International Journal of Engineering & Technology, 7 (3.33) (2018) 248-251

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Development of real-time big data analysis system for RHIPE-based marketing in the automobile industry

Young-Woon Kim[1*], Hyeopgeon Lee 2

[1]*Department of Data Analysis, Seoul Gangseo Campus of Korea Polytechnic University*
[2]*Department of Data Analysis, Seoul Gangseo Campus of Korea Polytechnic University*
**Corresponding author E-mail: luckkim@kopo.ac.kr*

Abstract

In the automobile industry, the contract information of vehicles contracted through sales activities, as well as the order data of customers who purchased cars, and vehicle maintenance history information all accumulate in relational databases over time. Although accumulated customer and vehicle information is used for marketing purposes, processing and analyzing this massive data is difficult, as its volume con-stantly increases. This problem of managing big data is commonly solved by utilizing the MapReduce distributed structure of Hadoop, which uses big data distributed processing technology, and R, which is a widely used big data analysis technology. Among the methods that interconnect Hadoop and R, the R and Hadoop integrated programming environment (RHIPE) was developed in this study as a real-time big data analysis system for marketing in the automobile industry. RHIPE allows us to maintain an interactive environment and use the powerful analytical features of R, which is an interpreter language, while achieving a high processing speed using Map and Reduce func-tions. In this study, we developed a real-time big data analysis system that can analyze the orders, reservations, and maintenance history contained in big data using the RHIPE method.

Keywords: *Big Data Analysis System, Hadoop, R, RHIPE, MongoDB, MapReduce*

1. Introduction

In the automobile industry, the contract information of vehicles contracted through sales activities, as well as the order data of customers who purchased cars, and vehicle maintenance history information all accumulate in relational databases over time. Although accumulated customer and vehicle information is used for marketing purposes, processing and analyzing this massive data is difficult, as its volume con-stantly increases. This problem of managing big data is commonly solved by utilizing the MapReduce distributed structure of Hadoop, which uses big data distributed processing technology, and R, which is a widely used big data analysis technology [1]. Furthermore, the big data stored in MongoDB must be extracted and aggregated for real-time big data analysis.

In order to construct a real-time big data analysis system in the automobile industry, we should select an appropriate method to interconnect R and Hadoop, as well as a method to extract large volumes of data from MongoDB.

Firstly, we compared the advantages and disadvantages of RHIPE, RHadoop, and Hadoop streaming, which are methods for interconnecting R and Hadoop; we selected RHIPE which can quickly process Map and Reduce and can enable the use of powerful analysis features of R.

Secondly, we compared the advantages and disadvantages of Aggregation Framework, MapReduce, and Group, which are extraction and aggregation methods for big data in MongoDB; we selected Aggregation Framework, which is designed for performance and development productivity.

2. Literature review

Hadoop is a distributed processing framework that uses clusters comprising multiple computers to process massive amounts of data. It comprises a middleware in the form of an engine and a software development framework.

In contrast to transaction processing, which responds instantly, it is designed to first collect the data to be processed and to respond upon completion of the request. Therefore, it is suitable for processing massive amounts of data, which requires a certain amount of time.

Hadoop fundamentally uses the distributed processing structure of MapReduce. MapReduce is composed of a Map phase, in which a single datum is divided into many pieces for processing, and a Reduce phase, in which the processed results are combined to extract a single result.

In order to perform MapReduce, it must be accessible from the entire system and be able to save a large amount of data. Hadoop uses HDFS (Hadoop Distributed File System). In other words, it is composed of a MapReduce module and HDFS[1,6,7,12].

HDFS is designed to save large files of over tens of terabytes (TB) or petabytes (PB) to the distributed server and to process many clients quickly. It can configure the storage using a low specification server[2,4,8,9,12].

NoSQL (Non-Relational Operation Database SQL) is a storage method that is able to accommodate the characteristics of big data such as large-sized, atypical, and real-time; and it stores and manages big data with high performance.

As a new data storage technology, NoSQL has the following three advantages that cause it to be more desired than RDBMS.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

First, it is open source and can be edited. In addition, it is freeware that can be developed at a low cost; hence, it is suitable for cloud compu-ting environments, which require flexible system architecture.
Second, it is designed with an atypical data structure so that it can avoid joins and is able to guarantee effective management performance.
Third, NoSQL technology is mostly implemented by memory mapping, so it is faster than RDBMS in reading/writing the big data. Because it can also be implemented on existing operating systems and hardware, it has greater flexibility and expandability.
Among the major NoSQL databases based on open source, Mongo DB is a document-oriented database that is reliable and scalable.
MongoDB, which aims for low management cost and convenient usability with big data, was developed by 10gen with open source and hence it can be commercially supported.

The minimum storage unit in MongoDB is a document. Each document is collected at "collection" and each collection is managed at the database to support the scope query, secondary index, alignment operation, and set operation of MapReduce.
MongoDB collects the document by collection and does not need a schema. A MongoDB query is created in JavaScript and document based query is conducted. This is a shell with real-time access and supports multiple program languages.
MongoDB is able to expand distribution by using auto-sharding. Sharding is a process in which data are divided and stored on different servers separately. By storing data to multiple servers, more data can be managed and processed, as shown in Fig. 1[3, 5, 10,11,12,13]

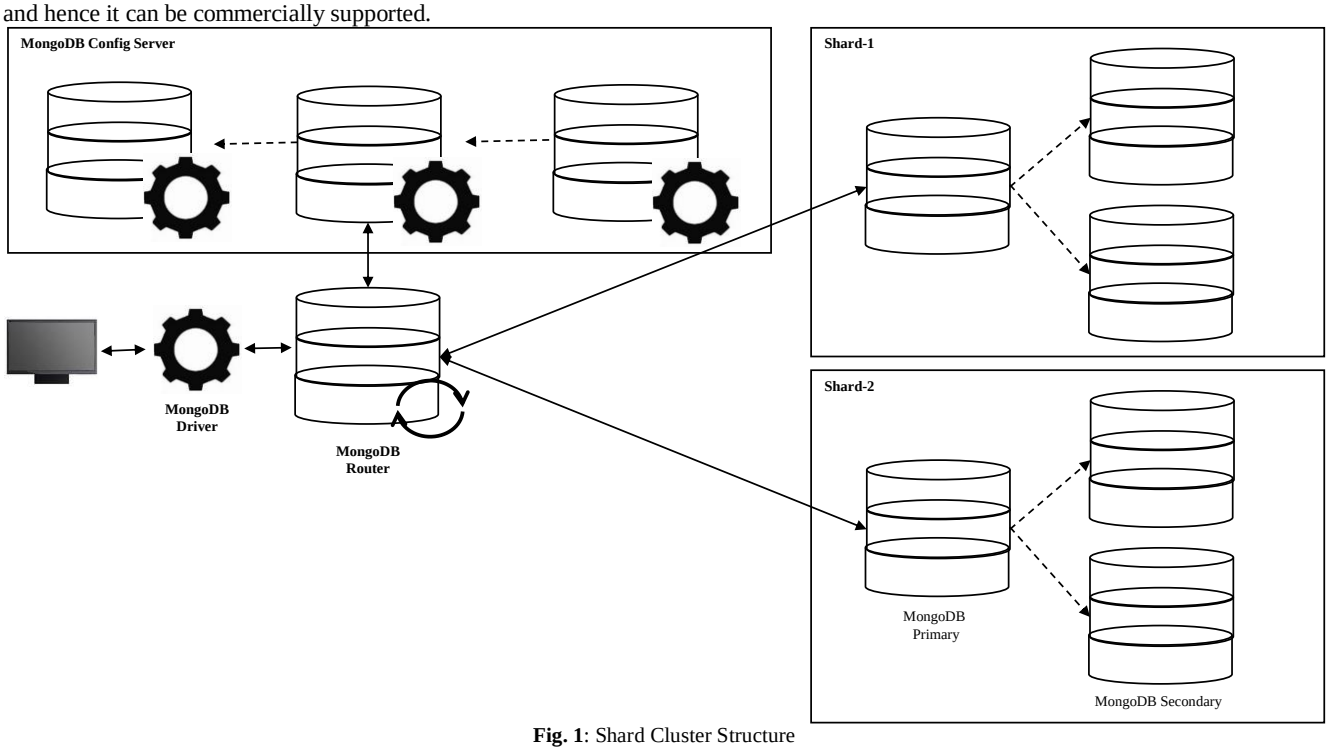

Fig. 1: Shard Cluster Structure

MongoDB provides three methods for the extraction and aggregation of large data.

- Aggregation Framework: Designed for the improvement of performance and development productivity, the Aggregation Framework can operate in both non-shard and shard cluster environments.
- MapReduce: The MapReduce provides functions for fast processing of large data sets and operates in both non-shard and shard cluster environments.
- Group: Simple syntax and functions are provided for grouping by a designated key, but the Group does not operate in a shard cluster environment.

R is the most widely used statistics-package freeware for analyzing big data and has the following advantages.
First, regardless of the statistical technique employed, the R package is already built in. Furthermore, R is scalable and offers a variety of features for creating tools and methods for data analysis.
Second, it is an unmatched package in terms of graphics and chart features. It facilities work through the dplyr and ggplot2 packages for data manipulation and plotting.
Third, R is an easily accessible language. Even those with no basic knowledge of programming can use it.

3. Proposed Work

Three means of interconnecting R and Hadoop exist for big data processing and analysis.

3.1. Using RHIPE

RHIPE is a software package that helps users perform MapReduce jobs for R users, as shown in Fig. 2[14]. Initially, RHIPE performs a MapReduce job by dividing big data into subsets using an analytical technique called Divide and Recombine, applies a numeric or visualiza-tion method to each subset, and then recombines the results. A major aspect of a MapReduce job is that it is conducted in a perfect R envi-ronment as soon as it uses the R equation. Therefore, it can quickly divide functions between Map and Reduce functions, and maintain the interactive environment of R while using it powerful analytical features.

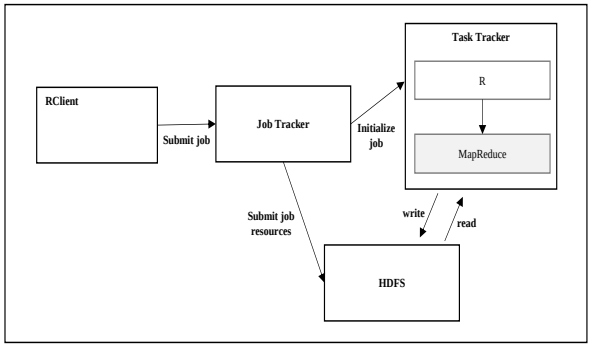

Fig. 2: Components of RHIPE

RHIPE was designed for the following two purposes:

1. Enable in-depth analysis of both large and small amounts of data.
2. Enable analysis with R using a lower-level language. For RHIPE, functions have been designed to perform the Hadoop distributed file system (HDFS) as well as MapReduce jobs using R console.

Many Hadoop components are used for data analysis of R and Hadoop. RHIPE has the following components.

- Rclient: Rclient is an R application that calls JobTracker, which runs jobs, according to the instructions of MapReduce job resources.
- JobTracker: JobTracker is a master node of a Hadoop MapReduce job used to initialize and monitor MapReduce jobs from a Hadoop cluster.
- TaskTracker: TaskTracker is a slave node of the Hadoop cluster. MapReduce is conducted using the sequence defined in JobTracker.
- HDFS: HDFS is a file system distributed in Hadoop clusters that have a data node.

3.2. Using RHadoop

RHadoop runs the R program on the Hadoop platform, as shown in Fig. 3[14]. RHadoop is a type of library package for R language and has the following modules:

- rmr, which provides an interface with MapReduce. Thus, it can create Mapper and Reducer with R code and load them into R.
- rhdfs, which provides access to HDFS. It can copy data between the R memory and HDFS using simple R code.
- Rhbase, which is used to interface with HBase.

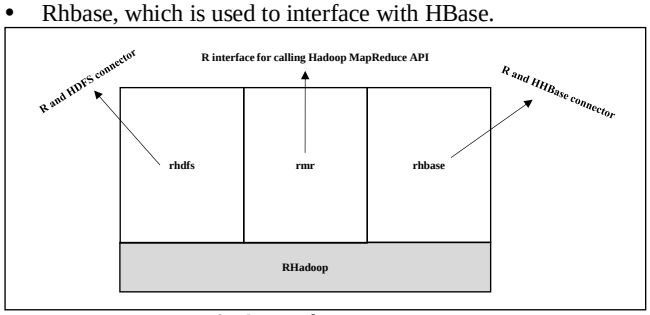

Fig. 3: RHadoop Ecosystem

3.3. Using Hadoop streaming

Hadoop streaming is a Hadoop utility for performing Hadoop MapReduce jobs that have executable scripts such as Mapper and Reducer as shown in Fig. 4[14]. It is similar to a pipe job in Linux. The advantage of the Hadoop stream utility is that it supports not only MapRe-duce jobs programmed with Java and that run in the Hadoop cluster, but also jobs programmed with non-Java.

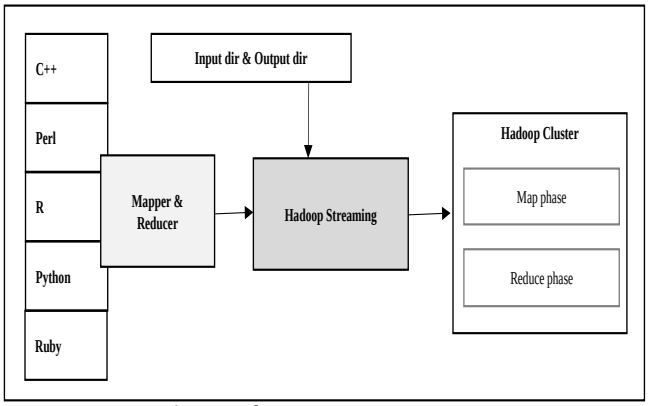

Fig. 4: Hadoop streaming components

RHadoop and Hadoop stream offer the following advantages: the possibility of programming MapReduce using R code on the Hadoop platform; high flexibility in using R, as it is well-integrated in the R environment; and the possibility of using various R packages. However, they also possess the disadvantages of requiring expert knowledge of MapReduce and extensive coding. These present potential obstacles to building a big data system. Therefore, this study used RHIPE, which enables the use of R's powerful analytical features while maintaining its interactive environment, as R is an interactive language. In addition, RHIPE can quickly process functions using the Map and Reduce functions.

RHIPE interconnects R and Hadoop, and is installed using the following process.

1. Install Hadoop : Install Hadoop in single node or multinode depending on the amount of data to be analyzed.
2. Install R : If you choose to use multimode Hadoop, you must install R in every TaskTracker node because multiple TaskTrackers are required to run MapReduce jobs.
3. Install Protocol Buffers : Protocol buffers must be installed to serialize data over the network.
4. Set environmental variables : Environmental variables must be set for RHIPE to perform compile correctly. Set PKG_CONFIG_PATH and LD_LIBRARY_PATH to configure Hadoop libraries.
5. Install rJava Package : RHIPE is a Java package and works like a Java bridge between R and Hadoop. Therefore, rJava must be in-stalled to enable the functions of RHIPE.
6. Install RHIPE : Download RHIPE from the repository and install the RHIPE package [6].

To select a method for extracting and aggregating big data for marketing applications, we compared the advantages and disadvantages of Aggregation Framework, MapReduce, and Group as shown in Table 1.

The advantages and disadvantages of the three methods that enable extraction and aggregation of large data and operate in a shard cluster environment were compared. The shard cluster utilizes a distributed processing technology that uses a cluster of several computers. Among the three methods, Aggregation Framework was selected because it was designed for the improvement of performance and development productivity.

Table 1: Comparison of advantages and disadvantages of big data extraction and aggregation methods in MongoDB

Category	Aggregation Framework	MapReduce	Group
Advantages	• Designed for the improvement of performance and development productivity • Operates in both non-shard and shard cluster environments	• Provides functions for fast processing of large data sets • Operates in both non-shard and shard cluster environments	• Provides simple syntax and functions for grouping by a designated key • Returns the results in-line arranged in an array of grouped items
Disadvantages	• Limits the size of the results document to 16 MB • Allows only the supported operators and expressions, and does not support user-defined functions	• MapReduce functions are difficult to debug. • Not intuitive to programmers who have experience with relational aggregation queries	• Processing takes a long time; hence, this method is used only when necessary. • Does not operate in a shard cluster environment

4. Conclusion

In the automobile industry, methods exist to interconnect Hadoop and R in order to analyze massive amounts of customer and vehicle in-formation. Among these methods, RHIPE allows us to use the powerful analytical features of R while maintaining its interactive environ-ment, and quickly process functions using Map and Reduce functions. In this study, we developed a real-time big data analysis system that can analyze the orders, reservations, and maintenance history contained in big data using the RHIPE method. The system saves the data in a MongoDB for real-time analysis using various search conditions. These search conditions can be used for marketing purposes in the auto-motive industry, as shown in Fig. 5.

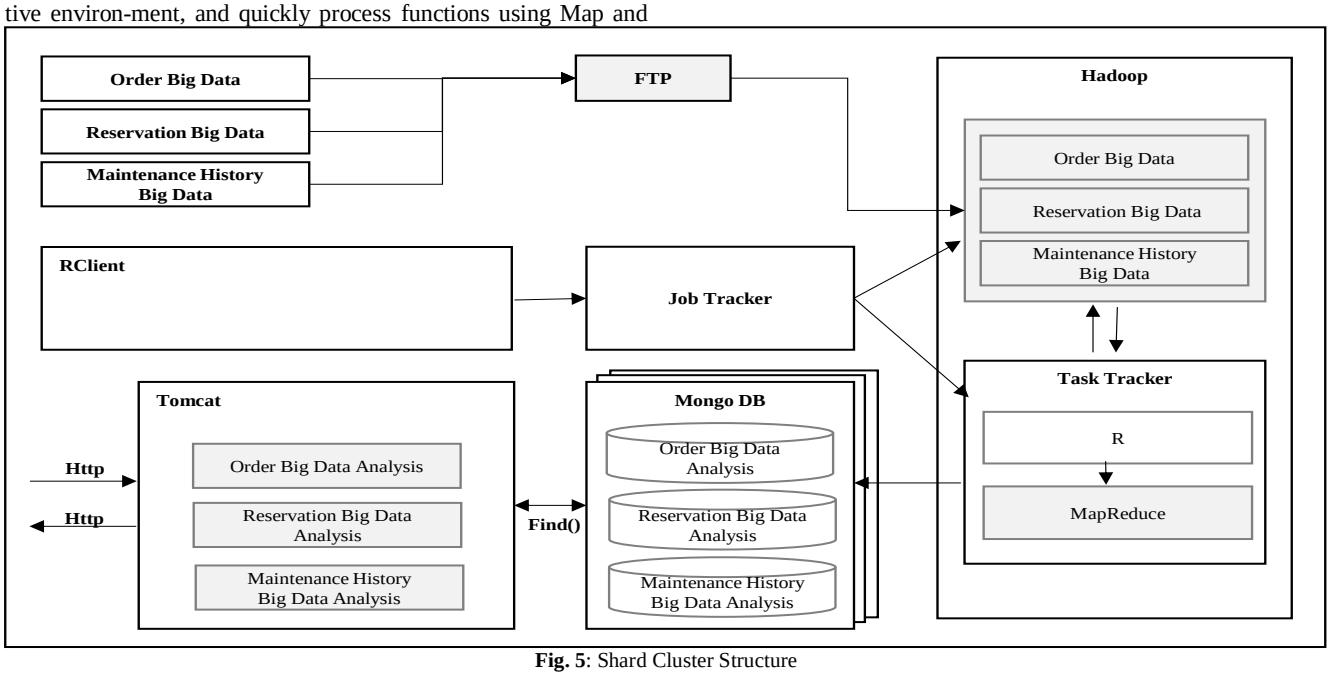

Fig. 5: Shard Cluster Structure

References

[1] Doo-sun Park, Yang-se Moon, Young-ho Park, Chan-hyun Yoon, Young-sik Jeong, Hyung-seok Chang , " big data computing technology", hanbitacademy, 2014

[2] Jung-jaehwa, "Beginning Hadoop Programming", wikibooks, 2012

[3] Ju-Jongmyeon, "NoSQL & mongoDB", DB, 2014

[4] Hadoop, https://www.thinkbiganalytics.com/leading_big_data_technologies/hadoop

[5] MongoDB, http://docs.mongodb.com/manual/

[6] Noh Kyoo-sung, Lee Doo-sik, Bigdata Platform Design and Implementation Model, Indian Journal of Science & Technology, 2015, 8(18), pp. 1-8.

[7] L. Greeshma,G. Pradeepini, Big Data Analytics with Apache Hadoop MapReduce Framework, Indian Journal of Science & Technology, 2016, 9(26), pp. 1-5.

[8] T. Y. J. Naga Malleswari, G. Vadivu, MapReduce: A Technical Review, Indian Journal of Science & Technology, 2016, 9(1), pp. 1-6.

[9] Munaza Ramzan, Farha Ramzan , Sanjeev Thakur, A Systematic Review of Type-2 Diabetes by Hadoop/Map-Reduce, Indian Journal of Science & Technology, 2016, 9(32), pp. 1-6.

[10] Arunkumar Thangavelu, N. Manoharan, Design and Analysis of an Effective Channel Distribution Approach for Agricultural Commodities using MongoDB, Indian Journal of Science & Technology, 2016, 9(47), pp. 1-10.

[11] P. Parthiban, S. Selvakuma, Big Data Architecture for Capturing, Storing, Analyzing and Visualizing of Web Server Logs, Indian Journal of Science & Technology, 2016, 9(4), pp. 1-9.

[12] Young-Woon Kim, Hyeopgeon Lee, Implementation of Big Data Analysis System to Prevent Illegal Sales in Cable TV Industry, Journal of Engineering and Applied Sciences, 2017, 12(23), pp. 6542-6545

[13] Sungwook Lee, "Real MongoDB", wikibooks, 2018

[14] Vignesh Prajapati, "Big Data Analytics with R and Hadoop", Packt Publishing Ltd., 2013