEBERT-BiLSTM-Attention-CRF 模型比 BERT-BiLSTM-Attention-CRF 模型的 *F*1 值高出了 3.54%.

表 8 各个模型的命名实体识别实验结果 (%)

模型	*Precision*	*Recall*	*F*1
LEBERT-BiLSTM-Attention-CRF	**94.72**	**94.92**	**95.02**
LEBERT-BiLSTM-CRF	94.11	93.95	94.03
BERT-BiLSTM-Attention-CRF	91.22	91.73	91.48

为了验证 Attention 在本数据集中的特征提取效果, 选取 LEBERT-BiLSTM-Attention-CRF、LEBERT-BiLSTM-CRF 进行对比实验. 实验结果表明, 在本数据集中, 引入 Attention 机制后, 特征提取效果有所提升, LEBERT-BiLSTM-Attention-CRF 模型比 LEBERT-BiLSTM-CRF 模型的 *F*1 值高出了 0.99%.

② 关系抽取

训练集标注 40 907 个关系三元组, 测试集标注 11 687 个关系三元组, 验证集标注 5 485 个关系三元组, 累计共 58 439 个关系三元组. 经过人工核查统计, 各类关系抽取的正确率如表 9 所示.

其中, 关系类型“公开日”“申请日”的正确率偏低. 经分析, 由于关系类型“公开日”“申请日”的头实体均为实体类型“PATENT”, 尾实体均为实体类型“TIME”, 此外, 实体类型“TIME”均为统一格式“××年××月××日”, 以上原因导致两类关系极易混淆, 从而出现评估指标偏低的情况.

表 9 关系抽取正确率 (%)

关系类别	Accuracy
主编	99.36
举办地点	95.55
任职于	98.21
位于	96.51
发明	99.14
发表	98.89
申请	98.60
职务	99.26
职称	98.76
公开日	61.71
申请日	66.58
主办	99.32
刊载	98.65
平均值	93.12

4.2 可视化系统实现

Neo4j 中可以使用 Cypher 语句进行实体、关系查

© 中国科学院软件研究所 http://www.c-s-a.org.cn

询, 但无法展示图片信息. 为了更好地展示相关图片数据, 本文采用 B/S 模式, 使用 Python 语言编写后台连接 Neo4j 数据库, 并基于 D3.js 设计实现可视化界面. 查询 Neo4j 数据库中人物类实体“张培颖”及其关联关系的结果如图 10 所示. 本文将查询得到的结果表示为网络的形式, 以不同颜色的圆代表不同类别的节点, 箭头代表关系.

通过点击节点, 可以获得该节点的属性信息, 如图 10 中, 点击节点“中国石油大学 (华东)”后, 页面右侧会出现相应的文本属性和图片属性. 图片的展示依赖于对图片本地链接的访问.

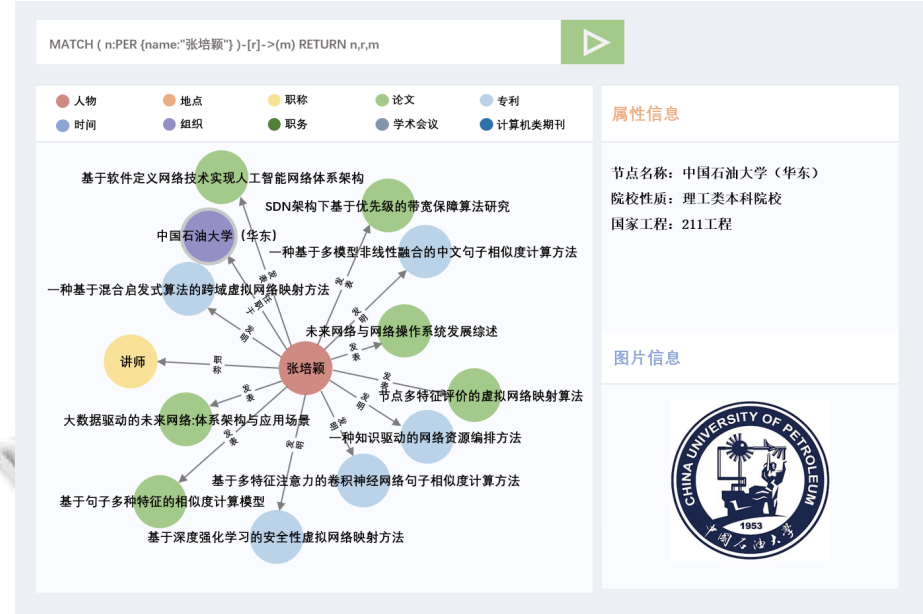

图 10 知识图谱实体查询结果展示

最终构建完成的知识图谱中共包含 34 996 个实体, 58 439 个关系, 109 005 个属性. 其中, 共包含 4 017 个图片属性.

5 结束语

本文构建了一个计算机学科领域的多模态数据集作为训练语料, 将 LEBERT 模型与 BiLSTM-CRF 结合用于训练语料的命名实体识别任务, 并在此基础上定义关系三元组的抽取规则, 实现了实体-关系联合抽取. 针对数据中可能出现的实体歧义问题, 设计了一种基于 BERT 模型和余弦相似度的实体消歧方法. 为了提供多模态的信息展示平台, 将图片链接作为属性存储在 Neo4j 数据库中, 并设计实现了可视化系统进行图片信息展示, 最终完成可视化的多模态计算机学科领域知识图谱的构建.

本文提出的跨模态领域知识图谱构建方法构建的多模态领域知识图谱不仅能系统化地整合领域多模态知识, 实现良好的可视化查询, 也是智能问答、推荐系统的底层支撑. 此方法具有良好的可迁移性, 适用于金融、医疗、生物等大多数领域. 但该方法依赖于领域本体的构建, 自顶向下的领域本体构建方法要求开发人员对目标领域的专业知识有一定程度的了解. 如果开发人员对目标领域没有系统化的了解, 也可以采用自底向上的方法进行领域本体的构建, 但这需要投入大量人工对原始数据进行审核.

参考文献

1 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]

2 李直旭, 何芙珍, 刘安. 多模态教学知识图谱的构建与应用. 福建电脑, 2019, 35(8): 5–8. [doi: 10.16707/j.cnki.fjpc.2019.08.002]

3 全威, 马志柔, 刘杰, 等. 基于医疗知识图谱的交互式智能导诊系统. 计算机系统应用, 2021, 30(12): 55–62. [doi: 10.15888/j.cnki.csa.008229]

4 马玉凤, 向南, 豆亚杰, 等. 军事系统工程中的知识图谱应用及研究. 系统工程与电子技术, 2022, 44(1): 146–153. [doi: 10.12305/j.issn.1001-506x.2022.01.19]

© 中国科学院软件研究所 http://www.c-s-a.org.cn

5 熊中敏, 马海宇, 李帅, 等. 知识图谱在海洋领域的应用及前景分析综述. 计算机工程与应用, 2022, 58(3): 15–33. [doi: 10.3778/j.issn.1002-8331.2106-0351]

6 陶天一, 王清钦, 付聿炜, 等. 基于知识图谱的金融新闻个性化推荐算法. 计算机工程, 2021, 47(6): 98–103, 114. [doi: 10.19678/j.issn.1000-3428.0057446]

7 陈烨, 周刚, 卢记仓. 多模态知识图谱构建与应用研究综述. 计算机应用研究, 2021, 38(12): 3535–3543. [doi: 10.19734/j.issn.1001-3695.2021.05.0156]

8 孙睿. 基于多模态知识图谱的推荐系统 [硕士学位论文]. 成都: 电子科技大学, 2021.

9 von Ahn L, Dabbish L. Labeling images with a computer game. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. Vienna: ACM, 2004. 319–326.

10 Bizer C, Heath T, Berners-Lee T. Linked data—The story so far. International Journal on Semantic Web and Information Systems, 2009, 5(3): 1–22. [doi: 10.4018/jswis.2009081901]

11 Hausenblas M, Troncy R, Bürger T, *et al*. Interlinking multimedia: How to apply linked data principles to multimedia fragments. Proceedings of the WWW 2009 Workshop on Linked Data on the Web. Madrid: CEUR-WS.org, 2009.

12 Deng J, Dong W, Socher R, *et al*. ImageNet: A large-scale hierarchical image database. Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition. Miami: IEEE, 2009. 248–255.

13 Chen XL, Shrivastava A, Gupta A. NEIL: Extracting visual knowledge from Web data. Proceedings of 2016 IEEE International Conference on Computer Vision. Sydney: IEEE, 2013. 1409–1416.

14 Vrandečić D, Krötzsch M. Wikidata: A free collaborative knowledgebase. Communications of the ACM, 2014, 57(10): 78–85. [doi: 10.1145/2629489]

15 Ferrada S, Bustos B, Hogan A. IMGpedia: A linked dataset with content-based analysis of Wikimedia images. Proceedings of the 16th International Semantic Web Conference. Vienna: Springer, 2017. 84–93.

16 Liu Y, Li H, Garcia-Duran A, *et al*. MMKG: Multi-modal knowledge graphs. Proceedings of the 16th International Conference on the Semantic Web. Portorož: Springer, 2019. 459–474.

17 Wang M, Wang HF, Qi GL, *et al*. Richpedia: A large-scale, comprehensive multi-modal knowledge graph. Big Data Research, 2020, 22: 100159. [doi: 10.1016/j.bdr.2020.100159]

18 孟卓宇. 基于多模态数据的生长发育知识图谱构建研究 [硕士学位论文]. 太原: 中北大学, 2021.

19 Gruber TR. Toward principles for the design of ontologies used for knowledge sharing? International Journal of Human-Computer Studies, 1995, 43(5–6): 907–928. [doi: 10.1006/ijhc.1995.1081]

20 Studer R, Benjamins VR, Fensel D. Knowledge engineering: Principles and methods. Data & Knowledge Engineering, 1998,25(1–2):161–197.[doi:10.1016/s0169-023x(97)00056-6]

21 任飞亮, 沈继坤, 孙宾宾, 等. 从文本中构建领域本体技术综述. 计算机学报, 2019, 42(3): 654–676. [doi: 10.11897/SP.J.1016.2019.00654]

22 Pan JZ. Resource description framework. Handbook on Ontologies. Berlin Heidelberg: Springer, 2009. 71–90.

23 Berners-Lee T, Hendler J, Lassila O. The semantic Web. Scientific American, 2001, 284(5): 34–43. [doi: 10.1038/scientificamerican0501-34]

24 陈皓, 周传生. 基于 Python 和 Scrapy 框架的网页爬虫设计与实现. 电脑知识与技术, 2021, 17(13): 3–5.

25 Wen Y, Fan C, Chen G, *et al*. A survey on named entity recognition. Proceedings of the 8th International Conference on Communications, Signal Processing, and Systems. Urumqi: Springer, 2019. 1803–1810.

26 Devlin J, Chang MW, Lee K. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2018. 4171–4186.

27 Liu W, Fu XY, Zhang Y, *et al*. Lexicon enhanced Chinese sequence labeling using BERT adapter. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. ACL, 2021. 5847–5858.

28 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.

29 Sun WW, Uszkoreit H. Capturing paradigmatic and syntagmatic lexical relations: Towards accurate Chinese part-of-speech tagging. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Jeju Island: Association for Computational Linguistics. 2012. 242–252.

30 He KM, Zhang XY, Ren SQ, *et al*. Deep residual learning for image recognition. Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778.

(校对责编: 牛欣悦)

© 中国科学院软件研究所 http://www.c-s-a.org.cn