# 《旅游景点热度分析系统》检测报告

PaperOK

## 总相似率：34.71%

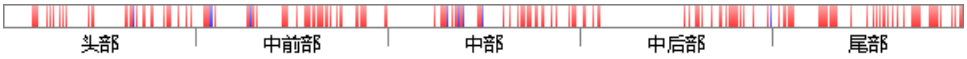
## 基本信息

| | |
|---|---|
| 文档名称 | 旅游景点热度分析系统 |
| 报告编号 | b2f9060c-49f9-4a5c-8ef6-c5638d59b479 |
| 文档字数 | 14321 |
| 提交人姓名 | 王俊 |
| 提交方式 | 粘贴文本检测 |
| 检测范围 | 学位论文库（含硕博）、学术期刊库、会议论文库、法律法规库、互联网资源库、自建比对库 |
| 提交时间 | 2018-05-24 09:26:02 |

## 检测报告指标详情

| 原创率 | 抄袭率 | 引用率 | 字数统计 | 参考文献字数 |
|---|---|---|---|---|
| 65.29% | 32.85% | 1.86% | 14321 | - |

## 相似片段位置图




注：红色部分为重度相似，橙色部分为中度相似，蓝色部分为引用部分

## 相似片段详情（仅显示前10条）

| 序号 | 篇名 | 来源 | 命中率 |
|---|---|---|---|
| 1 | 面向生态工业园区的企业能源综合管理系统部分设计与实现 | 学术期刊库 | 1.05% |
| 2 | 基于iOS的游乐园地图导览系统设计与实现 | 学位论文库 | 1.05% |
| 3 | 闵行招商中心综合业务平台的研究与开发 | 学位论文库 | 0.94% |
| 4 | 客户与工程师在线交流系统的设计与实现 | 学位论文库 | 0.94% |
| 5 | 洪都集团工装管理信息系统的面向对象设计与实现 | 学位论文库 | 0.93% |
| 6 | 在人工智能时代,为什么很多人都看好Python的发展?_搜狐教育_搜狐网 | 互联网资源库 | 0.78% |
| 7 | python调用百度地图API实现经纬度换算、热力地图全流程指南 - 知乎 | 互联网资源库 | 0.78% |
| 8 | 江苏省国税系统综合数据平台的设计和实现 | 学位论文库 | 0.78% |
| 9 | Python中使用Beautiful Soup库的超详细教程,pythonsoup | 互联网资源库 | 0.78% |
| 10 | 个性化混合推荐算法在旅游中的应用 | 学位论文库 | 0.55% |

## 文档原文标注

第二章 系统分析

2.1 问题定义

旅游景点热度分析系统利用网络上有关旅游景点详细的数据如最近的月销量、评论次数、游客评分等数据，以及用户对景点的操作行为数据，通过可视化的方式展现出景点的热度；同时根据景点的热度、用户的兴趣特点以及他近期的用户行为向他推荐可能感兴趣的景点。

2.2需求分析

2.1.1功能需求

旅游景点热度分析系统的主要功能包括以下内容：

功能1：游客注册。

功能2：注册用户登录。

功能3：登陆用户修改个人信息。

功能4：游客或登陆用户搜索景点。

功能5：游客或登陆用户根据景点类别查看景点。

功能6：游客或登陆用户查看景点详细信息。

功能7：登陆用户对景点进行收藏或者取消收藏。

功能8：登陆用户获取景点的热度分析图。

功能9：记录登陆用户搜索行为。

功能10：记录登陆用户点击行为。

功能11：记录登陆用户收藏或者取消收藏景点行为。

图2.1为旅游景点热度分析系统的用例图，主要用例的详细描述如下：

（1）用户注册：

用例名：用户注册

事件流: 用户浏览网站时进行注册

（1）基本流：用户输入手机号码、密码进行注册。输入完成后提交表单。然后注册成功，跳转个人信息页面。

（2）备选流：用户输入手机号码、密码进行注册。输入完成后提交表单。注册失败，跳转注册页面，提醒用户手机号码已被注册。

前提条件:用户没有注册。

后置条件：提醒用户是否注册成功。

（2）注册用户登陆：

同用户注册类似，在此不再赘述。

（3）登陆用户修改个人信息：

用例名：登陆用户修改个人信息

事件流: 用户登陆网站后进行用户个人信息的完善修改

（1）基本流：用户输入姓名、生日、手机号码、住址、密码进行修改。输入完成后提交表单。然后修改成功，跳转个人信息页面。

（2）备选流：用户输入姓名、生日、手机号码、住址、密码进行修改。输入完成后提交表单。修改失败，跳转个人信息页面，提示出错信息。

前提条件:用户已经登陆。

后置条件：提醒用户是否修改成功。

（4）游客或登陆用户搜索景点：

用例名：游客或登陆用户搜索景点

事件流: 用户登陆网站后按需求进行景点搜索

（1）基本流：用户输入进景点关键词或景点名称等，将搜索结果以列表分页的形式返回给用户。

（2）备选流：用户输入进景点关键词或景点名称等，无搜索结果，并提醒用户。

前提条件:用户搜索框中输入文本。

后置条件：提醒用户是否有搜索结果，若有结果，将结果返回给用户。

（5）游客或登陆用户根据景点类别查看景点：

用例名：游客或登陆用户根据景点类别查看景点

事件流: 用户在菜单中选择景点类别，将该类别景点返回给用户

基本流：用户在菜单中选择景点类别，将该类别景点返回给用户。

前提条件:用户已经登陆。

后置条件：该类别景点返回给用户。

（6）游客或登陆用户查看景点详细信息：

用例名：游客或登陆用户查看景点详细信息

事件流:用户点击景点列表中的景点，查看景点详细信息

基本流：用户点击景点列表中的景点，查看景点详细信息。

前提条件:用户已经登陆。

后置条件：提醒用户是否修改成功。

（7）登陆用户对景点进行收藏或者取消收藏：

用例名：登陆用户对景点进行收藏或者取消收藏

事件流: 用户查看景点详细信息时，可对景点进行收藏，或者对已经收藏的景点取消收藏。

（1）基本流： 用户查看景点详细信息时，对景点进行收藏，提醒用户收藏成功。

（2）备选流： 用户查看景点详细信息时，对景点进行收藏，收藏失败，提醒用户已收藏该景点。

前提条件:用户已经登陆。

后置条件：提醒用户是否收藏成功。

（8）登陆用户获取景点的热度分析图：

用例名：登陆用户获取景点的热度分析图

事件流: 登陆用户按自己需求获取景点的热度分析图

基本流：用户在菜单中选择“热度分析”这一按钮，用户可以根据自己需求查看景点热度分析图，返回用户。

前提条件:用户在菜单中选择“热度分析”。

后置条件：将景点热度分析图返回给用户。

（9）记录登陆用户搜索行为：

用例名：记录登陆用户搜索行为

事件流: 用户登陆网站后在搜索框中进行搜索时，将搜索词以及当前时间记录到数据库中

基本流：用户登陆网站后在搜索框中进行搜索时，将搜索词以及当前时间记录到数据库中。

前提条件:用户已经登陆。

后置条件：提醒用户是否修改成功。

（10）记录登陆用户点击行为：

同用例（9）类似，在此不再赘述。

（11）记录登陆用户收藏或者取消收藏景点行为：

同用例（9）类似，在此不再赘述。

图2.1 旅游景点热度分析系统用例图

2.1.2非功能需求

.1 实现技术

推荐系统是自动联系用户和物品的一种工具，它能够在信息过载的环境中帮助用户发现令他们感兴趣的信息，也能将信息推送给对它们感兴趣的用户。个性化推荐系统需要依赖用户的行为数据。个性化推荐系统通过分析大量用户行为日志，给不同用户提供不同的个性化页面展示。几乎所有的推荐系统都是由前台的展示页面、后台的日志系统以及推荐算法系统这三部分构成的[2]。本系统的推荐算法主要是基于景点内容的推荐。用户的首页推荐由用户属性和用户本身行为日志从两个方面决定，对收藏的景点推荐一些标签相似的景点。若是没有登录的用户则将系统分析出的景点热度较高的景点推荐给用户。

.2 用户友好性

尽量减少用户的输入动作。当输入条件较多时，考虑输入条件的各种组合都可能出现，并针对所有组合写出方法。如在用户完善个人信息时，对于有些输入框没有输入时，此时传入的空字符串也要处理，使这段空字符串不会对用户个人信息产生影响。

当用户的操作错误时，给出适当的提示信息，等待用户再次输入。如密码输入错误时，给出弹窗提示后，请求用户再次输入。

2.2数据分析

2.2.1 数据库概念设计[1、简单加几句话说明概念模型是怎么回事

2、给出CDM图

3、进一步结合CDM图进行说明，说明内容如下：本系统中共有两个数据实体：用户和景点，系统一方面将通过用户对景点的操作来进行个性化推荐，另一方面利用景点实体的属性值开展景点热度的定量计算.....]

概念结构设计就是将需求分析得到的用户需求抽象为信息结构（即概念模型）的过程。将需求分析阶段所得到应用需求首先抽象为信息世界的结构，这样才能更准确的用数据库关系系统实现这些需求。概念模型是各种数据模型的共同基础。

图2.1是旅游景点热度分析系统cdm图，本系统中共有两个数据实体：用户和景点。系统一方面将通过用户对景点的操作来进行个性化推荐，另一方面利用景点实体的属性值开展景点热度的定量计算，得出影响景点热度的因素。用户与景点是对多对的关系。即一个用户可以收藏多个景点，一个景点可以被多个用户收藏。

旅游景点热度分析系统cdm图：

图2.1[给出图题] 旅游景点热度分析系统cdm图

旅游景点热度分析系统pdm图：

图2.2 旅游景点热度分析系统pdm图

图2.2是旅游景点热度分析系统pdm图。与图2.1比较，pdm图将用户与景点的实体关系用“行为”表来表示，用此实现景点的被收藏数和被点击数。此外增加了“搜索”表来表示用户搜索记录

，实现用户搜索历史的功能。

2.2.2 景点信息的获取

由于本系统需要大量的数据实现景点热度的分析，所以如何获取大量数据是首先必须解决的部分。同时随着网络的迅速发展，所有的疑问几乎都可以在网络上找到答案，但是如何有效地获取信息是一个难题。而百度搜索等搜索引擎又具有一定的局限性，因为搜索的信息可能有用户不关心的内容，甚至搜索出的结果不一定是正确的。为了解决上述问题，定向抓取相关网页资源的网络爬虫应运而生。网络爬虫（又被称为网页蜘蛛，网络机器人），是一种按照一定的规则，自动地抓取万维网信息的程序或者脚本。它们被广泛用于互联网搜索引擎或其他类似网站，以获取或更新这些网站的内容和检索方式。它们可以自动采集所有其能够访问到的页面内容，以供搜索引擎做进一步处理（分检整理下载的页面），而使得用户能更快的检索到他们需要的信息[10]。

对于国内有很多比较大型的旅游网站，如携程、去哪儿。本系统主要爬取去哪儿网的景点数据作为参考，如图2.3、2.4、2.5、2.6所示在浏览去哪儿网页后可以获取的信息有景点[可以放一张截图上去，并且做出标记]的以下属性：景点名称、景点级别、景点类别、所在地点、开放时间、游客评分、评论次数、景点图片、月销量和景点介绍。

图2.3 去哪儿网景点详细页面图1

图2.4 去哪儿网景点详细页面图 2

图2.5 去哪儿网景点详细页面图3

图2.6 去哪儿网景点详细页面图4

本系统是对国内较为热门的景点热度的分析，景点信息的获取是以“热门景点”为关键词，并按四个类别获取景点数据，在处理后总共是一千四百三十七个景点信息数据。四个类别分别选了文化古迹（569）、自然风光（443）、主题乐园（382）、宗教文化（175）作为景点的类别，因为这四个类别符合时下用户的旅游主题。

2.2.3 数据预处理

查看景点信息的csv文件中，发现了以下两个问题：

(1)数值与字符混杂问题：数据的评论次数含有汉字，即如同这样的形式“全部（9874）”，本文选用正则表达式来处理这一情况。即正则表达式comment = re.compile(r‘\d(\d)+’)。“\d”在python中表示[0-9]的数字。“+”表示“前面一个或多个字符”。“(\d)+”表示只是出现一次或多次出现数字。由于python正则表达默认为贪心匹配，即匹配最长的字符串。于是用comment = re.compile(r‘\d(\d)+’)来获取评论次数超过十的景点，评论次数低于十的则舍弃了，在获取宗教文化类别的景点时发现数量较少，于是对宗教文化类别的景点并无此要求。

（2）景点的多分类问题：由于本文爬去数据时采用的分类爬取的方式，而去哪儿网站的景点存在多分类问题，比如文化古迹类别的景点信息文件中一个景点可能在其它类别出现，[可以更加具体到一个实例]这样直接导致了一个景点对应多条数据的情况，为了便于数据处理，本文将对多条数据进行合并，因此需要对爬取的数据进行去冗余，本文采用了简单遍历的方式解决此问题。

2.2.4 用户

为了用户更加良好的浏览体验和个性化服务，本系统在用户信息管理中增加了有关地址、生日等属性，使得在基于用户的景点推荐中可根据用户的地址与年龄推荐一些用户可能比较喜欢的景点，这部分用户个人信息将保留在用户基本信息表中。

除了使用用户的基本信息进行个性化推荐外，本系统还通过分析大量用户行为日志，给不同用户提供不同的个性化页面展示。所以系统用两个表来记录用户的用户行为。Search表记录用户搜索记录，可以实现用户的搜索历史功能。Action表记录用户点击、搜索景点的行为，这样可以更好的了解用户可能感兴趣的景点，同时也可以统计景点的被点击数与被收藏数量，用户行为数据能在一定程度上也反映出景点的热度，因此本文在后续的景点热度分析中将利用这些数据进行定量分析。

2.3 本章小结

本章介绍了系统开发的前期工作。对系统进行了需求分析后，记录了系统的基本功能需求，同时根据功能需求开展了数据分析，也对景点信息的获取与数据处理做了详细的介绍。

第三[另起一页]章 系统设计

3.1总体设计

3.1.1 体系结构

本系统采用B/S结构（Browser/Server，浏览器/服务器模式），它是WEB兴起后的一种网络结构模式，WEB浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，如谷歌浏览器、360浏览器等，服务器安装SQL Server、Oracle、MYSQL等数据库。本系统使用的是MYSQL数据库。因为它支持支持AIX、FreeBSD、HP-UX、Linux、Mac OS、Novell Netware、OpenBSD、OS/2 Wrap、Solaris、Windows等多种操作系统，同时也为多种编程语言提供了API。这些编程语言包括C、C++、Python、Java、Perl、PHP、Eiffel、Ruby和Tcl等。

3.1.2 Java Web的开发模式

本系统用JSP+Servlet+JavaBean+Dao开发模式。因为DAO模式实现了把数据库表的操作转化成对Java类的操作，提高了程序的可读性，并实现更改数据库的方便性。

在系统设计中，采用DAO的模式主要优点：

(1) 抽象出数据访问方式，在访问数据库时，完全感觉不到数据库的存在。

(2) 将数据访问集中在独立的一层，所有数据访问都由DAO代理，从而将数据访问的实现与系统的其余部分剥离。

[有关实现的问题都放在第四章]

3.2模块[3.2给出一个功能图之类的图解

3.3.数据爬取模块

3.4热度分析模块

3．5推荐模块

——这些模块可以先用简单文字描述，然后加上一个流程图或者伪码表示

3.6用户界面设计]设计

图3.1 系统功能结构图

3.3 数据爬取模块

如图3.1，网络爬虫基本过程如下：

（1）首先选取种子URL；

（2）将这些URL放入待抓取URL队列；

（3）从待抓取 URL队列中取出URL，将URL对应的网页下载下来，将下载下来的网页传给数据解析模块，再将这些URL放进已抓取URL队列。

（4分析下载模块传过来的网页数据，通过html解析，具体实现见第四章。 提取出需要的数据，并将数据写入csv文件中。

（5）URL调度模块接收到数据解析模块传递过来的URL数据，将这些URL数据放入待爬取URL队列中。

（6） 整个系统一直循环，直到待抓取URL队列里所有的URL已经完全抓取，或者系统无法继续爬取（网站限制访问），循环结束。

图3.2 数据爬取模块流程图

3.4 热度分析模块

3.4.1 景点数据

热度算法设计：

输入：景点评论次数、景点游客评分

输出：景点热度分值

分析：对分析景点热度有帮助的景点数据：月销量、评论次数（commet）、游客评分（grade）、景点的位置信息、经纬度、景点等级、景点所在城市、景点类别。由于月销量每天都在变化，并且每月更新一次。最重要的是免费的旅游景点是没有月销量的，所以并未将月销量作为分值计算放入热度算法中。单独对这一数据进行了景点热度的分析。在仅考虑评论次数、游客评分的情况下得出①式。

Hot = commet / 100 + grade * 10

3.4.2 用户行为因素

热度算法设计

输入：景点评论次数、景点游客评分、景点被点击次数、景点被收藏次数

输出：景点热度分值

分析：在考虑用户行为日志之后，也可以得到用户的被点击次数（clickNum、以一万为上限）与被收藏次数（collectNum、以一千为上限）。将这些数据结合在一起，可以更好的反映景点的热度。于是在查看这些数据之后（由于本系统用户暂时为零，所以景点的被点击数和被收藏数也为零），发现评论次数的数据范围在0到10000之间，于是取一半，高于5000的按5000计算，游客评

分集中在4.7至4.9这个分段（5.0满分）。综上对于景点的热度分析，热度计算公式如下(满分一百)。在计算热度时，将这些属性放在同一高度上进行比较。综合考虑这些因素，尽量得出较为客观的景点热度，热度算法如式②。

Hot = (commet / 100 + grade * 10 + clickNum / 200 + collectNum / 20 ) / 2 ②

3.4.3 景点其他因素

热度分析：

输入：景点热度、景点类别、所在城市、景点级别

输出：用echarts和百度地图API实现的图表分析

上式热度的计算并未将景点的类别、所在城市以及景点级别作为影响景点热度的因素。所以有必要将热度数值分别与以上三个数据进行对比。首先对景点数据进行筛选，筛选条件为大于等于①式得到的热度数值的平均数，因为系统暂时没有用户行为。

3.5 景点推荐模块

3.5.1 景点关键词的提取

为了实现用户搜索的良好体验，有必要将景点介绍的文本提取关键词以便用户的搜索。关键词抽取模型有以下几个算法：TF-IDF算法、TextRank算法还有基于语义的统计语言模型。对于此系统，在阅读相关文献后决定采用TF-IDF算法实现关键词的提取。TF-IDF算法是关键词提取算法中基础并且有效的一种算法，实现简单，并且效果显著。其主要思想是：如果某个词或短语在一篇文章中出现的频率高，并且在其他文章中很少出现，即反文档频率低，则认为此词或者短语具有很好的类别区分能力，适合用来分类。那么对于这篇文章来说，这个词也就可以算作该文章的一个关键性词语[3]。又因为对于景点的描述文档的篇幅限制，所以提取六个关键词作为该景点的标签。

信息提取即为信息抽取，是把文本里包含的信息进行结构化处理，变成表格一样的组织形式。在信息抽取系统中，输入的是原始文本，输出的是固定格式的信心点。这些被抽取的信息点以统一的形式集成在一起。这就是信息抽取的主要任务[4]。

提取关键词整体流程：文本、系统参数输入；分词、过滤停用词；单个词的权重计算、排序；提取文本关键词[4]。查看处理结果时，提取的关键词有时候可能无法体现景点的特征。比如景点描述的文本中可能会含有该景点的面积介绍，于是关键词中往往会出现“三十二”、“54”等数词；除此之外，提取的关键词中往往也会有人名的出现，然而这个人名又不能体现该景点的特征，于是抱着宁缺毋滥的态度，对于关键词的词性做了很大的限制，只允许出现地名、名词、其他专名、形容词和名形词（具有名词功能的形容词）。

3.5.2 推荐算法设计

推荐算法设计

输入：景点id

输出：景点列表

分析：本系统推荐的算法是基于景点内容的推荐。这种工作原理是，评估用户还没看到的景点与当前用户过去收藏的景点的相似程度。在实际情况下，相似度可以由不同方法衡量。举例来说，典型相似度度量方法会用到Dice系数，它比较适合多值特征集合[17]。该系数描述如下：如果每个景

点Si有一组关键词keywords（Si）描述，那么Dice系统计算景点Si和Sj之间相似度为：

Dice（keywords(Si),keywords(Sj)）

=2*comm(keywords(Si),keywords(Sj)/(leng(keywords（Si）)+leng(keywords（Sj）))。其中，comm(keywords(Si),keywords(Sj)是两个关键词组中相同关键词的个数。leng(keywords(Si))，leng(keywords(Sj))是两个关键词组的长度。

在系统推荐内容中，首先判断用户有没有登录。若是没有登录或者已登陆而收藏列表为空，就将热度最高的景点推荐给用户；若用户已经登录并且已经收藏景点列表中有景点，则根据用户收藏的景点的类别，将其同类别的景点与收藏景点进行上述算法。将Dice系数最高的，即相似程度最高的三个景点推荐给用户。一个收藏景点返回三个景点结果。若用户有多个收藏景点，将最近收藏的两个景点作为测试数据，并返回结果。

图3.3 推荐算法流程图

3.6 用户界面设计

3.4.1 一般交互

（1）保持一致性。为人机界面中的菜单选择、命令输入、数据显示以及其它功能，使用一致的格式。

（2）提供有意义的反馈。向用户提供视觉反馈，保证用户和系统之间建立双向联系。

（3）在执行有较大破坏性的动作之前要求用户确认。如果用户要终止程序的运行，需给出提示信息以请求用户确认他的命令。

3.4.2 信息显示

只显示与当前工作内容有关的信息。用户在获得有关系统的特定功能的信息时，不必看到与之无关的数据、菜单和图形。

用图形或图表来取代庞大的表格，以便于用户迅速吸取信息，尽量避免用数据淹没用户。如景点热度分布图的绘制可以让用户很快了解到景点热度相关的信息。

产生有意义的出错信息。如在登陆失败时弹窗提示用户“用户名或密码错误。”

3.4.3 数据输入

（1）减少用户输入动作。如用户搜索历史记录，并可以按热度、类别搜索景点。

（2）使在当前动作语境中不适用的命令不起作用。这可以使用户不去做哪些肯定会导致错误的动作。如用户在未登录情况下无法收藏景点。

3.5 本章小结

本章详细介绍了旅游景点热度分析系统的总体架构与主要模块的设计思路，主要对数据爬取模块、热度分析模块、景点推荐模块进行了详细的叙述并做出分析。数据爬取模块对网络爬虫的流程进行了叙述。热度分析模块则根据景点数据与用户行为，将景点热度量化，同时根据景点热度的量化值与景点其他因素进行分析，并用可视化的方式展示。景点推荐模块则对推荐算法进行了介绍，并给出流程图。

第四章 系统实现

4.1 开发平台

操作系统：windows10

数据库管理系统： mysql

（1）开发环境： pycharm（用于获取景点信息，并将清洗后的数据写入mysql数据库）、程序设计语言： python3.6

（2）主要开发环境：eclipse（系统实现）、程序设计语言：java

服务器：tomcat 6.0

系统在eclipse上开发，Eclipse 是一个开放源代码的、基于Java的可扩展开发平台。就其本身而言，它只是一个框架和一组服务，用于通过插件组件构建开发环境。Eclipse 附带了一个标准的插件集，包括Java开发工具（Java Development Kit，JDK）。对于本系统的设计语言，由于本人对java更加熟悉，而且对于java在网络应用程序开发上也有一定的经验。而java的相关特性也适合作为网络开发，并且时下java也是非常流行的一种语言。所以选择了java作为主要的程序设计语言。简单说一下java的几个特性：

（1）适用性强，一般来说几乎所有浏览器都支持java。

（2）java web可维护性强，具有开放性，可以增加模块而无需改变许多代码，只需要接入预留的接口就可以。

（3）可重用性，代码复用率多，无需写许多代码。

（4）可移植性很好.Java中对基本数据结构类型的大小和算法都有严格的规定，这样也有利于在编译时发现程序错误。

4.2 数据库的建立

pdm图（图2.2）中描绘的每个实体可映射为表格。实体中属性可映射为表格中列。

表4.1 用户基本信息表

表序号 1 表名 user

含义 用户的基本信息

序号 属性名称 含义 数据类型 长度 说明 约束

1 user_id 账号 int 11 not null 主键

2 name 用户名 varchar 20

3 ps 用户密码 varchar 20 not null

4 birth 用户生日 varchar 10

5 sex 性别 char 2

6 phone 手机号码 char 11 not null

7 address 地址 varchar 20

表4.2 景点基本信息表

表序号 2 表名 scenery

含义 景点基本信息

序号 属性名称 含义 数据类型 长度 说明 约束

1 s_id 景点id int 11 not null 主键

2 s_name 景点名称 varchar 1000 not null

3 level 景点级别 varchar 10

4 place 地点 varchar 20 not null

5 open 开放时间 varchar 50 not null

6 introduction 景点介绍 varchar 11 not null

7 comment 评论次数 int 11 not null

8 grade 游客评分 float 11 not null

9 keyword 关键词 varchar 100 not null

10 hotnum 热度 float not null

11 kind 景点类别 varchar 50 not null

12 lng 经度 float not null

13 lat 纬度 float not null

14 picture 景点图片 varchar 500 not null

表4.3 用户搜索信息表

表序号 3 表名 search

含义 用户搜索信息

序号 属性名称 含义 数据类型 长度 说明 约束

1 user_id 用户id int 11 not null 主键、外键

2 sTime 搜索时间 datatime not null 主键、外键

3 searchWord 搜索词 varchar 50 not null

表4.4 用户点击/收藏行为信息表

表序号 4 表名 action

含义 用户行为信息

序号 属性名称 含义 数据类型 长度 说明 约束

1 user_id 用户id int 11 not null 主键、外键

2 time 行为时间 datatime not null 主键、外键

3 s_id 景点id int 11 not null 外键

4 whatdo 事件 char 20 not null

备注：action表中的whatdo（事件）取值只有两种，分别为“点击”、“收藏”。

4.3 程序设计风格[注意复制比]

4.3.1 提高可重用性

系统使用一下方法提高可重用性。

保持函数的一致性。对功能相似的方法有相同的名字特征、参数特征、函数类型及出错结果。

全面覆盖。如果带个页面有多个输入框且输入条件的各种组合都可能出现，则针对所有组合写出方法。如在用户完善个人信息时，对于信息传入的空值也要处理。此外，函数不应该只处理正常值，对空值、极限值及界外值等异常情况也应该有所处理。

利用继承机制。本系统对连接数据库的代码进行分离，在其他函数在访问数据库时，调用此方法就可以实现数据库的连接。

4.3.2 提高健壮性

对于每一个接收用户输入数据的函数，都对接收的数据进行检查，并在发生错误时，给出提示信息，并等待用户再次输入。如密码输入错误时，给出弹窗提示后，请求用户再次输入。

检查参数的合法性。对于函数参数的合法性进行检查，因为用户在使用公有函数可能会违反参数约束条件。

4.4 类[依次与第三章设计相对应逐个实现模块]的实现

4.4.1 系统总体介绍

系统总体实现基本如下：jsp写前台代码，主要设计七个页面：登陆/注册，显示/修改个人信息、首页、景点总览、类别景点列表页面、景点详情页面、景点分析页面。Servlet响应web方面的请求，处理前台传来的数据，并实现页面的跳转。为数据库中的每一个表做一个bean文件，分别为景点

（scenery）、用户（user）、搜索（search）、收藏/点击行为（action）。为这四个表也做了四个DAO接口，接口中写servlet中需要用到的数据处理方法，并写四个类实现这四个接口。图4.1为系统的功能结构图。

图4.1 系统功能结构图

4.4.2 数据爬取模块

系统使用python实现网络爬虫。因为python，相对于java，C++等其他静态编程语言，抓取网页文档的接口更加简洁；相比其他动态脚本语言，如perl等，python的urllib2包提供了较为完整的访问网页文档的API。另外网页抓取后的处理，python的beautiful soup（Beautiful Soup 是一个可以从HTML或XML文件中提取数据的Python库）提供了简洁的文档处理功能，能用极短的几行代码实现大部分的文档处理。总而言之，python更加简洁。所以本系统利用python实现网络爬虫相关技术，获取旅游景点的详细信息。[爬虫的实现可以放在实现中，而且可以针对携程、去哪儿网站进行分析]

本系统爬取去哪儿网的景点数据作为参考，在浏览去哪儿网页后可以获取的信息有景点[可以放一张截图上去，并且做出标记]的以下属性：景点名称、景点级别、景点类别、所在地点、开放时间、游客评分、评论次数、景点图片、月销量和景点介绍。具体流程如下：url=“http://piao.qunar.com/ticket/list.htm?keyword=热门景点&region=&from=mpl_search_suggest&subject=文化古迹&page=1”，使用urlib库请求网页，request从因特网上下载文件和网页；Beautiful Soup解析html，使用beautiful soup内置的find_all()方法和find（）方法实现了对景点详细信息的获取。然后将数据写入CSV文件中。调用open()以写模式打开一个文件，然后将它传递给csv.writer(),创建Writer对象。Writer对象的writerow()方法接受一个列表参数[12]。每下载解析一次页面就将获取的景点信息放入列表参数中，写入CSV文件直到循环结束。

由于要实现景点的热度分布情况，了解景点的经纬度也是不可或缺的。而在经纬度的获取上，百度地图API提供了这样一个百度经纬度API，http://api.map.baidu.com/geocoder/v2/?address=地址&output=json&ak=百度密钥，修改网址里的“地址”和“百度密钥”，在浏览器打开，就可以看到经纬度的json信息。在百度注册并获取了百度密钥之后，通过百度经纬度API将利用爬虫获取的景点位置信息放到url的“地址”上，再将获得的网页用beautiful soup的find（‘lng’）方法与find（‘lat’）分别获取景点的经纬度。

4.4.3 热度分析模块

按①式获得的热度量化值作为景点属性写入数据库中。运用echarts和百度地图API展示景点热度与所在城市的发达程度、景点级别与景点类别的关系。

根据Hot = (commet / 100 + grade * 10 + clickNum / 200 + collectNum / 20）/ 2 ，计算旅游景点的热度。计算出的结果用触发器更新景点热度。

根据图4.6，UActionDAO.java类中findClickNum（int s_id ）方法用于查询id为s_id景点的被点击数；findCollectNum（int s_id ）用户记录此景点的被收藏数。SceneryDAO.java类中find（int s_id）方法用于用户查看景点详细信息，包括comment和grade。

.1 Echarts

ECharts，是一个使用 JavaScript 实现的开源可视化库，可以流畅的运行在 PC 和移动设备上，兼容当前绝大部分浏览器（IE8/9/10/11，Chrome，Firefox，Safari等），底层依赖轻量级的矢量图形库 ZRender，提供直观，交互丰富，可高度个性化定制的数据可视化图表。其拥有丰富的可视化类型，提供了常规的折线图、柱状图、散点图、饼图、K线图，用于统计的盒形图，用于地理数据可视化的地图、热力图、线图，用于关系数据可视化的关系图、treemap、旭日图，多维数据

可视化的平行坐标，还有用于 BI 的漏斗图，仪表盘，并且支持图与图之间的混搭。在处理景点数据之后，可以通过echarts较为直观的看出景点的分布情况。根据系统获取的景点数据，在经过echarts后，可以很轻松的看到拥有热门景点数量最多的热门城市、以及热门景点的销量对比等。

图4.1 热门景点四、五月销量对比

由于月销量每天都在变化，并且每月更新一次。最重要的是免费的旅游景点是没有月销量的，所以并未将月销量作为分值计算放入热度算法中。单独对这一数据进行了景点热度的分析。图4.1则是去哪儿网景点四月份最高月销量与五月份最高月销量对比图，同理也可得出评论次数最多的景点等。

图4.2 景点类别占比图

图4.3景点类别级别关系图

图4.4 拥有热门景点最多的城市

热度的计算并未将景点的类别、所在城市以及景点级别作为影响景点热度的因素。所以有必要将热度数值分别与以上三个数据进行对比。首先对景点数据进行筛选，筛选条件为大于等于①式得到的热度数值的平均数。实验分析图如下。

4.2图为热门景点与四个类别分布情况，图片结果显示游客更喜欢自然风光的景点，占比35.3%。主题乐园与历史文化分别占29.39%、29.21%。图4.3显示的是四个类别与景点级别的关系，很容易看出除主题乐园意外，景点级别对景点热度分值还是有一定影响的。5A景区与4A景区更受游客的欢迎。这也比较符合实际，主题乐园的热度与其自身景点级别无太大关系。图4.4显示为拥有热门景点最多的前十个省份。结果浙江、江苏两个省份拥有热门景点最多的城市，而北京排行第九；上海排行二十二，仅有17个热门景点。说明热门景点的分布与城市的发达程度无关。

.2 百度地图API

Echarts官方说明：ECharts 之前提供下载的矢量地图数据来自第三方，由于部分数据不符合国家《测绘法》规定，目前暂时停止下载服务。所以系统使用百度地图API实现热门景点在地图上的分布。系统在获取了景点的经纬度之后，使用这个API实现了热门景点在中国地图的分布情况。下图展示的是去哪儿网景点四月份月销量。根据地图可以很容易看出北京的故宫、上海的迪士尼乐园还是很热门的。

图4.5 景点四月份月销量分布图

4.4.4 景点推荐模块

景点推荐模块的实现是使用UActionDAO.java类findCollect（int id）方法为查询账号为id的用户收藏的景点，并返回最近收藏的两个景点id。然后根绝这两个景点id，利用SceneryDAO.java类中Recommend（int s_id）方法。此方法是将景点id传入后，得出此景点的类别，在遍历同类别下景点关键词与此景点关键词的Dice（ ）系数，将相似程度最高的三个景点放入list < scenery > ,并返回list < scenery > 。

类图中省略了servlet类，以com.servlet包名代替。系统将所有的servlet类放在com.serlvet包中。当前jsp页面调用servlet类跳转页面，而serlvet类通过调用其他DAO类实现数据库的交互与页面的跳转。如用户登录时，当前页面提交表单给Loign.java类，然后这个servlet通过调用UserDAO中find（int id）方法，若匹配成功，则当前servlet跳转到首页，若失败则用小窗口提醒用户“用户名或密码错误”。

类图关系：

图4.6 旅游景点热度分析系统类图

4.4.5 其他模块实现

用户管理模块实现了用户注册、登录，修改个人信息的功能。具体实现方式为UserDAO.java类中create（ ）方法用于用户的注册；find（int id）方法用于验证用户登录的id和密码；update（User user）方法用于更新用户的个人信息。

景点查询模块实现了用户搜素景点和用户按类别查询景点。具体是实现方式为SceneryDAO.java类中find（int s_id）方法用于用户查看景点详细界面。findSearch（String search）实现用户按按关键字搜索景点，根据用户搜索的字符串或类别，并返回景点列表。findKind（String kind）方法实现用户按类别搜索景点，并返回该类别景点列表。

4.5 本章小结

本章叙述了系统的实现过程。对开发环境、数据库系统进行说明之外，也详细的介绍了系统各个模块的实现。数据爬取处理获取景点的主要信息外，还通过百度地图API获取了景点的经纬度信息。在热度分析模块中，对景点热度与景点的其他因素进行分析，也实现了根据用户行为更新景点热度。同时也针对景点月销量对热度进行分析。景点推荐模块与其他模块的实现也进行了详细的说明。