计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(7):58−64 [doi: 10.15888/j.cnki.csa.006971]

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

科技政策库的系统集成与建设①

武　虹[1], 杨宝龙[1], 杜治高[2], 李涵露[2]

[1](中国科协创新战略研究院, 北京 100086)
[2](北京航空航天大学, 北京 100083)
通讯作者: 杜治高, E-mail: ld_poplar@163.com

摘　要: 为了满足科技政策研究需要, 中国科协设计并实现了一种科技政策库系统. 本文首先介绍了科技政策库的总体设计方案、系统工作流程; 然后详细介绍了系统组成, 整个系统由数据采集子系统、数据清洗子系统、数据分析子系统 3 个子系统组成. 数据采集子系统基于网络爬虫框架 Scrapy 软件针对大量异构站点设计了可管理的网络爬虫, 并基于 ABBYY FineReader 软件 (俄罗斯软件公司 ABBYY 发行的一款文档识别软件) 实现了历史文献 OCR 识别 (Optical Character Recognition, 光学字符识别) 和入库. 数据清洗子系统基于机器学习算法实现了数据去重、非相关数据识别、数据属性缺陷识别等功能. 数据分析子系统则对有效入库的科技政策进一步进行了文本分类、关联关系分析、全文检索. 从 2018 年 10 月上线以来, 该系统从 226 个数据源采集 564 749 条数据, 经过数据清洗之后入库 404 083 条数据, 能够有力地支撑科技政策研究工作.

关键词: 科技政策库; 网络爬虫; 数据清洗; 机器学习; 自然语言处理

引用格式: 武虹,杨宝龙,杜治高,李涵露.科技政策库的系统集成与建设.计算机系统应用,2019,28(7):58–64. http://www.c-s-a.org.cn/1003-3254/6971.html

System Integration and Construction of Science and Technology Policy Database

WU Hong[1], YANG Bao-Long[1], DU Zhi-Gao[2], LI Han-Lu[2]

[1](National Academy of Innovation Strategy, Beijing 100086, China)
[2](Beihang University, Beijing 100083, China)

Abstract: In order to meet the needs of science and technology policy research, China Association for Science and Technology designs and implements a policy database system. This study first introduces the overall design scheme and system workflow of the science and technology policy database. Then it introduces the system components in detail. The system consists of three subsystems: data acquisition subsystem, data cleaning subsystem and data analysis subsystem. The data acquisition subsystem is based on the Scrapy framework for designing manageable web crawlers for a large number of heterogeneous sites, as well as ABBYY FineReader-based OCR (Optical Character Recognition) for historical documentation. The data cleaning subsystem implements functions such as data deduplication, non-correlated data identification, and data attribute defect recognition based on machine learning algorithms. The data analysis subsystem further carries out text classification, association analysis and full-text search for the effective policies. Since its launch in October 2018, the system has collected 564 749 pieces of data from 226 data sources. After data cleaning, it stores 404 083 pieces of data, which can strongly support the research of science and technology policy.

Key words: science and technology policy database; Web crawler; data cleaning; machine learning; natural language processing

① 收稿时间: 2019-01-03; 修改时间: 2019-01-24; 采用时间: 2019-01-31; csa 在线出版时间: 2019-07-01

科技政策是国家为实现一定历史时期的科技任务而规定的基本行动准则, 是确定科技事业发展方向, 指导整个科技事业的战略和策略原则. 科技政策是否高效合理, 对科学技术能否快速发展具有重要的影响. 为了提升科技政策制定过程的系统性和科学性, 2005 年美国提出了“科学政策学” (Science of Science Policy, SoSP) , 把科技政策研究作为一门“科学”[1],并将 SoSP 作为建立美国政府“基于证据的决策系统”的重要举措[2]. 2008 年国家科学技术委员会 (NSTC) 和白宫科技政策办公室 (OSTP) 联合发布了《科技政策学: 联邦研究路线图》, 指导国家科技政策学的发展[3]. 2009 年日本科学技术振兴机构 (JST) 在日本发起科技政策学的研究与梳理工作, 加强日本科技政策的证据基础, 推进科技政策科学的发展[4]. 2010 年, 欧盟与美国联合举办了欧美科技政策学讨论会, 以推进欧洲科技政策学的规范化研究[5]. 自此, 世界科技政策研究迈入了科技政策科学的新阶段[6], 并形成了大量高水平的研究成果. 近年来, 科技政策研究在国内也得到了越来越多的关注[7,8].

作为科技政策研究的主体, 国内外的历史科技政策种类繁多、数量庞大, 近些年的历史政策散落在互联网各处, 2000 年以前的历史政策则一般只有纸版文档, 这些政策文献很难得到有效的收集整理, 对科技政策研究带来了不便和障碍. 随着网络爬虫技术的发展, 利用信息技术从互联网收集历史科技政策文献成为了可能; 而自然语言处理、大数据、机器学习等技术的发展, 则为科技政策研究提供了新的技术手段[9]. 部分科技政策研究单位已经开始收录和整理科技政策文献, 但是这项研究整体上仍然处于起步阶段. 部分现有科技政策数据库仅采集国内政策, 缺乏对国际先进经验的整理; 或者仅限于科技政策收集, 对政策解读、领导讲话、政策研究等相关文献缺乏关注; 还有部分政策库采集了政府部门制定的所有政策, 对科技政策研究而言针对性不强. 另外, 现有科技政策库建设的关注焦点仍集中在数据采集方面, 对数据清洗, 以及统计分析等研究支持能力缺乏深入研究 .

本文基于 Scrapy 爬虫框架[10]设计和实现了可管理的网络爬虫, 从 225 个互联网站点采集国内外科技政策文献; 并进一步对原始政策数据进行结构化信息提取、数据去重、非相关数据清洗等数据清洗操作, 构建了完整和统一的科技政策库; 在政策库的基础上实现文本分类、关联分析、全文检索、统计分析功能, 为科技政策的研究与制定提供了参考和依据.

1 系统总体设计方案

1.1 系统功能目标

(1) 面向 225 个国内国外、结构不一、安全策略各异的互联网站点, 设计可配置、可管理的网络爬虫, 采集科技政策相关的数据, 实现数据的增量更新. 利用 OCR 技术识别历史文献图书, 提取文献的结构化信息, 实现历史文献的批量入库.

(2) 采用机器学习、自然语言处理等技术, 对从互联网采集的 56 万条科技政策相关网页进行数据清洗, 通过数据去重、非相关数据清洗、数据属性缺陷处理等一系列操作, 去除噪音数据, 提升数据质量.

(3) 在数据清洗基础上实现科技政策库文献的分类、关联关系分析、全文索引, 并向用户提供文献检索、查阅和下载功能; 针对有效入库的文献实现时域分析、地域分析等功能.

1.2 系统流程设计

科技政策库系统通过网络爬虫采集互联网上的政策数据, 对纸版历史文献进行 OCR 识别; 这两类原始数据在采集之后被写入消息队列; 数据清洗子系统作为消息队列消费者, 对原始数据进行数据清洗, 并将有效数据写入文献存储子系统; 数据分析子系统则对文献存储子系统内的文献进行全文索引、文本分类、关联分析, 并向管理员和研究人员提供文献检索、查阅、下载、统计分析接口. 系统的具体流程见图 1.

(1) 数据采集子系统包括网络爬虫、增量爬取调度器、数据属性识别、爬虫配置、爬虫异常管理等组件. 对 225 个国内外站点按照网站结构、安全策略等特点进行分类, 基于 Scrapy 爬虫框架设计一系列爬虫, 每个爬虫负责一类站点的数据采集.

(2) OCR 子系统基于 ABBYY FineReader 软件实现历史文献的电子化, 并进一步提取电子文献的结构化数据, 批量导入消息队列.

(3) 采用 Redis 软件实现消息队列. 本系统采集的文献可以分为核心政策、领导讲话、政策解读、科技政策相关新闻、科技政策研究论文、科技政策研究项目等 10 类. 不同类型文献的数据属性存在较大差异, 通常来自同一站点栏目或者搜索结果列表的文献结构化信息类似. 因此, 基于文献来源在消息队列中划分消

息主题, 同一消息主题下的文献具有相同的数据结构.

(4) 数据清洗子系统包括数据去重、非相关数据清洗、数据属性缺陷处理等组件, 清除原始数据中的脏数据.

(5) 文献存储子系统包括: Mysql 数据库, 存储文献的数据属性信息; 文件系统, 存储原始 html、txt、pdf、doc 等各种格式的政策文本; Solr, 存储文本和部分结构化信息, 实现全文索引.

(6) 数据分析子系统包括文本分类, 文本关联关系分析, 文献检索、查阅、下载, 文献统计分析等组件.

(7) 系统包括管理员和研究人员两类用户, 管理员具有爬虫配置、异常处理、文献增删改查等系统管理权限, 研究人员则可以从系统检索、查阅、下载文献, 进行文献的统计分析和结果可视化查看.

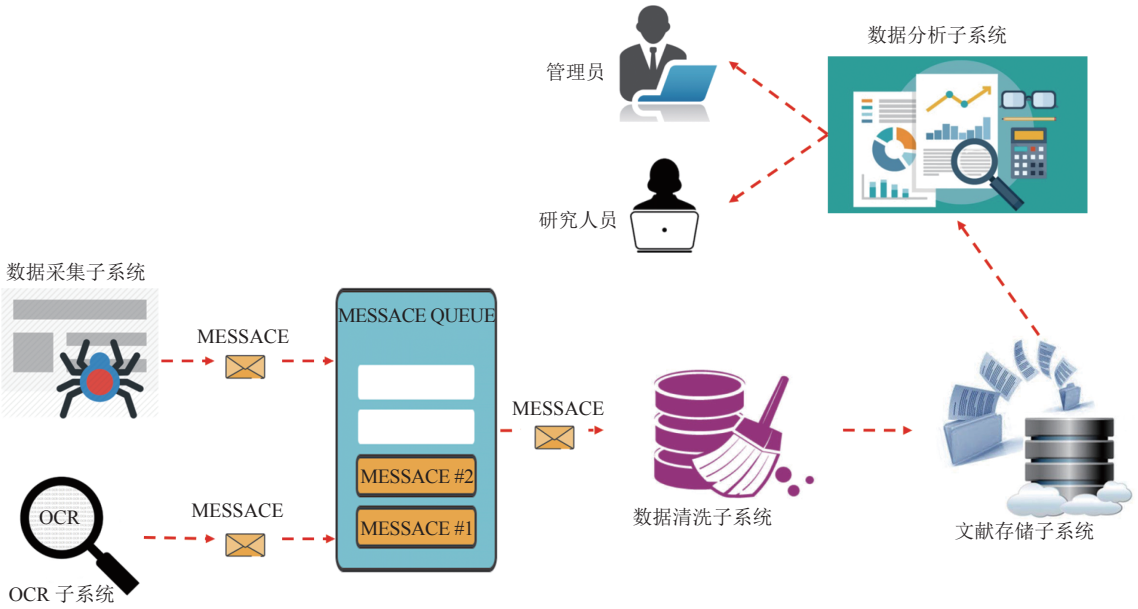

图 1 系统流程示意图

2 系统组成

2.1 数据采集子系统

科技政策库系统的采集源共 225 个站点, 其中中央政府和部委站点 80 个, 地方政府站点 50 个, 第三方门户和垂直资讯站点 9 个, 政策研究机构站点 13 个, 美国政府站点 18 个, 印度政府站点 48 个, 芬兰政府站点 7 个.

由于源站点范围广、种类多, 数据采集子系统的设计面临诸多挑战. 首先, 这些网站的结构差异明显, 部分站点科技政策相关的数据集中在某个栏目, 其他站点则需要通过检索接口查询获取; 各站点的政策列表页面翻页机制不尽相同; 部分站点的内容由 Javascript 代码动态生成. 其次, 各站点的政策列表和政策详情网页结构差异较大, 无法开发一致的数据属性识别策略. 最后, 各站点的数据保护策略不尽相同, 常见的策略包括监控访问频度、账号认证、动态 URL (Uniform Resource Locator) 等.

2.1.1 基于 Scrapy 框架的爬虫设计

本文基于 Scrapy 框架和 Splash 实现网络爬虫. Scrapy 是 Python 开发的一个快速 Web 抓取框架, 用于抓取 web 站点并从页面中提取结构化的数据. Scrapy 是目前广泛应用的爬虫框架, 非常适合特定站点和栏目的定向爬取. Splash 是一个实现了 HTTP API 的轻量级浏览器, 支持 Javascript 渲染. Scrapy 框架通过 Scrapy-Splash 模块引入 Splash 软件, 弥补了 Scrapy 无法抓取网页动态内容的缺陷.

根据网站结构和网页结构对源站点进行分组, 比如大部分部委的网站结构相似, 可以分成一个组. 针对每组站点设计单独的爬虫, 实现站点数据的爬取和结构化信息提取.

2.1.2 基于 XPath 的数据属性识别

本文基于 XPath 实现网页的数据属性识别. XPath 使用路径表达式来选取 XML 文档中的节点或者节点集, 由于 HTML 和 XML 结构基本一致, 因此 XPath 非

常适合从网页中提取结构化信息. 例如 XPath 表达式"//*[@id='article_author']/text()"在网页中查找所"article_author"标签, 提取列表中各篇文章的作者姓名.

2.1.3 反爬设计

为了应对各站点的数据保护措施, 本文采取了 3 种反爬方法. 首先, 在爬虫工作时,设置了最小访问时间间隔,并动态调整页面请求时间间隔. 第二种方法是采用动态 UserAgent, 部分站点会根据 UserAgent 判断用户的访问是否合理, 为了避免误判, 使用 Python 的 fake_useragent 插件动态模拟 UserAgent. 第三种反爬方法是动态代理 IP, 部分站点会对频繁访问的 IP 暂时或永久的禁止, 针对这些站点爬虫维护一个可用的代理 IP 库, 每次请求随机从该库中选择一个 IP 访问.

2.1.4 爬虫配置和管理

网络爬虫必须适应网站改版、站点安全策略的变化, 因此本文支持对爬虫的行为进行配置, 包括初始 URL、搜索关键字、最大失败重试次数、结构化信息的 XPath 表达式配置等.

对于爬虫采集数据中发生的各种错误, 例如 404、502、Timeout 等错误, 系统进行记录、报警, 并提供了错误查询接口.

为了实现科技政策数据的增量更新, 实现了爬虫调度器, 定期启动爬虫对源站点进行新的数据采集操作. 为了多次采集造成数据重复, 将曾经爬取的网页 URL 保存在 Redis 中, 每次采集时进行比对过滤.

2.2 数据清洗子系统

数据采集子系统从互联网上收集的原始数据质量无法保证, 首先, 虽然数据采集子系统避免了相同 URL 网页的重复采集, 但是很多文献在不同站点反复出现, 导致了原始数据集存在大量数据重复. 第二, 由于大部分站点的数据是通过其检索接口采集的, 因此爬虫程序采集了大量与科技政策无关的数据. 第三, 部分数据存在关键属性缺失、属性错误、属性值格式不统一等缺陷. 原始数据中夹杂的脏数据会误导科技政策的研究, 因此必须予以清除.

2.2.1 基于 Simhash 的数据去重

Simhash 是一种 LSH 算法 (Locality-Sensitive Hashing, 局部敏感哈希)[11], 是目前最好的海量文本去重算法. Simhash 算法对文本经过分词、散列、加权、合并、降维等一系列计算, 最终为文本生成 64-bit 的信息指纹. 判断两个文本相似度的方法是对其 Simhash 值进行异或操作:

$$|hammingDist(Simhash(str1), Simhash(str2)| \le K \quad (1)$$

其中, *hammingDist* 为计算两个整数海明距离的函数, 即为两个整数二进制编码中不同的位数, K 是最大容忍的不同位数, 取值 3.

本文采用 Jieba 分词软件对文本进行分词, 基于词表去除停用词, 采用 TF-IDF(Term Frequency–Inverse Document Frequency)[12]算法进行权重计算并降维, 将文本表示为特征向量; 之后为每篇文献进行 Simhash 计算; 最后逐篇文本进行 Simhash 计算, 比较去重.

为了降低计算次数, 将文本的 64 位 Simhash 值均分为 4 份, 并建立 16 bit 索引进行存储. 分析可知, 这种方案的存储开销变为原来的 4 倍, 但是单个文本的相似度计算次数降为:$4\times 4n/2^{16}$,其中 n 为文献总量. 常规的两两比较计算次数整体为: $n\times(n-1)/2$, 因此整体计算次数约降为原来的 $1/2^{13}$.

2.2.2 基于机器学习的非相关数据清洗

本文采用逻辑回归算法[13]将爬虫采集的原始数据分为科技政策相关、非科技政策相关两类, 从而实现对非相关数据的清洗. 逻辑回归模型作为广义线性模型类别, 属于概率性回归, 主要用来推断两分类或者多分类应变量与多维解释变量的关系. 使用逻辑回归算法进行科技政策文本分类的流程:

(1) 构建训练集. 从爬虫采集的原始数据中选择 1000 篇科技政策相关的数据, 政策类型覆盖核心政策、政策解读、政策研究等各种类型; 并选择 1000 篇非科技政策相关的数据.

(2) 文本预处理. 对训练集文本使用 Jieba 分词软件分词, 根据词表去除停用词.

(3) 特征提取. 使用 TF-IDF 算法构建文本的特征向量, 并降维.

(4) 训练模型. 从 2000 篇标注的文本中随机选择 1000 篇进行模型训练, 并利用其他 1000 篇验证模型分类概率. 不断调整梯度下降等算法参数, 以达到理想的分类效果.

(5) 使用训练好的模型对爬虫采集的数据进行分类, 并清除非科技政策相关数据.

2.2.3 数据属性缺陷处理

对爬虫提取的结构化信息进行分析, 常见的属性

缺陷可以分成四类: 第一类缺陷是数据属性值缺失, 例如文献没有标题; 第二类缺陷是数据属性错误, 例如日期属性的值为一段描述文字; 第三类缺陷是多个属性之间违反完整性约束, 例如政策的发布日期、生效日期、失效日期违反了先后顺序; 第四类缺陷是不同文献的统一属性格式不统一, 例如日期格式五花八门, 对后续的统计分析造成障碍.

本文采取基于规则的方法结合人工参与, 来识别和校正数据属性错误. 对于前三类类缺陷, 系统定义一系列规则去识别缺陷; 如果标题和正文等关键信息缺失或者错误, 则丢弃改文献; 如果非关键属性缺失, 则依赖人工补充. 对于第四类缺陷, 系统采用正则表达式实现数据属性的规格化, 首先针对每个数据属性, 枚举所有格式的正则表达式, 例如日期格式的[0-9]{4}[-./年][0-9]{2}[-./月][0-9]{2}或者[0-9]{2}[/][0-9]{2}[/][0-9]{4}等; 然后针对每个文献的属性值, 与这些正则表达式进行模式匹配; 不同的格式采用不同的转换方式, 最终全部转换为标准格式.

系统对于数据属性错误标识、审阅修正保留了记录, 方便后续对这些操作进行跟踪评估.

2.3 数据分析子系统

2.3.1 基于规则的政策分类

科技政策研究需要对文献进行多种维度的分类: 按照国别和地区分类; 按照政策性质分成核心政策、政策解读、领导讲话、政策研究论文、政策法案、政策研究课题等类别; 按照政策手段可以分成财税政策、人才政策等类别; 按照政策层次可以分成中长期规划、具体政策等类别.

系统依据数据来源和文本特点实现了国别和地区、政策性质的分类. 政策的采集来源可以作为重要的分类依据, 例如不同国家、不同地方政府发布的政策采集来源是非常明确的; 政策研究课题信息则来源于政策研究机构; 政策研究论文则来自于科研论文数据库等.

另外核心政策具有很多明确的特点: 发文机构有确定的范围, 政策具有发文字号, 标题中一般包含决议、决定、命令 (令)、公报、公告、通告、意见、通知、通报、报告、请示、批复、议案、函、纪要等字眼.

2.3.2 基于 Apriori 算法的关联分析

科技政策之间存在替代、合并、规划与落实等许多关联关系, 如果能够发现这些关联关系, 并在用户浏览政策时以推荐、可视化图谱的形式进行展示, 对科技政策研究具有重要意义. Apriori 算法[14, 15], 是最有影响的挖掘布尔关联规则频繁项集的算法, 其核心是基于两阶段频集思想的递推算法. 本文基于 Apriori 算法, 以政策文本中所包含的关键词作为政策的特征描述, 并结合政策发布的时效性特点, 计算政策之间的关联关系. 具体的分析流程:

(1) 所有政策数据集合为 D (Data), 通过预设以及关键词提取得到的关键词库集合为 K (Keyword), 单个政策文本数据为 P(Policy), 三者可以抽象表示为:

$$D=\{P_1,P_2,\cdots,P_n\} \tag{2}$$

$$P=\left\{K_1,K_2,\cdots,K_{pn}\right\} \tag{3}$$

(2) 定义一个政策特征变量 S, 可表示为一组关键词的集合 $S=\{K_1,K_2,\cdots,K_s\}$, 需要注意 S 与 P 的区别: P 是某个政策文本中提取出的关键词的集合, 而 S 是所有关键词组成的集合. 如果 $S\subseteq P$, 则说明政策 P 包含政策特征 S, 政策与政策特征的包含关系表明 S 中的各关键词是相互关联的.

(3) 政策数据集合 D 中包含特征 S 的政策文本数据 P 的数量为该特征政策的支持数 σ_s, 则该政策特征的支持度 $support(S)$ 为:

$$support(S)=\frac{\sigma_s}{|D|}*100\% \tag{4}$$

其中, D 为所有政策数据的数量, 若 $support(S)$ 小于系统规定的最小支持度, 则 S 为不频繁政策特征集; 若 S 大于等于最小支持度, 则 S 为频繁特征集. 在本系统中, 除了统计计算得到的频繁特征集外, 还可以预设频繁特征集.

(4) 若有两个互不包含的政策特征 S_A, S_B, $S_A\Rightarrow S_B$ 记为特征关联关系, 这个关联关系的可信度为在 D 中包含了政策特征 S_A 的政策文本同时又包含了政策特征 S_B 的数量百分比, 特征关联可信度 $confidence(S_A\Rightarrow S_B)$ 为:

$$confidence(S_A\Rightarrow S_B)=\frac{support(A\cup B)}{support(A)}*100\% \tag{5}$$

如果 $confidence(S_A\Rightarrow S_B)$ 小于系统规定的最小可信度, 则它们为弱关联关系, 否则为强关联关系.

系统在得到频繁特征集集合和强可信关联关系集合后, 根据每个集合中的政策文本的发文时间以及发布机构字段来确定同一集合内的政策间的追溯关系.

2.3.3 统计分析

系统在数据采集和数据清洗的基础上实现了初步的统计分析功能. 系统支持统计每个省、每年发布的科技政策数量, 以此为基础支持从时域、地域两个维度进行统计分析. 支持分析指定区域发布科技政策数量随时间的变化趋势; 支持分析在一定时间范围内, 各地区发布的科技政策总量的对比.

3 成果应用

从 2018 年 10 月在中国科协正式上线应用以来, 科技政策库系统对 225 个互联网站点进行了数据采集; 并实现了一套图书的 OCR 识别入库, 即《中共中央文件选集: 1949 年 10 月–1966 年 5 月 (全五十册)》; 共计获取 564 749 条科技政策相关的原始数据; 经过数据清洗, 有效入库数据 404 083 条.

3.1 数据清洗统计

通过基于 Simhash 算法的去重清洗了重复数据 62 336 条, 通过基于逻辑回归分类方法清洗了非科技政策相关数据 94 706 条, 清洗标题和文本等关键属性缺失的数据 3624 条. 经过数据清洗之后, 有效入库数据 404 083 条.

为了验证数据清洗的效果, 本文从有效入库的文献中随机抽取 1000 篇文献, 进行人工的重复、非相关文献统计. 经过 10 次试验求平均值, 可知数据清洗之后, 数据重复率约为 0.07%, 非相关文献数量比率约为 0.6%.

表 1 科技政策库数据清洗效果

清洗操作	清洗数量
数据总量	564 749
基于 Simhash 的数据去重	62 336
基于逻辑回归的非相关数据清洗	94 706
关键属性缺失清洗	3624
非关键属性缺失或错误	8742
有效入库数据量	404 083

3.2 有效入库统计

对于有效入库的 404 083 条数据按照国别和政策性质两个维度进行了统计, 结果见表 2 和表 3.表 3 中的 177 423 篇核心政策中, 包括中共中央文件选集 4248 篇, 美国科技政策法案 8157 篇. 相关数据包括科技政策相关的领导讲话、科技政策解读、科技政策新闻等相关文献.

表 2 有效入库数据按国别分类统计

国别	政策数量
中国	376 592
美国	15 232
印度	9375
芬兰	2884

表 3 有效入库数据按政策性质统计

政策性质	政策数量
核心政策	177 423
相关数据	211 854
政策研究论文	5815
政策研究课题	52
政策研究报告	8939

3.3 关键 UI 页面

系统基于 Spring Boot 和 Javascript、Vue(一种 JavaScrip 前端开发框架) 等技术实现了 B/S 架构的管理功能和 UI, 图 2–图 4 展示了科技政策库系统的部分界面.

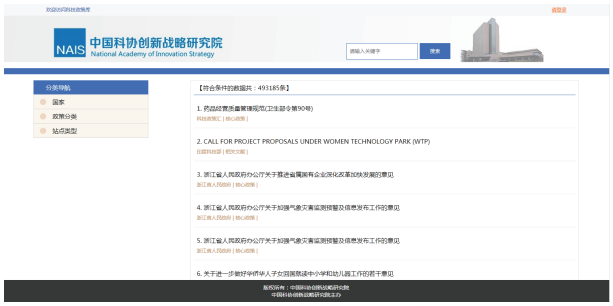

图 2 政策检索结果列表

图 3 政策在线阅读

图 4 政策发布趋势分析

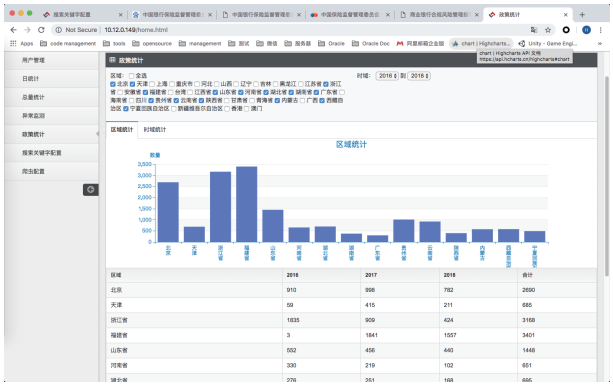

图 5 政策发布地区对比

4 结论与展望

科技政策库系统基于 Scrapy 框架针对大量异构站点设计了可管理的网络爬虫, 基于机器学习算法实现了数据去重、非相关数据识别、数据属性缺陷识别等数据清洗功能, 对有效入库的科技政策进一步进行了文本分类、关联关系分析, 系统基于 B/S 架构向用户提供了政策检索、在线阅读、统计分析等功能. 系统上线之后总计采集科技政策相关数据 564 749 条, 数据清洗之后有效入库 404 083 条数据, 为科技政策研究工作提供了坚实的基础. 下一步需要从国内外、历史文件等方面扩大数据采集范围, 引入众包等最新方法进一步提升数据清洗能力, 从自定义分析、数据可视化等方面丰富系统的统计分析手段, 以便更好地为科技政策研究提供支持.

参考文献

1 樊春良, 马小亮. 美国科技政策科学的发展及其对中国的启示. 中国软科学, 2013, (10): 168–181. [doi: 10.3969/j.issn.1002-9753.2013.10.016]

2 肖小溪, 杨国梁, 李晓轩. 美国科技政策方法学 (SoSP) 及其对我国的启示. 科学学研究, 2011, 29(7): 961–964.

3 NSTC & OSTP. The science of science policy: A federal research roadmap. Washington: The White House, 2008.

4 樊春良. 科技政策科学的思想与实践. 科学学研究, 2014, 32(11): 1601–1607. [doi: 10.3969/j.issn.1003-2053.2014.11.001]

5 陈光, 方新. 关于科技政策学方法论研究. 科学学研究, 2014, 32(3): 321–326. [doi: 10.3969/j.issn.1003-2053.2014.03.001]

6 樊春良. 科技政策学的知识构成和体系. 科学学研究, 2017, 35(2): 161–169. [doi: 10.3969/j.issn.1003-2053.2017.02.001]

7 李燕萍, 吴绍棠, 郜斐, 等. 改革开放以来我国科研经费管理政策的变迁、评介与走向——基于政策文本的内容分析. 科学学研究, 2009, 27(10): 1441–1447, 1453.

8 徐翔, 聂鸣. 我国科技创新政策研究综述. 科技进步与对策, 2005, 22(11): 178–180. [doi: 10.3969/j.issn.1001-7348.2005.11.066]

9 李萌. 大数据时代对我国科技情报事业发展的新思考. 中国软科学, 2016, (12): 1–4. [doi: 10.3969/j.issn.1002-9753.2016.12.001]

10 樊宇豪. 基于 Scrapy 的分布式网络爬虫系统设计与实现[硕士学位论文]. 成都: 电子科技大学, 2018.

11 Charikar MS. Similarity estimation techniques from rounding algorithms. Proceeding of the 34th Annual ACM Symposium on Theory of Computing. Montreal, Quebec, Canada. 2002. 380–388.

12 Bin L, Yuan G Y. Improvement of TF-IDF algorithm based on Hadoop framework. Proceedings of the 2nd International Conference on Computer Application and System Modeling. Paris, France. 2012. 391–393.

13 王济川, 郭志刚. Logistic 回归模型—方法与应用. 北京: 高等教育出版社, 2001.

14 Agrawal R, Srikant R. Fast algorithms for mining association rules. Proceedings of the 20th International Conference on Very Large Data Bases. Santiago, Chile. 1994. 487–499.

15 赵晨. 关联规则挖掘算法的研究及应用[硕士学位论文]. 西安: 西安电子科技大学, 2011.