

**本科毕业设计（论文）**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **题目** | **基于React的气候可视化平台** | | | |
| **的设计与实现** | | | | |
| **学生姓名** | | **廖新** | **学 号** | **201831107219** |
| **教学院系** | | **计算机科学学院** | | |
| **专业年级** | | **软件工程2018级** | | |
| **指导教师** | | **肖斌** | **职 称** | **教授** |
| **单 位** | | **西南石油大学** | | |
| **辅导教师** | |  | **职 称** |  |
| **单 位** | |  | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **完成日期** | **2022** | **年** | **05** | **月** | **26** | **日** |

**摘 要**

面对高频次、庞大的气象数据，传统的表达方式难以有效呈现数据背后所蕴含的信息内容,挖掘气象大数据背后的价值,进行数据可视化分析显得尤为重要。本文基于成都市历史天气数据，使用随机森林算法和AdaBoost算法分别建立最高气温模型与最低气温模型，预测成都市未来三天最高气温与最低气温。研究气温的变化趋势，可以为农业、林业等行业提供数据支撑服务，降低自然灾害带来的损失。在构建模型的实验过程中，使用scikit-learn机器学习库中的OrdinalEncoder方法分别对训练集与测试集样本的分类型特征变量进行特征编码，将时间序列预测问题转换为监督学习，使用学习曲线和网格搜索进行模型的参数调优，采用交叉验证评估模型预测的准确率，利用相关的回归评价指标判断模型的表现。实验结果表明，以随机森林算法构建的两个模型具有较高的预测精度和较强的泛化能力，最高气温模型的为0.826439，MAE为2.768079，RMSE为3.473466。最低气温模型的为0.918554，MAE为1.568638，RMSE为1.995579。为实现本文的气候可视化平台，前端采用JavaScript作为开发语言，React搭建平台，使用ECharts、L7实现页面可视化，利用Axios请求数据。后端采用Python作为开发语言，Flask搭建平台，数据存储选择MySQL作为数据库。本文的气候可视化平台主要实现了天气播报、气候可视化、历史天气功能模块，并且将两个气温预测模型投入平台使用，预测成都市未来三天气温变动，提高了平台的实用性。

**关键词：**气温预测；随机森林；可视化；React

**Abstract**

In the face of high-frequency and huge weather data, it is difficult to effectively present the information content behind the data in the traditional way, so it is especially important to explore the value behind the weather data and perform data visualization and analysis. In this paper, based on the historical weather data of Chengdu city, we use Random Forest algorithm and AdaBoost algorithm to build the maximum and minimum temperature models to predict the maximum and minimum temperature of Chengdu city in the next three days. The study of temperature trends can provide data support services for agriculture, forestry and other industries to reduce the losses caused by natural disasters. In the experimental process of constructing the models, the OrdinalEncoder method in the scikit-learn machine learning library is used to encode features for the subtype feature variables of the training set and test set samples, respectively, to convert the time series prediction problem to supervised learning, to perform parameter tuning of the models using learning curves and grid search, to evaluate the accuracy of model prediction using cross-validation , and judge the performance of the model using relevant regression evaluation metrics. The experimental results show that the two models constructed with the random forest algorithm have high prediction accuracy and strong generalization ability, with of 0.826439, MAE of 2.768079 and RMSE of 3.473466 for the highest temperature model. of 0.918554, MAE of 1.568638 and RMSE of 1.995579 for the lowest temperature model. To realize the climate visualization platform in this paper, the front-end uses JavaScript as the development language, uses React to build the platform, uses ECharts, L7 to visualize the page, and uses Axios to request data. The back-end uses Python as the development language, uses Flask to build the platform, and MySQL is chosen as the database for data storage. The climate visualization platform in this paper mainly implements weather broadcast, climate visualization, and historical weather function modules, and puts two temperature prediction models into the platform to predict the temperature changes in Chengdu city in the next three days, which improves the usefulness of the platform.

Key words: Temperature Forecast; Random Forest; Visualisation; React

**目录**

[1 绪论 1](#_Toc103349273)

[1.1 研究背景及目的意义 1](#_Toc103349274)

[1.2 国内外研究现状 2](#_Toc103349275)

[1.2.1 气象预测研究现状 2](#_Toc103349276)

[1.2.2 气候可视化系统研究现状 3](#_Toc103349277)

[1.3 论文的主要工作及结构 4](#_Toc103349278)

[2 关键技术与理论介绍 5](#_Toc103349279)

[2.1 React.js和Ant Design 5](#_Toc103349280)

[2.2 ECharts和L7 5](#_Toc103349281)

[2.3 Python和Flask 6](#_Toc103349282)

[2.4 MySQL数据库 6](#_Toc103349283)

[2.5 随机森林算法 6](#_Toc103349284)

[2.6 本章小结 7](#_Toc103349285)

[3 气候可视化平台的分析 8](#_Toc103349286)

[3.1 可行性分析 8](#_Toc103349287)

[3.1.1 技术可行性 8](#_Toc103349288)

[3.1.2 经济可行性 9](#_Toc103349289)

[3.1.3 操作可行性 9](#_Toc103349290)

[3.2 功能需求分析 9](#_Toc103349291)

[3.2.1 天气播报 9](#_Toc103349292)

[3.2.2 气候可视化 10](#_Toc103349293)

[3.2.3 历史天气 11](#_Toc103349294)

[3.3 本章小结 12](#_Toc103349295)

[4 气候可视化平台的设计 13](#_Toc103349296)

[4.1 平台开发准则 13](#_Toc103349297)

[4.2 平台整体框架设计 14](#_Toc103349298)

[4.3 平台功能模块设计 15](#_Toc103349299)

[4.4 数据库设计 17](#_Toc103349300)

[4.5 本章小结 18](#_Toc103349301)

[5 基于随机森林与AdaBoost的气温预测模型 19](#_Toc103349302)

[5.1 相关算法研究 19](#_Toc103349303)

[5.1.1 AdaBoost算法 19](#_Toc103349304)

[5.1.2 学习曲线和网格搜索 19](#_Toc103349305)

[5.1.3 回归评价指标 20](#_Toc103349306)

[5.2 数据爬取 21](#_Toc103349307)

[5.2.1 相关方法 21](#_Toc103349308)

[5.2.2 处理过程 22](#_Toc103349309)

[5.3 实验与分析 28](#_Toc103349310)

[5.3.1 数据预处理 28](#_Toc103349311)

[5.3.2 气温预测模型建立 32](#_Toc103349312)

[5.3.3 实验结果与对比分析 34](#_Toc103349313)

[5.4 本章小结 37](#_Toc103349314)

[6 基于React的气候可视化平台的实现 38](#_Toc103349315)

[6.1 开发环境配置 38](#_Toc103349316)

[6.2 数据准备 38](#_Toc103349317)

[6.3 平台的功能实现 38](#_Toc103349318)

[6.3.1 天气播报 38](#_Toc103349319)

[6.3.2 气候可视化 40](#_Toc103349320)

[6.3.3 历史天气 47](#_Toc103349321)

[6.4 系统功能的测试 50](#_Toc103349322)

[6.4.1 软件测试 50](#_Toc103349323)

[6.4.2 测试的结果 51](#_Toc103349324)

[6.5 本章小结 52](#_Toc103349325)

[7 总结与展望 53](#_Toc103349326)

[致谢 54](#_Toc103349327)

[参考文献 55](#_Toc103349328)

# 绪论

## 研究背景及目的意义

观察近几年的气象的变化，全球的气候越来越不稳定，全球气温上升明显，而极端气候出现的越来越频繁。气象千变万化且变化的过程极为复杂，而气候的异常可能会引发高温、冰雹、雷雨大风等自然灾害，进而影响到人们正常的生产生活，影响到国家社会的经济发展。气候变化具有不可见和不可预知性，其导致的危害和问题从时间和空间来说，公众难以感知得到。大数据时代的到来，信息化建设的推进，气象领域已经积累了丰富的数据，数据中蕴含着大量的价值亟待被开发出来，系统能否充分挖掘数据价值，为用户提供可靠的决策支持是用户十分关注的问题。并且气象作为基础的支撑信息，已经和各个行业形成紧密的联系，面对高频次、庞大的气象数据，传统的表达方式难以有效呈现数据背后所蕴含的信息内容，挖掘气象大数据背后的价值，进行数据可视化分析显得尤为重要。

气候异常导致的自然灾害常常会威胁到人民的经济财产损失,并且随着全面实现小康社会的快速发展的进程中，各个行业（农业、交通、水利、电力、新能源等）对气象预报的依赖程度越来越高，越发需要更加精准的气象服务作为支撑[1]。世界各国也在气象部门建立了气象数据处理平台，并对外提供了公共气象数据的调用接口，为社会公众提供数据查询下载和各类数据服务。由于传统的气象数据服务大多直接展示数据，不利于用户敏感地获取信息，另外实现的气象数据系统采用技术未紧跟时代潮流，导致用户交互性不强，界面不美观。为了使用户能与抽象化的气候数据直接交互，使之清晰化、生动化，平台使用ECharts、L7等相关技术实现了气候数据可视化功能。

此外，为了完善平台的功能与服务、提高平台的实用性，采用随机森林算法建立了最终的最高温度模型和最低温度模型用于气温预测。实验表明，模型的预测准确率较高。因此，本课题的研究不仅可以利用天气的历史数据来对未来的气温变动的情况做出预测，还可以将收集到的大量气象数据借助可视化技术进行显示，挖掘数据背后蕴含的意义，使得数据生动而形象，人们进而提前做好预防工作，调整策略。并且，通过对本课题的研究，既可以提高学术研究水平，丰富该领域的研究，又能通过数据可视化方式向公众传播气候变化[2]，具有一定的实践意义。

## 国内外研究现状

### 气象预测研究现状

目前已经有一些学者对天气预测进行了相关研究，在20世纪80年代，西方发达国家已经开始了区域数值模式的研究，一些外国学者开始将神经网络等各种算法模型应用于天气预报。Haidar A等[3]人提出一种新的预测方法，首次利用深度卷积神经网络（CNN）对澳大利亚东部选定位置预测月降水量。Pérez-Vega A等[4]对支持向量机的不同核函数在预测温度的效果中进行了比较分析，得出基于多项式核函数的支持向量机预测模型的预测效果最佳的结论。Joshi P等[5]基于人工神经网络算法对冬季的喜马拉雅山的极值温度(最高和最低温度)进行预测，主要基于地面的观测数据，并得到了良好的预测结果。Lathifah S N等[6]使用传统的分类回归树（CART）算法结合自适应采综合样（ADASYN）算法预测万隆地区的降雨量，成功的建立了一个准确的天气预测系统的模型。

随着科学技术的发展，国内不少学者开展了大量的研究，将各类算法模型应用到天气预测领域中，并取得了优秀的成果。朱晶晶等[7]人依据CMSVM2.0函数估计和交叉验证等方法，利用海南省各市县月平均气温资料，建立了气温的SVM回归方法预报模型。许倡泷[8]结合广西容县的月平均气温数据，提出自适应指数平滑算法，通过对不同预测模型的比较和仿真实验，结果表明基于自适应指数平滑算法的气温预测模型预测精度较高，实用性强，具有一定推广性。朱国栋等[9]人使用随机森林方法，结合不同数量的决策树进行模型训练，研究建立基于随机森林方法的乌鲁木齐机场逐时温度回归预报模型，通过对模型预测结果的检验，可以尝试使用此方法制作民航机场客观要素指导产品。白雪[10]在小时气温和日最高气温的预测研究中，提出了基于时间序列模型和XGBoost组合的模型，并且与单变量时间序列模型及三种代表性机器学习算法进行比较分析，发现在小时气温的预测中，ARIMA-XGBoost组合模型在测试集上的表现最佳;在日最高气温的预测中，Prophet-XGBoost组合模型在测试集上的表现最佳。陶晔等[11]人利用随机森林选择出与气温高度相关的气象要素作为输入变量,消除原始气象数据中的噪音、降低网络的复杂度,在此基础上利用长短期记忆网络建立总体预测模型,在采集的多要素气象数据上进行实验。结果表明,该模型在处理大规模多变量的时间序列数据时具有较高的预测精度和较强的泛化能力。胡玉杰[12]使用气象站点的历史数据进行训练,提出了一种改进的Adaboost-C4.5算法预测模型,利用PSO算法来优化Adaboost-C4.5弱分类器的权重,提高了降雨预测精度,并将PSO-Adaboost-C4.5算法模型集成于系统形成气象预报服务,大大提高了气象数据服务平台的实用性。

### 气候可视化系统研究现状

在对气候可视化方面的研究中，2015年9月28至10月2日，欧洲中期天气预报中心（ECMWF）在英国里丁大学举行了气象信息可视化会议[13]。该会议围绕ECMWF数值天气预报（NWP）核心业务，总结了可视化方法和技术在气象监测、预报分析与检验、服务、教育培训等方面发挥的重要的作用。ECMWF介绍了如何增强天气预报产品的可视化效果和可视化信心，还介绍了哥白尼项目（Copernicus）中大气监测服务中的可视化应用以及基于Magics++的Web全球洪水预警系统（GloFAs）。英国气象局介绍了基于Web、第 3方插件、图形工作站(VW)、地理信息系统(ArcGIS)以及通用气象信息显示软件(Weather Eye／Visual Cortex)等多种应用平台支持下气象信息可视化业务[13]。

在我国，相关学者也逐步展开了对气候数据可视化的研究。李涛等[14]人提出基于Hadoop的分布式计算和存储框架,使用了Hadoop生态体系中的HDFS分布式文件存储框架来存储管理分析海量气象数据，结果表明,使用分布式多节点集群可以有效提高海量气象数据的存储和计算效率,解决了传统Web GIS平台数据存储与计算的局限性问题。李成渊等[15]人设计的系统主要针对北京冬奥会和杭州亚运会的举办,从赛事需求角度出发,采集和分析天气和空气质量等历史数据,以Python为工具,使用环境云网站提供的服务，以参照城市气象数据作为基点,使用Matplotlib实现数据可视化。黎颖智等[16]基于ECharts可视化技术,采用B/S架构开发库区水文及气象数据监测系统，监测系统能将水库流量、降雨实时监测与预测情况以图表的方式呈现。陈东辉等[17]针对气象台站的历史沿革文件在开发利用过程中出现的查询检索效率低下、统计分析不便等问题，设计开发了气象台站历史沿革信息检索可视化系统。为直观了解气象台站分布及沿革信息，创建轻量级 WebGIS 框架实现可视化信息交互功能，为气候资料数据集研制、数据均一性的检验与订正等提供重要的数据支撑。

可视化方式相对单一，信息交互不够充分，气象服务产品出现专业性太强、可视化表达繁琐等问题难以得到及时反馈，影响气象服务效果。所以要提升服务效果，就更要借助数据信息可视化的力量，不断提升数据信息可视化水平，进而提升气象服务效果[18]。

## 论文的主要工作及结构

本论文研究的是基于React的气候可视化平台的设计与实现，通过可视化技术，既可以增强受众对气象信息的理解，使传播内容切实为受众服务，又能推进气象数据信息可视化在气象服务领域的发展。平台完成了三个功能模块：天气播报、气候可视化、历史天气。本文的主要工作和结构如下：

第一章介绍了课题研究背景及目的意义，气温预测以及气候可视化系统的国内外研究现状，最后则是介绍了本文的主要工作和组织结构。

第二章介绍了实现本文的气候可视化平台所需要的关键技术，以及使用到各技术的理由。

第三章主要介绍了气候可视化平台的分析，包括可行性分析、平台的功能需求分析。

第四章则是介绍了平台的设计，通过图片、表格、文字结合的方式对平台开发准则、平台整体框架设计、平台功能模块设计以及数据库设计进行详细的说明。

第五章则是基于随机森林与AdaBoost的气温预测模型的建立，重点阐述了数据的预处理过程，模型的建立过程，实验结果与分析。

第六章介绍了气候可视化平台的实现和测试，通过图片、表格、文字结合的方式详细的表述了平台的各功能模块和测试结果。

第七章则是全文的总结和展望，对本题进行简单的总结，主要介绍了本课题存在的一些不足之处，以及未涉及到的方面，同时也提供了未来人们可以进一步探讨的方向。

# 关键技术与理论介绍

## React.js和Ant Design

React是一个声明式，高效且灵活的用于构建用户界面的JavaScript库。使用React可以将一些简短、独立的代码片段组合成复杂的UI界面，这些代码片段被称作“组件”。React不强制使用JSX，JSX是一个JavaScript的语法拓展，可以很好地描述UI应该呈现出它应有交互的本质形式。JSX可能会使人联想到模板语言，但它具有JavaScript的全部功能，它还可以使React显示更多有用的错误和警告信息。

React具备的如下特点使得大多数人们愿意使用它开发软件。声明式，React创建交互式UI变得轻而易举。为应用的每一个状态设计简洁的视图，当数据变动时,React能够高效更新并渲染合适的组件。组件化，构建管理自身状态的封装组件，然后对其组合以构成复杂的UI。由于组件逻辑使用JavaScript编写而非模板，因此可以轻松地在应用中传递数据，并保持状态与DOM分离。灵活，无论现在使用哪种技术栈，在无需重写现有代码的前提下，通过引入React来开发新功能。

Ant Design，一套企业级 UI 设计语言和 React 组件库。Ant Design结合React.js可以快速搭建网站，极大地减少人力和时间成本，提高工作效率。

## ECharts和L7

ECharts，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，底层依赖矢量图形库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。ECharts提供了丰富的可视化类型，包括常规的折线图、柱状图、散点图、饼图，并且多种数据格式无需转换直接使用。通过增量渲染技术（4.0+），配合各种细致的优化，ECharts能够展现千万级的数据量，并且在这个数据量级依然能够进行流畅的缩放平移等交互。ECharts支持以Canvas、SVG（4.0+）、VML的形式渲染图表，多种渲染方案，跨平台使用。ECharts 3开始加强了对多维数据的支持。除了加入了平行坐标等常见的多维数据可视化工具外，对于传统的散点图等，传入的数据也可以是多个维度的。配合视觉映射组件提供的丰富的视觉编码，能够将不同维度的数据映射到颜色、大小、透明度、明暗度等不同的视觉通道。

L7是由蚂蚁金服AntV数据可视化团队推出的基于WebGL的开源大规模地理空间数据可视分析开发框架。L7专注于空间数据的可视化表达。图形符号学为理论基础，将抽象复杂的空间数据转化成2D、3D符号，通过颜色、大小、体积、纹理等视觉变量实现丰富的可视化表达。

## Python和Flask

Python是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言。Python易于学习，关键字相对较少，结构简单，语法定义明确。Python易于阅读，并且更清晰地定义了代码。Python拥有一个强大的标准库，提供了系统管理、网络通信、文本处理、数据库接口等额外的功能。Python解释器易于扩展，可以使用C或C++扩充新功能和数据类型。

Flask是一种采用Python语言编写的轻量级、可定制的框架，它比其它同类型框架更为灵活、轻便、安全且容易上手。Flask可以很好地结合MVC模式进行开发，还具有很强的定制性，用户可以根据自己的需求来添加相应的功能，可以在保留原有的核心功能的基础上，实现功能的丰富与扩展，而且其强大的插件库可以让使用者更好地进行个性化的网站定制，开发出功能强大的网站。

## MySQL数据库

MySQL是一种使用SQL语言进行增、删、改、查等操作的关系式数据库，目前是Oracle公司旗下的产品。MySQL关系数据库可以把数据存储在不同的表中，而不是将所有数据放在一个大仓库内，从而加快了数据存储的速度和灵活性。同时，MySQL支持多线程，充分利用CPU资源，支持多种存储引擎，还提供用于管理、检查、优化数据库操作的管理工具。此外MySQL是一个开源免费，支持跨平台、分布式的数据库，它的性能也很好，与Web开发语言如PHP、Python完美配合，非常适合中小型企业作为Web数据库。

## 随机森林算法

随机森林是一种非常具有代表性的Bagging（Bootstrap Aggregating）集成学习算法，由多棵决策树组成。Bagging方法对训练样本集进行多次Bootstrap抽样，用每次抽样形成的数据集训练一个弱学习器模型，得到多个独立的弱学习器，最后用他们的组合进行预测[19]。对于分类问题，一个测试样本会送到每棵决策树中，进行预测，然后投票，得票最多的类为最终分类结果。对于回归问题，随机森林的预测输出是所有决策树输出的均值。随机森林算法用多棵决策树联合预测可以提高模型的精度，这些决策树用对训练样本集随机抽样构造出的样本集训练得到。随机森林不仅对训练样本进行抽样，还对特征向量的分量随机抽样，在训练决策树时，每次寻找最佳分裂时只使用一部分抽样的特征分量作为候选特征[19]。正是因为有了这些随机性，随机森林可以在一定程度上消除过拟合。

## 本章小结

本章主要对该平台涉及到的相关技术与理论进行介绍，包括但不限于构建模型的随机森林算法，前端使用到的React.js、Ant Design、ECharts、L7，后端使用到的Python语言、Flask框架，同时简单介绍了MySQL数据库。

# 气候可视化平台的分析

随着大数据的发展，单一的天气状况已经无法满足公众的需求，不同的受众人群对气象信息的需求也不一样，在各行各业所具备的作用也不相同。对于普通民众来说，气象基础数据主要包括天气状况和自然灾害到来时的预报及预警[20]。对于大多企业来说，气象数据广泛应用于电力能源、智慧城市、智慧交通、智慧农业等多个领域。在农业生产方面，利用气候数据可以帮助农民将农业工作合理化，最大程度地减少由气候因素导致的农业损失。在物流运输方面，向运输公司提供天气信息可以帮助承运人确定最佳运输路线，确保货运时间，提高物流效率并降低货运成本[21]。在防灾减灾方面，可以分析植物、房屋、民航等并预防和减少灾区。

因此，针对不同用户、不同行业的需求，进行气候可视化平台的分析尤为重要。由于数据收集、时间等因素的限制，本文完成的功能有限。

## 可行性分析

### 技术可行性

在技术层面，采用前后端分离开发的模式，有效地减少后端服务器的并发压力，提升开发效率，增强代码的维护性和易读性。

前端采用JavaScript作为开发语言，它是一种高级的、解释性、面向对象编程的语言，并且它不依赖于操作系统，仅需要浏览器的支持。Ant Design结合React.js可以快速搭建网站，极大地减少人力和时间成本，提高工作效率。使用ECharts、L7实现页面可视化，给用户带来直观地视觉享受。

后端采用Python作为开发语言，它是一个高层次的结合了解释性、编译性、互动性和面向对象的脚本语言，另外Python易于学习，有相对较少的关键字，结构简单，和一个明确定义的语法。结合Flask搭建后端框架，Flask较其他同类型框架更为灵活、轻便、安全且容易上手，并且拥有强大的插件库。

数据存储选择MySQL作为数据库，前后端的通信采用Axios请求数据交互。项目的管理使用Git进行版本的迭代。Git是一个开源的分布式版本控制系统，可以有效、高速地处理从很小到非常大的项目版本管理。

从成熟的技术上来讲，软件的开发完全可行。

### 经济可行性

考虑到平台以后的发展前景，降低平台开发、运维的成本是很有必要的。在满足平台功能需求的前提下，从框架选型、数据库设计、硬件配置角度出发，选择免费开源的软件成为节约成本的重要措施。因此，从经济上来说也是可行的。

### 操作可行性

在软件的操作层面，前端的页面清楚地向用户展示了每个功能，用户可以根据导航进行功能模块的选择。可视化页面向用户直观地显示了每个地区的气候变化，用户可获取关心的数据信息。

用户只需要简单的学习软件的使用，无需复杂操作，很快就可熟练使用，因此，在操作上也是可行的。

## 功能需求分析

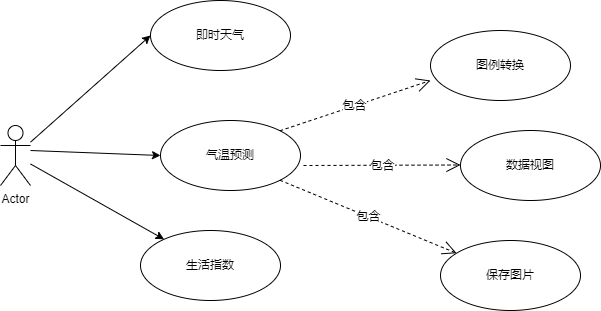
### 天气播报

天气播报内含三个功能，天气即时显示、天气生活指数、气温预测。用户可以参考天气播报的功能与建议决定一天的出行情况。

天气即时显示用途是展示成都市的即时天气情况，包括温度，风向风力、风速、大气压强、能见度、相对湿度。用户可以根据天气实况做出生活的调整。

天气生活指数用途是展示为居民生活出行而提供的参考数据，例如运动指数、紫外线指数、穿衣指数、感冒指数。

气温预测用途是利用本文研究实验建立的最高温度与最低温度的气温预测模型来预测成都市未来三天气温的变动情况。这些数据通过图表的形式显示。图表提供图例的切换，例如折线图、柱状图，数据视图，保存为图片等功能。天气播报功能用例图如图3.1所示。



**图 3.1 天气播报用例图**

### 气候可视化

为了用户可以直观地获取相关的气候信息，从而采取相应的措施，做好预防工作，调整策略。气候可视化是将相关的气候数据使用图表、地图的形式将数据进行可视化显示，利用图表的形式使得数据生动而形象。气候可视化需要包括中国地级市地图可视化、天气数据可视化、预警信息可视化、空气质量可视化、分钟级降水可视化。

中国地级市地图可视化功能是用户点击某个地区可以获取相应地区的经纬度用来查询相应地区的相关气候数据。

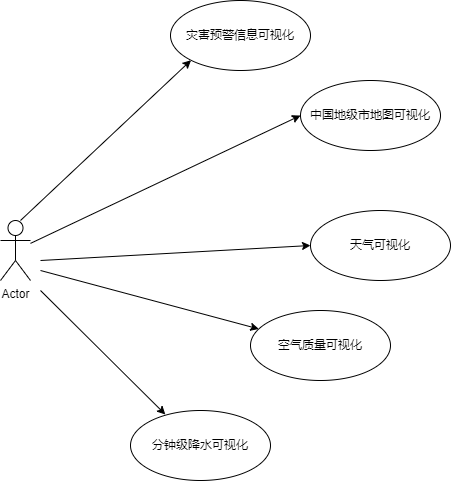
天气数据可视化主要用途是展示各地区的即时天气情况，包括温度，风向风力、风速、大气压强、能见度、相对湿度。用户可以根据天气实况做出生活的调整。

预警数据可视化采用表格的形式，主要用途是显示相应地区气象台具体时间发布的灾害预警数据和一些建议，如果地区未发布预警信息则不显示。

空气质量可视化分为两部分，空气质量指数和空气质量元素。空气质量指数使用仪表盘表达指数等级和相关指数，空气质量元素使用雷达图表达相关空气元素的浓度。

分钟级降水可视化采用折线图的形式，主要用途是显示各地区未来2小时每5分钟的降水量，并且给出相关的未来两小时降水概况信息，用户可根据降水量和概况信息调整策略，做好相应措施。

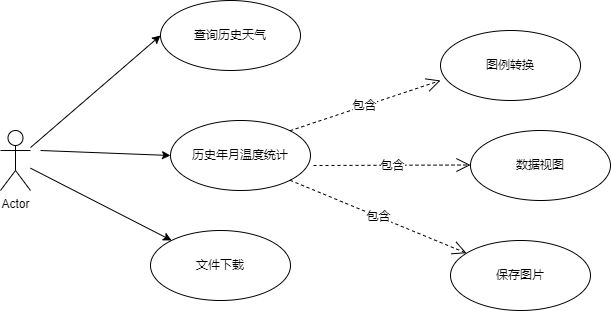
气候可视化功能用例图如图3.2所示。



**图 3.2 气候可视化用例图**

### 历史天气

用户可以根据特定的城市、年份、月份来查询相应的历史天气数据，数据包括历史月份每一天的详细数据、平均温度、平均空气质量数据。同时提供文件下载功能，用户点击下载按钮可以将表格中的数据以Excel的形式保存起来。另外根据特定城市、月份查询，可以显示相应城市历史年月的平均最高温度和最低温度，这些数据通过图表的形式显示。图表提供图例的切换，例如折线图、柱状图，数据视图，保存为图片等功能。该功能用例图如图3.3所示。



**图 3.3 历史天气用例图**

## 本章小结

本章介绍了对气候可视化平台的分析，从技术、操作、经济方面，做了平台的可行性分析。针对每个模块所需要的一些功能，进行功能性需求分析。这些分析构成了软件的需求方案，为平台的设计开发打下坚实的基础。

# 气候可视化平台的设计

## 平台开发准则

为避免软件在开发过程中出现中断、效率不高、难以维护等问题，本平台的开发需要遵循以下原则。

1. 实用性

本平台本着能够满足用户的基本需求，又能让用户简单快捷地操作气象数据的原则，整个平台基于B/S模式，页面设计美观且操作便利，融入了交互式可视化元素，实现了天气播报、气候可视化、历史天气功能，具有较高的实用价值。

1. 可扩展性

随着气象业务水平的不断提高以及用户对气象业务平台的需求越来越高，未来一定会对平台的功能模块进行扩展，因此，本文的气候可视化平台必须具备可扩展性。本平台基于React设计，它具备的特性使得开发人员在后期的更新与维护中可以很好地扩展功能模块。另外，得益于React强大的生态，可以采用社区中的方法与工具来帮助扩展。而且，本平台后端采用Python编码，Python拥有一个强大的标准库，提供了系统管理、网络通信、文本处理、数据库接口等额外的功能，易于扩展。

1. 可读性

编程有很大一部分时间是在阅读代码，不仅要阅读自己的代码，而且要阅读别人的代码。因此，可读性良好的代码能够大大提高编程效率。可读性良好的代码往往会让代码架构更好，在后期平台的维护中，程序员更愿意去修改这部分代码，而且也更容易修改。只有在核心领域为了效率才可以放弃可读性，否则可读性是第一位。

1. 可靠性

系统越复杂、意味着其承载的信息量越大，重要性越高、功能越强、适用范围也就越广，一旦失效所造成的损失也是巨大的。为减轻系统的风险，本平台会使用黑盒测试等测试方法保障软件的质量。

1. 兼容性

平台基于B/S体系结构设计，需要能够很好地兼容主流浏览器，例如，Chrome、火狐浏览器、Edge浏览器。在设计各功能时必须考虑浏览器内置模块、一些方法向下兼容。

## 平台整体框架设计

本文的气候可视化平台采用了B/S结构，B/S结构采取浏览器请求，服务器响应的工作模式。用户可以通过浏览器去访问Internet上由Web服务器产生的文本、数据、图片、动画、视频点播和声音等信息，而每一个Web服务器又可以通过各种方式与数据库服务器连接，大量的数据实际存放在数据库服务器中。B/S结构统一了客户端，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

平台的前端使用React.js和Ant Design进行快速开发。系统后端采用Python与Flask框架快速开发，数据存储选择MySQL。平台整体框架设计如图4.1所示。

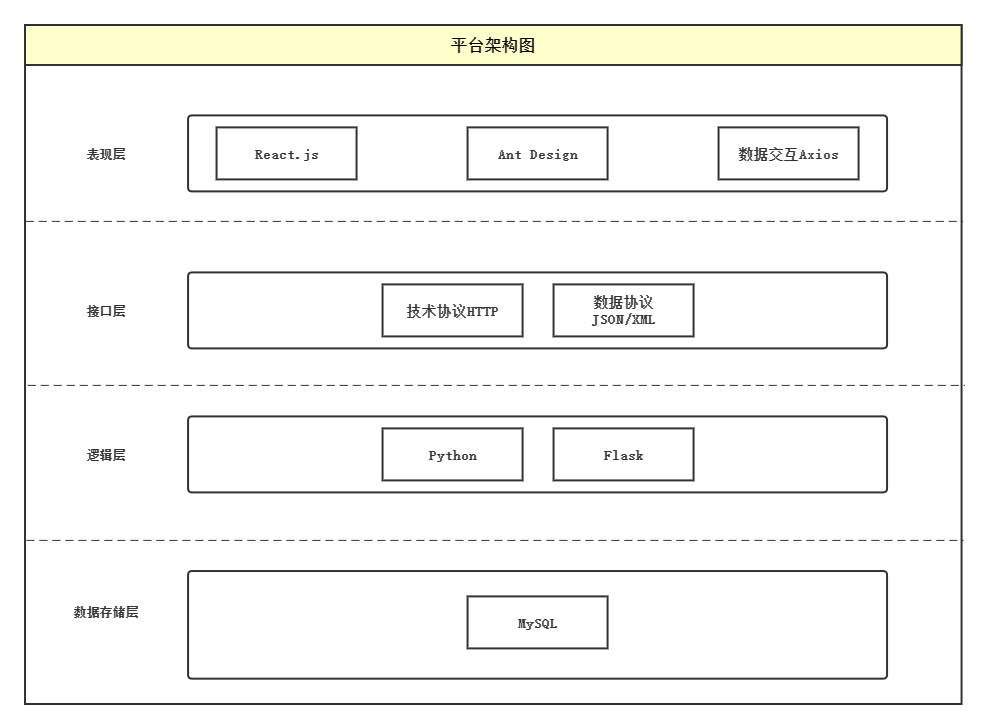


图 4.1 平台框架图

## 平台功能模块设计

本系统分为三大模块，分别是天气播报、气候可视化、历史天气。天气播报模块包含了天气即时显示、生活指数、气温预测等功能，气候可视化模块包括地图可视化、天气数据可视化、分钟级降水可视化、空气质量可视化、预警信息可视化等功能，历史天气包括历史年月气温图、下载天气历史数据等功能。整体功能结构图如图4.2。

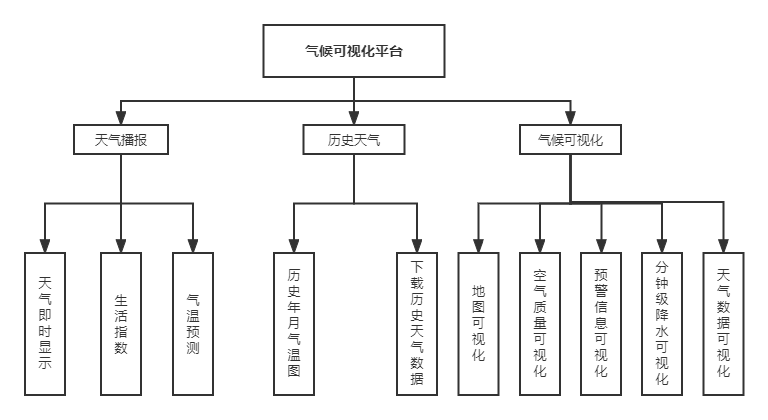
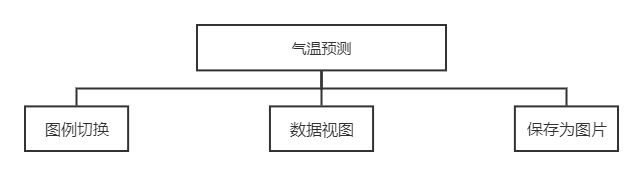


图 4.2 系统详细功能图

1. 天气播报

天气播报主要用途是显示即时天气信息、生活指数、气温预测，用户可以参考天气播报的功能与建议决定一天的出行情况。

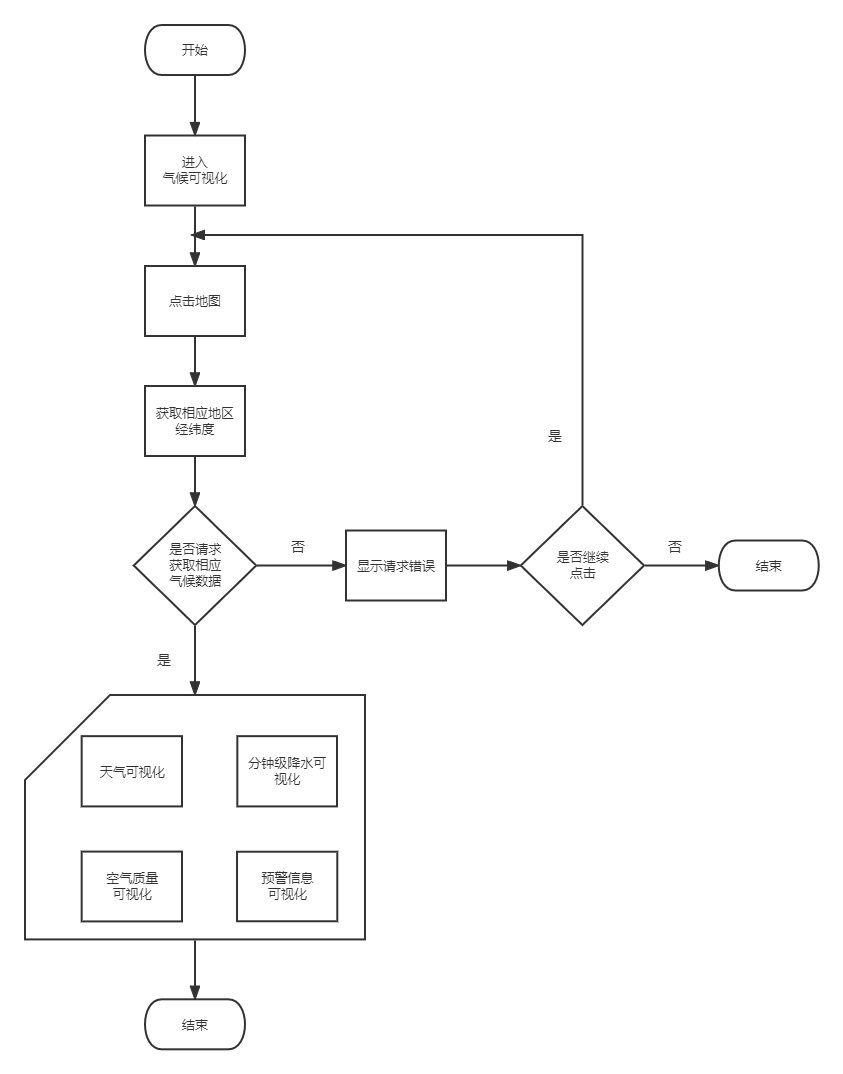
气温预测功能采用图表的形式进行展示，图表提供图例的切换，例如折线图、柱状图，数据视图，保存为图片等功能。气温预测功能结构如图4.3所示。



**图 4.3 气温预测功能结构图**

1. 气候可视化

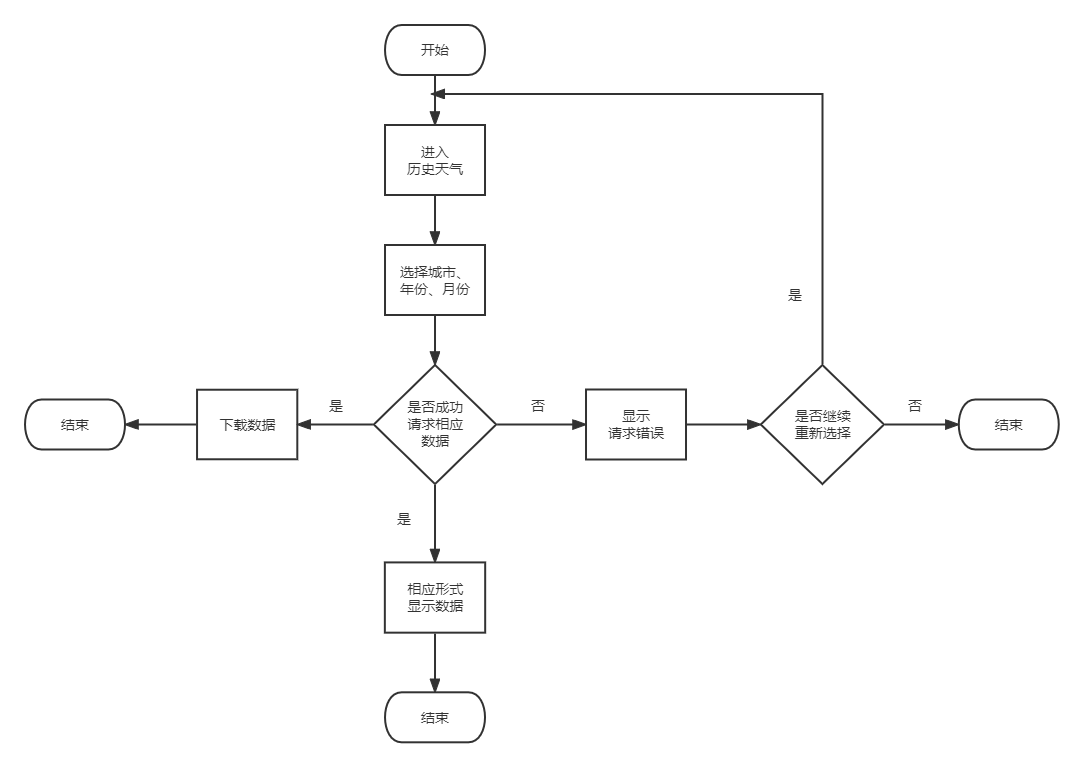
气候可视化主要用途是将相关的气候数据使用图表、地图的形式将数据进行可视化显示，利用图表的形式使得数据生动而形象。气候可视化功能流程图如图4.4所示。



**图 4.4 气候可视化功能流程图**

1. 历史天气

历史天气主要功能是用户可以根据特定的城市、年份、月份来查询相应的历史天气数据，提供文件下载功能，显示相应城市历史年月的平均最高温度和最低温度。历史天气功能流程图如图4.5所示。



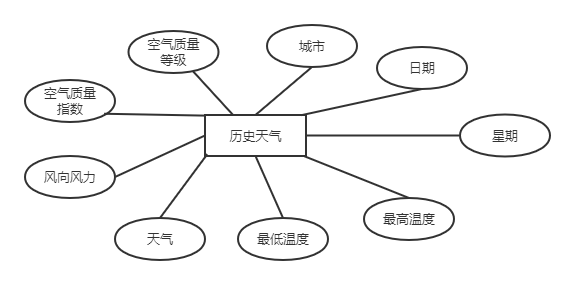
**图 4.5 历史天气功能流程图**

## 数据库设计

无论是应用程序，还是数据库如何变化，数据始终是最重要的部分。数据库是现代各种计算机应用系统的核心，数据库设计是数据库应用系统设计与开发的关键性作用[22]。良好的数据库设计，不仅可以节省数据的存储空间，能够保证数据的完整性，方便进行数据库应用系统的开发，同时拥有健壮、可靠的数据也有助于开发和测试等。

本平台数据库设计主要是为历史天气功能提供服务，通过使用数据库可以节约存储空间、方便快捷查询历史天气信息。本平台数据库选取MySQL数据库，支持多线程，充分利用CPU资源，提供用于管理、检查、优化数据库操作的管理工具。

本系统需要存储在数据库中的数据设计如表4.1所示，历史天气实体图如图4.6所示。



**图 4.6 历史天气实体图**

**表 4.1 历史天气数表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 允许空值 | 主键 | 说明 |
| 1 | city | varchar | 255 | Y | N | 城市 |
| 2 | date | varchar | 255 | Y | N | 日期 |
| 3 | week | varchar | 255 | Y | N | 星期 |
| 4 | max\_temp | varchar | 255 | Y | N | 最高温度 |
| 5 | min\_temp | varchar | 255 | Y | N | 最低温度 |
| 6 | weather | varchar | 255 | Y | N | 天气 |
| 7 | wind\_power | varchar | 255 | Y | N | 风向风力 |
| 8 | air\_quality\_index | varchar | 255 | Y | N | 空气质量指数 |
| 9 | air\_quality\_class | varchar | 255 | Y | N | 空气质量等级 |

## 本章小结

本章主要对此平台进行设计。首先阐述了本平台的开发准则以及平台整体框架设计，然后对平台功能模块进行详细的设计，最后对本平台的历史天气功能所需要的数据进行数据表设计。

# 基于随机森林与AdaBoost的气温预测模型

## 相关算法研究

### AdaBoost算法

AdaBoost算法全称是自适应提升，是一种有代表性的Boosting集成算法。Boosting算法它的学习器是由多个弱学习器组成，预测时用每个弱学习器分别进行预测，然后投票得到结果；训练时一次训练每个弱学习器，在这里采用了与随机森林不同的策略，不是对样本进行独立的随机抽样构造训练集，而是重点关注被前面的弱分类器错分的样本或是构造样本标签值[19]。AdaBoost的构筑过程大致如下：训练时，依次训练每一个弱分类器，并得到它们的权重值。在这里，训练样本带有权重值，初始时所有样本的权重相等，在训练过程中，被前面的弱分类器错分的样本会加大权重，反之会减小权重，这样接下来的弱分类器会更加关注这些难分的样本[19]。弱分类器的权重值根据它的准确率构造，精度越高的弱分类器权重越大。

### 学习曲线和网格搜索

在机器学习模型中，需要人工选择的参数称为超参数，比如随机森林中决策时的个数。超参数选择不恰当，就会出现欠拟合或过拟合的问题。而在选择超参数的时候，有两个途径，一个是凭经验微调，另一个就是选择不同大小的参数，带入模型中，挑选表现最好的参数，一般采用学习曲线以及网格搜索进行超参数的选择。

1. 学习曲线

在一定时间内获得的技能或知识的速率，又称练习曲线。在机器学习中，学习曲线是一条关于样本个数和经验损失函数直接的曲线，对机器学习中的过程进行判断，进一步调整学习参数，提高假设模型的泛化能力。

1. 网格搜索

在指定的参数范围内，按照步长依次调整参数，利用调整的参数训练学习器，从所有的参数中找到在验证集上精度最高的参数。网格搜索是一种参数调优手段，在所有候选的参数选择中，通过循环遍历，尝试每一种可能性，表现最好的参数就是最终的结果。

### 回归评价指标

1. 交叉验证

交叉验证（Cross Validation）是一种复杂的统计准确率的技术。k折交叉验证将样本随机、均匀地分成k份，轮流用其中k-1份训练模型，1份用于测试模型的准确率，用k个准确率的均值作为最终的准确率。

决定系数，反应因变量的全部变异能通过回归关系被自变量解释的比例，用来衡量回归模型的拟合度。越接近于1，模型的拟合优度越高。被定义如下：

其中N是样本数量，u是残差平方和，v是总平方和，i代表每一个数据样本，fi是模型回归出的数值，yi是样本点i实际的数值标签，是真实数值标签的平均值。

1. MAE

平均绝对误差。通过计算预测值和真实值之间的残差的绝对值的预测，平均绝对误差能更好的反映预测值误差的实际情况。MAE被定义如下：

其中N是样本数量，i代表每一个数据样本，fi是模型回归出的数值，yi是样本点i实际的数值标签。

1. RMSE

均方根误差，是均方误差（MSE）的算术平方根。均方误差（MSE）是反应估计量与被估计量之间差异程度的一种度量。均方误差可以评价数据的变化程度，MSE的值越小，说明预测模型描述实验数据具有更好的精确度。RMSE被定义如下：

其中N是样本数量，i代表每一个数据样本，fi是模型回归出的数值，yi是样本点i实际的数值标签。

## 数据爬取

### 相关方法

1. Request库

Requests库是用Python语言基于urllib编写的，采用的是Apache2 Licensed开源协议的HTTP库。简单来说就是处理http请求，模拟浏览器向服务器发出请求。相比urllib库，Requests库更便捷，可以节约大量的工作。

1. lxml

lxml是一款高性能的基于Python的XML和HTML的解析器，其主要功能是迅速、灵活地解析和提取XML和HTML中的数据，也可以利用XPath语法，来定位特定的元素及节点信息。

1. Beautiful Soup

Beautiful Soup 是一个可以从HTML或XML文件中提取数据的Python库。它可以通过熟悉的选择器来实现常用的文档导航、查找、修改文档的方式。由于 Beautiful Soup 是基于 Python，所以相对来说速度会比另一个XPath会慢点，但是它简单易用，其功能也是非常的强大。

1. Pandas

Pandas是基于NumPy的一种开源Python库，它是为解决数据分析任务而创建的，被广泛应用于快速分析数据，以及数据清洗和准备等工作。另外，Pandas纳入了丰富的库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具。

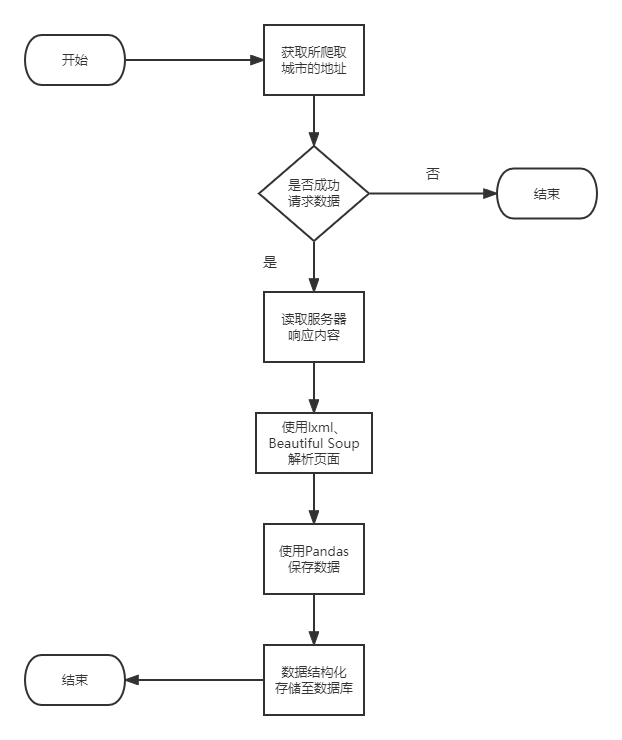
1. json

json模块是Python内置的库，主要功能是将序列化数据从文件里读取出来或存入文件。其中的dumps方法是将Python对象编码成json字符串，loads方法是将json字符串解码成Python对象。

### 处理过程

爬虫，通过编写程序，模拟浏览器发送网络请求，然后让其在互联网上抓取数据的过程。其具体的运行机制为，通过获取URL的规律来构建网址表，然后提取需要抓取的URL的数据，再予以读取、解析、爬取，将数据按指定的格式存放待用[23]。

本文所要爬取的是2345天气王网站，以爬取成都市历史天气为例进行阐述。首先，获取相应城市的历史天气的地址，使用request方法请求页面，读取服务器响应的内容，然后使用lxml、Beautiful Soup解析页面，获取相关内容，最后使用Pandas中的方法保存数据为Excel形式，将数据存储至数据库。具体流程如图5.1所示。



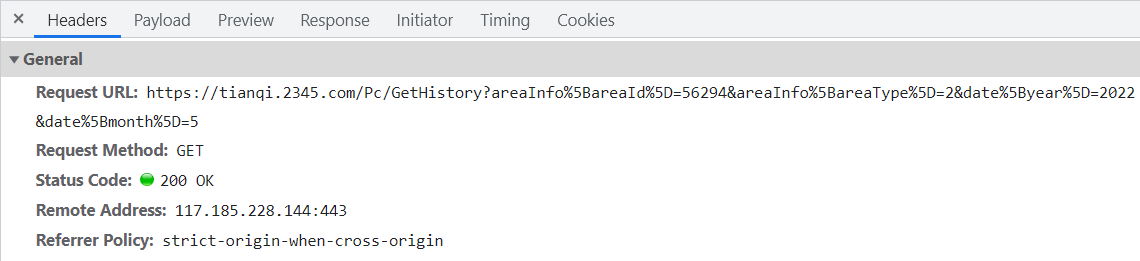
**图 5.1 爬取数据流程图**

本文爬取数据的具体过程如下。

1. 获取网址

首先，需要获取想要爬取的城市的历史天气数据的网址，为数据预处理做准备。

通过访问2345天气王网站，查询成都的历史天气，在Chrome浏览器的Network中可以发现有如下请求，如图5.2所示。从URL中可以看出请求需携带地区信息、城市ID、年份、月份。根据网站的这些信息去查找，然后拼接URL就可以得到完整的获取历史天气的地址。



**图 5.2 请求信息图**

1. 反反爬虫策略。

反反爬虫，就是爬虫程序可以通过制定相应的策略或技术手段，破解网站中具备的反爬虫机制，从而获取网站的数据。

本文使用伪装User-Agent 措施应对2345天气王网站中的反爬虫机制。User-Agent表示用户代理，是HTTP协议中的一个字段，其作用是描述发出HTTP请求的终端信息，比如操作系统及版本、浏览器及版本等，服务器通过这个字段可以知道访问网站的是谁。

User-Agent的角色就是客户端的身份标识。伪装User-Agent，这种验证请求头信息中特定头域的方式可以有效地屏蔽长期无人维护的反爬虫程序。为爬虫程序设置请求头User-Agent字段，如图5.3所示。



**图 5.3 User-Agent信息图**

1. 读取服务器响应内容

使用request方法模拟浏览器发出请求，获取服务器响应内容，如图5.4所示。



**图 5.4 原始响应内容图**

该内容属于json字符串，不利于后续的解析页面操作。为此，利用json模块中的loads方法将响应内容解码成Python对象，结果如图5.5所示。



**图 5.5 解码后响应内容图**

1. 解析页面

将上述解码获得的响应内容中的data属性的数据传入Beautiful Soup中的bs4方法，使用lxml解析器对页面进行解析，结果如图5.6所示。



**图 5.6 页面解析图**

可以发现，所需要的数据都蕴含在带有类名“history-table”的table标签中。接着借助Beautiful Soup的find方法，查找到此标签，接着使用find\_all方法查找所有的tr标签，然后获取每一个td标签的内容，提取内容后处理数据，将他们赋值在对应的列表内，页面的解析过程结束。关键代码如图5.7所示。



**图 5.7 爬取数据关键代码图**

1. Pandas保存数据

获取到数据后，使用Pandas中的DateFrame将每个列表的数据制成一张表，然后使用to\_excel方法以Excel形式保存。

1. 存储数据库

借助Navicat工具，将上述步骤中的表格数据存储至数据库，设置变量名称，类型，将数据永久保存，方便后续的平台开发操作。

数据保存至表格后，数据类型包括日期、星期、最高温度、最低温度、天气、风向、风力、空气质量指数、空气质量，数据无缺失。部分数据如表5.1所示。

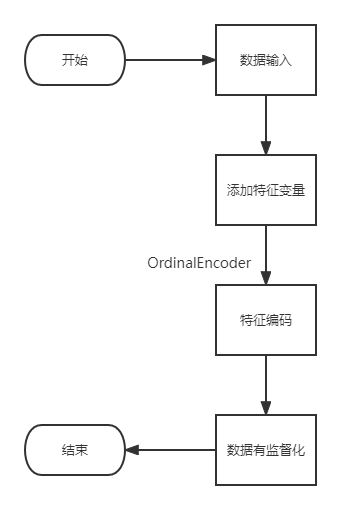
**表 5.1 成都市部分历史天气数据**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 日期 | 星期 | 最高温度 | 最低温度 | 天气 | 风向 | 风力 | 空气质量指数 | 空气质量 |
| 2019-02-27 | 周三 | 11 | 7 | 小雨 | 西北风 | 2级 | 61 | 良 |
| 2019-02-28 | 周四 | 11 | 7 | 阴~小雨 | 西北风 | 2级 | 76 | 良 |
| 2019-03-01 | 周五 | 11 | 7 | 小雨 | 西北风 | 2级 | 53 | 良 |
| 2019-03-02 | 周六 | 15 | 7 | 多云~晴 | 西北风 | 2级 | 29 | 优 |
| 2019-03-03 | 周日 | 16 | 8 | 多云 | 西北风 | 2级 | 49 | 优 |
| 2019-03-04 | 周一 | 16 | 4 | 小雨 | 东北风 | 2级 | 71 | 良 |
| 2019-03-05 | 周二 | 17 | 10 | 多云~小雨 | 西北风 | 2级 | 59 | 良 |
| 2019-03-06 | 周三 | 15 | 9 | 小雨 | 东北风 | 2级 | 54 | 良 |
| 2021-08-23 | 周一 | 34 | 21 | 阴~晴 | 南风 | 2级 | 44 | 优 |
| 2021-08-24 | 周二 | 32 | 19 | 多云~阵雨 | 东南风 | 1级 | 59 | 良 |
| 2021-08-25 | 周三 | 23 | 18 | 阴~阵雨 | 东北风 | 2级 | 43 | 优 |
| 2021-08-26 | 周四 | 23 | 19 | 阴 | 西南风 | 2级 | 24 | 优 |
| 2021-08-27 | 周五 | 27 | 19 | 多云~阵雨 | 南风 | 2级 | 30 | 优 |
| 2021-08-28 | 周六 | 22 | 20 | 中雨~阵雨 | 南风 | 1级 | 24 | 优 |
| 2021-08-29 | 周日 | 29 | 20 | 阴~多云 | 南风 | 1级 | 31 | 优 |
| 2021-08-30 | 周一 | 28 | 22 | 雾~阵雨 | 东南风 | 2级 | 32 | 优 |
| 2021-08-31 | 周二 | 27 | 22 | 雾~阴 | 东北风 | 1级 | 42 | 优 |
| 2021-09-01 | 周三 | 30 | 22 | 雾~阴 | 东北风 | 1级 | 45 | 优 |

## 实验与分析

### 数据预处理

本文实验数据来源为2345天气王网站，是关于成都市2016年1月至2022年2月11日的历史天气数据。对于采集到的原始数据，存在特征不足、分类型特征变量未编码等缺点，而时间序列数据做回归分析时，需要先转化为有监督的数据。因此，本文需要进行数据预处理过程才能建模。数据预处理过程主要分为三步：添加特征变量、特征编码、数据有监督化，如图5.8所示。



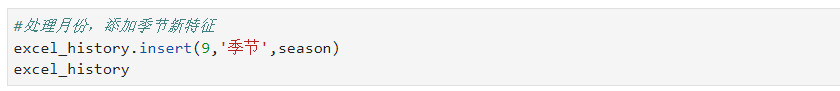
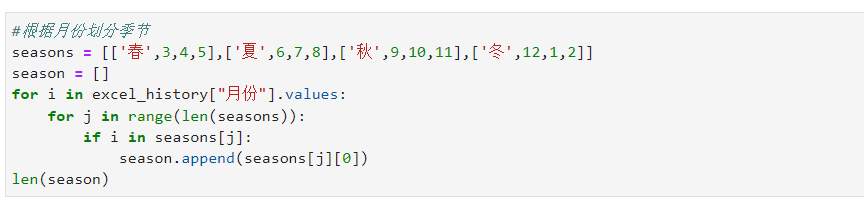
**图 5.8 数据预处理过程**

1. 添加特征变量

众所周知，月份、季节会影响天气的状况，为后续建立模型提高预测准确率，将这两个因素添加为特征变量。关键代码如图5.9，图5.10所示。

****

**图 5.9 添加月份特征变量**

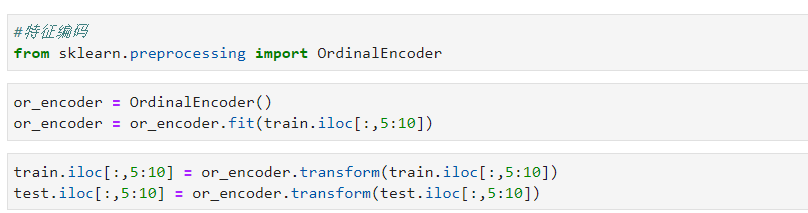
****

**图 5.10 添加季节特征变量**

1. 特征编码

特征编码，对特征变量进行编码的工作过程。在机器学习中，大多数算法，譬如随机森林、AdaBoost等算法都只能够处理数值型的数据，不能处理文字型数据。为了让数据适应算法和库，本文将数据进行编码工作。

首先，将原始数据分为两部分，第一部分是2016年1月至2020年12月的数据，将其作为训练集用来构建模型；第二部分是2021年1月至2022年2月11日的数据，将其作为测试集用来验证模型的预测准确率。其次，因为现实中测试集数据都是随时可以收集的，所以，本文利用sklearn中的OrdinalEncoder方法先训练训练集中的分类型的特征变量的数据，本质是将训练集中已经存在的类别转换成数字，然后我们再使用接口transform分别在测试集和训练集上来编码我们的特征矩阵。当使用接口在测试集上进行编码的时候，如果测试集上出现了训练集中从未出现过的类别，那代码就会报错，测试集上或许存在错误值，异常值，曾经的训练数据中并没有这个类别。以此为基础，调整模型。具体操作如图5.11所示。

****

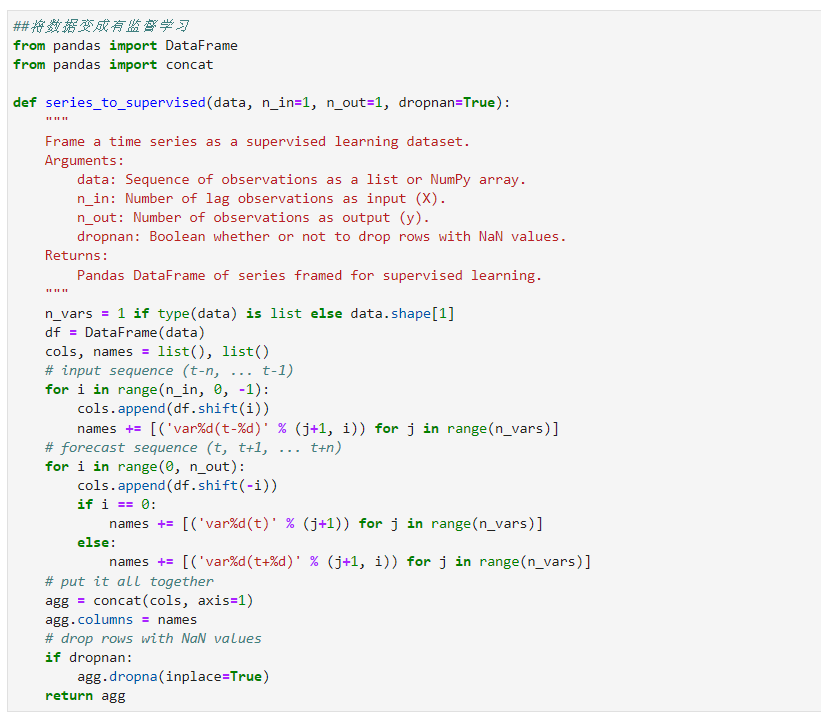
**图 5.11 特征变量编码**

1. 数据有监督化

监督学习是具有输入变量（x）和输出变量（y）的问题，并且可以使用算法来学习x到y的映射函数y = f(x)。算法的目标可以很好地近似真实的映射关系，以便有新的输入数据x时，可以预测该数据的输出变量y。

时间序列，也叫时间数列、历史复数或动态数列。它是将某种统计指标的数值，按时间先后顺序排到所形成的数列。时间序列预测就是通过编制和分析时间序列，根据时间序列所反映出来的发展过程、方向和趋势，进行类推或延伸，借以预测下一段时间或以后若干年内可以达到的水平。时间序列预测可以被认为是监督学习问题。只需要对数据进行转换，重新构建时间序列数据，使其转变为监督学习即可。

将训练集与测试集中的最高温度与最低温度分别转换为有监督的数据，将最高温度与最低温度转化为过去五天的温度特征变量与未来三天的温度标签，调用如下图的方法即可，关键代码如图5.12所示。

****

**图 5.12 数据有监督化**

最后，数据预处理过程结束，将数据进行拼接，其中，天气、风向、风力、空气质量、空气质量指数、季节、月份、过去第五天最高（低）温度、过去第四天最高（低）温度、过去第三天最高（低）温度、过去第二天最高（低）温度、过去第一天最高（低）温度作为预测最高温度与最低温度模型的输入属性，最高（低）温度、未来第一天最高（低）温度、未来第二天最高（低）温度作为输出属性。部分处理后的数据如图5.13，图5.14所示。



**图 5.13 训练集最高温度数据**



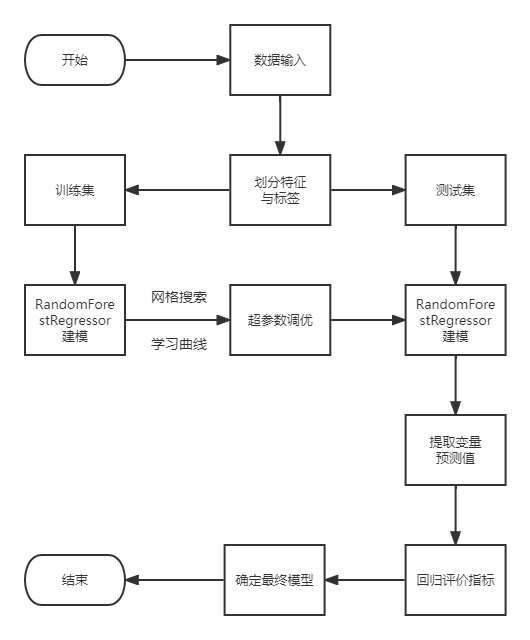
**图 5.14 训练集最低温度数据**

### 气温预测模型建立

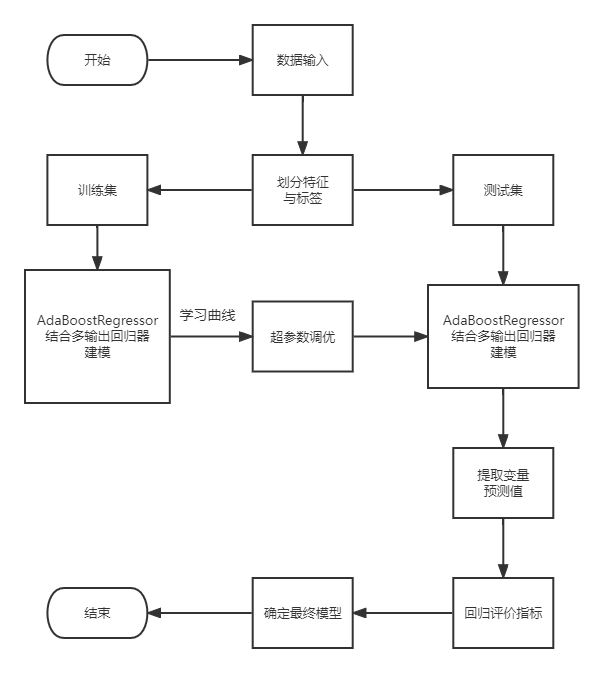
本文研究了对成都市未来三天气温的预测，将上面提出的12个天气要素作为输入，3个天气要素作为输出，建立最高温度模型和最低温度模型。

为验证本文随机森林算法在气温预测模型上面的优势，将本文算法与其他气温预测模型作对比，分别进行了最高温度与最低温度的预测，其他模型是基于AdaBoost算法的气温预测模型。

在实验中采用10折交叉验证方法将训练集随机分为10份，其中9份作为训练，1份作为测试，将10次试验结果的平均值作为最终的评价的标准。使用学习曲线和网格搜索确定各算法的超参数，应用、MAE、RMSE三个评估指标来检验模型效果，从而确定最适合气温预测的模型。两个算法建立气温预测模型过程分别如图5.15，图5.16所示。



**图 5.15 随机森林模型构建过程**



**图 5.16 AdaBoost模型构建过程**

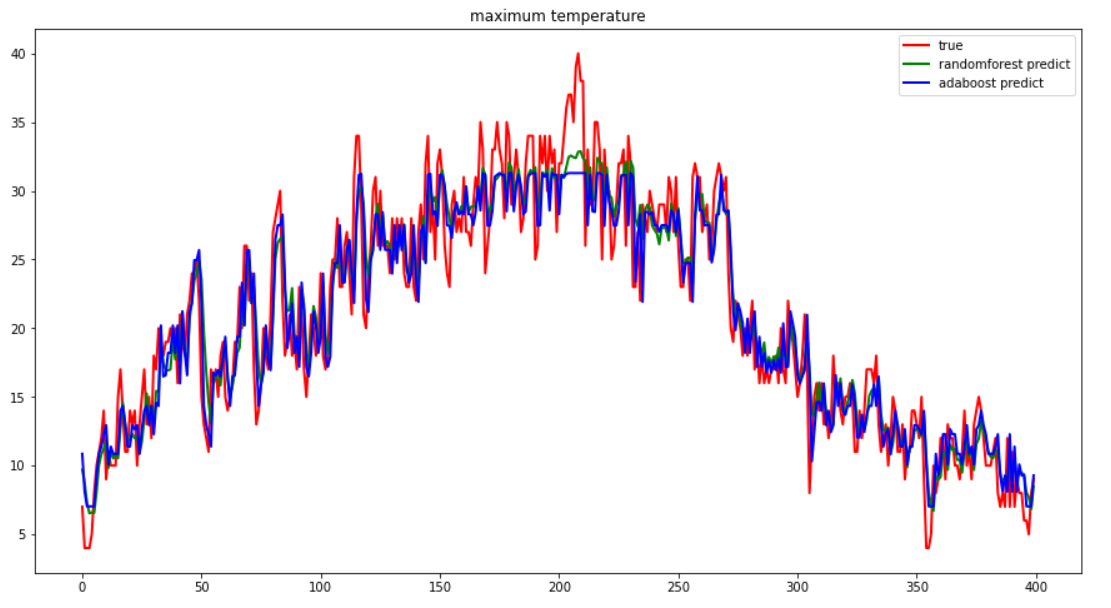
在模型建立，超参数调优的过程中，确定了最终的模型，结果如下。

随机森林算法最高温度模型的参数分别为n\_estimators=467，max\_features=5，min\_samples\_leaf=9，min\_samples\_split=35，random\_state=420。随机森林算法最低温度模型的参数分别为n\_estimators=134，max\_depth=8，max\_features=5，min\_samples\_leaf=11，min\_samples\_split=27，max\_leaf\_nodes=49，random\_state=420。

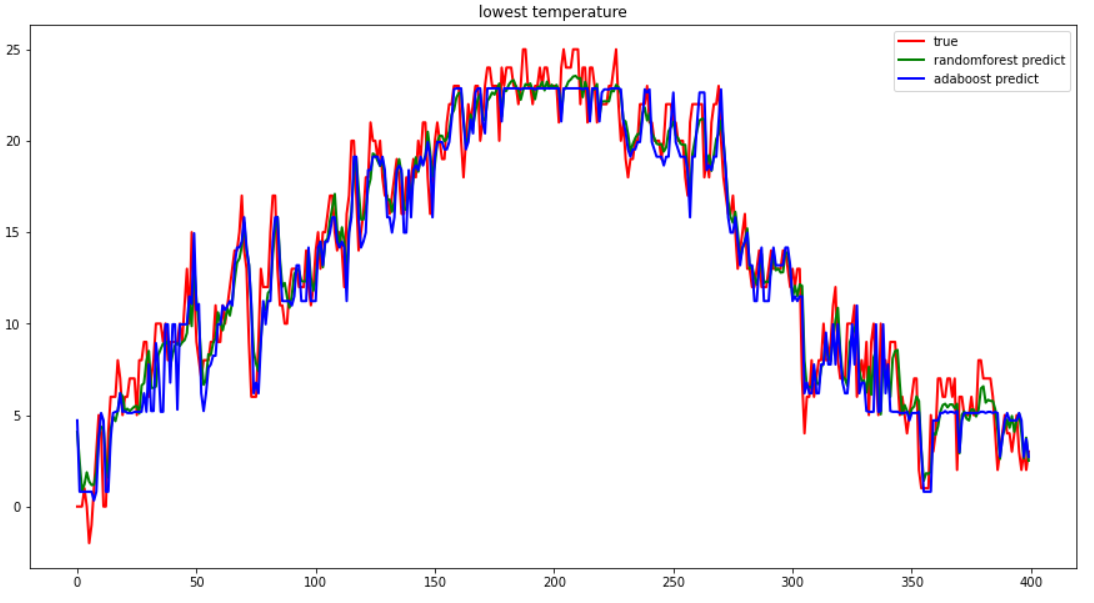
AdaBoost算法最高温度模型的参数分别为n\_estimators=15，learning\_rate=0.45，loss='exponential'，random\_state=420。AdaBoost算法最低温度模型的参数分别为n\_estimators=9，learning\_rate=1，loss=linear，random\_state=420。

### 实验结果与对比分析

在实验过程中，使用最高（低）温度模型在测试集上预测出结果，借助matplotlib库，使用预测结果中的第一列数据绘制了真实值、随机森林预测值、AdaBoost预测值数据拟合对比图，可得出结论：使用随机森林构建的模型在数据上的拟合程度比使用AdaBoost构建的模型拟合程度高。对比图如图5.17最高温度图,图5.18最低温度图所示。红色线条代表测试集的真实值，绿色线条代表使用算法构建模型的预测值，蓝色线条代表使用AdaBoost算法构建模型的预测值。



**图5.17最高温度对比图**



**图5.18最低温度对比图**

本文的模型构建结束，应用、MAE、RMSE进行模型的检验与评估，其结果如表5.2所示。通过各模型在测试集上的预测表现，本文可以得出如下结论：在气温预测中，随机森林在测试集上的表现优于AdaBoost。

**表 5.2 各模型检验评价结果**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型 | 数据集 |  | MAE | RMSE |
| 随机森林最高温度 | 测试集 | 0.826439 | 2.768079 | 3.473466 |
| 随机森林最低温度 | 测试集 | 0.918554 | 1.568638 | 1.995579 |
| AdaBoost最高温度 | 测试集 | 0.820671 | 2.817229 | 3.530817 |
| AdaBoost最低温度 | 测试集 | 0.896680 | 1.623141 | 2.027428 |

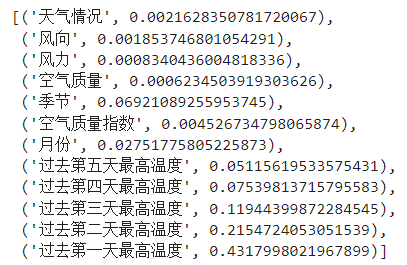
在使用随机森林算法和AdaBoost算法构建气温预测模型的过程中，发现了二者的异同。具体区别如表5.3所示。

**表 5.3 两种算法对比结果**

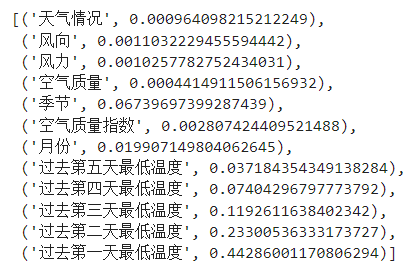
|  |  |  |
| --- | --- | --- |
| 比较项目 | 随机森林 | AdaBoost |
| 决策树规模 | 大 | 小 |
| 是否对样本随机采样 | 是 | 否 |
| 是否对特征随机采样 | 是 | 否 |
| 弱分类器是否有权重 | 无 | 有 |
| 训练样本是否有权重 | 无 | 有 |
| 是否支持多输出回归 | 是 | 不直接支持 |

随机森林与AdaBoost算法都是通过构造不同的样本集训练多个弱分类器，前者通过样本抽样构造不同的训练集，后者则通过给样本加上权重构造不同的样本集。随机森林中的决策树不能太简单，过于简单的决策树会导致随机森林精度低。AdaBoost即使使用深度很低的决策树，将他们集成起来也能得到较好的精度。

不过，由于AdaBoost不直接支持多输出回归问题，在本文的气温预测研究中使用了多输出回归器与AdaBoost结合完成了对比试验的预测。并且，二者的结合导致使用AdaBoost并不能查看模型的特征变量的重要性，图5.19，图5.20为使用随机森林构建模型的特征变量的重要性。



**图 5.19 随机森林最高温度模型特征重要性**



**图 5.20 随机森林最低温度模型特征重要性**

特征重要性指的是为预测建模中的输入特征分配分数的一种技术，即表示每个特征在进行预测时的相对重要性。特征重要性得分在预测建模具有重要的作用：相对得分可以突出哪些特征与目标最相关以及哪些特征与目标最不相关，通过对重要性得分检查，可以深入了解特定模型以及在进行预测时，哪些特征对模型来说最重要，这可以作为模型支持他的模型解释。另外，重要性得分为降维和特征选择提供依据，从而提高预测模型的效率和有效性。

综上所述，在本实验中得出了适用于本文气温预测的算法——随机森林，以随机森林算法构建的两个模型具有较高的预测精度和较强的泛化能力。并且，使用随机算法构建的最高气温模型与最低气温模型的评价结果如下：最高气温模型的为0.826439，MAE为2.768079，RMSE为3.473466。最低气温模型的为0.918554，MAE为1.568638，RMSE为1.995579。

## 本章小结

本章主要围绕基于随机森林的气温预测模型的建立展开。首先介绍了本章所有的方法，包括AdaBoost算法，学习曲线、网格搜索、回归评估指标。其次描述了数据预处理的过程，主要包括添加特征变量、特征编码、数据有监督化。最后，在气温模型的建立过程中，使用AdaBoost与随机森林作对比，实验结果表明，在气温预测中，随机森林在测试集上的表现优于AdaBoost，得出了适用于本文气温预测的算法是随机森林。

# 基于React的气候可视化平台的实现

## 开发环境配置

操作系统：Windows

前端开发环境：Visual Studio Code 1.66.1，Node 16.14.0

后端开发环境：Python 3.9.5，PyCharm Community Edition 2021.3.3

数据库：MySQL 5.0

## 数据准备

本平台的数据来源有两部分，一部分来自于2345天气王网站，使用上章中数据爬取的方法将部分数据存储在了数据库，用于历史天气功能的实现。一部分来自于和风天气平台，调用此平台的数据接口在本平台的前端用作气候可视化功能的实现。

## 平台的功能实现

### 天气播报

天气播报实现了天气即时显示、生活指数、气温预测，整体功能界面如图6.1所示。

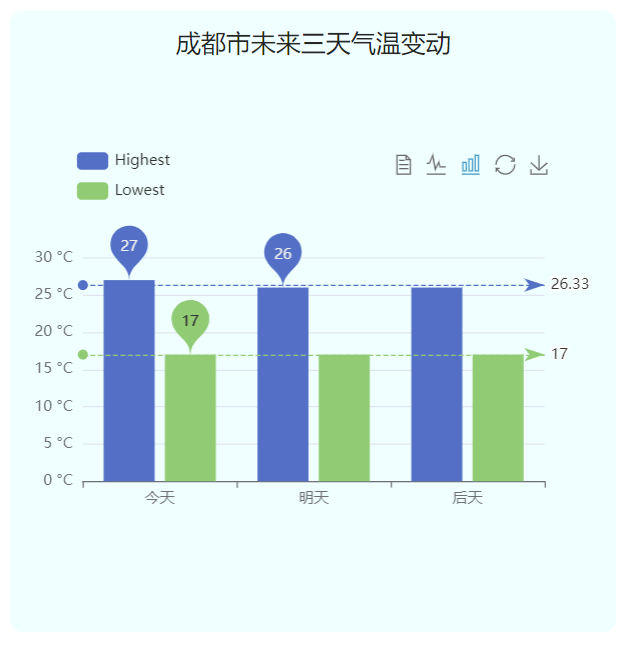


**图 6.1 天气播报界面图**

气温预测使用ECharts中的折线图与柱状图进行实现的。在天气播报页面挂载结束时，选择绘制图表的容器，配置图表的位置、大小、横轴、纵轴、工具盒以及相关的样式，关键代码如图6.2所示。点击数据视图图标，可以进行数据的预览，点击柱状图图标，可以显示以柱状图表示的数据，如图6.3所示。点击折线图图标，可以转换回折线图形式。点击下载图标，可以将数据以图片的形式保存，便于时时查看。

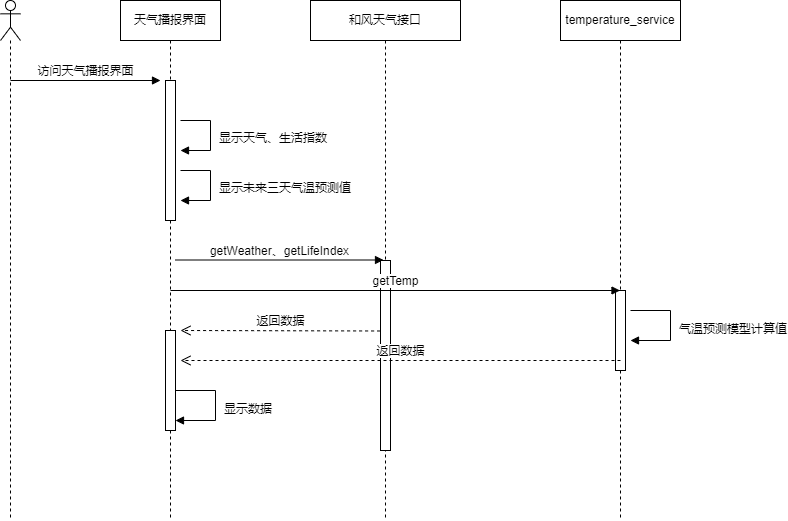
 

**图 6.2 绘制气温预测图表关键代码**



**图 6.3 气温预测柱状图与书籍视图界面**

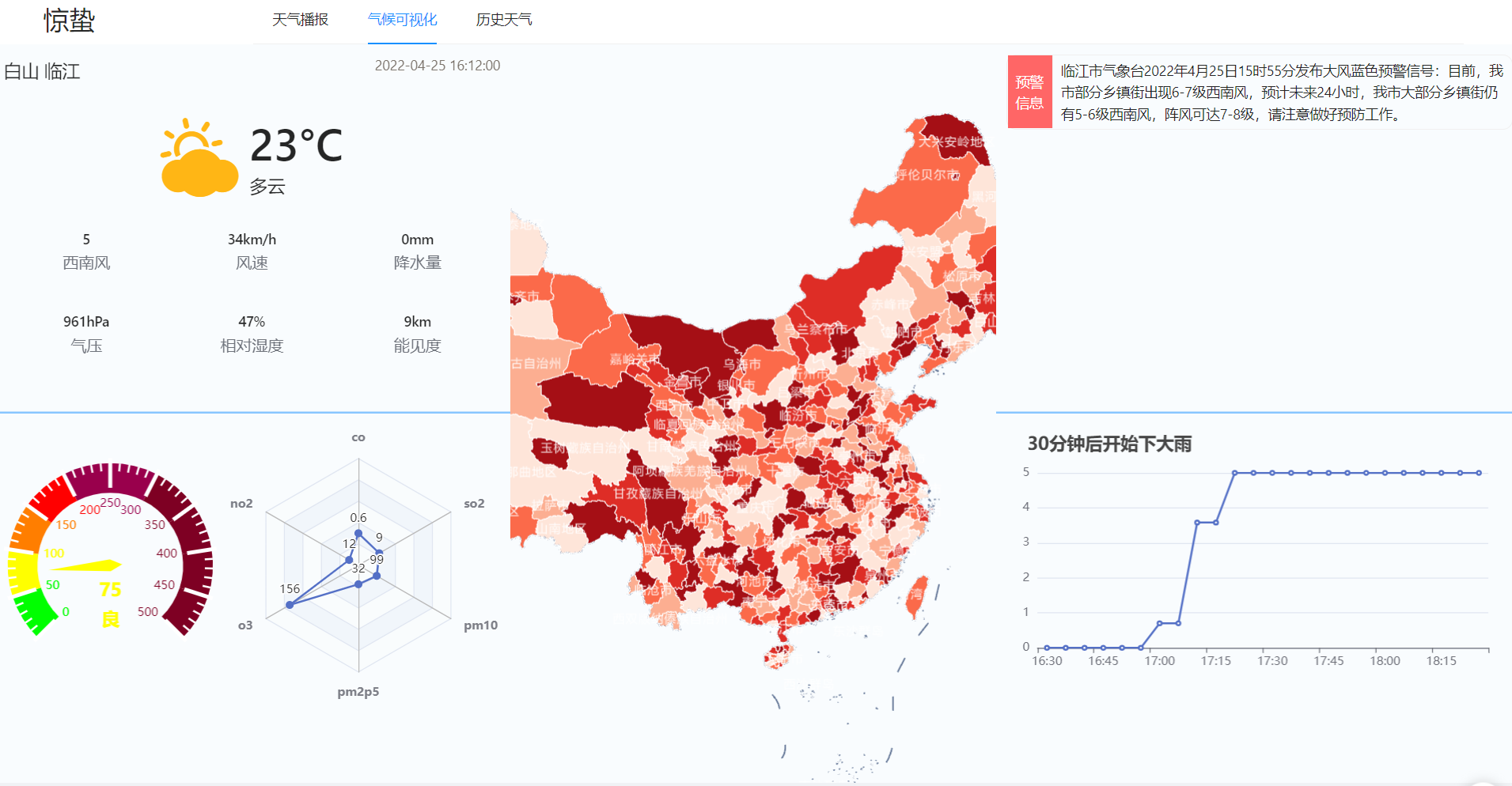
天气播报时序图如图6.4所示。



**图 6.4 天气播报时序图**

### 气候可视化

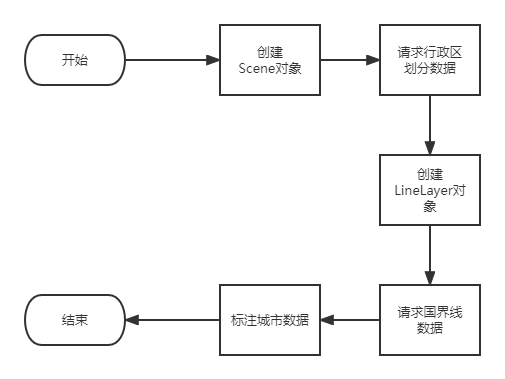
该功能主要通过ECharts和L7实现的，负责向用户提供天气、空气质量、灾害预警信息、降水可视化服务，用户通过选择地图中的具体地点，获取相应的气候数据展示在各可视化图表中，整体功能界面如图6.5所示。



**图 6.5 气候可视化界面图**

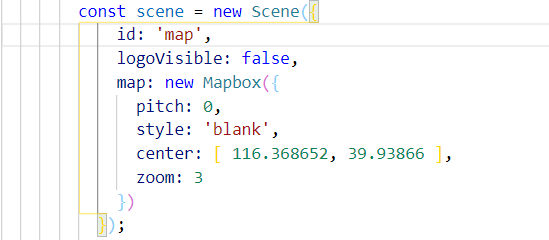
1. 中国地级市地图可视化

此功能是使用L7实现的，用户通过点击城市，可获取城市的经纬度，用于后续气候数据可视化的操作，并且编写代码实现了地图可拖拽、缩放的功能。整体流程图如图6.6所示。



**图 6.6 中国地级市地图可视化流程图**

首先在整体页面挂载完成后，创建Scene对象，进行地图的配置，包括容器的id，容器的中心位置、主题颜色，关键代码如图6.7所示。



**图 6.7 配置Scene对象**

当Scene对象加载时，发送请求，获取行政区数据，创建图层，配置图层的尺寸、颜色、样式等，然后在图层中显示数据，关键代码如图6.8所示。



**图 6.8 获取行政区数据**

接着继续发送请求，获取国界线数据，创建新的图层，配置图层大小、颜色、样式等，然后将数据放入图层显示，关键代码如图6.9所示。



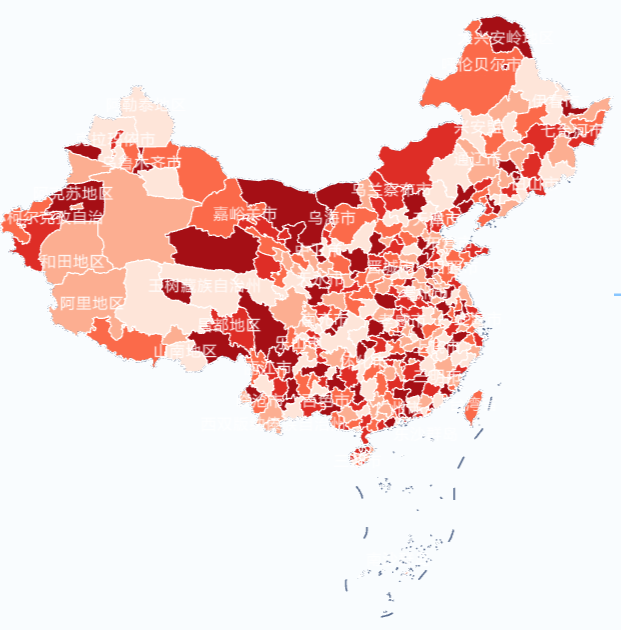
**图 6.9 获取国界线数据**

最后请求城市数据，然后采用不同样式将数据标注在地图上，地图可视化过程结束，关键代码如图6.10所示。



**图 6.10 标注数据**

中国地级市地图绘制的效果如图6.11 所示。



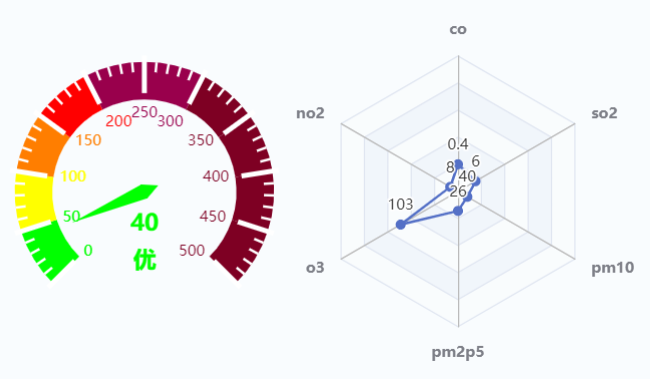
**图 6.11 中国地级市地图**

1. 空气质量可视化

实现空气质量可视化时，考虑到空气质量指数等级不同会有不同的表示方式，通过查看中华人民共和国国家环境标准文档中的环境空气质量指数（AQI）日报技术规定，将空气质量指数分为了6个级别，并且根据文档中的空气质量指数及表示的颜色绘制空气质量的可视化图表，区分等级颜色关键代码如图6.12所示。采用仪表盘绘制空气质量指数图表，采用雷达图绘制空气质量元素浓度，功能界面如图6.13所示。



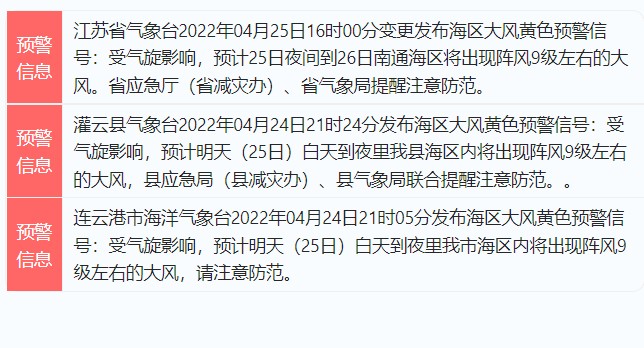
**图 6.12 空气质量指数关键代码**



**图 6.13 空气质量可视化界面**

1. 预警信息可视化

本文使用表格描述性列表的形式，展示了该地区的气象台具体时间发布的预警信息，并且提醒用户居民注意防范。如果此地区存在灾害预警信息，则会如图6.14所示，如果此地区不存在预警信息，则会提示信息为空，如图6.15所示。



**图 6.14 灾害预警信息界面**



**图6.15 灾害预警信息为空界面**

1. 分钟级降水可视化

使用了折线图绘制地区未来2小时每5分钟的降水量情况，并且给出相关的未来两小时降水概况信息，用户可根据降水量和概况信息调整策略，做好相应措施。功能界面如图6.16所示。



**图 6.16 分钟级降水可视化界面**

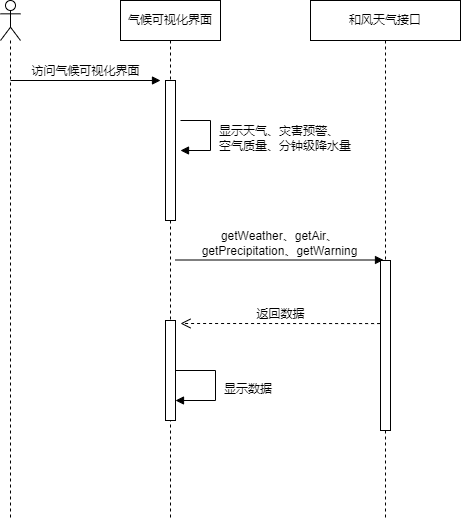
1. 天气可视化

天气可视化会显示地区即时天气情况，包括温度，风向风力、风速、大气压强、能见度、相对湿度。用户可以根据天气实况做出生活的调整。功能界面如图6.17所示。



**图 6.17 天气可视化界面**

气候可视化时序图如图6.18所示。

****

**图 6.18 气候可视化时序图**

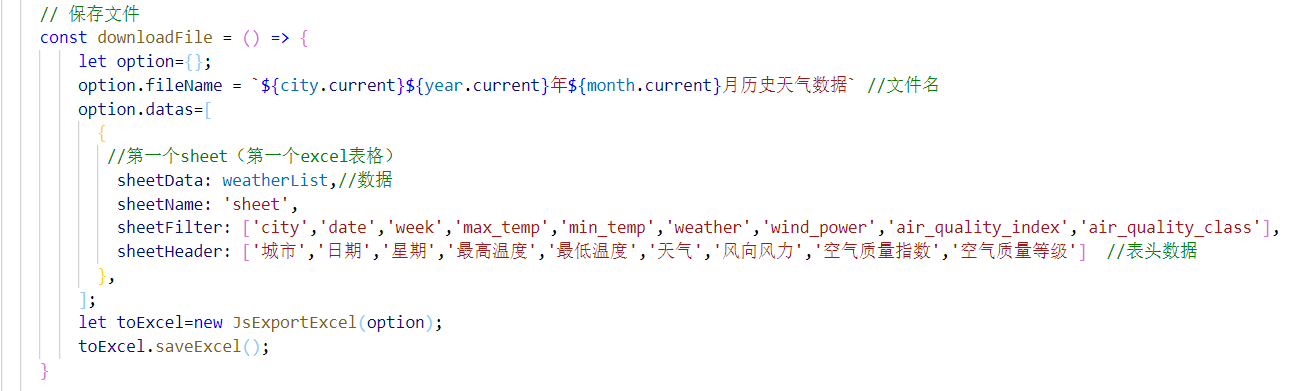
### 历史天气

该功能模块中用户可以手动选择城市、年份、月份，平台会根据前端传入的数据展示相关的平均温度信息、平均空气质量信息、历史天气信息，并且统计分析了历史年月温度，以图表进行绘制。查询到的历史天气信息使用表格进行绘制，并且添加了分页处理。另外用户也可点击下载按钮，保存天气信息为表格的形式，方便用户使用数据。整体功能界面如图6.19所示。



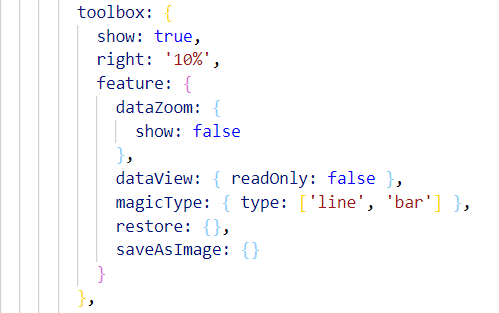
**图 6.19 历史天气界面图**

文件下载功能使用了js-export-excel库，只需为即将保存的文件设置文件名、保存的数据、表头即可，操作简捷，关键代码如图6.20所示。

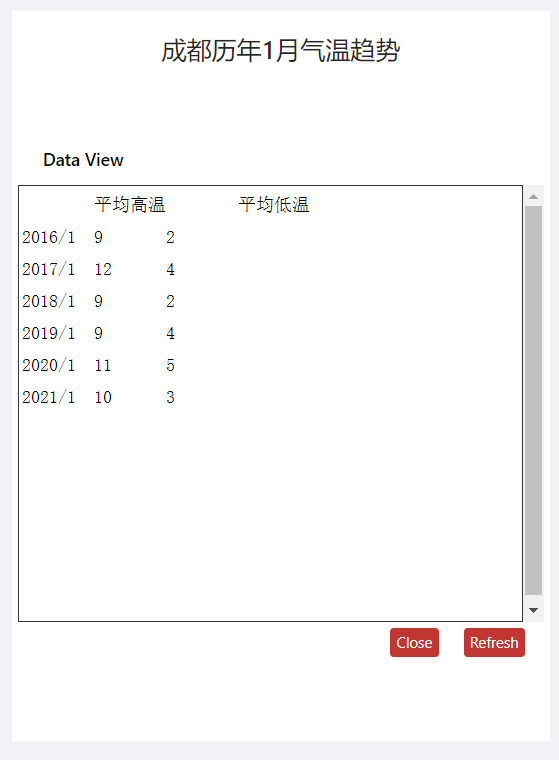
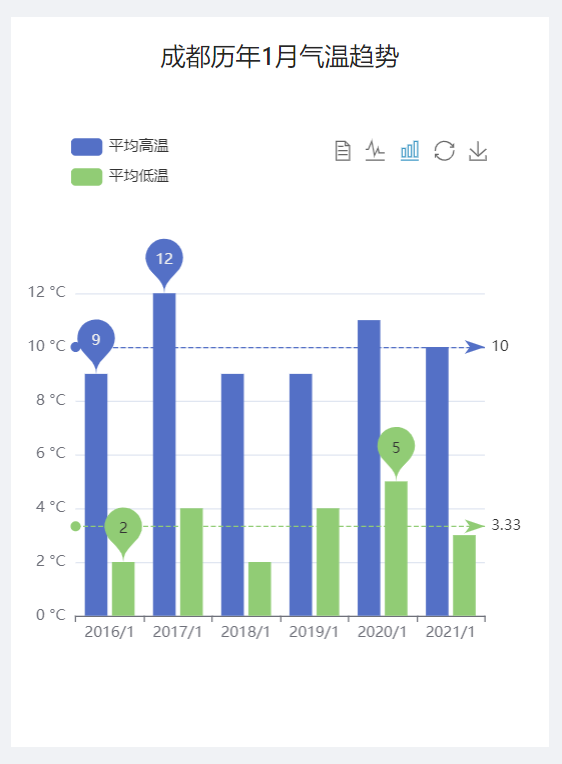


**图 6.20 文件下载关键代码**

历史年月气温趋势图提供了图例切换、数据视图、刷新、保存为图片的功能，关键代码如图6.21所示。气温趋势柱状图与数据视图界面如图6.22所示。

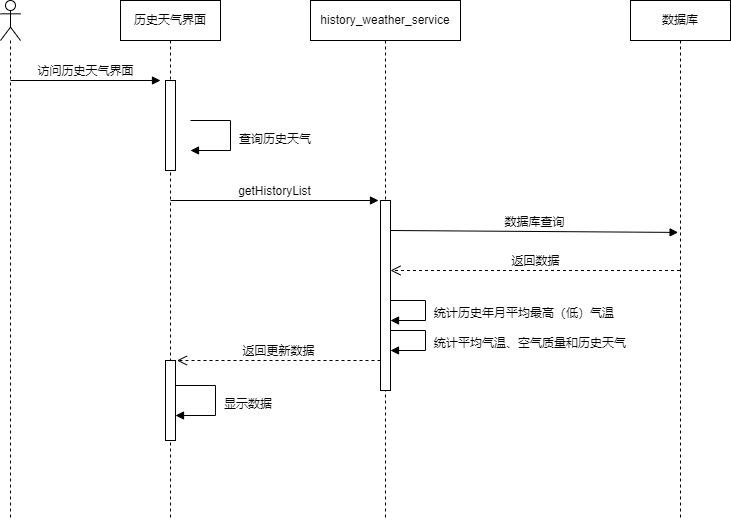


**图 6.21 气温趋势图关键代码**



**图 6.22 气温趋势柱状图与数据视图界面**

历史天气时序图如图6.23所示。

****

**图6.23 历史天气时序图**

## 系统功能的测试

### 软件测试

软件测试指的是通过自动或人工的手段来运行某个软件系统，并对其运行过程进行综合测定[24]。软件测试就是在软件投入运行之前，对软件需求分析、设计规格说明和编码实现的最终审查，它是软件质量保证的关键步骤。软件测试的目的是保证软件产品的最终质量，在软件开发的过程中，对软件产品进行质量控制，制定测试计划、测试方案、测试规范，实施测试，尽可能地寻找出软件的缺陷，然后予以修正，对测试记录进行分析。

### 测试的结果

软件测试是为了证明程序中有错误，而不是证明程序中无错误[25]。为了尽快尽早地发现在软件产品中所存在的各种问题，编写了本系统各功能模块的测试用例，采用黑盒测试方法验证测试用例。

天气播报模块的测试用例如表6.1所示。

表 6.1 天气播报测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **测试编号** | **测试功能** | **预期结果** | **是否符合预期** |
| 1 | 即时天气 | 显示相关天气 | 是 |
| 2 | 生活指数 | 显示生活建议 | 是 |
| 3 | 气温预测 | 显示气温趋势图 | 是 |
| 4 | 保存图片 | 气温图保存在本地 | 是 |

气候可视化模块的测试用例如表6.2所示。

表 6.2 气候可视化测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **测试编号** | **测试功能** | **预期结果** | **是否符合预期** |
| 1 | 中国地级市地图可视化 | 地图可以点击拖拽 | 是 |
| 2 | 空气质量可视化 | 显示空气质量指数和浓度 | 是 |
| 3 | 分钟级降水可视化 | 显示未来2小时降水量 | 是 |
| 4 | 天气可视化 | 显示天气信息 | 是 |
| 5 | 预警信息可视化 | 有内容则显示，无内容提示 | 是 |

历史天气模块的测试用例如表6.3所示。

表 6.3 历史天气测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| **测试编号** | **测试功能** | **预期结果** | **是否符合预期** |
| 1 | 查询历史天气 | 显示历史天气、平均天气信息 | 是 |
| 2 | 文件下载 | 文件下载到本地 | 是 |
| 3 | 历史年月气温统计 | 显示气温趋势图 | 是 |
| 4 | 保存图片 | 历史气温趋势图保存图片在本地 | 是 |

本平台根据黑盒测试方法验证所有的测试用例并进行了平台后期维护，确保了所有的功能都实现。本平台最终测试的结果如表6.4所示。

表 6.4 平台功能测试结果

|  |  |  |
| --- | --- | --- |
| **序号** | **功能** | **结果** |
| 1 | 天气播报模块 | 通过 |
| 2 | 气候可视化模块 | 通过 |
| 3 | 历史天气模块 | 通过 |

## 本章小结

本章节主要根据平台设计进行了平台功能详细设计和实现，并且利用图表展示了主要功能效果，最后对该平台进行了功能测试。

# 总结与展望

本课题的研究是基于React的气候可视化平台的设计与实现，完成了天气播报、气候可视化、历史天气等功能，解决了面对高频次、庞大的气候数据，传统的表达方式难以有效呈现数据背后所蕴含的信息内容的问题。

针对于气温预测的研究，本文是基于成都市历史天气数据，利用随机森林算法和AdaBoost算法进行对比实现，分别构建两个气温预测模型：最高气温模型与最低气温模型，预测成都市未来三天最高气温与最低气温。为建立两个气温模型，首先进行数据预处理，添加特征变量、特征编码、数据有监督化。其次，使用学习曲线和网格搜索寻找超参数，采用交叉验证评估模型预测的准确率，利用相关的回归评价指标判断模型的表现。实验结果表明，随机森林算法适用于本文的气温预测。

针对于气候可视化的研究，本文采用了ECharts、L7等可视化技术，完成了中国地级市地图可视化、分钟级降水可视化、空气质量可视化等，借助可视化的力量，使用户能与抽象化的气候数据直接交互，使之清晰化、生动化，提高了界面的生动性、美观性。

其实，在本课题的研究中还有其他方面未能涉及，同时也有许多优化的地方。在气温预测方面，可以尽可能收集挖掘其他城市的历史天气数据进行数据预处理，对于数据的预处理可以努力探寻其他的与气温有关的特征变量，在模型的构建过程中，可以使用深度学习的算法，或是进行算法的融合。在平台的实现方面，未来可以添加一些新的功能，例如动态、讨论帖，也可将平台实现移动端，增强用户黏度。也可以根据不同受众人群增添不同气象功能，还可以在用户体验方面进行优化，增加访问量。

# 致谢

行文至此，落笔为终，意味着四年的本科生涯即将闭幕。始于2020年初秋,终于2022年盛夏，点滴生活、满眼繁华，目光所及、皆是回忆，心之所向，也为过往。在这充满活力的校园里，留下的是青春痕迹和沉甸甸的收获。思绪繁杂，感慨万千，词穷致谢，唯有感恩。

桃李不言，下自成蹊。首先，特别感谢我的论文指导老师，他学识渊博，视野广阔，态度严谨，和蔼可亲。从论文的选题到体系的建立，从观点推敲到字字斟酌，每一步都离不开老师的悉心指导和无私帮助。其次，要感谢各位授课老师的悉心教导，使我度过了受益匪浅的四年。饮其流者怀其源，学其成时念吾师，学生深感遇良师不易，再次衷心感谢所有老师的谆谆教诲!

二十余载，含辛茹苦。感谢我的父母，这二十余载的疼爱、养育，带给了我内心的坚强与温暖，成为我焦虑沮丧时候的一束光。感谢你们一直以来的爱与支持，感谢你们在我每一个开心与不开心日子里的陪伴。养育之恩，无以为报，唯有不断成为你们的骄傲。

山水一程，三生有幸。感谢大学里遇见的舍友们和同学们，我们一同欢笑一同前进，我们风雨同舟，苦乐共济。与你们相处的日子快乐又充实，你们的优秀与个性都深深吸引着我。聚是一团火，散是满天星，庆幸相遇，无憾别离。

以梦为马，不负韶华。感谢坚定勇敢、不曾放弃的自己，从高中升入本科，再到拿到互联网大厂录用书，这一路走来实属不易。我仍觉那无数个奋笔疾书的日夜，无数个自我治愈的瞬间，无数个含泪坚持的时刻，都在不经易间汇聚成生命的宽度。

纵有千古，横有八方，前途似海，来日方长。山水相逢，终有一别。再见，我的新都大道8号，我们后会有期!

承蒙时光不弃，感恩一切经历。路漫漫其修远兮，吾将上下而求索。

# 参考文献

闫佳逸,马建芬.气象数据的跨行业应用实践[J].农业与技术,2018,38(10):236-236。

张永宁.数据可视化在气候传播中的价值及策略[J].传播力研究,2019,3(30):7-8.

Haidar A, Verma B. Monthly Rainfall Forecasting Using One-Dimensional Deep Convolutional Neural Network[J]. IEEE Access, 2018:69053-69063.

Pérez-Vega A, Travieso C M, Hernandez-Travieso J G, et al. Forecast of temperature using support vector machines[C]//2016 International Conference on Computing, Communication and Automation (ICCCA). IEEE, 2016:388-392.

Joshi P, Ganju A. Maximum and minimum temperature prediction over western Himalaya using artificial neural network[J]. Mausam, 2012,63(2):283-290.

Lathifah S N , Nhita F , Aditsania A , et al. Rainfall Forecasting using the Classification and Regression Tree (CART) Algorithm and Adaptive Synthetic Sampling (Study Case: Bandung Regency)[C]// 2019 7th International Conference on Information and Communication Technology (ICoICT). 2019.

朱晶晶,赵小平,吴胜安,吴慧,邢彩盈.基于支持向量机的海南气温预测模型研究[J].海南大学学报：自然科学版,2016,34(1):40-44.

许倡泷.基于改进指数平滑算法的气温预测研究[J].科学技术创新,2018,0(2):29-31

朱国栋,孙建杰,陈阳权,王照刚.随机森林方法在机场温度预测中的应用[J].气象科技进展,2021,11(4):34-36.

白雪.基于时间序列模型和XGBoost组合的气温预测与高温热浪预警分析[D].南京信息工程大学,2021.

陶晔,杜景林.基于随机森林的长短期记忆网络气温预测[J].计算机工程与设计,2019,40(3):737-743.

胡玉杰.基于微服务的气象数据处理与可视化平台研究[D].南京信息工程大学,2021.

吴焕萍.气象信息可视化会议简介[J].气象科技合作动态,2016,0(3):17-21.

李涛,冯仲科,孙素芬,程文生.基于Hadoop的气象大数据分析GIS平台设计与试验[J].农业机械学报,2019,50(01):180-188.

李成渊,俞越,彭伟明.大数据环境下北京冬奥、杭州亚运历史同期气候可视化设计——以数据抓取部分为例[J].江苏科技信息,2019,36(31):75-77.

黎颖智,李勇,李有华,史彩霞.基于ECharts的库区水文及气象数据监测系统的设计与应用[J].气象研究与应用,2021,42(2):68-72.

陈东辉,高峰,刘娜,韩瑞,罗岚心.气象台站历史沿革信息检索可视化系统设计与实现[J].计算机应用,2021,41(S01):119-124.

白雪莹.气象数据可视化表达研究与分析[J].科技传播,2017,0(12):58-5961.

雷明.机器学习 原理、算法与应用[M].北京：清华大学出版社,2019.

郝飞,郝海,郝强,郝泽东.公共气象服务中基础气象数据的开发与应用[J].南方农机,2018,49(2):210-210.

区永光.大数据在气象服务中的应用研究[J].科技风,2020(33):78-79.

陈志泊主编；王春玲，许福，范春梅编著.数据库原理及应用教程 第3版[M].北京：人民邮电出版社,2014.

于学斗,柏晓钰.基于Python的城市天气数据爬虫程序分析[J].办公自动化,2022,27(7):10-139.

丁凝.软件测试过程的质量控制分析[J].网络安全技术与应用,2021(5):65-66.

窦万峰主编；李亚楠，潘媛媛，林燕平参编.软件工程方法与实践 第3版[M].北京：机械工业出版社,2016.