基于Spark的人工智能知识的知识图谱构建

摘要：

关键词：知识图谱；Spark；可视化

Abstract:

Keywords:

0 引言

人工智能是当前最热门研究领域之一，甚至被誉作世界三大尖端技术之一[1]，近年来我国甚至将其上升到国家战略的高度：2017、2018与2019年的政府工作报告中均被提及[2]。可见，人工智能在现代科学技术与经济社会中有着不可替代的地位，随着5G时代的到来，人工智能必将展现更广阔的应用前景。与此同时，人工智能相关方向的人才匮乏也正越来越成为（市场）关注的议题[3]，而在培养人才时，如何准确把握所授相关领域知识的准确性、全面性与前沿性成了一个难题，知识图谱是解决这一难题的有效工具。知识图谱是人工智能领域重要的一个技术分支，其目的是将现有的人类知识构建为一个结构化的知识库。目前，已经有许多大型知识图谱被构建出来，如DBpedia 、Freebase等，然而，当前的知识图谱工具普遍存在以下问题：1）通用知识图谱工具涉面较广，但知识冗余混乱、组织零散、系统性差，不利于用户的专业学习；2）垂直知识图谱工具种类少，成熟的应用仅限于某些领域，在一些具有较大应用需求的领域未获重视，前景广阔[4]。

综上所述，本文的目的是构建一个面向学习者尤其是本科生的人工智能领域的垂直知识图谱。人工智能领域繁多，我们选取机器学习、自然语言处理与机器视觉等三个领域作为代表。

1 相关工作

知识图谱的构建技术仍在持续发展中，目前存在多种流派，每一种技术手段途径各异、效果良莠不一。随着相关技术的不断演变与发展，新的知识图谱构建方法被不断推出，有些研究也在尝试使用经典的方法在新的应用领域构建相应的垂直知识图谱，均取得了一定效果。

金婧等[6]侧重于知识图谱表示学习方法，在TransE[5]模型的基础上提出了一种融合实体类别信息的知识表示学习模型（TEKRL），实验表明该模型在各项评价指标上得到了提升；杨玉基等[7]在对领域知识图谱的系统研究上，提出了一种构建领域知识图谱的“四步法”，该方法可以在较短时间内构建准确率较高的学科知识图谱；孙昊天等[8]实现一种基于带权三元组构建时政类知识图谱的方法，该方法在参数设置得当的情况下可以得到较为理想的以亲密程度为关系的知识图谱；董永强等[9]提出了一种基于YANG[10]模型由数据模型驱动的网络领域知识图谱构建方法，通过该方法构建的知识图谱可为网络维护大数据提供支持，降低了人工成本。

而在通过经典方法构建垂直知识图谱上，熊晶等[11] 基于多源异构数据源构建了甲骨学融合知识图谱，所得的知识图谱节点较多，可以满足甲骨学研究的基本需求；刘燕等[12]利用相关技术构建了医学知识图谱，在医药卫生知识服务系统平台取得了理想的效果；白如江等[13]提出科学事件的概念，并利用 LTP [14]语言云根据所谓科学事件模型构建了图情领域的知识图谱，实验结果差强人意；陈成等[15] 提出了意图知识图谱的定义并完成了构建，通过有关范例说明了该图谱可以作为政府治理的一种依据。

有鉴于新兴理论与技术在构建知识图谱，以及使用经典方法在新的应用领域构建有关垂直知识图谱所取得的成功与不足，本文基于大数据处理平台Spark，并借助Jiagu模型出色的知识关系提取能力，并使用从国内两大流行的技术博客平台CSDN与博客园爬取到的元数据，构建了一个小规模的人工智能领域的知识图谱。

2 数据来源

2.1 爬取工具的选择

本文选择CSDN与博客园作为主要的元数据获取平台，因其主要数据采

用网页来展现，因此本文选择Python作为爬取工具。较之其他静态编程语言，如C++、Java、C#等，Python爬取网页文档的接口更简洁。而相比其他动态脚本语言，如perl、shell、ruby等，python的urllib2包提供了较为完整的访问网页文档的API。

值得一提的是，抓取网页有时需将爬虫程序伪装成普通的浏览器。因为许多网站都采取了防爬措施，单纯的爬取操作极容易被网站检测出来并封杀。Python提供了许多鲁棒的第三方包如requests和mechanize，可以帮助爬虫轻松地越过网站的防爬策略，例如模拟user agent行为构造合适的请求，如：用户登陆、session/cookie的存储和设置。

2.2 提高爬取效率的方法

传统的网络爬虫是运行在本地，稍优化的策略是采取“单机多核”的方

式。为了更有效地解决爬取效率过低的问题，同时结合实际的实验条件，本文采用主从分布式爬虫。

本文将一台阿里云服务器作为master服务器，用于分发所需爬取内容的URL，同时维护存储在redis中待爬取URL的列表。由三台本地的笔记本电脑组成slave服务器组，用于对各自从master服务器所获得的URL执行网页爬取任务；若slave在爬取过程中遇到新的URL，一律将其返回master服务器由master解析处理，slave服务器间不进行通信。本文所用master服务器与slave服务器组的性能配置如表2.1所示，主从分布式爬虫的逻辑结构如图2.1所示，爬虫的类图结构如图2.2所示。

此外，为了防止网站服务器锁定爬虫的IP，本文所使用的爬虫程序对爬取频率进行了限制，以及使用代理IP。

3 Spark与Jiagu模型

3.1 Spark与hive平台

Spark[16]是 基于内存计算的大数据并行计算框架，因为它基于内存计算，所以提高了在大数据环境下数据处理的实时性，同时保证了高容错性和高可伸缩性，允许用户将 Spark部署在大量廉价硬件之上，形成集群。hive[17]是一个基于Hadoop的数据仓库平台，通过hive我们可以快速地对存储在数据库中数据进行抽取、加载与转换（ETL）等操作。hive定义了一个类似于SQL的查询语言：HQL，能够将用户编写的查询语句转化为相应的Mapreduce程序并基于Hadoop执行。需要注意的是，hive本身并不存储数据，因而用户需要选择一个传统的数据库进行数据存储，基于可操作性与成本等角度考虑，本文采用MySQL。

本文将使用Spark平台的相关工具进行数据预处理。

3.2 数据预处理

第2节所爬取到的元数据杂源异质，散乱冗余，并且由于网页文本本身的结构导致数据中存在大量标签，无法直接用于下一步操作。因此本文借助Spark平台快速的数据处理能力及hive对数据库高效的ETL操作，对文本进行预处理。

首先，在spark-shell上将数据成功加载到hive中，为后续存取提供了数据来源。其次，在hive上创建了数据库，在spark-shell上依次将爬虫爬取的json文件导入成表。而后，在IDEA上编程对数据去重，这里主要使用了Spark的几个API，如：duplicate、filter、regexp\_replace、regexp\_extract等。完成数据的存储、去重和标签过滤后，借助于github上开源的敏感词汇库[18]，对表数据进行敏感词过滤，以此得到更干净的数据。本文所用部分spark-shell处理命令如图3.1，数据预处理的程序类图如图3.2所示，预处理后的部分数据如图3.3所示。

3.3 Jiagu模型

3.4 知识抽取

4 结果与分析

4.1 三元组的转换

4.2 图谱可视化

5 结束语