

基于Python的天气信息可视化分析系统的设计与实现

于路遥, 宋瑾钰*

浙江理工大学计算机科学与技术学院, 浙江 杭州

收稿日期: 2022年10月17日; 录用日期: 2022年12月20日; 发布日期: 2022年12月28日

摘要

天气预测与分析与日常出行、农业生产、自然灾害预防等多个领域都息息相关, 是现代社会正常运转不可或缺的一个部分。近年来随着网络数据规模的不断增大, 网络爬虫应运而生, 它可快速收集网络数据, 为数据分析和信息可视化奠定了基础。本文设计的天气信息可视化系统, 基于Python网络爬虫技术采集所需要的天气数据, 通过数据分析和可视化技术, 实现了对不同城市 and 不同时间的天气数据多角度、多维度的对比, 同时对历史天气数据进行多类别图表展示, 包括了天气现象占比图、风力风向占比图、空气质量分布图、最高最低温度变化图等, 充分展现了Python在天气数据处理的应用及在可视化分析方面的优越性。

关键词

Python, 网络爬虫, 天气预报, 数据可视化

Design and Implementation of Visualized Analysis System of Weather Information Based on Python

Luyao Yu, Jinyu Song*

School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou Zhejiang

Received: Oct. 17th, 2022; accepted: Dec. 20th, 2022; published: Dec. 28th, 2022

Abstract

Weather information prediction and analysis is closely related to daily travel, agricultural pro-
*通讯作者。

duction, natural disaster prevention and other fields, and is an indispensable part of the daily operation of modern society. In recent years, as the scale of network data continues to increase, web crawlers emerge as the times require. They can quickly collect network data and lay a foundation for data analysis and information visualization. The weather information visualization system designed in this paper is based on Python web crawler technology to collect the weather data needed. Through data analysis and visualization technology, it realizes the comparison of weather data in different cities and at different times from multiple angles. At the same time, it displays historical weather data in multiple categories of charts, including the proportion of weather phenomena, the proportion of wind and direction, the distribution of air quality, the maximum and minimum temperature diagram, etc., which fully demonstrates the advantages of Python in weather data processing and visual analysis.

Keywords

Python, Web Crawlers, Weather Forecast, Data Visualization

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

伴随着网络科学技术的不断发展进步, 天气信息在我们日常的生活中, 占据着越来越重要的地位。气象预报关系到人类生产和生活的方方面面, 也是现代社会的重要组成部分。实时掌握气象资料, 可提前防范诸如暴雨、大风、冰雹等自然灾害, 尽量降低自然灾害对人类的影响[1]。那么, 如何高效地获取并使用天气信息数据就成为了一个巨大的挑战。在信息技术快速发展的今天, 网络爬虫技术受到了越来越多的重视, 利用网络爬行技术可以快速找到自己想要的数据并将其反馈给更多的人, 使用这种技术能够节省更多人力和物力[2]。

国内天气服务做得较好的有墨迹天气、搜狐天气和新浪天气等。墨迹天气 app, 需在智能手机上下载并安装。网页版的搜狐天气和新浪天气中数据简单, 信息不够完整。文献[3]中使用 Scrapy 框架编写天气爬虫, 用 Python 字符切割与 Python 的 Split 技术, 获取目标数据, 并使用基于 Python 的 Web 框架——Django 进行服务器端编程实现, 将清洗后的数据存储到 MySQL 数据库中, 使用 ECharts 在网页上呈现天气数据的可视化图表[3]; 文献[4]中提出将 C#与 Jumony 相结合, 利用网页捕捉技术和网页分析技术获得中国某一城市的天气预报[4]。

基于天气预报的重要性以及网络爬虫技术的优越性, 本文利用 Python 爬虫技术, 从中国天气网、2345 天气王网站上获取需要的气象资料, 并利用 re 规则工具过滤和编辑所获得的天气资料, 通过数据分析和可视化技术, 将全国实时天气信息、城市天气预报、历史天气等进行对比和展示。

2. 相关技术

2.1. 网络爬虫

2.1.1. 基于 Python 的爬虫技术

Python 被广泛地用于爬虫, 它为爬虫提供了许多强大的第三方库和标准库, 利用 Python 可以更方便的进行爬虫编程。Python 的 urllib2 包中包含了可以访问 Web 页面 API, 因此利用 Python 对于网页文档

接口的抓取更简单有效。通过 Python 内置的网络请求库 Requests 和内置正则工具 re 爬取和解析网页。其中 Requests 在 Python 内置模块的基础上进行了高度封装, 完全满足 HTTP 测试需求, 可以对网络上的天气预报和历史天气数据爬取并对数据进行预处理。re 正规运算式是一种用于处理字符和特殊字符的逻辑公式, 可以根据各个城市、时间等参数构建爬取地址, 使用 Requests 做 HTTP 请求, 用正则表达式对请求到的 html 界面进行正则筛选来获取数据。

气象资料具有很强的时效性, 所以需要爬虫对所获取的网页进行持续的更新, 这就要求爬行器对站点进行定期的扫描, 以确定哪些页面需要更新。爬虫基本流程如图 1 所示。

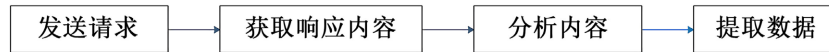


Figure 1. Basic flowchart of crawler
图 1. 爬虫基本流程图

2.1.2. Requests 请求库

Requests 是一种基于 urllib 的 HTTP 库, 它使用了 apache2licensed 的开放源码, 对 Python 的内部模块进行了高度的封装, 代码更加简单[3]。Requests 支持 HTTP 连接保持, 连接池, 支持文件上传和 URL 国际化[5], 支持自动确定响应内容的编码, 它能完全满足当前网络的需求。

Requests 库是一种常见的 HTTP 请求模块, 引入 Requests 库可以方便地进行页面的爬取。

2.1.3. Re 正则工具

在爬虫的开发中, 需要从大量的文本中抽取有用的信息, 而正则表达式就是其中的一种。正则表达式是一种通常用来检索、替换符合某种格式的文字的一种形式, 它会设置特定的文字和文字, 然后用它们的组合来筛选和匹配我们所需要的特定信息[6]。正则表达式常用的匹配模式如表 1。

Table 1. Common matching patterns of regular expressions

表 1. 正则表达式常用的匹配模式

| 模式 | 说明 | 实例 |
|--------|------------------------|---------------------------|
| . | 匹配除了换行符之外的任意字符 | 无 |
| [...] | 表示字符集, 对单个字符给出取值范围 | [a-z]表示匹配字母 a 到 z 的单个字符 |
| [^...] | 表示非字符集, 对单个字符给出排除范围 | [^abc]表示匹配除 a、b、c 之外的单个字符 |
| * | 匹配前一个字符零次或多次 | abc*表示匹配 ab、abc、abcc 等 |
| + | 匹配前一个字符 1 次或无限次 | abc+表示匹配 abc、abcc、abccc 等 |
| ? | 匹配前一个字符 0 次或 1 次 | abc?表示匹配 ab、abc |
| {m} | 匹配前一个字符 m 次 | abc{2}表示匹配 abcc |
| {n, m} | 匹配前一个字符 n 到 m 次, 含 m 次 | abc[1, 2]表示不匹配 abc、abcc |
| a b | 表示匹配 a 或 b | ab cd 表示匹配 ab 或 cd |
| ^ | 匹配字符串开始位置 | ^abc 表示 abc 在一个字符串的开始位置 |
| \$ | 匹配字符串结尾位置 | abc\$表示 abc 在一个字符串的结尾位置 |
| () | 分组标记 | (abc)表示匹配 abc |
| \d | 表示匹配任意数字 | 无 |
| \w | 表示匹配字母字符和下划线 | 无 |

re 库中有所有 Python 的规则运算, 它可以很容易地检测出一个字符串是否符合特定的模式, 以达到获得所需要的数据[7], 它具有很强的灵活性、逻辑性和实用性, 可以快速地从字符串中查找、替换、提取一段有规则的信息, 使用规则运算式可以更有效地执行, 但缺点是规则表达式不太直观, 且构造表达式也很复杂[8]。

我们可以合理的使用正则表达式, 实现城市等关键字的搜索, 进而实现对各个城市各个时间天气信息的爬取, 为后续数据的处理打下基础。

2.2. Flask 框架

Flask 在 2010 年问世, 是 Armin ronachert 利用 Python 的 Werkzeug 工具包编写的轻量级 Web 开发框架[9]。这是一款非常灵活、轻便、安全、易用的服务器架构, 非常适合团队的开发[10]。采用 Flask 技术可以进行前后端的数据交互, 而其前端的 JinJa2 模板引擎也为前端的数据调用提供了便利[11]。

现在有很多 Flask 框架可以很容易地将不同的插件结合起来, 从而满足快速的迭代开发需要。开发者可以结合不同的 Flask 插件, 使其能够满足自身的个性化要求[12]。

2.3. Web 页面开发

2.3.1. Vue 框架

Vue 是目前最受欢迎的前端开发框架, 它是一个友好、高性能和多功能的框架, 可以帮助开发者在前端开发中建立更强大的代码库, 能够实现软件工程项目的高效开发, 并且可以缩短开发者的交付周期, 非常适用于构建用户界面, 能够在浏览器界面上直接进行相关的调试或修改等工作[13]。

Vue 框架核心功能是快速响应的双向数据绑定, 如图 2, 简单来说, Vue 对象的改变将会影响 html 的标记, 而标记的变更将会影响到 Vue 对象的属性。通过双向绑定, 开发者只需关注 json 数据的改变, 将 Vue 和 html 之间的关系映射到 js 对象, 代码变得更加简洁、易于理解。

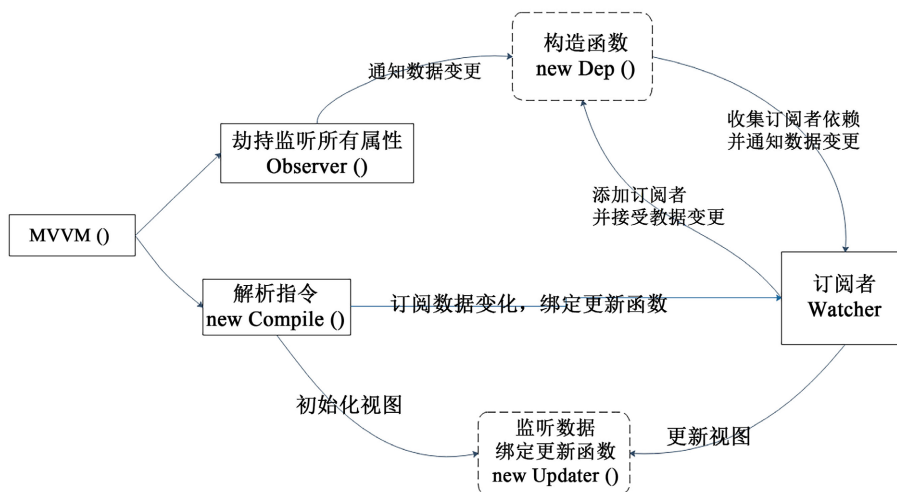


Figure 2. Schematic diagram of bidirectional data binding
图 2. 双向数据绑定的原理图

2.3.2. ECharts 图表库

ECharts 是一款开放源码的数据可视化图表, 遵循 Apache-2.0 开源协议, 拥有大量开源的可视化图表文件, 可以提供用户下载使用[14], 也可以免费商用。它是一款非常优秀的可视化前端框架, 目前兼容绝

大部分浏览器和多种设备, 可以随时随地展示, 拥有各个产品及行业所需要的基础性图表, 提供了常用的折线图、柱状图、散点图、饼图, 以及图形和图形的混搭满足各种需求, 提供直观、生动、可交互、可个性化定制的数据可视化图表[15]。

3. 系统架构与功能模块

3.1. 系统架构

在本项目的开发中, 主要有网络爬虫、数据层、前端、后端四部分, 系统架构如图 3。

网络爬虫是基于 Python 对中国天气网和 2345 天气王网站进行数据爬取。首先根据各个城市、时间等参数构建爬取地址, 导入 Requests 模块代替浏览器做 HTTP 请求, 访问相关信息网站, 对整个页面进行下载。然后使用 re 正则工具对请求到的 html 界面进行正则筛选, 检查字符串是否与某种模式匹配, 来获取所需要的数据。

数据层采用 Python 连接 MySQL 数据库的 PyMySQL 模块对爬虫所采集的数据进行存储。同时还运用了 Python 内置了字典, 字典是另一种可变容器模型, 且可存储任意类型对象, 具有极快的查找速度。

后端使用 Flask 框架处理前端发送的请求, 进行控制器逻辑和业务逻辑的开发, 在处理某个任务时可调用与之对应模块中的函数来处理或计算。在后端可以利用 Flask 框架返回前端网页并替换网页模板值, 同时编写 API 数据接口, 返回处理前台来的请求得到的响应数据。

前端采用 html/Vue/CSS3 开发网页, 与主流浏览器兼容。采用后端 API 接口的方式, 实现了前端和前端的独立开发。

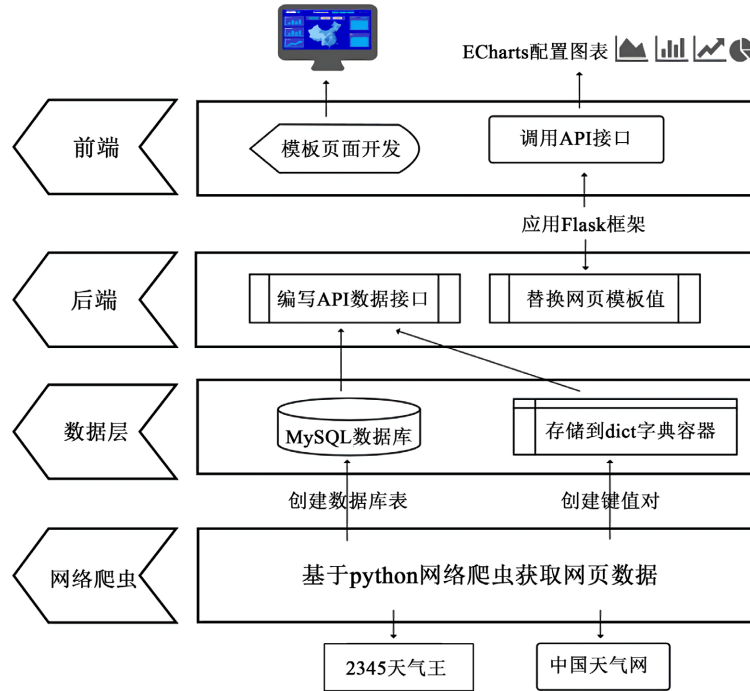


Figure 3. System architecture diagram
图 3. 系统架构图

3.2. 功能模块

本系统主要实现天气信息的可视化, 因此设计了四个模块支持系统的整体功能实现, 如图 4 所示。

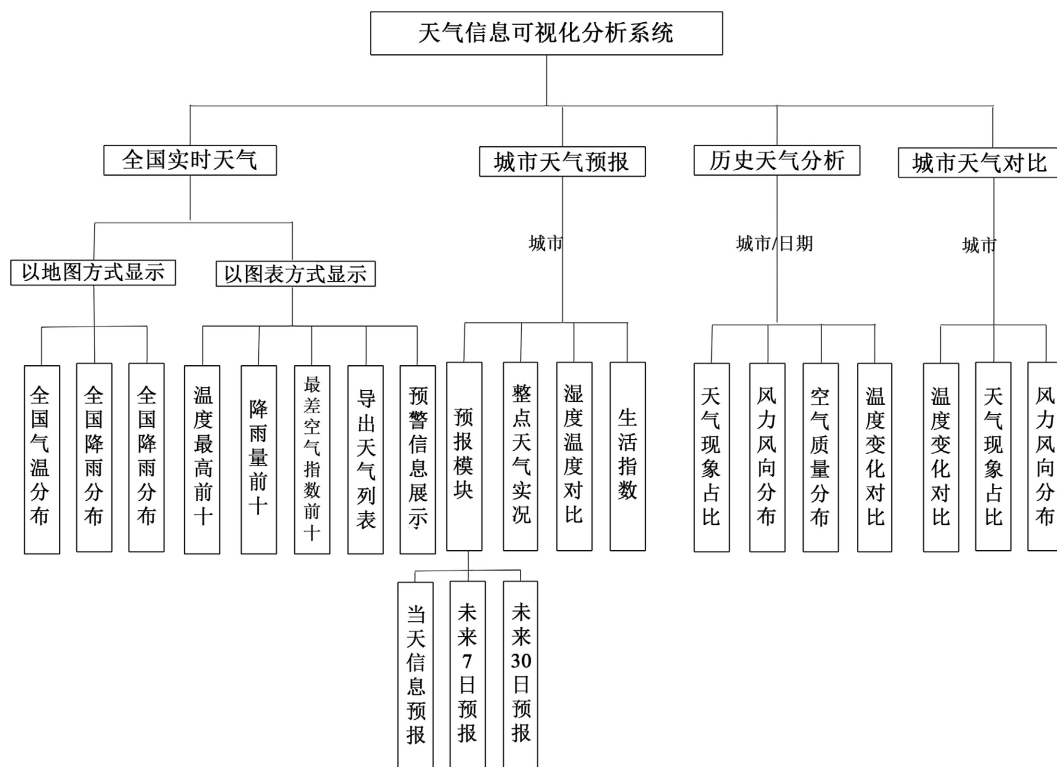


Figure 4. System function model diagram
图 4. 系统功能模块图

第一模块展示的是全国实时天气界面，可以显示全国的气温、降雨、湿度在地图上的分布，以及全国温度最高、降雨最多、空气指数最差的前十的城市，还可以导出某一天的各地区天气列表，其中包括各城市的天气、温度、降雨、湿度等。同时可以查看一些预警信息的类型和数量。

第二模块是城市天气预报界面，可以查看当天各个小时段的空气质量、温度、相对湿度、风力风向的变化图、相对湿度和温度的对比变化图，同时还可以查看当天的生活指数，对出行、穿衣等日常生活进行合理的安排和一定调整，也可以对未来7日以及未来30日的基本天气信息进行查看和了解。

第三模块是历史天气分析界面，可以选择一个城市，选择一个时间段，然后对该时间段内该城市的天气信息进行分析展示，其中包括天气现象占比图、风力风向占比图、空气质量分布图、最高最低温度变化图。

第四模块是城市天气对比界面，可以选择两个不同的城市对未来十五天内的温度、风力风向、天气现象占比情况。

4. 系统实现

4.1. 全国实时天气

在全国实时天气界面中，主要是以地图方式显示和以图表方式显示，如图5所示。

中间地图的放大缩小主要是应用 ECharts 图表库，通过实例化 option、配置属性、data 置入数据来实现自己需要的各类样式，去实现展示地图的形状，实现用红色和数字去标识信息。

在导出的天气列表中包括了各城市的天气、温度、降雨、湿度等，实现表格的导出主要应用了 openpyxl 方法。首先是获取数据库中所需要的数据，再应用 Python 中的 openpyxl 方法操作 excel，给单元格和行

赋值并创建 sheet, 同时设定保存临时文件的地址, 最后再利用 Flask 的 send_from_directory 函数和临时文件名从服务器位置下载文件并保存。



Figure 5. National real-time weather interface
图 5. 全国实时天气界面

4.2. 城市天气预报

4.2.1. 当天天气预报展示

在城市天气预报界面中, 主要通过调用数据接口实现预报信息的展示, 如图 6 所示。

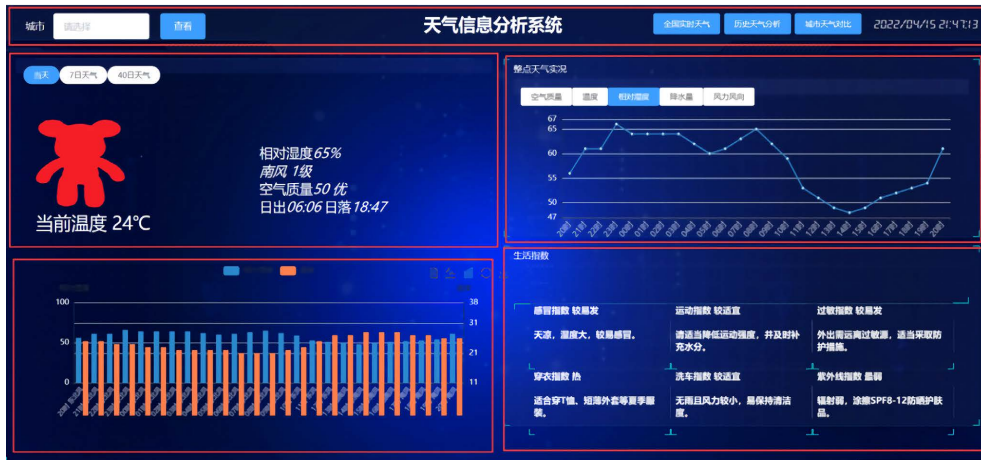


Figure 6. Urban weather forecast interface
图 6. 城市天气预报界面

在当天的天气预报模块中, 可以通过小熊的颜色判断当天大致温度情况(温度越高颜色越红), 通过准备好的几张颜色不同的图片去对应不同的温度, 如:

```
elif temp > -30 and temp <= -20:  
img = '/static/temp_bear/-30--20.png'
```

当温度在 -20 至 -30 摄氏度之间时, 这一段温度绑定 /static/temp_bear 目录下的 -30--20.png, 由此来实现小熊随温度变化而变化。

4.2.2. 未来 7 日天气预报展示



Figure 7. 7-day weather forecast interface

图 7. 7 日天气预报界面

未来 7 日的天气预报使用列表展示如图 7 所示, 主要展示日期、天气、最值温度、风向、风力、降雨概率这八项数据信息。未来 7 日的天气预报显示利用了 AJAX 技术, 其最大的优势在于, 可以和服务器进行数据的交换和更新, 而无需重装加载整个页面。首先是找到获取 7 天信息数据的路由接口, 然后再根据不同城市的代码去找到对应的数据。

url: "/get_city_weather_day7?city_code=" + city_code。

最后利用 html 的<table>标签定义好各个表头, 使用 ECharts 图表库对图表中的具体属性进行配置。

4.3. 历史天气分析

历史天气分析界面如图 8, 可以通过选择一个时间段对某一城市的天气信息数据进行对比分析展示。

最上方是导航栏, 与之前有所不同的是添加了一个时间的选择, 可以通过两个时间点的选择来对一个城市的一段时间里的天气信息数据进行对比分析。

其次是五张分析图表, 有饼图、雷达图、条形图、折线图, 分别是天气现象占比图、风力风向占比图、空气质量分布图、最高最低温度变化图。

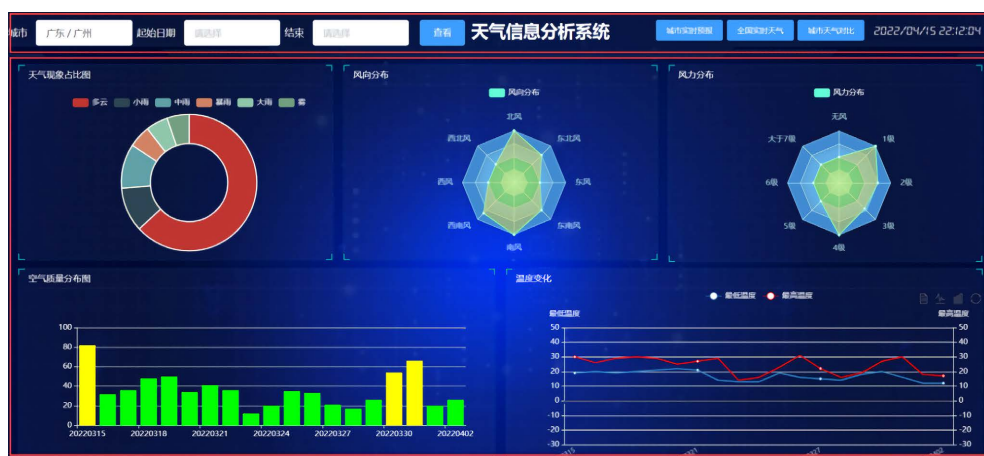


Figure 8. Historical weather analysis interface

图 8. 历史天气分析界面

4.4. 城市天气对比

城市天气对比界面如图 9, 可以通过选择两个不同城市对未来 15 天内的天气信息数据进行对比分析展示。

首先是导航栏, 这里的导航栏添加了一个城市的选择, 可以通过两个城市的选择来对同一个时间段两个城市的天气信息数据进行对比分析。

其次是四张分析图表, 有折线图、面积图、雷达图, 分别是最高气温最低气温对比图、风力风向分布图、天气现象占比图。



Figure 9. Urban weather comparison interface
图 9. 城市天气对比界面

5. 结语

网络爬虫技术作为搜索引擎的构件之一, 可以快速收集网络数据, 因此成为人们获取大量数据的有效途径之一。随着大数据应用的发展, 爬虫的应用范围越来越广。本文利用现在的 Flask 技术和爬虫技术, 对天气信息数据进行爬取和存储, 并对各种数据进行分析、对比, 设计了一个天气信息可视化分析系统。通过对爬虫技术、Flask 框架、网页开发、MySQL 数据库、ECharts 图表库等技术的应用, Python、html、JavaScript 等多种语言相结合进行编程, 可查看历史和未来的天气信息, 实现了对天气数据多角度、多维度的对比, 分析了同一城市不同时间以及同一时间不同城市的天气数据, 对历史天气数据进行多类别图表展示, 包括了天气现象占比图、风力风向占比图、空气质量分布图、最高最低温度变化图等。本系统采用了前后端分离的方式进行代码编写, 通过使用接口对数据进行封装实现外部灵活调用, 可以更方便地编写代码并对代码进行修改。后期, 在数据爬取方面还将编写更多的爬虫文件使得天气预报和分析模块能够展示更多更详细的天气信息, 在前端页面开发方面也将进行完善和美化, 设计出更新颖的展示方式, 图表类型更加多元化, 在展示上可以更加美观和有针对性。

基金项目

浙江理工大学 2021 年高等教育科学研究课题、浙江省一流课程建设项目资助。

参考文献

- [1] 李璐, 郭淇汶, 陆宇, 王跟悦. 基于 Python 的天气预测系统研究[J]. 计算机技术与发展, 2020(10): 18-20.

-
- [2] 陆树芬. 基于 Python 对网络爬虫系统的设计与实现[J]. 电脑编程技巧与维护, 2019(2): 26-27+51.
 - [3] 金静梅. 基于网络爬虫的城市天气服务系统设计与实现[J]. 办公自动化, 2021, 26(19): 58-59+30.
 - [4] 王锡良, 卿光勇, 张铭. 利用网络爬虫技术获取天气预报[J]. 电脑编程技巧与维护, 2015(19): 18-19.
 - [5] 肖新风, 张绛丽, 邓祖民. 基于 python 的爬虫技术的网站设计与实现[J]. 信息技术与信息化, 2020, 4(14): 73-75+78.
 - [6] 冯艳茹. 基于 Python 的网络爬虫系统的设计与实现[J]. 电脑与信息技术, 2021, 29(6): 47-50.
 - [7] 苻玲美. 正则表达式在 Python 爬虫中的应用[J]. 电脑知识与技术, 2019, 15(25): 253-254.
 - [8] 沈承放, 莫达隆. Python 语言中 re 库的使用技巧与目标网络数据的抓取[J]. 贺州学院学报, 2019, 35(3): 151-156.
 - [9] 牛作东, 李捍东. 基于 Python 与 flask 工具搭建可高效开发的实用型 MVC 框架[J]. 计算机应用与软件, 2019, 36(7): 21-25.
 - [10] 周文帆. 基于 Flask 的形式化验证系统的设计与实现[D]: [硕士学位论文]. 北京: 北京交通大学, 2020.
 - [11] 李相霏, 韩珂. 基于 Flask 框架的疫情数据可视化分析[J]. 计算机时代, 2021(12): 60-63+68.
 - [12] 张艺豪, 盛丹红, 翟丹丹. 基于 Flask 和 Echarts 的光缆数据管理平台[J]. 电脑编程技巧与维护, 2021(6): 94-95.
 - [13] 刘亚茹, 张军. Vue.js 框架在网站前端开发中的研究[J]. 电脑编程技巧与维护, 2022(1): 18-19+39.
 - [14] 彭曙光, 王梦梅, 赵麒博, 申刘宝, 彭玉杰, 陆雪艳. 面向 ECharts 的疫情信息可视化系统[J]. 福建电脑, 2022, 38(4): 80-83.
 - [15] 郑嘉怡, 陈涛, 齐世玲, 朱东. 基于 Echarts 的智能车间数据可视化系统[J]. 电脑知识与技术, 2021, 17(25): 68-69.