The Performance of MapReduce Over the Varying Nature of Data

Azhar Rauf[1], Adnan Amin[2], Saeed Mahfooz[1], Shah Khusro[1]

[1.] Department of Computer Science, University of Peshawar, Peshawar, KP, Pakistan
[2.] Institute of Management Sciences, Phase VII, Hayatabad, Peshawar, KP, Pakistan
Phone: +92-91-9216732
E-mail: azhar.rauf@upesh.edu.pk

Abstract: This paper is about the performance of mapper, shuffle, and reduce operations on varying nature of data using different types of Hadoop's clusters. Datasets, without duplicated and partially duplicated records were taken on single and two nodes Hadoop clusters. Experiments prove that mapper, shuffle, and reduce operations outperform on columns having more unique values as compared to columns having duplicated values. Experiments further prove that reduce operation takes the least time followed by shuffle and then mapper on single as well as on two nodes Hadoop clusters. Results prove that primary key columns or columns having more unique values are good potential attributes for efficient MapReduce operation.
[Azhar Rauf, Adnan Amin, Saeed Mahfooz, Shah Khusro. **The Performance of MapReduce Over the Varying Nature of Data.** *Life Sci J* 2013;10(4):1263-1266]. (ISSN:1097-8135). http://www.lifesciencesite.com. 166

Keywords: Big data, Hadoop, MapReduce Performance

1. Introduction

Big data is a buzz word in today's IT community as this era has already arrived. The digital revolution has facilitated almost every aspect of our lives but has introduced brand new challenges. One of them has been the challenge of big data which is at the same time a big opportunity as well. Electronic devices including wireless sensors, smart phones, computers, telephone switches, and other smart devices are producing a huge amount of data on the daily basis. Database management systems are primarily used to store the data. However, because of the exponential growth in the amount of data, these systems are not capable of storing such a huge amount data. According to an estimate 2.5 quintillion of data is generated on the daily basis. The existing database tools and technology were not designed to cope with as much data and here the challenge of big data comes in.

Big data is not only big in nature but big in dimensions. It consists of structured, semi structured, unstructured, and multimedia data. The V3 characteristics of big data are volume, variety, and velocity (deRoos, Eaton et al. 2012). Traditional databases systems store the data in the schema dependent form. It has been a great feature of relational databases which is based on pure mathematical foundation proposed by E.F. Codd in 1970s. Existing database tools and techniques including physical database design, query optimization, ETL, data warehouse, data mining, and business intelligence are circling around the schema dependent form of the data. This technology is applicable only when the data is transformed into a structured form. On the other hand, the schema dependency feature of databases confines the data inside the limits of schema. Data is free in nature and should be allowed to grow in its natural form. The revolution of social media data is a proof of this fact. Social media websites and web search engines produce as much data that cannot be handled by the existing databases systems. World's 80% of data is stored in unstructured form (deRoos, Eaton et al. 2012) and its analysis is a challenge for today's database community.

In recent years, Apache's Hadoop(Dean and Ghemawat 2008) has evolved has a de-fecto model for big data. Its architecture is designed as per the free nature of data. Data in Hadoop is stored beyond the limits of schema and database management systems. This research is focused on the efficiency of MapReduce based on varying nature of data for different types of Hadoop clusters.

2. Architecture of Hadoop

Hadoop is named after its creator Doug Cutting son's elephant toy (deRoos, Eaton et al. 2012). It was inspired after Google's distributed File System (GFS). A task in GFS is broken down into two steps: *mapper* and *reducer* in order to process the task in parallel mode which is spread across a cluster of nodes. Apache's Hadoop has employed the same concept on large datasets with the aim of bringing function-to-data model instead of the conventional data-to-function model (deRoos, Eaton et al. 2012).

Hadoop's architecture is mainly divided into two parts: Hadoop Distributed File System (HDFS) and the programming paradigm (MapReduce). Hadoop redundantly stores the data across huge inexpensive cluster allowing a node to fail and automatically

reprocess the unprocessed data (deRoos, Eaton et al. 2012). This adds up the great features of scalability and fault tolerance to Hadoop. HDFS is used to store data which is divided into smaller pieces (blocks) of 64MB size, spread across multiple clusters. This allows Hadoop to be scalable across thousands of nodes in a cluster. An individual file in Hadoop is divided into small blocks whose replicas are stored on multiple nodes which offer inherent features of fault tolerance and availability to Hadoop. Such non-sequential smaller blocks can be executed more optimally and increases the efficiency of Hadoop (deRoos, Eaton et al. 2012).

Hadoop has a number of components and the most popular components are discussed below:

MapReduce: MapReduce is a software framework that serves as the compute layer of Hadoop. MapReduce jobs are divided into two parts. The "Map" function divides a query into multiple parts and processes data at the node level. The "Reduce" function aggregates the results of the "Map" function to determine the "answer" of the query.

Pig: It is a high level programming language for Hadoop computations. One strong feature of Pig is that it can support high parallelism and can handle very large datasets. It consists of a compiler that produces sequences of MapReduce program.

Hive: A Hadoop-based data warehouse developed by Facebook. Is has the SQL kind environment called HiveQL where users can write typical SQL queries which are then converted to MapReduce. This helps users to write SQL queries without experience of MapReduce that can be integrated with BI and visualization tools for example, MicroStrategy and Tableau.

HBase: It is the Hadoop Database. It is a non-relational distributed, column oriented database which can scale up to billion of rows. HBase uses HDFS for the underlying storage.

Sqoop: It is the connectivity tool of Hadoop designed for bulk data load efficiently from

non-Hadoop data stores – such as relational databases and data warehouses – into Hadoop. It allows users to move data from Oracle, Teradata or other relational databases to the target.

Flume: It is used for collection and importing of log and event data into Hadoop. It is most often used as a log aggregator. Flume collects log data from many diverse sources, for example, web servers, application servers, and mobile devices then integrates and moves them to Hadoop.

Mahout: Mahout is a data mining library of Hadoop. It takes the most popular data mining algorithms for performing clustering, regression testing, statistical modeling and implements them using the Map Reduce model.

3. Related Work

Opportunities created by the petabyte world are discussed in (Schlieski and Johnson 2012). Authors suggest that new roles for the relationships with data need to be comprehended. One way is to create stories which themselves can become adaptive algorithms that can create a far engaging future in entertainment business.

Experiments are performed on different virtual systems with Hadoop in (Yang, Xiang et al. 2013). These Experiments show that Xen performs better than other virtual machines both on performance and stability. Moreover, better performance can be achieved with more virtual machines and adequate memory configuration of virtual machines. Results show that *get* is much quicker as compared to *put* operation. Their results prove that operating system virtualization is not a good choice for Hadoop because of memory problems. The proper configuration of the MapReduce computing optimizes and greatly improves the performance.

In (Guanghui, Feng et al. 2012), the advantages of Hadoop installation on virtual environment are discussed that include full utilization of computing resources, reliability, and saving of power. But it has a disadvantage of lower performance on virtual environment.

The problem of privacy in big data specially in social media is discussed in (Smith, Szongott et al. 2012). They focus on analysis of the threat to an individual's privacy that is created by other peoples' social media. Almost all the social media and Big Data research work is being utilized to create and make analysis on our profiles, for example, for market research, targeted advertisement, workflow improvement or national security. These are controversial issues because it is entirely up to the controller of the Big Data sets if the information gathered is used for good or bad purposes. In the context of the social media, there is an increasing awareness of the value, and its potential risk of the personal information which we voluntarily upload to the social media and websites.

Reference (Bakshi 2012) focuses on the infrastructure including processing, network, and storage systems of Hadoop and reviews design criteria and implementation considerations. They focus on performance considerations and describe relevant benchmarks with a Hadoop analytics cluster.

This paper is focused on the performance of mapper, shuffle, and reduce operations over the varying nature of data.

4. Material and Methods

This research work is focused on comparing the three well known operations of Hadoop i.e., map,

shuffle, and reduce for different datasets. The objective of research is to study the performance variation of the three operations over different nodes of a cluster for different nature of data. Different datasets are taken having non-duplicated and partially duplicated records spread randomly in the dataset to study the processing nature of Hadoop.

Experiments were conducted on a single node and two nodes Hadoop cluster respectively over Intel® Core™2 Duo 2.1 GHz CPU with 4GM RAM running on 32 bit Oracle virtual Box version 4.2.6. The underlying operating system was Ubuntu version 12.10 Quantal Quetzal. A CSV file of one million rows was imported to Hadoop version 1.0.4. Hive version 0.9.0 environment was selected to test the parallel processing capabilities of Hadoop. Initially there were no duplicate records in the dataset.

The following kind of query was run in Hive:

```
SELECT [aggregate_function]
FROM [relation]
GROUP BY [column_name];
```

Tests were run for 25, 50, and 75 percent duplicate rows out of one million rows dataset. Each experiment was run ten times and average values were calculated which are given in table 1.

Table 1 shows that Hadoop's single node cluster with data having no duplicate records has the highest performance. It further shows that for any type data, reducer takes the least time followed by shuffle and then the mapper operation.

TABLE 1: SINGLE NODE CLUSTER

Average Time in Seconds			
DataSet Types	Mapper	Shuffle	Reducer
Type -0% (No duplication)	29.3	22.6	20.9
Type-25% (25% duplication)	62.3	51.3	33.1
Type-50% (50% duplication)	68.3	48.3	24.6
Type-75% (75% duplication)	71.7	55.7	27.7

The same experiment was repeated for two nodes cluster on same host computer using the same dataset. Results are shown in table 2.

Table 2 shows that the two nodes cluster having no duplicate records in the data has performed efficiently as compared to the duplicated data. Results show that Reducer operation takes the least time followed by shuffle and then mapper operation.

TABLE 2: TWO NODES CLUSTER

Average Time in Seconds			
DataSet Types	Mapper	Shuffle	Reducer
Type -0% (No duplication)	32	18.5	12.5
Type-25% (25% duplication)	32.7	28	18.4
Type-50% (50% duplication)	64.2	50.5	24.2
Type-75% (75% duplication)	55.3	32.7	14.8

5. Results

The following figures 1 and 2 show the performance of Mapper, Shuffle, and Reduce phases on single node and two nodes clusters respectively.

Figures 1 and 2 show that reducer has taken the least time in both single and two nodes clusters. Mapper takes the largest time in both cases.

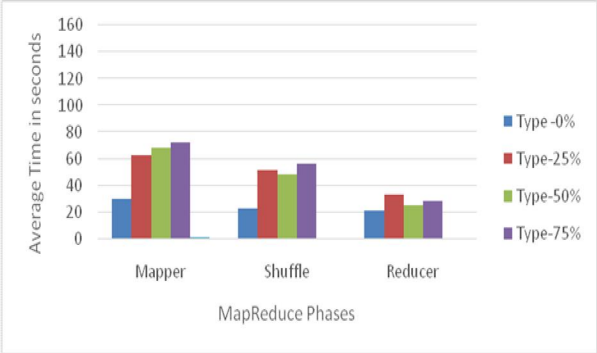

Figure 1: MapReduce phases on Single Node Cluster

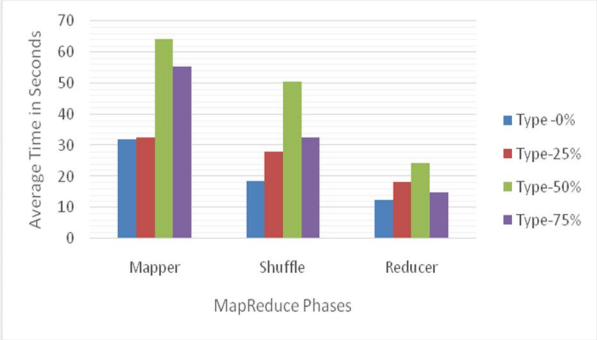

Figure 2: MapReduce phases on Two Nodes Cluster

Consider figures 3 and 4 that show the performance of Mapper, Shuffle, and Reduce on varying nature of data for a single and two nodes clusters respectively. Both diagrams prove that data with no duplication or having distinct values show highest performance than data having duplicate values. This shows that primary key columns or columns having more unique values are good potential attributes for the MapReduce operations.

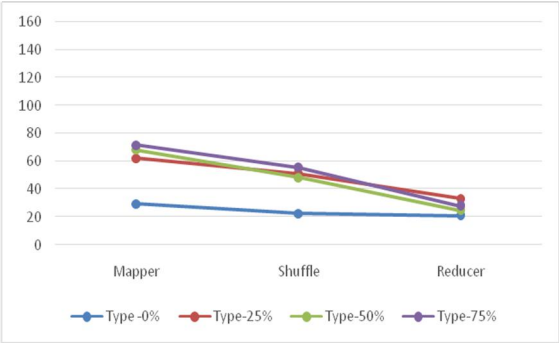

Figure 3: Performance on Single Node Cluster

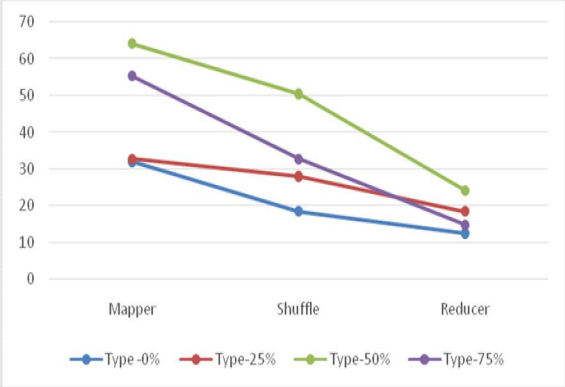

Figure 4: Performance on Two Node Cluster

6. Discussions

This paper discusses the performance of Hadoop's MapReduce paradigm for the varying nature of data having no duplicate and densely duplicated data. Hadoop's single and double nodes clusters using virtualized environment was used for the experiment purposes. Results prove that the reduce process takes the least time followed by shuffle and then the map process for any nature of data i.e., with no duplicate and densely duplicated records. The MapReduce operations show good performance for non-duplicated data as compared to duplicated data in cases of single and double nodes clusters.

References

1. Bakshi, K. (2012). Considerations for big data: Architecture and approach. Aerospace Conference, 2012 IEEE.
2. Dean, J. and S. Ghemawat (2008). "MapReduce: simplified data processing on large clusters." Communications of the ACM **51**(1): 107-113.
3. deRoos, D., C. Eaton, et al. (2012). Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data.
4. Guanghui, X., X. Feng, et al. (2012). Deploying and researching Hadoop in virtual machines. Automation and Logistics (ICAL), 2012 IEEE International Conference on.
5. Schlieski, T. and B. D. Johnson (2012). "Entertainment in the Age of Big Data." Proceedings of the IEEE **100** (Special Centennial Issue): 1404-1408.
6. Smith, M., C. Szongott, et al. (2012). Big data privacy issues in public social media. Digital Ecosystems Technologies (DEST), 2012 6th IEEE International Conference on.
7. Yang, Y., L. Xiang, et al. (2013). Impacts of Virtualization Technologies on Hadoop. Intelligent System Design and Engineering Applications (ISDEA), 2013 Third International Conference on.

10/22/2013