TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 15, No. 1, July 2015, pp. 171 ~ 176
DOI: 10.11591/telkomnika.v15i1.8080

Effective and Efficient Way of Reduce Dependency on Dataset with the Help of Mapreduce on Big Data

Satish Londhe*[1], Smita Mahajan[2]
Research Guide, Symbiosis International University, Pune, 412115, India
*Corresponding author, e-mail: Satish.Londhe@sitpune.edu.in[1], smita.Mahajan@sitpune.edu.in[2]

Abstract

With the fast development of networks these days organizations has overflowing with the collection of millions of data with big number of combination. This big data challenges over trade troubles. It requires more analysis for the high-performance procedure. The new method of hadoop and MapReduce methods are discussed starting the data mining standpoint. In the proposed research work we have to progress performance through parallelization of different operations such as loading the information, index building and evaluating the queries. Thus the performance analysis is completed with the minimum of three nodes with in the Amazon cloud environment. Hbase is a open source, non-relational and distributed database model. It executes on the pinnacle of Hadoop. It consists of a single key with multiple values. Looping is avoid in retrieving a meticulous data from huge datasets and it consume less amount of time for execute the data. HDFS file system is used to store the data after performing arts the map reduces operations and the execution time is decreased when the amount of nodes gets increased. The performance analysis is tuned with the parameters such as the carrying out complexity.

Keywords: *mapreduce, data mining, big data, hadoop*

1. Introduction

Data mining [4, 5] is designed to request the data by concerning systematic relationships involving the variables and then by applying the detected pattern to the new separation of data, then findings can be validate. The machinery of data mining is to relate standard algorithms to the large sets of data to take out the information commencing it. This large set of data that is connected to business or marketplace is known as “Big Data [1]”. The theme of business intelligence (BI) is to switch the data that tends increased value to the venture. Rather than collect the information on what organization are really doing, it is enhanced to understand how organizations view big data and to what extent they are presently using it to benefit their business. Now organization are begun to understand and discover how to process and analyze this big data. Big data and data mining are dispersal not only in BI but also in science fields such as meteorology, fuel exploration and bio-informatics. This series of data needs hold up of software, hardware and primitive algorithms for the new level of investigation.

2. Background

2.1. Hadoop and MapReduce

Big data [2] plays an imperative role in Business applications. The coverage of big data includes Data gaining, cleaning, allocation, and best practices. Big Data contain the risk of threat analysis, predict failures of the network data and deal control. Big data analytics found that Apache Hadoop is favored as solution to the tribulations in the conventional Data Mining. It acts as extensible for getting better the failures of the data storage and dispensation in the distributed system.Apache Hadoop is an Open-source software framework [13] for storage and processing of big data-sets on clusters of the hardware. Here the hadoop is designed with the assumption of hardware failures that are automatically handled by software framework. The major components of Hadoop are Hadoop distributed file system **(HDFS)** which is practical for large files and MapReduce which acts as heart of Hadoop. HDFS is high bandwidth cluster storage space. MapReduce performs two dissimilar tasks in Hadoop program. the First job is to map, in which it takes a compilation of data where it is distorted into another set of information.

Received April 3, 2014; Revised May 23, 2015; Accepted June 4, 2015

After transformation the data is busted in to tuples (with key/value pairs).The job of decrease is to take the output from the map job which acts as its contribution and these data tuples are combined into lesser sets of tuples. In this manner the decrease job is for all time achieve after the map job.

2.2. Using Hadoop Techniques with Parallel Databases

In the previous days of Hadoop and MapReduce there are numerous problems. But now the present versions have been used with different data management technique to decrease the concert gap. In this context the Hadoop is studied in the sense of similarity and differences with parallel database. The parallel databases techniques like job optimization, data layouts and indexes will focus in this conversation. Even then the convention of Hadoop requires a diminutive knowledge of databases background; it can't compete with the equivalent databases with their efficiency of query dispensation. Many researchers found that it is resourceful to use equivalent databases with the grouping of Hadoop MapReduce. In the preceding versions of Hadoop, most of the tribulations are found in the substantial organization of data counting data layouts and indexes. In common Hadoop and MapReduce affected starting row-oriented layout. Hence other data layout technique are projected for Hadoop MapReduce correspondingly Hadoop has deficit of appropriate indexes.A high-quality foundation of indexing techniques has been anticipated for Hadoop MapReduce [15-18].

2.3. Hadoop and Data Warehouse

As the publicity of Hadoop is unrestrained, the practitioners are easily affected by diversity of opinions like Hadoop is flattering the new data stockroom. But it is not really as it seem. There are a lot of differences involving Hadoop with data warehouse. This context explores when to bring into play hadoop and when to switch data store. Let us think an example of compact uses Hadoop to preprocess raw click stream generated by clientele using their website. These click stream are passed to the data warehouse as its dispensation provides the vision of customers preferences. The data stockroom sets these customer preferences with marketing campaigns and recommendation engines to offer speculation suggestions and psychoanalysis to customers. So the Data Warehouse is used as a foundation in complex Hadoop job. This will bring the advantages of given two systems in equivalent. Finally choosing hadoop and data warehouse depends on the necessities of the association. In most of the cases Hadoop and data storehouse work mutually as a group in the in sequence supply.

3. Proposed System Framework

On top of the DFS, a lot of different higher-level programming frameworks have been urbanized. The most commonly implemented encoding framework is the MapReduce framework [4], [11-12]. MapReduce is a budding programming framework for data-intensive application proposed by Google. MapReduce borrows ideas from purposeful programming [12], where the programmer define Map and Reduce errands to process huge sets of scattered data. Implementations of MapReduce [11] facilitate many of the most widespread calculations on important data to be performed on computing clusters professionally and in a way that is broadminded of hardware failures during calculation. However MapReduce is not appropriate for online transactions [11, 12]. The key strengths of the MapReduce encoding framework are the high degree of parallelism combined with the unfussiness of the indoctrination framework and its applicability to a large assortment of application domains [4, 11]. This require dividing the workload crossways a large number of equipment. The amount of parallelism depends on the input data size. The map function processes the input pairs (key1, value1) returning some other intermediary pairs (key2, value2). Then the intermediary pairs are grouped together according to their key. The reduce function will output some new key-value pairs of the form (key3, value3). Figure 1 shows an example of a MapReduce algorithm used to count words in a file. In this example the map input key is the provided data chunk with a value of 1. The map output key is the word itself and the value is 1 every time the word exists in the processed data chunk. The reducers perform the aggregation of the key-values pair output from the maps and output a single value for every key, which in this case is a count for every word. Figure 1 provides further explanation of the generation of the key-value pairs produced during the processing phases of the Word Count MapReduce program.

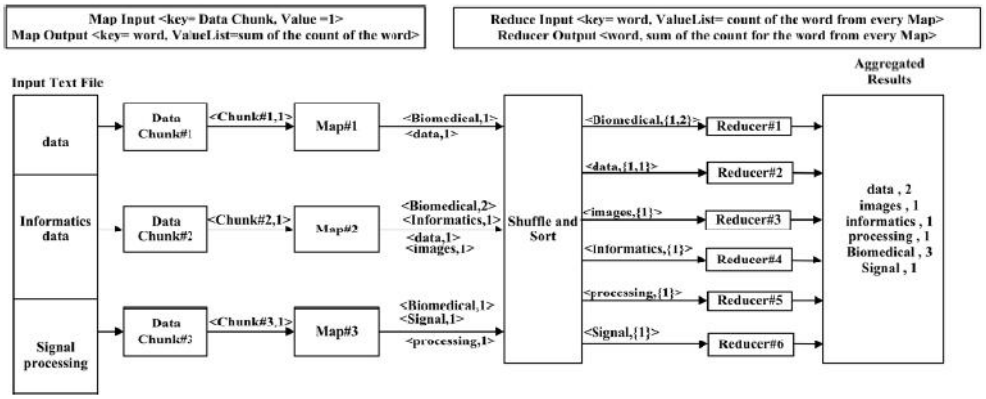

Figure 1. System block diagram

High presentation is achieve by breaking the dispensation into small units of work that can be run in parallel across potentially hundreds or thousands of nodes in the cluster. Programs written in this functional style are automatically parallelized and executed on a large cluster of commodity machines. This allows programmers without any experience with parallel and distributed systems to easily utilize the resources of a large distributed system [3, 4].

3.1. Secrecy View

Definition 1: (Attribute separation and Columns) In attribute separation, D (database) consists of several subsets, such that each attribute belongs to exactly one subset. Each subset of attributes is called a column. Specifically, let there be C columns C1; C2; . . . Cc, then U(c)i=1,C=D; and for any $1 \leq i_1 \neq i2 \leq c$, $Ci_1 \cap Ci2 = \Theta$. For simplicity of discussion, we consider only one sensitive attribute S. If the data contain multiple sensitive attributes, one can either consider them separately or consider their joint distribution. Exactly one of the c columns contains S. Without loss of generality, let the column that contains S be the last column C. This column is also called the sensitive column. All other columns {C1, C2…… Cc-1} contain only QI attributes.

Definition 2: (Tuple Partition and Buckets). In tuple partition, T consist of several subsets, such that each tuple belongs to exactly one subset. This tuples subset is called a bucket. Specifically, let there be b buckets. B1, B2……Bb then $U^{b}{}_{i-1}B_i=T$ and for any $1 \leq i_1 \neq i2 \leq b$, $Bi_1 \cap Bi2 = \Theta$.

Definition 3 (Slicing): Given a microdata table T, a slicing of T is given by an attribute partition and a tuple partition. For example, suppose tables a and b are two sliced tables. In Table a, the attribute partition is {{Age}, {Gender}, {Zipcode}, {Disease}} and the tuple partition is {{t1; t2; t3; t4}, {t5; t6; t7; t8}}. In Table b, the attribute partition is {{Age, Gender}, {Zipcode, Disease}} and the tuple partition is {{t1; t2; t3; t4}, {t5; t6; t7; t8}}.

Definition 4 (Column Generalization): Given a microdata table T and a column Ci = (Xi1,Xi2,Xi3…….Xij) where Xi1,Xi2…..Xij are attributes, a column generalization for Ci is defined as a set of non overlapping j-dimensional regions that completely cover D[Xi1]* [Xi2] * ….. D[Xij]. A column generalization maps each value of Ci to the region in which the value is contained.

Column generalization ensures that one column satisfies the k-anonymity requirement. It is a multidimensional encoding and can be used as an additional step in slicing. Specifically, a general slicing algorithm consists of the following three phases: attribute partition, column generalization, and tuple partition. Because each column contains much fewer attributes than the whole table, attribute partition enables slicing to handle high-dimensional data. A key notion of slicing is that of matching buckets.

4. Challenges

Health care systems in general suffer unsustainable costs and lack data utilization. Therefore there is a pressing need to find solutions that can reduce unnecessary costs.

Advances in health quality outcomes and cost control measures depend on using the power of large integrated databases to underline patterns and insights. However, there is much less certainty on how this clinical data should be collected, maintained, disclosed, and used. The problem in health care systems is not the lack of data, it is the lack of information that can be utilized to support critical decision-making. This presents the following challenges to big data solutions in clinical facilities:

1) Technology straggling. Health care is resistant to redesigning processes and approving technology that influences the health care system.

2) Data dispersion. Clinical data is generated from many sources (e.g. providers, labs, data vendors, financial, regulations, etc.) this motivates the need for data integration and maintaining mechanism to hold the data into a flexible data warehouse.

3) Security concerns and privacy issues. There are lots of benefits from sharing clinical big data between researchers and scholars, however these benefits are constricted due to the privacy issues and laws that regulate clinical data privacy and access.

4) Standards and regulations. Big data solution architectures have to be flexible and adoptable to manage the variety of dispersed sources and the growth of standards and regulations (e.g. new encryption standards that may require system architecture modifications) that are used to interchange and maintain data.

4.1. Hadoop MapReduce Advantages

The main advantage of Hadoop MapReduce it allows the users (even though if they are not experts) to easily handle analytical risk over Big data. It gives complete control on processing the input datasets. MapReduce can be easily used by the developers without having much knowledge of databases but with a little knowledge of java is needed. It gives satisfied performance in scaling large clusters.

a) It supports distributed data and computation
b) The computation is performed local to data and thus it prevents the network overload.
c) The tasks are independent hence, it can easily handle partial failures such as when the nodes fail, and it can automatically restart.
d) It is a Simple programming model. The end-user programmer only writes MapReduce tasks.
e) HDFS stores vast amount of information.
f) HDFS is simple and robust coherence model thus it stores data reliably.
g) HDFS provide streaming read performance.
h) Flat scalability [20]
i) It has the ability to process the large amount of data in parallel.
j) HDFS has capability for replicating the files which can easily handle situations like software and hardware failure.
k) In HDFS the data can be written only once and it can be read for many times.
l) It is more economic way as the data and processing are distributed across the clusters of personal computers.
m) It can be offered as on-demand service, for example as part of Amazon's EC2 cluster computing service.
n) Ability to write MapReduce programs in Java, a language which even many noncomputer scientists can learn with sufficient capability to meet powerful data-processing needs.

4.2. Disadvantages or Limitations of Hadoop

These are major common areas where the Hadoop framework is found uncertain.

a) As the both the Hadoop HDFS and MapReduce software are under active development, they are found to be uneven.
b) Possibility of preventing central data leads to restrictive programming model.
c) HDFS is weak in handling small files, and inadequacy of transparent compression. The design of HDFS is such that it doesn't work with random reads on small files because of its optimization for sustained throughput.
d) There is a necessary of managing job flow is when there is intermediate data.

e) Managing the cluster is hard in operations like debugging, distributing software, collection logs etc.
f) Because of single-master model, it requires more care and may limit scaling.
g) Hadoop offers high security model, but because of its complexity it is hard to implement it.
h) MapReduce is a batch-based architecture which means it doesn't allow itself to use cases that needs real-time data access.

5. Results and Discussion

We have done setting up the Hadoop 2.0 cluster nodes. Client-Server communication between Hadoop clients and servers (e.g., the HDFS client to NameNode protocol, or the YARN client to Resource Manager Protocol). System first generates a high dimension health care data and load on database with end user GUI. The different SQL queries show the execution time as well algorithm complexity.

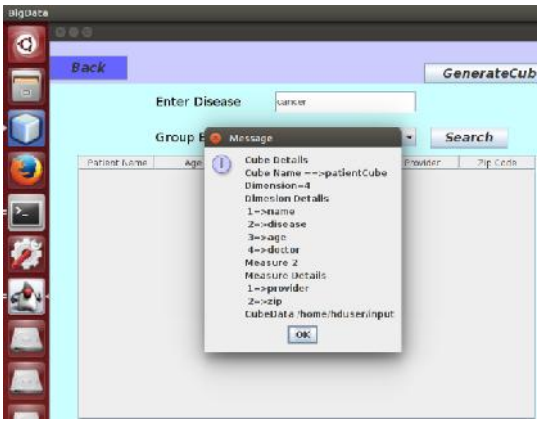

Figure 1. Cube generate

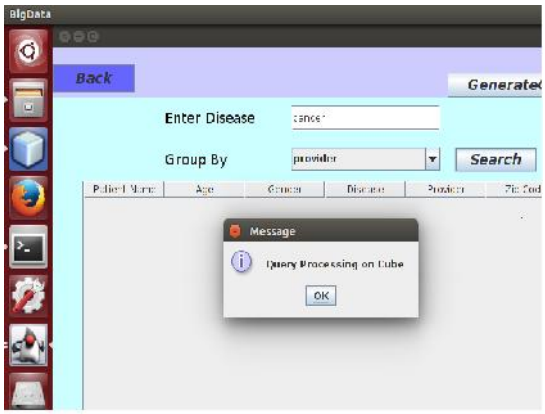

Figure 2. Cube query

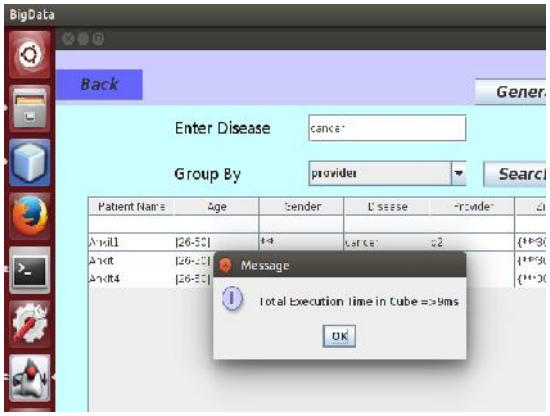

Figure 3. Cube time

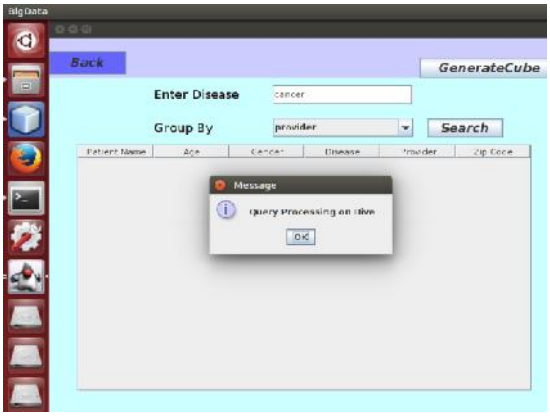

Figure 4. Hive query

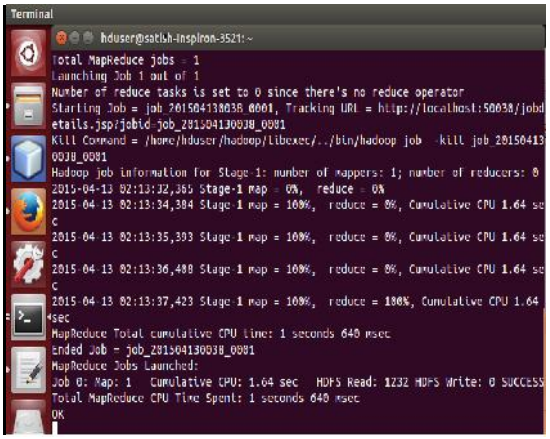

Figure 5. Hive time

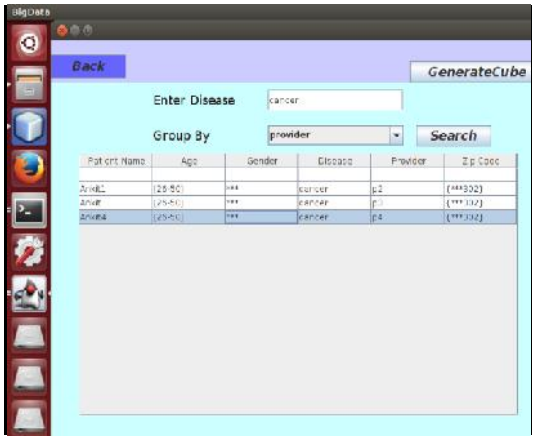

Figure 6. Search Result

6. Conclusion

An integrated solution eliminates the need to move data into and out of the storage system while parallelizing the computation, a problem that is becoming more important due to increasing numbers of sensors and resulting data. And, thus, efficient processing of clinical data is a vital step towards multivariate analysis of the data in order to develop a better understanding of a patient clinical status (i.e. descriptive and predictive analysis). This highly demonstrates the significance of using the MapReduce programming model on top of the Hadoop distributed processing platform to process the large volume of clinical data.To keep track of current state of business, advanced analytical technique of big data such as predictive analysis, data mining, statistics and natural language processing are to be examined. New techniques of big data such as **Hadoop** and **MapReduce** create alternatives to traditional data warehousing. Traditional Hadoop with combination of new technologies explores a new scope of study in various fields of science and technologies.

References

[1] O'Reilly Media. Disruptive Possibilities: How Big Data Changes Everything. 2013.
[2] McKinsey Global Institute. Big Data:The next frontier for innovation, competition, and productivity.
[3] Coulouris GF, Dollimore J, Kindberg T. Distributed Systems: Concepts and Design. Pearson Education. 2005.
[4] de Oliveira, Branco M. Distributed Data Management for Large Scale Applications. Southampton, United Kingdom: University of Southampton. 2009.
[5] Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R. Advances in knowledge discovery & data mining. Cambridge, MA: MIT Press. 1996.
[6] Han J, Kamber M. Data mining: Concepts and Techniques. New York: Morgan-Kaufman. 2000.
[7] Hastie T, Tibshirani R, Friedman JH. The elements of statistical learning : Data mining, inference, and prediction. New York: Springer. 2001.
[8] Pregibon D. Data Mining. Statistical Computing and Graphics. 1997: 7-8.
[9] Weiss SM, Indurkhya N. Predictive data mining: A practical guide. New York: Morgan-Kaufman. 1997.
[10] Westphal C, Blaxton T. Data mining solutions. New York: Wiley. 1998.
[11] Dean J, Ghemawat S. *MapReduce: simplified data processing on large clusters*. Commun ACM 2008. 2008; 51(1): 107-113.
[12] Peyton Jones SL. The Implementation of Functional Programming Languages (Prentice-Hall International Series in Computer Science). New Jersey, USA: Prentice-Hall, Inc. 1987.
[13] http://en.wikipedia.org/wiki/Apache_Hadoop.Apache .
[14] https://infosys.uni-saarland.de/publications/BigDataTutorial.pdf.
[15] Shvachko K, Kuang H, Radia S, Chansler R. *The hadoop distributed file system*. In Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on: 2010, IEEE. 2010: 1-10.
[16] The Apache Software Foundation. http://apache.org/.
[17] Olson M. *Hadoop: Scalable, flexible data storage and analysis*. IQT Quart 2010. 2010; 1(3): 14-18.
[18] Xiaojing J. *Google Cloud Computing Platform Technology Architecture and the Impact of Its Cost*. In 2010 Second WRI World Congress on Software Engineering 2010. 2010: 17-20.