第五届“泰迪杯”
全国数据挖掘挑战赛

论
文
报
告

通用论坛正文提取方法研究与实验报告

摘要：信息价值的作用日益凸显，利用网络爬虫技术提取论坛网站的有价值信息效果明显。报告针对论坛网站主帖和回帖的标题、作者、时间、内容的提取，提出了三种可行方案，并通过对比分析、综合三种方案的优缺点，提出最优方案，且该方案对网络论坛具有普遍适用性。

该方案基于网页分块的论坛爬虫思想，利用CSS选择器和正则表达式（用于筛选时间），结合HTML DOM树及HTML标签的特性，考虑到部分网络论坛的特殊性，具有高度通用性。

整个实验过程（解题步骤）大致分为以下几个步骤：

第一步选取样例数据。除大赛组委会提供的测试样例数据外，还选取了具有权威性的网站排名网站“站长之家”排名前300的论坛作为测试样例数据，大大提高了方案的普适性。

随后提出方案并论证各个方案的可行性。提出的初步方案为：1）利用网络开源论坛模板分类选择；2）使用HTML标签属性共性进行分类筛选；3）利用简单的经验进行判断。与此同时，通过开源模板统计等具体数据的对比分析，综合方案的优缺点和实施难度，提出以方案二为基础的最优方案。

第三步是论坛标签统计和主回帖判断。通过使用Firefox、Chrome等浏览器的调试工具，统计各个论坛目标内容的共性，形成汇总表。由于部分URL所指向的页面中仅包含回帖内容，因此需要通过解析URL中的参数或者格式进行主回帖的判断。通过判断，设计了一种具有普遍适用性的URL主回帖判断方案。

最核心的部分是程序编写。由于所选方案中需要用到大量的CSS选择器，所以选择对CSS选择器支持完全的、使用要求宽松的MIT license授权的开源HTML解析库“jsoup”作为程序的核心依赖库，用以完成对HTML的爬取和解析。同时，设计编制特例处理通用框架。由于任何先进的算法和理论都不能保证程序对所有的论坛网站均适用,所以设计了一种用于处理特殊论坛的简单框架。

最后一步是设计测试指标和程序测试。依据现有的查全率和查准率的概念，结合论坛通用爬虫的特点，设计了一种用于论坛通用爬虫的测试指标。最后通过人工筛查的方式对程序运行结果进行判定，并统计计算得出相应的测试指标。

在报告的最后分析了网络论坛爬虫的核心思路及其价值，浅谈完成实验的收获和感想，同时对网络论坛要素爬取提出新的展望。

关键词：网络爬虫；网络论坛；通用爬虫；CSS选择器；Java

Research and experimental report on the general method of text extraction in network forum

Abstract: The value of information is becoming more and more important. And, the effect of web crawler technology to extract valuable information from the forum site is valid. This report which point to the extract of the forum site's title, author, time and content, put forward many feasible schemes. Through comparison and analysis, the optimal scheme which is generally applicable to Network Forum. is selected.

The idea is based on Web partition. Using CSS selectors and regular expressions (for filtering time), the solution combines with the characteristics of HTML DOM tree and HTML tags and take into account the special nature of some network forums, which is with a high degree of versatility.

The process of the experiment (Question Solving Steps) can be divided into the following steps:

The first step is to select the sample data. In addition to the test data provided by the Organizing Committee of the contest, but also we select the authority of the site ranking website named ChinaZ.com to choose the ranking of the top 300 as the test sample data. That will improve the universality of the program greatly.

Then put forward the scheme and demonstrate the feasibility of each scheme. The proposed schemes are: 1) Using the network open source forum to template and selection; 2) Classification and screening using HTML tag attributes; 3) Using some simple experiences to get judgement. At the same time, through the comparative analysis of open source template statistics, and analyzing the advantages and disadvantages of the integrated program and the difficulty of implementation, we can know that the second proposed scheme is the optimal scheme.

The third step is that count the forum label and judge the main content and reply content. It create a summary table by using the Firefox, Chrome and other browser debugging tools and counting the common content of each forum target. As part of the URL point of the page contains only the reply content, so we need to resolve the argument in the URL format on the main content and reply content.

Programming is the core step. Because the selected scheme requires a large number of CSS selectors, it's better to choose a core of the program database. The database is supported by CSS selector completely, and use the loose requirement analysis of open source HTML MIT license authorized "jsoup". Meanwhile, it need design the general framework for handling special cases. Since any advanced algorithms and theories can not guarantee that the program can be applied to all forum sites, a simple framework for dealing with special forums is designed.

The final step is to design test metrics and program testing. Based on the concept of recall ratio and precision ratio, combined with the characteristics of the general crawler, we designed a kind of test index for general crawler. Finally, the results of the program are judged by the way of manual screening, and the corresponding test indexes are obtained by statistical calculation.

At the end of this paper, the core idea and the value of web crawler are analyzed deeply. And share the feelings and gains through the experiment. Last but not least, we put forward some perspectives on the factors of network forum crawling.

Keywords: web crawler; Network Forum; general purpose web crawler; CSS Selector; Java

目录

一、引言

随着信息技术、信息产业的飞速发展，信息传播和更新的速度日新月异，这也使得信息的增长呈现井喷式发展，增长速度异常迅猛。如何利用好这些信息，逐渐成为企业、政府关注的焦点。传统波特价值链理论将信息作为增值过程中的支持要素，而非增值源泉，但是随着企业信息化进程的深入发展，信息有可能变成有用的资源[1, 2]。深层挖掘信息价值，使其与信息技术结合、与企业流程再造以及生产技术融合，就能从根本上提高企业的竞争力，实现价值增值[3]。在政府方面，大量信息的搜集、整理，以价值为导向的数据挖掘，能为政策的制定、方案的出台提供很好的数据与理论支撑。那么，如何在繁杂的信息中抓取有价值的信息呢？网络爬虫技术可以算得上是一个很好的解决方案[4]。

网络爬虫（Web Crawler）又称为网络蜘蛛（Web Spider）或 Web 信息采集器，是一个自动下载网页的计算机程序或自动化脚本，是搜索引擎的重要组成部分。网络爬虫通常从一个称为种子集的 URL 集合开始运行，首先将这些 URL 放到一个有序的特定爬行队列里，按照一定的顺序从中取出 URL 并下载所指定的页面，分析页面内容，提取新的 URL 并存入待爬行的 URL 队列里，如此反复，直至 URL 队列为空或达到某一终止条件，从而遍历 Web[5]。目前，已有一些聚焦网络爬虫技术、主题爬虫网络技术。但是，对于论坛这种非结构化的网页来说，爬取还有一定的难度[6, 7]。

移动互联网已经潜移默化地使人们的生活方式发生改变，越来越多的兴趣社区、交流论坛应运而生。网民可以随时随地在论坛分享新鲜事物、讨论热门话题。但是，这些分布在互联网的各个角落，被成千上万的网页所湮没，因此，正是需要网络爬虫技术将这些“信息孤岛”联合成一个集体[8-10]。这样，企业可以从中获取商机、政府可以从中找出趋势，以便为决策提供支持。

本项目着重阐述如何利用网络爬虫技术快速而准确地获取所需要的论坛信息：主帖和回帖的标题、作者、时间、内容。同时，消除可能存在的“噪音”，如广告、垃圾营销等[11, 12]。重点强调其通用性原则，不论何种网络论坛、不论何种论坛模板，都能通过设计的技术快速而准确地获得上述目标要素。本报告包括引言、实验方案、实验过程、实验结果、实验感想五个部分。

二、实验方案

2.1 初步方案设计（解题思路）

为了达到查全、查准且执行速率快的目的，本项目初步提出以下三种解决方案（图 1）：

（1）方案一：先利用网络论坛模板来将论坛进行分类选择，目前常用模板如 Discuz、Colpwind、Phpwind、Drupal、Joomla、Wordpress 等。属于某一特定论坛模板的 URL 归为一类，进行一次分类以后，再利用 HTML 的标签属性进行筛选、抓取。

（2）方案二：通过对 HTML DOM 树进行多次深度优先遍历，利用 HTML 标签中的 id 属性值、class 属性值、标签名称或者其他共性选择出特定的标签集合，提取出标签中的文字，组合成论坛帖对象。

（3）方案三：通过简单的经验进行判断，例如时间的格式可能是 YYYY-MM-DD HH:mm:ss；一般来说主帖长度比回帖长；页面第一个正文为主帖，以后均为回帖等。总结粗放的经验来做分支判断，而后再逐渐细化[5, 13-17]。

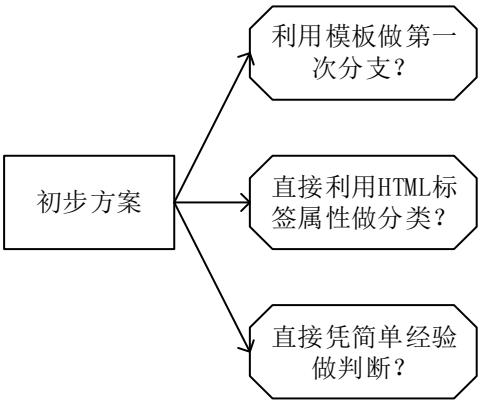

图 1 初步方案

2.2 方案具体分析

2.2.1 方案一可行性分析

首先对第一种方案进行试验，将所有论坛粗略分为使用模板类和自主开发类。考虑到普适性要求，除却实验给出的样例数据，将统计范围扩大到网络上论坛排行。通过在网络上搜集各类论坛的排名，综合数据可靠性、权威性和来源的真实性，选择了“站长之家”（chinaZ.com）Alexa 论坛网站排行榜中排名前 300 的论坛，

如下图：

图 2　chinaZ.com（站长之家）Alexa 排名

（1） Top300 模板统计

排名前 300 论坛分类统计结果如下（表 1）。

表 1 站长之家排名前 300 排名类型统计

	个数	占百分比
Discuz 模板	63	27.7%
Phpwind 模板	6	2%
Wordpress 模板	1	0.3%
非特定模板	210	70%

对于应用模板的论坛，其论坛主页左下角都有模板来源，例如智能电视网应用了 Discuz 模板，其网站主页便标明“Powered by Discuz”，如下图，其他模板类似。

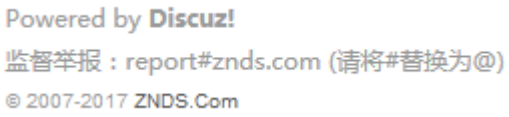

图 3 Discuz 模板引用示例图

因上述模板中以 Discuz 为主，其他类型模板所占比率均小于 3%，故对 Discuz 模板占有率及其变化情况进一步统计，统计结果如下：

表 2　Discuz 模板占有率统计

总体样本	计数	比率	累计个数	累积比率
Top　0-100	21 个	21%	21 个	21%（占前 100）
Top100-200	32 个	32%	53 个	26.5%（占前 200）
Top200-300	30 个	30%	83 个	27.7%（占前 300）

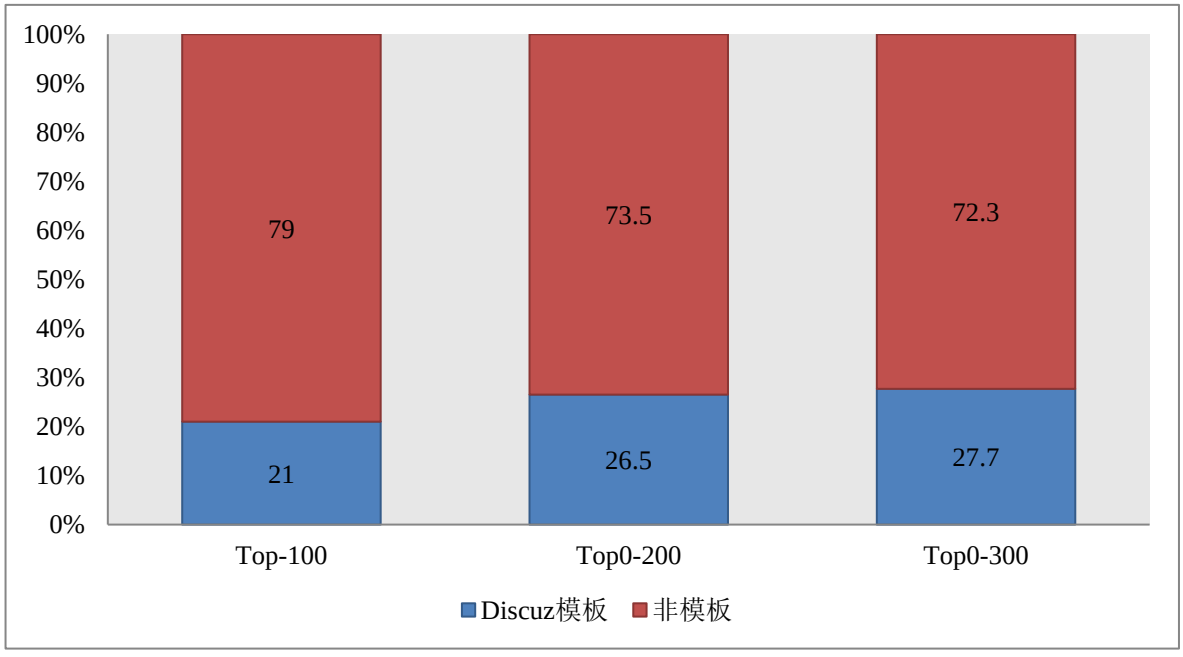

图 4　Discuz 模板占总体的结构相对数

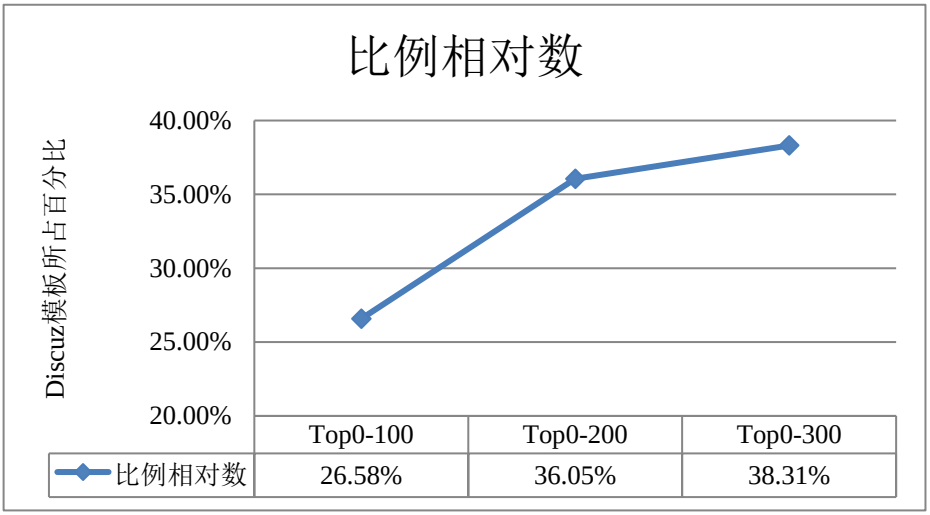

图 5　Discuz 模板所占论坛数量与非 Discuz 模板论坛数量比例相对数

由以上图表可知，Discuz 模板论坛所占比例虽在逐步上升，但总体比重仍在40%以下。相对而言非模板论坛数目较多。从总体来看，三类模板的总数占比不到三成，超过 2/3 的论坛采是自主开发或依据模板修改而来，而没有完全采用上述模板，使得网络爬取的困难大大增加。同时，通过对比发现，模板的不同版本标签属性也会有差别，如 Discuz X2、Discuz X3 之间也会有差别，为提高查准率，不能将其笼统的归为一类。

（2）样例模板统计

将组委会提供的样例数据（共 177 条 URL，涉及的论坛网站个数为 82）进行了分类统计。得出结果如下（表 3）：

表 3 样例数据的模板类型统计

	个数	占百分比
Discuz 模板	10	12.2%
Phpwind 模板	6	7.3%
Wordpress 模板	0	0
非特定模板	66	80.5%

从样例数据汇总统计来看，样例数据中 80%以上的的网络论坛均未使用上述模板。

（3）结论

经过统计，82 个样例论坛属于排名前 300 的仅有 15 个，不在前 300 的为 67

个，占比 81.7%，故而将样例论坛与排名前 300 论坛分别统计是十分必要的。

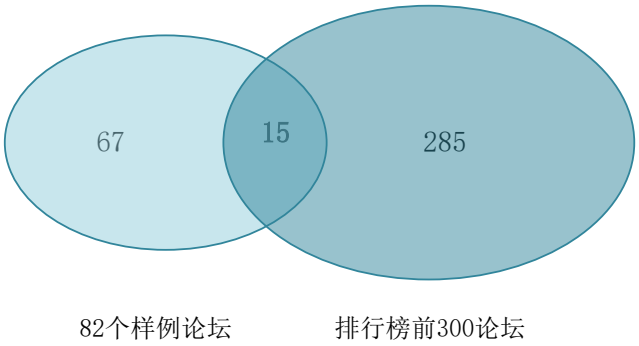

图 6 论坛分布图

对方案一的可行性研究，综合以下两点原因：

1）在排名前 300 以及样例论坛里面模板使用比率都较低，模板的占有率均处于 10%~30%；

2）即便是同一模板，不同版本之间也会有差别，意味着经过模板筛选后，还需进一步细化，大大增加工作量。

得出结论：方案一从模板进行筛选不可行。但对模板的研究仍是有一定意义的，已知模板论坛的标签具有一定规律，而在进一步研究后得出自主开发类论坛的属性标签也有此种规律的结论，故而将方案二作为主要方案进行实验。

2.2.2 方案二核心思想

在传统的 B/S 架构中，网页可看作由多个块状元素构成的。

图 7 网页分块图（主帖）

图 8 网页分块图（回帖）

而在 HTML DOM 树中，网页中的块状元素可看作是由多个 HTML 标签组成的，每个标签包含的不同内容对应着页面上各个块状元素。查询这些块状元素的方法，如下图：

图 9 查看元素示意图

在浏览器上任意打开一个帖子，选中元素如作者，使用查看器查看元素，可知其 class 属性值为 authi。

在浏览器的渲染过程中，每个标签的位置和样式由默认值和自定义 CSS 样式共同决定。各个论坛为保证良好的用户体验，增加用户量，需要友好的交互界面。而默认的标签样式和默认位置难以满足需求，所以需要通过自定义 CSS 样式来自定义标签的位置和样式。

自定义 CSS 样式一般有两种方式：

1）直接自定义 HTML 标签的 style 属性；

2）通过 CSS 选择器自定义一个或者多个标签。

为保证 CSS 代码的复用性，一般使用方式 2 进行标签样式自定义。因此可通过如下方式通过 CSS 选择器获取到对应的内容：

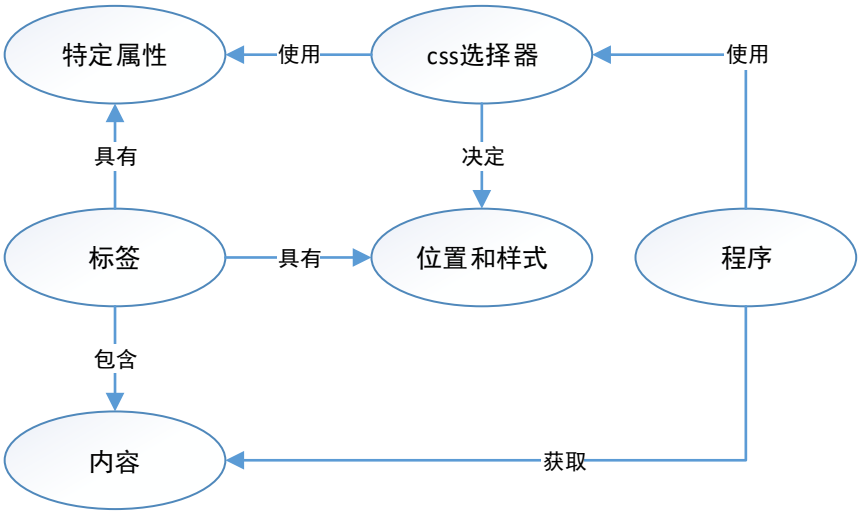

图 10 方案二核心思想图

如上文所述，主帖和回帖的每个字段都包含在相应的具有名称、位置、或者属性特征的 HTML 标签中。而这些标签都具有不同的样式，样式是由 CSS 确定的，CSS 语法通过 CSS 选择器选择不同的标签从而给一类标签设定不同的样式，并由浏览器显示在页面上。因此，可以使用 CSS 选择器来选择这些字段所在的标签，从而获取字段值。

2.2.3 方案三可行性分析

为保证程序的通用性，需同时兼顾大论坛和小论坛，因此接利用简单经验会出现诸多问题。

例：有些 URL 是论坛的第 2 页或第 3 页，并不包含主帖，故而判定第一条 content 是主帖的结论不完全准确；主帖的长度并不一定比回帖长，比如一些问答帖极有可能回帖长度超过主帖；有些论坛的时间格式不是 YYYY-MM-DD HH:mm:ss，而是动态生成发表于 XX 分钟前、XX 小时前。

因此，虽然用经验判断可以大大减少时间和工作量，但查全率和查准率都不能达到预期标准。

由此得出结论：方案三利用简单经验进行判断也是不可取的。但也不是全盘否定经验的判断，简单经验判断可以作为方案二用 CSS 样式属性选择的支撑，但不起决定性作用。

2.2.4 方案综合分析

综合对三种方案的可行性研究分析，可知将每个方案独立进行的思路并不正确，而应将三种方案融合成一种新方案——以方案二为主，方案一、三为辅。在新方案中，以方案三作为 HTML 标签提取的经验支撑，如内容属性标签大多含有 content 字符串，故而在使用 CSS 样式选择器时主要提取相关内容，可大大节省提取时间；此外，方案一中属于同一模板的论坛其 HTML 标签基本相同，故可将方案一并入方案二。鉴于新方案与方案二的核心思想一致，为使用简便，此后对新方案仍称为方案二。

方案二核心思想：利用 HTML 标签中的 id 属性值、class 属性值、标签名称或者其他共性进行筛选，同时结合排名前 300 以及 82 个样例的论坛经验总结综合作为支撑依据，来进行目标要素的爬取。

三、实验过程

3.1 前期准备

3.1.1 样本统计分析

（1）首先对给出的样例进行了分类统计，177 个帖子属于 82 个论坛，根据这些论坛的性质和应用，又可做出以下分类：

1）腾讯：5 个，腾讯娱乐、腾讯汽车、腾讯电脑管家、腾讯手机管家、腾讯大粤网

2）网易：2 个：网易居家论坛，网易女性

3）有关电子产品、软件的论坛：18，涉及的品牌有 360、华为、酷派、联想

4）游戏论坛：5 个

5）汽车论坛：8 个

6）亲子、女性：8 个

7）经济、股票：8 个

8）异常论坛：6 个

9）娱乐八卦：3 个

10）地方性论坛：17 个，北京、上海、连云港、南阳、湖南、广西、大粤、深圳、晋江、青岛、海南、广州、济南、宁波；其中上海有 3 个：上海网，上海滩论坛，四四论坛

11）综合大论坛：6 个，环球论坛、天涯户外、华声论坛、人民网、西祠胡同、海内外

综合统计如表 4：

表 4 样例论坛分类表

	电子产品、软件	地域性论坛	汽车	亲子、女性
个数	18	17	8	8
	其他论坛	综合论坛	游戏	娱乐八卦
个数	6	6	5	3

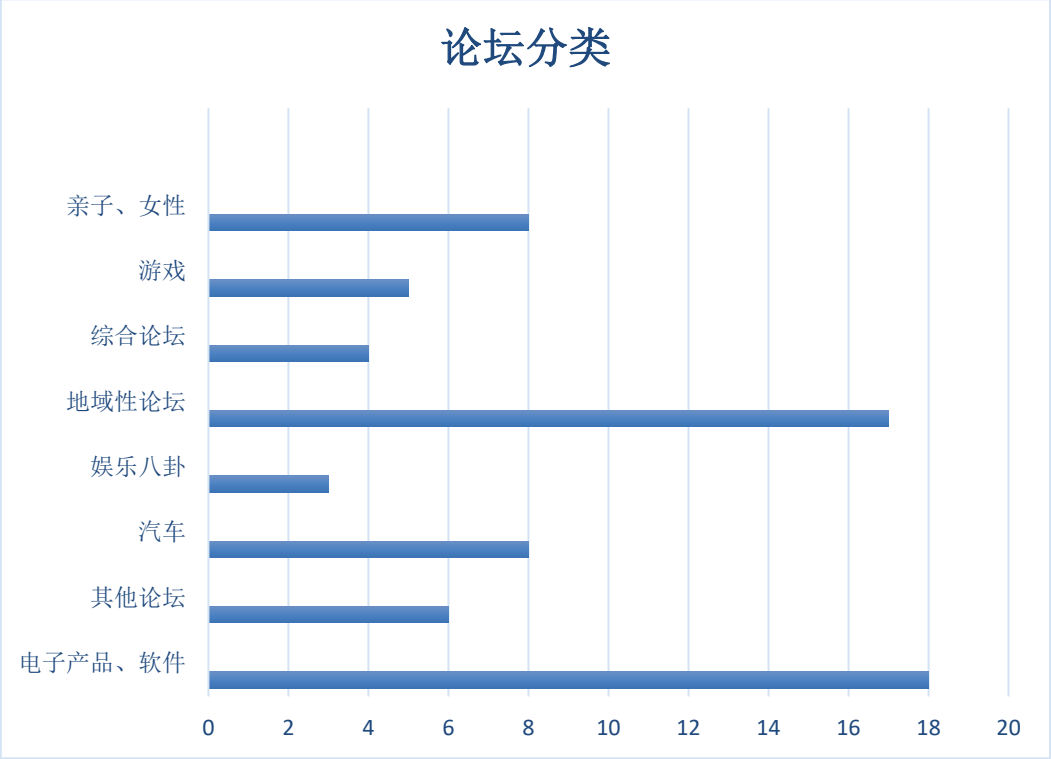

图 11 论坛分类图

（2）根据对样例帖子内容的分析，可将帖子话题概括为：

股票分析、原油价格分析、明星八卦、体育、汽车、奶粉、游戏策略、数码产品、淘宝刷单、微信营销、招聘信息、节假日、情感。

（3）分析总结：从统计结果看，样例所涉及的论坛和帖子类型较为广泛，对于程序的普适性要求极高；此外，部分论坛如异常论坛会产生 URL 失效、禁止内容爬虫等问题，在之后的实验过程中均不使用此部分论坛。

3.1.2 论坛标签统计

（1）由题目要求可知，每个帖子无论主帖或是回帖，都需提取出标题（title）、时间（publish_date）、作者（author）、内容（content），先后对 Top300 和 177 个样本的帖子源代码中相关内容选择器进行提取分析，常见格式如：

标题：

```
<h1 class="ts">
    <span id="thread_subject">【第 2 件半价】</span>
</h1>
```

作者：

```
<div class="authi">
    <img src="#">
    <a href="#" target="_blank" class="xw1">麻辣 e 仔</a>
</div>
```

时间：

```
<div class="authi">
    <img class="authicn vm" id="authicon61216918" src="#" />
    <em id="authorposton61216918">发表于 2016-2-16 15:48</em>
</div>
```

内容：

```
<div class="pct">
    <div class="pcb">
        <div class="t_fsz">第二件半价……
        </div>
    </div>
</div>
```

（2）根据大量的提取结果，得出以下数据，由数据可知，CSS 选择器之间具有极大的相似性，而主帖与回帖的区别仅在于回帖几乎没有标题，故而提取方法是相同的，之后仅提取上述四个 CSS 选择器。

CSS 选择器提取部分表格：

表 5 CSS 选择器提取表示例

网站	作者	标题	内容
天涯社区	uname	<h1 class="atl-title">	
天涯论坛	uname	<h1 class="atl-title">	
东方财富网股吧			
美食天下吃货俱乐部		<h1>	
虎扑体育论坛			
铁血论坛		<h1 class="post_tit">	
汽车之家汽车论坛		<h1>	内容由 script 加载
城市中国		<h1>	内容由 script 加载
凯迪网络			<div class="posts-cont"
凯迪社区			<div class="posts-cont"
智能电视网		<h1 class="ts">	
小米官网手机社区	class='user_name'	<h1 >	
西祠胡同		<h1 class="ts">	
猫扑网			评论和内容未找到
小米手机社区官方论坛	class='user_name'	<h1 >	
得意生活		<h1 class="ts">	
厦门小鱼网		<h1	
吉和网	<p class="news_meta">	<h1 class="news_title">	
色影无忌论坛	<h2 class="title1"><b class="bt">作者:	<h1 class="title">	
幼儿学习网	<p class="article_info">	<h1 class="title">	
宽带山		<h1	
乐彩论坛		<h1 class="ts">	
九游论坛		<h1 class="ts">	

3.1.3 主回帖判断分析

根据题目要求与对样例帖子的观察可知，每个 URL 指向的页面可能是包含主帖的页面（以下简称为**第一页**）或仅包含回帖的页面（以下简称为**其他页**），为区第一页与其他页，对样本给出的 177 条 URL 进行统计，部分统计结果如下所示：

表 6 样例主回帖统计表

网址	第一页	非第一页
36.01ny.cn	thread-4726712-1-1	thread-4726712-2-1
8.7k7k.com	thread-869666-1-1	thread-869666-2-1
baa.bitauto.com	thread-10874068 or thread-10874068-1	thread-10874068-2
bbs.360.cn	thread-14899653-1-1	thread-14899653-2-1
bbs.9game.cn	thread-20674778-1-1	thread-20674778-2-1
bbs.auto.ifeng.com	thread-2854693-1-1	thread-2854693-2-1
bbs.cheshi.com	thread-5000217-1-1	thread-5000217-2-1

初步分析可知 URL 共分如下情况：

（1）有*page*参数：page=1 时为第一页；page=2 时为第二页；

（2）有参数但无*page*参数，此时为*page*被省略，为第一页

（3）无参数，但 URL 中 path 部分类似 thread-xxx-x-x。此时 path 中的第二个数字为标识页码的参数，如 thread-xxx-1-x 为第一页；thread-xxx-2-x 为第二页；

（4）仅有主帖页，此时不区分第一页和其他页；

（5）失效 URL：因为各种原因，URL 已失效。

基于上述分析，除却异常论坛，对剩余 65 个论坛的 URL 格式进行统计，得出页码在 URL 中位置统计图，如下：

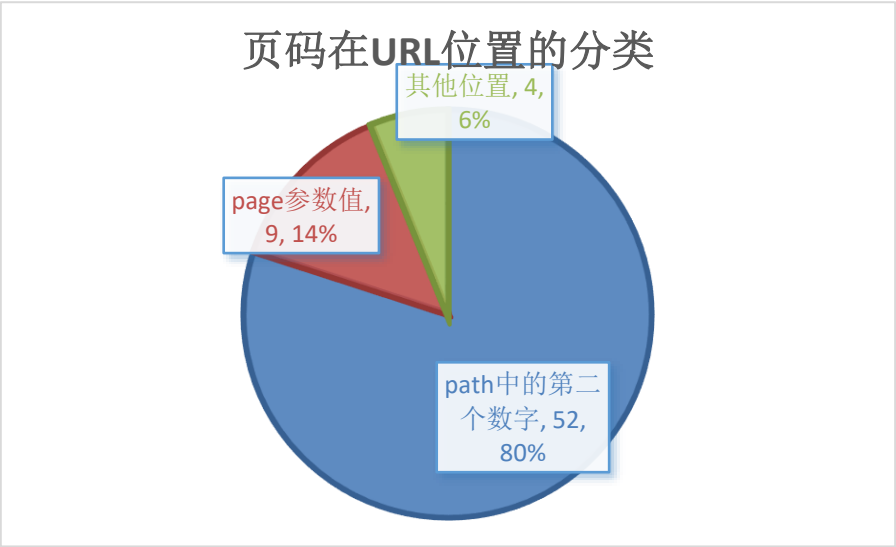

图 12 页码在字段 URL 中位置统计图

由上表可知，在 URL 中没有参数的 56 个论坛中，其中 52 个 path 部分出现的第二个数字为页码数。对于不属于以上两种类型的 URL，通过逐个排查，发现均为第一页，则可认定此类 URL 所指向的页面均为第一页。

综合以上统计分析可知，由 CSS 样式选择器提取内容的方案可行，可以进一步进行规律提取，并开始形成初步选择器汇总。

3.2 方案形成

3.2.1 选择器规律汇总

样本 1：题目中给出的 177 条 URL；

样本 2：将 Top300 的论坛每 10 个分为一组，采用数据挖掘中 10 折交叉验证的思想，每组前 9 个样本作为训练数据集，第 10 个样本作为测试数据集，由训练数据集构建模型，汇总出以下选择器，测试数据集用于结果测试。

选择器所有项均按照该项所表示内容的范围由小到大进行排序（div.a< .a），即精确程度由大到小的顺序排序。

以正则表达式为例：

在一已定字符串中获取与正则表达式匹配的子串，一般情况下，正则表达式规则越复杂，其所匹配的数据越精确，同时所能匹配到的数据的数量越少。

设定一字符串为“发表于 2016-10-14 20:14:23”，在精确匹配的情况下，可以获取出精确时间“2016-10-14 20:14:23”，在非精确匹配情况下可以获取多种时间形式。

通过上述方式设定项的顺序能够有效的减少代码数量，同时增加程序运行效率。其中时间一项是以正则表达式形式汇总，部分选择器数据如下：

表 7 选择器汇总表

类别	汇总
标题（title）	#subject_thread .subject_thread h1 .title h2 ……
作者（author）	.personinfo-name a .num a[target="_blank"] .authi a[target="_blank"]:first-child .user_name a.mingzi:nth-last-child(2) .user_name a[target="_blank"] ……
时间（publish_date）	([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[\s]{0,}(.*\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[\s]{0,}(.*\s{1,})[<>] >[\s]{0,}发表于[\s]{0,}([\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[：:]{0,}[\s]{0,}([\d]{4}-[\d]{1,2}-[\d]{1,2}) ……
内容（content）	.bbs-content .article-detail .topic-content #topic-content #zwconbody ……

分析：由上表汇总出每项内容所对应的选择器统计，由上至下为样本容量逐步扩大时各项内容选择器的增加情况及其所占该项内容全部选择器的比重变化（此处样本容量为对应范围内的训练数据集）：

表 8 各项内容选择器数量及比重变化表

	标题	作者	时间	内容
Top 0-100 （比重）	6 （85.71%）	12 （57.14%）	8 （61.53%）	31 （67.39%）
Top 0-300 （比重）	7 （100%）	18 （85.71%）	11 （84.61%）	38 （92.68%）
Top300+样本 （比重）	7 （100%）	21 （100%）	13 （100%）	46 （100%）
合计	7	21	13	46

由上表绘制选择器增长曲线（注：横坐标 1 表示 Top 0-100；横坐标 2 表示 Top 0-300， 横坐标 3 表示 Top300+样本）

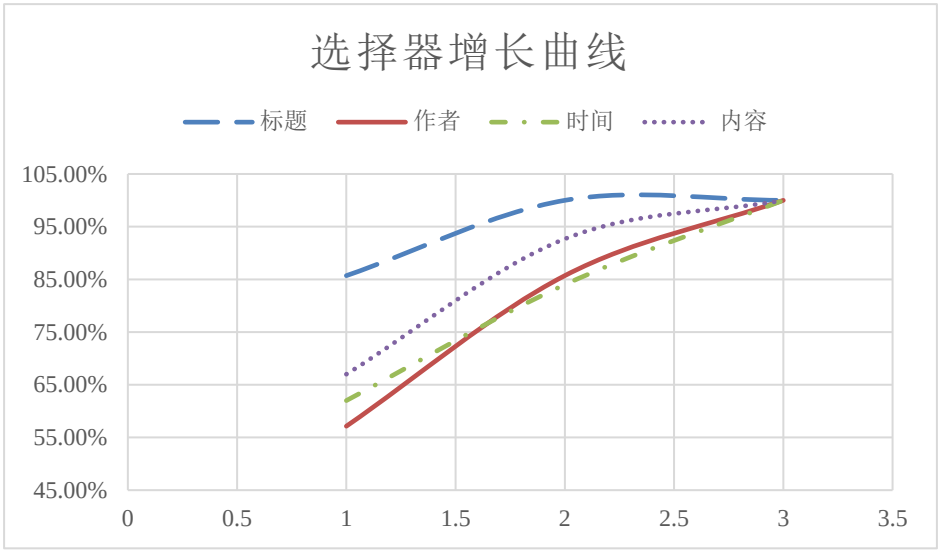

图 13 选择器增长曲线图

由上图和表可知，四项内容所提取的选择器增长速度即斜率逐渐减小，随着样本容量的增加，选择器并没有相应的增加，即当样本容量增大到一定程度时，无需再添加新的选择器，也能提取出相应内容。

3.2.2 主回帖判断

判断 URL 属于有 page 参数值类型还是无 page 参数值类型。

（1）具有*page*参数

1）取出对应键值对，例：

URL：http://bbs.yaolan.com/forum.php?mod=viewthread&tid=53256535&page=1

取得结果为：mod=viewthread

tid=53256535

page=1

2）循环遍历取得的键值对，根据 page 所对应的值判断此 URL 所指示的页面是第一页还是其他页：若 page=1，则为第一页；若 page!=1，则为其他页。

3）若 URL 中有参数但没有形如*page*的参数，则如上文**主回帖判断分析**中所述，一般情况下为第一页。

（2）无参数类型

1）取出 URI， 例：URL：http://bbs.rednet.cn/thread-46025684-1-1.html，取

得 URI：/thread-46025684-1-1.html

2）根据正则表达式 "[\\d]+"进行匹配 URI 中的数字，得出第二个数字为页码（此处已进行规律统计，准确性详见 **3.1.3**），此处页码为 1，即为第一页，非 1 即为其他页。

3）不符合上述两种情况的 URL 一般为包含主帖的页面，此结论解释详见 **3.1.3**。

3.2.3 初步方案

根据上述研究，制定以下方案：

（1）依据标签规律，由 CSS 选择器或正则表达式提取相应内容：title、author、tpublish_date、content；

（2）总结 URL 规律，将主帖与回帖 URL 分开，包含主帖页面的第一个帖子即主帖，其余均为回帖，仅包含回帖页面的所有帖子均为回帖；

（3）测试，修改。

3.3 后期测试

已知测试样本为 3.2.1 中得出的测试数据集，依据数据挖掘中交叉验证的思想，设计了一种方法来对此程序进行测试，即若获取内容同时符合以下两个条件：

（1）获取数量author = content = publish_date

（2）获取数量title = 1 || title = content

此判断条件具体原因详见 4.1.2（4）。

则该 URL 内容已查全，继而判断获取到的内容是否正确，若内容无误，则为正确结果，如图：

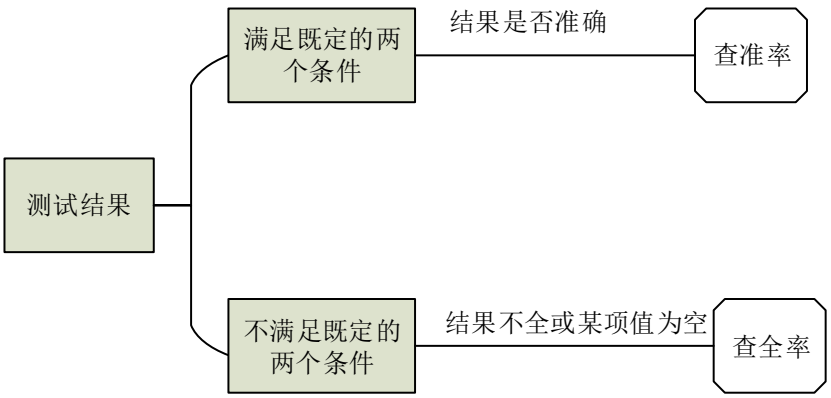

图 14 测试结果分类图

测试结果可按下表分类[18, 19]：

表 9 测试结果分类表

	结果正确	结果不正确
检索到	A	B
未检索到	C	D

A：检索到完整结果且完全正确

B：检索到完整结果但结果并不完全正确

C：未检索到完整结果且这部分属于正确结果

D：未检索到完整结果且这部分属于不正确结果

查准率 Precision 表示检索到的结果准确度，查准率越大表示查询结果越准确。

查全率 Recall 表示检索到的结果完整度，查全率越大表示查询结果越完整。

3.3.1 查准率测试

$$\text{Precision} = \frac{A}{A + B}$$

3.3.2 查全率测试

$$\text{Recall} = \frac{A}{A + B + C}$$

3.4 特殊论坛提取框架

经测试发现有些网址因与通常规则不符，需编写特殊规则，这些特例所占比重为 8.67%。

故而设计了一种简单框架，用于添加此种特殊规则，以使程序具有普适性与简便性。在 Java 语言中，通过使用 Annotation 注解与反射相结合的方式来实现这种规则，而在 Python 和其它语言中，也有类似方法[20]。

类似于 Spring 3.0 @RequestMapping 和@Controller 注解，兼顾到本程序的特殊性，设计了如下轻量级框架：

3.4.1 容器的初始化流程

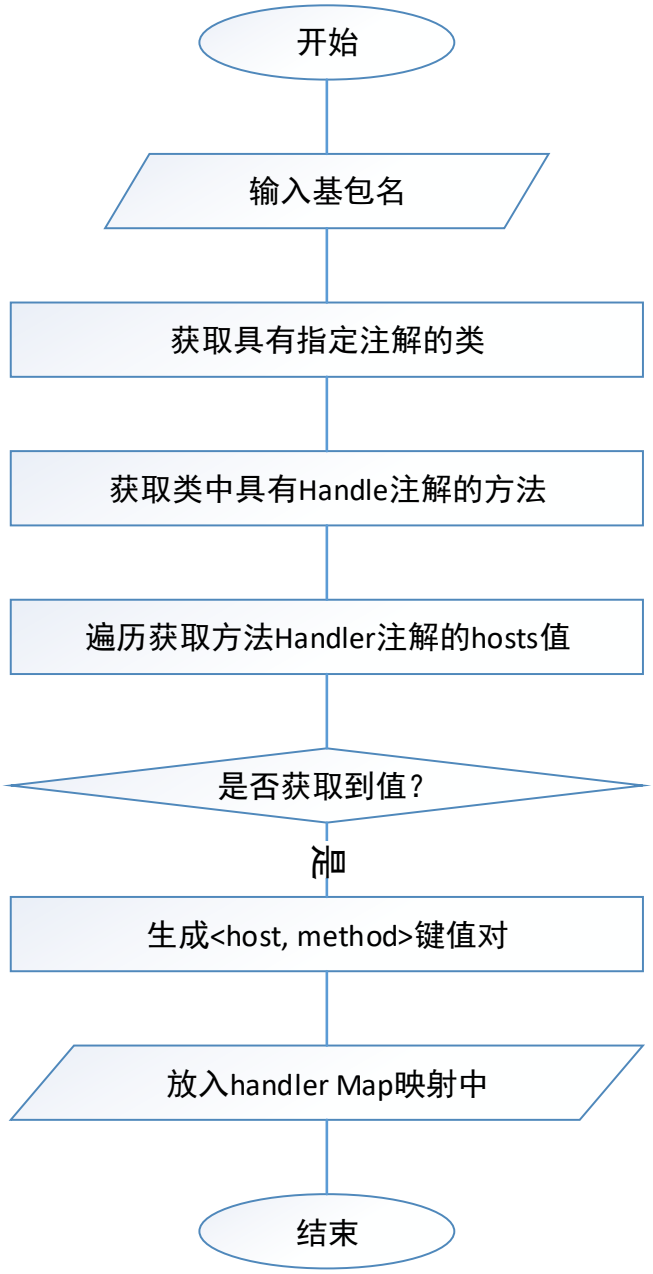

图 15 容器初始化流程图

3.4.2 框架的使用流程

（1）在类上使用 Handler 注解用以标识此 class 中具有处理特殊网站的方法；

（2）在方法上使用 Handler 注解用以标识该方法用于处理特定网站，其中 Handler 注解中的 hosts 值为一个字符串数组，其中包含了一系列可以使用该方法的网站主机名；

（3）在方法中实现具体的处理逻辑，从而获取到 URL 所对应的 Post 对象。

框架的使用示例代码：

```java
package hhbs.mis.tdb.handler;

import hhbs.mis.tdb.annotation.Handler;
import hhbs.mis.tdb.model.Post;

@Handler
public class TdbHandler {

    /**
     * 适用于不符合通常规律的 bbs.360.cn 的处理方法
     * @param url
     * @return
     */
    @Handler(hosts = "bbs.360.cn")
    public static Post bbs360cn(String url) {
        Post post = new Post();
        // 逻辑代码
        return post;
    }
}
```

四、实验结果

4.1 实验结论

4.1.1 核心成果

（1）程序流程图如下：

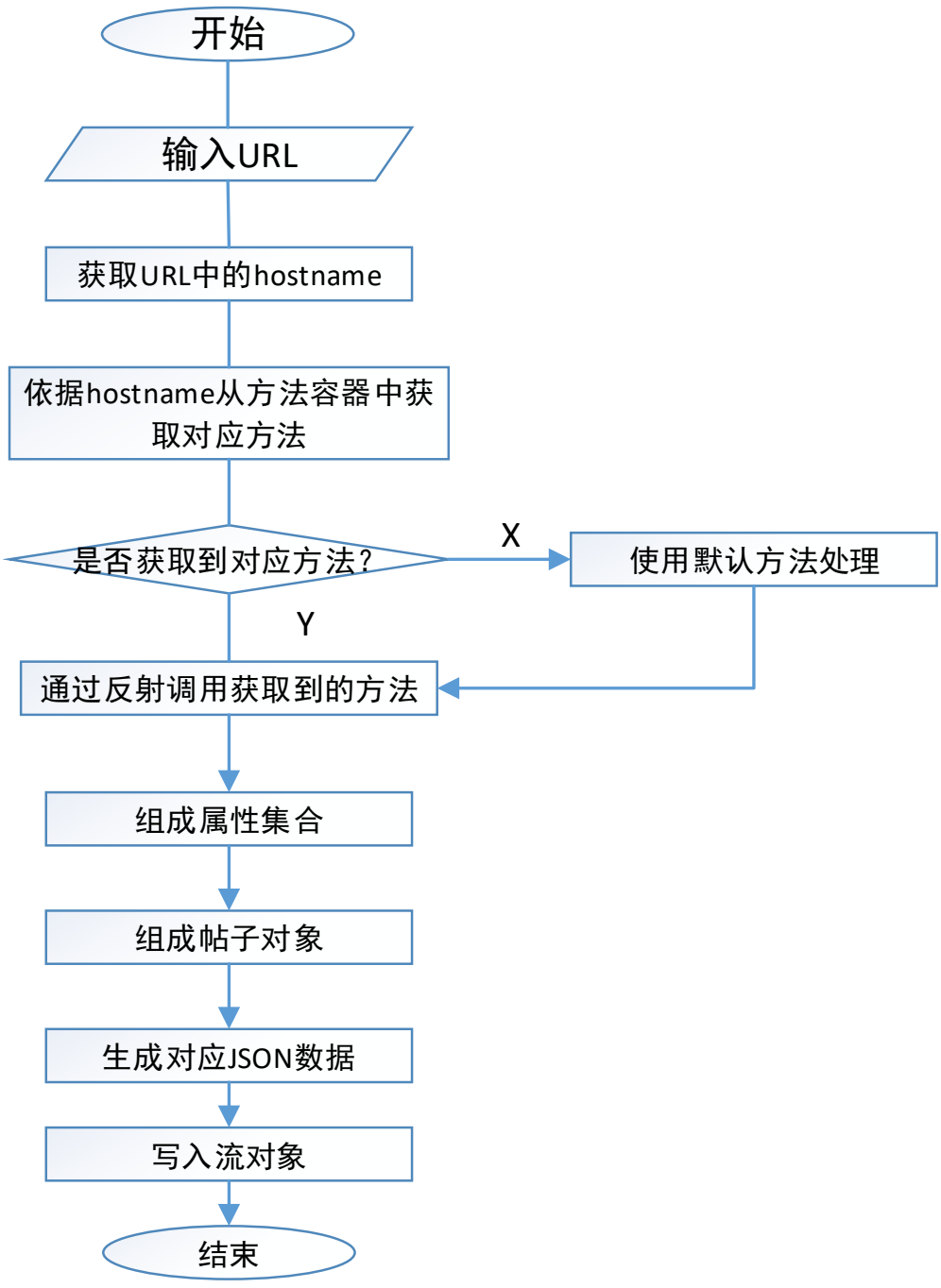

图 16 程序运行流程图

经过上述准备过程，得到如图 17 所示的文件列表，其中文件内容为 CSS 选择器或者正则表达式（图 18）：

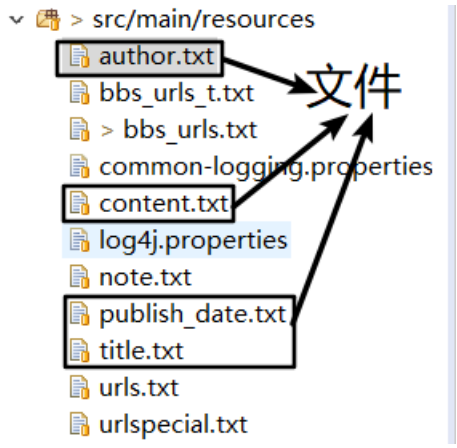

图 17 文件列表

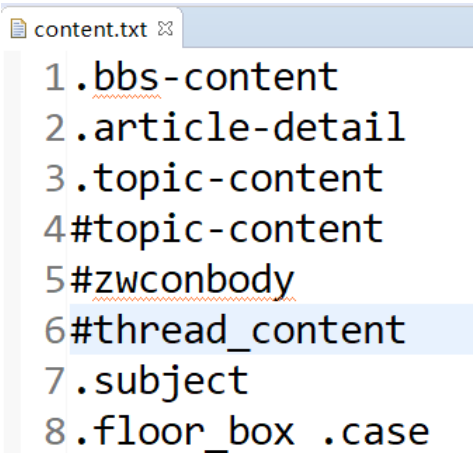

图 18 文件内容示例

（2）程序语言及相关工具库的选择

1）语言：通用论坛爬虫具有一定的商业价值，同时具有广阔的发展前景。为保证程序的运行效率，程序应当易于分布式部署，结合相关要求，选定使用原生分布式语言 Java 进行程序编写。

2)jsoup 作为全球知名的 HTML 轻量级解析库，其中包含大量的针对 HTML 文档的简单易用的 API，同时支持原生 CSS 选择器对 HTML 标签进行选择。

综上，使用 Java 语言及 jsoup 开源解析库进行程序编写。

4.1.2 程序运行

（1）初始化：

程序初始化除 3.4.1 所述的**方法容器的初始化流程**外，还包括文件内容的加载，将文件中 CSS 选择器内容以字符串形式加载到如下对应的 List 数组中，具体初始化流程如下代码所示：

```
1.  private static List<String> author;
2.  private static List<String> content;
3.  private static List<Pattern> publish_date = new ArrayList<>();
4.  private static List<String> title;
5.  private static List<String> urlspecial;
6.  private static Map<String, Method> handlers;
```

```
7.  static {  //  初始化过程
8.      author = getSource("author.txt");
9.      content = getSource("content.txt");
10.     title = getSource("title.txt");
11.     urlspecial = getSource("urlspecial.txt");
12.     List<String> source = getSource("publish_date.txt");
13.     for (String s : source)
14.         publish_date.add(Pattern.compile(s));
15.     handlers = new HashMap<>();
16. }
17. public static List<String> getSource(String name) {// 获取文件资源
18.     InputStream inputStream = TdbApplication.class
19.                                             .getClassLoader()
20.                                             .getResourceAsStream(name);
21.     BufferedReader reader = new BufferedReader(
22.                                 new InputStreamReader(inputStream));
23.     List<String> list = new ArrayList<>();
24.     String line = null;
25.     try {
26.         while ((line = reader.readLine()) != null)
27.             list.add(line);
28.     } catch (IOException e) {
29.         System.out.println(e.getMessage());
30.     }
31.     return list;
32. }
```

（2）获取网页 HTML 内容：

利用 jsoup 提供的 API，可以方便地依据指定的 URL 从网络上获取 HTML 文档，并生成对应的 Document 对象实例。

（3）获取帖子字段信息：

在获取到的 HTML 文档中，多数情况下都是由多个目标内容（主帖或回帖）组成的，由上文方案 2 的描述可知，单个 CSS 选择器可以选择出一个 HTML 文档中所有目标内容的同一属性所在的标签（例如一个页面中所有主帖和回帖的作者所在的标签）。

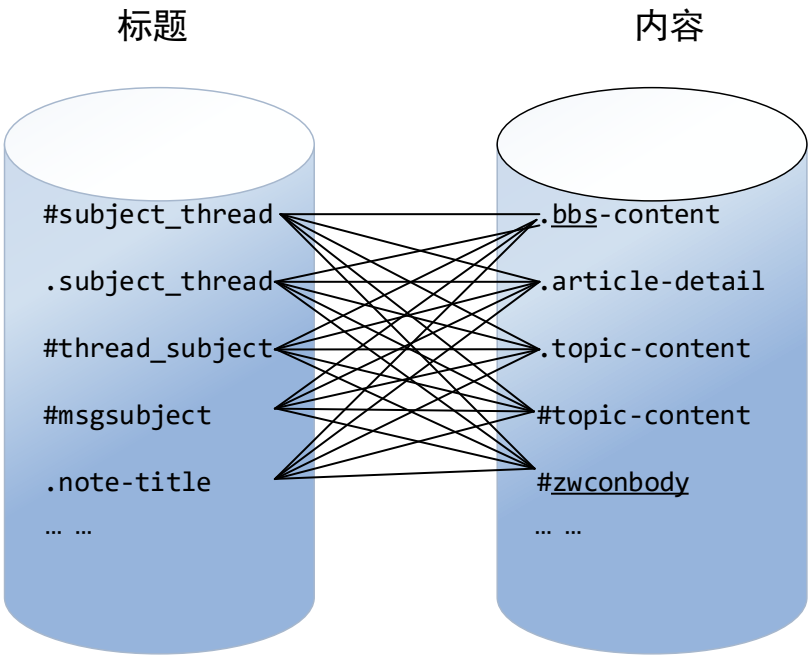

图 19 标题和内容的随机组合示意图

假设网站标题和内容所在的标签的属性是在如图 19 范围内随机出现的。在使用 CSS 选择器进行选择的过程中，可能随机使用到标题和内容所对应的选择器集合中的任意一个元素。即对一已确定论坛，图 19 中标题和内容之间的连线是唯一且确定的。

扩大到论坛网站目标内容的所有属性，一确定网站的目标内容选择器元组（一网站帖子的标题、内容、作者、时间所对应的选择器所构成的元组）是全范围选择器集合（多个网站帖子的标题、内容、作者、时间所对应的集合）的笛卡尔积中的唯一元组，即在图 13 所示的随机组合连线中，只有图 20 中的唯一一条连线与网站对应。

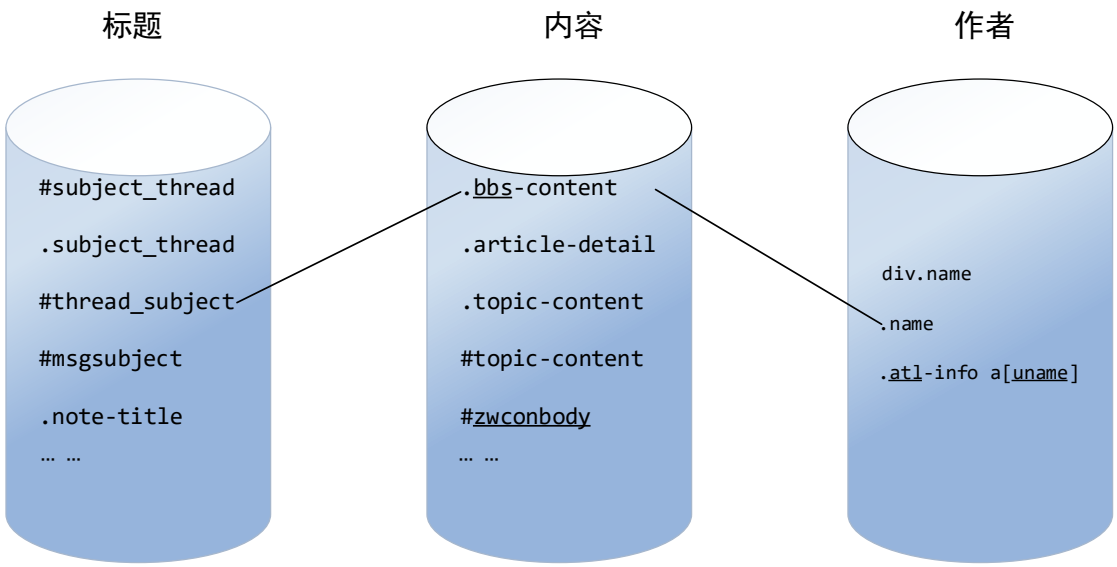

图 20 全范围选择器集合的笛卡尔积中的唯一元组连线示意图

在程序的运行过程中，依据获取到的 HTML 页面，遍历内容（正文）选择器集合，若选择到相应的内容，则停止遍历，并进入其他内容选择器（如作者）的遍历过程，直至结束。依据 3.2.1 选择器规律汇总 中所述：选择器的顺序按照所表示内容的范围由小到大进行排序，即精确程度由大到小的顺序排序。假设页面的选择器组合是随机的，则每个循环的复杂度为：

$$\frac{\sum_{i=1}^{n} n}{n} \times 4 \sim O(n)$$

在程序的实际运行过程中，选择器出现在集合中前半部分的概率远大于出现在后半部分，也因此程序的复杂度远小于平均复杂度$O(n)$。

（4）下面以作者和内容正文为例详细说明属性的获取流程：

1）遍历内容正文的 CSS 选择器，直至选择到内容，并构成如下所示的 List 集合：

```
1.  [ac, bc, cc, dc]
```

内容提取的具体代码如下所示：

```
1.  public static List<String> getContent(Element doc) {// 获取主贴和回帖内容
2.      doc = Jsoup.parse(doc.html()).body();
3.      List<String> res = new ArrayList<>();
4.      for (String con : content) {
```

```
        Elements elements = doc.select(con);
        for (Element e : elements) {
            if (e.hasText())
                res.add(e.text());
            else
                res.add(e.html().replaceAll("\n", ""));
            e.remove();
        }
        if (res.size() >= 1)
            break;
    }
    return res;
}
```

2）遍历作者的 CSS 选择器，直至选择到内容，并构成如下所示的 List 集合：

```
[aa, ba, ca, da]
```

其具体代码与内容提取类似。

3）依照**方案二核心思想**中的内容，同一个目标内容所处的位置应当在同一个 HTML 标签下，即整个 HTML 构成如下的树状图：

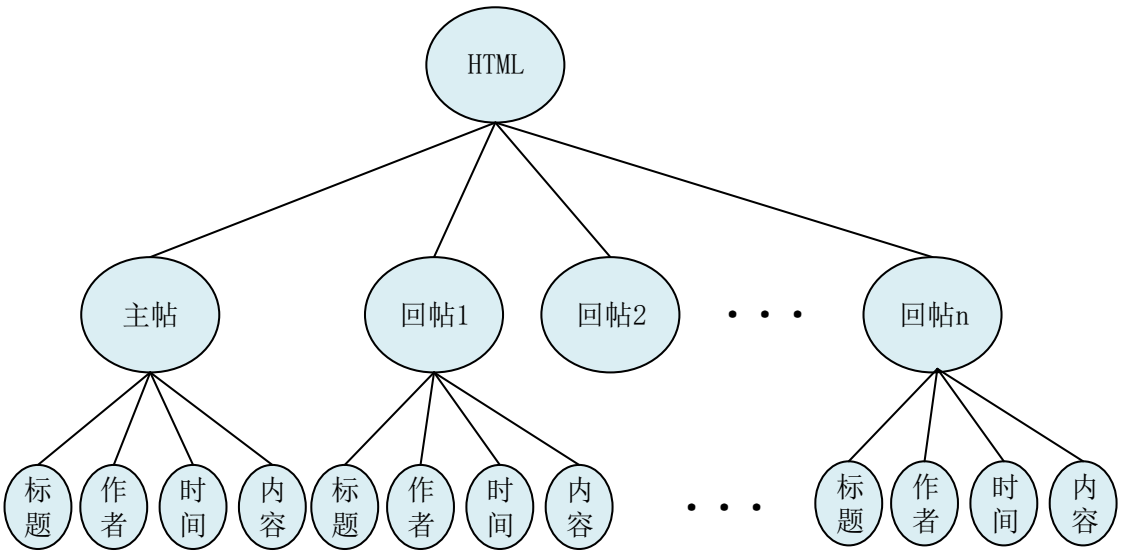

图 21　HTML DOM 树和目标内容的对应关系示例

因此，在对 HTML DOM 树进行深度优先遍历的过程中，CSS 选择器选择的标签所构成的 List 集合是具有顺序的[21]。即内容 List 中元素与作者 List 中元素依照顺序一一对应（**3.3 后期测试**的条件依赖于此对应规则）：

```
1. 帖子：[贴 1，贴 1，贴 3，贴 4]
2. 正文：[ ac，  bc，  cc，  dc]
3. 作者：[ aa，  ba，  ca，  da]
```

依照属性集合组合帖子对象：

```
1. 贴 1 {"content":"ac","author":"aa"}
2. 贴 2 {"content":"bc","author":"ba"}
3. 贴 3 {"content":"cc","author":"ca"}
4. 贴 4 {"content":"dc","author":"da"}
```

提取过程完成。

4.2 测试结果

4.2.1 样本测试结果

从 177 条 URL 排除异常 URL，得到可用 URL 数量为 150。运行程序，得到部分结果如下图所示：

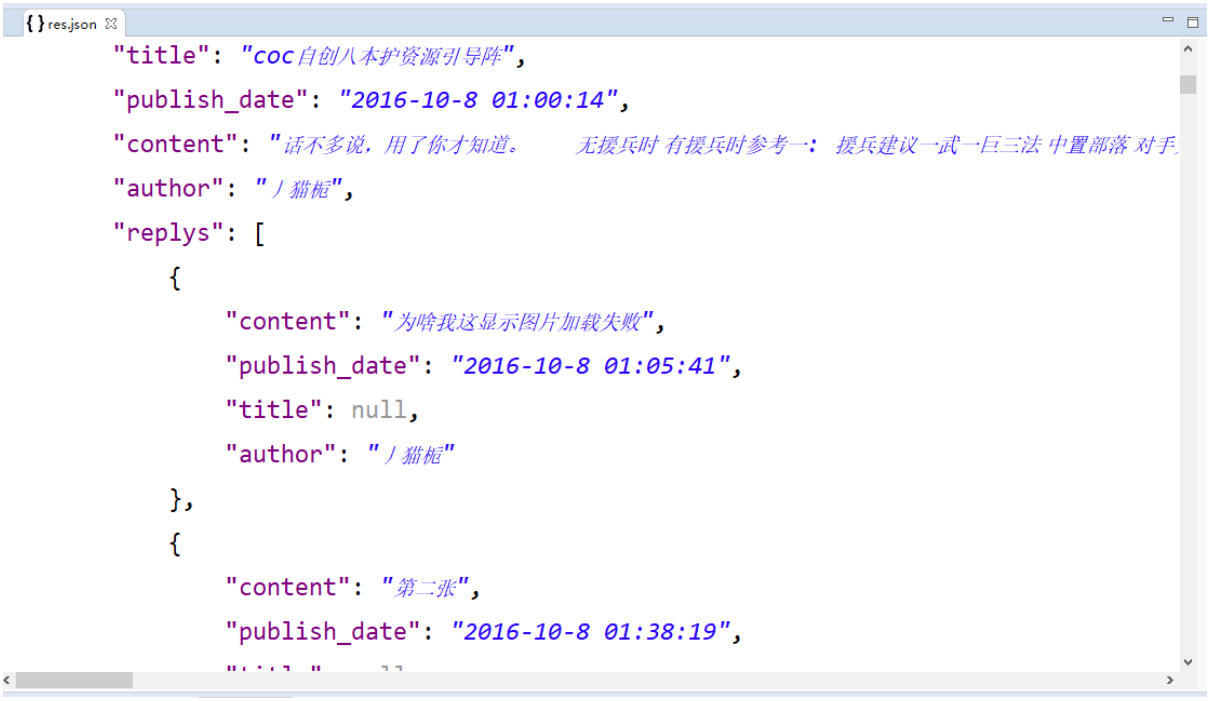

图 22 测试结果截图 1

图 23 测试结果截图 2

通过测试，其中 11 条 URL 结果有不同程度的缺失，2 条 URL 结果不准确。具体情况如下：

表 10 样例 URL 情况统计表

主帖内容缺失	5
主帖时间缺失	2
主帖时间缺失且主帖、回帖作者缺失	1
主帖标题、时间缺失且主帖、回帖作者缺失	1
主帖作者、时间缺失且回帖内容缺失	1
所有要素均缺失	1

查全率：

$$a = \frac{150 - 11}{150} \times 100\% = 92.66\%$$

查准率：

$$a = \frac{150 - 11 - 2}{150 - 11} \times 100\% = 98.56\%$$

4.2.2 论坛排行榜测试结果

汇总 Top 0-300 的测试数据集 30 条为样本容量，得出结果有缺失值得数目为

2 条，结果获取不准确的有 1 条。

查全率：

$$b = \frac{30-2}{30} \times 100\% = 93.33\%$$

查准率：

$$b = \frac{30-2-1}{30-2} \times 100\% = 96.43\%$$

由测试结果可知，无论实验样本中还是论坛排行榜中，查准率与查全率均在90%以上，而该程序的正确率为 91.33%，明此程序的普适性在 91.33%以上，对于不能普适的8.67%论坛，设计了一种简单框架以填充特殊规则来处理，详见3.4。

五、实验感想

在本次实验的过程中，可以发现许多问题与规律，主要可以概括为以下几点：

（1）即使论坛的格式变化无穷，但其主要内容如标题、作者等所应用的 CSS 选择器始终在一个集合内，其极大的相似性与概括性足可以普适当今网络上 91.33%以上的论坛，这种更为精确的爬虫方法有助于提取更为精确的内容。

（2）由于网络上信息瞬息万变，许多原来存在的内容或许已经更改或消失，体现在本报告中提到的异常论坛，这些异常或是由于该帖被删除，或是网站域名已修改，或是网站关闭等等，而对于这类信息，获取意义已然不大。

（3）规则具有局限性，每种规则都有不适用的特例存在，所谓普适性十指比较普遍的适用于同类对象或事物。由于这些特例的模式千变万化，不能仅仅使用一种或几种规则来提取，故而本实验单独设计了一种框架，用来填写与之相对应的提取规则。

当今时代的竞争是信息的竞争，随着国家信息化的全面推进，信息的价值更加凸显，同时，信息更能作为企业的一种战略资源，从中获得更大的商机。网络爬虫也非想象中那么简单，更是有通用网络爬虫、聚焦网络爬虫、增量式网络爬虫和深层网络爬虫等多种形式，通过看视频、有关书籍学习这些技术，自学能力得到了提高、能力得到了锻炼。

由于信息的可共享性和转换性，使得信息可以快速传播并广泛应用，并转换为知识、金钱或社会地位。当然，这些都是建立在有价值的信息的前提下，本次网络论坛的爬取便是一个很好的佐证。互联网上涌现了愈来愈多具有较高信息聚合度的网络论坛，但这些论坛都是独立的个体，分散在互联网的各个角落，只有利用网络爬虫技术，将这些特定主体、相似内容的论坛信息综合在一起，才能挖掘有价值的信息。其实，网络爬虫只不过是数据分析的准备阶段罢了，为后续的数据挖掘提供信息集合，但是这也是不可或缺的前提。因此，也是本实验花费大量时间和精力的意义所在。

一个人的思路和视野永远是有局限性的，与个人背景、阅历经验紧密相关，“三个臭皮匠赛过诸葛亮”。有些问题一个人思考时往往会陷入死胡同，而且走不

出来。但是当小组在一起讨论时，利用集体的智慧，你一言我一语，局面豁然开朗，甚至可以获得意料之外的收获。经过团队的分工协作、互帮互助才获得今天的成果，当今社会是团队协作的社会，是需要集思广益、优势互补的时代。

致谢

借此报告完成之际，首先感谢我们小组的导师。本次实验的成果离不开导师耐心的指导和监督，感谢导师的帮助。

感谢 jsoup 开源组织，正是 jsoup 开发人员的无私奉献，本次实验才得以顺利完成。

同时感谢大赛组委会，给予我们这次前进创新的机会和展示的平台，我们将会持续关注网络爬虫方面的技术研究，争取下一次做得更好。

最后感谢小组的每一位成员，是我们努力的付出换回了今日的成果。

参考文献

[1] 毕楠, 银成钺, 蓝海平. 信息价值对消费者影响的实验研究——以微信信息为例[J]. 情报科学, 2015(04):93-97.

[2] 张海涛, 靖继鹏. 信息价值链:内涵、模型、增值机制与策略[J]. 情报理论与实践, 2009(03):16-18.

[3] 何绍华, 康斌. 信息价值和信息服务价值评价研究[J]. 图书情报工作, 2005(05):72-75.

[4] 周德懋, 李舟军. 高性能网络爬虫:研究综述[J]. 计算机科学, 2009(08):26-29.

[5] Web crawling spies hunt corporate pirates[J]. Network Security, 2003,2003(7):1-2.

[6] 李田. 面向论坛便捷化获取信息爬虫程序分析: 2015年6月建筑科技与管理学术交流会, 中国北京, 2015[C].

[7] MUFTI S. Web search software hacks into secretive online forums[J]. New Scientist, 2010,206(2755):20.

[8] 唐勇. 网络论坛爬虫的设计[J]. 电脑知识与技术, 2012(03):570-572.

[9] 于成龙, 于洪波. 网络爬虫技术研究[J]. 东莞理工学院学报, 2011(03):25-29.

[10] 于娟, 刘强. 主题网络爬虫研究综述[J]. 计算机工程与科学, 2015(02):231-237.

[11] BHARAT K, BRODER A. Mirror, mirror on the Web: a study of host pairs with replicated content[J]. Computer Networks, 1999,31(11–16):1579-1590.

[12] DIKAIAKOS M D, STASSOPOULOU A, PAPAGEORGIOU L. An investigation of web crawler behavior: characterization and metrics[J]. Computer Communications, 2005,28(8):880-897.

[13] CHAKRABARTI S, van den BERG M, DOM B. Focused crawling: a new approach to topic-specific Web resource discovery[J]. Computer Networks, 1999,31(11–16):1623-1640.

[14] LIU H, JANSSEN J, MILIOS E. Using HMM to learn user browsing patterns for focused Web crawling[J]. Data & Knowledge Engineering, 2006,59(2):270-291.

[15] 曹志杰. 面向互联网机票数据抓取与票价预警系统的设计与实现[D]. 电子科技大学, 2014.

[16] 丁星. 基于文本倾向性分析技术的微博监控系统[D]. 江苏科技大学, 2015.

[17] 李红霞. 基于Web的比较观点挖掘方法研究[D]. 山西大学, 2011.

[18] 沈建人. 查准率和查全率之间的关系[J]. 情报探索, 2006(04):32-34.

[19] 余丹. 关于查全率和查准率的新认识[J]. 西南民族大学学报(人文社科版), 2009(02):283-285.

[20] AHMADI-ABKENARI F, SELAMAT A. An architecture for a focused trend parallel Web crawler with the application of clickstream analysis[J]. Information Sciences, 2012,184(1):266-281.

[21] 潘婷婷. 基于List集合的多条件查询优化算法研究[J]. 湖南邮电职业技术学院学报, 2014(01):25-28.

附录 1：选择器汇总表（完整）

类别	汇总
标题（title）	c:#subject_thread c:.subject_thread c:h1 c:.title c:h2 c:h3 c:h4
作者（author）	.uName a[target="_blank"] .fsu-info .name,.floor-top-nolou>span>a[target] .num a[target="_blank"] .lt_oneboxL_avatar a .plc>.dfsj_postt a[target]:first-child .pls .authi a[target]:first-child .authi a[target="_blank"]:first-child .zhanzhuai_authi a,.authi a[target="_blank"] .user_name a.mingzi .user_name a[target="_blank"] .user_center_name a[target="_blank"] .lt_oneboxL_avatar a[target="_blank"] .name a:first-child,.tit a.auor .name a[name="users"] .author a.u[target] .articleInfo a.userNick .author a[target="_blank"] li.txtcenter a[target="_blank"] .sk_authi a[target="_blank"]:first-child

	.t_user a[target="_blank"] .t_user a.bold .name a:first-child,.tit a.auor .name a[target="_blank"] div.name .name .atl-info a[uname] .postinfo > a[target="_blank"] .ftrmod a[target="_blank"]:first-child .readName a .article-author p .avatar+p .tit_user a[target="_blank"]:first-child .replyfloor-info a[target] .postauthor cite a
时间 （publish_date）	>[\s]{0,}发表于[\s]{0,}([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[：:]{0,}[\s]{0,}([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[\s]{0,}([\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[：:]{0,}[\s]{0,}([\d]{4}-[\d]{1,2}-[\d]{1,2}) >[\s]{0,}发表于：[\s]{0,}\[([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2})\] 时间：[\s]{0,}([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}:[\d]{1,2}) ([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}:[\d]{1,2}) ([\d]{2}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}:[\d]{1,2}) ([\d]{2}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) ([\d]{4}/[\d]{1,2}/[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}:[\d]{1,2}) ([\d]{4}-[\d]{1,2}-[\d]{1,2}\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[\s]{0,}(.*\s{1,}[\d]{1,2}:[\d]{1,2}) >[\s]{0,}发表于[\s]{0,}(.*\s{1,})[<>]
内容（content）	.bbs-content

	.article-detail
	.topic-content
	#topic-content
	#zwconbody
	#thread_content
	.subject
	.floor_box .case
	.quote-content
	.conttxt,.x-reply
	.posts-cont
	.invitation_content
	.pcb
	.cont_zhengwen
	.plc
	.pct
	.news_show_nr
	.tpc_content
	#Cnt-Main-Article
	.artibody
	.post-content
	.post_content
	#msgMainContent,.fay
	.layer_c .text
	.content
	.article
	.mc
	#postcontent0
	.post_body
	.postmessage

	.t_msgfont #topic .entry-content .post_main .viewmessage #article_content .sa-box .thread-cont .l-content .post_width .con .txtmain .cont .topiccontent,.replycontent .topic-item .Amain_main, .p1_txt .list-item .t_msgfont1

注：此表内容更新截止到2017年4月14日