International Journal of Research
Available at https://pen2print.org/index.php/ijr/

e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 20
September 2018

Applications of MapReduce in big data concepts

M.Akhila
MTech
CSE
Anurag group of institutions
Hyderabad

V.Amarnath
Assistant Professor
MTech, Computer Science
Anurag group of institutions
Hyderabad

Dr G.Vishnu Murthy
HOD & Professor
Dept CSE
Anurag group of institutions
Hyderabad

Abstract:

Big data analytics is challenging for storage and creating a cluster in Hadoop and using map reduce technique analysis of large data sets. Due to increase huge volume of data, processing and availability of data, the size of storage data is increasing in zeta-bytes. Hadoop is one of the technologies in the real world for analysing the data through Hadoop Distributed File System MapReduce component for processing the data concurrently to improving throughput. In this paper discussed how the map reducer algorithms is implemented in various clustering algorithms of big data

Keywords: Big data, MapReduce, Algorithms

1. Introduction

Big data is the term for collection of datasets so large and complex that it becomes difficult to process using on-hand database system tools for data processing applications. But it's not the amount of data that's important. Its what organizations do with the data that matters. Big data can be analysed for insights that lead to better decisions and strategic business moves.

Characteristics of bigdata:

1. Volume:

This is the aspect that comes to most people's minds when they think of Big Data Volumes of data have increased exponentially in recent times. It is not uncommon for business to deal with petabytes of data, and typically analysis is performed over the entire data set, not just a sample.

2.Variety:

Big data is not always structured data and it is not always easy to put big data into a relational database. Big data includes data types such as videos, music files, emails, unstructured word documents and social media feeds. Dealing with a variety of structured and unstructured data increases the complexity of both storing and analysing Big Data.

3. Velocity:

Big Data is not just about the volume though. Just as important is the rate of changes of the data. For a large volume of data which doesn't change very often, analysis that takes a number of hours or days to complete may be acceptable, but if the dataset is growing by terra bytes per day, or the data is changing at a high rate of speed, the processing time of analysis becomes much more important.

4.Value:

This is the most important aspect of big data. It costs a lot of money to implement IT infrastructure systems to store big data, and business are going to require a return on investment. At the end of the day, if you can't extract value from your data, there is no point in building the capability to store and manage it.

5.Veracity:

When we are dealing with a high volume, velocity and variety of data, it is inevitable that all of the data need to be completely correct there will be wrong data. Often the data does not need to be perfect but does not be close enough to gain relevant insight. Dependent on the application, the veracity, or verification of the data may be essential.

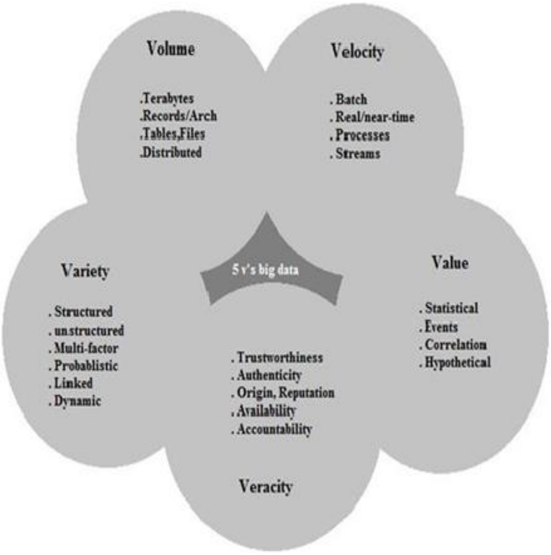

Fig 1. Parameters of big data

2. MapReduce Framework

In the current realities, it is a major concern about handling large volume of data for researchers. Many Applications like data mining, Image Processing, data analytic etc are processing for large data. Google had developed a MapReduce framework suitable for processing parallel data in spread computing environment [1]. MapReduce is a programming model for processing huge data sets in Hadoop's distributed clusters. [2,3]. It processes the data to be split in to parallel distribution, synchronization and fault tolerance are handled automatically by the framework.

MapReduce has been divided in to two phases:

Map Phase

In the Map phase the execution is processed on a chunk of inputs which the data is independent and Processed in similarly.

Reduce Phase:

Inthereducephase, execution is processed on one or more key (list of values) pairs and the results has been aggregated to get thefinal resultsofeachpartitionarecombinedtogetthe finalresultsortheinputforanothermapreducejob. MapReducehasbeenutilizedtoprocessmassivevolumeofdataindifferentareasbecauseof itssimplicity, scalability andfault-tolerance [4-5].

In spite of being featured such as scalability in clusters, ensuring availability, handling failures Google's MapReducehas been unusable for certain kind of applications requires iterative computation, execution of high-level language such as SQL and work on an Internet desktop grid. Since the MapReduce introduced, numerous MapReduce frameworks have been developed by several companies including Google's MapReduce [6], Apache's Hadoop MapReduce [7], Disco [8] etc. MapReduce technologies have also been used by a increasing amount of groups in industry (e.g., Facebook [9], and Yahoo [10]), and there are several database vendors such as Greenplum [11] and Aster Data [12], who leverage concepts of MapReduce in their data warehousing solutions

Initially the paper describes the overview of MapReduce as well as algorithms used in MapReduce. The primary focus of this survey paper is to describe about the MapReduce clustering algorithms where to be used.

3. MapReduce Algorithms

MapReduce data diagnostic claims are considered on the basis of their functions:

A. Clustering Algorithms

In a clustering environment it uses so many clustering algorithms for MapReduce systems. It requires large amount of multiple clusters, a massive computation makes it compute intensive method. for eg, K-means, Fuzzy K-means, canopy clustering etc.

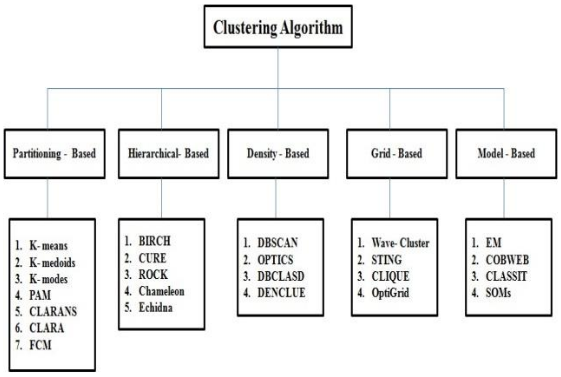

Fig 1. Clustering Algorithms

B. Classification Algorithms

This algorithm works on a training data and query set to calculate k near values which required to enough memory space to store the data [7]. It is also compute intensive method because a vector product is carried out to calculate the similarly between two vectors. For K-nearest neighbour etc.

1)Hash Join-It is a different of broadcast join by Banas et al[12].In the combine operation, only Map function is used to add two tables i.e. data table (S) and reference table (R) [8] .A hash join is not compute intensive application and its time complexity is O(|S|).

2)K-Means-K-means application is used to divide a set of n sample objects into K clusters for input parameter k. This algorithm is memory intensive and compute intensive which in turn limiting the number of cluster K-means can generate [9]. The time complexity O(|n|*|K|).

3)K-nearest neighbours-K-nearest is a classification algorithm that uses a large in-memory data set, KNN method uses two data sets, a query method is O(|Q|*|T|) because it calculates the distance between every point in Q and in T. So, the KNN is compute intensive as well as memory intensive applications [10].

5.Related Work

The most related work associated with the introduction of MapReduce Framework and its relation with the database processing [12]. This tutorial provides the insight about the algorithms used with MapReduce in big data.

6.Conclusion

MapReduce provides a distributed parallel computing across multiple nodes and return result on a particular node. MapReduce plays a important role in parallel data processing because of its salient features such as scalability, flexibility and fault tolerance. Previous Research showed that MapReduce framework is not sufficient to handle some specific kind of applications [11]. It raised a question regarding improvement and enhancement of the Map Reduce architecture to address those issues and challenges. In this survey paper, our focus was on the extended Map Reduce framework with additional algorithms to support some specific kind of tasks. Initially, we reviewed Google invented Map Reduce architecture and its various applications. Many organizations have invented various Map Reduce frameworks with additional features after Google's invention. We had compared the design and functionalities of frameworks with Apache Hadoop and Phoenix.

A lot of research work has been done on the extension of Map Reduce carried out with algorithms and mechanism to optimizing it for a new set of problems. We reviewed the extended version of MapReduce for more data intensive applications.

7.References

[1] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters" In ACM OSDI, 2004.

[2] Google spotlights data centre inner workings | etch news blog - CNET News.com (http://news.cnet.com/8301-10784 _3-9955184-7.html)

[3] MapReduce: Simplified Data Processing on Large Clusters(http://static.googleusercontent.com/media/research.google.com/es/us/archive/mapreduce-osdi04.pdf)

[4] K.-H.Lee,Y.-J.Lee,H.Choi,Y.D.Chung,B.Moon,ParalleldataprocessingwithMapReduce:asurvey,SIGMODRec.40(2012)11–20.

[5] M.Rionda to,J.A.DE Brabant,R.Fonseca,E.Uppal,PARMA: aparallelrandom-sizedalgorithmforapproximateassociationrulesmininginMapReduce,in:Proceedingsofthe21stACMinternationalconferenceonInformationandknowledgemanagement,Maui,Hawaii,USA,2012.

[6] J. Zhao, J. Pensive-Griotic, MapReduce: The Programming Model And Practice, 2009.

[7] D. Jiang, B. C. Aoi, L. Shi and S. Wu, The performance of MapReduce: An

in-depth study. Proc. VLDB Endow., 3 pp. 472–483 (Sept 2010),

[8] S. Blanas, J. M. Patel, V. Erceg vac, J. Rao, E. J. Shechita, and Y. Tian, "A Comparison of Join Algorithms for Log Processing in MapReduce," in Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, SIGMOD ‟10, (New York, NY, USA), pp. 975– 986, ACM, 2010.

[9] E. A Mohammed, B. H Far and C. Naugler, Application of the MapReduce programming framework to clinical big data analysis:Current landscape and future trends, BioData Mining. Vol. 7, 22. Oct 29, 2014.

[10] M Jonas, S Solangasenathirajan and D Hett. Annual Update in Intensive Care and Emergency Medicine 2014. New York – USA: Springer. Patient Identification, A Review of the Use of Biometrics in the ICU; pp. 679–688. (2014).

[11] E. Arslan, M. Shekhar & T. Kosar, "Locality and Network-aware reduce task scheduling for data intensive applications", published in Proceedings DataCloud‟14 Proceedings of the 5th International workshop on Data Intensive Computing in the Clouds, page 17-24, ISBN:978-1-4799-7034-6

[12] D. Gillick, A. Faria and J. DeNero, "Map Reduce: Distributed Computing and Machine Learning", Dec-2006 [15]Owen O‟Malley, "TeraByte Sort on Apache Hadoop", Yahoo! owen@yahoo-inc.com May 2008.