基于链接和页面分析的主题爬虫算法在个性化信息推送系统的应用研究

第一章 绪论

1.1 背景

1.2 研究现状

1.3 研究内容

1.4 本章小结

第二章 相关技术综述

2.1 通用爬虫简介

2.2 主题爬虫简介

2.3 消息队列技术综述

2.4 个性化新闻推送系统综述

2.5 本章小结

第三章 爬虫爬行策略研究与分析

3.1 传统主题爬虫爬行策略

3.2 主题爬虫爬行策略

3.3 策略对比

3.4 本章小结

第四章 爬虫爬行策略中主题相关度算法的改进

4.1 Page Rank算法

4.2 Fish Search算法

4.3 改进的Fish Search算法

4.4 Page Rank结合改进的Fish Search算法

4.5 本章小结

第五章 实验结果及分析

5.1 实验目的

5.2 实验环境及数据

5.3 评价指标

5.4 实验设计

5.5 实验结果及分析

5.6 本章小结

第六章 个性化新闻推送系统“及时推”的实现

6.1 需求分析

6.2 系统概述

6.3 系统相关模块实现

6.4 系统相关模块运行效果

6.5 本章小结

第七章 总结

第一章 绪论

1.1 背景

1998年5月，联合国新闻委员会正式提出：网络是继报纸、广播和电视三大传统媒体之后的第四媒体【1】。现代网络媒体还包括网络报纸、网络电视和网络短信等新媒体，是网络媒体通过强大的计算机技术将不同的媒体融合在一起产生的新媒体形态，网络作为一种媒体，已经是一个不争的事实。2015年2月3日，中国互联网信息中心（CNNIC）在京发布第35次《中国互联网网络发展状况统计报告》，报告显示，截至2014年12月，我国网民数量达6.49亿，计算机普及率达47.9%，与2004年比较，网民数量增加了9倍【2】，如图1所示。

随着计算机技术的发展和信息化的日益深入，传统的信息获取方式已不能满足人们的需要。一方面，当今网络上的信息正在以几何级数的速度增长，囊括了政治、经济、文化等多方面的大量的资讯，使得用户可以在越来越丰富和广阔的网络信息中，查询自己所需要的信息；但是，网络上的关键信息没有任何规律和结构特征，人们很难从大量网络信息中准确找到对自己有用的信息。另一方面，准确及时地掌握相关新闻信息的动态是现代企业和个人发展的必要条件之一。现代企业中，企业管理者如果及时地掌握了相关行业的最新、最准确的信息，就能在任何关头做出合适的决策，不错过任何商机；同样的，个人准确及时的获取信息也很重要，个人的成功很大程度上依赖于其信息获取的快慢，有了信息上的准备才能不错过机会，才能成功。

那么如何准确及时地找到对自己有用的信息呢？搜索引擎是目前最高效最常用的查找信息的途径。搜索引擎系统通过网络爬虫、索引、排序等技术将网络上的信息归类，只需要在搜索引擎系统中输入关键词，就能快速找到相关的信息。但是，网络上的信息数量很大，人们感兴趣的只是其中的一小部分，每天花费大量的时间浏览网页，非常浪费时间；并且，现代人的生活节奏越来越快，极易忘记搜索网络信息而错失重要通知或者招商机会。

根据用户的特点，定制其感兴趣的网站或者关键词，每天推送网站更新的网络信息给用户，这种新的信息获取途径被称为个性化信息推送。个性化信息推送的关键在于网络信息的爬取，其实质就是一个用来搜索网络上符合查询主题的信息的搜索引擎，而主题爬虫又是主题搜索引擎的核心，因此是本文的主要研究内容。

1.2 研究现状

1.2.1 主题搜索引擎研究现状

（1）搜索引擎

国外的搜索引擎主要有Google、Yahoo、AltaVista【】。如图2所示，其中，Yahoo是最早的提供目录索引的搜索引擎，是第一代搜索引擎的代表，于1994年4月发布，在很长一段时间内是搜索引擎的领跑者。直到1998年10月Google的出现，Google采用PageRank算法对所有网页进行排序，根据网页的重要程度返回搜索结果，已经证明是非常有效的搜索引擎。目前，Google已经有40多种语言版本，可以根据IP地址自动切换语言，Google已经被公认是全球规模最大的搜索引擎。AltaVista虽然没有Google那么多用户数量，但是它的搜索结果更丰富，搜索范围更大，就连一些鲜为人知的偏僻站点也能访问到，AltaVista也被公认为最好用的搜索引擎之一。

国内的搜索引擎主要有百度、SOSO搜索、网易搜索等。如图3所示百度公司于2001年1月成立，于2001年10月发布百度搜索引擎，并于2005年在纳斯达克上市。百度搜索引擎主要针对中文搜索，经过多年的发展，其已成为国内搜索引擎的领跑者，中国市场份额超过Google。SOSO搜索是腾讯公司2006年推出的搜索引擎，依托于腾讯用户群和平台，有着广阔的前景。网易搜索是网页公司于2007年推出的搜索引擎，也在不断的发展中。

随着信息的多元化增长，就连Google和Baidu等功能强大的搜索引擎也难以满足特定领域和主题的信息检索，这是需要一个分类更精确、数据更专业、更新更及时的搜索引擎，于是面向主题的搜索引擎诞生了，并成为当今社会的研究热点【】【】。

（2）主题搜索引擎

主题搜索引擎不同于Google、Baidu等通用搜索引擎，使用主题搜索引擎的用户都是需要了解某一特定领域的信息，并且都是相关行业的专业人士。近些年来，主题搜索引擎发展很快。

在国外，代表性的主题搜索引擎有Scirus系统、Medical Matrix系统、LIBClient-IRISWeb系统，如图4所示。Scirus系统主要用于检索科学信息，于2001年被《搜索引擎观察》评为最佳专业搜索引擎，是目前全球最全面的科技论文网站之一。Medical Matrix系统主要用于检索医学行业信息，为医药行业人员提供医药类关键词的检索，是目前全球最具权威的医学行业主题搜索引擎。LIBClient-IRISWeb系统由North Carolia大学的计算机科学系和法学系共同开发的搜索引擎，主要用于检索法律信息。

在国内，也出现了一些非常成功的主题搜索引擎，代表性有搜房网、搜职网、百度视频搜索等，如图5所示。搜房网拥有非常全面的房产项目信息，主要用于房产项目相关信息搜索，改变了传统房地产的营销模式。搜职网主要用于招聘信息和求职信息的搜索。百度视频搜索拥有大量的中文视频资源，提供视频搜索。

1.2.2 主题爬虫研究现状

（1）网络爬虫

为了满足不同需求，网络爬虫发展到现在形成了很多不同类型的爬虫。大致可分为通用网络爬虫、增量式网络爬虫、主题网络爬虫、基于Agent的网络爬虫和基于元搜索的网络爬虫。

通用网络爬虫是最早出现的爬虫，它的目标是遍历整个互联网，搜集尽可能全面的页面，随着信息的不断膨胀，对整个网络的一次爬行的代价越来越大，出现了性能瓶颈。增量式网络爬虫与通用网络爬虫的区别是只搜集新增加或者发生了变化的页面，大大减少了搜集的页面数，节约了时间开销。主题网络爬虫主要针对用户在特定领域和主题的搜索，它优先采集主题相关性高的页面，丢弃主题相关性低的页面，很好的满足了特定用户对特定领域的需求，成为最近几年的研究热点，也是本文的主要研究内容。Agent是一个能够自主运行的软件，能够自我学习和推理，对外界变化做出反应，并进行调整以适应变化。美国Stanford大学研究了一种基于Agent的主题信息爬取系统【12】，使用TF-IDF算法和向量空间模型给采集的文本排序。元搜索的网络爬虫是元搜索引擎的核心，它将用户的搜索请求提交给不同的大型搜索引擎，并整合返回的搜索结果，返回给用户。美国Binghamton大学深入研究了元搜索引擎技术数据库选择的问题【13】【14】。

（2）主题爬虫

主题爬虫是网络爬虫的一种，其与通用网络爬虫最主要的区别就是爬行策略的不同。通用爬虫一般采用的是深度优先、广度优先和深度和广度优先结合的搜索策略，由于上述策略忽略了网页爬行顺序和重要性的计算，会爬取大量无关的页面，并且效率很低。所以，本文主要的研究内容为主题爬虫的爬行策略和页面重要性的评估方面，对页面的主题相关度算法进行改进。

基于通用爬虫的爬行策略，主题爬虫提出了一种结合已知的信息而自我调整的搜索策略，主要有基于网页内容和基于网页链接结构的搜索策略。Fish Search【6】和Shark Search【7】算法等是基于网页内容的。Fish算法是通过匹配页面出的关键词与主题关键词，使用不是1就是0的离散值来判断页面的主题相关度；Shark Search算法改进了Fish Search离散值表示主题相关度的方法，采用连续值表示页面的主题相关度，即余弦法计算页面关键词和主题关键词的相似度。PageRank和Hits算法是基于网页链接结构的，都是通过分析网页链接之间的关系来确定网页的重要程度。

1.3 研究内容

本文受到《及时推新闻推送》（四川省成果转化项目）项目的支持，该项目主要有三个研究重点：1）基于回复式神经计算模型的研究；2）爬虫子系统的研究；3）推送子系统的研究。本文的研究重点便是爬虫子系统，主要研究内容如下：

1. 针对爬虫的爬行策略关键问题，对传统爬虫和主题爬虫的几种爬行策略进行了分析与对比。
2. 针对主题爬虫爬行策略的主题相关度计算的关键问题，对Fish Search算法进行了分析与研究，提出了一种改进的Fish Search算法。实验的结果显示，改进的算法能够更加准确地计算页面的主题相关度。
3. 针对主题爬虫爬行策略的待爬取链接的优先级选择的问题，结合Page Rank算法和改进后的Fish Search算法，取合适权重综合得到待爬取链接的优先级。实验的结果显示，改进的算法能更加准确的爬取主题相关的页面。
4. 将相关算法应用到《及时推新闻推送》项目的爬虫子系统中，并将爬取信息在多个客户端展示。

1.4 论文的组织结构

本文共分为七个章节，其内容安排如下：

第一章在详细论述了本文的研究背景的基础上，对主题搜索引擎及主题爬虫在全球范围内的学术研究现状进行了描述，进而指出了研究与分析主题爬虫的重要性，引出本文的研究内容。

第二章首先根据搜索引擎的背景，介绍了搜索引擎的工作原理，接着分别介绍了传统搜索引擎和主题搜索引擎的特点，进一步展示出主题爬虫在搜索引擎的作用。然后重点介绍了主题爬虫，通过对比主题爬虫与传统爬虫的基本原理和结构，详细的讨论了主题爬虫的几个关键问题。最后，介绍了相关的技术包括信息推送技术和消息队列技术。

第三章主要分析爬虫的爬行策略关键问题，介绍了常用的几种爬行策略，并对比起优缺点，为后文的算法改进打下基础。

第四章主要是对主题爬虫爬行策略的改进，首先，通过对基于页面内容的Fish Search算法的研究，针对其不足的地方提出了一种改进的Fish Search算法，更加准确地计算页面的主题相关度；然后，结合PageRank算法和改进后的Fish Search算法，综合计算待爬取链接的优先级，能够爬取到更多与主题相关的页面。

第五章属于实验设计和实验结果分析部分，实现了一个通用的爬虫框架，分别对原算法和改进后的算法进行实验，并分析实验结果，说明本文提出的算法是切实有效的。

第六章将本文提出的算法运用到《及时推新闻推送》项目的爬虫子系统中，并简单介绍了系统的架构，最后展示系统相关模块的实现效果。

第七章是对全文工作的总结以及对未来工作的展望。

第二章 相关技术综述

2.1 主题搜索引擎技术

搜索引擎（Search Engine）【8】【9】是指根据一定的策略，运用特定的程序在网络上搜集大量信息，在对信息处理和组织后，为用户提供搜索服务，并将搜索结果返回给用户的系统。

2.1.1 搜索引擎的背景

在网络诞生的初期，由于网络上的信息很少，查找比较容易。但是，随着互联网的发展，网络上的信息量越来越大，信息查找变得越来越难，这样，为了满足用户检索信息的搜索引擎便出现了。

现代搜索引擎的起源是McGill University三名学生（Alan Emtage、Peter Deutsch、Bill Wheelan）发明的[Archie](http://baike.baidu.com/view/566277.htm)【10】。[Archie](http://baike.baidu.com/view/566277.htm)虽然是一个基于文件名查找的系统，搜索的不是网页，但是它已具有搜索引擎的基本特征，即通过计算机程序自动搜索信息，然后对其进行处理和组织后，提供检索服务。

一直到九十年代中期，随着万维网（WWW）的出现，搜索引擎得到进一步的发展。出现了以Yahoo【10】为代表的提供分类服务的目录式搜索引擎，可以支持简单的搜索。到现在为止，一批使用先进计算机技术的大型搜索引擎，如Google、Bing和Baidu【10】等，已成为互联网上搜索信息的必不可少的一部分。

总之，随着技术的不断提高，搜索引擎会向专业化、个性化和智能化的方向不断发展。

2.1.2 搜索引擎的工作原理

搜索引擎主要由搜索器、检索器、索引器和用户接口等四个部分组成，如图6所示。

1. 搜索器：即爬虫程序，他的作用主要是采集、通过HTTP协议访问网络上的超链接并爬取其对应的页面信息。
2. 索引器：索引器通过分析爬取到的网页，提取网页内容的一些重要信息，根据一些相关度算法，计算得到网页中的所有关键词的相关度，最后将这些相关度信息建立网页的索引数据库。
3. 检索器：检索器根据用户输入的搜索关键词，在索引数据库中查找，并将搜索结果返回给用户。并且检索器根据一定的评估算法，将用户最感兴趣的内容放在前面。
4. 用户界面：搜索引擎的用户界面主要提供搜索关键词的输入接口，和显示搜索结果的区域。

搜索引擎的工作流程十分复杂，但可以简单归纳为三步：如图7所示，首先，搜索引擎的搜索器在互联网上尽可能多的发现网页超链接并爬取其对应的网页内容，从而建立起一个网页数据库；然后，索引器对爬取到的网页中的关键信息建立索引，保存这些索引到数据库中；最后，由检索器根据用户搜索的关键词，在索引数据库中查找用户最关心的页面，按用户感兴趣程度依次排序返回到用户界面。

其中爬取网页是搜索引擎的基础，每个搜索引擎都有自己网页爬取程序（spider），也称为机器人（robot），就是所谓的爬虫程序。该爬虫程序从初始链接节点开始，访问该链接并爬取其链接对应的网页内容，这些网页内容被称为网页快照。然后，继续根据一定的爬行策略，访问其他未访问过的链接，保存对应页面，一直爬行，达到结束条件就停止。

2.1.3 传统搜索引擎

搜索引擎按照其出现的年代可以分为三代：第一代是目录式的检索，用户只能按预先设定的分类进行搜索；第二代是基于关键词的检索，用户可以输入一个或者多个关键词进行搜索；第三代是在第二代上的进一步改进，加入了智能分析，用户能更快更多的搜索到对自己有用的信息。因此，传统搜索引擎根据工作方式的不同，可以分为三类：

1. 目录式搜索引擎：一般以人工方式搜集网页信息，形成摘要，并存放于事先确定的分类框架中，提供目录浏览和关键词搜索的服务。代表的搜索引擎有YAHOO、搜狐和新浪等。
2. 机器人搜索引擎：一个称为蜘蛛（spider）的机器人程序（robot）以一定的爬行策略自动在互联网上搜集信息，由索引器对爬取的页面建立索引，由检索器根据用户输入的关键词搜索索引数据库，返回检索结果给用户。是目前应用最多的搜索引擎，是面向所有网页信息的检索服务，代表的搜索引擎有Baidu、Google等。
3. 元搜索引擎：这类搜索引擎没有自己搜集网页，而是将用户的查询提交给多个搜索引擎进行检索，并将检索结果经过重新排序和去重后，作为自己的结果返回给用户。代表的搜索引擎有搜星等。

传统的搜索引擎都是面向所有信息的，随着信息的多元化增长，这种面向所有信息的综合性搜索引擎显然不能满足特定用户的特殊查询要求，这类用户需求的信息主要是针对单一领域和特定主题的，综合性搜索引擎的查全率和查准率都会比较低。为了解决这类用户的问题，需要一种更加专业的搜索引擎，提供面向主题的检索服务。

2.1.4 主题搜索引擎

搜索引擎经过前三代的发展，人们检索信息的效率已经得到很大的提高，但是，搜索引擎的查全率和查准率还有很大的提高空间【9】，因此只针对特定领域或者特定主题的、用户能定制检索方向和内容的新一代搜索引擎应运而生。

所谓的面向主题的引擎即主题搜索引擎，是指以某一特定领域或主题的网页资源库为目标，通过智能分析采集符合这一领域或主题的网页信息，能够为包括专业信息机构、特定行业领域、学科信息门户、行业专家、公司信息中心等在内的用户，提供一整套的网络信息资源服务解决方案【10】。代表性的主题搜索引擎系统有Elsevier的Scirus系统、Berkeley的Focused Project等，如图8所示。

主题搜索引擎只采集关于特定领域或者主题的网页信息，其索引更小，更易于管理。同时，对于专门领域，却有着强大的搜索功能。主题型搜索引擎和通用搜索引擎的差别主要体现在以下几个方面：

1. 不同的服务目的：通用搜索引擎面向大众用户，提供对普遍主题信息的查询，而主题搜索引擎，则面向专业用户提供对特定专业领域的信息搜索服务。
2. 不同的搜索方式：通用搜索引擎对网络进行海量的提取，以遍历整个Web为目标，而主题型搜索引擎则采用一定的策略预测对相关网页进行预测，动态调整网络爬虫的爬行方向，使系统尽可能地围绕设定主题进行爬行，从而节约网络资源。
3. 不同的硬件和网络要求，通用搜索引擎的硬件需求通常很大，而主题型搜索引擎不对整个互联网进行遍历，不仅节约网络资源，而且不需要维持一个大规模的索引数据库，所以硬件要求相对也比较低。

2.2 主题爬虫技术

2.2.1 网络网络爬虫简介

网络爬虫【11】，也称为蜘蛛WebSpider、WebCrawler、WebRobot或者WebWorm，是一个根据URL链接来遍历网络上的页面，并按照标准HTTP协议访问其页面内容的程序。网络爬虫是搜索引擎的基础核心部分，为了每次检索的信息能够准确全面，需要它把网页爬取下来，建好索引存储在数据库中，检索结果直接从数据库中查找并返回，它的性能好坏直接影响搜索引擎的整体搜索质量。随着技术的发展和人们需求的提高，网络爬虫越来越多地应用于内容安全检测、用户兴趣分析、微博话题挖掘以及个性化信息推送等多种应用中。

通用网络爬虫的基本流程如图9所示，网络爬虫的爬取过程从一个或者多个初始URL开始，不断的增加和移除URL，首先，利用HTTP协议访问初始URL并下载其对应的页面，然后，分析并抽取出网页中出现的新URL，并过滤掉非法URL，最后将未被访问的新URL加入到待访问URL队列中。整个过程一直循环，直到满足提前定义的结束条件。网络爬虫实际上是沿着网页超链接，根据深度优先、宽度优先、或者其他策略遍历网络信息的过程。现代网络爬虫除了必须要具有好的健壮性，还需要具有分布式、可扩展、高性能、高效率、高时效等特性。

通用网络爬虫的结构如图10所示，下面介绍各个模块的主要功能：

1. url分析模块，存放待访问的url队列
2. 页面采集模块，该模块根据HTTP协议访问初始种子URL或者待访问URL队列中的链接，爬取其对应的网页内容，保存下来以供后续模块使用。
3. 页面分析模块，该模块根据页面采集模块爬取到的页面信息进行分析，抽取特征信息和URL超链接，并规范化抽取的URL，以供链接过滤模块使用。
4. 链接过滤模块，该模块对页面分析模块分析过的链接进行过滤，过滤掉已访问过的，或者已存在待访问URL队列中链接。
5. 页面库：存放页面采集模块采集的页面。
6. 初始种子URL：预先设定的一条或者多条初始URL。
7. 待访问URL队列：存放采集下来的页面经过页面分析模块分析后提取出的新URL，以页面采集模块一直访问，直到队列为空或者达到预先设定的结束条件，爬虫结束。

2.2.2 主题爬虫工作原理

主题爬虫是通用爬虫的扩展，不仅仅是对页面的简单抓取，是有选择性的根据主题爬取更多的与主题相关的页面。主题爬虫的基本流程如图11所示，主题爬虫在第一次爬取页面时，需要一个初始种子URL，初始种子URL必须是与主题相关度很高的大型门户网站，这样才能获得较多与主题相关的URL，提高爬取的准确率。接下来，从爬取的页面中抽取超链接信息和特征信息，按照事先确定好的判定网页相关性的算法分析网页的相关性，保留与主题相关性高的页面。然后提取上述页面的超链接，过滤掉非法链接，结合网页内容的相关度或者其它重要度综合得出URL的优先级，放入待访问的URL队列中，最后，取出优先级最高的URL继续爬取它的内容，循环此过程，直到URL队列为空或者满足结束条件就退出。

2.2.3 主题爬虫体系结构

主题爬虫的一般结构如图12所示。主题爬虫基于通用爬虫，增加了主题相关性分析的模块和URL优先级评价模块，其他模块的功能都大同小异，下面主要介绍这两个新增加的模块的功能：

1. 主题相关性分析模块：该模块采用多种相关度计算方法，计算采集到的页面是否属于某一个主题。通过页面的主题相关度值来判断某页面是否与主题相关，如果该页面的主题相关度值高于某个阈值，就存入页面库中，否则，就丢掉该网页。
2. URL优先级评价模块：该模块的用于对待爬取的URL评价优先级，此优先级用于指导爬虫选择下一次爬行的URL，即确定下一个需要爬取页面的URL。一个URL优先级得分越高，就会越早被爬取，反之，得分低到一个阈值时则丢弃该页面及其所有子链接。好的URL优先级评价算法不仅能爬取更多主题相关的页面，还能减轻爬虫抓取到无关页面的负担，节约带宽。

2.2.4 主题爬虫设计目标

主题爬虫的目标是在较少的时间内以较少的资源爬取更多与主题相关的网页【10】。主题爬虫的搜索策略，即主题相关性分析和URL优先级评价策略是实现这一目标的关键所在，其与主题爬虫的目标是紧密相关的。因此主题爬虫主要需要解决以下两个问题：

1. 如何判定爬取下来的网页与主题的相关性？ 对于已爬取下来的页面，一般通过文本检索和挖掘技术计算其主题相关度。
2. 如何确定待访问的URL的优先级顺序？通常是根据已爬取下来的页面的重要程度，这个重要程度可以是页面相关度等多种角度描述，然后将此重要度按一定规则分配给子链接，并按照重要程度高低插入到待访问的URL队列中。

以上两个问题的策略、算法及其改进将在本文的第三章和第四章进行详细阐述。

2.3 其他相关技术

2.3.1 信息推送技术

随着互联网信息数量的快速增长，用户很多时候花费大量时间也难以在互联网上查到自己需要的信息。信息推送技术正是这样的情况下诞生的，推送技术通过用户特定的需求，按时将用户感兴趣的最新内容主动发送给用户，这样，用户就不必浪费精力在互联网上搜索。

所谓信息推送技术就是指网络公司通过一定的标准或者协议，从互联网或者专业制作商获取信息，通过特定的渠道向用户发送信息的新型信息传播技术【11】。

目前的信息推送技术主要有以下几种：

1. 邮件型推送：需要用户设置自己的邮箱，然后将用户感兴趣的信息推送到用户的邮箱中。
2. 网页型推送：在一个特定的网页上推送用户感兴趣的信息，用户登录自己的账号查看。
3. 专用型推送：在一个特定的APP软件上推送用户感兴趣的信息，用户登录自己的账号查看。

信息推送技术的核心主要是通过用户访问纪录等信息对用户兴趣进行建模，然后跟据用户兴趣模型对用户推送其感兴趣的内容。主要方法有基于内容的推送、基于协同过滤的推送和混合推送【11】。

个性化信息推送系统则是指运用信息推送技术、爬虫技术等，向用户推送其感兴趣的信息的新型信息传播系统【11】。结构如图13所示，个性化信息推送系统让用户自己输入关注的网站和领域，及时推送网站上该领域上更新的信息给用户，弥补了传统搜索引擎难以解决的信息过载的不足，极大提高了搜索的准确率。

2.3.2 消息队列技术

消息队列技术是应用程序间的通信方法。应用程序间通信不需要建立专用连接，通过读取队列里的消息来通信，使得应用程序接收方和发送方不必同时执行。其成熟的产品主要有有IBM WEBSPHERE MQ、RabbitMQ等。

RabbitMQ是基于AMQP的erlang实现的。其结构如图14所示，核心是Message Exchange、Message Queue两种实体，一个应用程序产生的消息不是直接发送给另一个应用程序，而是发送给Message Exchange，同时指定一个Rounting Key，如果与Message Queue的Binding Key匹配，Message Exchange就将消息发送给Message Queue，等待另一个应用程序的读取。主要有以下几个特点：

1. 可以在多个平台上运行，支持多种操作系统。
2. 提供消息的持久性支持，即服务器意外崩溃后消息能够恢复。
3. 提供了消息接受者和消息发布者的确认机制，保证消息准确接收。

消息队列技术应用到个性化信息推送系统中，使得个性化信息推送系统中采集到的信息准确地推送给用户，保证了信息的准确性和安全性。

2.4 本章小结

本章首先根据搜索引擎的背景，简单介绍了搜索引擎的基本概率和工作原理，接着分别介绍了传统搜索引擎和主题搜索引擎的特点，进一步展示出主题爬虫在搜索引擎的作用。然后，重点介绍了主题爬虫，通过对比主题爬虫与传统爬虫的基本原理和结构，详细的讨论了主题爬虫的几个关键问题。最后，介绍了个性化信息推送系统其他相关的技术，包括信息推送技术和消息队列技术。

第三章 爬虫爬行策略

主题爬虫与通用爬虫的最大区别在于其主题相关度计算及相关的爬行策略，所以要实现一个完整的爬虫系统，采用合适的爬虫爬行策略及其重要。本章结合已有的爬行策略和自己的理解，对现有的爬行策略进行了分析、研究与对比。

3.1 爬虫爬行策略

爬虫爬行策略是爬虫的核心问题，指的是如何在海量的信息网站中爬取到人们感兴趣的信息，由此可见爬虫爬行策略的好坏直接影响整个爬虫系统的性能。如果能设计出一个好的爬行策略，那么爬虫就能最大可能达到我们需要的效果。但是由于目标不同，爬虫的爬行策略也会不同，接下来本节先主要讲述已有的爬虫爬行策略。

3.2 通用爬虫爬行策略

通用爬虫的目标是遍历整个互联网，尽可能多的采集网页。其爬行策略主要为深度优先、广度优先和深度优先和广度优先结合三种。深度优先、广度优先以及深度优先和广度优先结合的爬行策略都是将网络看成一个复杂而庞大的有向图，将网络上的所有网页当成节点，网页之间的超链接当成节点之间的连接，根据图论的搜索方法，对网页节点采取深度优先、广度优先或者深度优先和广度优先结合的遍历策略。下面以一个结构图15来说明这三种算法的区别。

3.1.1 深度优先

深度优先的爬行策略是从初始网页节点开始，首先访问该节点，然后沿着一条超链接路径出发，访问其邻结点，一直爬行到没有包含任何超链接信息的网页节点为止，再返回访问其他未被访问的节点，循环下去，直到所有的节点都被访问到。那么根据深度优先的爬行策略，图上的网页节点访问路径应该是ABDFCE。深度优先的爬行策略优点是爬取的页面很全面；缺点则是效率很低，当页面目录很深的时候，容易导致爬虫陷入。

3.1.2 广度优先

广度优先的爬行策略是从初始网页节点开始，访问此节点后，就访问其他没访问过的相邻节点，然后再访问剩下节点的邻结点，直到爬行结束。比如上图，从A节点访问开始，继续访问时可以选择A的邻结点B或者C访问，所以广度优先的爬行策略有多种爬行顺序。按照上图总共会出现多种爬行顺序，ABCDFE是其中的一种。广度优先的爬行策略优点是当初始网页有足够多的链接时，会得到很好的爬取结果；但是，广度优先的爬行策略缺点也很明显，当爬行深度越来越高时，则会产生许多无效的网页，导致算法的效率降低。

3.1.3 深度优先和广度优先结合

针对深度优先和广度优先各自的优缺点，又出现了深度和广度结合的优先爬行策略。这种爬行策略会预设爬行深度或广度，当爬虫爬行到预设深度或者预设广度时就停止工作。所有它相比深度优先和广度优先，都提高了效率，但还是会爬取不少的无效网页。

3.1.4 对比分析

从以上几节发现，深度优先、广度优先以及深度优先和广度优先结合都几乎遍历整个网络，能爬取大量的网页，非常适用于通用爬虫的爬行。但是，对于主题爬虫来说，上述三种策略都没有对网页和网页内容作出评估分析，会爬取大量与主题无关的网页，不能满足主题爬虫的需求。

3.2 主题爬虫爬行策略

由于通用爬虫的爬行策略效率低，会产生大量无效网页的问题，不能应用于主题爬虫算法中。目前主题爬虫的爬行策略主要有基于网页链接、基于网页内容的爬行策略，下面讲述这两种爬行策略的原理和优缺点，并进行对比。

3.2.1 基于网页链接的爬行策略

万维网是超文本文档的集合，超文本文档相互之间有一定的链接关系，形成一定的链接结构。基于网页链接的主题爬行策略最重要的就是要对超文本文档的链接结构进行分析，通过对链接的有效评估，决定爬行的顺序。万维网信息分散，但经研究表明，其链接结构具有自组织性：通过超链接，相同或者相关主题的网页结点聚集在一起，形成结点区域，区域内部的结点存在紧密联系，而区域之间的结点很少或者根本没有联系。并且，区域内部的结点也存在一定的规律，其结点主要分为两类，一类指向其他结点，一类被前一类结点所指向。前一类结点称为中心结点，后一类结点称为权威结点。万维网的这种链接自组织性结构为链接的评估提供了依据，如何评估链接主要有以下假设：

1. 从页面A到页面B的一条链接是页面A对页面B的一种推荐。
2. 若页面A与页面B被同一页面链接，则他们可能有相同或者相关的主题。

Page Rank算法和Hits算法是常见的链接分析算法，两者都是通过分析网页的拓扑结构，得出网页的重要程度评价。

1. Page Rank算法

PageRank算法由S.Brin和L.Page发明，S.Brin和L.Page利用网络的超链接结构给所有的页面定义了一个Page Rank值，用Page Rank值的高低代表页面的重要性，以此确定爬行的页面顺序。Page Rank算法的原理是将学术论文的引用分析方法应用到网页链接中。一篇论文的重要性是可以根据其被引用的次数来判断，同样一个网页的重要性也可以根据其被链接的次数来判断。具体来讲，如果网页A链接到网页B，相当于网页B的被链接次数（Page Rank值）增加，同时网页A的Page Rank值减小，最后由每个网页的Page Rank值来评估链接的重要程度。

Page Rank算法最开始是运用于Google搜索引擎搜索结果的排序，最近较多应用于网页链接重要性的评估中。Page Rank计算公式如下：

其中，代表网页u的重要度，代表指向网页u的页面的重要度，代表指向页面u的超链接数，为指向u的网页的集合，c为常数，一般c取0.85。经观察公式，可以看出影响页面u的重要度的因素有以下几点：

1. 页面u的入度，如果链接到u的网页数量越多，越大。
2. 链接到页面u的网页重要度，链接到u的网页重要度越高，越大。
3. 链接到u的网页出链数量，链接到u的页面出度越大，平均传递给u的重要度值就越小。

Page Rank算法通过网页链接的Page Rank值的高低代表网页的重要性程度，并成功运用与Google搜索引擎，证明了其有效性。但这种算法只考虑了网页链接，没有考虑到网页内容，如果运用到主题爬虫中，易造成“主题漂移”现象，导致爬取到许多与主题不相关的页面。。所以在本文的爬虫系统中，考虑采用结合一种基于网页内容判断主题相关度的算法，在4.1节研究了Fish Search基于网页内容判断主题相关度的算法。

1. Hits算法

Hits算法由Kleinberg于1999年提出，是继Page Rank算法后又一著名的基于网页链接的计算页面重要程度的算法。在Hits算法里面，Kleinberg提出了两个新概念：权威型网页（authority Page）和中心型网页（hub Page）。权威型网页指的是被很多链接指向的页面，即与主题相关度很高的页面，是Hits算法最后得到的页面；中心型网页则是本身没有什么特点，却指向很多权威型网页的页面。每一个页面都有一个权威值（authority）和中心值（hub），如果一个页面被很多页面链接，则其权威值就会很高，如果一个页面链接到很多权威型网页，则其中心值也应该会很高。权威型页面和中心型页面之间的关系如图16所示，页面4的权威值等于指向它的页面1、页面2和页面3的中心值的和；页面4的中心值则等于页面4指向的页面5、页面6和页面7的权威值的和。通过权威型页面和中心型页面的这种相互关系，来发现重要程度高的网页，这就是Hits算法的基本原理。

Hits算法构造Web子图主要有以下两个步骤，如图17所示：

1. 构造根集合（Root Set）：将预先设定的初始URL加入到跟集合 ，
2. 扩展根集合到基集合（Base Set）：将根集合中的网页所指向的URL加入根集合，将指向根集合中的网页的URL也加入到根集合，最后形成基集合。

Hits算法通过以上的步骤得到Web子图后，进行迭代计算直到收敛，计算出每个页面的权威值和中心值，最后，找出权威型页面。Hits算法虽然收敛速度快，但其同Page Rank算法一样，忽视了网页内容与主题的相关性判断，造成“主题漂移”现象，从而爬取到许多与主题不相关的页面。

3.2.2 基于网页内容的爬行策略

基于网页内容的爬行策略，主要以向量空间模型等信息检索模型为基础，利用网页页面内容（包括网页中文本信息、URL、锚文本和锚文本的上下文等）来评估网页的重要性，以此决定爬行顺序。基于网页内容的爬行策略的核心问题就是对上述纯文本和超文本进行分类和聚类。

1. 纯文本的分类聚类就是对页面纯文本进行检索，准确的说，指通过计算机使用自然语言中的词语对纯文本进行匹配查找。常用的检索技术有精确检索、截词检索、布尔检索和限定范围等。在主题爬虫中，由于文本检索技术忽略了网页页面内部的结构信息，必须限定检索的关键词集合下进行检索，很少单独使用。
2. 超文本的分类聚类不仅包含了对一般的文字，还包含了一些对文本信息处理有用的特征信息的检索。如图18所示，网页文本是一个半结构化的标记语言，其中的HTML标记语言具有结构特征和语义信息，根据特定的HTML标签检索文本，排除了无关的干扰信息，使得文本检索更加精确、快速。从而提高了爬虫爬行的效率。

Best First Search、Fish Search是常见的基于页面内容的搜索算法。都是通过分析网页内容与主题的相关度，来得出网页的重要程度评价。

1. Best First Search 算法

Best First Search算法是最早出现的基于网页内容的搜索算法，该算法的基本思想是通过比较网页内容与主题关键词的相关度来评估网页的重要程度，以确定待爬行URL队列中爬行顺序。Best First Search一方面不断从待爬行URL队列中选择主题相关度高的链接爬行，另一方面，如果待爬行URL队列满了，则从队列中丢弃主题相关度低的页面。

Best First Search算法的具体步骤如下：

1. 首先选择一个初始URL插入待爬取URLs队列中。
2. 从待爬取URLs队列中取出初始URL，下载初始URL对应的页面，提取初始网页的子链接，并计算子链接对应的页面与主题的相关度值。
3. 如果待爬取URLs队列未满，将新URL插入到该队列中。
4. 如果待爬取URLs队列已满，比较新URL与队列中的所有URL的相关度值，丢弃队列中相关度值最低的URL。
5. 队列为空时停止爬取。

Best First Search 算法根据网页内容与主题的相关度来评价网页的重要程度，以确定待爬取URL的爬行顺序。它的优点是算法流程简单清晰，容易实现，在主题相关性强的网页爬行时效果较好。但评估网页重要性的时候只考虑网页内容，忽视了网页链接的结构。

1. Fish Search算法

标准的Fish Search算法是由De Bra在1994年提出的。该算法将爬虫的爬行过程比喻成鱼的捕食过程，算法中的网页就是鱼的食物，沿某个方向的遍历就是鱼的捕食路径。当爬行过一个网页后，提取的新的URL就是鱼的Children，一直循环下去，直到鱼找不到食物饿死，即爬虫找不到任何链接的时候就结束爬行了。下一章将详细介绍Fish Search算法及其改进。

3.3 策略对比

基于网页内容的爬行策略需要对纯文本和超文本进行分类聚类，首先使用超文本分类聚类方法将HTML语言中的纯文本抽取出来，再使用纯文本分类聚类方法，检索其中的关键词，最后与主题关键词进行比较判断页面与主题的相关度。

基于网页链接的爬行策略需要计算页面节点的入度和出度值，只有在爬虫对网络拓扑结构有相对完整的统计情况下，才能计算出准确的网页的重要性程度值。另外，该爬行策略没有考虑到网页内容的重要程度，也许页面在当前网页范围里很重要，但并不是所需要的。所以在主题爬虫系统中，基于网页链接的爬行策略不能单独使用，需结合能判断网页主题的基于网页内容的爬行策略。

3.4 本章小结

爬行策略是主题爬虫的核心，本章介绍了现有的多种爬行策略并进行了对比。首先介绍了通用爬虫的三种爬行策略，并对比分析其优缺点；接着重点介绍了主题爬虫的主要爬行策略：基于网页链接和基于网页内容的爬行策略，并对这两类爬行策略的代表性算法进行了研究；最后，详细分析和对比了基于网页链接和网页内容这两种爬行策略。

第四章 爬虫爬行策略的改进

主题爬虫的目的是爬取到更多与主题相关的页面，减少爬取到与主题不相关的页面的可能性。爬行过程中，需要采用一定的爬行策略，优先选择与主题最相关的网页进行爬行，并丢弃与主题不相关的网页。本章对基于页面内容的爬行策略的代表算法Fish Search算法，进行了详细的分析和研究，并针对Fish Search算法存在的问题，提出了一种改进了的Fish Search算法，最后，结合Page Rank算法和改进了的Fish Search算法设计出基于链接和页面内容的主题爬虫策略。

4.1 Fish Search算法

Fish Search 算法是一种基于网页内容的爬行策略的经典算法，本节会对该算法进行研究和分析。

4.1.1 Fish Search算法思想

Fish Search 算法，也称为鱼群算法。该算法将互联网假想成海洋，互联网中的链接假想成鱼，主题相关性高的网页假想成鱼的食物，因此爬虫的爬行过程就是鱼群的捕食过程，沿某个方向的遍历就是鱼群的捕食路径。当爬行过一个网页后，即捕食成功后，鱼群会在这里繁衍后代，提取的新的链接就是鱼群的后代，接着鱼群继续前行寻找食物。如果鱼群游到下一个地方时又发现了食物，就将父结点的遍历深度赋值给儿子结点；如果鱼群游到下个地方时没有发现食物，即网页与主题不相关，就将儿子结点的遍历深度设置为父结点的遍历深度减一，继续前行。若一直找不到食物，链接的遍历深度会逐渐变为0，鱼就饿死了，即爬虫在一定的深度找不到主题相关的链接的时候就结束爬行了。这就是Fish Search算法的基本原理。

4.1.2 Fish Search算法过程 算法过程不一样

Fish Search算法的关键之处在于动态维护了三个URL队列，分别是前端队列B、中间队列M和末尾队列E，其算法描述如下：

1. 首先选择一个初始URL放入待爬取URLs队列中，并对最大爬取深度depth赋值。
2. 提取初始网页的子URL，如果初始网页是主题相关的，则其子URL的相关度值为1，depth值不变，如果初始网页不是主题相关度哦，则其相关度值为0，depth值减一。并按照（3）（4）（5）将新的URLs插入队列。
3. 若该URL对应的网页与主题相关，就选取m\*width（m，width为预先确定的参数）个URL插入到前端队列B中。
4. 若该URL对应的网页与主题无关且depth不为0，就选取width（width为预先确定的参数）个URL插入到中间队列M中。
5. 剩余的URLs插入到末尾队列E中。有充足的时间才会爬取这个队列的URLs。
6. 队列为空时停止爬取。

算法的伪代码如下：

Get初始URL，关键词，时间限制；

将初始URL插入空的队列中；

While队列为空 and 时间限制未到

Pop the first node as 当前节点；

Compute 相关度 of 当前节点；

If 当前节点相关度为1

Then For each 当前节点的子节点

加入到队列头，set potential-score 为 1；

Else

For each 当前节点的子节点

加入到队列右侧，set potential-score 为 0；

END While

4.1.3 Fish Search算法优缺点

Fish算法对待爬行的URL进行了优先级判断，且其原理简单，容易理解。但是该算法存在两个问题，第一个问题是只是考虑简单匹配关键词来计算相关度，导致结果准确度不够；第二个问题是使用离散点0或者1来表示主题是否相关，无法全面表示页面与主题的相关性。

4.2 改进的Fish Search算法

4.2.1 算法分析

由上一节我们了解到，Fish Search算法存在一些问题。首先，Fish Search算法采用布尔模型信息检索方式计算页面主题相关度。首先将网页中的文本提取出来分词，得到一个关键词的集合，然后与主题中的关键词集合对比，统计同时出现的关键词个数，最后用同时出现的关键词个数除以主题中关键词的个数，得到的比值就是判断该网页的主题相关度的重要参数。如果相关，其Potential-score值就为1，如果不相关，其Potential-score值就为0，该Potential-score值是离散的，无法完全表示页面与主题的相关性。

然后，Fish Search算法对所有页面上的关键词没有区分，忽视了关键词位置的重要性。一个网页上有多种文本，不同标签里的文本意义不同，对网页主题的重要性影响也不同，不考虑关键在网页中的位置，导致相关度结果不准确。因此本节对Fish Search算法的主题相关度方法进行了改进。

4.2.2 算法改进

准确获得一个网页的主题是计算网页与主题相关度的前提，网页的标签是语义化的，能表示网页主题的文本主要存在于页面标题title和关键词keywords、页面里链接a的锚文本以及页面里的其他标签。页面标题和关键词最能代表网页的主题，其权值最高；页面里链接的锚文本一般是其子网页的主题，对页面的主题判断也有一定的作用，其权值居中；其他正文中的文本一般只是提到而已，对页面的主题判断作用不大，其权值最小。所有针对关键词在网页中出现的不同的位置赋值公式如下：

设主题关键词出现在网页标题中的权值为 ，并根据公式2带入参数，得到的计算公式入公式3所示。

设主题关键词出现在网页关键词中的权值为 ，并根据公式2带入参数，得到的计算公式入公式4所示。

设主题关键词出现在网页锚文本中的权值为 ，并根据公式2带入参数，得到的计算公式入公式5所示。

公式（3）（4）（5）中的T、K和L分别代表主题关键词在网页标题、网页关键词和网页锚文本中出现的次数，表示关键词i分别在网页标题、网页关键词和网页锚文本中出现的次数，w为公式2中的权重。

此时可得出网页的权值的公式如下。

将公式（3）（4）（5）的带入公式6得：

主题关键词的权重包括两部分，第一部分关键词权重指的是出现在页面中的关键词的权重，即；另一部分关键词权重则指的是为未出现在页面中的关键词权重，设为，其计算公式如下：

其中，

由此得到网页与主题相关度的计算公式如下，用sim表示相关度的值。

将公式（6）和公式（7）代入上式可得：

由公式观察得sim的值在0到1之间，是连续的，使用sim表示页面的主题相关度，改进了Fish Search相关度值离散的问题，可以通过其大小评价网页与主题的相关程度和优先级排序。

4.2.3 算法具体实现

伪代码

4.3 Page Rank结合改进的Fish Search算法

4.3.1 算法分析

改进后的Fish Search算法度量页面主题相关度时，考虑了主题关键词在网页中的位置，并最后得到一个连续的主题相关度度量公式，对比原始的Fish Search显然要好些。但是，无论是原始的还是改进后的Fish Search算法，都没有考虑网络拓扑结构的影响，不能发现高重要度的网页。并且，爬虫初期由于爬取的网页数量少，无法确定比较完整的网络拓扑结构，导致算不出准确的Page Rank值。所以，针对Fish Search算法和PageRank的不足，本节将PageRank算法与改进后的Fish Search算法结合，设计出基于Page Rank和改进后的Fish Search算法的主题相关度计算方法。

4.3.2 算法改进

该算法由Page Rank算法与改进后的Fish Search算法组成，其主题相关性便是这两种算法的相关度综合得到。设相关度为，其综合相关性的计算公式为：

其中，权值为给定的参数，用于平衡R和sim的比例。R值是公式2计算出来的页面的Page Rank值，sim是公式8计算出来的页面内容的主题相关度值。

4.3.3 算法步骤

（1）初始化URL列表和主题关键词。

（2）提取初始URL的子链接。

（3）根据相关度计算方法计算URLs的主题相关度，并跟预先设置的阈值进行比较，并放入队列指定位置。

（4）一直循环下去直到队列没有URL。

算法的流程如图所示：

4.3.4 算法的具体实现

伪代码

4.4 本章小结

本章对爬虫爬行策略的核心问题，如何计算页面的重要程度以确定待爬行的URL的优先级进行了研究。首先研究了基于网页内容的爬行策略的常用算法Fish Search；然后针对Fish Search算法的不足，考虑了页面关键词的位置，提出了一种改进的Fish Search算法；最后结合Page Rank算法，设计出基于Page Rank算法和改进后的Fish Search算法的页面主题相关度计算方法。

第五章 实验结果及分析

本章设计了一个通用爬虫框架，分别对Page Rank、Fish Search、改进后的Fish Search以及结合PageRank和改进后的Fish Search的算法进行实验，分析实验结果并得出结论。

5.1 实验目的

基于第四章提出的主题相关度计算算法，分别实现基于Page Rank、Fish Search、改进后的Fish Search以及结合PageRank和改进后的Fish Search的算法的主题爬虫。目标是：改进后的算法能在较短的时间内，爬取大量与主题相关的页面，并且系统能稳定运行。

5.2 实验环境

实验基于MyEclipse开发环境，使用Java编程语言和Mysql数据库开发完成。硬件环境为多台台式机，配置为：Pentium 4双核，主频1.4GHz，内存4GB。

5.3 评价指标

为了很好的评价改进后算法的效果，引入了查准率、发现率和平均每个单位页面的爬取时间性能指标。查准率指的是爬取到的主题相关的页面数与实际主题相关的页面数之比。发现率是爬取到的主题相关的页面数与爬取到的页面总数之比。平均每个单位页面的爬取时间指的是爬取页面的总时间与爬取到的页面总数之比。

5.4 实验设计及数据

实验1采用新浪网体育类目主页和关键词足球，每爬取100个页面，统计与主题相关的页面个数，计算其与爬取页面总数的比值，即发现率。

实验2采用上述新浪网体育类目的数据作为数据源，其中足球类页面有2000个页面，同样的，每爬取100个页面，计算查准率、发现率和平均每个单位页面的爬取时间。

实验实现了一个通用的爬虫框架，下面结合框架界面介绍实验的主要步骤：

1. 输入初始URL和主题关键词。如图1所示。
2. 点击开始爬取。 系统下载初始URL对应的页面，抓取页面上的链接，并选择优先级高的链接爬行。如图2所示。
3. 爬取下来的页面保存在Mysql数据库。如图3所示。

5.5 实验结果及分析

实验1

实验2

5.6 本章小结

第六章 及时推新闻推送系统的实现

6.1 需求分析

随着时代的进步和计算机技术的迅速发展，以互联网为特征的第四媒体给传统的新闻媒体带来冲击和挑战，伴随着互联网信息时代的带来，信息数量高速增长，据不完全统计，全球互联网文本数据100亿条，其中招聘信息大于10亿条，新闻大约8亿条，招商信息大约3亿条，通知信息大约2亿条。并且互联网信息更新速度飞快，在浩如烟海的网络资源里，想要及时的获取想要的信息变得越来越难。针对上述问题，有以下几种解决办法：

1. 使用搜索引擎，目前的搜索引擎都是面向网络上所有的信息的，随着信息的增长和全面化，针对受限领域和特定主题的搜索的准确率和查全率都是很低的。
2. 使用新闻网站以及客户端，用户可以设置自己感兴趣的主题，阅读自己感兴趣的内容。但是，新闻网站以及客户端上的新闻大部分是用户不关心的，由于不能定制关注的网页，不能推送用户真正需要的信息。

综上，我们知道，用户真正需要的是一个完全定制的信息推送系统。在这个系统里，通过用户定制的关注的网站和关键词，将每天推送该网站更新的包含该关键词的页面给用户。该系统的核心就是爬虫子系统，即如何准确全面地爬取用户定制的感兴趣的网页，因此将本文设计并实现的主题爬虫算法应用到该系统中，为系统提供用户定制的感兴趣的网页。

6.2 系统概述

及时推新闻推送系统是四川省成果转换项目，分别实现了PC端、Android客户端和IOS客户端，并于2014年5月20日全部成功上线。该系统主要由3个子系统组成：

1. 爬虫子系统：根据用户定制的感兴趣的网页和关键词，爬取该网页上每天更新的包含关键词的所有URL，并存入数据库。
2. 信息推送子系统：根据用户特征、用户浏览数据等，每天为用户推送信息到PC网页客户端、Android客户端、IOS客户端以及手机或邮箱。
3. 信息展示子系统：通过PC网页客户端、Android客户端和IOS客户端显示系统给用户推送的信息，方便用户随时随地查看。

以上三个子系统中我负责的是爬虫子系统以及各个客户端的展示子系统。整个过程中，首先，爬虫子系统为信息推送子系统提供爬取到的用户定制的网页信息，接着，信息推送子系统根据特定规则将信息推送给用户，最后显示在信息展示子系统上。因此，爬虫子系统是整个系统的基础，也是关键的一个步骤，如果爬取到的网页信息的准确率和覆盖率不够，那么后面的推荐和展示就没有办法实现。

图2就是该系统的软件架构图。

图2中的几个关键模块主要是主题相关网页信息爬取、网页信息推送和网页信息展示。主题相关网页信息爬取模块是最基础关键的模块，主要包括网页采集模块、网页主题分析模块、链接优先级计算模块和网页信息持久模块。主题相关网页信息爬取模块将爬取的网页信息持久化后存入数据库中，网页信息推送模块从数据可中获取所有的网页信息进行分析后推送个用户，网页信息展示模块就直观地显示推送给用户的信息。

图3就是该系统的硬件结构图

图3中可以看出，系统总共使用了三台服务器，一台用于主题爬虫，一台用于web服务，一台用于数据库。为了充分使用服务器的资源进行主题爬虫，将用于主题爬虫的服务器虚拟化，运行了两个虚拟机，分别装linux操作系统运行主题爬虫的爬行子系统。Web服务器采用开源的Tomcat服务器，用于给用户推送网页信息，并展示在各个客户端上。数据库服务器则是用于存储爬虫子系统爬取的网页信息数据。

6.3 系统主要模块实现效果

由于爬虫子系统和信息推送子系统都是在后台运行的，本节主要分别以不同客户端展现信息展示子系统的实现效果。

6.3.1 PC客户端

6.3.2 Android客户端

6.3.3 IOS客户端

6.4 本章小结

本章首先对及时推新闻推送系统进行了需求分析，接着介绍了系统的构成和软硬件结构，最后分别以PC客户端、Android客户端和IOS客户端的运行情况展示了系统相关模块的实现效果。

第七章 总结

【8】李晓明，闫宏飞，王继民著.搜索引擎-原理、技术与系统.北京：科学出版社.2005

【9】候震宇.主题型搜索引擎的研究与实现.中国科学院研究生院硕士学位论文.2003:2-9页

【10】杨薇.基于主题的搜索引擎.长春:长春工业大学硕士学位论文.2007.

【11】邱哲,符滔滔.开发自己的搜索引擎【M】.人民邮电出版社,2007.6

【12】M.Balabanovie, YShoham. ”Leaming Information Retrieval Agents:Experiments with

Automated WebBrowsing”.In Working Notes of the AAAI Spring SymPosium on

Information Gathering from Distributed, Heterogenous Envlronments, Stanford University,

1995.

【13】Zonghuan Wu, Weiyi Meng, Clement Yu, Zhuogang Li. ”Towards a Highly-Sealable and

Effective Metaseareh Engine”, In Proeeedings of the 10th Intemational World Wide Web

Conferenee,2001.

[14]Erik selberg and Oren Etzioni. ”Multi-Service Search and Comparison Using the

MdaCrawler”. In Proe.4th world Wide Web Conferenee,Boston,MAUSA,Deeember1995