

数据驱动的智慧校园分析系统设计与实现

王曼^{1, #}, 牛梦欣^{1, #, *}, 徐冰^{1, #}, 周青峰²

¹ 河南工业大学, 河南郑州, 中国

² 科大讯飞股份有限公司, 安徽合肥, 中国

共同贡献

* 通讯作者

【摘要】针对目前学生各种行为数据分散、异常状态发现滞后以及管理决策缺乏数据支撑等问题,设计并实现基于大数据分析技术的校园运行分析系统,将传统的基于经验管理模式转化为基于大数据的数字化运行模式。系统构建出“Flume 采集→Kafka 缓冲→Spark 计算→ClickHouse 存储”的大数据处理链路实现多维数据进行采集、清洗、转换与存储,并基于 Vue 和 SpringBoot 框架开发可视化平台,面向学生、教师和后勤人员三类用户群体。该系统能够刻画出准确的学生画像,实现了从整体校园运行状态监控到学生个体行为分析的全面覆盖,有效的提升校园管理效率与信息化服务水平。

【关键词】智慧校园;数据驱动;学生画像;大数据分析;可视化平台

1.引言

在教育信息化 2.0 行动的推进下,高校数据体量快速增长,但“重建设、轻应用”的现象仍普遍存在,数据价值未能充分挖掘[1]。目前校园管理面临三个主要困境:一是业务系统独立建设互不连通而出现数据壁垒,学生各种行为数据无法整合,实现数据的统一分析比较困难;二是异常识别主要依赖人工巡查排查,学业预警、消费异常等问题大多数只能事后察觉;三是日常管理决策多依赖以往经验,缺少数据作为支撑,决策的科学性不足。现有这些问题是管理模式落后于技术发展的体现,当下亟需通过搭建系统性平台实现从“数字校园”向“智慧校园”的深层转型。

现有的研究大多数在单一的维度上取得进展,如依托消费数据的困难生识别、利用成绩预测的学业预警等,但多呈“烟囱式”架构,缺乏整体统筹,用户视角分散割裂,数据处理与业务场景脱节[2],难以落地。更关键的是,技术应用与管理变革并不同步,数据驱动校园管理大多还停留在理论层面,缺乏贴合实际运行场景的实践检验。

本文从校园整体运行视角出发,设计搭建覆盖数据采集、处理、分析、展示全链路的一体化系统,打通技术实现与管理应用的衔接壁垒,探索数据驱动管理模式的可行路径。

2.系统开发流程

(1) 需求分析与原型设计。基于目前教育数字化转型的背景,梳理学生、教师、后勤

三类用户的核心痛点。使用 Figma 完成三大子系统核心页面的高保真原型设计,组织评审确认页面功能完整性与逻辑交互合理性。

(2) 大数据处理链路搭建阶段。部署 Flume Agent,将数据推送至 Kafka 消息队列,按学生基础信息、成绩等创建不同的 Topic[3]。编写 Spark Structured Streaming 消费 Kafka Topic 中的数据,实现数据的预处理与基础聚合计算,结果写入 ClickHouse 列式数据库[4]。

(3) 前后端开发阶段。后端基于 SpringBoot 构建 RESTful API,集成 MyBatis-Plus 操作 ClickHouse,使用 JWT 实现三个角色独立 Token 及接口权限控制[5],按 Entity-Mapper-Service-Controller 分层开发用户权限、校园事务、活动资源、学业管理等核心模块。前端基于 Vue3+Vite 搭建,配合 Vue Router 管理路由[6],选用 Element Plus 组件库,通过 Axios 调用后端的 API,集成 ECharts 实现可视化组件[7],针对三类用户设计独立界面:学生端、教师端、后勤端。

(4) 测试与部署阶段。主要使用 Postman、JMeter、Selenium 等测试工具对项目进行功能、接口和性能测试,并实现漏洞管理和回归测试,以及使用 Docker 容器化技术将项目部署到虚拟机中。

3.大数据处理链路实现

3.1 数据来源

数据来源于静态数据文件和模拟数据集两部分。静态数据以 CSV 格式存储,涵盖学

生基础信息、成绩记录、泡馆信息等核心维度；模拟数据参考真实的业务场景，设置一定的规则，使用 Python 程序生成数据。两类数据相互配合，为学生数字画像的构建与平台功能验证提供数据支持。

3.2 数据采集、处理与存储

针对校园数据分散、格式不一的特点，设计"采集→缓冲→计算→存储"全链路方案。

(1) 数据采集：部署 Flume Agent，配置 TAILDIR Source 文件监控本地 CSV 数据目录，将采集到的数据推送至 Kafka 消息队列，按学生基础信息、成绩、借阅、出勤等创建不同 Topic 实现数据分类传输。部分数据文件采集配置映射如表 1 所示。

表 1.部分数据文件采集配置映射表

配置文件	监控文件	Kafka Topic
flume-students-base.conf	学生基础信息.csv	students_base
flume-students-score.conf	学生成绩信息.csv	students_score
flume-library.conf	学生消费信息.csv	library_records
students_attendance.conf	学生出勤信息	students_attendance

(2) 数据缓冲：采用 Kafka 作为分布式消息队列，用在采集层和计算层中间用作消息队列，把数据暂存起来，实现采集层与计算层的解耦和流量削峰填谷。

(3) 数据计算：采用 Spark Structured Streaming 消费 Kafka Topic 中的数据，完成缺失值过滤、异常值去除，格式校验、多源关联及画像标签计算，实现数据的预处理以及聚合计算。

(4) 数据存储：选用 ClickHouse 列式数据库，利用其向量化执行引擎满足聚合查询的高性能需求。部分核心数据表及描述如表 2 所示。

表 2.部分核心数据表及描述

表名	描述
students_base	学生基础信息表，存储学生的基本档案信息
students_attendance	学生出勤汇总表，汇总学生的出勤情况
library_records	图书馆记录表，图书馆出入及座位预约记录
classroom_schedule	教室课表，用于存储教室的排课和使用情况
Students_score	学生成绩记录表，记录学生入学以来的所有成绩

4.系统实现

4.1 角色权限管理

为进一步解决校园管理中职责不清、跨角色协作不协调、任务分配不清晰，数据访问边界不明确的问题，系统根据用户角色：学生、教师、后勤人员，设置不同的功能权限与数据访问范围，为三类用户定制个性化系统访问界面，每类用户只可登录到对应的系统，行使相应的权限功能，从而加强智慧校园运行分析系统的权限控制，系统部分功能权限分配如表 3 所示。

表 3.部分系统功能权限分配

功能模块	学生	教师	后勤人员
学生信息	查看个人	查看修改	—
学生成绩	查看个人	查看修改	—
预警发送	接收预警	发送预警	—
图书馆	借阅预约	查询所有	资源分配
校园活动	创建报名	查看所有	—
教学楼	—	预约教室	清扫维修
消费记录	查看个人	查看所有	统计数据
调查问卷	填写问卷	创建查看	—
请假事务	提交申请	审核审批	—
学生行为分析	—	查看画像与预警	—
AI 智能助手	—	使用分析功能	—
宿舍报修	提交报修	—	处理报修

4.2 用户功能

(1) 学生端功能

学生端支持学生查询个人账号信息、成绩信息，每日课表与消费数据，同时提供图书借阅预约、问卷填写、请假申请、宿舍报修等服务，学生可以通过书馆预约页面提供座位表格，如图 1 所示：实时展示座位的占用状态，支持学生选择座位并提交预约申请，有效提升图书馆资源利用率。

同时学生端也集成了几个常用的办事入口在仪表盘页面，图书借阅和座位预约放在仪表盘的同一个模块里，方便同学预约的同时进行图书借阅，提高学习的兴趣；在仪表盘还有一个消息列表，用于接收老师的预警消息，学生可以参考老师的提醒来调整自己的学习状态，从而有效提高了学生的学习效率；仪表盘还设置展示学生的每日课表，通过查询可以知道当日的课程安排，使学生不会因为记错课程而导致缺勤。



图 1.图书馆座位实时监控大屏

(2) 教师端功能
系统从宏观层面为教师端提供校园运行数据总览功能,如图 2 所示,校园大屏展示了

六个统计图表,有助于教师从总体直观地分析学生数据和校园资源。



图 2.校园数据总览大屏



图 3.学生行为分析页面

微观层面上整合所有学生成绩信息、出勤记录、图书馆预记录据等多维度信息构建每个学生的数字画像,并提供学生行为分析 AI 助手来辅助教师进行学习预警与生活帮扶,如图 3 所示,这个 AI 助手基于 Coze 平台搭建,该系统调用模型的接口来实现问答功能,该助手

会根据学生画像里的信息,以表格的形式整合出综合性的数据,简化了教师所需要的查询操作,并自动生成一段分析报告,比如该学生哪门课成绩明显下滑、出勤率最近是否变低、图书馆学习时长是否异常减少,还会附带一些具体的建议,极大地方便了教师对学生行为

分析。

同时系统提供在线服务,教师可以在线预约教室,完成请假审核与发布评教问卷,将线下事务改为线上处理,极大地提高校园事务的处理效率。

(3) 后勤端功能
系统整合报修处理、空教室清扫与维修、

图书馆资源分配,餐厅数据分析等后勤功能模块,后勤人员可以快速并且高效的完成工作,如图 4 所示,报修处理页面按状态分类展示待处理、处理中、已完成等记录,后勤人员可对每条报修申请执行状态变更操作,实现报修申请从提交到完结的全流程跟踪管理,极大地提高了校园事务的处理。



图 4.后勤报修处理页面

5.系统测试与部署

5.1 系统测试

(1) 功能测试:采用等价类划分、边界值分析与场景法设计测试用例[8],通过人工操作逐一核验各页面交互逻辑与实际业务流程运行状态,同时依托 Selenium 工具编写 Python 代码针对日常高频的操作场景开展 UI 自动化测试校验[9]。测试阶段排查出的各类系统问题与潜在的漏洞,均完成整改优化与多次回归复测,目前所有问题已经全部处理完毕并闭环。

(2) 接口测试:使用 Postman 工具对系统 RESTful 接口开展全覆盖测试,分别针对常规业务调用、非法参数提交、访问权限控制以及各类边界场景开展逐项验证,实测结果表明系统接口返回格式符合设计标准,能够有效拦截各类异常访问行为,整体数据交互运行稳定,在运行可靠性与数据安全层面均达到预期使用标准。

(3) 性能测试:通过 JMeter 工具模拟多用户同时在线操作,依次开展负载测试、并发测试、压力测试以及长时间稳定性测试,系统在高并发极限场景下保持零错误率,吞吐量和响应时间数值曲线平稳且满足日常需求,证明其具备良好的负载能力。

5.2 系统部署

回归测试完成后,将项目打包上传到 Linux 虚拟机环境使用 Docker 完成容器化部署:拉取 Nginx 镜像,将前端 dist 静态目录挂

载至 Nginx 站点目录,配置端口映射与重启策略,并在 Nginx 中配置反向代理将/api 请求转发至后端服务;后端以独立 jar 包方式运行;ClickHouse 通过数据卷挂载实现数据持久化。通过 Docker Compose 实现容器化编排并一键启动,形成前后端分离、数据层持久化的稳定部署架构[10]。

6.结论

本文设计并开发了智慧校园数据分析系统,实现了从多源数据采集、处理到分析展示的完整流程。首先整合学生消费、考勤、借阅、活动等异构数据,构建学生画像。其次搭建平台,赋予教师、学生、后勤用户不同的权限,将校园事务流程串联起来。经过一系列测试,系统功能完整、接口稳定、具备高并发承载能力。最后在虚拟机中成功部署,系统的可行性得到初步验证。

智慧校园数据分析系统的价值在于有效解决了校园数据孤岛问题,将异常识别从人工经验判断转向数据驱动预警,为学生信息管理、教学资源优化、后勤调度与管理决策提供了科学的数据支撑,推动校园管理从经验化到数字化的转型。

参考文献

[1] 魏比贤.数字化赋能视域下高校智慧校园一体化管理系统设计[J].软件, 2025, 46(05): 139-141.
[2] 张辉,李健明,杨强.大数据视角下高校数

- 据治理体系研究与实践[J].中国高等教育, 2022(Z2): 16-18.
- [3] 金双喜, 李永, 吴骅, 等.基于 Kafka 消息队列的新一代分布式电量采集方法研究[J].智慧电力, 2018, 46(02): 77-82.
- [4] 蒋雷, 白伟丽, 李小红.基于 ClickHouse 的实时数据仓库的基础架构研究[J].现代计算机, 2024, 30(11): 91-95.
- [5] 虎.一种基于 JWT 认证 token 刷新机制研究[J].软件工程, 2019, 22(12): 18-20.
- [6] 谢振华.基于 Vue.js 与 Spring Boot 的教务管理系统设计[J].电脑与信息技术, 2024, 32(04): 95-97+101.
- [7] 徐欣威.基于 ECharts 的科技统计数据可视化设计与实现[J].天津科技, 2019, 46(03): 66-70.
- [8] 王峰, 李江, 张永涛.测试设计分析方法之场景分析法优化研究[J].信息技术与信息化, 2023, (11): 58-61.
- [9] 姜文, 刘立康.基于 Selenium 的 Web 软件自动化测试[J].计算机技术与发展, 2018, 28(09): 47-52+58.
- [10] 窦浩.面向容器化部署的高可用网站开发技术研究[J].软件, 2025, 46(08): 93-95.