开放获取期刊资源采集系统研究与实现

1. 引言
   1. 研究背景
      1. 开放获取简介

“开放获取”的首次定义出现于2002年布达佩斯开放获取计划，即可以在公共网络上被免费获取，允许任何用户对该文献的全文信息进行阅读、下载、复制、分发、打印、检索、超链接，支持爬行器收割并建立本地索引、用作软件的输入数据、用于其他任何法律允许的用途。而在使用这些文献时用户不存在财力、法律或技术上的障碍，只需在获取文献时保持其完整性，而对文献复制和发布的唯一限制，或者说版权在该领域的唯一作用就是给予作者控制其作品完整性及作品被正确理解和引用的权利[1]。随着开放获取运动的深入和互联网的发展，开放获取资源形成了不同类型。目前最为常见的是开放获取资源包括开放获取仓储(Open Access Archives)和开放获取期刊(Open Access Journals)等两种类型[4]。开放获取仓储(Open Access Archives)是遵循开放获取原则的一种电子版科技文献存储和检索的数据库。研究学者们通过向开放获取仓储提交已发表或未发表的研究成果，使得成果能迅速被其他学者学习并讨论，从而促进当前学科的发展。开放获取仓储中可以包括：电子文档、论文、课程资料、数据文件、声像文件、机构记录以及任何类型的数字文档。开放获取仓储主要分为两种类型：一种是收集某一特定学科或主题的学科知识库，另一种是由多个学术机构共同组成的机构知识库[6]。开放获取期刊(Open Access Journals)是遵循开放获取原则的期刊。开放获取期刊经过同行评审，并通过网络载体，对期刊资源进行描述。开放获取期刊与传统期刊的区别不在于期刊的载体是纸本还是电子，而在于对期刊的访问方式和访问权限。所有的用户可以不受价格和权限的限制免费从互联网上获取此类期刊的学术信息[7]。

近年来提供开放获取服务的期刊资源逐步增多。DOAJ上收录了来自136个国家的11000多个开放获取期刊，可检索的论文达到2280000多篇[2]，在中国科技期刊开放获取平台(COAJ)上收录的期刊为650个，可检索的论文数量接近142000篇[3]。随着开放资源数量和质量逐步提升，其价值越来越逼近主流商业学术资源。对于越来越多可以免费获取的学术资源，无论是学习交流还是二次研究都有非常重大的价值。诸如DOAJ之类的国内外开放获取期刊平台，已提供了论文的检索、链接等功能，基本上能满足学者们学习交流的目的。但这些开放获取期刊平台并不提供元数据的采集。对于开放获取资源的深层研究，如学科领域发展趋势研究、知识演化研究等，研究者需要自己寻找方法对元数据进行采集。对开放获取资源的采集方法的研究就是在这种背景下产生。

* + 1. 开放获取期刊资源采集

当前开放获取期刊资源采集方法可以总结为两种，一种是基于接口协议的元数据采集方法，即数据提供方和数据采集方遵循统一收割协议，数据提供方按照规定的格式存储数据，并提供数据采集接口，数据采集方调用统一的接口进行元数据的采集。另一种是基于Web信息的元数据采集方法，此方法针对没有统一接口协议，以Web页面为载体来描述元数据。用户通过对Web页面的访问来查看元数据。此类资源采集需结合Web信息采集技术。

基于接口协议的元数据资源采集的特点在于，数据的提供方和采集方都遵守特定的元数据收割协议。其中具有代表性的是OAI-PMH(Open Archives Initiative Protocol for Metadata Harvesting)协议接口[4]。基于OAI协议的元数据获取框架主要是由三部分组成：数据提供方(Data Provider)、服务提供方(Service Provider)以及注册服务器（Registrar Server）。数据提供方负责元数据的生成，即按照统一的规范，在本地建立本地元数据仓库（Repository），并且以OAI的响应(response)向服务提供方发布元数据。服务提供方和数据提供方主要通过6个OAI指令，即Identify、ListMetadataFormat、ListSets、ListIdentifiers、ListRecords、GetRecord， 来对元数据进行收割、整理加工，最终提供增值服务。注册服务器是对服务提供方和数据提供方进行管理的模块。数据提供方与服务提供方的交互过程为：服务提供方向数据提供者发起一个HTTP请求；数据提供方对此HTTP请求进行解析，判断是否为OAI-PMH协议中的六个有效请求之一；若为有效请求，数据提供方根据请求进行相应，返回元数据结果，若不是有效请求则返回错误信息。

基于协议的元数据采集特点在于：数据提供方遵循统一的协议，并提供统一的数据采集接口，提高了元数据采集收割的效率。但不足之处是数据提供者必须要遵守收割协议，对于网络上存在的大量的非统一接口的数据来源，无法进行采集，具有一定的局限性。

基于网页信息的元数据指的是不提供统一的元数据收割协议，由数据提供方自定义描述规则，在网页中进行展示的元数据。元数据采集方通过对元数据网页进行解析，编写特定抽取规则来抽取其中的元数据。网页信息采集方法从不同的角度可以将分为不同种类[5]。最常见的分类方式是根据包装器形成方式的不同将网页信息采集分为基于自然语言处理的网页信息抽取、基于本体的网页信息抽取、基于包装器归纳方式的信息抽取、基于HTML页面结构分析的网页信息抽取、基于Web查询的网页信息抽取和基于页面视觉特征的网页信息抽取[6-11]。不同Web信息采集方法适用于不同类型的元数据资源，本文将在第二章具体讨论。

* 1. 研究内容

王思丽等[]在《开放知识资源等元数据自动采集策略研究》中指出开放获知识资源的元数据格式繁多，其中最常见的是通过网页展示给用户，而不同的资源组织形式千奇百样，为第三方用户自动采集增加了一定的技术难度。本文以专家遴选的134个国内开放获取期刊网站和15个国外开放获取仓储为对象，主要探究：

(1)对开放获取期刊资源进行充分调研分析，进一步细化开放获取资源的主要特点；

(2)对各元数据采集方法进行比较分析；

(3)结合开放获取资源特点以及各类采集方法，提出针对开放获取资源采集的采集方案；

(4)根据提出的方案，通过计算机软件技术开发出开放获取期刊采集系统。

* 1. 研究目的与意义

当前，元数据资源采集方法有多种，商用的网页信息采集系统也层出不穷，这些方法和系统适用于结构单一的信息资源采集。但对于组织形式多样化的开放获取期刊资源，进行采集效果并不佳。因此，本文主要是通过调研开放获取期刊资源以及当前的元数据采集方法，提出针对开放获取期刊资源采集的方案，并使用计算机技术实现开放获取期刊资源采集系统，以此达到以下几个目的：1.进一步细化开放获取期刊资源多样化的特点；2.提出针对开放获取期刊资源的采集方案；3.设计并开发一套开放获取期刊采集的系统，有效的对开放获取期刊资源进行采集。

研究具有一定的理论意义和实践意义：

(1)理论方面：通过对开放获取期刊采集调研分析，细化开放获取期刊资源特点，并结合当前信息采集方法，总结出一套针对开放获取期刊资源采集的方案，为开放获取期刊资源采集进行指导。

(2)实践方面：通过设计开发开放获取期刊资源采集系统，可以提供开放获取期刊资源采集服务，在实际应用角度为期刊资源采集提供方便，为进一步研究提供数据基础。

* 1. 论文框架

文章分为五个章节，首先对研究背景、研究内容、研究目的与意义进行交代，再对开放获取期刊特点和当前信息采集的方法和系统进行对比分析，总结当前开放获取期刊采集面临的问题，进而提出开放获取期刊资源采集方案，然后对开放获取期刊资源采集系统进行需求分析和系统设计，接着使用计算机技术对系统进行开发与测试，最后对本文在开放获取期刊采集上的研究进行总结与展望。具体章节安排如下：

第一章，引言。主要阐述文本的研究背景、研究内容、研究目的与意义以及本文的框架。

第二章，开放获取期刊资源采集。主要对开放获取期刊资源的特点进行进一步总结，对当前信息采集方法和采集系统进行对比分析，总结开放获取期刊采集面临的主要问题，并提出采集方案，为开放获取期刊采集系统提出指导。

第三章，系统分析与设计。根据上一章提出的开放获取期刊资源采集方案，对开放获取期刊资源采集系统进行需求分析和系统设计，并介绍系统开发主要采用的相关技术。

第四章，系统实施与运行。先介绍开放获取期刊资源采集系统开发中的关键技术，如网页结构检查、多线程采集以及非关系型数据库使用等。接着选择典型开放获取期刊，对其资源进行采集，测试并评价系统功能。

第五章，总结与展望。对现有的研究和系统功能进行总结，分析不足和待解决的问题，在此基础上对未来进一步的研究工作作出展望。

1. 开放获取期刊资源采集

2.1开放获取期刊资源特点与分类

**2.1.1 开放获取期刊资源的特点**

本文通过对专家遴选的134个国内开放获取期刊网站和15个国外开放获取期刊仓储进行调研，梳理了当前开放获取期刊资源的特点主要有以下几点：

1. 描述粒度细。开放获取期刊资源中的元数据包含了众多字段，如期刊的元数据字段，包括：期刊名、年、卷、期等；论文元数据，包括：题名、其他题名、关键词、英文关键词、摘要、英文摘要、作者、机构、唯一标识号、总页数、起止页、参考文献数量等。相比于其他网络资源，开放获取资源所需采集到元数据更细。
2. 描述复杂。开放获取期刊资源元数据的字段众多，而这些字段通常是以不同的形式展现在网页中，造成了开放获取资源元数据描述的复杂性。主要表现为部分元数据字段在网页中是按照单个字段进行展示，如文章的标题、摘要等是以一个标签单独展示。而部分元数据则是多个字段组合成一条文本信息进行展示，如文章的年卷期等是组合成了一个文本信息进行展示。
3. 描述载体结构多变。在对国内开放获取期刊资源调研过程中发现，部分期刊网站的资源展示页面，在不同时期所采用了不同的网页模板。在结构发生变化的开放获取资源网站中，一般会存在1-3套不等的网页模版。而其他网络资源，如电商平台、论坛等，通常只采用统一的网页模版。

**2.2.2 开放获取期刊资源分类**

开放获取期刊资源以不同的组织形式在不同的网页中进行展示。通过对国内外开放获取期刊资源的调研和特点总结，根据开放获取期刊资源在网页上的组织形式，将开放获取期刊资源分为：单一型资源和组合型资源。

单一型资源是指网页的一个HTML标签仅展示一个元数据字段信息的资源，该类资源包含的元数据有期刊名称、文章的中英文标题、摘要、关键字、全文获取链接等。此类资源信息揭示简单明了、层次清晰。在对该类资源进行采集时，只需要提取HTML标签对中的信息即可。

组合型资源是指网页的一个HTML标签中封装了多个字段信息的资源，多个字段通常是组合成为一个文本信息进行展示，如期刊的年卷期字段通常是组合成一条文本在网页中进行展示。此类资源信息揭示较复杂。在对该类资源进行采集时，提取HTML标签对中的文本信息之后，还需要进一步对文本进行处理，进而得到各个元数据字段信息。

**2.2元数据采集方法**

开放获取期刊资源包含了期刊元数据、论文元数据、论文全文获取链接等，期刊资源的采集重点在于对各元数据的采集，因此本文对元数据采集方法进行分析总结。当前元数据采集方法主要分为两种，一种是基于接口协议的元数据采集方法，即数据提供方和数据采集方遵循统一收割协议，数据提供方按照规定的格式存储数据，并提供数据采集接口，数据采集方调用统一的接口进行元数据的采集。另一种是基于Web信息的元数据采集方法，此方法针对没有统一接口协议，以Web页面为载体来描述元数据。用户通过对Web页面的访问来查看元数据。此类资源采集需结合Web信息采集技术。下面就这两种方法的实现以及在开放获取资源采集中的应用进行讨论，为进一步提出针对开放获取资源元数据采集方法提供指导。

**2.2.1基于接口协议的元数据采集方法**

基于接口协议的元数据资源采集是对遵循收割协议并提供统一采集接口的元数据进行采集。该方法特点是数据的提供方和采集方都遵守特定的元数据收割协议。其中具有代表性的是OAI-PMH(Open Archives Initiative Protocol for Metadata Harvesting)协议接口[1]。基于OAI协议的元数据获取框架主要是由三部分组成：数据提供方(Data Provider)、服务提供方(Service Provider)以及注册服务器（Registrar Server）。数据提供方负责元数据的生成，即按照统一的规范，在本地建立本地元数据仓库（Repository），并且以OAI的响应(response)向服务提供方发布元数据。服务提供方和数据提供方主要通过6个OAI指令，即Identify、ListMetadataFormat、ListSets、ListIdentifiers、ListRecords、GetRecord， 来对元数据进行收割、整理加工，最终提供增值服务。注册服务器是对服务提供方和数据提供方进行管理的模块。

基于协议的元数据采集特点在于：数据提供方遵循统一的协议，并提供统一的数据采集接口。该方法已十分成熟，能保证元数据采集的准确性与采集效率，但该方法需要数据提供者必须要遵守收割协议，对于网络上存在的大量的非统一接口的数据来源，无法进行采集，具有一定的局限性。

**2.2.2基于Web信息的元数据采集方法**

基于Web信息的元数据采集方法是指对不提供统一的收割协议的元数据进行采集。元数据由数据提供方自定义描述规则，在Web页面中进行展示。元数据采集方通过对元数据网页进行解析，编写特定抽取规则来抽取其中的元数据。

Web信息采集方法从不同的角度可以将分为不同种类[2]。最常见的分类方式是根据包装器形成方式的不同将网页信息采集分为基于自然语言处理的网页信息抽取、基于本体的网页信息抽取、基于包装器归纳方式的信息抽取、基于HTML页面结构分析的网页信息抽取和基于Web查询的网页信息抽取 [3-8]。表1分别对各种采集方法进行对比，并对其在开放获取资源采集中的应用进行分析。

表 1 各采集方法比较分析

|  |  |  |  |
| --- | --- | --- | --- |
| 方法名称 | 代表系统 | 包装器形成方式 | 适用元数据 |
| 基于自然语言处理的网页信息抽取方法 | 基于WHISK[9]算法的系统 | 依赖自然语言处理技术和语意字典，建立短语之间的关系，所产生的抽取规则通常基于语法约束和语义约束[10]。 | 适用于文本且句子完整的元数据。 |
| 基于本体的网页信息抽取方法 | SNAX[11]系统 | 建立本体集，对网页进行解析，将处理后的网页信息传递给计算模块，根据本体集来计算信息的相关度，进而保留相关信息。 | 适用于特定主题或领域的元数据。 |
| 基于包装器归纳的信息抽取方法 | WIEN[12]系统 | 根据事先由用户标记的样本实例，应用机器学习方式归纳算法，生成基于定界符的抽取规则。 | 适用于结构清晰，信息较为复杂的元数据。 |
| 基于HTML页面结构分析的网页信息抽取方法 | RoadRunner | 将网页解析成结构树，通过自动或半自动的方式产生抽取规则，定位主题信息，将网页信息抽取转化成对解析树的操作[13]。 | 适用于结构清晰且信息单一的元数据 |
| 基于Web查询的网页信息抽取方法 | Web-OQL[14] | 将Web文档解析成一棵抽象的HTML语法树，根据信息在页面代码中的唯一的标记，写出合适的类似于SQL语句的Web查询语言对语法树进行查询。 | 适用于信息单一且标识明确的元数据。 |

通过以上对Web信息采集方法的分析，可以从各方法适用元数据的角度将其分大致分为两类：(1)单一节点元数据采集方法，包括基于Web查询和基于HTML页面结构分析的网页信息抽取方法。该类方法主要是对结构化网页中封装的信息进行提取，需要用户对网页结构进行分析，找到所需提取元数据在网页中的定位，再通过爬虫工具对元数据字段进行采集。(2)文本类元数据采集方法，包括基于自然语言处理、基于本体和的基于包装器归纳的网页信息抽取方法。该类方法主要对文本信息中包含的元数据进行提取，通过建立语意字典、本体知识库，或者归纳学习生成抽取规则的方法，对一段文本信息中所包含的元数据字段进行提取和采集。

**2.3网页信息采集系统**

上一节从采集方法的角度对当前资源采集方法进行了讨论，这一节将从系统等角度对比分析几款典型的网页信息采集工具，并从易用性和准确性两个方面分析各采集系统在开放获取期刊资源采集中的应用。

表2国内外典型的网页信息采集器比较和分析

|  |  |  |  |
| --- | --- | --- | --- |
|  | 特点分析 | 优点 | 应用于开放获取期刊资源 |
| 八爪鱼采集器[35] | 八爪鱼采集器通过内嵌的浏览器与用户进行交互，用户在内嵌浏览器中点击所需爬取的信息，系统记录所选取信息在网页中的唯一标识，并通过此标识来提取此系列其他网页中的信息。 | 简单的可视化交互，无需用户分析网页结构。可对不同深度的网页进行循环抽取。可进行云采集但需要付费，可设置定时采集。 | 网页中唯一标识发生变化，则无法提取信息。 |
| 火车采集器[36] | 火车采集器依赖于用户对网页的解析，通过用户查看网页源代码，找到所需提取信息的唯一标识，或者找到能分割采集区域的标签字符串，或者根据所提取信息的网页标签中的规律编写正则表达式来生成抽取规则。 | 可通过正则表达式扩充元数据的抽取规则，来应对网页中唯一标识变化的情况，提高抽取的准确度。支持丰富的拓展。 | 非常依赖用户对网页结构的识别，对正则表达式要求较高，不能定时采集，不如八爪鱼采集器易操作。 |
| ET采集器[37] | ET采集器是通过用户对网页源代码进行分析，人工识别需要提取信息的网页源代码，将不变的区域保留，将变化的区域设置为变量标记，以此来生成类似正则表达式的抽取规则。通过内置的测试工具能快速的查看当前抽取规则的抽取结果，并反复调整抽取规则，来获取最佳的抽取规则。 | 通过自动生成类似于正则表达式的抽取规则可提高抽取的准确度，内置的测试工具可以快速的查看抽取结果，并对抽取规则进行调整。 | 需要用户反复对网页源代码进行分析。 |
| 网络矿工采集器[38] | 网络矿工采集器将用户所需采集的网页分为导航页和采集页，网页URL可以通过系统所提供的按钮设置参数。对需要提取的信息，系统需要用户通过对网页源码的解析，找到所需要提取信息所在的页面代码块，通过设置所需提取信息的起始位置和终止位置，来设定类似正则表达式的抽取规则。 | 类似正则表达式的抽取规则使采集准确度高，系统提供大量的按钮和提示，减少了用户使用的难度。 | 需要用户通过识别网页结构寻找抽取信息的起始位置。 |
| Import.io[39] | Import.io是国外一款较流行的网页信息采集系统，采用的是B/S架构。用户通过输入URL，系统自动对网页中的信息进行采集。用户也可以通过简单的点击操作，选择其他想要抓取的信息。然后通过选择保存数据的格式，将采集到的数据保存在本地。 | 交互简单，易操作，对用户要求非常低，工作量少，适用于列表类的网页。 | 多数开放获取资源的元数据页是非列表类型，采集准确度不高。 |

国内外网页信息采集工具基本上是采用的是通过页面结构解析，归纳形成包装器的网页信息采集技术，主要是针对结构化很强的网页来进行信息的抽取，抽取过程中不需要大量的训练样本，只需用户对网页进行初步的分析，定位所需抽取的信息，自动或手动的生产抽取规则。

从易用性的角度来看，Import.io和八爪鱼采集器提供内嵌的浏览器，用户只需在浏览器中点击所需采集的信息就能自动生成抽取规则。而其他采集器都需要用户分析网页源代码，因此易用性上，Import.io和八爪鱼采集器最优。在需要用户分析网页的采集器中，网络矿工采集器提供了大量的按钮与提示信息供用户通过点击生成抽取规则，减少的用户使用的难度。ET采集器提供的测试工具可以使用户快速的查询到当前抽取规则的抽取结构，进而调整抽取规则，在一定程度上也提高的了系统的易用性。因此网络矿工采集器和ET采集器在易用性上次之。火车采集器提供了多种方式来生成抽取规则，如前后截取、正则提取、XPath提取、JSON提取等，有些方式需要用户有一定的专业知识背景才能生成有效的抽取规则，因此对于普通用户的使用有些难度。

从准确性的角度来看，本文通过对某一期刊网站进行试验发现，对于单个页面或者多个具有相同编码特征时，如文章标题的HTML标签中的唯一标识都为“J\_biaoti”，经过多次调整采集规则，所有采集器采集都能准确采集。但当此期刊网页中的编码特征变化，如文章标题的HTML标签中的唯一标识由原来的“J\_biaoti”变成了“J\_ZhaiYao\_BiaoTi”，各采集器都无法采集到相关信息。因此采集器的准确性除了与采集规则有关之外，与网页结构的统一性也有非常大的关系。当网页结构统一，通过调整采集规则，各采集器的准确率会非常高，但是网页结构不统一，则需对不统一的网页重新编写采集规则。

通过以上分析得出，各类采集器基本上能适用于网页结构一致的开发获取资源。但对于多种网页结构的开放获取资源，则需要编写多个采集规则。而且由于采集器在采集开放获取资源信息时，只能采集标签对中的信息，而多个元数据信息可能存在于一个标签对中，因此还需要对采集器采集到的文本进行分割、删减等二次处理，才能得到最终需要的元数据。

2.4开放获取期刊资源采集方法

针对遵循收割协议的开放获取期刊资源，基于接口协议的元数据采集方法已十分成熟，但具有一定的局限性，因此本文重点讨论不遵循收割协议且没有统一采集接口的开放获取期刊资源。通过对开放获取期刊资源的特点进行分析，根据资源的组织形式将开放获取期刊资源分为单一型资源和组合型资源。而当前基于Web信息元数据采集方法大致可以分为单一节点元数据采集方法和文本类元数据采集方法。接下来将结合开放获取期刊资源的特点，分析不同类型的期刊资源适应的采集方法。

对于单一型资源，该类资源的元数据是将单个元数据字段作为一条信息封装在HTML的标签对中。在采集过程中，只需准确采集HTML标签对中的信息即可。因此可以采用单一节点元数据采集方法，即基于HTML页面结构分析的网页信息抽取或基于Web查询的网页信息抽取。通过爬虫工具将Web页面解析成HTML语法树，找到元数据所在标签对的标识，按照此标识和标签在文档中的位置编写抽取规则，即可对单一性资源进行采集。

对于组合型资源，该类资源的元数据是多个元数据字段组合成一条文本信息封装在HTML标签对中。在采集过程中，除了将封装在HTML标签对中的文本信息进行采集外，还需要对采集到的文本进行进一步分析，提取文本中包含的元数据字段信息。因此，可以先采用单一节点元数据采集方法，即基于HTML页面结构分析或基于Web查询的网页信息抽取方法，将标签对中封装的文本信息进行采集，接着使用文本类元数据采集方法，即基于自然语言处理或基于本体或基于包装器归纳方式的信息抽取方法，对文本信息进行分解和提取，进而得到最终需要的元数据字段。

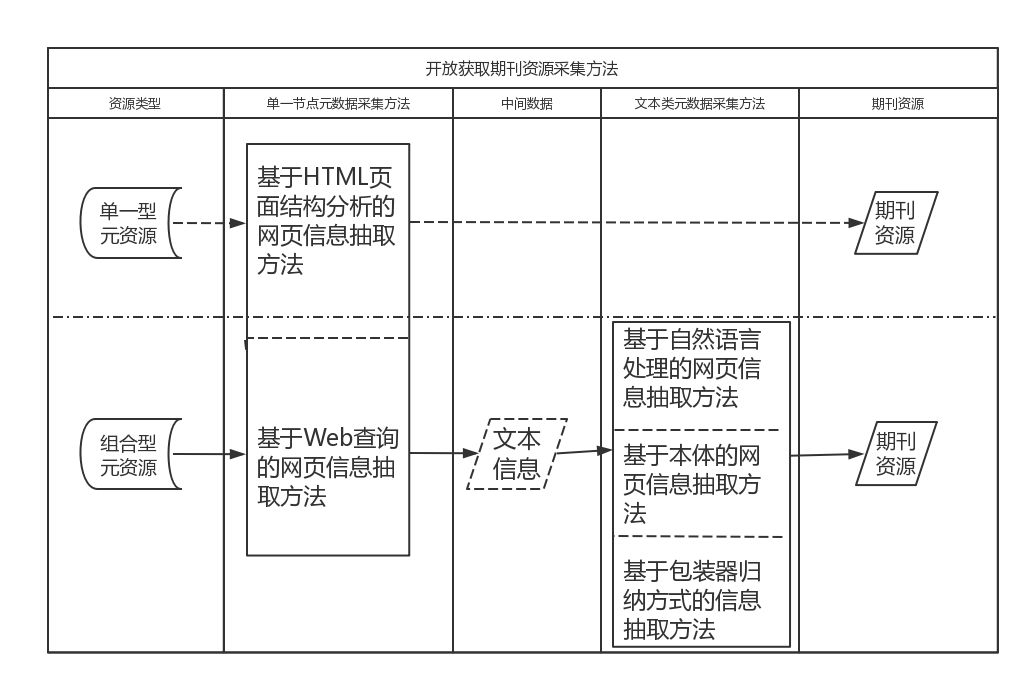


图 1开放获取期刊资源采集方法

第三章 开放获取期刊资源采集系统分析与设计

3.1开放获取期刊资源采集系统

随着互联网的发展，期刊的发行方式从传统的纸质版发展为电子版期刊与纸质版期刊并存的现状。而开放获取运动的不断推广，越来越多的期刊加入到了开放获取的行列中来。在互联网上这些大量高质量、可免费获取的学术资源如果能得到很好的利用与研究，将会为科学研究带来巨大的帮助。但是，由于各开放获取期刊均由各自的机构单位来维护，每个单位对期刊在网络上的描述形式各不一样，甚至同一期刊在不同的时期都采取了不同的描述方式，可谓是“千刊千面”，这给开放获取期刊资源的采集带来了很大的困难。传统的资源采集方法是人工对网页进行解析，找出所需采集字段在网页中的路径，再通过爬虫工具进行采集。该方法工作量大，还需要一定的专业背景，对于期刊中结构发生变化的网页无法预知，进而导致采集数据不全。

开放获取期刊资源采集系统需要达到以下几个目标：

(1)便捷性。所有用户的操作必须是在图形化的界面上，无需用户识别网页源代码，尽量降低用户专业背景要求。系统的操作应该尽量的简洁化，在满足基本需求的前提下减少不必要的操作，并对各类操作给出明确的提示，指导用户进行操作。

(2)结构检查。开放获取期刊资源采集比较困难的重要原因是期刊的结构多变，一个期刊在不同的时期所采取的描述方式有所不同，因此一般采集工具仅凭某一个页面所生成的采集规则无法采集到所有的期刊资源。结构检查的功能是在采集过程中检查网页结构是否发生变化。如果发生变化，则将变化后的页面在系统中反馈给用户，让用户对结构变化后的页面进行操作，形成新的采集规则，进而采集到更全面的数据。结构检查的具体实现过程将在后文中详细介绍。

(3)高效性。一个期刊的文献资源少则上百篇多则上万篇，开放获取期刊资源采集系统可能会需要对期刊所有的文献资源进行采集，大量的采集任务对系统采集的效率提出更高的要求，系统考虑如何在更短的时间里采集更多的资源信息。因此，多线程采集等提高效率的做法是系统所需要采取的。

(4)灵活的数据存储方式。不同的用户在使用开放获取期刊资源采集系统时，采集需求各不相同，所采集的字段多种多样。传统的关系型数据库固定的数据结构模式已经满足不了多样化的采集需求，因此需要引入非关系型数据库，提高数据存储的灵活性。

针对以上几点的考虑，开放获取期刊资源采集系统需要有一套可视化信息采集的界面，一套适用的采集策略以及一套灵活的数据存储策略。下面在讨论开放获取期刊资源采集系统具体需求之间，将介绍系统将采取的主要技术。

3.2相关技术

3.2.1Java

开放获取期刊资源采集系统的开发必须选择一套编程语言来进行开发。Java因具有简单、健壮、安全、高性能、可移植等众多特性而被众多人使用。Java语言提供大量的API和第三方库来拓展其应用领域，使得使用Java进行应用的开发变得更加简单方便。Java面向对象的编程思想以及其所包含的封装、继承和多态这三大特征已被广泛的认可和使用，也使其成为最受欢迎的编程语言之一。Java语言在各机构颁布的编程语言排行中都位列前卫[https://www.oschina.net/news/79650/2017-top-programming-languages]。

Java语言支持十分丰富的数据结构，为系统开发带来了很大的便利性。以下文本介绍几种系统开发过程中使用较多的数据结构。

ArrayList<T>，动态数组，可以是任意基本数据类型或对象的数组，无须制定数组大小，数组大小动态改变。

HashMap<T,T>，映射表，元素以键值对形式存在，并且不允许有重复的键。

LinkedHashSet<T>,顺序集合，是Set接口哈希表和链接列表实现，主要能顺序存储不重复的基本数据类型或对象，可以自动的对数据进行去重，保证集合中的数据是唯一的。

3.2.2可视化布局工具JavaFX

开放获取期刊资源采集系统首先需要一个可视化的工具，除了能提供浏览器浏览网页的功能外，还需供用户在页面上进行元素选择、数据展示、数据下载等可视化的操作。此外丰富的页面展示组件和页面布局也是必不可少的，因为这能够使系统更加美观和易操作。

JavaFX是一个强大的图形和多媒体处理工具包集合，它允许开发者来设计、创建、测试、调试和部署富客户端程序，并且和Java一样跨平台[http://www.javafxchina.net/blog/2015/06/doc01\_overview/]。JavaFX除了提供丰富的外观和样式，如基本的UI控件：标签、按钮、单选框、复选框、文本框、滚动条、表格视图、提示信息等，还提供内嵌浏览器组件，并可以通过相关API来实现web查看器和完整的浏览功能。WebView是一个使用了WebKitHTML技术的Web组件，可用于在JavaFX应用程序中嵌入Web页面。在WebView中运行的JavaScript可以方便地调用Java API，并且Java API也可以调用WebView中的JavaScript。JavaFX满足了开放获取资源采集系统可视化的需求。

3.2.3网页解析工具Jsoup

用户在可视化的界面上选择需要采集的元数据资源时，系统需要同时形式采集规则，以及在用户选择完毕之后，系统需要通过采集规则对开放获取期刊资源进行批量化的采集。以此开放获取期刊资源采集系统需要一个网页解析工具来对网页进行解析，并根据采集规则来定位和采集元数据资源。

Jsoup 是一款Java 的HTML解析器，可直接解析某个URL地址、HTML文本内容。它提供了一套非常省力的API，可通过DOM，CSS以及类似于jQuery的操作方法来取出和操作数据。[http://baike.baidu.com/link?url=8O8q8A7uvmFVWO83r9IsFhnYF-oejalUs9OxBkW6uCg5qKdN\_sCY-8zCCINPXmeS2\_ulHhTb5pA9fUahuRyUL\_] Jsoup的主要功能包括解析给定的URL所指向的HTML，通过DOM或者CSS选择器来查找或抽取数据，以及对HTML元素进行操作等。Jsoup提取网页元素的过程是先将网页解析成Document类型的对象。然后通过其丰富的选择器，即类似于SQL的API，对HTML文档进行搜索。进而将匹配到的节点存放到Elements或Element对象中，最后通过tag、text、absUrl等方法抽取节点中的元数据信息。

Jsoup的强大之处在于其对文档的搜索能力，其提供的选择器有众多方法对文件进行搜索，如[attr=value]、[attr^=value]、[attr$=value]、[attr\*=value]来匹配属性值等于、开头、结尾或包含属性值来查找元素，[attr~=regex]用于使用正则表达式来匹配属性值的元素。选择器还可以通过组合的方式来选择节点，如“root>grandparent[atrr=value]>parent[attr^=value]>child[atrr\*=value]”可以选择该路径下的元素集。此外选择器还提供伪选择器的语法，如has(tag) 表示匹配含有该tag的元素，contains(text)表示匹配含有text文本的元素集，matches(regex)表示匹配文本符合制定正则表达式的元素集。

Jsoup丰富的搜索和采集功能满足开放获取期刊资源采集系统对网页中元数据的采集需求。

3.2.3非关系型数据库MongoDB

在开放获取期刊资源采集系统的使用中，不同的用户因为需求的不同会采集不同的元数据字段。关系型数据库在使用前，需要确定好数据存储模式，即数据的字段和类型，适用于字段固定的或者变化较少的数据存储。开放获取期刊资源采集系统所采集的字段数目和类型都不确定，因此无法使用关系型数据库。

MongoDB是一个面向文档(document-oriented)的数据库，不是关系型数据库，不采用关系模型是为了获得更好的拓展性。与传统的关系型数据库相比，MongoDB不在有“行”的概念，取而代之的是更为灵活的“文档”(document)模型，不再有预定义模式(predefined schema)，文档的键(key)和值(value)不用再是固定的类型和大小[mongodb权威指南]。文档是MongoDB中数据的基本单元，是一系列键值对的有序集。如{“key”:“value”}，这个文档中只有一个键值对，键key对应的值为value，大多数文档中会包含多个键值对。集合是MongoDB中类似于表的概念，是一系列文档的组合。MongoDB提供关系型数据库一样的数据数据操纵语言，如创建、读取、更新和删除操作。MongoDB也提供丰富的API来适配各种编程语言，其中也包含Java API。MongoDB Java API提供了常用的数据库操作功能，如getDatabase()连接数据库，createCollection()/getCollection()创建／获取集合，insertOne()/insertMany()插入一条／多条文档，find()/find(doc)查找文档等等。MongoDB丰富的拓展型和强大的API为开放获取期刊资源采集系统的开发提供了很好的数据存储支持。

3.3开放获取期刊资源采集用户行为分析

在对开放获取期刊资源采集系统进行具体需求分析之前，本文根据前期对开放获取期刊资源采集的调研经验，对开放获取期刊资源采集的用户行为进行分析，以及对其进行定义，以便之后更好的分析系统的需求。

期刊网站的网页根据在采集过程中作用的不同可以分为不同的类型，本文将其分为种子页面、中间页面和详情页面。其中种子页面是指采集的入口页面，此类页面中包含了该期刊所有资源的链接入口。比如期刊网站的“过刊浏览”页面，包含了该期刊各年份或者各卷期的链接，从该页面开始进行层层点击，就可以访问到该期刊所有的资源。详情页面是指期刊网站最终展示元数据资源的页面，该页面展示了各种论文元数据，如标题、作者、摘要、关键字等，以及期刊元数据，如期刊名称、期刊的年卷期等。详情页面是采集者根据采集需求选择其所需要的元数据资源的页面。中间页面是指从种子页面到详情页面之间所经过的页面，如期刊每期的展示页等。

资源采集行为可以分为批量采集和单个采集。批量采集使用的场景是采集者需要对当前页面中某一类型的所有资源同时进行选择和采集。而单个采集是指采集者只对当前页面中一个资源进行采集。比如，在某个期刊的种子页面中展示了该期刊每期的链接，而采集者对所有期链接里面的资源内容进行采集，此时用户的采集是批量采集，而如果只采集某一期里的资源内容，则称为单个采集。

开放获取期刊资源最终存储与展示是以论文为单位，数据库中的记录是由论文元数据、期刊元数据以及全文下载等资源组成，而论文元数据、期刊元数据和全文下载等资源则是根据采集者的需求而定。

3.3开放获取期刊资源采集系统需求分析

3.3.1系统业务需求

(1)网页浏览功能

开放获取期刊资源归根结底是一种网络资源，而开放获取期刊资源采集系统是一个需要与采集者进行交互系统，因为每个采集者都有不同的资源采集需求，如采集不同的期刊或者采集不同的元数据资源，采集者通过输入、点击、选择等操作来采集其所需要的资源。因此开放获取期刊资源采集系统必须具备类似于浏览器的网页浏览功能，以供采集者能根据自己的需求，浏览、选择和采集不同期刊或不同元数据资源。

(2)元数据选择功能。开放获取期刊资源采集系统最终是要采集期刊资源的元数据信息，而采集什么样的元数据是由采集者的需求而定，因此系统需要提供采集者选择元数据的功能。元数据选择根据采集页面类型的不同可以分为不同的功能。对于种子页面和中间页面中的链接，元数据的选择根据资源采集行为，可以对应分为批量元数据选择和单个元数据选择。批量选择对应批量采集，采集者对页面中某个节点使用批量选择之后，系统会自动选择当前页面与选中节点同类型的节点。单个选择对应单个采集，系统只会选中采集者选择的节点。对于详情页面，选择的是所需要采集的元数据字段，根据元数据资源类型可以分为单一型元数据选择和组合型元数据选择。如果选择的资源是单一型资源，则系统需要提供元数据字段详情填写功能，来标识采集者所采集的元数据信息。如果选择的资源是组合型资源，则系统提供组合型资源分解功能，将其分解成单一的元数据并填写相关字段详情。此外，对于采集者选择的所有元数据，系统需要进行高亮显示。

(3)元数据采集时字段详细信息填写功能。由于不同的采集者元数据资源采集需求不同，为增加系统的灵活性，系统提供采集者自定义字段详情的功能。采集者在期刊网站的详情页面上选择了需要采集的元数据字段后，需要对该字段进行自定义，其中包括字段名称、字段类型以及字段是否是必须等，以此来标识该字段，以便系统对其进行存储和展示。字段名称为文本框，供采集者填写。字段类型为下拉选框，包括“文本”、“链接”等选项。字段是否必须为下拉选框，包括“是”和“否”两个选项。本文认为采集的资源需要有一个或多个字段来作为必须字段，以此来衡量采集的资源是否有意义，类似于关系型数据库里键的概念。如果必须字段为空，则说明该条记录不符合要求。

(4)组合型资源分解功能。对于单一型资源，可直接填写字段的详细信息，而组合型资源是由多个单一型资源组合为一条文本信息，因此无法直接填写字段详细信息，需要对资源进行分解为多个单一型资源之后，再填写字段详细信息。组合型资源分解的解决方案是由用户在组合型文本中插入分割符，如{}，分割符之间所包含的信息则是需要提取的单一型资源。系统自动根据分割符的位置提取单一型资源信息，并对分割后的每一个单一型。资源提供详细信息填写的功能。

(5)采集规则自动生成功能。采集者通过系统提供的内置浏览器选择了需要采集的期刊资源并填写了相关详细信息之后，系统需要采集期刊网站其他页面的相同类型信息，因此系统得记录下采集者所选择资源的采集规则，并根据此规则采集其它页面中的对应信息。采集规则自动生成的解决方案是根据当前选择的节点，往上寻找父类节点，直到网页根节点，将每个节点的标签和属性进行拼接，形成Jsoup能识别的采集规则，如“TR>TD[class][align=left][bgcolor=#CCCCCC][valign=top][height=50][width=90]>A[class]”。

(6)页面结构检查功能。开放获取期刊资源采集困难的原因之一就是期刊网页结构多变，不同的期刊有不同的展示结构，同一期刊不同时期也有不同的组织形式。不同期刊资源的采集需要采集者在系统中进行元数据资源选择，而对于同一期刊，采集者进行选择之后还需要判断该期刊是否有结构的变化，如果发生变化，则当前的采集规则无法对结构变化后的页面进行采集，就需要采集者对结构变化后的页面进行重新选择，进而补充当前的采集规则。因此页面结构检查功能是资源采集完整的保证。页面结构检查的解决方案是根据必须字段来判断当前页面结构是否发生了变化。如文章的标题选择为必须字段，如果对某一页面进行采集时，根据标题的采集规则，没有采集到对应的标题资源，则说明当前采集的页面不符合要求或者页面结构发生了变化。当判断当前页面为结构变化的页面后，便将该页面的链接存放到结构变化网页链接数组中，继续采集剩下的页面，等这一轮采集结束之后，再对链表中存在对链接进行重新选择和采集。如此循环，直到链表中没有结构变化后的链接。

(5)期刊资源展示和下载功能。在完成了期刊资源采集之后，系统还需提供采集资源展示功能，以及资源数据下载功能。数据以表格的形式进行展示，提供下载为文本文件。

(6)日志功能。从开发者的角度，为了便于维护和调试，系统应该有日志功能，记录系统在运行过程中暴露的问题，其中包括错误出现的时间，错误出现的地方以及错误的内容等。

3.3.2系统性能需求

开放获取期刊资源采集系统在满足了基本功能需求的基础之上，在系统性能方面也得到达一定的要求，主要表现在以下两个方面：

(1)效率，即尽可能快的采集期刊资源。开放获取期刊资源采集系统是一个需要与用户反复进行交互的系统，因此系统需要尽可能快的完成采集任务，将采集结果反馈给用户。通过对开放获取资源采集的调研，在使用单线程采集时，一千篇文章资源采集需要30分钟左右的时间。系统访问页面的时间要比在页面上通过采集规则采集数据的时间慢了近十倍，因此系统效率性能的瓶颈在于页面的访问上。要提高系统采集效率，可以采用多线程并发访问页面，减少在页面访问上的等待时间，提高资源的利用率。

(2)完整性，即期刊资源采集需要尽可能的全面。一方面在设计多线程时，如果线程数量过多，系统资源使用过度，会导致一部分链接没访问就返回，进而丢失了一部分数据。另一方面，对于结构多变的期刊资源，系统需要将每种结构的页面都进行采集，页面结构检查功能就是用于解决此问题。

3.3.3系统其他需求

(1)可拓展性。开放获取期刊资源采集系统的拓展性主要表现为能采集不同的开放获取期刊资源，甚至不同的开放获取资源。

(2)健壮性。不同的采集需求对应着不同的采集量。面对多则几万条数据的采集，系统在保证正常运行的同时也需要保证采集数据的完整性。

(3)简洁性。系统界面设计需美观、简洁。系统操作需有简洁明了的介绍，操作尽量简单直接。

3.4开放获取期刊资源采集系统总体设计

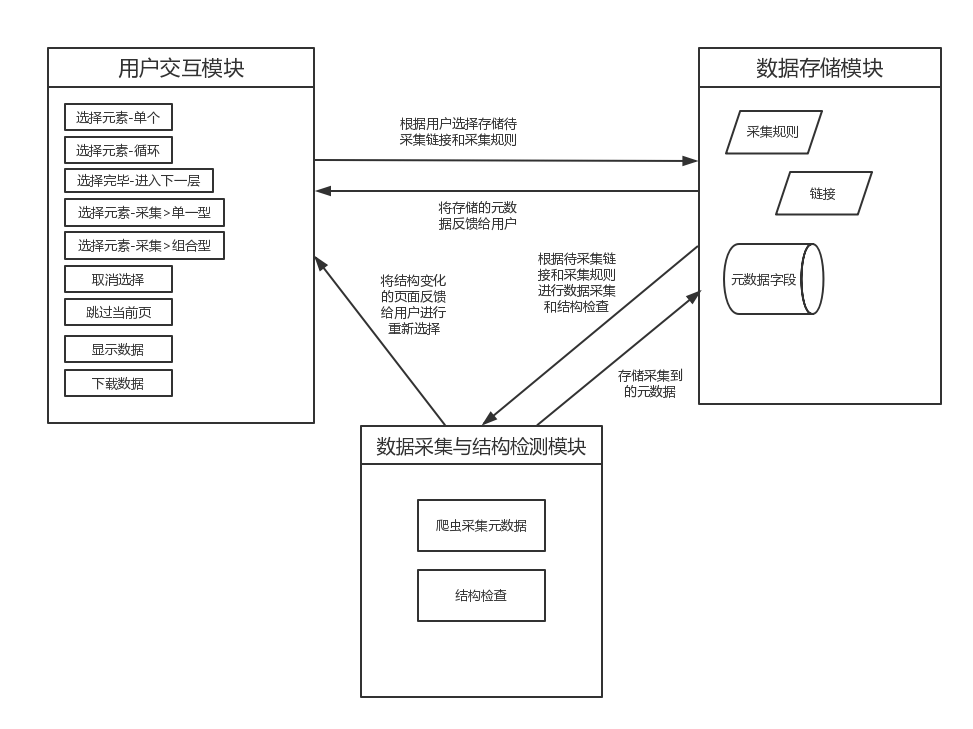
3.4.1系统设计目标

开放获取期刊是遵循开放获取原则的期刊，其在网络上同步发行，并允许免费下载。随着开放获取期刊数目的增加，大量免费获取的开放获取期刊资源对科技情报研究具有很高的利用价值。因此一些科技情报工作者着手采集这些开放获取期刊资源。开放获取期刊资源采集系统便是面向这些采集开放获取期刊资源的科技情报工作者。传统的采集方法是人工在期刊网页对元数据进行复制粘贴，这种方法工作量大、效率低。之后有科技情报工作者使用网络爬虫来采集资源，但需要人工的分析网页源代码，制定资源的采集规则。此种方法提高了工作效率，但对采集者的计算机基础有一定的要求，没有学习过相关技术的科技情报工作者难以上手。而且开放获取期刊网页结构多变，一个样本网页制定的采集规则往往无法采集到所有的资源。

开放获取期刊资源采集系统设计的目的就是结合网络资源采集方法和信息采集工具，搭建可视化的开放获取期刊资源采集工具。系统通过可视化的交互界面，让采集者在界面上进行简单的点击、选择、输入等操作，系统自动形成采集规则，在进行自动采集的同时检查页面结构是否发生变化，最后将结构变化后的页面反馈给用户进行重新选择，进而保证数据采集的完整性。开放获取期刊资源采集系统意在降低系统使用门槛，让没有计算机知识背景的工作者能便捷、有效的采集到完整的开放获取期刊资源。

3.4.1系统结构设计

系统根据数据流向分为用户交互模块、数据采集与检查模块以及数据存储模块，如图所示[]。各模块之间采用低耦合、高内聚的设计原则，每个模块各司其责，又通过参数相互联系，提高系统灵活性和拓展性。



用户交互模块，即用户将通过内嵌浏览器与系统进行交互。在此模块中用户在内嵌浏览器的网页中点击所需提取字段，无需人工解析网页源代码，后台自动形成抽取规则。该模块主要的用户行为包括：“选择元素——单个”即只选中当前页面中被选择的元素；“选择元素——循环”即选中当前页面中与被选择元素相同类型的所有元素；“选择完毕——进入下一层”即当前页面的选择完毕，系统将进入下一层；“选择元素——采集”即在详情页中选择元数据进行采集，其中包括单一型元数据采集和组合型元数据采集；“选择元素——采集>单一型元数据”即采集单一型元数据，弹框填写元数据字段详情，包括字段名称、字段类型、是否必须等；“选择元素——采集>组合型元数据”即擦剂组合型元数据，弹框显示组合型元数据分解界面，用户在填充有被选择的组合型元数据的文本框中插入分割符，系统根据分割符分割并显示各个单一型元数据，同时让用户填写各个单一型元数据的字段详情；“选择完毕——采集”即所有需要采集的元数据资源都选择完毕，开始采集；“取消选择”即取消对该节点的选择；“跳过当前页”即根据必须字段进行结构检查后，会将结构变化后的页面反馈给用户，而有的网站存在脏页，即该页面上没有任何资源，点击“跳过当前页”将该页面从待采集页面中删除。除以上功能外，用户交互模块还包括数据展示和下载功能。用户交互模块的目的是不仅降低了对用户的要求和工作量，而且通过用户自主的选择所需提取的字段，使得采集信息的准确性和可拓展性都得以保证，而且使用于任何形式的网页结构，通用性得到增强。

数据采集与结构检测模块。数据采集主要包括：链接采集和元数据字段采集。链接采集是指根据用户的选择采集种子页面、中间页面和详情页面的链接。由于网页访问是系统采集过程中花费时间最多的步骤，因此先将需链接采集有利于使用多线程来异步访问网页，进而提高效率。元数据字段采集是指根据用户在详情页面选择的元数据字段，采集所有详情页面相同的字段，采集到的元数据也是最终反馈给用户的数据。结构检测与数据采集是同时进行。系统先根据当前的采集规则对元数据进行采集。对于必须字段为空的页面就判定为结构发生变化的网页。结构发生变化的网页链接将加入到结构变化网页的队列中。待系统该轮采集结束之后，系统将取结构变化网页队列队首的链接进行跳转，再次进入用户交互模块，让用户重新选择，形成新的采集规则，再进行采集。此模块解决了开放获取资源网页结构变化的问题，提高了采集的覆盖度。

数据存储模块分为临时存储和物理存储。临时存储是指在程序中使用临时变量存储的中间数据，此类数据并不进入数据库，只是为元数据采集提供支持，如中间页面或详情页面的链接、元数据采集规则、结构变化页面链接等。这些数据不进入数据库进行物理存储的原因在于其变化性很大。此外根据“所见即所得”的软件设计思想，系统返回给用户的数据就是用户在界面上选择的数据，这些中间数据是为了系统顺利完成采集而生成的，对用户而言并没有什么意义，所以此类数据只做中间存储，并不进入到数据库中。物理存储是指根据用户选择最终采集到的元数据字段。由于用户采集的字段是无法事先预知的，因此传统关系型数据库预先设定数据模式的做法无法满足当前的情况。也有不少学者将采集到元数据字段封装成XML文档，此种存储方式可以解决资源存储的问题，但对于元数据选择和存储多了一道封装和解析的过程。如果单独取采集的某个字段查看，就得解析文档并匹配到该字段，显得复杂很多。本文选择使用非关系型数据库存储元数据字段。近几年随着非关系型数据库发展和完善，有多种非关系型数据库可以免费使用，其中最具代表性的就是文档型数据库MongoDB。由键值组成的文档是MongoDB数据库的基本单位，关系型数据库中的记录。MongoDB无需事先定义数据模式，安装和使用都十分简单，很好的满足了开放获取期刊资源采集系统的需求，从数据存储方面提高了系统的灵活性。

3.4.2系统流程设计

开放获取期刊资源采集困难的主要原因是期刊资源描述载体结构多变，同一期刊存在多种结构。一般的采集方法和采集系统形成的采集规则无法对结构变化后的期刊资源进行采集，因此在采集过程中需要增加结构检查的流程。为得到全面的采集规则，人工识别网页结构的变化不切实际。但可以将网页结构变化检测交给计算机，即如果已形成的采集规则不适合当前页面，说明页面结构发生了变化，然后返回给用户，让用户重新选择，如此循环来获得全面的采集规则，进而保证采集数据的全面性。

此外，不同的采集者对开放获取资源元数据采集需求可能不同，因此，采集到字段应该由用户自己定义。用户可以通过交互式的界面，根据采集需求对网站中的元数据进行选择，并由计算机记录下当前字段的采集路径，进而形成当前资源的采集规则。

系统具体流程如下图所示：

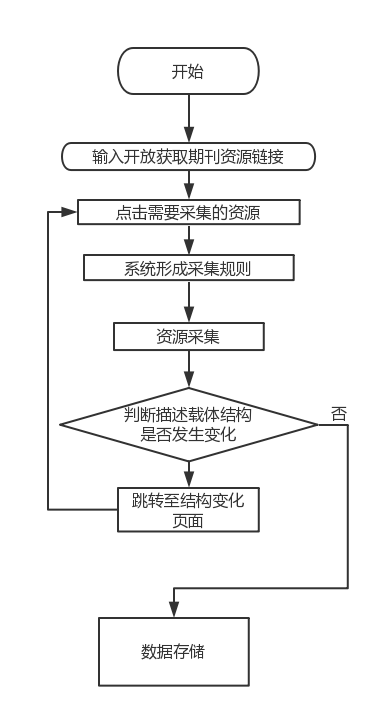


图 1开放获取期刊资源采集系统流程

具体的工作流程步骤如下：

(1)用户将其需要采集期刊的种子页面输入到系统内嵌浏览器地址栏中，并点击跳转，系统跳转至该页面。

(2)用户在种子页面中选择需要采集的链接，可以选择单个采集和批量采集。

(3)系统根据用户的选择，抓取待采集到链接，并存储到待采集链接数组中，同时生成采集规则，存储到链接采集规则数组中。数组的下标便是当前页面所在的层级，种子页面为0层，依次递增。数组的值即为采集规则。

(4)用户选择完毕之后，点击进入下一层，系统将会取待采集数组第一个的链接，进行跳转。重复(2)(3)(4)，直至到达详情页面。

(5)用户在详情页选择需要采集的元数据字段并填写字段详情，系统将记录元数据字段的采集规则，存储到元数据字段采集规则队列中。用户选择完毕之后点击“选择完毕-采集”。

(6)系统将先判断此轮采集是初次采集还是对结构变化页面的采集，如果是初次采集，则待采集链接数组就是步骤(3)形成的数组，如果是对结构变化页面进行采集，则待采集链接数组就是结构变化网页链接数组。根据种子页面的待采集链接数组，以及链接采集规则数组和元数据采集数组，对元数据进行多线程采集。同时根据必须字段是否为空来判断采集页面是否为结构变化或者无效页面，并将结构变化货无效页面的链接存放到结构变化网页链接数组中。待此轮采集完毕，便取结构变化网页链接数组第一个的链接反馈给用户，重新进入到步骤(5)，直至结构变化网页链接数组中没有元素。

第四章开放获取期刊资源采集系统实现

4.1开发平台与工具

表格 2 硬件平台

|  |  |
| --- | --- |
| 类别 | 说明 |
| CPU | 2.7 GHz Intel Core i5 |
| 内存 | 8 GB 1867 MHz DDR3 |
| 硬盘 | Macintosh HD 249.8 GB |
| 网卡 | AirPort |

表格 3 软件工具

|  |  |
| --- | --- |
| 类别 | 说明 |
| 操作系统 | MacOS Sierra 10.12.3 |
| 开发语言 | Java |
| 开发环境 | JDK1.8.0\_111 |
| IDE | IntelliJ IDEA 2016.2.5 |
| 数据库 | MongoDB 3.4.1 |
| 第三方插件 | Jsoup-1.10.1、log4j-1.2.17 |

4.2系统总体模块类图

开放获取期刊资源采集系统主要分为三大模块：用户交互模块、数据采集与结构检查模块和数据存储模块。每个模块对应这一个大类，系统类图如下所示。

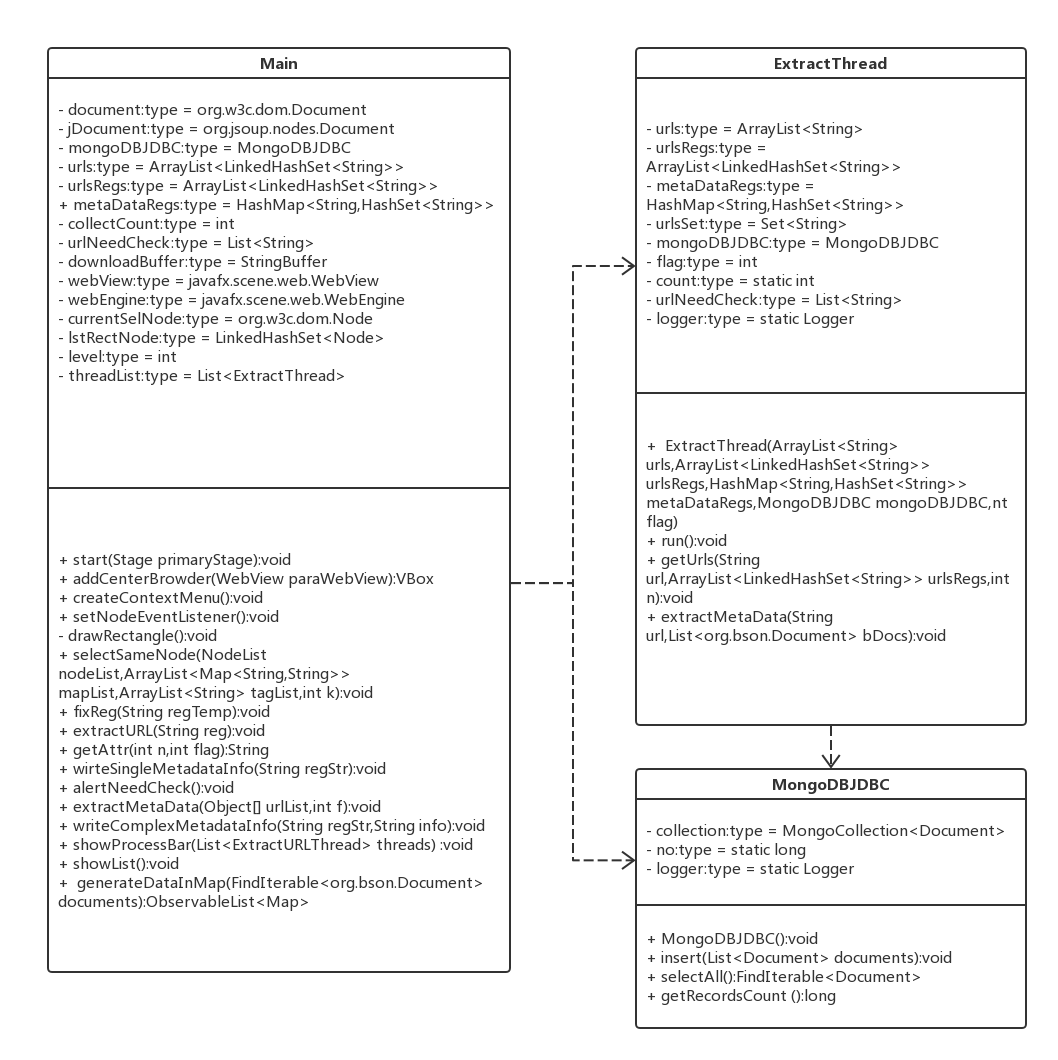


图 2 系统类图

Main类属于用户交互模块，继承于JavaFX 的Application类，用于实现页面展示和个各种用户操作，以及对采集线程的调用。ExtractThread类属于数据采集和网页结构检查模块，是一个线程实现类，通过Main类进行多线程调用，主要功能是根据采集规则采集数据和根据必须字段检测页面结构。根据采集功能需求，构造ExtractThread对象需要传入五个变量，即待采集链接、中间链接采集规则、元数据采集规则、数据库连接对象以及标识当前采集是初次采集还是结构变化页面采集的标志。MongoDBJDBC属于数据存储模块，主要实现连接数据库、查询数据和插入数据等功能。

4.3用户交互模块

用户交互模块是开放获取期刊资源采集系统最直观的展现，是用户采集需求的主要入口，也是系统采集任务形成和启动的出发点。用户交互模块的主要功能包括但不限于页面节点选择、同类型元素选择、采集规则形成、元数据采集多线程启动等。用户交互模块主要是由Main类实现。

4.3.1主要数据结构介绍

为实现各种功能，Main类中有一些复杂的数据结构，下表是对其中主要的变量进行简单介绍。

表格 4 用户交互模块主要数据结构介绍

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量解释 | 变量类型 | 作用 |
| level | 页面层级 | int | 表示当前页面的层级，种子页面为0层，以此递增，直至详情页。 |
| urls | 待采集链接数组 | ArrayList <LinkedHashSet<String>> | 存储用户在每层页面中选择的待采集链接，数组下标代表页面层级。 |
| urlsRegs | 待采集链接采集规则数组 | ArrayList <LinkedHashSet<String>> | 与urls相对应，存储每层待采集链接的采集规则，数组下标代表页面层级。 |
| metaDataRegs | 元数据采集规则映射表 | HashMap<String,HashSet<String>> | 存储元数据字段和元数据采集规则，映射表键值为元数据字段，值为元数据采集规则集合。 |
| urlNeedCheck | 结构变化网页数组 | List<String> | 存放结构变化网页的链接。 |
| threadList | 采集线程数组 | List<ExtractThread> | 存放多个元数据采集线程 |
| currentSelNode | 当前用户选中的页面元素节点 | org.w3c.dom.Node | 存放用户在内嵌浏览器中选中的节点 |
| lstRectNode | 当前被选中节点集合 | LinkedHashSet<Node> | 存放当前页面中被选中的所有节点 |

4.3.2网页显示和页面元素选择及高亮显示

WebView组件是JavaFX提供的内嵌浏览器组件，通过API提供web查看器以及完整的浏览功能，是基于开源web浏览器引擎WebKit，支持CSS、JavaScript、DOM和HTML5。其主要功能为：加载本地或远程URL的HTML内容、获取Web历史、执行JavaScript指令等。WebEngine类是WebView组件中主要实现类，其提供了基本的web页面功能，支持加载HTML内容和访问DOM对象等基本功能，也支持执行JavaScript指令。WebEngine类一次处理一个web页面，web页面的链接通过构造WebEngine对象传入，或者通过load()方法传入。WebView类是WebView组件另一个主要实现类，其作用是在应用程序的scene中展示web页面。

[http://www.javafxchina.net/blog/2015/07/html\_webview/]

开放获取期刊资源采集系统提供类似于浏览器地址栏的文本框，供用户输入其需要采集的期刊的种子页面链接。用户点击跳转之后，系统将地址文本框中的链接传递给WebEngine对象，并使用WebEngine.load()方法获取对应的HTML内容，进而通过WebView类在应用程序的scene中展示。

页面元素选择也是利用了WebView组件对DOM对象访问的功能。具体做法是给WebEngine对象加上事件监听，每当WebEngine对象所加载的HTML发生变化，即跳转至新的页面时，通过执行JavaScript命令将HTML内容转化为DOM对象，并递归给DOM对象每个节点加上鼠标点击事件监听。事件监听主要作用是当鼠标进行点击时，便将当前鼠标点击的节点元素赋值给当前选中节点变量currentSelNode，同时将该变量加入到当前选中节点集合中lstRectNode。

由于WebEngine执行JavaScript命令获取到的DOM对象是org.w3c.dom类型，该类型的DOM对象对节点的搜索功能并不丰富，为了采集数据的方便，本文在WebEngine转化HTML内容为DOM对象的同时，也使用Jsoup将HTML内容转化为易搜索的org.jsoup.nodes.Document类型的DOM对象。

对选中的元素进行高亮显示主要是利用WebView组件允许在内嵌浏览器展示层增加覆盖的特性。具体做法是获得当前选中节点之后，当前节点调用JavaScript指令getBoundingClientRect，获取节点在页面中的坐标。接着声明一个JavaFX的矩形图形，矩形大小和颜色自定义，再通过StackPane.setAlignment()和StackPane.getChildren().add()指令将矩形块添加到节点对应的位置。

4.3.3采集规则生成以及同类型节点自动选择

对于开放获取期刊资源采集系统实现自动批量采集，最根本的是获得页面上待采集元数据资源的采集规则。Jsoup是一款功能强大的HTML解析器，出了解析URL对应的HTML内容外，还提供具有丰富API的选择器来搜索并取出HTML标签中的内容。Jsoup选择器对于采集规则有很丰富的支持，通过HTML标签名以及标签属性值可以定位相应的节点。为清楚的解释Jsoup选择器采集网页数据的过程以及开放获取期刊资源采集规则形成的解决方案，本文选取某个开放获取期刊的一个典型网页为对象进行具体阐述。经过简化后的HTML文档内容如下图所示。

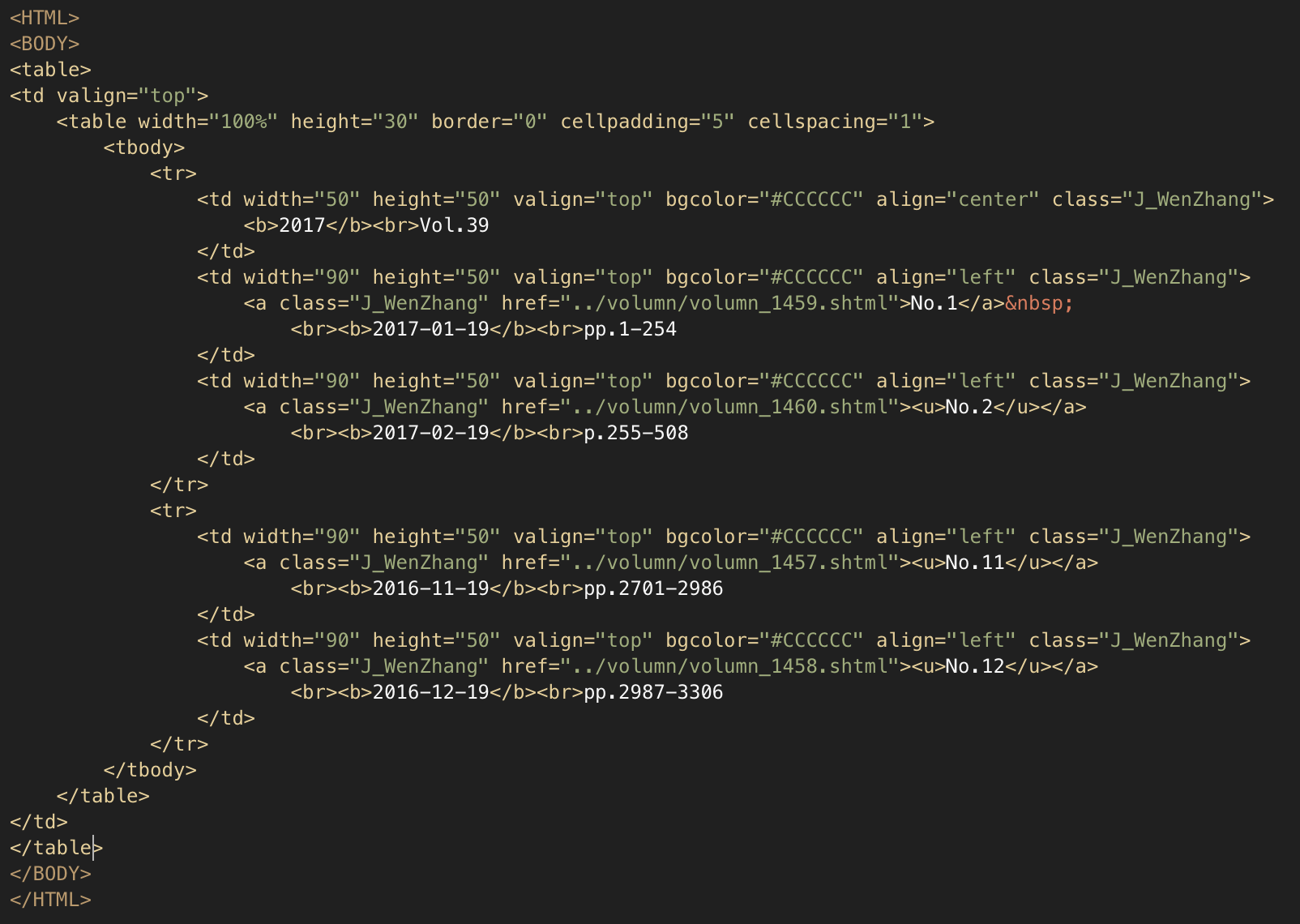


图 3HTML文档示意图

文档展示的是某个期刊的种子页面，包含有该期刊各期的链接。现需要采集期刊各期的访问链接，即文档中标签为a的“href”属性的值。Jsoup选择器是根据标签名和属性值进行定位，可以是单个节点的标签名和属性值组成的规则，也可以是由父子节点的标签名和属性值组合而成的规则。以父子节点组合而成的采集规则为例，该文档中期刊各期链接的定位规则为“body>table>td[valign=top]>table[width=100%][height=30][border=0 ][cellpadding=5][cellspacing=1]>tbody>tr>td[width=50][height=50][valign=top][bgcolor=#CCCCCC][align=left][class=J\_WenZhang]>a”,通过select方法就可以采集到对应的<a>节点，再通过absUrl方法便可采集到对应的链接。

传统的资源采集和部分商用资源采集工具都需要人工对网页内容进行分析，寻找到能定位到采集节点的规则。在制定定位规则时容易出现两种问题，第一个问题是定位规则粒度过细，人工分析工作量大。即如果从最顶层的标签开始往下搜索，到采集节点为止，期间所经过的节点都作为规则的一部分，这样虽然可以精准的定位到节点，但是需要花费大量的时间和精力，特别是对结构复杂的页面。第二个问题是定位规则粒度过粗，采集到脏数据。即如果只使用需采集节点的标签名和属性值组成定位规则，则页面中与采集节点具有相同标签名和属性值，但不是所需采集的节点会被采集到，这种情况经常出现在资源采集过程中。

开发获取期刊资源采集系统通过使用WebView组件加载HTML内容时，为每个结点都添加了事件监听，在鼠标点击某结点时，系统会将该节点赋值给org.w3c.dom.Node类型的变量。Node类提供getParentNode()的方法来获取当前节点的父类节点。因此在形成定位规则时，完全可以使用计算机来往上递归节点，直到网页的根节点，并自动记录下该节点在网页文档中的路径。这样既保证了采集的准确度，又解放了人参与的工作量。

本文根据资源组织形式将开放获取期刊资源分为单一型资源和组合型资源，同时也根据资源的类型分为文本和链接。不同类型的资源有不同的采集方式。对于单一型文本资源采集的是待采集节点的文本信息，即节点标签包含的文本信息就是待采集元数据字段信息。因此单一型文本资源的采集方式是先定位到待采集节点，再通过text()方法提取文本信息。而单一型链接资源，采集的是待采集节点标签中的href属性值。因此单一型链接资源采集方法是定位到待采集节点，再通过absUrl(“href”)方法提取出链接信息。组合型资源是由多个单一型资源组合成一条文本信息进行展示，采集时需要将组合文本分割为各个单个资源。分割步骤在用户填写组合型元数据字段详情时，由用户向文本信息中插入分割符来完成，由分割符便可以知道各单一资源在文本信息中的前后字符，进而便可得知单一资源的位置。如常见的组合型资源期刊年卷期文本信息为“2017,Vol.39 No.2”，“年”字段是从文本开始处到“,Vol.”，“期”字段信息时从“,Vol.”到“No.”，“卷”字段信息是从“No.”到文本结束。因此组合型资源的采集方法是先定位到待采集节点，通过text()方法得到文本信息，再根据用户设定的分割符的位置，找到单一节点在文本中的起始分割字符，进而切割文本得到各个单一型资源。

不同类型的资源对应着不同类型的采集方法，进而对应着不同的采集规则形成方式。单一型资源的采集规则就是其定位规则，只是单一型文本资源的取值方法是text()，单一型链接资源的取值方法是absUrl()。组合型资源的采集规则是其定位规则加上组成组合型资源的各单一型资源的起始分割符位置。而这些资源的识别和采集规则的形成，都是系统根据用户的选择和操作自动生成。

同类型节点的自动选择是在用户选择了“选择元素-循环”操作之后触发，主要是对采集当前页面同类型的链接而设计。由于选择同类型元素还需要在页面中高亮显示，即需要与JavaFX的WebView组件进行交互，WebEngine执行JavaScript获取到的DOM对象是org.w3c.dom类型，而Jsoup操作的DOM是org.jsoup.nodes.Document类型，因此Jsoup选择元素在此无法使用。在形成当前选择节点的采集规则时，系统记录了从根节点到当前选中节点所经过的所有节点的标签名和各属性值。因此本文对同类型节点自动选择的解决方案是先通过org.w3c.dom对象提供的getElementsByTagName()方法，从当前选中节点所在路径的最顶层节点开始，首先选择dom对象中与该节点标签名相同的所有节点，再逐一比较各节点与该节点属性值是否相同，如果相同，进入下一层的选择和比较，直到找到所有相同元素。并将节点加入到当前选中节点集合lstRectNode中。待遍历完链表，对当前选中节点集合中的所有节点进行高亮操作。

4.3.4多线程控制

待用户选择完毕之后，待采集链接、中间链接采集规则、元数据采集规则都已经自动形成并保存在中间变量中。从上一章的需求分析中，得出开放获取期刊资源采集系统的效率瓶颈在于页面访问需要花费大量的时间。因此，多线程异步访问链接是一个提高系统性能的可选方案。

线程是操作系统中进行调度的基本单位，被称为轻量级进程。线程允许在同一个进程中同时执行多个任务，同一进程中的线程共享进程范围内的资源。多线程是指在一个进程中同时由多个线程在多个处理器上执行。如果设计正确，多线程程序可以通过提高处理器资源的利用率来提升系统吞吐率[java并发编程实战]。Java虚拟机在启动线程时会占用一些内存，操作系统在做线程调度时也会消耗一定的资源，因此过多的线程会影响系统性能，过少的线程又使系统资源得不到充分利用。

设计多线程数量时，一般会先进行一些测试。开放获取期刊资源采集是先记录下种子页面待采集链接，再根据中间链接的采集规则，采集和访问中间链接，直至详情页面。类似于森林的数据结构，种子页面的待采集页面是森林中各个树的根节点，再根据中间链接采集规则向下发展直到详情页面，各个详情页面便是各个树的叶子结点。

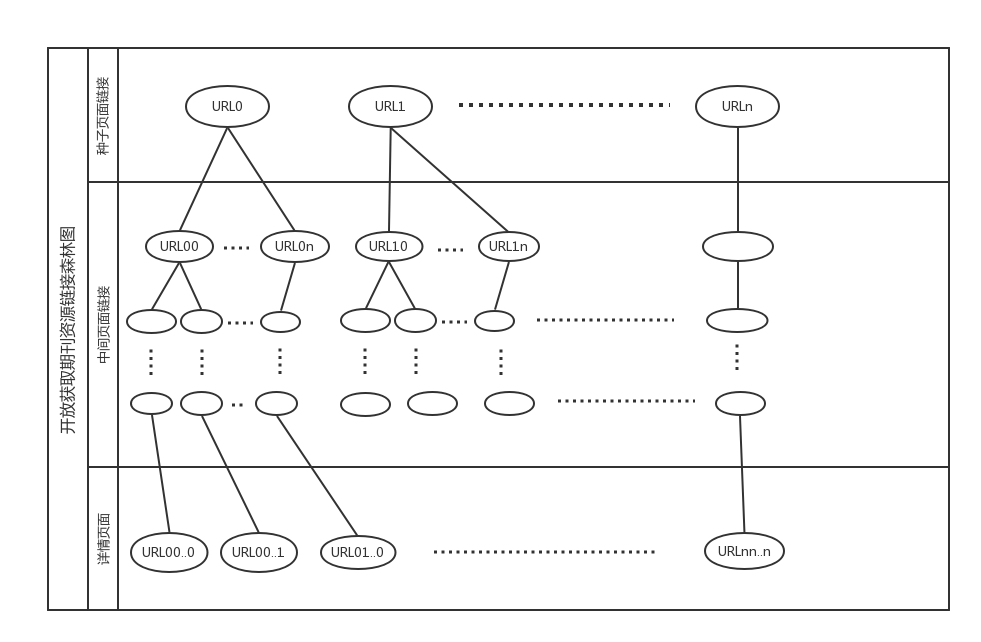


图 4 开放获取期刊资源链接森林示意图

开放获取期刊资源的线程采集任务是访问种子页面上选择的待采集链接，根据中间页面采集规则，层层递进访问到详情页面，再根据元数据采集规则，采集元数据资源，并保存到数据库中。因此开放获取期刊资源采集的线程数量是由种子页面待采集页面链接数量和一个线程负责访问多少链接而决定。在对开放获取期刊资源采集调研中发现，开放获取期刊网站的种子页面一般展示期刊各年卷期的访问链接，展示的粒度不尽一致。有的期刊以“年”进行展示，此类期刊种子页面上待采集的链接数量较少，基本上都是以十为数量级。有的期刊以“卷”进行展示，此类期刊种子页面上待采集的链接数量一般以百为数量级。虽然种子页面待采集链接数量不定，但是可以知晓其数量级在十、百之间。在经过一些简单测试之后，本文以50个链接为基本访问单位，即每个线程最多访问50个种子页面的待采集链接。线程数量控制在10个以内。

4.4数据采集和网页结构检查模块

数据采集和网页结构检查模块主要是由ExtractThread类具体实现。构造ExtractThread类需要有5个参数：待采集链接数组，即线程需要访问的种子页面的链接，为保证线程采集质量，链数量不会超过50个；中间连接采集规则，即从种子页面链接访问开始到详情页之间的采集规则；元数据采集规则，即详情页面上需要采集的元数据的采集规则；数据库连接，与数据库进行交互；采集类型标志，判断是初次采集还是对结构变化后的页面进行采集。

4.4.1主要数据结构

表格 5 数据采集和网页结构检查模块主要数据结构

|  |  |  |  |
| --- | --- | --- | --- |
| 变量名称 | 变量解释 | 变量类型 | 作用 |
| urls | 待采集链接数组 | ArrayList <LinkedHashSet<String>> | 存储用户在每层页面中选择的待采集链接，数组下标代表页面层级。 | |
| urlsRegs | 中间链接采集规则 | ArrayList <LinkedHashSet<String>> | 与urls相对应，存储每层待采集链接的采集规则，数组下标代表页面层级。 | |
| metaDataRegs | 元数据采集规则 | HashMap<String,HashSet<String>> | 存储元数据字段和元数据采集规则，映射表键值为元数据字段，值为元数据采集规则集合。 | |
| urlsSet | 待采集详情页面链接 | Set<String> | 存储详情页面的链接 |
| urlNeedCheck | 结构变化网页链接数组 | List<String> | 存放结构发生变化或的链接 |
| flag | 采集类型判断标识 | int | 标识当前采集是初次采集还是对结构变化的页面的采集 |
| mongoDBJDBC | 数据库连接 | MongoDBJDBC | 链接数据库并进行操作 |

4.4.1数据采集

数据采集分为初次采集和对结构变化的页面采集。初次采集时，传入的待采集链接时种子页面的链接，还需要根据中间链接采集规则层层递进，最终到详情页面的链接。再依次访问详情页面，根据元数据采集规则，采集最终的元数据资源。而对结构变化的页面采集时，传入的待采集链接时结构发生变化的详情页面的链接。因此不用中间链接的采集规则，直接使用元数据采集规则，对结构变化后的链接进行采集。对结构变化页面的采集会持续多轮，直到所有结构的页面都进行了采集。

在每次对详情页面进行采集时，会新建一个MongoDB所支持的文档类型(org.bson.Document)变量，对采集到的各个元数据资源会以元数据名称为键、元数据资源内容为值，添加到文本类型变量中。最终将所有的文本组合而成的数组插入到数据库中。

4.4.2结构检查

结构检查是开放获取期刊资源采集系统区别于其它采集系统的主要表现。当前网站为了方便管理和后期维护基本上都采用的是“模版+数据”的模式。一般的网站会采用一套模版，而开放获取期刊的网站一般会有两到三套模版。结构变化往往会导致数据采集不完整。因此，本文在采集系统中增加了结构检查功能，以应对采取多个模版的期刊网站，进而采集到更全面的数据。

在关系型数据库中，在设计表时，一般会制定一个字段为主键，用以唯一标识标中的纪录。主键不能为空，否则该条纪录就为非法纪录，不允许插入到表中。主键的存在是为了规范表中的数据，保证数据的有效性。同样，在开放获取期刊资源采集时，也需要有类似于主键的字段来保证资源采集的有效性。比如文章的标题，如果当前采集的资源没有文章标题，即可认定为当前采集资源是无效资源。因此在用户交互模块，在用户填写字段详情时，需要填写该字段是否为必需字段，如果为必需字段，则系统在采集时会对其进行检查，如果为空则可能有两种情况，一是当前页面确实没有该字段，此页面为脏页面，二是当前页面存在此字段，但之前此元数据的采集规则不适用于当前页面，即可以判断此页面为结构变化后的页面。系统无法识别必需字段为空时到底出现的是哪种情况，因此系统都会将当前页面加入到结构变化页面链接数组中。当此轮采集结束之后，会取结构变化网页数组的第一个链接，在内嵌浏览器中进行展示，由用户判断字段为空属于哪种情况。系统对两种情况都提出了解决方案，对于第一种脏页的情况，系统提供“跳过当前页”操作，来跳过此页面，并将此页面的链接，从结构变化网页链接数组中删除。对于第二重网页构变化的情况，用户会重新在结构变化后的页面上进行元数据选择，并重新填写相关详情。与此同时，元数据字段形成的新采集规则会加入到原采集规则集合中。待选择完毕进行采集时，系统会使用新的采集规则对其它待采集链接进行采集。这样往复两三次便可以将网站所有模版遍历到，采集到最全面的数据。

4.5数据存储模块

数据存储模块的临时存储主要体现在以上的两个模块中。物理存储使用的是面向文档的非关系型数据库MongoDB。文档是MongoDB的最小存储单元，类似于关系型的纪录。一组文档被称为集合，类似于关系型数据库中的表的概念。MongoDB无需预定义数据模式，集合是动态模式，这就意味着一个集合中可以存在各式各样的文档。但是如果将格式不一样的文档存放在同一个集合中，这对于开发者是一个噩梦。开发者要么确保每次查询只返回特定类型的文档，要么让执行查询的应用程序处理所有不同的文档[mongdb权威指南]。此外对数据库查询性能也会带来负面影响。因此同一个集合里只存放一种数据模式的文档，不同类型的文档存放于不同的集合中。在开放获取期刊资源采集系统中，每次使用系统所采集的数据可能均不相同，构成的文档结构每次都不尽一致。因此每次启动采集系统都需要新建一个集合来存储文档。MongoDB使数据存储免去了模式设计的步骤，解放了在数据存储方面的工作量，将精力更多的投放于开发获取期刊资源采集方案的探究。此外系统将在一些主要操作的位置设置打印日志，以便测试和维护系统功能。

第五章系统测试及效果展示

为验证开放获取期刊资源采集系统的采集效果，以及更加直观的展示系统的采集过程，本章将从系统功能和系统性能两个方面对系统进行全面测试，同时将展示系统主要的操作界面，最后对系统进行综合评价。

5.1测试准备

5.1.1软硬件环境准备

表格 6 系统测试软硬件环境

|  |  |
| --- | --- |
| 名称 | 版本 |
| 操作系统 | MacOS Sierra 10.12.3 |
| JDK | JDK1.8.0\_111 |
| Java IDE | IntelliJ IDEA 2016.2.5 |
| MongoDB | MongoDB 3.4.1 |
| 第三方插件 | Jsoup-1.10.1、log4j-1.2.17 |
| CPU | 2.7 GHz Intel Core i5 |
| 内存 | 8 GB 1867 MHz DDR3 |
| 网卡 | AirPort |

5.1.2测试用例准备

本文从专家遴选的中文开放获取期刊资源中随机挑选了10个期刊作为测试用例，其中网页结构发生变化的期刊有5个，网页结构没有发生变化的有5个，如下表所示。采集的元数据字段包括：标题、作者、作者结构、英文标题、摘要、关键字、全文获取链接、年、期、卷等字段，其中标题为必需字段，其它均为可选字段。

表格 7 测试用例

|  |  |  |
| --- | --- | --- |
| 期刊名称 | 期刊链接 | 网页结构是否变化 |
| 电子与信息学报 | http://jeit.ie.ac.cn/CN/article/showOldVolumn.do | 是 | |
| 北京航空航天大学学报（社会科学版） | http://118.145.16.213/bhxb\_skb/CN/article/showTenYearOldVolumn.do | 是 | |
| 地理科学 | http://geoscien.neigae.ac.cn/CN/article/showOldVolumn.do | 是 | |
| 材料研究学报 | http://www.cjmr.org/CN/article/showOldVolumnList.do | 否 | |
| 电子设计工程 | http://mag.ieechina.com/oa/dlistnum.aspx | 否 | |
| 工程设计学报 | http://www.zjujournals.com/gcsjxb/CN/article/showTenYearOldVolumn.do | 否 | |
| 吉林大学学报（工学版） | http://xuebao.jlu.edu.cn/gxb/CN/article/showTenYearOldVolumn.do | 是 | |
| 光学精密工程 | http://www.eope.net/CN/article/showTenYearOldVolumn.do | 是 | |
| 心理学报 | http://118.145.16.229:81/Jweb\_xlxb/CN/article/showOldVolumn.do | 否 | |
| 冶金分析 | http://journal.yejinfenxi.cn/CN/article/showTenYearOldVolumn.do | 否 | |

5.2系统测试

5.2.1系统功能测试

系统功能测试是验证系统各个模块的功能是否达到系统设计阶段预定的需求。具体的测试方案如下表所示：

表格 8功能测试方案

|  |  |  |
| --- | --- | --- |
| 测试模块 | 测试内容 | 测试方案 |
| 用户交互模块 | 网页展示和页面元素选择 | 在系统中防蚊测试用例中的开放获取期刊链接，并选择需要采集的元素，看操作之后是否高亮。 |
| 采集规则自动生成 | 用户选择元素之后，系统将自动形成采集规则，将形成的采集规则通过日志的形式输出到日志文件中。通过查看日志文件来判断采集规则是否生成正确。 |
| 组合型元数据切分 | 组合型元数据切分测试主要分为两个方面：1.从系统页面上查看切分后的字段是否正确；2.采集完毕之后查看切分字段是否正确。 |
| 数据采集和网页结构检查模块 | 数据采集 | 先用爬虫计算出当前采集期刊论文的总数，待系统采集完毕之后，通过下载数据，查看和对比数据采集是否正确和完整 |
| 网页结构检查 | 对结构有变化的网页进行采集，观察系统是否能识别出结构变化的网页，并检查是否对结构变化网页中的数据进行采集。 |
| 数据存储模块 | 数据库设计与数据存储是否满足需求 | 通过对多个测试用例进行采集测试，检查数据是否能高效、正确存储。 |

本文对选择的测试用例，按测试方案对系统进行了测试，测试结果如下：

(1)页面显示正常，能正确进行跳转、退回等操作。右键菜单显示正常，能对网页元素进行单个和批量选择，选择之后，页面元素能高亮显示。

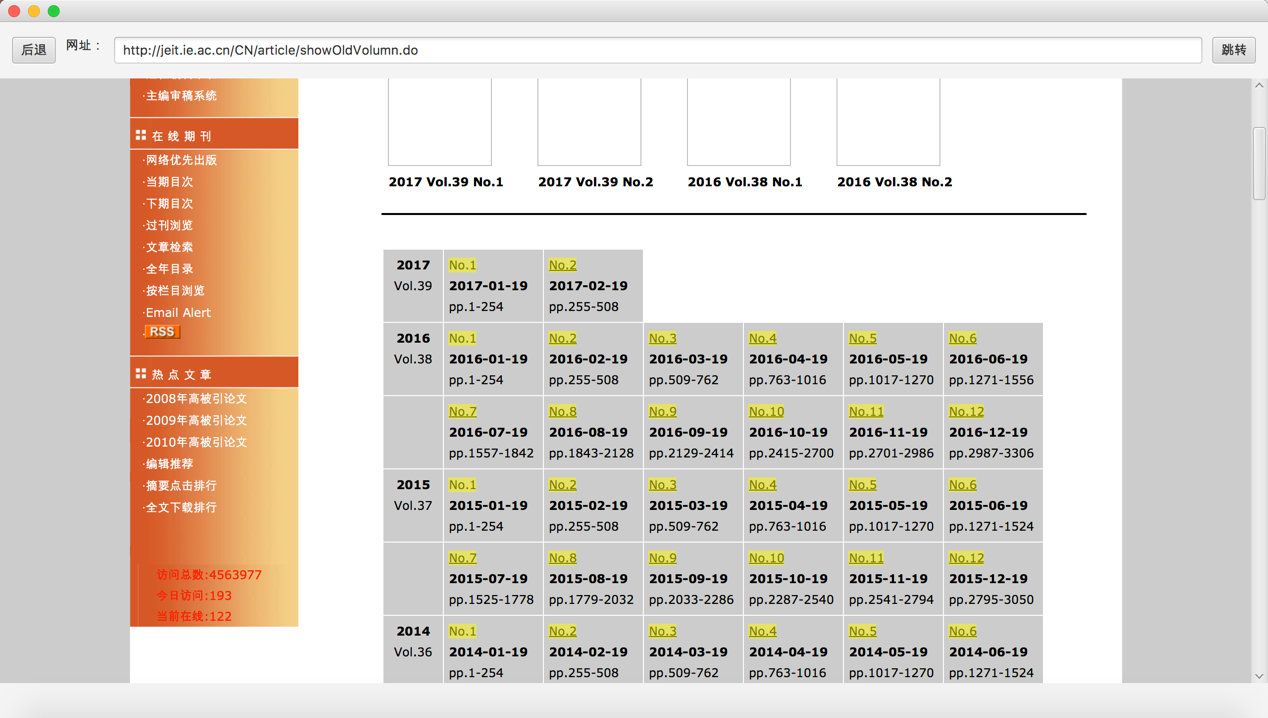


图 5批量选择并高亮显示网页元素

(2)在选择元素之后，查看日志文件，采集规则能自动形成，与网页源代码进行对比，规则均正确。

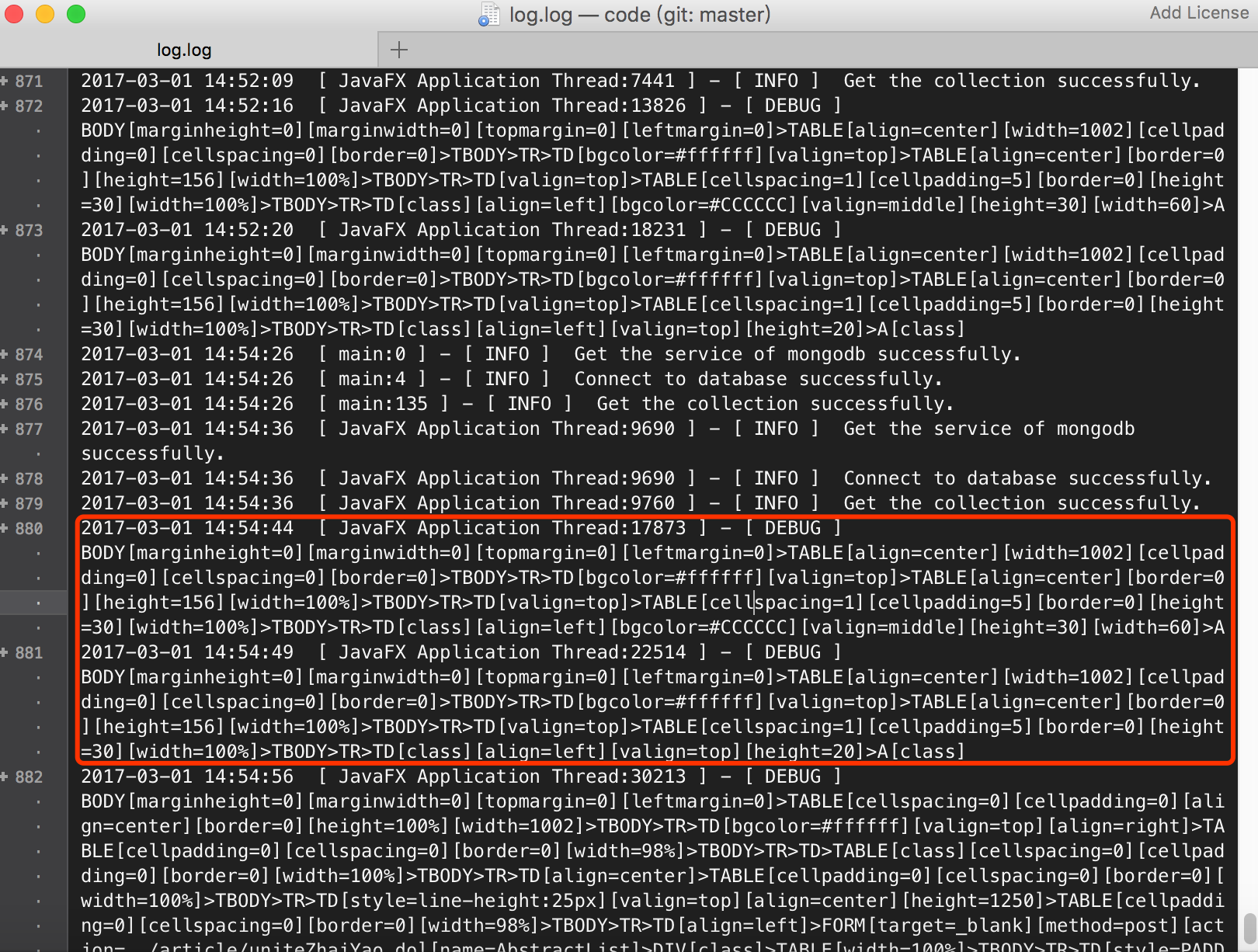


图 6 采集规则自动形成

(3)组合型元数据切分功能可以对当条文本信息进行准确切分。切割后自动形成的单一型元数据采集规则，对较规则文本可以很好批量采集，但对于不规则的文本，采集不够精确。

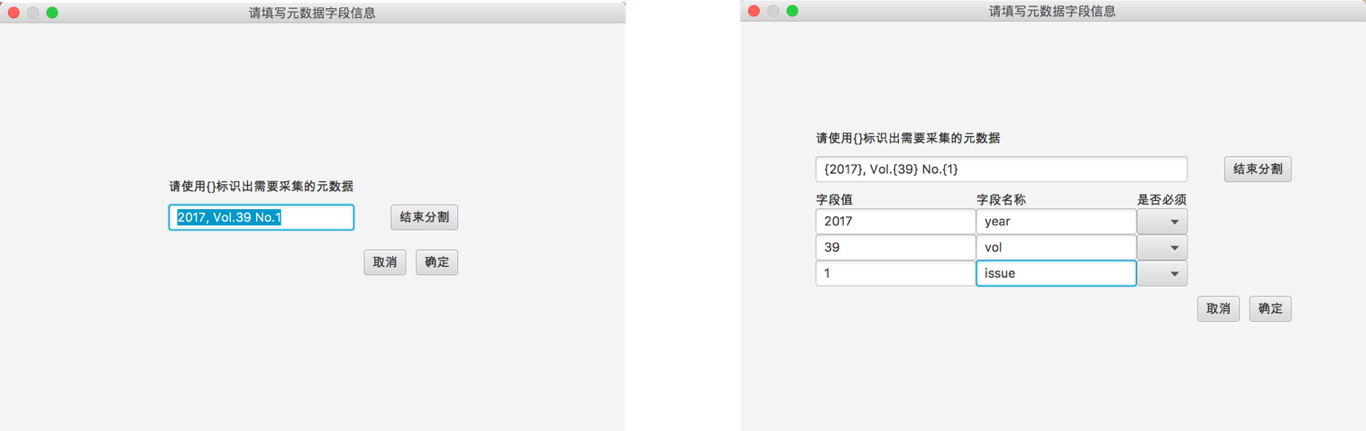


图 7组合型元数据切割

(4)对测试用例进行采集，能准确的识别出开放获取期刊网站结构的变化的网页和脏页，并将结构变化后的网页反馈给用户，重新采集。系统采集的结果跟事先统计的结果相符。但对于大量的脏页无法一次处理，只能每个页面进行人工识别后舍弃。

(5)数据存储和数据下载均能良好的允许和使用。

总体而言，系统能基本满足资源采集的功能需求，同时在操作上也简洁明了。系统能有效识别开放获取期刊网站的结构变化，并反馈给用户。系统自动形成的采集规则也能准确采集到期刊资源。系统数据存储和下载的功能可以满足用户的需求。但对于组合型元数据切割方面存在一定的局限，目前对结构固定的文本表现良好，对复杂多变的文本效果不理想。

5.2.2系统性能测试

开放获取期刊资源采集系统性能测试主要从两个方面展开，一是采集完整性，二是采集花费的时间。在使用系统对测试用例进行采集之前，本文通过另外的爬虫程序，统计了各期刊的论文数量。因此系统采集的完整性可以从采集结果和统计结果对照知晓。系统采集花费的时间可通过查看系统日志可知。系统性能测试的结果如下：

5.3系统运行展示

开放获取期刊采集系统提供网页浏览、开放获取期刊资源元素选择、开放获取期刊网页机构检查、开放获取期刊资源元数据采集和下载等功能。系统运行界面如下图所示。



图 8开放获取期刊采集系统运行页面

如图8所示，系统提供地址栏供用户输入希望采集的开放获取期刊网站，并在内嵌浏览器中展示。

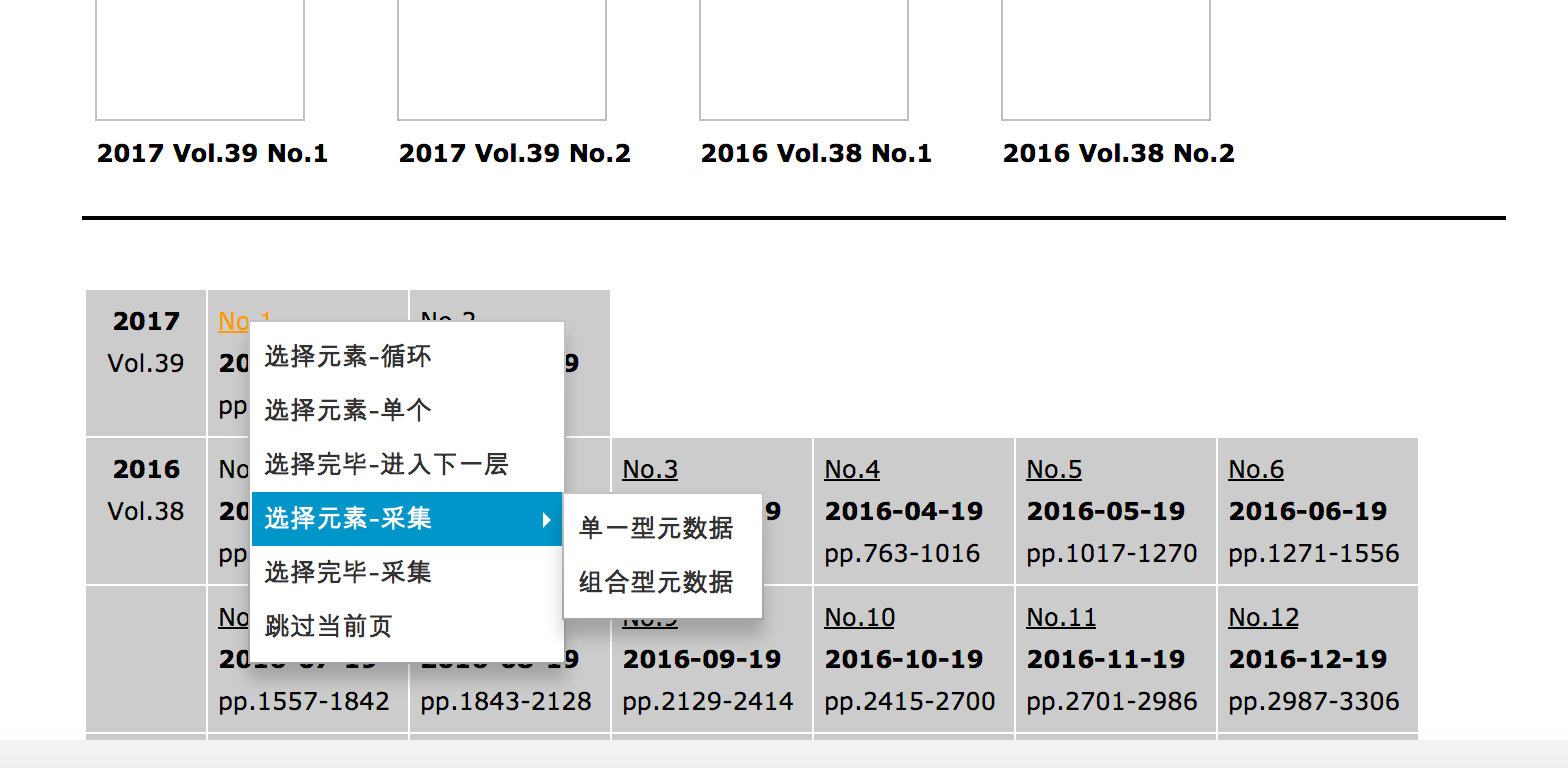


图 9 系统邮件菜单

如图9所示，根据用户行为分析，系统右键菜单提供了用户会使用到基本操作。“选择元素-循环”批量选择同类元素，“选择元素-单个”只选择当前选中的元素，“选择完毕-进入下一层”当前页面选择完毕进入到下一层的选择，“选择元素-采集”选择需要采集的元数据资源，“选择完毕-采集”调用采集线程对待采集链接中的资源进行采集，“跳过当前页”跳过当前不符合采集需求的页面并进入到下一个采集页中。



图 10 字段详情填写

如图10所示，当选择了待采集到字段之后，系统会自动高亮，并弹出对话框填写字段的详细信息，主要填写字段名称、字段类型和是否必须。

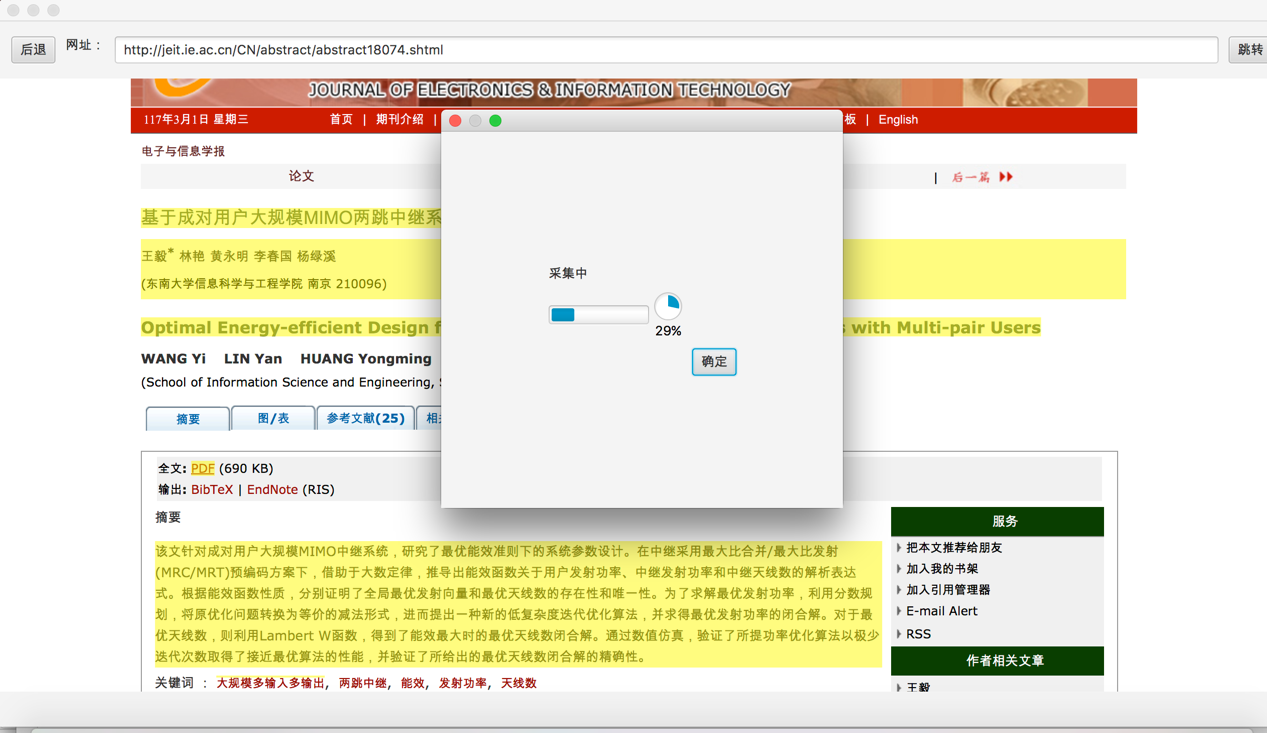


图 11系统数据采集中

如图11所示，选择完元数据字段之后，点击“选择完毕-采集”，系统会根据待采集链接数量分配采集线程，并将线程的完成进度通过进度条展示，以此来给用户反馈当前采集的采集的进度。



图 12异构页面提醒

如图12所示，当系统一轮采集完毕之后，如果发现有结构变化的网页，会弹出提示框告知用户，用户点击确定之后，系统会自动跳转到结构变化后的页面，供用户重新选择。

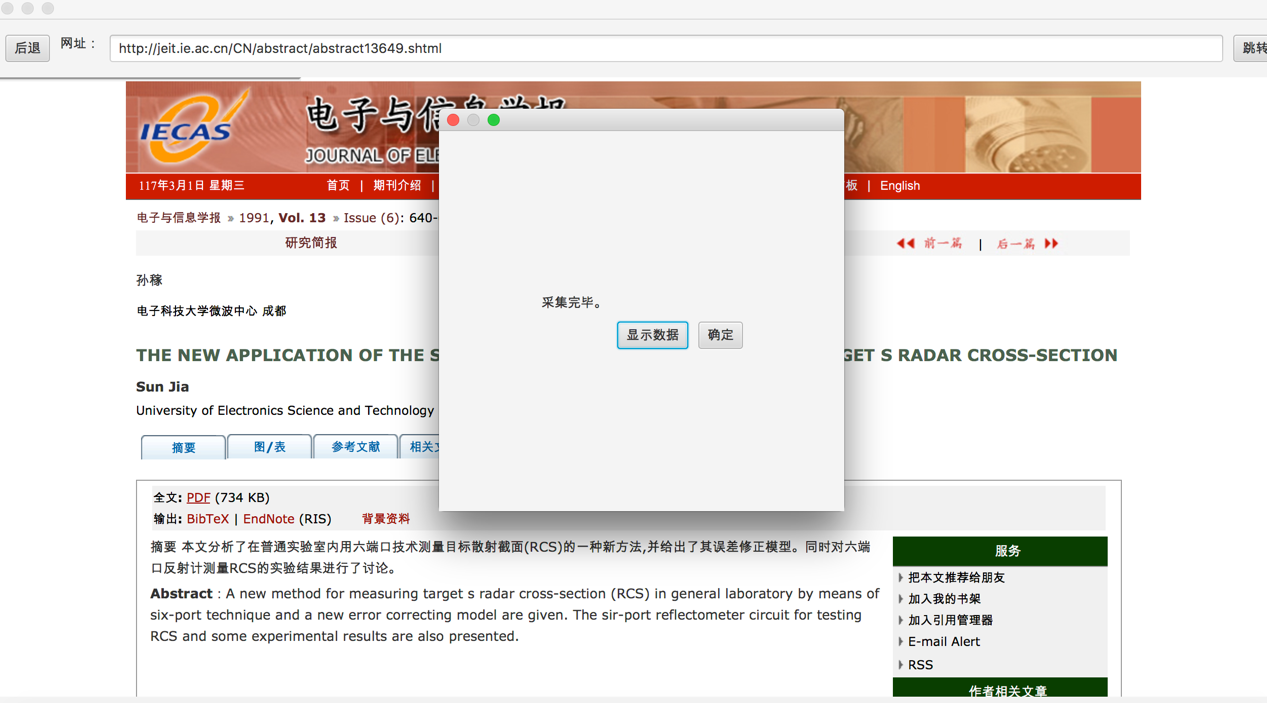


图 13 采集完毕

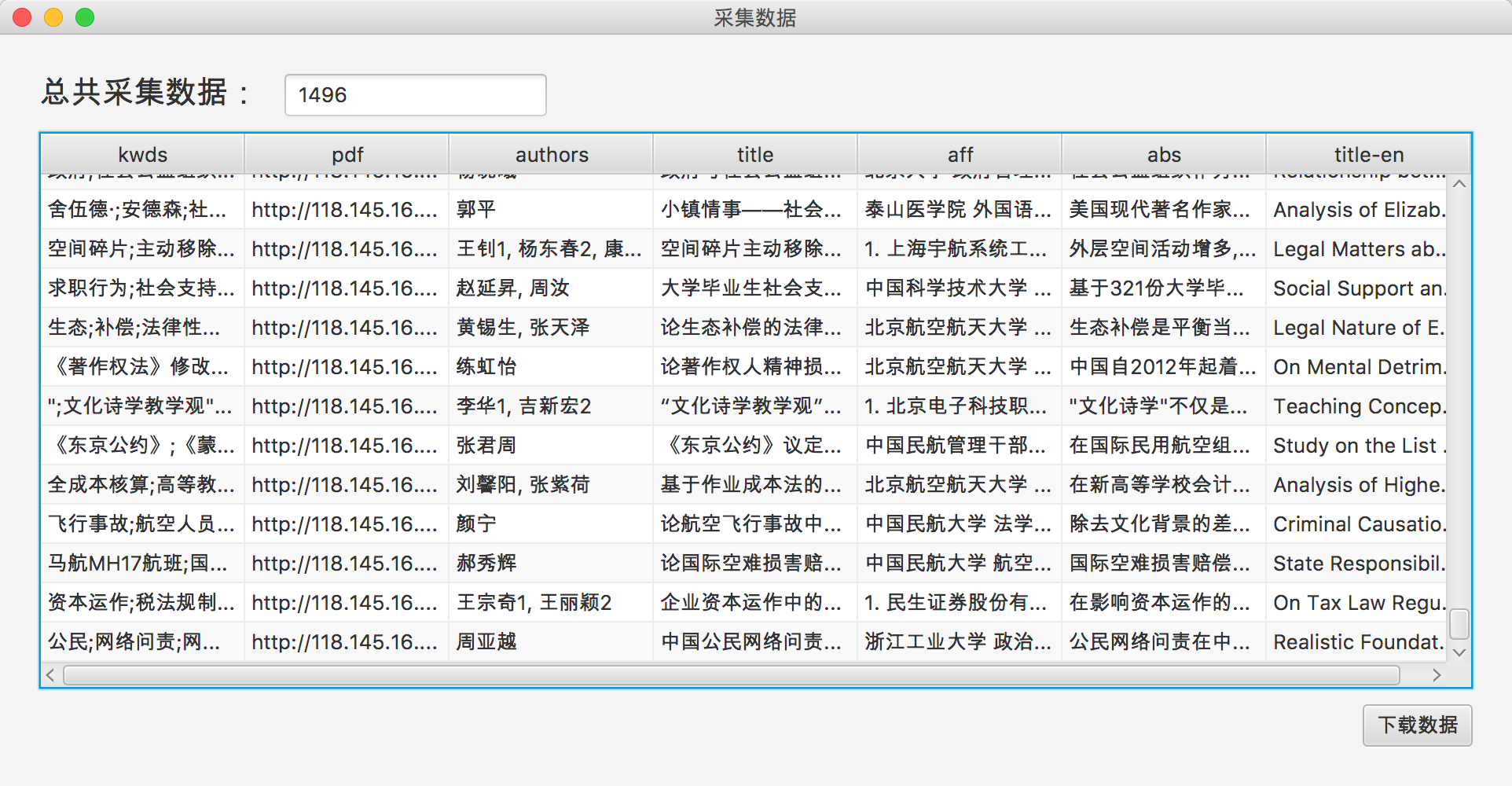


图 14采集数据显示

如图13所示，当系统采集完所有资源后，会弹框显示采集完毕。用户点击显示数据，便将采集到的资源以表格形式展示，如图14所示。

5.4系统评价

开放获取期刊资源采集系统在功能上主要分为三个模块：用户交互模块、数据采集和网页结构检查模块以及数据存储模块。以下将分别对三个模块进行具体评价。

用户交互模块是用户的操作界面，提供了浏览器的基本功能，允许用户在内嵌浏览器上选择元素。系统自定义的右键菜单提供了6种基本操作，基本满足了用户在采集资源过程中的操作需求。系统实现的选择元素后高亮显示的功能，给用户直观的视觉反馈。但系统界面过于简洁，各操作之间的跳转略显生硬。

数据采集和网页结构检查模块是系统的核心，提供了采集规则自动生成、网页结构检查等功能。就测试用例采集效果而言，系统网页结检查功能能准确的识别出期刊网站中网页结构变化的页面，为系统采集完整的数据提供支持。系统自动生成采集规则的功能，能准确的采集到网页中的数据，并能正确的采集文本类资源和链接类资源。但对于组合型资源的分割还不够精准，只能对网页中结构固定的文本资源进行采集，一方面是因为文本切割形成的规则适用不了所有的网页，另一方面是因为期刊网页组合型资源形成的文本信息结构多变，无法形成统一的规则。

数据存储模块是系统的基础，提供了数据的存储和全量查询功能。非关系数据库MongoDB的应用打破了预先定义数据模式的局限性，能很好的应对每次不同采集需求。

第六章总结与展望

6.1论文工作总结

开放获取运动的发展使越来越多的期刊加入开放获取的行列，越来越多高质量的期刊资源可以在网络上免费获取和下载，这些资源对科技情报研究具有重要的意义。而开放获取期刊资源由各个期刊机构自行组织，期刊资源在网络上展示的形式可谓是“千刊千面”，但基本上是采用了“模版+数据”的网站建设模式。而在对开放获取期刊资源采集调研中发现，即使是同一期刊里，在不同的时期也会采用不同的模版，这使期刊资源采集带来了不便。本文围绕开放获取期刊资源采集进行了研究，并开发出一套用户交互的开放获取期刊采集系统。具体完成的研究工作主要体现在以下几个方面：

(1)对国内外开放获取期刊资源进行调研、分析、归纳和总结。根据开放获取期刊资源在网页上的组织形式分为单一型资源和组合型资源。总结出开放获取期刊资源具有描述粒度细、描述复杂、描述载体结构多变的主要特点，其中描述复杂和描述载体结构多变是开放获取期刊资源采集的主要困难。这为之后对开放获取期刊资源采集方法的选择奠定了理论基础。

(2)对当前资源采集方式进行了调研和总结，将当前采集方法分为基于接口协议的元数据采集方法和基于Web信息的元数据采集方法。其中基于接口协议的元数据采集方法主要用于遵循相同协议的期刊机构之间的数据采集，此类方法以十分成熟，但局限性很强，无法对网络上存在的大量的开放获取期刊资源进行采集。基于Web信息的元数据采集方法适用于网络资源的采集，而且已经有许多Web信息采集方法成功的应用于一些网络资源的采集，如论坛、新闻等。但开放获取期刊资源多样的组织形式，使得这些方法在应用于开放获取期刊资源采集时有些不适用性。因此本文针对不同类型的开放获取期刊资源总结出了对应适用的采集方法。为开放获取资源采集系统开发奠定了基础。

(3)开放获取期刊采集系统用户行为分析、需求分析、总体设计、编码、测试、评价。根据对开放获取期刊资源和采集方法的调研，以及前期对开放获取期刊资源的人工采集调研，本文将开放获取期刊资源采集系统主要分成了三个模块：用户交互模块、数据采集和网页结构检查模块以及数据存储模块。并使用Java、JavaFX、Jsoup、MongoDB等工具进行开发。实现了用户自定义采集、系统自动形成采集规则、网页结构检查、组合型资源切分、批量采集、数据展示和下载等功能。使开放获取期刊资源采集更多的倾向于计算机的自动采集，解放人的工作量。

6.2论文创新之处

(1)通过大量的开放获取期刊采集调研，总结开放获取期刊资源特点和分类，以及对当前采集方法对比分析，并对每一类开放获取期刊资源对应的采集方法进行了归纳总结。

(2)实现了可视化的开放获取期刊资源采集系统。用户只需进行简单的浏览、点击、选择、填写等操作，系统将自动形成采集规则并对元数据资源进行采集，同时对期刊网站的网页结构进行检查，对结构变化的网页，系统将反馈给用户进行重新选择，进而采集到全面完整的数据。通过多线程提高了资源采集效率。

(3)使用非关系型数据库使系统更加灵活，突破关系型数据库需要预定义数据模式的局限，能很好的满足各种采集需求。

6.3存在的问题

(1)系统界面单一，不够美化，各操作之间的跳转较为生硬，用户在使用起来流畅感不强。

(2)系统在对组合型开放获取期刊资源采集时，对结构固定的文本资源能准确切分和采集，而对结构变化的文本资源采集效果不佳。主要是因为用户根据当前页面的文本进行分割符插入，系统根据待采集资源在文本中的位置形成的采集规则不适用于文本结构变化后的资源。

(3)虽然在引入多线程采集之后，采集时间较单线程而言提升了很多，在网络条件较好时，平均1000篇论文元数据的采集需要7-8分钟。从用户交互的角度而言，用户等待的时间还是过长。

6.4进一步工作展望

(1)美化开放获取期刊资源采集系统界面，是用户采集资源时获得更好的使用体验。进一步优化采集规则形成，更加准确的采集各类型的开放获取期刊资源。

(2)搭建分布式的开放获取期刊采集平台，多台机器分布式采集，进一步提升系统的采集效率。

(2)开放获取期刊资源采集系统的研究是对开放获取期刊资源研究的第一步，为更好的使用网络上大量存在的高质量开放获取期刊资源，应该还有开放获取期刊元数据清洗、开放获取期刊元数据仓库构建、开放获取期刊元数据数据统计分析平台以及开放获取期刊资源分析结构的可视化等等一系列工作。

6.5总结

开放获取期刊资源采集是有效利用网络上大量存在的高质量期刊资源的基础，是期刊资源分析利用的数据来源。因此本文对开放获取期刊资源采集系统的研究正是为了解决这个实际问题。虽然系统存在一些不足之处，但基本上能够满足开放获取期刊资源的采集，并且通过日后系统的测试和使用，能更近一步完善系统的功能。开放获取期刊资源采集研究是对开放获取期刊资源利用的起点，希望本文的研究能为科技情报者的工作带来一些方便，为开放获取贡献微薄之力。