ISSN 2616-5775 Vol. 6, Issue 7: 46-50, DOI: 10.25236/AJCIS.2023.060707

Research on an Unstructured Data Processing Strategy for Software Development

Ping Xu

School of Information Science and Technology, Taishan University, Tai'an, 271000, China

Abstract: *The management of unstructured data is considered to be a big problem in today's internet technology, because the tools and technologies that can effectively manage structured data in the past are not suitable for unstructured data. The key work of the unified storage platform for unstructured data is to realize the unification of data storage interfaces, and at the same time realize heterogeneous storage according to data characteristics to ensure the high availability and consistency of data. In traditional software development, relational database technology is the main way to record and update data. However, with the development of information technology, the format of these data is uncertain and it is very difficult to record them. Taking unstructured data as the research object, aiming at the problem of data recording in the process of software development, this paper puts forward a processing strategy of unstructured data in software development: batch processing based on unified storage platform, Hadoop computing framework and distributed unstructured data index framework, so as to solve the problem of data recording and updating in the process of software development.*

Keywords: *Software development; Unstructured data; Data management*

1. Introduction

The management and application of unstructured information is becoming more and more important. Due to the different hardware devices used for data acquisition, there are differences in the working mode and structure between the data acquired by the system and the data acquired by the acquisition module. In the process of data storage, it is necessary to select a big data storage platform to complete data outsourcing management, but the controllability of data will be affected. In order to maintain the rationality of data, it is necessary to comprehensively supervise the privacy information, and establish a data storage privacy control mode on the basis of ensuring the management of private users [1].

Traditional data management, especially relational database system, should only provide some surface management for unstructured data in application [2]. Therefore, how to manage unstructured data effectively, and how to mine data and knowledge, and how to extract the hidden information and support decision-making have become the main problems to be solved urgently.

2. Unstructured data type

Office documents, texts, pictures, images, audio and video information in all formats belong to unstructured data. The content requirements for different unstructured data analysis are different. The traditional full-text retrieval technology is based on keyword matching, and the results are difficult to meet the demand. Intelligent retrieval uses word segmentation dictionary, synonym dictionary and homophone dictionary to improve the retrieval effect, and combines the techniques of user retrieval context analysis and user relevance feedback to assist the query, giving users intelligent knowledge tips, and finally returning effective information to users accurately.

Most of the data flooding the Internet are unstructured data such as documents, pictures and videos with different formats. The management of these unstructured data is considered to be a big problem in today's Internet technology, because the tools and technologies that can effectively manage structured data in the past are not applicable to unstructured data [3-4]. With the rapid increase of unstructured data on the Internet, it is necessary to develop a unified storage platform that can efficiently manage these unstructured data, including unified data interface, heterogeneous storage, high availability and

consistency of data and other challenging issues. A complete unstructured data feature extraction process is shown in Figure 1.

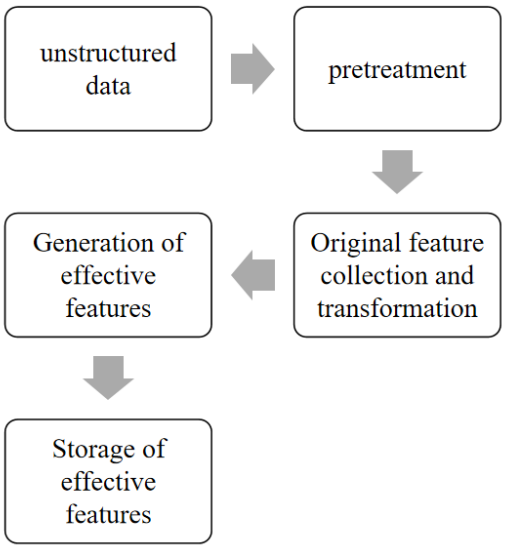

Figure 1: Feature extraction process of unstructured data

Unstructured data needs to be preprocessed first. After that, it is necessary to extract and transform the unstructured data after preprocessing. After obtaining the original features, it is necessary to generate effective features, the main purpose of which is to greatly reduce the feature dimension and reduce the computational complexity of the algorithm.

3. Unstructured data processing strategy

3.1. Batch processing based on unified storage platform

The key work of the unified storage platform for unstructured data is to realize the unification of data storage interfaces, and at the same time realize heterogeneous storage according to data characteristics to ensure the high availability and consistency of data. After the unified storage of data is completed, whether the characteristics of data storage can be fully utilized effectively is also an important technical problem [5-6]. Batch calculation is mainly used to calculate the total amount of a certain kind of data set. This kind of calculation is characterized by long calculation time and low real-time requirements, and the calculation involves a large amount of data, which is difficult to complete in a short time with a single machine.

In order to solve the key problem of efficient management of these massive, heterogeneous, related and unstructured digital resources, it is necessary to build a unified storage platform system for unstructured data, namely D-Ocean Repository [7]. The key of D-Ocean Repository, a unified storage platform for unstructured data, is to realize the unified storage interface of digital resources, and realize the efficient storage of digital resources on the basis of unified storage to ensure the high availability of data resources.

As shown in Figure 2, when the metadata management module is running, the data flow of metadata is controlled by the instantiated MetaManager, and the specific functions are realized through direct interaction with the bottom layer. When adding or deleting metadata, the corresponding metadata table is located according to the metadata type, and then MetaManager interacts with the bottom layer to modify the metadata in the database or file system accordingly. When querying metadata, the query interface is called to return metadata records of the specified type.

The metadata management module in the unified storage platform is mainly responsible for managing, accessing and updating the metadata information of the system, and providing necessary metadata information for system initialization, data access, analysis and indexing. The metadata of plug-in information needed for analysis, indexing and query are mainly definition files and plug-in configuration files, which are stored in the form of binary files [8].

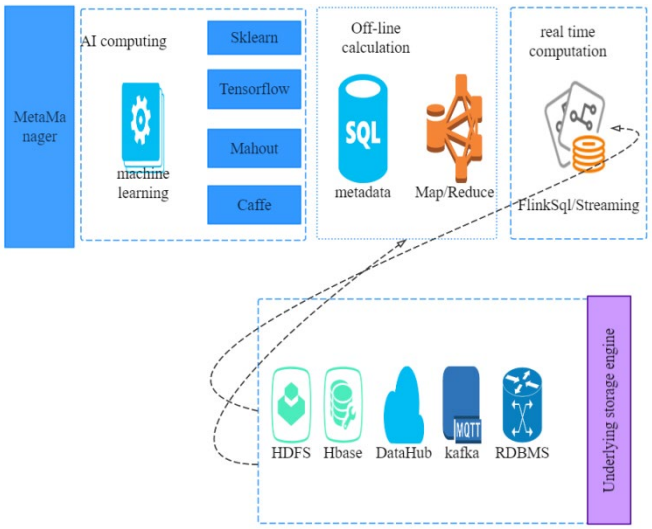

Figure 2: Schematic diagram of metadata management

3.2. Building Hadoop computing framework

HBase database technology is a data technology that can effectively handle multi-format small files. It is written and designed by Lucene retrieval engine, adopts hierarchical architecture, has the ability of full-text retrieval, can handle a large number of unstructured data files, and can be effectively applied to those unstructured data records with uncertain formats [9]. The development of unstructured data and software development industry presents a straightforward demand chain relationship, and in order to adapt to this data storage environment with increasing unstructured data, it is necessary to develop Hadoop-based document storage technology.

The difficulty in the management of unstructured data lies in the diversity of its formats, which requires the database to have high compatibility. Therefore, the database with unstructured data as the storage object must be supported by enough high-performance hardware. In order to solve this problem, open source computing methods are born.

Hadoop, as a distributed computing framework, adopts streaming data access mode. Hadoop was designed for the storage of large files from the beginning of writing, which makes Hadoop more compatible than traditional computing frameworks. In order to ensure that the Hadoop computing framework can be integrated with the management system being implemented in software development, designers generally design the Hadoop computing framework as a three-tier architecture, namely, the module resource center, the interface layer and the storage layer (Figure 3).

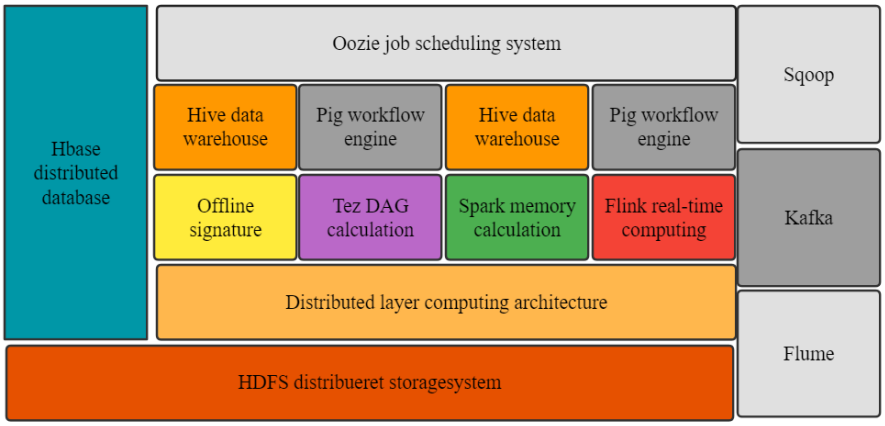

Figure 3: Hadoop computing framework design

Access to the underlying data is realized by two parts: access interface and functional module, and data storage and management are realized by three data technologies in storage layer. Structured data is still stored by Oracle database. In the process of building Hadoop computing framework, an extensible

and reusable module resource center must be designed according to unified development standards, and the services and data of HBase can run seamlessly with backup servers. It is also necessary to provide a dual-machine hot standby solution just in case.

3.3. Using distributed unstructured data indexing framework

Information retrieval system refers to a programmed system for information collection, processing, storage and retrieval established according to specific information needs, and its main purpose is to provide information services for people. The traditional method based on stand-alone index in database can't meet the index demand of massive data. Using distributed system to realize distributed indexing system can meet the indexing requirements of massive data. The framework of distributed full-text indexing system consists of three parts: index cluster, query cluster and distributed file system.

An index cluster contains an index master node and multiple index nodes. Using this structure, the indexing task can be decomposed into each index node, which ensures that each index node can establish indexes in parallel and improve the ability to process massive data. When the system stores each piece of data, it will send a system message of incremental indexing task to the index master node. According to the attribute and content of the data in the message, the index master node calls the index fragmentation strategy to determine the index fragmentation to which the data belongs, and stores the message in the distributed index message queue. In the query cluster, when there are new index files or some index files updated in the distributed file system, the query master node will receive the corresponding notification. The query master node allocates new index files according to the load of each query node. The node assigned to the index file updates the new index file locally and starts the query service.

Query cluster consists of three parts: query master node, query node and query client. The query cluster also adopts Master-Slave structure, which aims to ensure that the index files are quickly deployed to each query node, so as to improve the availability of the query service, increase the response speed of the query service and improve the query experience of users. Query nodes mainly provide query services for query clients. Users can publish queries through the query client, get the query results returned by each query node, and finally merge these query results. In this system, the log system is mainly used for system disaster tolerance [10]. When an index node fails, the system can restore the index operation that was successfully submitted according to the operation log of the index node. This system uses HBase to realize the log system.

Each index task can be divided into several index fragmentation tasks, and each index fragmentation task is the responsibility of each index node. Each index node is only responsible for establishing an up-to-date index sub-fragment, and then uploading it to the distributed file system. The system adopts a three-tier index structure as shown in Figure 4.

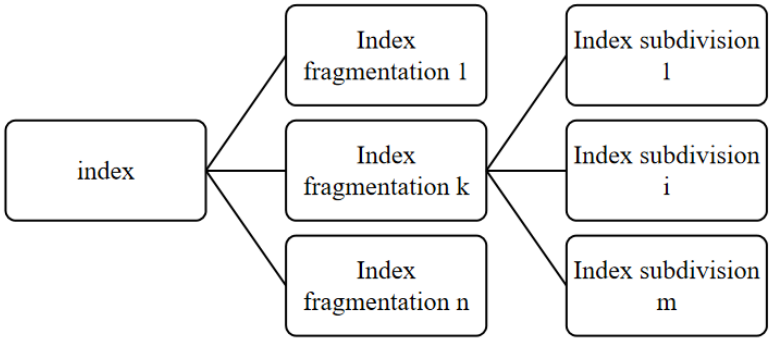

Figure 4: Index file hierarchy

A specific index file is composed of a specific number of index fragment files. Specifically, on the one hand, each index node is responsible for establishing each index fragment to solve the index consistency problem, and at the same time, multiple index nodes are responsible for one index task, so that data can be indexed in parallel and the indexing speed can be improved. On the other hand, each query node is responsible for each index sub-fragment, and provides index services to the outside world. At the same time, an index sub-fragment is distributed on different query nodes, which can improve the usability of distributed index.

4. Conclusions

Office documents, texts, pictures, images, audio and video information in all formats belong to unstructured data. The content requirements for different unstructured data analysis are different. The traditional full-text retrieval technology is based on keyword matching, and the results are difficult to meet the demand. Unstructured data needs to be preprocessed first. After that, it is necessary to extract and transform the unstructured data after preprocessing. After obtaining the original features, it is necessary to generate effective features, the main purpose of which is to greatly reduce the feature dimension and reduce the computational complexity of the algorithm. In this paper, the idea of batch processing based on unified storage platform, Hadoop computing framework and distributed unstructured data index framework is proposed to realize core functions such as file storage management and intelligent retrieval of the whole database.

References

[1] Mei J, Moura J. Signal Processing on Graphs: Causal Modeling of Unstructured Data [J]. IEEE Transactions on Signal Processing, 2015, 65(8):2077-2092.

[2] Adnan K, Akbar R, Wang K S. Development of Usability Enhancement Model for Unstructured Big Data Using SLR [J]. IEEE Access, 2021(99):1-1.

[3] Wang H, Tian Y, Yin H. Correlation Analysis of External Environment Risk Factors for High-Speed Railway Derailment Based on Unstructured Data[J]. Journal of advanced transportation, 2021(6): 2021.

[4] Bakaev M, Avdeenko T. Intelligent information system to support decision-making based on unstructured web data[J]. Icic Express Letters, 2015, 9(4):1017-1023.

[5] Liu Y, Yin C, Qiu C, et al. 3-D inversion of transient EM data with topography using unstructured tetrahedral grids[J]. Geophysical Journal International, 2019(1):301-318.

[6] Han J, Yang X, Li H, et al. Error Correction of Measured Unstructured Road Profiles Based on Accelerometer and Gyroscope Data[J]. Mathematical Problems in Engineering, 2017, 2017(8):1-11.

[7] Selvaraj G, Taboada K, Gonzales E, et al. Content enrichment with expressive document modelling to leverage the understanding of unstructured data[J]. MATEC Web of Conferences, 2019, 277(42-49):02003.

[8] Xin H, Xu Y. Research on Semi-Structured and Unstructured Data Storage and Management Model for Multi-Tenant[J]. Journal of information technology research, 2019, 12(1):49-62.

[9] Madhusudhanan S, Jaganathan S, Jayashree L S. Incremental Learning for Classification of Unstructured Data Using Extreme Learning Machine[J]. Algorithms, 2018, 11(10):158.

[10] Vo N, Liu S, Li X, et al. Leveraging unstructured call log data for customer churn prediction[J]. Knowledge-Based Systems, 2021, 212(4):106586.