Research of the Data Mining Engine Based on Big Data

Mei Wan

Guangzhou College of Technology and Business, Guangzhou, Guangdong, 510850, China

Keywords: Big data, Data mining, Search engine

Abstract: With the rapid development of Internet technology, people are accumulating more and more data, and the scale of data has risen from the previous GB level to IB or even PB. In order to discover the potential value in the data, it is common practice to flexibly use various data mining algorithms according to the actual situation. Although data mining has been fully utilized and developed on traditional small data sets, proving its value and guiding significance, on large data sets, the implementation of data mining algorithms faces execution efficiency, algorithm parallelism, and easy platform Usability challenges. In order to solve the problems of data mining in big data, the paper researches the data mining engine under big data, using Spark as the core engine, and based on Spark's memory computing operators, a number of traditional data mining The parallel computing of the algorithm enables the traditional data mining algorithms to run in parallel in a cluster environment, and thus is well applied in big data. Then, through the system layering method, the data mining system is designed in layers to realize a complete big data mining platform.

1. Introduction

Due to the popularity of computers and the development of the Internet, the Internet is becoming more and more widely used in people's daily life and work. Every day, a large number of people use the Internet, and a large amount of data is also generated. Over time, people are accumulating more and more data, up to terabytes and even petabytes. In order to obtain useful information in these data, people use various data mining algorithms to mine the potential value in it. Although the application of data mining on small data sets has good results, for large data sets, there are problems in the execution efficiency, parallelization of the algorithm, and ease of use of the platform. In order to solve the above problems, the paper studies the data mining engine under big data. Using Spark as the core engine, and based on Spark's memory calculation operators, multiple parallel data mining algorithms are implemented in parallel to make the traditional the data mining algorithm can run in parallel in a cluster environment, so it is well applied in big data. Then, through the system layering method, the data mining system is designed in layers to realize a complete big data mining platform.

2. Data Mining Overview

Modernly developed data mining algorithms can be divided into the following four categories: Association rule analysis algorithms: This type of algorithm is usually used to find two types of data with greater relevance. For example, when buying breakfast, it is found that the buns are purchased. Everyone chooses to buy a pack of paper, so providing paper in the bun shop will increase sales of both. Clustering algorithm: This type of algorithm is used to find similar items between data, which is equivalent to generalizing data with a certain attribute. Prospective regression algorithm: This type of algorithm is mainly used to forecast the trend of big data. Commonly, there is a linear regression algorithm. Sorting algorithm: This type of algorithm sorts related data by specifying a certain index and the number of this index. The classic is Google's PageRank algorithm. Orange, Apache Mahout, and Spark are common data mining tools. These three tools are applied to different scenarios for data mining. For example, Orange uses python to implement data mining. As a technology that has been developed for more than ten years, the predecessors have developed many classic big data mining algorithms and tools. These algorithms and tools still have certain

  DOI: 10.25236/ictmic.2020.013

application limitations. For example, such algorithms cannot be applied to data sets. Dig. The main reasons are the following five points: The memory footprint of big data is surprisingly large, and traditional software simply cannot afford such a large amount of data. For example, traditional relational databases appear to be incapable of performing relevant operations on big data. The way of storing, reading and calculating big data is different from traditional data. At the same time, in order to improve the processing power of big data algorithms, many models and computing frameworks are parallel. Generally speaking, big data mining is performed on computer groups, which not only has high requirements on equipment, but also has higher requirements on related practitioners. There are few data mining tools for large data sets, the operation is difficult, the user-friendliness is low, and it is difficult to realize data visualization. To process data transmitted in real time, request a transmission system with high availability and large throughput.

3. Big Data Related Technologies and Tools

Before explaining the related technology of big data storage, we first review and describe the evolution of data and storage methods. Earlier, data was stored directly in files. The shortcomings of file storage are obvious. The lack of formatted information makes it difficult to support operations such as query, insert, and delete. Then relational databases appeared, such as Oracle and MySQL. Relational database is now the mainstream way of data storage, because it very well supports data addition, deletion, modification, query, database transaction, stored procedure and other functions. General relational database, the performance is still in the hundreds of thousands of data levels. However, when the amount of data increases, the efficiency of the stand-alone database decreases significantly, and another disadvantage is that once the stand-alone fails, the database service cannot be provided, which can even cause data loss. In recent years, providing database services in a cluster mode is a new data storage method. The designer splits the database according to certain fields, spreads the data to different machines, reduces the pressure on a single machine, improves the data storage capacity and the performance of the entire system. In addition, multiple database backups are performed to ensure that a certain machine will not cause the entire application to be unavailable. However, this distributed service also has some costs, such as support for database transactions, database consistency, and the complexity of multi-table queries, all of which pose challenges to developers.

Big data processing not only involves analyzing the data that has been stored on the hard disk, but also performing real-time analysis of streaming data is another important aspect. The process of streaming data is generally to obtain data from data sources, such as log output, socket, etc., / n? To filter and analyze the data in the data stream, the data must be stored persistently to the file system. This process is similar to the process of data extraction, transformation, and loading in traditional data warehouse technology, that is, ETL. Big data streaming data has the following characteristics: large amount of data, high real-time requirements, and high throughput. In order to reduce the overall compatibility of the system and enable the streaming data analysis system to be flexibly added to the existing system, data stream middleware should be used to receive the streaming data generated by the data source, and 11 to pass the data to the streaming The data processing engine is then persisted to HDFS. The two most representative data flow middleware are Apache Flume and Apache Kafka. The basic bit of data transmitted by Flume is sufficient for an event. For example, a line record of a text file is usually an event. The core of Flume's operation is the agent. It is a complete data collection tool, consisting of three core components: Source, Channel, and Sink. Event data is sent from the Source to the Sink through the Channel in the form of a byte array. Event represents the smallest complete unit of a data stream, which comes from an external data source and goes to an external cause. The main function of Source is to collect streaming data, convert it into events and send to Channel; Channel provides a queue data structure, caches the data provided by Source, and removes the data in the queue. Sink takes out the data in the Channel, filters it, and selects it for processing, and then persists it to the distributed file system or passes it to downstream processing tools. Through these components, the processing of events is completed. A Source receives events from external sources, and it can send events to one or more channels. A

channel is a queue that buffers events waiting for sink to consume in a first-in-first-out manner. Sink takes out the event buffered by the channel towel and hands it to downstream processing. The most agile way to access Flume in the existing program is to directly read the log files originally recorded by the program, which can basically achieve seamless access without any changes to the existing programs. Apache Kafka is a distributed messaging service. Compared with traditional messaging systems, Kafka is more described in scenarios with big data and high availability requirements, mainly because of its features: First, it is a distributed architecture subscription publishing system. Publishers, subscribers, and processors can be established in a cluster manner, which is in line with the application scenario of big data and expands. Second, it supports multiple subscribers, which means that a Kaflca cluster can cope with the same message in the system. There are scenarios for different approaches. Third, it persists message data to the hard disk, so it supports batch consumption methods such as ETL and real-time processing.

4. Construction of Data Mining Engine

This article intends to use Spark as the basic engine. Because of its obvious speed advantage over Mabout, especially in the context of big data processing. Spark, as a programming model with RDD (Resilient Distributed Data Set) and shared variables, not only enables the memory of all nodes to be parallel, but also replicates shared vectors at different nodes. Users can use a certain RDD again without having to re-establish a new RDD, which greatly improves its operating speed. But at present, Spark still has some disadvantages, for example: there are only eight algorithms, the coverage is small, and there is no PageRank algorithm and association algorithm (Apriori algorithm) involved. Therefore, this paper mainly combines these two algorithms to optimize Spark. First, the association algorithm is a basic data mining method, which can mine events related to big data events within a set value range. The traditional association algorithm traverses the entire database during each calculation request. When the amount of data is small, the advantage of the association algorithm is obvious. However, if it is facing a large amount of big data, it is difficult to complete the entire work efficiently and quickly. The characteristics of Spark can be used to optimize the correlation algorithm. We can store the data in RDD, perform the first stage calculation, and then perform iterative calculation. Fan Jiaqi of Beijing University of Posts and Telecommunications has successfully established such a parallel algorithm model. The PageRank algorithm can also be optimized using Spark. This type of algorithm was first developed by Google. By calculating the number of specific links on a web page, the pages are sorted in order from the number of links.

The design of the entire system should consider practicality and efficiency, so it must be user-friendly and easy to use, and in order to improve the user's efficiency, the underlying transparency must be achieved. Therefore, the entire system is structured into three levels during design: The data processing engine is the lowest level, and the main body of this engine is the Spark cluster. As the core layer of data mining, it mainly includes three components: SparkSQL, data mining algorithms, and Spark Streaming. These three components each have their own functions, namely: sentence query function, data mining algorithm, streaming data processing. The second layer is the middle layer. This layer is mainly used to solve the problem of multiple users requesting control and remote calling at the same time. Not only can users directly make data calls locally, but also ensure that multiple users can perform data processing at the same time. The top layer is the user layer. This layer is designed to allow users to directly access the system and includes the Orange plug-in.

5. Conclusion

Big data-based data mining algorithms have problems in their execution efficiency, algorithm parallelism, and platform usability. In order to solve these problems, this paper has carried out research calculations on the data mining engine under big data, so that the traditional data mining algorithms can run in parallel in a cluster environment, and thus get better application in big data. Then, through the system layering method, the data mining system is designed in layers to realize a

complete big data mining platform. The experiments show that the parallel calculation of Apriori algorithm and PageRank algorithm based on Spark can effectively reduce the execution time and has a good application in big data mining.

References

[1] Wang Xiaoyan, Zhang Limin. Research on Data Mining Engine Based on Big Data [J]. Electronic Design Engineering, 2017, 25 (15): 31-34.

[2] Zhang Xueyong, Wu Yuling. Research Progress of Empirical Asset Pricing Based on Network Big Data Mining [J]. Economics Update, 2018, No.688 (06): 131-142.

[3] Gao Minghao. Research on IoT data mining based on ship big data platform [J]. Ship Science and Technology, 2018.

[4] Kong Jiankun, Qiu Weina, Wang Zhiguo, et al. Research on Big Data Open Engine Based on User Network Portrait [J]. Shandong Communications Technology, 2017 (1).

[5] Du Yan, Wang Juquan, Zhang Jiacheng. Research on Fire Service Data Analysis Based on Big Data [J]. Telecommunications Letters, 2017 (10): 25-29.

[6] Li Jun. Research on Big Data Credit Information Management Platform Based on SOA [J]. Computer Products and Distribution, 2018 (04): 106-113.