International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/IJSRCSEIT

Comparative Analysis of Map Reducing Algorithm and Coding Approach in Using Big Data Technique

Prof (Dr.) Rajender Kumar
Research Scholar, D. Sc Computer Science & Applications (Post-Doctoral Fellow), Desh Bhagat University, Punjab India

Article Info

Publication Issue :
Volume 8, Issue 6
November-December-2022

Page Number : 464-475

Article History
Accepted: 01 Dec 2022
Published: 12 Dec 2022

ABSTRACT

Big Data has fascinated a lot of concentration from academia, industry as well as government sectors. We are living in an era when a dangerous amount of data is being generated every day. Data is produced through many sources like commerce processes, transactions, social networking sites, web servers, etc. and leftovers in well thought-out as well as shapeless form. Processing and extracting a vast amount is data is a demanding task. Big Data refers to technologies and initiatives that grip data that is too diverse, fast- altering or enormous for conservative technologies, skills and infrastructure to address efficiently. Big Data size is continually moving target currently ranging from few dozen terabytes to many petabytes in a single data sets. Accurateness in big data may escort to more positive managerial. And better decisions can mean greater functioning efficiency, cost reductions and reduced hazard.

Big Data is the newest trend in business and IT world right now. Big Data is derived from the datasets involved are so bulky that distinctive database systems are not able to store and analyse the datasets. Big Data is a well-liked term used to describe the exponential growth and availability of data, both structured and unstructured. Big Data is defined as the depiction of the progress of human being cognitive processes, regularly includes data set with sizes further than the capability of present technology, mode and hypothesis to capture, manage, and process the data within adequate elapsed time. The terms imply Big Data have following properties

Keywords : Big Data, Data Analytics, Algorithm and Coding Approach.

I. INTRODUCTION

Big data comes and is composed through electronics operations from multiple sources.

It requires proper processing power and high capabilities for analysis [9].

The importance of big data lies in the analytical use which can help generate an informed decision to provide better and faster services [10].

The term big data is called on the huge amount of high-speed big data of different types; this data cannot be processed and stored in regular computers.

The main characteristics of big data, called V's 5 As in Figure 1 , can be summed up in the fact that the issue is not only about the volume of data, other dimensions of big data, known as 'five Vs', are as follows :

1.2.1. Volume

It represents the amount of data produced from multiple sources which show the huge data in numbers by zeta bytes.The volume is most evident dimension in what concerns to big data.

1.2.2. Variety

It represents data types, with, increasing the number of Internet users everywhere, smart phones and social networks users, the familiar form of data has changed from structured data in databases to unstructured data that includes a large number of formats such as images, audio and video clips, SMS, and GPS data [11]

1.2.3. Velocity

It represents the speed of data frequency from different sources, that is, the speed of data production such as Twitter and Face book.

The huge increase in data volume and their frequency dictates the need for a system that ensures super-speed data analysis.

1.2.4. Veracity

It represents the quality of the data, it shows the accuracy of the data and the confidence in the data content.

The quality of the data captured can vary greatly, which affects the accuracy of analysis.

Although there is wide agreement on the potential value of big data, the data is almost worthless if it is not accurate [12]

1.2.5. Value

It shows the importance of data after analysis. This is due to the fact that the data on its own is almost worthless. The value lies in careful analysis of the exact data, the information and ideas it provides.

The value is the final stage that comes after processing volume, velocity, variety, contrast, validity and visualization [13]. There have been numerous revisions to the big data until they reached (7 v) [14]. In this paper, based on the relationship between cloud computing and big data, will suggest a new term, virtualization, which virtually represents the data structure is by default. The virtualization of big data is a process that focuses on creating virtual structures for big data systems.

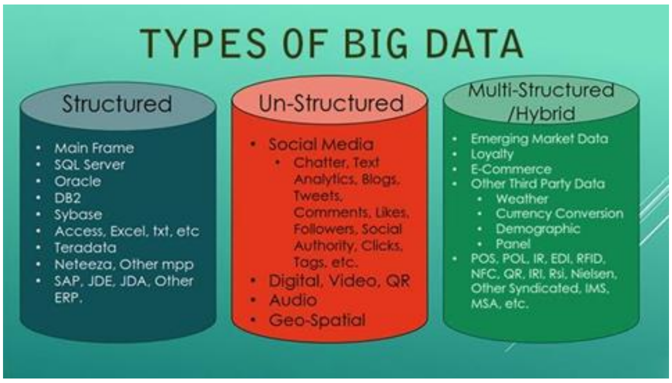

Fig1. Type of Big Data

1.2. Why Big Data

Big data is used to characterize structured and unstructured data in a standard manner. Though Big Data can agreement with the various types of organized and unorganized applications activities and channels such as emails, tweeter, web logs, Facebook etc. The main technical hitches with big data include capture, storage, search, sharing, analysing, and visualization. The core of Big Data is Hadoop which is platform for distributing computing problems across number of servers. Hadoop Map Reduce involves distributing a dataset amongst numerous servers and operating on data: the “map” stage. The biased stage is then recombined: the “reduce” stage. Technologies today not only carry collection of data, but also help in utilizing such amount of data successfully. [2] Before using Big Data we depend on database to store up our data. But a single database technology cannot make happy the need of every organization. This can be factual in terms of multi-structured data and that is why various applications are using Big Data technology now a days because with this we arrangement with a large volume of data. Web-persistent companies such as Google, Yahoo & Amazon are totally dependent on Big Data these companies have significant volume of information to index and analyse is the instance of optimized solutions. These companies have located its systems into public area so that they can be obtainable as open source software. [3]

1.3. Difference between Traditional Data and Big Data

In general, the data in the world of technology is a set of letters, words, numbers, symbols or images, but with the evolution of multitasking technology tools the data has in light of this, big data emerged which differs from traditional data.

Big Data		**Traditional Data**
PBs And ZBs	MB and GB	Volume
More rapid	Long periods of time	Data GenerationRate
Sim-Structure , Unstructured	Structure	Data Type
multiple sources, and distributed	Centralized	Data sources
HDFS, No SQL		

Table 1 Comparison between traditional and big data

1.4 Map Reduce

Map Reduce is a software construction introduced by Google to act upon parallel processing on large datasets supercilious that large dataset storage is distributed over a large number of machines. Each machine computes data stored locally, which in turn contributes to distribute and parallel processing. There are two parts to such a computation - Map and Reduce. Data nodes assigned to the Map phase, take raw input data and based on the type of computation required produce intermediate data that is stored locally. Reduce nodes take these intermediate outputs and combine them to derive final output which is then stored in HDFS. Hadoop tries to collocate data and the computation. Name node with its knowledge of how the data is distributed, tries to assign the mission to the node in which the data locally resides. Programmers can write custom map and reduce functions and Map Reduce function repeatedly takes care of distributing and parallelizing the tasks across an array of commodity machines in the cluster underneath. It as well manages inter- machine communication leaving programmers to focus on actual map-reduce functions.

Hadoop uses this fault tolerant, reliable, distributed, parallel computing framework to analyze large datasets distributed over HDFS.

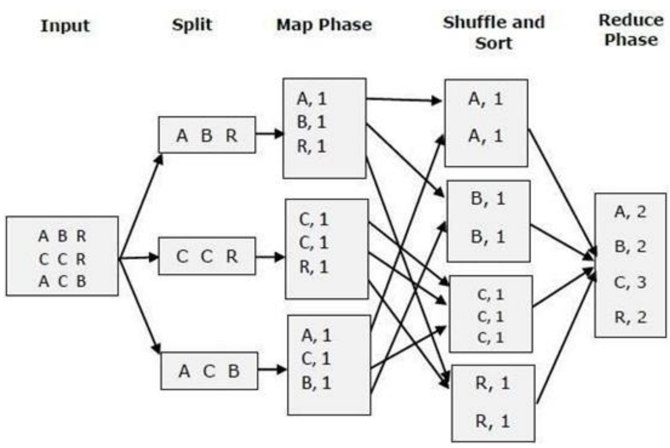

Fig 2 Map Reduce Example

2. Problem statement

Map Reduce [2] is a programming model invented by researchers at Google. It is meant for processing large amounts of data by a large cluster. Since it is hard to parallelize such large amounts of data over multiple hundreds of machines, the authors developed the Map Reduce model. The Map Reduce algorithm is easy to use and handles all parallelization, so the programmer does not have to concern about that. Google used the Map Reduce model for many applications, such as the indexing of websites on the internet. They use large clusters of cheap machines to process all data. Since Map Reduce was originally developed for high performance computations on heterogeneous clusters of computers, there are quite some research papers about optimizing performance of multi-core computers. Problems may occur in these configurations because of the processors having one shared memory instead of having smaller pieces of memory available for each single processor core. One of those optimized implementations of Map Reduce is the distributed system [9]. There are of course many more implementations of Map Reduce. One of the biggest frameworks for Map Reduce is Hadoop [5]. Hadoop is used by large, international companies like Amazon and Facebook [6]. Hadoop uses its own file system (HDFS) as storage for its machines and can be used for many purposes.

3. Map

The Mapper is the function that processes the data by reading all the data and returning a list of key-value pairs. Each pair contains the object that is processed and the value that is sent to the reducer. These pairs will be noted as following: The results of the map function are stored in the memory of the machine. Periodically these values are written to the disk of the machine and the Master node is notified of the location of the intermediate results. The Master node then marks this mapper node as free and may assign a new task to it. This task may either be a map task or a reduce task. For example: When all the words in one document are counted by Map Reduce, the Map function just runs through the document and returns a pair for each word it encounters. The key is set to the word that is just read and the value is set to 1. All these pairs are stored in the local memory and finally written to the disk. For example: if the text 'more and more' is read, the following pairs are stored: Notice that the same pair is stored twice.

4. Reduce

The Reducer is the function that reads the intermediate data from the Mapper and reduces this by processing the objects by running the user-defined Reduce function. First it receives a location of the mapped data from the Master node and fetches this data. Then this data must be ordered by key.

Then the Reducer iterates through the data and groups all keys that are equal by processing their values. Finally this output file is written and the Master node is notified. If the output of the above Mapper is processed by a Reducer, the following file will be generated. You can see that the key 'more' is grouped and their values are added up, while the key 'and' did not change at all. Also the output is now ordered by key. The output files of the Reducer tasks may be input for another Map Reduce run or may be used as separate files.

5. Master node

When a Map Reduce task is started, one Master node is chosen from all machines in the network. This Master node is the machine that delegates tasks to other machines and is also the only node that has a complete overview of what is happening in the network. The Master node assigns

new tasks to worker nodes and reassigns tasks that take too long. A sequence overview is as follows. 1. The input data is split into a number of pieces of a specified size. The algorithm is started on all nodes.
2. One node is set to be Master node and starts delegating work to other nodes. All pieces created in the first step are first mapped by the mapping function. The number of reduce tasks at the start should be low. 3. If a worker gets a map task, it runs the map task and stores the result in the memory of the machine. 4. Periodically these stored results are written to the disk and the Master node is notified of the location. 5. When the Master node gets notified about a location of mapped pairs, it will start a reduce tast on one of the free workers. 6. When a reduce task is called, first of all it fetches the stored results from the remote machine on which the map task has run. Secondly, these results are sorted by key. Thirdly, the results are reduced. When there are no more data to process, the Master node returns the final results to the user program. All this time the Master node has an overview of what all nodes are doing. The master will also re- assign already assigned tasks to idle nodes, because this might improve overall performance.

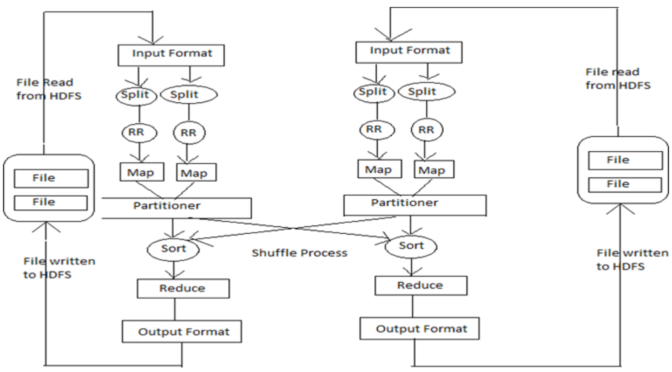

Fig 3 : Flow chart for Map Reduce

6. Different algorithm used in Map Reduce

- Mapper
- Reducer

6.1 Mapper

The map step essentially solves a small problem: Hadoop's partitioner divides the problem into small workable subsets and assigns those to map processes to solve. In Mapper each input request is being partitioned into intermediate requests i.e. system can individually perform different algorithms on file which can act as intermediate processes. The output from certain mapping processes needs to be accumulated before the reducers can begin. Or, some of the intermediate results may need to be processed before reduction. Some of this output may be on a node different from the node where the reducers for that specific output will run. The gathering and shuffling of intermediate results are performed by a partitioner. The map tasks will deliver the results to a specific partition as inputs to the reduce tasks. After all the map tasks are complete, the intermediate results are gathered in the partition and a shuffling occurs, sorting the output for optimal processing by reduce. Map definition purposely used the work "essentially" because one of the things that give the Map step its name is its implementation. While it does solve small workable problems, the way that it does it is that it maps specific keys to specific values. For example, if we were to count the

number of times each student name appears in a file, our Map Reduce application would output each name as a key and the value as the number of times it is seen. Or more specifically, the file would probably be broken up into independent student record, and the Map step would return each name mapped either to the number of times it appears in the file and then the reducer would combine the keys by adding their values together. [19]

6.2 Reducer

The reducer combines the results of the mapping processes and forms the output of the Map Reduce operation. The framework calls the application's

Reduce function once for each unique key in the sorted order. The Reduce can iterate through the values that are associated with that key and produce zero or more outputs. In the word count example, the Reduce function takes the input values, sums them and generates a single output of the word and the final sum.In our University system, the Reduce function take the input, calculate their corresponding result and give the final result as a single output.

7 Map Reduce Algorithms

We use five algorithms of Map Reduce to access our data through the chunks of the files which are stored as a cluster in a folder which behave as an arbitrary server for our system. Algorithm used in the Map Reduce technique are shown below

7. 1 Example 2 : Join Algorithm

A join combines records from two or more data sets by a common key. There are several ways to implement join in Map Reduce. A

straightforward approach is the reduce-side joins that take advantage of the identical keys to the same reducer. In practice, join, aggregation, and sort are frequently used together, e.g. finding the student scores maximum during the period. In Map Reduce, this has to be done in multiple phases. The first phrase filters the data base on the click timestamp and joins the client and click log datasets. The second phrase does the aggregation on the output of join and the third one finishes the task by sorting the output of aggregation.

1. Public class JoinThread
2. JoinThread(ArrayList
3. r1,ArrayList r2,ArrayList r3,int r)
4. Public void run()
5. String A[]=rsl1.get(rno)
6. String B[]=rsl2.get(rno)
7. String C[]=rsl3.get(rno) 8. For(i=1;i<=4;i++)
9. A=parse integer(A[i])
10. B=parse integer(B[i])
11. C=parse integer(C[i])
12. Total[i-1]=a+b+c
13. End for
14. End run
15. End class

7.2 Example 3: Aggregation Algorithm

Aggregation is a simple analytic calculation such as counting the number of access or users from different colleges. Word count, the hello world program in the world of Map Reduce, is an example of aggregation.

1. Public class FrequencyThread
2. FrequencyThread(String
3. nm,ArrayList al1,ArrayList al2)
4. Public void run()
5. If (alA.contains(name))
6. Int i= alA.indexOf(name)
7. Int f= Integer parse int((String)alB.get(i)) 8. f++
9. set alB(I,f+””)
10. Else
11. alA.add(name)
12. Add 1 to alB
13. End if
14. End class

7.3 Map Reduce Program Implementation

Once we are ready with the pre-requisites, we'll start writing the first MR program to solve the preceding problem. This MR program would contain three files:

1. Mapper Program
2. Reduce Program

Let's open Netbens and start the program:

Step 1: Store the data in small chunk of files on a cluster.

Step 2: Create new project with the name "Map Reduce" by using the JAVA environment built on Java 1.6 and above.
Step 3: Include the dependent java libraries into "Map Reduce" project.
Step 4: Create a folder in C: Directory and add the files which assume as a big data. We have creates eight files for university data considering it as a big data.
Step 5: Write a Map Reduce Program, its main file, which acts as a Job to submit the Mapper and Reducer programs.

Step 6: Write a Mapper Program. It reads file line by line, parses each line separated by a ',' (comma), and then it writes data in the form of a key and value pair and makes it available to the reducer program. In our case, Key will be the state name and value will be the whole line.

Step 7: The user Mapper.java program after being running.
Step8: User have a Intermediate frame where we have five algorithm:
Step 8.1: Sorting: We used Quick Sort in which user name act as a key and value corresponding to the name is being sorted it works on multithreaded working.
Step 8.2: Searching: In this we use linear search in which searching is being done corresponding to the name.
Step 8.3: Indexing: Indexing is done on basis of key value pair i.e. on basis of name all the names are being sorted and being indexed.
Step 8.4: Joining: Joining is used to join the records of two files. In joining name is act as a key for files and each value is being joined with each record of files on basis of key value pair.

Step 8.5: Term Frequency: Term Frequency is used to compare the name as a key value to another names and find which name occurs for how many times it compare each key of file with another key to find frequency.

Step 9: The Sorting. java, Searching. java, Indexing. java, Termfrequency.java, joining. java program is being run.
Step 10: Write a Reducer Program. It reads mapper output, and processes each key and value of mapper. In our case, we have to names are in files.
Step 11: The Reducer. java program.

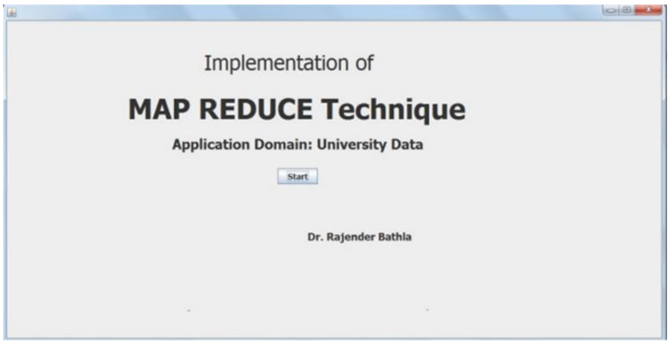

Fig 4: Main Frame of Using MapReduce

Description: Main Frame is a start up frame which Batch jobs can be taken off from main frame systems, processed using pig, Hive or Map Reduce and result can be moved back to main frame systems which helps reduce MIPS(Million instructions per second) cost.

8. Mapper Frame

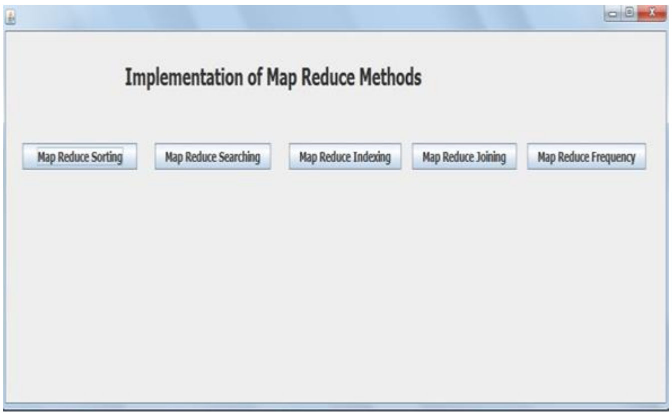

Fig 5. Implementation option frame

Description: Implementation Option frame is used to offer the user options of the algorithm used in Map Reduce. User can use different algorithm and can access the data of the files. Mapper task is the first phase of processing that processed each input record from record reader and generates an intermediate key-value pair. Hadoop mapper store intermediate-output on the local disk.

9. Joining

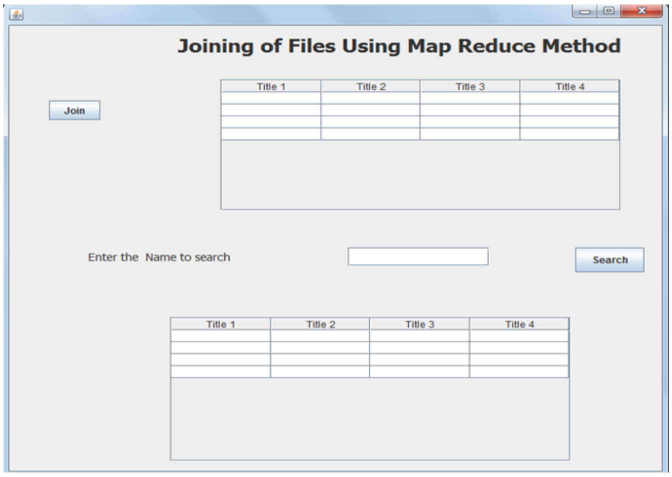

Fig 7 Result after joining and searching

Description: Joining frame is used to join the files on one screen. In join frame regn no. act as key for files in which we apply joining. We join the four files in this module. In this frame we uses three marks file of student which contain three year marks of student of different subject. We add the marks of four subjects and show the result in output screen. In this frame user can also access the result for any particular student result along with the complete file for the result.

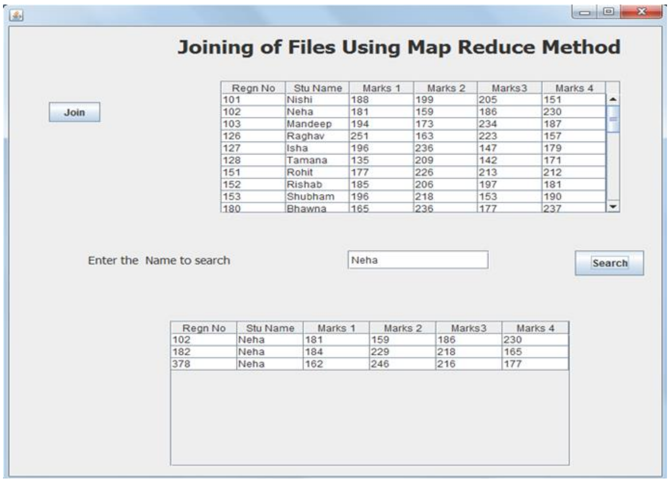

Fig 6 : Join Frame

Join Thread Coding

```
package mapreduce; import java.util.*;
public class JoinThread extends Thread
{
ArrayList rsl1,rsl2,rsl3; int rno;
int tot[];
JoinThread(ArrayList r1,ArrayList r2,ArrayList r3,int r)
{
rsl1=r1; rsl2=r2; rsl3=r3; rno=r;
tot=new int[4];
}
public void run()
{
int i,a,b,c;
String A[],B[],C[];
A=(String[])rsl1.get(rno);
B=(String[])rsl2.get(rno);
C=(String[])rsl3.get(rno); for(i=1;i<=4;i++)
{
a=Integer.parseInt(A[i]);        b=Integer.parseInt(B[i]);
c=Integer.parseInt(C[i]); tot[i-1]=a+b+c;
}
}
}
```

Conclusion

As we all know today thousands of users are active on internet, they want to access the required data in filtered form. This can be achived t hrough Big Data. Big data is very challenging research area. Data is too big to process using conventional tool of data processing. Academia and industry has work together to design and develop new tools and technologies which effectively handle to processing of Big Data. Big Data is an emerging trend and there is immediate need for new machine learning and data mining techniques to analyze massive amount of data in future. After years of practice, the community has realized these problems and try to address them in different ways. In this dissertation we conclude Hadoop & MapReduce tools. These tools are used for storing of structured\unstructured data in large amount this can be done through HDFS in which data is stored in clusters. MapReduce is used to process such a large amount of data in systematic manner. By using Hadoop MapReduce algorithms we can

using the basic algorithms of Hadoop and MapReduce. We can also process the data with the queries or through the interactive frames. We prefer to use the data processing through interactive frame work in which we can access the result of students from different colleges. And can also capable of getting all the records from clusters with high efficiency. Enterprises are increasingly looking to find actionable insights into their data. In our Dissertation we can conclude Hadoop MapReduce algorithm which can act on saved data in our files which behaves as a clusters. In this we can access all the files through various algorithm within efficient time. We analyse five algorithms: First is Searching, In which we can search the values corresponding to any one record, if the key we entered is available in our file then it shows the result else it shows no result found. Second is Sorting, In which all the records are being sorted through the name as name act as a key for our system. Third is Indexing, In which all the keys or names are being first sorted and then arranged according to their index number. Fourth is Joining: In which first the records of three files are being merged by the key value and we can also search for any particular record after applying joining operation. Fifth is Term Frequency: In which we can find for how many times a particular key or name can be occurred in our file. Many big data projects originate from the need to answer specific business questions.

II. REFERENCES

[1]. Netto, M.; Calheiros, R.; Rodrigues, E.; Cunha, R.; Buyya, R. (2018). "HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges". ACM Computing Surveys. 51 (1): 8:1–8:29

[2]. M. Tang, X. Dai, J. Liu, J. Chen, "Towards a trust evaluation middleware for cloud service selection", Future Gener. Comput. Syst., 74 (2017), pp. 302-312

[3]. M. Tang, X. Dai, J. Liu, J. Chen, "Towards a trust evaluation middleware for cloud service selection", Future Gener. Comput. Syst., 74 (2017), pp. 302-312

[4]. Y. Charband, N.J. Navimipour, "Online knowledge sharing mechanisms: a systematic review of the state of the art literature and recommendations for future research", Inf. Syst. Front. (2016), pp. 1-21

[5]. Nidal Hassan Hussein, Ahmed Khalid, "A survey of Cloud Computing Security challenges and solutions", International Journal of Computer Science and Information Security (IJCSIS), Vol. 14, No. 1, January 2016.

[6]. James Kobielus, I., & Bob Marcus, E. S. (2014). Deploying Big Data Analytics Applications to the Cloud: Roadmap for Success. Cloud Standards Customer Council.

[7]. J.K. Adjei, C. Blackman, C. Blackman, "Explaining the role of trust in cloud computing services", Info, 17 (2015)

[8]. Swarnalata Bollavarapu and Bharat Gupta, “Data Security in Cloud Computing”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 3, March 2014.

[9]. Chandrashekar, R., Kala, M., & Mane, D. (2014). Integration of Big Data in Cloud computing environments for enhanced data processing capabilities. International Journal of Engineering Research and General Science, 240-245.

[10]. Thilakanathan, Danan; Chen, Shiping; Nepal, Surya; Calvo, Rafael A.; Liu, Dongxi; Zic, John, "Secure Multiparty Data Sharing in the Cloud Using Hardware-Based TPM Devices," Cloud Computing (CLOUD), 2014 IEEE 7th International Conference on , vol., no., pp.224,231, June 27 2014-July 2 2014.

[11]. Wang, L., Wu, Z.: A Trustworthiness evaluation framework in cloud computing for service selection. In: IEEE 6th International Conf. on Cloud computing technology and science, Washington, DC, USA, 101-106 (2014).

[12]. A.F. Barsoum, A. Hasan, "Enabling dynamic data and indirect mutual trust for cloud computing storage systems", IEEE Trans. Parallel Distrib. Syst., 24 (2013), pp. 2375-2385

[13]. P. Manuel, "A trust model of cloud computing based on quality of service", Ann. Oper. Res. (2013), pp. 1-12

[14]. J. Dykstra, A.T. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: exploring and evaluating tools, trust, and techniques", Digital Invest., 9 (2012), pp. 90-98.

[15]. Satyajeet N Srujan Kotikela, Mahadevan Gomathisankaran, “CTrust: A framework for Secure and Trustworthy application execution in Cloud computing”, International Conference on Cyber Security, 2012.

[16]. Monoj Kumar Muchahari, Smriti Kumar Sinha, “A New Trust Management Architecture for Cloud Computing Environment”, IEEE International Symposium on Cloud and Services Computing (ISCOS), 2012.

[17]. Shaikh, R., Sasikumar, M.: Trust framework for calculating security strength of a cloud service. In: Proceedings of International conference on Communication, Information & Computing Technology, 703-709 (2012).

[18]. Rizwana Shaikh, M. Sasikumar, “Cloud Security issues: A Survey”, International Journal of Computer Applications, April 2012.

[19]. Punyada M. Deshmukh, Achyut S. Gughane, Priyanka L. Hasija, Supriya P. Katpale, “Maintaining File Storage Security in Cloud Computing”, International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 10, October 2012.

[20]. Yanjiang Yang and Youcheng Zhang. A Generic Scheme for Secure Data Sharing in Cloud. Parallel Processing Workshops (ICPPW), 2011 40th International Conference. Pages 145-153. September 2011.

[21]. E. C. Liu, X. Zhang, J. Chen and C. Yang, "An Authenticated Key Exchange Scheme for Efficient Security Aware Scheduling of Scientific Applications in Cloud Computing”, IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing (DASC), (2011), pp. 372 – 379.

[22]. Y. Singh, F. Kandah and W. Zhang, "A secured cost-effective multi-cloud storage in cloud computing", IEEE Computer Communications Workshops (INFOCOM WKSHPS), (2011), pp. 619 – 624.

[23]. F.B. Shaikh and S. Haider. Security threats in cloud computing. In Internet Technology and Secured Transactions (ICITST), 2011 International Conference for, pages 214-219, December 2011.

[24]. Hyukho Kim, Hana Lee, Woongsup Kim, Yangwoo Kim, “A Trust Evaluation Model for QoS Guarantee in Cloud Systems”,

International Journal of Grid and Distributed Computing, March, 2010.

[25]. Google Scholar: http://scholar.google.com

[26]. Microsoft Corporation: Microsoft azure: cloud computing platform and services, (2015).

[27]. Park K, Nguyen MC & Won H, "Web-based collaborative big data analytics on big data as a service platform", 17th International Conference on Advanced Communication Technology (ICACT), (2015), pp.564–567.

[28]. Vakintis I, Panagiotakis S, Mastorakis G & Mavromoustakis CX, "Evaluation of a Web crowd-sensing IoT ecosystem providing Big data analysis", Resource Management for Big Data Platforms, (2016), pp.461-488.

[29]. Park K, Nguyen MC & Won H, "Web-based collaborative big data analytics on big data as a service platform", 17th International Conference on Advanced Communication Technology (ICACT), (2015), pp.564–567.

[30]. Zhang L, Xie T, Tillmann N, De Halleux P, Ma X & Lv J, "Environment modeling for automated testing of cloud applications", IEEE Software, Special Issue on Software Engineering for Cloud Computing, Vol.1, No.20, (2012), pp.1-10.

[31]. Nathiya T, "Reducing DDOS Attack Techniques in Cloud Computing Network Technology", International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE), Vol.1, No.1, (2017), pp.23–29.

[32]. Mahalakshmi B, "A Detailed Study on Deduplication in Cloud Computing", International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE), Vol.1, No.1, (2017), pp.1–5.

[33]. Klous, Sander, and Nart Wielaard. We are Big Data: The Future of the Information Society. Springer, 2016.

[34]. SHAN, Y. C., Chao, L. V., ZHANG, Q. Y., & TIAN, X. Y. (2017). Research on Mechanism of Early Warning of Health Management Based on Cloud Computing and Big Data. In Proceedings of the 23rd International Conference on Industrial Engineering and Engineering Management 2016 (pp. 291-294). Atlantis Press, Paris.

[35]. Parvin Ahmadi Doval Amiri and Mina Rahbari Gavgani, 2016. A Review on Relationship and Challenges of Cloud Computing And Big Data: Methods of Analysis and Data Transfer. Asian Journal of Information Technology, 15: 2516-2525

[36]. Demchenko, Yuri, et al. "Big security for big data: Addressing security challenges for the big data infrastructure." Workshop on Secure Data Management. Springer, Cham, 2013.

[37]. Calheiros, Rodrigo N., et al. "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms." Software: Practice and experience 41.1 (2011): 23-50.

[38]. R.Subhulakshmi , S.Suryagandhi , R.Mathubala , P.Sumathi, An evaluation on Cloud Computing Research Challenges and Its Novel Tools, International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST) Volume 2, Special Issue 19, October 2016.

[39]. Fonseca, N., & Boutaba, R. (2015). Cloud services, networking, and management. John Wiley & Sons.

[40]. Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. Journal of internet services and applications, 1(1), 7-18.

[41]. Vacca, J. R. (Ed.). (2016). Cloud Computing Security: Foundations and Challenges. CRC Press. ch-15.

[42]. Terzo, O., Ruiu, P., Bucci, E., & Xhafa, F. (2013, July). Data as a service (DaaS) for sharing and processing of large data collections in the cloud. In Complex, Intelligent, and Software Intensive

Systems (CISIS), 2013 Seventh International Conference on (pp. 475-480). IEEE.
[43]. Rajesh Saturi, Data as a Service (Daas) in Cloud Computing [Data-As-A-Service in the Age of Data] Data as a Service Daas in Cloud Computing, Global Journal of Computer Science and Technology Cloud & Distributed Volume 12 Issue 11 ,2012.
[44]. Klous, Sander, and Nart Wielaard. are Big Data: The Future of the Information Society. Springer, 2016.
[45]. Chen, Min, et al. Big data: related technologies, challenges and future prospects. Heidelberg: Springer, 2014.
[46]. R. Subhulakshmi, S. Suryagandhi, R. Mathubala, P. Sumathi, An evaluation on Cloud Computing Research Challenges and Its Novel Tools, International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST) Volume 2, Special Issue 19, October 2016.
[47]. Big data: The next frontier for innovation, competition, and productivity. James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers. McKinsey Global Institute. May 2011.
[48]. Rouse, M. (2010a, August). Software as a service. Retrieved from http://searchcloudcomputing techtarget.com/definition/Software-as-a-Service.
[49]. IOS Press. (2011). Guidelines on security and privacy in public cloud computing. Journal of E- Governance, 34 149-151. DOI: 10.3233/GOV-2011-0271.
[50]. Apache: Apache Hadoop, http://hadoop.apache.org

Author Details

Prof.(Dr.) R. K. Bathla is working as a Professor in Computer Science Department at Desh Bhagat University, Mandi Gobindgarh, Punjab, India. He is also Pursing D.Sc in Computer Science & Applications (Post Doctoral Fellow) from Desh Bhagat University, Punjab. He has 21 years experience in the field of Computer Science/Information Technology. He has published over 100 Research papers in SCI, Scopus, Springer and IEEE. He has 40 Patents Published and Granted in his name at National and International level. He has been guiding 08 Ph.D. scholars in his supervision, 06 have been awarded degree. In over 50s International journals, he is in editorial board or working as reviewer. He has been awarded India Top 100 Best Researcher Winner 2022 by India Prime Award and Awarded Cash Prize and appreciation certificate for Best Research paper from Hon'ble Chancellor of Desh Bhagat University, Punjab on the occasion of 05th Sep 2022.

Cite this article as :

Prof (Dr.) Rajender Kumar, "Comparative Analysis of Map Reducing Algorithm and Coding Approach in Using Big Data Technique", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 6, pp. 464-475, November-December 2022.
Journal URL : https://ijsrcseit.com/CSEIT228670