International Journal of Advanced and Innovative Research (2278-844)/

Volume 7 Issue 11

A Consideration Independent Score Based Benchmark for Distributed Database

[1]MD. Zaheda Parveen, [2]P Sravanthi

[1]Research Scholar, [2]Associate Professor, Department of Computer Science & Engineering

KODADA Institute of Technology & Science For Women, Kodad

Abstract: Big data and hadoop plays an important role in dealing with the large number of process and big data processing in different scenario. There are multiple nodes participate in the Distribution and different type of data movement scenario. The existing algorithm which is taken for the further improvement work is distributed dataset algorithm, which is phase based Distribution algorithm to process the large number of data and data packets. Our enhance distributed dataset algorithm technique outperform as compare with the existing distributed dataset algorithm and other scheduler so far. Thus the requirement leads us to perform further improvement to the distributed dataset algorithm Distribution technique. In this dissertation our work perform to create a new heuristic based Distribution algorithm which make use of existing scenario and improve it with balanced distribution technique. The proposed technique is based on enhance model which is heuristic & priority based Distribution and load balancing Distribution based scenario which make use of all the resources properly and outperform the complete distribution using the provided Distribution algorithm. The proposed work implemented and tested in java technology and upon processing data range from 1 gb to 5 gb there are performance monitored in terms of computation processing time, throughput with hardware configuration as 4 gb ram, 750 gb hard disk and the competitive results were monitored. Thus the result as compare to existing technique were monitored which are 1.1 x improved while comparing with existing algorithm.

Keywords: Big Data; Distribution Model; Hadoop; Node; Parameters.

1. INTRODUCTION

Big data is set of large volume of data which is difficult to manage, store and analyze by conventional tools. Huge amount of data is generated every minute by various online applications like social networks, e-commerce applications etc. The International Data Corporation (IDC) report estimated that Data will grow to 40 zeta bytes (10^{21} bytes) in 2020, at a 40% annual increase. Web is an extraordinary data set with which we interact with the use of search engines. Such huge data sets possess rich information and knowledge which are very helpful in various sectors such as medicine, health care and business and many more. The management and processing of huge data sets are time-consuming, costly and hindrance to research. The Big Data technology has strategies and analytical tool so that large datasets can be efficiently handled to extract out creative ideas and new values out of them.

Big Data is an important research area for cloud service providers. Cloud computing is a technology trend in which a user can utilize software, hardware and infrastructure on rent according to a per use basis. Many cloud providers are there all over around the world, which provide facilities for setting up, manage and maintain private storage infrastructure. Some of the examples are Amazon S3, Google Cloud Storage and Microsoft Azure. There are data centres in online cloud environment which are located at distributed geographical areas all over the world. Cloud service providers apply various techniques in order to provide good service, fast response time, availability and content according to user interest in various geographical areas. In online cloud environment, there is transfer of big data between different geographically distributed data centres which require efficient methods of migration. This problem arises due to huge size and complexity of Big Data which is rapidly increasing. Framework Map Reduce consumes lot of bandwidth for processing of big data at different geographical

locations. One approach to solve this problem is Data aggregation. Data aggregation is the smart placement of data at a particular data center so that it is cost efficient and easy to access. This work proposes an online algorithm to find optimal cost data aggregation site. Optimal cost data aggregation site provides efficiency in data processing at single site using distributed framework.

Figure 1.1: 3V's Factors Of Big Data.

In cloud environment, data is stored at data centres in all the geographical location. There is movement of data in both way, inside the data centres and between the geographically distributed data centers. High speed WAN are used to connect all the data centeres. These WAN links have high bandwidth and have specific design to facilitate traffic between these data centres.

2. LITERATURE REVIEW

Cost Effective Social Network Data Placement And Replication Using Graph Partitioning

Users are connected on social networks according to their various interest, ideas and different groups they are associated with. Well known examples of social networks such as Facebook and Twitter have billions of users all over the world who share interconnected data. Users wish low latency access to data whether it may be their own or it is of their friend. However Social service network providers want to pay least possible cost to store data to meet low latency access requirement for their user. As mentioned above, Cloud Services having virtually unlimited capability is best for storage of such unlimited data scattered all over the globe. This data require optimal cost placement and replication. Also proper distribution of requests among data centres is also necessary to reduce monetary cost while satisfying requirement of low latency.

Yun Yang et al. [2] proposed a novel graph partitioning based approach to find a near optimal data placement of replicas to reduce cost with having latency requirement fulfilled. They performed experiments on a Facebook dataset and demonstrated effectiveness of their approach in outperforming other methods for data placement and replication. Each user has a primary copy of data located in his primary data center. It is assumed that every user read his data from his primary datacenter and his all friends read their data from their nearest datacenter which stores any secondary copy of their data. Users' data are accessed by their connected friends. So, every user's data and all friends' data can be placed in the same datacentres. They presented a novel vertex-cut graph-partitioning algorithm to group connected users to the same partitions. Then, data placement and replication are done for users in each partition by placing data of every partition to the nearest datacenter to the user with the maximum number of friends in the partition. In a vertex-cut algorithm in which users are assigned to various partitions is considered, users' data can be replicated in various datacenters depending on the partitions they are assigned to. They used the SNAP Facebook dataset to test their prototype and experimental results show effectiveness of their algorithm. Comparing to various placement strategies, this proposed algorithm can find the most cost effective data placement and replication technique while guaranteeing the latency requirement for online social network users.

This GP based strategy can find the minimum cost while guaranteeing the latency requirement for 90 percentile of users. Some strategies like genetic algorithm could guarantee the latency requirement but at a higher cost than this Graph partitioning based strategy. Updating of data is not considered in the work which is postponed to the future. Selection of the number of partitions will be optimized in the future.

An Efficient Transportation Architecture For Big Data Movement

Big Data being large in size require efficient protocols for data transfer. In packet networks, data are transferred across the network as a combination of a series of packets, delivered one by one, without entirely considering the data with respective service quality requirements. Therefore, the quality of service of interactive big data applications is hard to manage, and the utility of network is low and the enhancement in transmission protocols becomes necessary. By relaxing the delivery time of big data files according to their priorities, the network resources may be used more effectively and network congestion can be reduced, and optical network running in circuit switching pattern could play a new role in doing this.

Shilin Xiao et al, [3] proposed and discussed a new data transfer model on optical network to complement the existing per packet forwarding paradigm on packet network. In this data transfer model, instead of transferring these data on per packet bases immediately after entering the network, the optical network stores the data as long as it finds necessary, or enough network resource is available for data transfer, or delivery the data in step-by-step ways by using relay storage in each switching nodes.

They showed improvement the performance for data movement over other transportation methods by their proposed model. Big Data transfer on packet network includes usage of transmission control protocol (TCP), FAST TCP, InfiniBand WAN and Remote Direct Memory Access over Converged Ethernet. On optical network Big data transportation are done through Data Driven dynamic Optical Pipelines, Multiple Connections Per Optical Interface and Flexible Optical Pipelines. Architecture proposed by them is called a Store, Schedule and Switch (SSS) model by employing both advantages of storage and circuit switching. Simulation shows that increasing the storage capacity on switching nodes is helpful to minimize the number of discarded data caused by shortage of storage.

Meanwhile, increasing the storage capacity will increase the number of accepted data flows and hence will increase the number of data flows that cannot be delivered without the node storage. So the overall discard ratio drop slightly as the storage capacity increases.

Optimal Decision Making For Big Data Processing At Edge-Cloud Environment: An SDN Perspective

Aulja et al, [4] proposed an efficient workload slicing scheme using Software Defined Network for handling Big Data in Edge Cloud Environment which lowers the energy consumption of data intensive applications. In the present age of cloud computing, data explosion needs to expand data centres for faster response. So, to minimize the load on data centres, some applications may be executed on the edge devices near to the proximity of the users. However, such type of multi edge-cloud environment involves huge amount of data to be migrated across the underlying network infrastructure which may create long migration delay and cost.

Hence, in the above mentioned work, an efficient workload slicing scheme is proposed to handle data-intensive applications in multi edge-cloud environment using software defined networks. To handle the inter data centre migrations efficiently, a SDN-based control scheme is put forward which provides energy aware network traffic flow scheduling. Lastly, a multi-leader multi-follower Stackelberg game is proposed to ensure cost-effective inter data center migrations. The scheme has been evaluated on the basis of various parameters such as energy, delay, migration rate, and cost. The obtained results show that the scheme minimizes the energy consumption of overall multi edge-cloud environment and underlying networks. Moreover, a reduced delay and cost of inter data centre migration is also achieved.

3. PROBLEM IDENTIFICATION

As per the literature survey is performed with different techniques and different result from the algorithms which are based on multiple other schedulers and their technique on large amount of packet size and with their data Distribution is being observed. .

Upon verifying different scenario and the available technique different short comes with the existing algorithm for Distribution, technique distributed dataset algorithm Distribution procedure follow the manner of distribution in hadoop platform.

The following are the monitored points which identified as problem and further analyzed and performed further with enhancements.

1. Previous technique such as distributed dataset algorithm & other Distribution algorithm for the processing process the data but still the computation

is large and proper node data balancing is not performed. This technique persist better result than existing but still enhancement is required which is provided by the proposed procedure.

2. In previous technique distributed dataset algorithm doesn't allow the high priority based search and further working with priority order with the existing working node in hadoop platform.

3. In the existing distribution throughput as well as io utilization with the system is not highly efficient, where as in new technique enhance distributed dataset algorithm an efficient procedure is used, which provide a flexible framework for the process.

Thus in order to proposed a better prediction model using classification and further combine approaches requirement is to further acquire an scheme which contribute on getting better outcome and system, here our proposed methodology enhance distributed dataset algorithm is utilize scheme in place of traditional Distribution approach.

4. PROPOSED METHODOLOGY

As per our observation about the previous technique and their disadvantage in different terms and scenario's. Our work present a new approach which is efficient while comparing with the existing model which product low computation and a proper throughput while working with large data packet size.

Our work propose a new algorithm enhance distributed dataset algorithm which utilize a new aprioristic distribution technique, which give a relation between the node, its available data and also provide a flexible environment for the complete process and thus it generate a better Distribution model for data transmission over the hadoop platform.

The proposed algorithm is described below:

1. Loading of all the available data & packets from the created given message which is participating for the communication.

2. Loading the complete node information and its configuration from the setup framework in hadoop.

3. Selection of the algorithm procedure.

4. Perform node data packet transmission operation and further re-Distribution approach and conclude that further using model for the data shifting either it is working or not.

5. Perform model and monitored the complete balance data over the node by the system.

6. Obtaining parameters and monitors the result obtained over the console window such as computation, throughput and other parameter such as IO.

7. Observing the values and thus it effect accuracy and efficiency for the complete scenario.

8. Exit.

Algorithm Pseudo Code:

Proposed Distribution heuristic based technique:

Input: node data qi,

Output: algorithm process, metadata, node values.

Steps:

Active either distributed dataset algorithm or heuristic

While(true) do{

Node distribution{p1, p2…..pn};

Dictionaryrequest();

If(scorematching()==1)

{

Recognition();

Perform heuristic model;

Compute the prediction values;

{

Result computation;

}

Set status=finish and exit;

}

if(scorematching()>=2)

{

Re-distribution;

{

Computing parameter upon distribution;

}

Set status=finish and exit;

}

5. RESULT ANALYSIS

Data Set

Data packet which can be simulate can be given as input and to process in the flow where the further distribution among the node can be perform.

Performance Measures

Computation time(parameter name)

A training time of a dataset in java is computed with the help of start and end time long variables defined in the tool and here as perform the Re-Distribution and taking their features for consideration or not is the time taking process to identify and to load the complete data and processing it via re-Distribution operation. Thus the after balancing of complete data a computation from initial time to last monitored time difference is considered as computation time.

Table 1.1: Comparison Between Data Packets And The Existing Technique.

TECHNIQUE APPROACH DATE PACKETS	EXISTING TECHNIQUE MODEL
128	459ms
256	1121ms
512	2123ms
1024	3390ms

The above table represents the number of data values from the data and algorithm is performed.

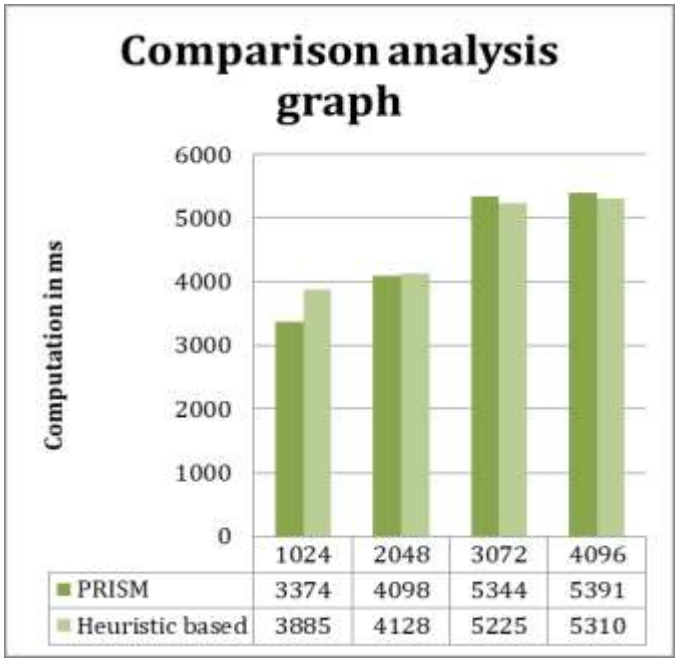

Figure 1.2: Comparison Line Graph For Technique Analysis.

In the table present below is a statistical comparison of the values which are retrieved as throughput by the different

process algorithm, throughput and other parameter can be observe.

Table 1.2: Data Distribution For Different Data Packet.

TECHNIQUE APPROACH DATE PACKETS	EXISTING TECHNIQUE MODEL	PROPOSED TECHNIQUE MODEL
1024	3.32	3.55
2048	4.51	5.1
3072	5.89	6
4096	7.12	7.8

The above table represents the number of data values from the data and algorithm is performed.

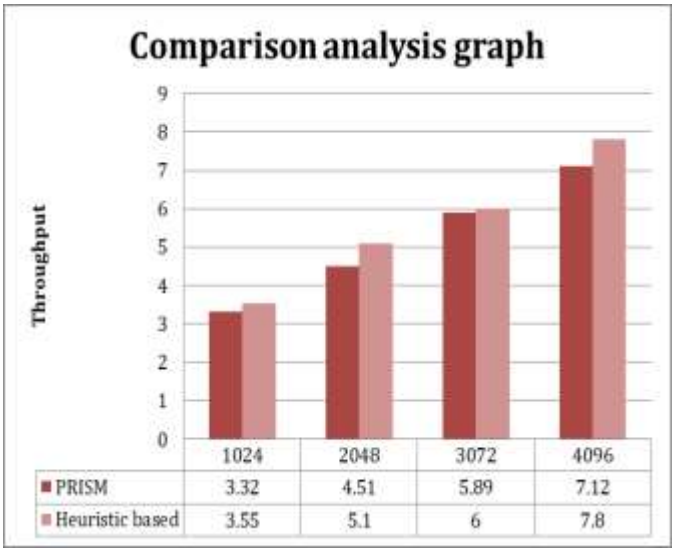

Figure 1.3: Comparison Line Graph For Technique Analysis.

In the above graph drawn x axis as data from which post were extracted for the query processing for specified dataset and line graph is printed using the graphic library.

The graph representation shows the efficiency of our proposed algorithm work and it outperforms effective parameter value**.**

5. CONCLUSION & FUTURE WORK

In this dissertation work research is performed with the hadoop file & Distribution algorithm. Where the existing algorithm such as distributed dataset algorithm is taken as the previous scheduler who work with the multiple node and to apply with large data transfer in minimum effect of time. Thus working with the hadoop distribution file system and re-scheduling algorithm working performance with the enhance distributed dataset algorithm is taken in consideration , which work with low computation time event while considering large data packet size to transfer and re-Distribution on node failure. The observed results were applied using machine having configuration 4 gb ram, 750 gb hard disk and i3 processor. Observed result using the simulation setup and algorithm process found the efficiency of proposed enhance distributed dataset algorithm based algorithm over the existing scenario with multiple data processing nodes.

Distribution algorithms having a future driven scenario requirement, where the different algorithm take participate and outperform Distribution over the multiple node participants. Here in the proposed work a better Distribution is outperform and further improvement can be done by checking the security measures in between the algorithm. Thus the efficient way to data transmission can be done in between the different participant nodes.

REFERENCES

[1] D. Agrawal, S. Das, and A. El Abbadi. Big Data and Cloud Computing: Current State and Future Opportunities, Proc. 14th Int'l Conf. Extending Database Technology (EDBT/ICDT 11), ACM, 2011, pp. 530–533.

[2] D.J. Abadi, Consistency Tradeoffs in Modern Distributed Database System Design: CAP Is Only Part of the Story, Computer, vol. 45, no. 2, 2012, pp. 37–42.

[3] Rakesh Rathi, Sandhya Lohiya, Big Data and Hadoop, International Journal of Advanced Research in Computer Science & Technology (IJARCST 2014), vol. 2, pp. 214-217, April-June 2014.

[4] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung (October 2003), The Google File System, 19th Symposium on Operating Systems Principles (conference), Lake George, NY: The Association for Computing Machinery, CiteSeerX: 10.1.1.125.789, retrieved 2012-07-12.

[5] Soubhagya V N1 , Nikhila T Bhuvan,Highly Available Hadoop Name Node Architecture-Using Replicas of Name Node with Time Synchronization among Replicas,IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 3, Ver. II (May-Jun. 2014), PP 58-62.

[6] Y.A. Adekunle, A Comparative Study of Scheduling Algorithms for Multiprogramming in Real-Time Systems, Vol. 12 No. 1 Nov. 2014.

[7] Arun Pasrija1 and Shikha Sharma², Reviewof Web Pre-Fetching and Caching Algorithms, Volume No. 3, Issue No. 1, January 2014

[8] Jungha Lee1, Jaehwa Chung2, Efficient Data Replication Scheme based on Hadoop Distributed File System, Vol. 9, No. 12 (2015), pp. 177-186.

[9] G.L. Prajapati, Study of Selected Shifting based String Matching Algorithms, International Journal of Computer Applications (0975 – 8887) Volume 140 – No.9, April 2016.

[10] Mohammad Hammoud and Majd F. Sakr, Locality-Aware Reduce Task Scheduling for MapReduce.