Rule-based Data Quality Intelligent Monitoring System

Huayun Zhang, Shaolei Wang and Xiang Wang
1 Nari Group Corporation/State Grid Electric Power Research Institute, Nanjing 211106, China
2 China Realtime Database Co., Ltd., Nanjing 210012, China.

Abstract. The rapid development of Internet information technology brought great convenience to people's lives and work, but it also brought the problem of information overload[1]. The data center is the middle layer that mediates the contradiction between the foreground and the background[2]. It provides a reusable, standardized, and agile multifunctional platform through the modeling of background data and the aggregation of data services to support the foreground for the demand[3] of quickly change according to market changes. Through the data center, we have opened up the data islands of each business system. However, when each business system uses data, it found that the data quality is poor, unstable, not timely, and inconsistent. Traditional cluster data quality monitoring has the characteristics of inconsistent data quality requirements for various services, difficult development, low processing efficiency, difficult maintenance, and insufficient intelligence.

Based on this situation, we design a data quality monitoring system based on rules and artificial intelligence. General users perform simple configuration on the interface to complete real-time online monitoring of data, and real-time statistics on the number of data quality problems and a detailed list for users to trace the cause of the problem and identify abnormal scenarios in advance.

The comparison between experiments and mainstream products proves that the system has greatly improved in terms of user use and flexibility.

Keywords: Data quality; rule configuration; intelligent monitoring; machine learning; text emotion recognition.

1. Overview

The State Grid Corporation of China has accumulated abundant data resources in business processes such as infrastructure, personnel team, marketing, operation and inspection. At present, basic equipment is connected to 540 million smart meter terminals, and more than 280,000 connected charging piles are connected to the Internet of Vehicles. According to incomplete statistics, the company has more than 1.86 million employees, more than 1.1 billion people with power supply, and 225 million registered users of e-commerce platforms. 336 municipal power supply service command centers have been built[4]. In order to enable the effectively support of daily operations and decision-making of the organization, the data is required to be reliable and error-free, and to accurately reflect the real world conditions [5].

Solving the quality of such a large data set has become a major problem. Data quality management runs through all stages of the data life cycle like product quality management [6]. Poor data quality will cause downstream business systems to fail to complete reasonable calculations and even cause system downtime. In solving this problem, each business system splits the business and monitors the problems of poor data quality of each type, resulting in high development costs and low computational efficiency.

Due to the big data technology, the problem of low computational efficiency has been solved, but the high develop cost and the difficulty of maintaining functions have also become a problem.
The data quality monitoring system based on rules and artificial intelligence implemented by us solves all the above problems. We achieve the data monitoring based on big data technology and interface configuration. We provide a detailed result list and support the ability to retain the secondary development of the platform.

2. System Architecture

The Ali DataHub implemented in this article is a streaming data processing platform. The source system data is entered into the corresponding datahub through the ETL tool. Users configure the fields of the datahub in the system and filters the data through real-time calculation tool (Blink). Then they write the data into MySQL version of Ali analytical database in real time, and finally realize the visual analysis of the data, so as to regularly collect various indicators in the system and programs. For the database, we use MySQL version of Ali analytical database to achieve the data storage layer requirements. In addition, the data display layer is closest to the user, which customizes echarts components to realize the display of data in a centralized and unified way. The overall system architecture is shown in Figure 1 below.

An intelligent monitoring system for data quality based on regular data in the platform, including various functions such as interface configuration, data storage, data processing, and data display. In the functions of interface configuration, the MySQL version of the Ali analytical database is introduced, and the data entered by the user is stored in the database. The MySQL database is a PB-level real-time data warehouse with high concurrency and low latency. It can perform real-time multi-dimensional analysis of trillion-level data in milliseconds, thereby improving data access and writing efficiency. It is introduced in the data processing stage. The Ali DataHub streaming data processing platform realizes the matching of data filtering rules and verification rules, and enters the data that does not match the verification rules into the MySQL version of the Ali analysis database, which can efficiently process large batches of data. In the entire link, there is no data import and export. The written data is stored in interactive analysis, and there is no redundant storage. In the system interface, the components and templates of echarts are used in a customized way, and the data that does not pass the verification rules are managed neatly and uniformly.

3. System Design and Implementation

3.1 System Design

3.1.1 Overall design. Source system data is entered into our datahub through ETL tools. Users configure the fields of the datahub in the system through rules, and the Blink cluster performs data reading, rule mapping, serialization, compression, deserialization, decompression, operator calculation, real-time
It supports the establishment of multiple verification tasks for the same datahub-topic, and only one piece of data is used in the background for multiple verification, reducing the impact of data flow on disk capacity and I/O. It supports multiple types of common function calculation operators, such as max, min, avg, length, toDate, concat, etc. It supports multiple types of comparison operators, such as >, <, =, in, like, etc. It supports multiple types of machine learning operators, such as text recognition based on sentiment analysis, predictive random forest algorithm. It supports the result data writes into different adbs to ensure the efficiency of data fall in the big data scenario and reduce the impact of back pressure on the blink cluster.

3.1.2 Rule design. The rule configuration of this system includes filtering rules and verification rules. Two sets of rules adopt the same design of model, which can realize the quality verification of the rules under specific conditions. The basis of system rule configuration is calculation operator, comparison operator and rule operator. Through the custom combination of multiple operators, users can configure more complex rules independently, and realize a zero-threshold, high-throughput data quality verification tool through the powerful stream computing processing capabilities of Blink.

In addition, the system includes long and short-term memory network unit (LSTM), which is another module in RNN. From an abstract perspective, LSTM saves long-term dependent information in the text. As we have seen before, the traditional RNN network is very simple. This simple structure cannot effectively link historical information together. For example, in the field of question and answer, we can see that the middle sentence has no effect on the question being asked. However, there is a strong connection between the first sentence and the third sentence. For a typical RNN network, the amount of information stored in the hidden state vector for the second sentence may be much larger than that of the first sentence. But LSTM basically judges which information is useful, and saves the useful information in LSTM.

3.1.3 Data model design. The data model of the system implementation includes seven tables. They are task table, verification rule table, filter rule table, calculation operator table, comparison operator table, statistics, and finally the information is stored in adb. The effect of online real-time monitoring of data quality required by users is presented. The system business process is shown in Figure 2 below.
result configuration table, and real-time result statistics table. The task table stores the task name, topic related information at the source, and output result table related information. The verification rule table and the filtering rule table store detailed rule information, and have a many-to-one relationship with the task table. The calculation operator table and the comparison operator table are rule descriptions of the two types of operators, and have a one-to-one relationship with the verification rule table and the filter rule table. The result configuration table stores detailed configuration information of the task output table. The real-time result statistics table stores the detailed input quantity and the number of questions for each task at a minute granularity.

3.2 System Implementation

The system implementation is divided into data access module, data verification module, distribution and storage module, data display module.

The data access module refers to converting the source-end topic data into a fixed format and extracting it to real-time computing topic according to the information configured by the user. We will maintain a piece of memory to store task verification information for the corresponding topic. When the memory information of a topic is added, we will start the thread corresponding to the topic. When the task information is modified, the memory will be updated directly, so as to achieve seamless conversion of the verification information changes. When all verification tasks under the corresponding topic are stopping, the extraction thread of the corresponding topic will be closed. Through the combination of task rules, each topic only needs to start the extraction thread corresponding to the number of shards to achieve the effect “one extraction, and multiple verification” of multiple tasks.

The data verification module refers to the process of performing multi-task simultaneous verification of data through the Blink calculation engine and entering the corresponding result table at the same time. The access to the source data is realized through the function of Blink Sql, and the verification results are output to the result table and real-time statistics table at the same time. The core verification process is realized through Blink's custom operator function.

Distributing and storage module refers to the process of separately writing data verification results into the corresponding database according to user configuration information. The specific implementation process is as follows: First, start the Topic data reading thread corresponding to the number of shards, and write the data into the queue. Second, start the data distribution thread, and distribute the data to the queue where the corresponding ADB table data is located. Third, start the threads corresponding to the number of ADB tables to take data from the corresponding queues and write them into the result table specified by the user. This ensures that in extreme cases, if a certain result table is blocked in storage, it will not affect the storage operation of other result tables.

The data display module supports two forms of real-time monitoring and query by time period. If you check the real-time button, the page will refresh the latest monitoring result trend graph and latest problem data details every minute. If you don't check it, and query by time period, the monitoring results of the corresponding time period will be queried.

4. System Experiment and Evaluation

4.1 Experimental Environment

The configured rules are shown in the figure below:
Figure 3. Rule configuration information.

The calculation server configuration is shown in Table 1.

Table 1. Computing server configuration information.

| CPU            | Mem | OS   |
|----------------|-----|------|
| Intel(R)Xeon(R)CPUE5 2630v4@ 2.20GHZ | 32GB | Centos |

Blink server: 64CU

4.2 Experimental Results

The experimental results are shown in the table 2.

Table 2. Experimental test results

| The amount of data | Number of CU consumed | Average processing efficiency |
|--------------------|------------------------|------------------------------|
| 5000000            | 3.56                   | 15k/s                        |
| 10000000           | 3.78                   | 15.9k/s                      |
| 30000000           | 4.58                   | 14.7k/s                      |

5. Conclusion

Aiming at the current rule-based data quality monitoring system in terms of large data volume, single and complex matching, this article adopts the combination of multiple filtering rules and multiple calculation rules, combined with Blink's powerful stream computing processing ability, high throughput characteristics are designed and implemented. First, filter rules and check rules use the same model design. At the same time, we use task merging to enable a piece of data to be read for multiple tasks, which greatly improves computing efficiency and saves the use of computing resources. The verification results are entered into the MySQL version of different Alibaba analytical databases to ensure the efficiency of data entry in the big data scenario and reduce the impact of back pressure on the Blink cluster. It can be seen from the experimental results that the high-performance data quality monitoring system based on rule and artificial intelligence implemented in this paper has great advantages in practical application scenarios.

References

[1] Wang Guoxia, Liu Heping. Overview of personalized recommendation systems (J). Computer Engineering and Applications, 2012, 48[7]:66-76.

[2] The State Council. Several Opinions of the Central Committee of the Communist Party of China and the State Council on Further Deepening the Reform of the Electric Power System [R]. Beijing: Secretary Bureau of the General Office of the Communist Party of China, 2015.

[3] Li Xinpeng, Liu Wei, Yang Zhiping, et al. Research on the data center of power grid enterprises [J]. Electric Power Information and Communication Technology, 2020, 18(2):1-8.

[4] Internet Department of State Grid Corporation of China. Outline of Ubiquitous Power Internet of Things [R]. Beijing: State Grid Corporation of China, 2019.
[5] Han Jingyu, Xu Lizhen, Dong Yisheng. *A review of data quality research* [J]. Computer Science, 2008.

[6] Tamraparni Dasu, Theodore Johnson. *Exploratory data mining and data cleaning* [M]. *John wiley*, 2003.