基于 Django 框架的智能商务监控系统的设计*

夏志富[1],王晗璐[1],李玉平[1],曹 磊[2],夏 斌[1]
(1. 上海海事大学 信息工程学院,上海 201306; 2. 同济大学 电子与信息工程学院,上海 201804)

摘 要:随着电子商务的迅速发展,商品在电商平台的排名变化信息愈来愈受到大家的关注。市场上现有的排名查询工具主要是基于 C/S 构架,因为电商平台的变化,需要频繁更新软件,使用较为不便。为了方便用户对商品排名信息的查询设计出一种基于 B/S 框架的排名查询工具。该工具实现了同一商品的多关键词实时排名查询,并且能够让用户自定义产品监控列表并对列表中的产品排名变化情况进行长期监控。本系统构架采用 Django 来设计,主要功能采用 Python 2.7 语言来开发,云端采用稳定便捷的亚马逊公司的 AWS 云计算平台进行服务器端的部署和搭建,经过上线测试后发现系统达到了良好的效果。

关键词:电子商务; 爬虫; 文本相似度; 云计算
中图分类号:TP391.9 **文献标识码**:A DOI: 10.19358/j.issn.1674-7720.2016.12.008
引用格式:夏志富,王晗璐,李玉平,等. 基于 Django 框架的智能商务监控系统的设计[J]. 微型机与应用,2016,35(12):21-23,27.

The design of intelligent business monitoring system based on the Django framework

Xia Zhifu[1], Wang Hanlu[1], Li Yuping[1], Cao Lei[2], Xia Bin[1]
(1. College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China;
2. College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China)

Abstract: With the rapid development of e-business, the changing information of the rank of the commodities becomes more and more important. The existing ranking query tools are mainly based on C/S architecture. Because of the change of the e-commerce platform, users need to frequently updated software, which brings much inconvenience. To provide a convenience query tool for the users, we developed a B/S framework based query tool, which is able to query multi-keywords of one commodity at same time. It allows user to manage the product list and support long term morning for the ranking information. The system architecture is designed with Django and the programming language is Python 2.7. The Amazon's AWS Cloud computing platform is used as Cloud server in this system. After deployed on AWS, the online test result shows that the proposed system achieved all objectives with good performance.

Key words: e-business; the crawler; text similarity; Cloud computing

0 引言

电子商务的兴起促进了跨境贸易的发展,作为当前最流行的跨境电商平台,阿里巴巴拥有数量庞大的用户群体,约有 40 万家电子商务公司入驻阿里巴巴平台。平台上每家公司商品的销量与其商品在阿里平台上的排名情况紧密相连。商品排名越靠前,关注度就越高,销量就会更好。因此提升商品排名是提升销量的重要手段。

目前关于阿里国际站的产品排名查询工具主要有两类,一类是阿里后台提供的排名查询工具,但这个工具只能一次查询一个关键词,使用起来不太方便而且没有自定义关键词查询排名功能。另外一类就是由第三方公司提供的排名查询工具,但主要是 C/S 构架,需要安装客户端软件。因为阿里巴巴服务器经常会有变化,所以客户端软件也需要经常更新,给用户使用过程中带来不便。并且此类软件不具备长期追踪产品排名变化的功能,公司不能及时了解自己商品排名变化情况。因此本文设计了一个基于 B/S 构架的产品排名查询及监控系统,用户通过浏览器登录本系统就可以进行商品排名查询,并且可以长期追踪商品排名变化情况。

*基金项目:上海市科学技术委员会资助项目(14441900300);国家自然科学基金(61550110252);同济大学嵌入式系统与服务计算教育部重点实验室开放课题

1 系统设计

1.1 系统架构

系统基于 Django 架构[1]的 MVC 模式:分为 Model 层、View 层、Control 层,将业务逻辑、显示逻辑和数据逻辑以低耦合、高复用的形式展现出来,便于系统后期的扩展和维护。

在 View 层,利用 Django 自带的模板系统[2]跟前端开源框架 Bootstrap 结合,增强用户的交互体验和提高前端页面开发效率。在 Model 层,系统采用 MySQL 关系型数据库,并利用 Django 的 ORM 机制将 MySQL 中的数据以对象接口的方式进行封装,极大方便了数据的查询和操作。

在 Control 层,系统控制器通过分析请求、逻辑判断、模型操作以及重定向视图等将整个系统业务流串联起来。系统结构及逻辑流程如图 1 所示。

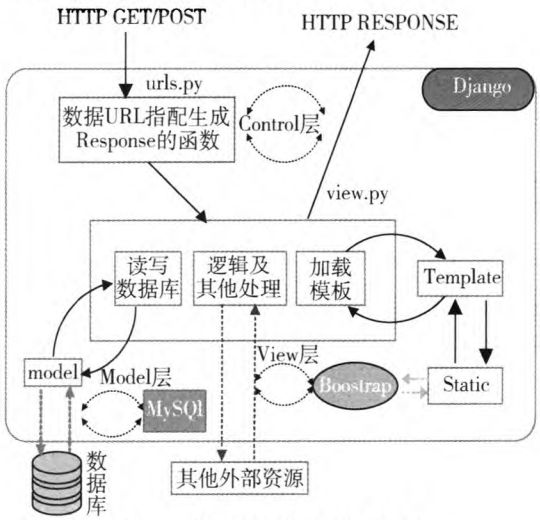

图 1 系统组成结构及逻辑流程图

1.2 系统功能结构

该系统功能主要分为三个部分。

(1)显示逻辑模块

系统前端静态页面利用前端开源框架 Bootstarp 实现,里面内置了丰富的 CSS 样式库,可以快递开发优美的页面。系统动态页面采用 Javascript 开源框架 Jquery 实现,能够很方便地操控鼠标点击事件和后台数据的异步传输。

(2)业务逻辑模块

用户注册登录后输入商品名称就可以直接检索出该商品对应的 3 个关键词,并可以在下拉框中选择备选商品,或者删除备选商品。当用户输入商品名发生错误时可以通过纠错机制告知用户,并利用相似度算法[3]自动从数据库中匹配出最相近的商品名,减少用户输入时间。在批量导入查询模块中,用户可以上传 txt 格式的待查询商品名文件,系统会自动检索出其排名结果,并以 Excel 格式供用户下载查看。在管理产品页面中,用户可以添加和删改监控的商品并观察商品排名的变化趋势,可以按时间段选择商品在指定日期的排名变化情况。

(3)数据逻辑模块

通过后台 Celery 定时任务设定闲时爬取数据[4],定期自动地通过多线程并发更新数据,并在后台服务器计算好商品排名的变化情况,以便用户可以立即从数据库中调取数据查看,无需等待时间。

1.3 数据处理流程

在查询页面中进行商品查询时,如果用户是首次查询某个商品则系统进行实时商品排名查询,并将排名信息存入数据库。这些信息被保存下来以后,系统后台设置了每天定时任务,会在设定的时间闲时爬取数据以更新排名和排名变化情况。当用户输入以前查询过的商品名时就可以直接从数据库中调取其排名和排名变化数据,这样可以减少服务器在同一时间的压力,提升系统查询的响应速度。系统数据处理流程图如图 2 所示。

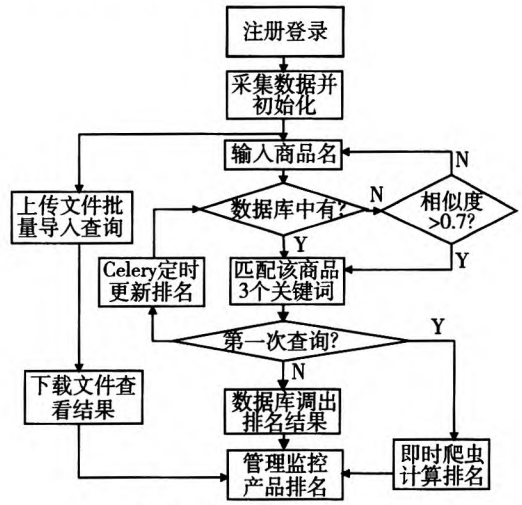

图 2 系统数据处理流程图

2 系统实现

2.1 获取数据资源

网络爬虫是获取数据最快速有效的方法,是构建搜索引擎最重要的组成部分之一,通过对阿里国际站点爬虫获取数据是该系统构建的基础。

本系统获取商品数据分为以下流程。

(1)通过用户输入的商品名在数据库中检索出其对应的关键词,根据其关键词匹配出对应搜索结果的 URL 列表集合。

(2)通过 Python 多线程爬虫[5]获取到 URL 列表集合对应的网页源代码,并对每个网页源代码打好标记后装载于 queue 队列中,以便后面将数据以原顺序展示出来。

(3)取出 queue 队列里的网页源代码,并使用 Xpath 解析工具通过多线程方式去解析网页源代码得到商品数据列表,然后通过原先打好的标记对商品数据列表按照原网页索引排序,最终得到以原顺序输出的商品列表,最后通过列表索引计算排名。

2.2 数据库设计

系统中利用 Django ORM 对象设定表之间的外键关联,建立好数据之间的从属关系,从而方便通过条件筛选出对应的数据。本系统创建了 8 个数据表,主要通过 loginuser(用户信息表)和 middleuser(查询中间键表)作为桥梁与其他数据表建立外键关联。通过 loginuser 表与其他表关联使得用户的查询和数据信息管理可以通过外键把数据独立起来,形成以每个用户为单元的数据块,以便于信息的维护和查询速度的优化。通过 middleuser 表和其他表的关联可以使得用户的下拉输入框查询变得容易处理,减少了前端 javascript 的交互逻辑,并且能够记录好用

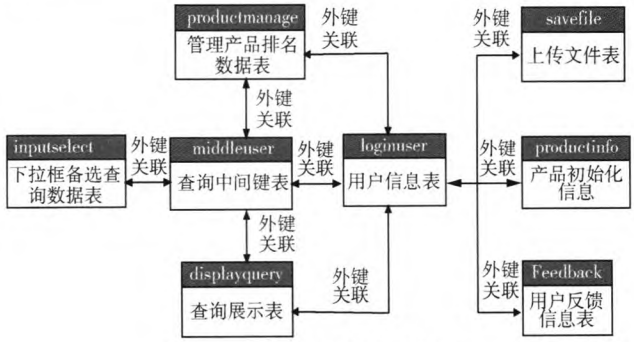

图 3 系统外键关联表关系图

户备选框中已经添加了但还未得到查询结果的商品列表，方便用户下次直接一键查询。

系统的外键关联表关系图如图 3 所示。

2.3 基于 TF-IDF 算法的相似度纠错检测

2.3.1 TF-IDF 算法的原理

TF-IDF（Term Frequency-Inverse Document Frequency）是一种用于信息搜索和信息挖掘的常用加权技术[3]。TF-IDF 模型的主要思想是：用一个具有很强区分能力的词 w 将文章 d 与其他文章区分开来，该词必须具备以下条件：在 d 文章中有很高的出现频率并且该词在其他文档中较少出现。该模型主要包含了两个因素：

(1)词 w 在文档 d 中的词频 TF(Term Frequency)，即词 w 在文档 d 中出现次数 count(w, d)和文档 d 中总词数 size(d)的比值：

$$\mathrm{tf}(w,d) = \mathrm{count}(w,d)/\mathrm{size}(d) \quad (1)$$

(2)词 w 在整个文档集合中的逆向文档频率 idf (Inverse Document Frequency)[6]，即文档总数 n 与词 w 所出现文件数 docs(w,D)比值的对数：

$$\mathrm{idf} = \log(n/\mathrm{docs}(w,D)) \quad (2)$$

查询串 q 与文档 d 的匹配度可以由一个权重表示，该权重是通过 tf-idf 模型为每一个文档 d 和由其关键词 $w[1]\cdots w[k]$ 组成的查询串 q 计算出来的：

$$\begin{aligned}&\mathrm{tf-idf}(q,d)\\&= \mathrm{sum}\{i=1..k/\mathrm{tf-idf}(w[i],d)\}\\&= \mathrm{sum}\{i=1..k/\mathrm{tf}(w[i],d) * \mathrm{idf}(w[i])\}\end{aligned} \quad (3)$$

2.3.2 相似度检测的实现

系统利用 Python 自然语言处理中的开源框架 Gensim 可以对文本进行分词，再对分词进行向量化处理并自动提取特征，利用这些向量化特征构建 TF-IDF 算法的模型从而计算出两个文本之间的余弦夹角[7]，夹角越小则相似度越高。按照此原理把用户输入的商品名与该用户对应的店铺所有商品名进行 TF-IDF 算法的相似度对比，对比值放在列表中，取出其最大值，则可得到相似度最大的商品名，实现了用户的纠错检测功能。

3 系统测试

通过上线测试和每天监控商品排名数据的变化情况，发现系统达到了预期效果。后台定时爬虫任务的数据能够保证每天的更新，并且正常稳定运行。数据能够准确地反映真实商品的排名情况，并且能够计算出每天的商品排名变化，通过手动方式查询对比符合真实情况的排名变化结果。系统部分测试效果如图 4 所示。

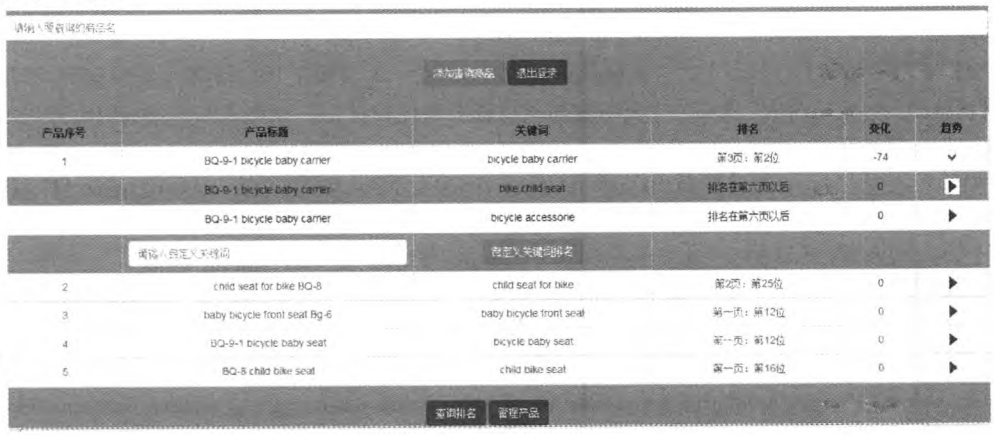

图 4 系统测试样例图

4 结论

通过将商品数据自动抓取下来，并利用 Django 框架开发出一个智能化的商品排名监控系统，能有效监控商品排名及其变化趋势，大大节约了众多店铺商的手工查询时间，帮助他们实现更好的收益。本文利用互联网技术简化了电子商务平台上的繁杂性工作，并把相似度算法应用于用户输入检测，便于输入信息的检索，实现了商务数据监控的智能化。本系统能够对境外电商贸易者提供极大的便利，有很强的应用价值。

参考文献

[1] 柴庆龙，谢刚，陈泽华，等. 基于 Django 框架的故障诊断和安全评估平台[J]. 电子技术应用，2015,43(4):19-21.

[2] 王晓斌，闫果，基于 Django 开发的桥梁健康监控数据查询的 Web 应用[J]. 电子技术与软件工程，2009,24(4):23-24.

[3] XU W, CALLISON-BURCH C, DOLAN W B. SemEval-2015 task 1: Paraphrase and semantic similarity in Twitter (PIT)[C]. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval), 2015.

[4] DETTINGER R D, KOLZ D P, STEVENS R J, et al. Automated data model extension through data crawler approach[P]. US: US8165989, 2012.

[5] SINGHAL N, DIXIT A, SHARMA A K. Design of a priority based frequency regulated incremental crawler[M]. LAP LAMBERT Academic Publishing, 2014.

（下转第 27 页）

满足及时响应通信任务的要求，节省定时器资源，更好地处理相关控制任务，提高软件控制性能。

3 结论

针对当前空间控制器软件任务现状，本文中所述的通过串口中断资源进行任务调度的方案已经应用于多个型号的空间控制器软件任务调度中。该方案充分利用CPU中断资源，避免资源浪费以及由此导致的功能较单一问题，为软件处理更复杂任务调度及控制功能提供了资源，提高了软件响应速度和控制性能，便于拓展系统功能。

参考文献

[1] 颜军. SPARC 嵌入式系统设计与开发[M]. 北京：中国标准出版社，2013.

[2] 宁改娣，杨拴科. DSP 控制器原理及应用[M]. 北京：科学出版社，2002.

[3] 胡乾斌，李光斌，李玲，等. 单片微型计算机原理与应用[M]. 武汉：华中科技大学出版社，2005.

[4] 张少展，张春梅. 基于软件规模的需求优先级排序方法应用[J]. 微型机与应用，2015，34(1)：81-84

[5] 潘灵. RapidIO 高性能通信中间件设计[J]. 电子技术应用，2014，40(12)：107-109.

[6] 饶运涛，邹继军，郑勇芸. 现场总线 CAN 原理与应用技术[M]. 北京：北京航空航天大学出版社，2004.

[7] Data Device Corp. ACE/Mini-ACE Series BC/RT/MT advanced communication engine intergrated 1553 teminal user's guide [Z]. New York: Data Device Corp. 2005.

[8] 康晓军，王劲强，王芸. 基于扩展块的星载软件控制流容错评价方法[J]. 航天返回与遥感，2007，28(3)：33-39.

(收稿日期：2016-02-25)

作者简介：

张新玉(1983－)，男，工程师，主要研究方向：空间控制器软件设计。

(上接第 23 页)

[6] ROUL R K, DEVANAND O R, SAHAY S K. Web document clustering and ranking using TF-IDF based Apriori Approach[J]. arXiv Preprint arXiv，2014，10(1)：55-56.

[7] 申剑博. 改进的 TF-IDF 中文本特征词加权算法研究[J]. 软件导刊，2015，32(4)：16-18.

(收稿日期：2016-01-28)

作者简介：

夏志富(1992－)，男，硕士研究生，主要研究方向：云计算与智能信息处理。

王晗璐(1992－)，女，硕士研究生，主要研究方向：机器学习与智能信息处理。

李玉平(1990－)，男，硕士研究生，主要研究方向：脑电信号与睡眠数据研究。