基于 Spark 的人工智能知识图谱构建

文华，刘宏鑫，周余

摘　要：随着计算机大数据的快速发展，可以借助于互联网平台的各种工具找到有价值内容，但海量数据給筛选、组织与评价带来极大困难。知识图谱具有强大的语义处理与开放互联能力，可以精确地表达概念及其相互关系所构成地语义网络，更好地为机器所理解；且能够帮助用户快速、准确地检索所需要地信息。本文基于 Spark 平台构建了人工智能中的机器学习、自然语言处理与机器视觉等三个领域的知识图谱，完成相关知识的重整，取得了较好的实验效果。

关键词：知识图谱；Spark；可视化

Abstract: With the rapid development of computer data, it has been developed into reality, finding valuable content with the help of various tools of Interne. However, the massive data has brought great hardships to screening, organization and evaluation. Knowledge map owns a powerful semantic processing and opens interconnection ability, which can accurately express the semantic network formed by concepts and their mutual relations that can be understood by the machine better. Furthermore, it can help users retrieve the required information quickly and accurately. Based on the spark platform, this paper constructs a knowledge map of machine learning, natural language processing and machine vision that are related to Artificial Intelligence, which completes the reorganization of relevant knowledge, and achieves good experimental results.

Keywords: Knowledge Graph; Spark; Visualization

0. 引言

人工智能（Artificial Intelligence，简称 AI），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[2-4]。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[5]，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（Knowledge Graph）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。

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（Machine Learning，ML）、自然语言处理（Natural Language Processing，NLP）与机器视觉（Machine Vision，MV）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、效果良莠不齐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。构建知识图谱的一般技术流程如图 1.1 所示。

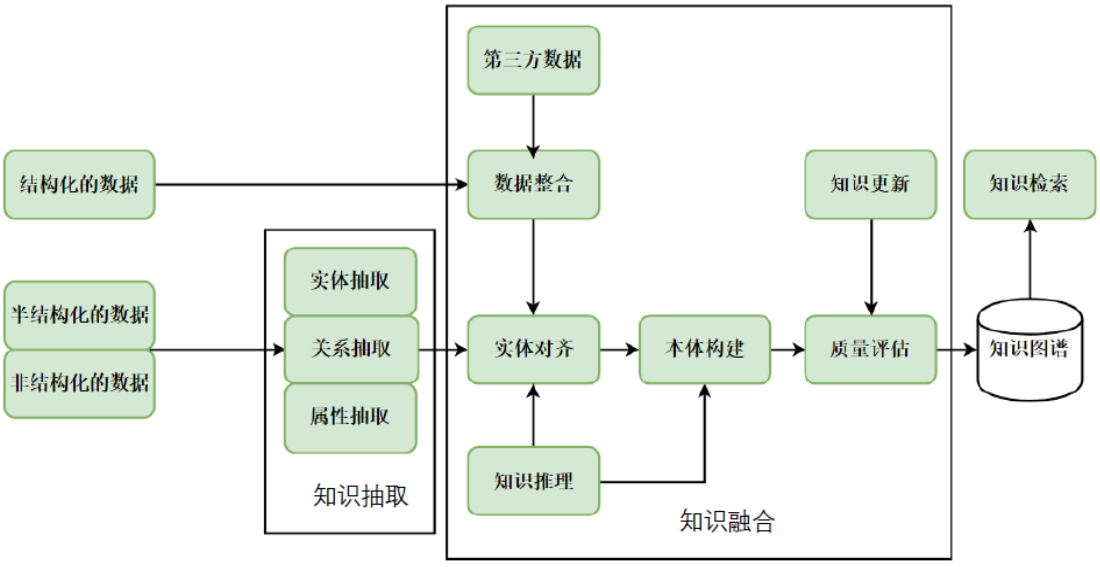


图 1.1 知识图谱构建流程

金婧等[6]侧重于知识图谱表示学习方法，在 TransE[7]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[8]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[9]实现了一种基于带权三元组（Unit Triplet）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[10]提出了一种基于 YANG[11]模型由数据模型驱动（Data-Model-Driven）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（Big Data）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[12] 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[13]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[14]提出科学事件（Scientific Events）的概念，并利用 LTP[15]语言云根据所谓科学事件模型构建了图情（Library Information）领域的知识图谱，实验结果差强人意；陈成等[16] 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，构建了一个学习者尤其是本科生适用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（Metadata）获取平台，因其主要数据采用网页来展现，所以本文选择 Python 作为爬取工具。python 不但用于抓取网页文档的接口简洁，同时其访问网页文档的 API 也相当完整。

值得一提的是，抓取网页有时需将爬虫（Crawler）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、mechanize、selenium，可以帮助爬虫轻松地越过网站的防爬策略。

在抓取了网页之后，仍需进一步的处理，如过滤 html 标签，提取文本等，而 python 的 beautifulsoap 库等使编写非常简洁的代码即可完成大部分文档的处理成为可能。

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（Master-Slave Distributed Crawler）[17]。

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。

表 2.1 master 服务器与 slave 服务器组性能配置

Server	Processor	RAM/GB	Storage/GB	CPU core(s)
master	Intel(R) Xeon(R) CPU E5-2682 v4 @ 2.50GHz	2	40	1
slave 1	Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz 2.30 GHz	16	128(SSD) + 1024	4
slave 2	Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz	16	128(SSD) + 1024	4
slave 3	Intel(R) Core(TM) i7-8750H CPU @ 2.20GHz 2.21 GHz	16	128(SSD) + 1024	6

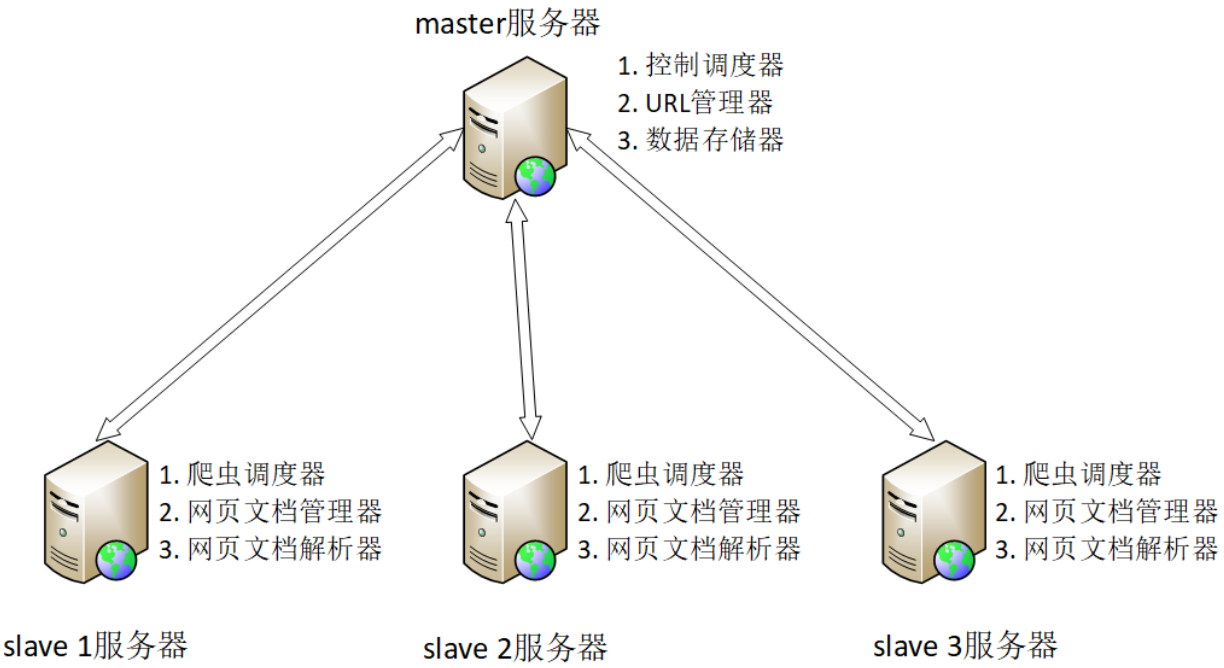


图 2.1 主从式分布爬虫逻辑结构

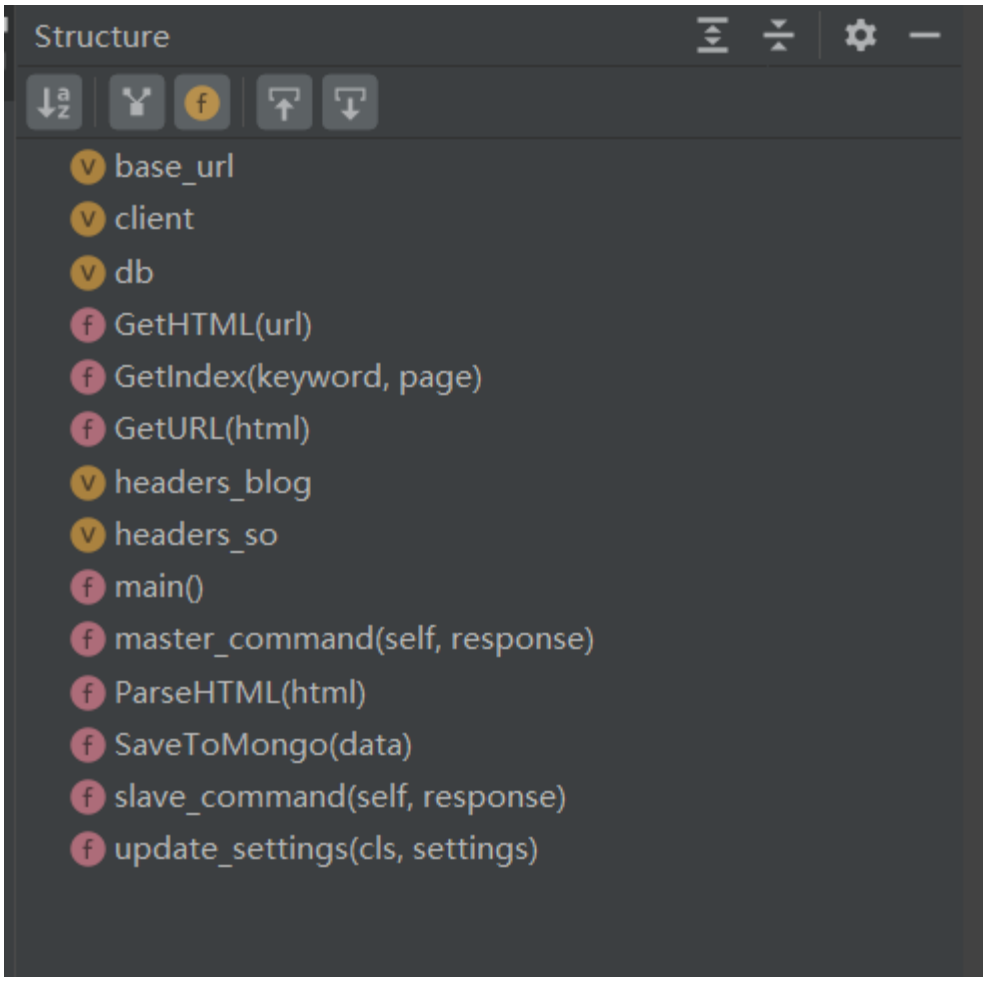


图 2.2 爬虫程序的类图结构

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 池。

3. Spark 与 Jiagu 模型

3.1. Spark 与 hive 平台

Spark[18]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[19]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（，Extract， Transform， Load， 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[20]，对表数据进行敏感词（Sensitive Word）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。

```
scala> val dataDF1 =
spark.read.format("json").load("file:///home/hadoop001/hadoop/data/Spider-
Data/cnblog_computer_version.json")

scala> dataDF1.select(dataDF1.col("author"),
dataDF1.col("content"),dataDF1.col("date"),
dataDF1.col("title")).write.saveAsTable("dachuangppreprocessingdata.cnblog_computer
_version")
```

图 3.1 spark-shell 处理命令

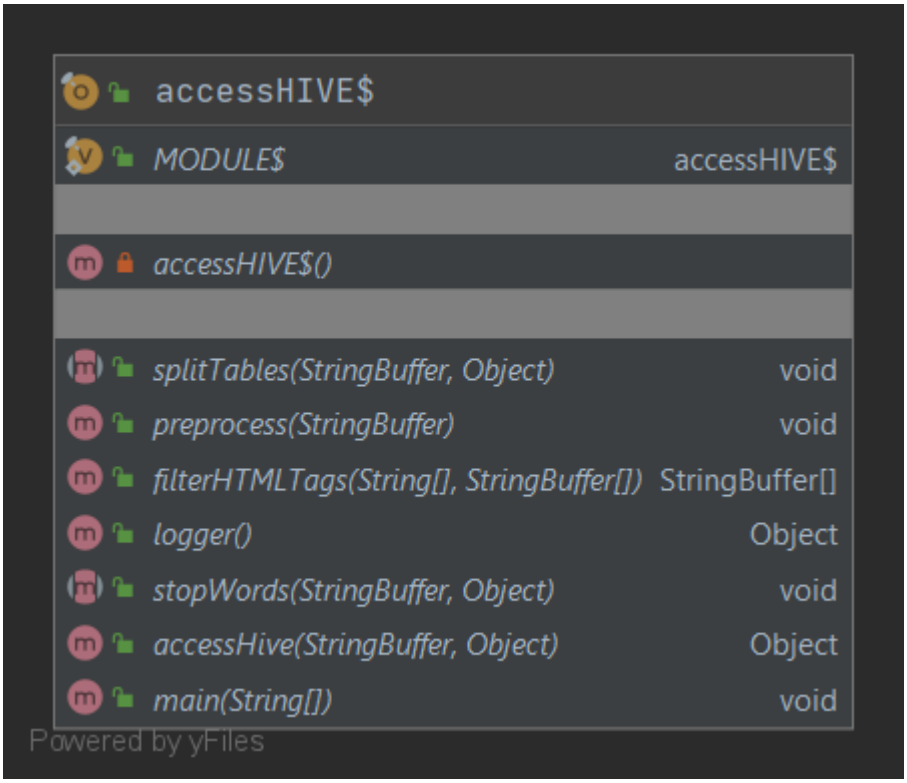


图 3.2 数据预处理程序的类图

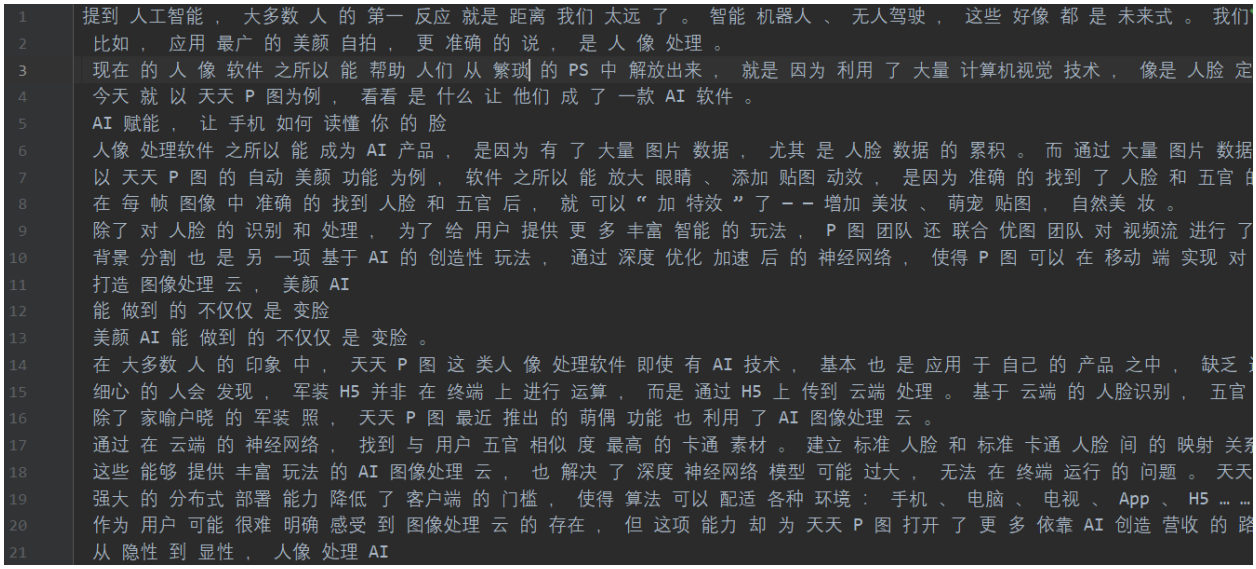


图 3.3 预处理后的部分数据

3.3. Jiagu 模型

Jiagu 模型[21]是一个国产的开源自然语言处理工具，以 BiLSTM 等模型为基础，使用大规模语料训练而成。Jiagu 模型提供中文分词、词性标注、命名实体识别、情感分析、知识图谱关系抽取、关键词抽取、文本摘要、新词发现、情感分析、文本聚类等常用自然语言处理功能，API 丰富，且操作便捷、稳定性高。本文选择 Jiagu 模型作为知识抽取的工具，取得了十分理想的效果。

3.4. 知识抽取

在知识图谱中，知识一般以三元组(p, r, q)的形式来表示，其中 p 与 q 分别代表前后两个实体，r 代表前后实体之间的关系[22]。显然三元组是构建知识图谱的重要基础，三元组中实体间的关系是否准确、完整等也是知识图谱的构建成功与否的重要判据。

本文采用 BIO 方式[23]对待训练文本进行实体命名标记，每行一个字符，并按 19:5 的比例分别设置训练数据与验证数据，且为测试训练所得模型的准确程度设置了较训练数

据 75%的测试数据，详细信息如表 3.1 所示。在分别调节学习率（Learning Rate）、迭代次数（Iterations）、阻尼系数（Damping Coefficient）等参数后对标记文本进行训练，参数详情如表 3.2 所示。实验结果用 held-out 方法[24]进行评估，即统计知识图谱中已有的实体被 Jiagu 模型检测出的数量，正确的实体被排序靠前的数量愈多，则在准确率/召回率曲线上，随着召回率（Recall Rate）的增长准确率（Accuracy Rating）就下降得越慢，也即知识抽取的质量愈高。实验结果的准确率/召回率曲线如图 3.4 所示，所得部分三元组如图 3.5 所示。

表 3.1 数据集的统计信息*[1]

数据集	关系数量	语料行数
训练集	10	2435796
验证集		634547
测试集		1849620

表 3.2 所用训练参数

Learning rate	Iterations	Damping coefficient
0.001	50000	0.85

* 在训练文本中，每行一个字符。

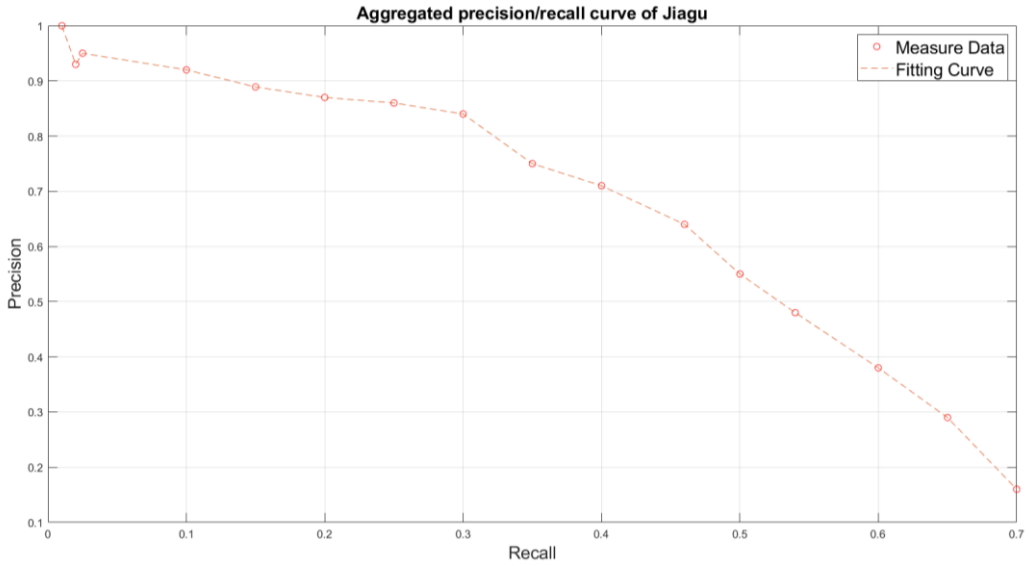


图 3.4 准确率/召回率

```
逻辑回归,优势,处理非线性效应
逻辑回归,缺点,仅用于二进制分类
随机森林,优势,防止过拟合
随机森林,用于1,回归
随机森林,用于2,分类
随机森林,缺点1,容易生长
随机森林,缺点2,随机子集高
评价矩阵,术语1,真阳性(TP)
评价矩阵,术语2,真阴性(TN)
评价矩阵,术语3,假阳性(FP)(I型错误)
评价矩阵,术语4,假阴性(FN)(II型错误)
特征选择,也称为1,变量选择
特征选择,也称为2,属性选择
特征选择,也称为3,变量子集
特征选择,选择,最佳相关特征
特征选择,帮助1,简化ML模型
特征选择,帮助2,提高ML模型的准确性
特征选择,有助于,更快地训练
特征选择,防止,过拟合
```

图 3.5 三元组数据

4. 知识图谱的可视化

4.1. 三元组的转化

本文所选可视化工具为基于 TypeScript 开源的可视化框架 amCharts 4，其与 TypeScript、Angular、React、Vue 和纯 JavaScript(ES6)进行了原生集成[25]。由于用户通过某个关键字请求实体的三元组信息时，其数据量可能是非常大的。此外，amCharts 4 要求数据以特定的 json 格式存储，显然 3.4 节所得的三元组无法直接用于可视化（Visualization）。出于存取效率、数据可拓展性等因素考虑，本文将三元组数据预先导入 MySQL 数据库，当前端发出数据请求时，通过 PHP 编程实现从服务器端查找相应的原始三元组数据并使用相应 API 转换为 json 格式返回给前端。前端在接收到 PHP 返回的原始三元组数据后，需要对原始三元组数据进行预处理，将原始的 json 数据转化为 amCharts 可识别的特定格式 json 数组，并最终作为 amCharts 的数据源加载，渲染（Render）到指定的 SVG 画布上，最终形成可操作的力导向图谱。具体交互的流程如图 4.1 所示。

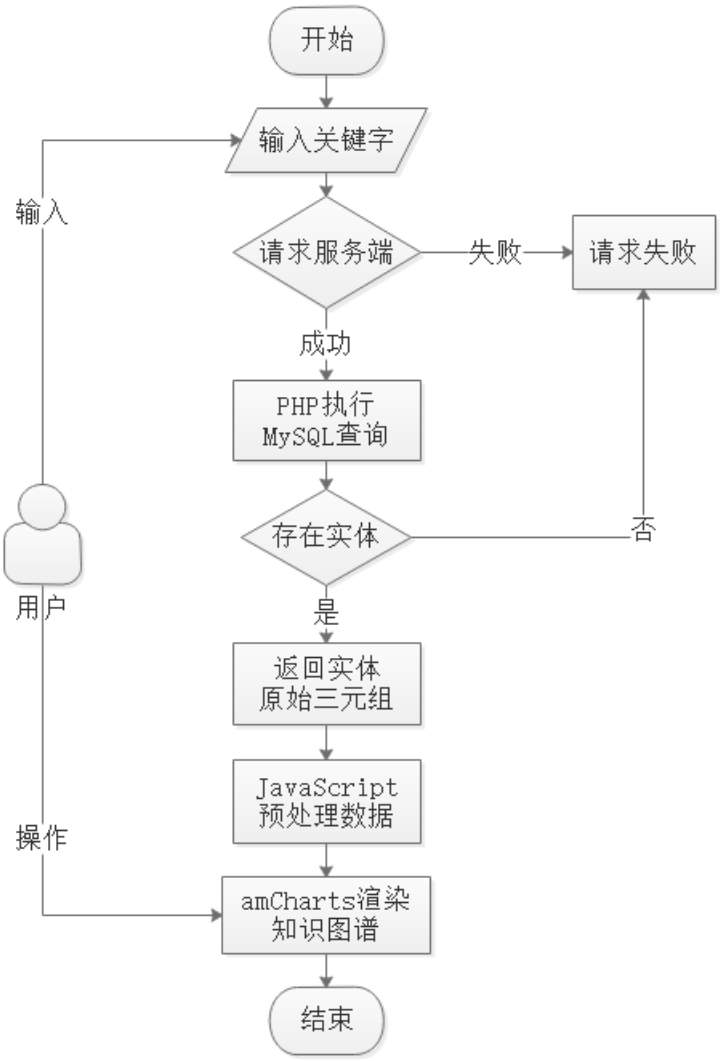


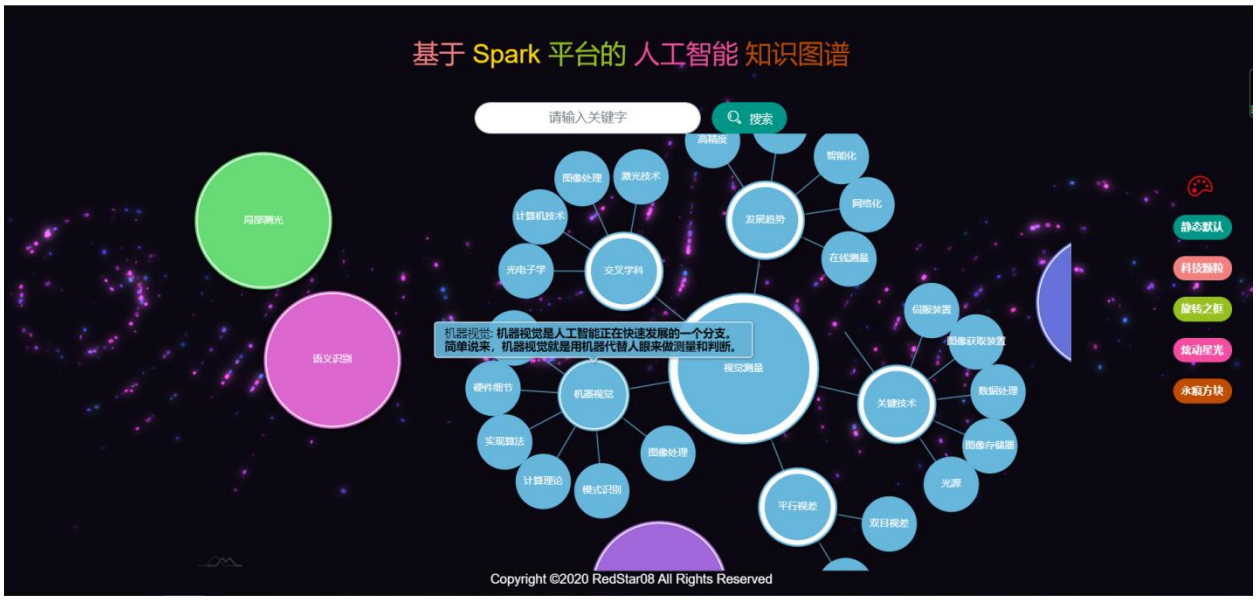
图 4.1 知识图谱可视化流程图

4.2. 图谱可视化

amCharts 4 是一个基于 TypeScript 开源的可视化框架，具有图表种类丰富、图形效果炫丽、动画或静态呈现、与平台无关等特点，适用于各个行业的可视化需求场景，因此本文将其作为知识图谱的可视化工具。本文使用 HTML/CSS/JavaScript 设计页面元素及基本布局，并采用力导向图作为图谱的呈现形式。当用户在搜索框键入查询关键词时，通过 GET 请求关键字，后台通过 PHP 查询数据库并返回请求的数据。前端得到请求的数据

后，通过 JavaScript 进行预处理并借助 amCharts 进行可视化展示。本文所构建知识图谱的可视化结果示例如图 4.2 所示，此外，本文所采用的图谱可视化工具支持多种主题背景的选择，如图 4.3 所示。

a)



b)

图 4.2 知识图谱可视化结果

图 4.3 丰富的主题选择

5. 结果与分析

本文成功地构建了人工智能领域的知识图谱，首次将本科计算机类专业的课程内容知识以知识图谱的形式展示出来；可以帮助用户准确、快速地检索人工智能领域相关术语并提供解释，同时给出术语的联想结果，利于用户进一步学习；形象化地展示人工智能领域的脉络、历史沿革与发展趋势，为用户复习、深入学习提供参考。

下一步的工作将从几个方面进行研究：采用知识联想等方法增加知识图谱中的知识实体规模，进一步优化知识关系抽取，改善知识融合等。

6. 结束语

垂直知识图谱的应用前景广阔，囿于构建技术尚在发展、仍未成熟，相关的产品较少。本文大胆对目前热门的人工智能领域进行了知识图谱构建，初步探索出了相关图谱的构建步骤，得到了效果较为理想的实验结果。本文的构建方法可以应用于大多数针对特定学科或领域的垂直知识图谱的构建，以期在扩大训练语料的基础上得到较本文实验结果覆盖率更广的领域知识，即规模更为庞大的 RDF。值得一提的是，本文在构建图谱的过程中认识到：汉语作为一门分析语所具备的固有特点是构建汉语知识图谱的障碍之一，在后续工作中或可以考虑以英文语料为基础构建知识图谱，待完成后再行翻译。

本文还以人工智能领域的机器学习、自然语言处理与机器视觉三个分支为例，介绍了构建相关垂直知识图谱的技术流程。以期能够抛砖引玉，使其他有志之士有所参考。

参考文献：

[1]邹蕾, 张先锋. 人工智能及其发展应用[J]. 理论与研究, 2012 年第 02 期.

[2]国务院. 2017 年国务院政府工作报告[R]. 第十二届全国人民代表大会第五次会议, 2017 年 3 月 5 日.

[3]国务院. 2018 年国务院政府工作报告[R]. 第十三届全国人民代表大会第一次会议, 2018 年 3 月 5 日.

[4]国务院. 2019 年国务院政府工作报告[R]. 第十三届全国人民代表大会第二次会议, 2019 年 3 月 5 日.

[5]陈劲, 吕文晶. 人工智能与新工科人才培养：重大转向[J]. 高等工程教育研究, 2017 年 06 期.

[6]金婧, 万怀宇, 林友芳. 融合实体类别信息的知识图谱表示学习方法[J]. 计算机工程, https://doi.org/10.19678/j.issn.1000-3428.0057353

[7]XIE Ruobing, LIU Zhiyuan, SUN Maosong. Representation learning of knowledge graphs with hierarchical types[C]// International Joint Conference on Artificial Intelligence. New York, NY, USA: AAAI Press, 2016: 2965 - 2971

[8]杨玉基, 许斌, 胡家威, 仝美涵, 张鹏, 郑莉. 一种准确而高效的领域知识图谱构建方法[J]. 软件学报, 2018,29(10): 2931-2947. http://www.jos.org.cn/1000-9825/5552.htm

[9]孙昊天, 杨良斌. 基于带权三元闭包的知识图谱的构建方法研究[J]. 情报杂志, 2019, 38(6): 168 - 173.

[10]董永强, 王鑫, 刘永博, 杨望. 异构 YANG 模型驱动的网络领域知识图谱构建[J]. 计算机研究与发展, 2020 年 04 期: 699 – 708.

[11]Bjorklund M. RFC 7950: The YANG 1.1 Data Modeling Language[OL]. IETF, 2016[2019-12-01]. https://tools.ietf.org/html/rfc7950

[12]熊晶,焦清局,刘运通. 基于多源异构数据的甲骨学知识图谱构建方法研究[J]. 浙江大学学报(理学版), 第 47 卷第 2 期: 131 – 150.

[13]刘燕,傅智杰,李姣,侯丽. 医学百科知识图谱构建[J]. 中华医学图书情报杂志, 2018 年 6 月, 第 27 卷第 6 期: 28 – 34.

[14]白如江, 周彦廷, 王效岳, 王志民. 科学事件知识图谱构建研究[J]. 情报理论

与实践, http://kns.cnki.net/kcms/detail/11.1762.G3.20200317.1708.008.html.

[15]https://www.ltp-cloud.com/

[16]陈成, 陈跃国, 刘宸, 吕晓彤, 杜小勇. 意图知识图谱的构建与应用[J]. 大数据, 2020 年 02 期: 57 – 68.

[17]刘泽华, 赵文琦, 张楠. 基于 Scrapy 技术的分布式爬虫的设计与优化[J]. 信息技术与信息化, 2018 年 2 - 3 期: 121 – 126.

[18]赛金辰. 基于 Spark 的 SVM 算法优化及其应用[D]. 北京邮电大学, 2017 年 1 月.

[19]李爽. 基于 Spark 的数据处理分析系统的设计与实现[D]. 北京交通大学, 2015 年 6 月.

[20]https://github.com/fighting41love/funNLP

[21]https://github.com/ownthink/Jiagu

[22]徐增林, 盛泳潘, 贺丽荣, 王雅芳. 知识图谱技术综述[J]. 电子科技大学学报, 2016 年 7 月, 第 45 卷第 4 期: 589 – 606.

[23]刘哲宁, 朱聪慧, 郑德权, 赵铁军. 面向特定标注数据稀缺领域的命名实体识别[J]. 指挥信息系统与技术, 2019 年 10 月, 第 10 卷第 5 期: 14 – 18.

[24]MINTZ, BILLS S, SNOW R, et al. Distant supervision for relation extraction without labeled data[C]// Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Proceeding of the AFNLP. Stroudsburg: ACCL, 2009: 1003 – 1011.

[25] 孙启民, 胡莉丽, 黄威. 基于 SNMP&Amcharts 的性能监测技术在动环监控系统的应用[J]. 技术创新, 2016 年 02 期: 35 – 38.