以 Selenium+Chrome 为核心的数据采集系统设计

黄孝伦,王 东
(重庆市卫生信息中心,重庆 401120)

摘 要:医疗大数据往往存在数量庞大、系统异构等问题,如何快速获取数据并解决系统异构性是目前医疗卫生信息化研究亟待解决的问题。该文依托重庆市医疗卫生信息专网,利用网络爬虫技术从不同医院的不同系统中抓取指标数据,并按医改监测平台相关要求对数据进行清洗处理,将全市各医院的异构系统构成一个松散耦合结构,解决医院信息系统数据与医改监测平台指标之间不一致等问题。测试结果显示,该系统可采集到涉及收入支出、医疗服务量、医保报销、分级诊疗、床位使用、医师出诊、重点人群服务及公立医院改革情况等内容的部分指标,基本上可以满足医改监测平台要求,具有开发成本低、配置灵活、可动态调整指标体系等优势,极大地提高了工作效率,降低了数据采集成本。

关键词:医改监测;数据采集;网络爬虫;系统设计;Chrome
中图分类号:TN957.52 **文献标识码**:A **文章编号**:1673-629X(2020)09-0216-05
doi:10.3969/j.issn.1673-629X.2020.09.039

Design of Data Acquisition System Based on Selenium+Chrome

HUANG Xiao-lun, WANG Dong
(Chongqing Health Information Center, Chongqing 401120, China)

Abstract: There are many problems in medical big data, such as large quantity and heterogeneous system. How to quickly obtain data and solve the system heterogeneity is the urgent problem to be solved in the current medical and health information research. Relying on the Chongqing Medical and Health Information Network, we use the technology of web crawler to grab the index data from different systems of different hospitals, and clean the data according to the requirements of the medical reform monitoring platform. The heterogeneous system of all hospitals in the city forms a loose coupling structure to solve the inconsistency between the hospital information system data and the indicators of the medical reform monitoring platform. The test results show that the system can collect some indicators related to income expenditure, medical service volume, medical insurance reimbursement, hierarchical diagnosis and treatment, bed use, doctor visits, key group services and public hospital reform, basically meeting the requirements of medical reform monitoring platform, with the advantages of low development cost, flexible configuration, dynamic adjustment of indicator system, etc., greatly improving work efficiency and reducing data collection cost.

Key words: medical reform monitoring; data acquisition; internet worm; system design; Chrome

0 引 言

医改监测不仅是了解和评价医改的重要数据来源,也是科学决策、推进医改的重要抓手[1]。但如何快速、有效地构建医改监测平台是各省市面临的一个重要问题,这主要是因为各医院系统存在异构性,而且系统数据与部分监测指标之间不一致,因此产生大量的系统改造和接口开放等费用。网络爬虫[2]是一个抓取网页内容的程序,利用网页格式特征进行网页分析[3],可以快速、高效地获取数据,并且可以实时更新数据,是构建医改监测平台指标数据库的有力工具。文中依托重庆市医疗卫生信息专网,利用网络爬虫技术构造了一个采集系统,可从各医院异构系统[4]中抓取指标数据,并按医改监测平台相关要求对数据进行处理,极大地提高了工作效率,降低了指标数据采集成本。

1 关键技术

Selenium[5]是一个开源的、便携式的基于 Web 应用的测试工具集合,最初是为网站自动化测试而开发的。Selenium 直接运行在浏览器中,通过一系列命令来模拟用户操作,如将界面元素定位、窗口跳转、结果

收稿日期:2019-10-31 **修回日期**:2020-03-03
基金项目:重庆市科研重点项目(2015ZDXM026)
作者简介:黄孝伦(1977-),男,硕士,主要从事智能计算研究;通信作者:王 东(1979-),男,博士,主要从事新媒融合、期刊出版、医学检验诊断学等方面的研究。

比较等命令转化成实际的 HTTP 请求在浏览器中运行[6-7]。Selenium 本身不带浏览器,不支持浏览器的功能,需要与第三方浏览器结合使用。Selenium 支持多种平台(Windows,Linux,Solaris)、多种浏览器(IE,Firefox,Opera,Safari)和多种语言(Java,Ruby,Python,Perl,PHP,C#)。Chrome[8]是可在 Headless 模式下运行的浏览器。Headless 浏览器可在 Web 浏览器的环境中提供对网页的自动控制,但其通过命令行接口或使用网络通信来执行,即以浏览器相同的方式呈现和解释 HTML[9-10]。

ChromeDriver 通过 Chrome 的自动代理框架控制浏览器[11]。Selenium+Chrome 借助 ChromeDriver 实现导航到网页、用户输入、JavaScript 执行等功能,简单来说就是用浏览器来对目标 URL 进行解析、CSS 渲染、JavaScript 执行,通过 API 模拟用户行为(鼠标点击、键盘输入),但不提供用户界面渲染。

2　系统设计

2.1　系统架构

本采集系统主要采用 CS 模式,构成一个逻辑集中、物理分布的二级数据采集平台,为实现区域医疗信息监测提供了数据源,如图 1 所示。服务器端负责接收数据并交给医改监测平台进行存储和处理;客户端负责在各医院信息系统中进行数据采集、清洗和传输,其主要采用 Selenium+Chrome 技术构建网络爬虫,根据配置文件定向抓取医院信息资源。

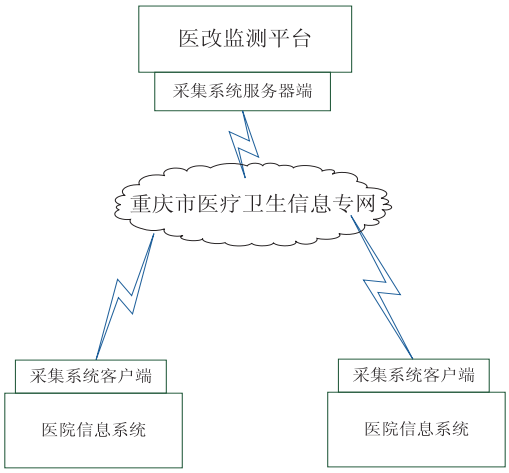

图 1　系统架构

2.2　系统流程及主要原理

本采集系统的核心原理是利用网络爬虫技术构建指标数据库。与传统网络爬虫不同的是,本采集系统的客户端根据指标特点构造网络爬虫,在医院信息系统中抓取数据,大大简化了网络爬虫的流程。由于各个医院采用的是不同厂商的信息系统,数据所依赖的应用系统、数据库管理系统或操作系统以及在存储模式上都存在异构性,因此每个客户端通过配置方式获得指标数据在每个页面中的元素位置,然后抓取相应数据,抓取流程见图 2。在程序运行时,客户端根据配置文件生成 Java 源代码[12-13],然后调用 javac 来编译[14]。这种方式可以满足各医院自定义配置文件的需求,解决了异构系统带来的阻碍。

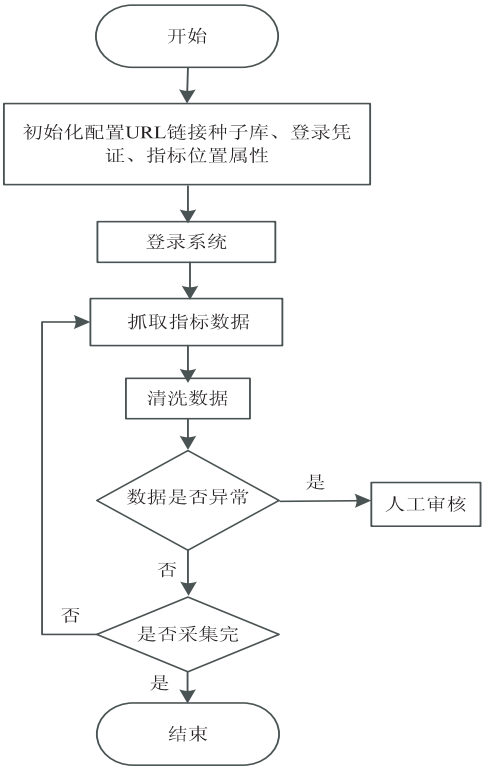

图 2　抓取流程

2.2.1　URL 及登录信息配置

根据指标的数据来源,设置相应的 URL 链接种子库,遍历链接种子库中所有的 URL 即可获得医改监测平台所要求的所有指标数据。为了提高响应速度,本采集系统针对链接种子库中每一条 URL 生成一个网络爬虫,以并行方式分别抓取相应系统中的数据。登录验证是网络爬虫抓取需要用户权限的数据的必要模块。在配置 URL 的同时,每一条 URL 对应一对用户名和密码,然后根据该凭证进行模拟登录。配置后,系统根据配置信息生成登陆代码。

```
public boolean Logoin(WebDriver driver, String url, String username, String password){
    try{driver.get(url);
    //根据配置信息获取用户名文本框
    WebElement usertbox = driver.findElement(By.name("txtUserName"));
    Usertbox.sendKeys(username);
    //根据配置信息获取密码文本框
    WebElement pwtbox = driver.findElement(By.name("txtPassword"));
```

```
pwtbox. sendKeys(password);
//根据配置信息获取登陆按钮
WebElement lgbtn = driver. findElement( By. name( " btnLog-
in" ) );
Lgbtn. click( );
……
}
```

由于部分系统的 session 凭证设定了时效,session 自动失效时,系统会要求用户重新登录。为了解决这个问题,在检测到异常页面 URL 时重新模拟登录,获取新的 session;同时,当抓取完数据后,自动退出系统。浏览器登录系统抓取数据时采用 Headless 模式,无需显示图形界面,代码如下:

```
public static WebDriver getWebDriver( Page page) {
System. setProperty( " webdriver. chrome. driver" ," E:\\Web
\\chromedriver_win32\\chromedriver. exe" );
ChromeOptions chromeOptions = new ChromeOptions( );
//设置为 Headless 模式
chromeOptions. addArguments( " --headless" );
WebDriver driver = new ChromeDriver( chromeOptions);
driver. get( page. getUrl( ) );
return driver;
    }
}
```

2.2.2 数据抓取

登入系统后,采用配置方式获取指标数据在系统页面中的位置元素。虽然各医院的业务系统存在异构性,但业务流程大体上是一致的。因此在抓取指标数据时,本采集系统根据不同系统设计了一个基本的流程模版。各医院在配置时可依据该模版进行调整。以“向基层医疗卫生机构转诊人次数”指标为例,其所在的系统截图如图 3 所示。在配置该指标的位置标签时,需先点击“双向转诊”按钮展开菜单,然后在菜单中点击“下转管理”按钮,最后点击“查询”按钮,在该页面抓取总条数即获得该医院向基层医疗卫生机构转诊的人次数。该过程对应的配置文件及对应的关键代码如下。元素“el-menu”用于存储菜单名(元素“el-button”用于按钮名),元素“value”存储对应元素在页面中的位置属性。

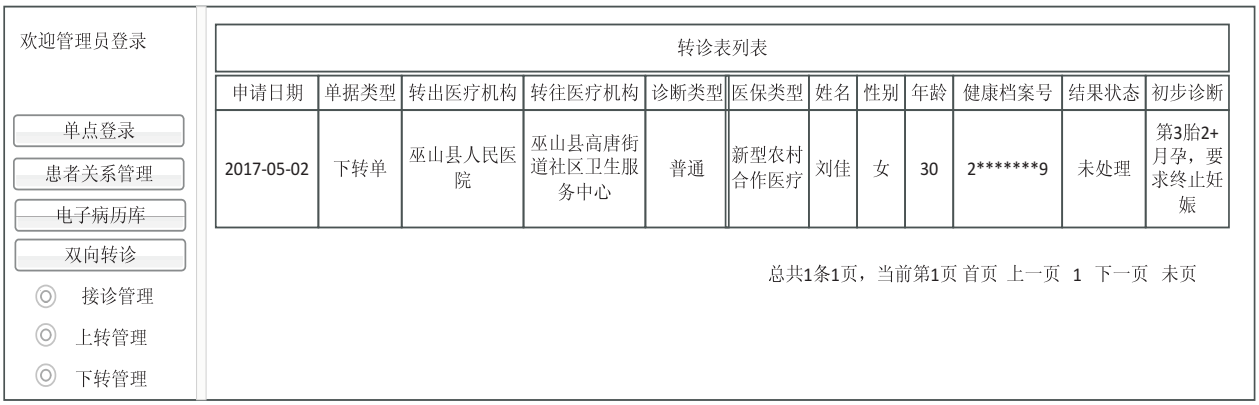

图 3 转诊表列表图

```
<path>
<path1>
<el-button>双选转诊</el-button>
<value>By. xpath( "// * [ @ id = 'form1' ]/bt[ 5 ]" ) </value>
</path1>
<path2>
<el-menu>下转管理</el-menu>
<value>By. xpath( "// * [ @ id = 'href' ]/h3/a" ) </value>
</path2>
<path3>
<el-menu>查询</el-menu>
<value>By. xpath( "// * [ @ id = 'xzcontent' ]" ) </value>
</path3>
<target>
<el-menu>向基层医疗机构转诊人次数</el-menu>
<value>By. xpath( "// * [ @ id = 'totalcount' ]/h1" ) </value>
</target>
</path>
```

```
public void ZZTarget( WebDriver driver) {
    ……
//跳转到 leftFrame
driver. switchTo( ). frame( " leftFrame" );
//点击双选转诊按钮
WebElement path1 = driver. findElement( By. xpath( "// * [ @
id = 'form1' ]/bt[ 5 ]" ) );
path1. click( );
//点击下转管理菜单
WebElement path2 = driver. findElement( By. xpath( "// * [ @
id = 'href' ]/h3/a" ) );
path2. click( );
//跳转到 rightFrame
driver. switchTo( ). frame( " rightFrame" );
//点击查询按钮
WebElement path3 = driver. findElement( By. xpath( "// * [ @
id = 'xzcontent' ]" ) );
path3. click( );
```

```
    //获取向基层医疗机构转诊人次数
    WebElement zztarget = driver. findElement( By. xpath( "// *
[@ id = 'totalcount']/h1" ) ) ;
    Stringxzcount = zztarget. getText( ) ;
        ……
    }
```

2.2.3　数据存储及清洗

由于网络爬虫抓取的大都是医院的业务系统,不断抓取数据时可能会影响医院业务工作的正常进行,因此本采集系统在客户端将网络爬虫抓取的原始页面文件进行缓存,减少网络爬虫对医院业务系统造成的负担。在出现问题时,通过读取缓存页面获取数据,不必重新进行抓取。

由于抓取的数据来自不同医院的多个系统,在格式、内容方面都需要进行相应的清洗处理[15],主要包含以下几个方面:

(1)指标汇总处理。

由于各医院信息系统建设时依照的是原有相关标准,如系统按 2004 版收费项目设定了“挂号费”及“诊查费”。但是,医改监测平台是按新的相关标准开发的,如按新修订的《重庆市医疗服务项目价格表》增加了“诊察费收入”,其与“挂号费”及“诊查费”对应。这导致医院系统与医改监测平台存在指标数据不完全一致的问题。因此,抓取的数据以二维表(见表 1)的形式存储在客户端,该表中的指标数据与医院系统一一对应,便于跟踪分析。同时,按照医改监测平台的指标要求抓取医院信息系统中“挂号费”及“诊查费”,然后进行汇总处理。

(2)数据描述一致性处理。

各医院是根据自身特点建设的系统,在某些描述方面可能存在不一致性,如“患者情况”这个指标中,有的医院将患者情况采用“1 = 感受较好,2 = 感受一般,3 = 感受较差,4 = 无感受”,而有的医药可能采用其他方式进行描述。因此,在客户端抓取到数据后要按医改监测平台要求,对数据描述进行一致性处理。最后,将清洗好的标准数据上传至医改监测平台。

表 1　指标二维表

指标	对应的 URL	值	对应指标
挂号费	http://172.16.2.103:6022/his/sf/001.html	10.0	诊察费收入
诊查费	http://172.16.2.103:6022/his/sf/004.html	15.0	

2.2.4　数据质量控制

医改监测平台对数据的准确性要求比较高,同时对响应速度也有一定要求。为了更科学严谨地完成数据抓取过程,本采集系统通过以下 3 个方面进行改进。

(1)将指标集根据系统进行分类,如财务系统类指标、HIS 系统类指标、RIS 系统类指标等。每一类指标对于一个 URL,即对应一个网络爬虫。这样即可保证指标数据的准确性,还可以通过并发方式提高响应速度。

(2)部分医疗指标数据具有阶段性,如“门诊人次数”可能因秋季感冒发病率高而剧增,因此采用分阶段统计法对抓取的数据进行准确性验证。如果抓取的数据与统计值存在较大差异时,发出系统警报,采用人工方式对数据进行确认或重新抓取。

(3)根据医改监测指标定义及审核条件对抓取数据进行审核,如病床使用率定义为实际占用总床日数/实际开放总床日数×100%,且要求小于 150%。抓取数据时按该定义进行复核,如果病床使用率大于 150%,则警报提示至人工核对处理。

3　应用分析

自 2017 年启动重庆市公立医院综合改革以来,本中心针对 226 家参加改革的公立医院设置了预约量、门急诊量、医务人员数量、医生排班、医生坐诊日志、病床使用情况等 116 个基础监测指标及医疗服务效率、质量、病人医药费用等 40 余项分析监测指标,并将 156 个监测指标再次细分为多个细项数据。

经测试,本采集系统可采集到涉及收入支出、医疗服务量、医保报销、分级诊疗、床位使用、医师出诊、重点人群服务及公立医院改革情况等内容的部分指标(图 4 为全市次均门诊费用指标的明细列表),基本上可以满足医改监测平台要求。

其优势如下:

(1)开发成本低,各医院不需承担接口开放、升级等费用,简单安装即可上线使用;

(2)配置灵活,各医院可根据自己的系统特点按医改监测平台的指标要求进行配置即可,无需进行系统改造,且不需考虑异构系统带来的问题;

(3)可满足动态调整指标体系的需求。

本采集系统的不足之处在于初始化配置时,工作量较大,且抓取的数据量较大时对 HIS 系统业务有一定影响(加大硬件性能可以解决或在业务较少期间进行抓取)。


渝医改监测表

⊞ 月报（医院类）

全市门诊次均费用

全市门诊次均费用

月份：20**月1月

单位：重庆市

分类	门急诊收入	门急诊人次数	次均门诊费用（元）							
			小计	诊察费	药品费	检查费	治疗费	手术费	卫生材料费	化验费
**区	2235.02	18952	100.1	10.9	22.2	15.0	9.0	19.0	7.5	13.2
**区	3000.00	25000	120.0	12.0	24.0	20.0	8.0	20.0	8.0	12.0
**区	2400.00	21000	115.6	11.6	22.8	18.5	8.0	20.0	8.0	12.5
**区	2700.00	23000	118.0	11.9	23.5	17.9	7.5	19.0	7.5	11.0
**区	1352.60	16070	89.5	8.9	20.0	12.0	6.0	18.0	6.5	10.0

总共8页，第1页 上一页 下一

图 4 全市次均门诊费用图

4 结束语

医疗大数据往往存在数量庞大、系统异构等问题，如何快速获取数据并解决系统异构性是目前医疗卫生信息化研究亟待解决的问题。文中采用网络爬虫技术在各异构系统间进行抓取，将全市各医院的异构系统构成一个松散耦合结构，保证了数据的完整性、准确性，提高了数据质量和利用效率。所构建的医改监测指标数据库，在医疗大数据的提取、分析方面具有一定实用价值。

参考文献：

[1] 国家卫生健康委员会统计信息中心. 统计信息中心召开 2017 年医改监测方案讨论会[EB/OL]. [2019-05-21]. http://www.nhc.gov.cn/mohwsbwstjxxzx/s7967/201705/aa5cad2b7018420eaf6e7fc40c981949.shtml.

[2] KAUSAR M A, DHAKA V S, SANJEEV S K. Web crawler: a review[J]. International Journal of Computer Applications, 2013, 63(2): 31-36.

[3] 杨 洋, 李晓风, 赵 赫, 等. 基于网络爬虫的文献检索系统的研究和实现[J]. 计算机技术与发展, 2014, 24(12): 35-38.

[4] 陈红玲, 郎六琪, 刘立勋, 等. 远程医疗监护诊断异构系统的集成实现[J]. 计算机测量与控制, 2014, 22(12): 3929-3931.

[5] MAURIZIO L, DIEGO C, FILIPPO R, et al. Repairing selenium test cases: an industrial case study about web page element localization[C]//2013 IEEE sixth international conference on software testing, verification and validation. Luxembourg: IEEE, 2013: 487-488.

[6] 沈大框, 黄永锋, 罗保国. 基于 Selenium 的 Web 自动化测试解释器[J]. 计算机系统应用, 2018, 27(11): 51-56.

[7] SATISH G, RAHUL J, DHANASHREE G. Analysis and design of Selenium webdriver automation testing framework [J]. Procedia Computer Science, 2015, 50: 341-346.

[8] 傅建明, 梅戌芬, 郑 锐. Chrome 扩展安全[J]. 武汉大学学报: 理学版, 2019, 65(2): 111-125.

[9] 马富天. Web 应用程序跨站脚本漏洞检测技术研究与实现[D]. 无锡: 江南大学, 2018.

[10] 刘 源. 一种基于模拟浏览器行为的 XSS 漏洞检测系统的研究与设计[D]. 北京: 北京工业大学, 2016.

[11] 陈萧宇, 黄 震, 刘譞哲, 等. Scratch: 一个基于 Chrome 浏览器的用户操作捕捉与回放工具[J]. 计算机科学, 2014, 41(11): 112-117.

[12] 吴家菊, 纪 斌, 刘振吉, 等. 一种将 XML 模式转化为编程语言的算法[J]. 现代电子技术, 2019, 42(11): 169-173.

[13] PAVEL L, JOACHIM N. A uniform programming language for implementing XML standards [C]//41st international conference on current trends in theory and practice of computer science. Pec pod Sněžkou: Springer, 2015: 543-554.

[14] 吴泽智, 陈性元, 杜学绘, 等. 基于自动机的 Java 信息流分析[J]. 计算机应用研究, 2019, 36(1): 246-249.

[15] BLOODGOOD M, STRAUSS B. Data cleaning for XML electronic dictionaries via statistical anomaly detection[C]// 2016 IEEE tenth international conference on semantic computing (ICSC). Laguna Hills: IEEE, 2016.