Neurocomputing 181 (2016) 139–146

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Big traffic data processing framework for intelligent monitoring and recording systems

Yingjie Xia [a], Jinlong Chen [a,b,*], Xindai Lu [c], Chunhui Wang [b], Chao Xu [b]

[a] College of Computer Science, Zhejiang University, Hangzhou 310012, China
[b] Hangzhou Institute of Service Engineering, Hangzhou Normal University, Hangzhou 311121, China
[c] Electric Power Research Institute, State Grid Zhejiang Electric Power Company, Hangzhou 310000, China

ARTICLE INFO

Article history:
Received 14 March 2015
Received in revised form
23 June 2015
Accepted 7 July 2015
Available online 12 December 2015

Keywords:
HBase
Hadoop
Intelligent monitoring and recording system
Big traffic data

ABSTRACT

In recent years, intelligent monitoring and recording system (IMRS) has been widely used in many cities. It is a networked system which includes the front-end image acquisition system and the back-end data processing platform. In the front-end acquisition system, after recording the image data the front-end processing system analyzes the image data and automatically extracts some information of the passing vehicles, such as time, location, direction, the number and color of license plate, etc. After that, the information is sent to the back-end data processing platform for deeper analysis, such as vehicle trajectory tracking and traffic state estimation. However, as the data scales to a great amount, it becomes difficult for traditional analysis framework and tools to meet the requirement of big data analysis. HBase is the column-oriented database based on Hadoop, along with Map-Reduce programming paradigm. It can deal with the problems in processing big data that intelligent monitoring and recording system faces nowadays. In this paper, a big traffic data processing framework using HBase to analyze the data of IMRS is proposed. The performance of the proposed framework is compared with the Oracle strategy on three different kinds of applications, and the experimental results show that the proposed strategy is with promising performance in computation latency.

1. Introduction

With the development of modern surveillance equipment, intelligent monitoring and recording system (IMRS) has been widely used for monitoring traffic system in cities. IMRS can automatically capture the moving vehicle and recognize the vehicle features. We can benefit from the IMRS in the areas such as dynamic road control, traffic management, crimes striking and prevention, and emergency disposal.

IMRS is made up of two components, namely front-end data acquisition system and the back-end data processing platform. The front-end system is in charge of capturing the image data of the passing vehicles, identifying the vehicle features, and sending the vehicle information to the back-end data processing platform [1]. In the back-end platform, many machine learning algorithms are used to filter the data and mine valuable information. The traditional data storage and analysis schemas of IMRS are mainly based on the Relational Database Management System (RDBMS), which can only support some simple database operation tasks such as query, insert or some statistical applications. However, as the scale of data amount increase in a great deal, and the data type becomes more diversed, the traditional RDBMS is unable to deal with these Big Data both in storage and processing.

In order to solve the problem mentioned above, this paper proposes a HBase-based [2,3] big traffic processing framework to improve the vehicle behavior analysis efficiency. The proposed framework provides IMRS the ability to deal with the large scaledata. HBase originates from a technique named BigTable [4] which was proposed by Google for dealing with the big data storage and retrieval task. With the HBase, Hadoop is able to support real-time random access to very large datasets. The advantage of HBase is the high scalability, high availability and fault-tolerance in distributed data storage and real-time big data processing. Although it is not relational and does not support SQL, it can provide queries in large scale sparse dataset on distributed clusters.

The proposed HBase-based big traffic data processing framework for IMRS faces two main challenges. The first challenge is that in HBase the records are stored based on KVS (Key-value Store) [5,6]. But KVS only supports simple search functions. To solve the challenge, we propose a three-table based schema for adapting different requirements in three vehicle behavior analysis cases on IMRS. The second challenge is the data transmission problem. HBase integrated

* Corresponding author.
E-mail address: xtommy@163.com (J. Chen).

http://dx.doi.org/10.1016/j.neucom.2015.07.140
0925-2312/

MapReduce which has the ability to analyze the distributed data. However, in the data processing progress based on MapReduce, the big data transmission will generate huge network traffic which will decrease the performance with the data scale increment. To solve this challenge, HBase Coprocessor (HC) can be used in the data analysis process which requires the HC programs to be deployed on the same physical machine with the data. It avoids the data transmission between different machines and reduces the network traffic overhead.

Overall, our contribution can be summarized as follows:

(1) We propose an HBase-based big traffic processing framework to solve the big data storage and analysis problem for the vehicle behavior analysis.
(2) We propose a method to design appropriate Rowkey of HBase to increase the data query speed a lot.
(3) We introduce HBase Coprocessor based distributed framework to process the data to improve the computing efficiency.

The remainder of this paper is organized as follows. First, Section 2 presents some related research work in big data processing from several areas. Section 3 introduces the proposed framework. Then we introduce a solution for IMRS data storage based on KVS in HBase table in Section 4. In Section 5, we describe a solution to solve the data transmission problem. In Section 6, we describe three cases to illustrate the process of big data analysis based on HBase in IMRS. The experiments study the HBase performance by comparing the processing efficiency of our framework with the three cases in Section 7. At last, we conclude the paper and present directions of future work.

2. Related work

In this section, we will give a review on some related work of big data storage and processing with traditional solutions and Hbase-based solutions.

2.1. Traditional solutions for big data storage and processing

In recent years, researchers have recognized the limitations of traditional relational database mainly on the big data distributed storage and processing. To solve the shortcoming of traditional database, various systems have been designed. For example, distributed relational database management systems (DRDBMSs) and in-memory computing [7] are respectively developed for improving capacity of data storage and computing efficiency. However, DRDBMSs is unable to maintain and retrieve data among servers efficiently for the reason that it takes too much time to implement the data consistency [8]. Besides, DRDBMSs are severely restricted by the relational database feature set, e.g. the joins, complex queries, triggers, views, and foreign-key constraints make it difficult to run on a scaled relational database. In-memory computing is the storage of information in the main random access memory (RAM) of dedicated servers rather than complicated relational databases operating on comparatively slow disk drives, for real-time processing big data. It can be used to analyze the common data and load the data probability into memory, so as to improve the efficiency of data processing.

However, because of the shortcomings of linear scalability, the size of the data processed by in-memory computing are limited [9]. DRDBMSs and in-memory computing are not designed for large-scale and scalable distributed processing, since all of those solutions are unable to achieve large-scale linear expansion. Xia et al. [10] designed a parallel computing framework by formalizing computational intensity of big traffic data. A computational domain theory is proposed to formally represent heterogeneous big traffic data and evaluate the computation intensity, which is leveraged to decompose the domain into sub-domains for load-balanced parallel computing. However, this computational domain supports only the structured data, and the traffic image and video in IMRS cannot be directly and accurately encapsulated into the domain.

In addition, some other systems have also been developed to meet the requirement of computing scalability and availability, such as Cassandra [11] and SimpleDB [12], however, both of them provide weak consistency. SimpleDB is another service from Amazon that offers HBase the functionalities which are like those of HBase. However, the value of HBase is an uninterpreted array of bytes, while SimpleDB can only store strings SSDS [13] which has string, number, datetime, binary and boolean data types. Dynamo [14] is another distributed storage system of Amazon. It focuses on writing. Microsoft Boxwood Project [15] provides components with similar functionality as HDFS and HBase. However, Boxwood is just a research project and there is no performance evaluation for large scale of deployment and practical applications [16].

2.2. HBase-based solutions for big data storage and processing

In this paper, HBase-based vehicle tracking is introduced to improve the computing efficiency. HBase belongs to the Hadoop Ecosystem. As an open-source software, the Hadoop platform is easy to program and run applications for processing big data. And the distributed storage [17] and processing [11] strategies are adopted by Hadoop [6]. In this system, the distributed computing framework MapReduce stores data in the distributed file system (DFS) [12], which guarantees the data analysis and processing efficiency [18]. And the redundancy backup mechanism is used to ensure the data security. Moreover, the high fault tolerance of Hadoop allows that it can be deployed on cheap server cluster [19]. Hadoop is made up of distributed file system (HDFS) and distributed computing framework (MapReduce).

As a part of the Hadoop ecosystem, HBase which built on the basis of HDFS, is a column-oriented and scalable distributed storage system with high reliability and performance. It is designed to solve the big data processing problem that traditional relational database faces today. The traditional relational database is lack of scalability and reliability because of the data consistency [20]. While HBase is designed for big data storage and high-speed reading and writing, these data can be accessed at high speed by a large number of concurrent users. Google cooperates with INRIX to use the collected massive GPS data from more than 30,000,000 taxies, trucks, etc. to estimate the traffic states. The massive GPS data are stored on HBase and processed by Hadoop.

In this paper, we design a HBase-based framework using the scalable distributed storage capacity and processing power provided by HBase for big traffic data analysis in the IMRS system.

3. Overview of the proposed framework

Fig. 1 shows the HBase-based framework for the big data analysis in IMRS. The whole process can be divided into three phases. Firstly, the front-end data acquisition system collects the data related to the vehicle passing records which contain the vehicle plate number and passing time information [21]. Those data will be sent to the back-end data processing platform which is built on HBase clusters [22] through IMRS network and distributed storage in the HBase table [23]. Secondly, the back-end data analysis is based on the HBase table structure which greatly affects the data query operations [24]. Thirdly, the HBase Coprocessor is introduced to provide distributed computing capacity and reduce the