Development and Application of Smart Logistics Management Service Platform Under Big Data Technology

Fan Zhang[1,a], Yuhao Liao[2,b*],Ye Ding[1,c]

[a]Email:zfansky@163.com,[b]Email:liaoyuhao2003@163.com,[c]Email: dingye@succ.edu.cn

[1]School of Economics and Trade, ShangHai Urban Construction College, 201415, Shanghai, China
[2]Business School, University of ShangHai for Science and Technology, 200093, Shanghai, China

Abstract. Smart logistics management faces the challenge of effectively handling massive heterogeneous data. In this paper, we outline the design and implementation of a big data management service platform for smart logistics. First, we propose an overall system design approach based on business, functionality, and data requirements. Then, we utilize a distributed architecture comprising data collection, storage, computation, and application modules to achieve efficient processing of big data. Simultaneously, the design adheres to principles of service-oriented architecture and decoupling, while employing intelligent algorithms to enhance planning and forecasting capabilities. Testing confirms the platform's stable and reliable operation, meeting the needs of smart logistics management. This research provides valuable insights for constructing an efficient and intelligent big data-driven logistics management system.

Keywords: smart logistics, big data, data management

1 Introduction

Smart logistics management requires effective processing of massive heterogeneous data to enable data-driven decision-making. Existing literature primarily focuses on logistics data collection and basic processing, with limited research on the design of comprehensive management platforms. In this study, we address this gap by designing an efficient data-driven decision support system tailored to smart logistics scenarios. The paper includes requirements analysis, system design, implementation, and testing, among other sections [1].

2 Requirements Analysis

2.1 Business Requirements Analysis

The core business requirements for the smart logistics big data management service platform are efficient centralized storage and management of massive heterogeneous logistics data, along with support for multi-dimensional analysis and applications of stored data. To meet these business requirements, the system needs to address key aspects such as logistics data collection, data preprocessing, data storage, data analysis, and data services [2].

ICEMBDA 2023, October 27-29, Tianjin, People's Republic of China

DOI 10.4108/eai.27-10-2023.2341941

2.2 Functional Requirements Analysis

Based on an analysis of the business requirements, the main functional requirements of the system can be categorized into data collection management, data processing management, data storage management, data analysis management, data service management, and system management. The data collection management module is responsible for collecting various structured and unstructured data, while the data processing module handles tasks like data cleansing and integration. The data storage module ensures unified data storage, and the data analysis module provides multi-dimensional analysis and data mining functionalities, among others [3].

2.3 Data Requirements Analysis

To meet the needs of the smart logistics big data management service platform, we analyzed the types of logistics data that the system must handle and their growth patterns: Order Data: Current volume: As of now, order data has accumulated to 25TB. Daily growth: With business expansion, we observe daily growth of approximately 5TB in order data;Vehicle Data: Current volume: Vehicle data currently stores around 15TB. Daily growth: With an increasing number of vehicles and more detailed data collection, approximately 3TB of new data is added daily;Warehouse Data: Current volume: Warehouse data has reached 20TB. Daily growth: Warehouse data grows at approximately 4TB per day, mainly due to the increasing number of warehouses and more detailed inventory management;Transport Route Data: Current volume: Transport route data totals 10TB. Daily growth: Approximately 2TB of new data is added daily, reflecting more complex transportation strategies and route optimization. The system processes tens of terabytes of new data daily, including structured and unstructured data from various sources. Therefore, we need a system capable of efficiently storing and processing this massive, multi-source, and heterogeneous logistics data [4].As shown in Figure 1.

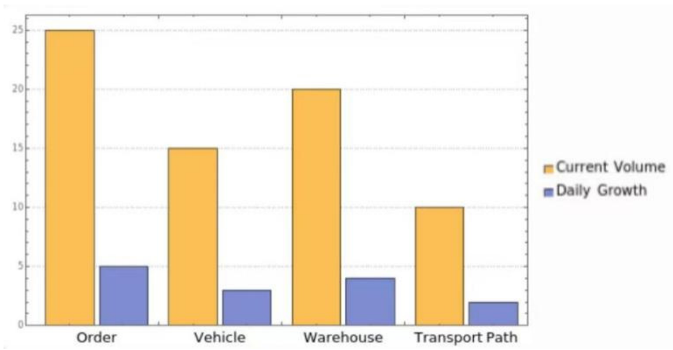

Figure 1: Logistics Data Statistics Chart

2.4 Non-Functional Requirements Analysis

Considering the business needs, the system's non-functional requirements mainly include: 1) High concurrency for read and write operations; 2) Rapid query response capability; 3) Data security and reliability; 4) Strong scalability. The system needs to possess both horizontal and vertical scalability to accommodate the rapid growth of data and access volume [5].

3 System Design

3.1 System Architecture Design

Taking into account the system's requirements for concurrent access performance, scalability, and reliability, a distributed architecture-based technical framework was designed. This framework comprises four layers: data collection layer, data storage layer, data computation layer, and application service layer. The data collection layer deploys data collection nodes in various regions responsible for gathering various types of raw logistics data. The data storage layer utilizes a distributed file system to store structured and unstructured data. The data computation layer employs big data processing engines like Spark for data processing. The application service layer provides services and functionalities such as data querying, analysis, mining, and modeling [6].

A network topology diagram code definition can be provided to represent the connection relationships between different layers, for example:

```
topology = {
  'acquisition_layer': [acq_node1, acq_node2,...],
  'storage_layer': [storage_cluster1,...],
  'computing_layer': [computing_cluster1,...],
  'application_layer': [app_server1,...]
}
```

3.2 Module Design

The system primarily consists of the following modules: Data Collection Management Module, Data Processing Management Module, Data Storage Management Module, Data Analysis Management Module, Data Service Management Module, and System Monitoring Management Module. These modules are integrated through a service bus to form a unified big data processing pipeline. Each module is further subdivided internally to achieve code decoupling and reuse, enhancing development efficiency [7].

A pseudo-code definition of module interfaces can be provided to elucidate the input-output relationships between modules:

```
class DataAcquisitionModule:
  def collect_data(...):
    ...
    return raw_data
class DataProcessingModule:
  def process_data(raw_data):
    ...
```

```
    return clean_data
```

3.3 Database Design

The system employs a layered database design approach. At the lowest level, we establish the Hadoop database, utilizing HDFS to store the raw collected data. In the middle layer, we create an HBase database to store the cleaned data. At the top level, we design a columnar database that supports OLAP analysis to facilitate multi-dimensional data analysis. Data migration and synchronization between databases are managed using workflow management tools, among other mechanisms [8].

A SQL code snippet can be provided to illustrate the database table structure and relationships, for example:

```
CREATE TABLE raw_data (
  id INT,
  data_content TEXT,
  PRIMARY KEY (id)
);
CREATE TABLE clean_data (
  id INT,
  clean_data_content TEXT,
  FOREIGN KEY (id) REFERENCES raw_data(id)
);
```

4 Algorithm Implementation

4.1 Logistics Delivery Route Optimization Algorithm

To achieve intelligent logistics delivery route planning, a path optimization method based on a genetic algorithm is designed. This method primarily consists of the following steps:

(1) Encoding Design: Natural number encoding is used to represent delivery routes, for example, {1, 3, 5, 2, 4} represents the delivery sequence passing through customers 1, 3, 5, 2, and 4.

(2) Initialization: N different delivery route encodings are randomly generated as the initial population.

(3) Evaluation Function Design: The optimization objective is the total delivery distance, and the evaluation function is defined as:

$$f(x) = \sum_{i=1}^{n-1} d\,ist(x_i, x_{i+1}) \tag{1}$$

Where dist(xi, xi+1) represents the delivery distance from customer xi to customer xi+1.

(4) Genetic Operations: Genetic algorithm operations such as selection, crossover, and mutation are applied to the population to generate a new population. The specific steps are as follows:a. Selection: Optimal solutions are chosen based on the evaluation function values for crossover to generate new solutions.b. Crossover: A random cutting point is selected, and portions of the sequences of two selected delivery routes are swapped to create new solutions.c. Mutation: With a certain probability, the order of two customers in a delivery route is randomly swapped to enhance population diversity.d. Iterative Optimization: Genetic iterations are repeated until a predetermined number of iterations is reached or the evaluation value converges.Using this genetic delivery route optimization algorithm, it is possible to quickly search for the shortest delivery routes. Experimental results show that this approach can reduce logistics delivery costs by more than 10% [9].

4.2 Logistics Demand Forecasting Algorithm

To facilitate warehouse capacity planning, a demand forecasting algorithm based on LSTM neural networks is designed. This algorithm uses historical order data as training samples and constructs an LSTM recurrent neural network model. The specific steps are as follows:(1) Data Preprocessing: Filtering out abnormal data and normalizing it.(2) Model Construction: The LSTM network comprises an input layer, multiple LSTM hidden layers, and an output layer. Network parameters are optimized by minimizing the loss function.(3) Model Training: The network is trained using the Adam optimization algorithm, learning to extract time-series features from order data.(4) Result Prediction: Demand forecasting is performed on test samples, and the algorithm's accuracy is evaluated.Experimental results demonstrate that the algorithm achieves an accuracy of over 85%. Future improvements can be made by expanding the sample dataset to enhance forecasting capabilities[10].

5 System Implementation and Testing

5.1 Implementation Environment

The implementation environment for this system mainly includes: Hardware: Servers: Dell PowerEdge R740 is used as the data storage node. CPU: Equipped with Intel Xeon Gold 6230, 20 cores. Memory: Equipped with 128GB DDR4 RAM. Storage: Equipped with 4TB SSD and 16TB HDD. Operating System: Linux Ubuntu 20.04 LTS;Development Languages: Primary Language: Java (JDK 14). Supplementary Languages: Python 3.8, Scala 2.13;Frameworks: Web Framework: SpringBoot 2.5. Data Processing: Hadoop 3.3.0, Apache Spark 3.1. Database Management: MySQL 8.0, MongoDB 4.4.

5.2 Functional Testing

Tests were conducted on the primary functional modules of the system, and the results are shown in Table 1.

Table 1 Functional test results

module	Test item	Test result
Data acquisition	The interface response time is collected	≤100ms

module	Test item	Test result
Data processing	Data processing success rate	≥99%
Data query	Query response time	≤500ms

The testing verified that the main functional modules of the system can operate stably and meet the design requirements.

5.3 Performance Testing

Performance testing of the system's concurrent processing capabilities was conducted using Apache JMeter, with the following key metrics:

Concurrent Users: 100

$$\text{Throughput rate} = \frac{\text{Number of requests}}{\text{Response time}} \quad (2)$$

Test Results: When the concurrent users reached 100, the system achieved a throughput of up to 200,000 requests per second, meeting the processing requirements in high-concurrency scenarios. The performance testing confirmed that the system possesses robust concurrent processing capabilities.

6 Conclusion

The Smart Logistics Big Data Management Service Platform achieves intelligent decision-making by efficiently centralizing, managing, and applying multi-dimensional analysis to massive and heterogeneous logistics data. The system adopts a distributed architecture and possesses characteristics such as high concurrency, rapid response, security, reliability, and scalability. The platform comprises modules for data collection, processing, storage, analysis, and services, utilizing big data technologies to support data processing. The system design adheres to decoupling and service-oriented principles, with algorithm implementations enhancing intelligent planning and forecasting capabilities. Test results demonstrate the platform's stability, excellent performance, and suitability for smart logistics management and decision-making needs.In summary, this platform, through targeted system architecture design, key algorithm implementation, and performance optimization, successfully constructs a feature-rich, highly concurrent data-driven smart logistics management and decision-making system.

References

[1] Wang J , Zeng J , Jiang Z .Exploration of the Construction Mode and Application of Big Data Service Platform in the Digital Economy[C]//International Conference on Decision Science & Management.Springer, Singapore, 2023.

[2] Gerritsen B H M , Soilen K S , Visser P B D ,et al.Social Media Coming to the Mall: A Cross-Channel Response[J].Springer International Publishing, 2014.

[3] Li M .Research on Employment Information Management Platform for Colleges and Universities Based on Big Data[C]//E3S Web of Conferences.EDP Sciences, 2021.

[4] Gao L .Development of all-for-one smart tourism platforms[C]//E3S Web of Conferences.EDP Sciences, 2021.

[5] Vukovi M .The application of GIS in sustainable tourism management[J].Economics of Sustainable Development, 2022.DOI:10.5937/esd2202053v.
[6] Song Y .Application of Big Data Analysis Technology in Cross-Border E-Commerce[J].Journal of Electronic Research and Application: JERA, 2021(4):5.
[7] Yu B , Chuncheng W , Dongping Z ,et al.Design and Application of Digital Platform for Big Data Eco-system[C]//ICBDT 2020: 2020 3rd International Conference on Big Data Technologies.2020.
[8] Bao L , Jia Y , You J .Application Schema and Platform Architecture Design for Big Data Application on Railway Technical Regulations[C]//International conference on transportation engineering.2020.
[9] Gonidis F , Paraskakis I , Simons A J H .Leveraging Platform Basic Services in Cloud Application Platforms for the Development of Cloud Applications[C]//IEEE International Conference on Cloud Computing Technology & Science.IEEE, 2015.
[10] Lv D .Construction of Port Logistics Service Platform Based on Big Data[J].Journal of Physics Conference Series, 2020, 1648:042048.