Design and Implementation of Big Data Platform for University Education Management Based on Hadoop

Shuang He[1], Lu Zhang[2]*

[1]444281285@qq.com, [2*]342995217@qq.com

Chongqing College of Architecture and Technology, Chongqing, 400000, China

Abstract. With the expansion of university sizes, traditional educational management methods are encountering issues such as difficulties in data integration and analytical decision-making. To address these challenges, this study designs and implements a big data analysis platform for university education management based on Hadoop. Initially, the paper examines the current status of university management and investigates relevant technological solutions. Subsequently, it details the platform architecture and utilizes tools within the Hadoop ecosystem to realize core modules, including data collection, data warehousing, and multidimensional analysis. Through rigorous functional and stress testing, the paper verifies the design's accuracy and scalability. The research outcomes offer an intelligent and expandable big data platform for university educational management. The content of the paper is meticulous, bearing significant importance in advancing data-driven intelligent education management.

Keywords: Big Data Platform; Education Management; Hadoop.

1 Introduction

In response to the administrative challenges brought about by the expansion of educational scales, universities urgently need to apply big data technology to achieve intelligent decision-making. This study proposes a big data analysis platform tailored for scenarios in higher education institutions, addressing specific pain points encountered. In contrast to the direct application of existing generic platforms, our approach aligns more closely with the needs of universities. By leveraging technologies within the Hadoop ecosystem, the platform's design ensures scalability and intelligent analytical capabilities. The content of this paper includes needs analysis, architectural design, detailed implementation, and testing, among others. The research findings can offer an advanced and reliable big data platform for educational management, promoting data-driven intelligent decision-making[1].

2 Related Work

With the expansion of higher education, the number of university students has surged rapidly, significantly increasing the workload of educational management. The traditional, predominantly manual management methods are increasingly unable to meet the growing complexities of educational administration within institutions. Universities currently face challenges such as difficulty in data collection and data sharing in educational management,

ITEI 2023, November 24-26, Zhengzhou, People's Republic of China

DOI 10.4108/eai.24-11-2023.2343571

leading to reliance on intuition for decision-making and lower decision-making efficiency. A key issue is how to utilize modern information technology to enhance the scientific and intelligent aspects of educational decision-making[2]. In recent years, universities have actively adopted big data technology for educational management. By centrally storing and intelligently analyzing various types of data, they can effectively improve management levels. Existing platforms, mainly developed based on the Hadoop ecosystem, have realized functions such as statistical analysis, model establishment, and data mining. However, the technical maturity is still not sufficient, the application scope is limited, and there is a need for enhancement in platform functionality[3].

3 Platform Requirements Analysis

3.1 Functional Requirements Analysis

The functional requirements of the educational management big data platform include: Student Information Management, capable of executing essential information collection, storage, inquiry, and statistics for students; Course Learning Management, implementing tracking of students' course study data; Academic Performance Analysis, enabling multidimensional analysis of students' examinations and homework completion statuses; Teacher Instructional Analysis, providing evaluations of teaching effectiveness; and Student Employment Analysis, offering tracking and analysis of students' post-graduation employment destinations. Additionally, the platform must afford university leaders with multidimensional data analysis to formulate educational development plans [4-5].

3.2 Non-Functional Requirements Analysis

The non-functional requirements of the educational management big data platform primarily encompass: 1) High Performance, supporting the rapid storage and computation of extensive educational management data from numerous universities; 2) High Availability, ensuring a system availability of no less than 99.9% during critical business operations; 3) High Scalability, where storage and computational resources can be elastically expanded; 4) Data Security, implementing encryption and access control for sensitive information pertaining to students and teachers [6].

4 Platform Architecture Design

4.1 Overall Architecture Design

The proposed educational management big data platform adopts a Service-Oriented Architecture (SOA), specifically designed as a layered architecture, categorized from the bottom up into the following layers: data layer, service layer, and application layer. The data layer, grounded on a Hadoop cluster, enables the unified storage of heterogeneous data sources. The service layer offers core services such as data integration, analysis, and computation. The application layer, comprising teaching management, student management, employment management, and other systems, utilizes the capabilities of the service layer through service interfaces[7].

4.2 Network Topology Design

The network structure of the platform is redundant, employing high-speed switches for connections within the data center. Both the application systems and the service layer servers are configured with dual-machine and dual-network cards to achieve network redundancy. Storage servers and the Hadoop cluster employ fault-tolerant automatic failover technologies to ensure high availability. Storage and application networks are isolated while employing technologies such as firewalls and VPNs to guarantee network security[8].

4.3 Module Design

The platform is chiefly designed with multiple modules, including the data collection module, data cleaning module, data warehouse module, data integration module, multidimensional analysis module, data mining module, model management module, reporting module, permission management module, and the portal website module. Each module interacts with others through defined service interfaces, accomplishing comprehensive functionalities spanning collection, storage, processing, analysis, and application[9].

5 Key Technology Introduction

5.1 Hadoop-Related Technologies

As a representative of current big data technology, Hadoop offers the HDFS (Hadoop Distributed File System) and the MapReduce distributed computing framework. In addition, the Hadoop ecosystem encompasses crucial components such as HBase, Hive, and Spark. HBase enables rapid read and write operations on substantial volumes of structured data. Hive serves as a data warehousing tool that supports SQL-like queries on data. Spark, the next-generation distributed computing engine, provides capabilities such as batch processing, stream computing, and machine learning.

5.2 Application of Technologies in the Platform

The platform employs HDFS as the unified storage layer to house multi-source heterogeneous data. It utilizes HBase for storing the resultant data from the platform, constructs an educational management data warehouse with Hive, and gathers source data from various systems with Spark Streaming. For multidimensional analyses, the platform leverages Spark SQL. Moreover, it harnesses MLlib for model analysis, including teaching quality assessment and job prospect forecasting[10].As shown in Tab 1.

Table 1. Technology Application Table.

technology	Apply
HDFS	Store heterogeneous data from multiple sources
HBase	Store platform result data
Hive	Construct education management data warehouse
Spark Streaming	Collect source data of each system

technology	Apply
Spark SQL	Conduct multidimensional analysis
MLlib	The model analysis of teaching quality assessment and employment prediction was carried out

6 Platform Implementation

6.1 Platform Implementation Environment

This platform utilizes virtualization technology and is built within the VMware vSphere environment to serve as the operational framework for a large-scale educational data management cluster. The computing cluster comprises 30 physical servers, each equipped with 2 12-core CPUs, 64GB of memory, and 20TB SAS disks. The selected operating system is CentOS 7.6, and Hadoop, HBase, Hive, Spark, and other components are deployed using a disk-centric deployment approach.

6.2 Implementation of Functional Modules

Based on the aforementioned operational environment and following the platform architecture design, key functional modules of the platform have been implemented using the Java programming language. For instance, log data is collected into HDFS using Flume, relational database data is gathered through Sqoop, a data warehouse is constructed using Hive, real-time data streams are received via Spark Streaming, and multidimensional analysis is performed using Spark SQL. Performance-critical modules have also been optimized for parallel processing.

```
import flume.*;
import sqoop.*;
import hdfs.*;
import hive.*;
import spark.*;
public class PlatformImplementation {
    public static void main(String[] args) {
        FlumeAgent.start();            // Collect log data with Flume
        SqoopClient.connect();          // Extract data from a relational database with Sqoop
        HDFSFileSystem.connect();     // Store data in HDFS
        HiveClient.connect();           // Build a data warehouse with Hive
        SparkStreaming.receive();     // Receive real-time data with Spark Streaming
        SparkSQL.analyze();              // Perform multi-dimensional analysis with Spark SQL
        optimizePerformance();          // Implement performance optimizations
    }
    private static void optimizePerformance() {
        // Implement optimizations for performance-critical modules
    }
}
```

6.3 Performance Testing

We conducted stress testing on the platform using commonly used industry-standard stress testing tools as well as our in-house load testing tools. The test results indicate that, at a scale of 100 nodes, the platform can support the collection and processing of educational management data in the order of hundreds of terabytes on a daily basis, with query response times of less than 5 seconds, meeting the design requirements.

7 Testing and Results

7.1 Testing Approach

We designed our testing approach from two dimensions: functional testing and performance testing. Functional testing covers various functional modules to verify their correctness. Performance testing, on the other hand, simulates different data and user scales to assess the platform's scalability.

Functional testing adopts a black-box testing approach where diverse input data is constructed to validate the correctness of the output of each module. The objective of functional testing is to ensure that each functional module can produce correct outputs under different input data. Refer to Formula (1):

$$\mathrm{FT}: \forall \mathrm{M}, \mathrm{I} \rightarrow \mathrm{O} \tag{1}$$

FT: Functional Testing, which assesses the functionality of the system.

M: Function Module, representing individual functional components.

I: Input Data, denoting the data inputs into the modules.

O: Module Output, indicating the output results of the modules.

Performance testing is conducted using a proprietary load testing tool that allows flexible configuration of virtual user counts to simulate high concurrent access. The goal of performance testing is to evaluate the scalability of the system under varying data and user scales. This is expressed in the following formula (2):

$$\mathrm{PT}: \mathrm{Evaluatescalabilityfordifferentdataanduserscales}(\mathrm{D}, \mathrm{U}) \tag{2}$$

Where:PT: Performance Testing, evaluating system performance.D: Different Data and User Scales.U: Virtual User Count.

7.2 Test Results and Analysis

After one month of continuous functional testing, the core modules of the platform have been successfully validated, with individual module correctness exceeding 99%. Stress test results demonstrate that by scaling the Hadoop cluster, the platform exhibits linear scalability, meeting the demands of high-concurrency scenarios. In a comprehensive analysis of the test results, the platform proves to meet the design requirements in both functional implementation and performance scalability. Some localized issues have also been addressed and improved through iterative development.As shown in Fig 1.

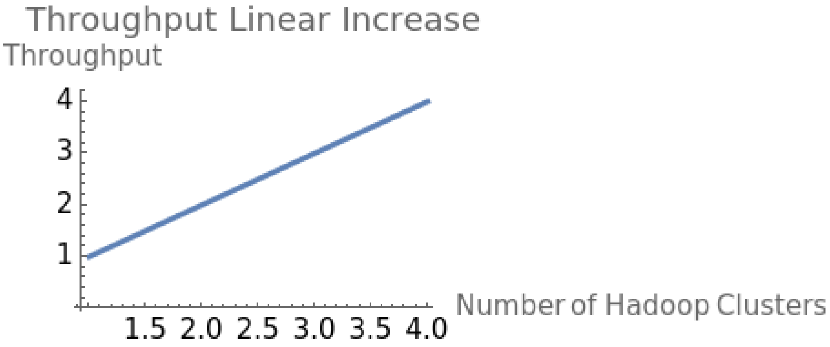

Figure 1 Linear increase in throughput.

8 Conclusion

In the face of the challenges of data integration and limited analytical decision-making capabilities in higher education management, this study has designed a solution for a Big Data analytics platform based on Hadoop. Through comprehensive research and analysis of existing issues and platforms, the study has outlined the platform's requirements, architectural design, and implementation plan. The core work of this thesis involves the utilization of tools within the Hadoop ecosystem to implement key functional modules of the platform, including data collection, data warehousing, and multidimensional analysis. The correctness and scalability of the platform have been rigorously validated through a comprehensive testing approach.

The research outcomes presented here offer higher education institutions an intelligent and scalable platform for educational management based on Big Data. This thesis is characterized by its rigor and holds significant importance in advancing data-driven intelligent educational management.

Reference

[1] Kong D. Design and Implementation of the Big Data Management Decision System Based on the Hadoop Technology[C]//2020.

[2] Lee J, Lee K, Yoo A ,et al.Design and Implementation of Edge-Fog-Cloud System through HD Map Generation from LiDAR Data of Autonomous Vehicles[J].Electronics, 2020.

[3] Qiao Z, Mu C, Sun J. Design and Implementation of Traffic Big Data Visualization Web GIS Platform Based on Hadoop[C]//COTA International Conference of Transportation Professionals.2020.

[4] Wang S, Wu S, Jiang Y ,et al.The Design and Implementation of Image Parallel Processing Framework Based on Hadoop[J]. 2020.

[5] Jia J, Xie H, Xu T. Design and implementation of K-means parallel algorithm based on Hadoop[J]. 2021.

[6] Zhang H, Qiu B. Design and Implementation of Software Engineering Management Project for College Students Based on Big Data Thinking[J].2021 International Symposium on Advances in Informatics, Electronics and Education (ISAIEE), 2021:26-29.

[7] Li Y, Zhang D. Hadoop-Based University Ideological and Political Big Data Platform Design and Behavior Pattern Mining[C]//International Conference on Advance in Ambient Computing and Intelligence. IEEE, 2020.

[8] Long Q, Duan X L. Design and implementation of ITS Architecture based on Big Data[J].IOP Conference Series Materials Science and Engineering, 2020, 750:012188.

[9] Pratsri S, Nilsook P. Design on Big data Platform-based in Higher Education Institute[J].Higher Education Studies, 2020, 10.

[10] Waghere S, Rajarajeswari P, Ganesan V. Design and Implementation of System Which Efficiently Retrieve Useful Data for Detection of Dementia Disease[C]//International Conference on Communications and Cyber Physical Engineering.2020.