International Journal of Early Childhood Special Education (INT-JECSE)
DOI:10.9756/INTJECSE/V14I6.340 ISSN: 1308-5581 Vol 14, Issue 06 2022

STUDY AND ANALYSIS ON SECURE AND TRAFFIC AWARE MAPREDUCE PROGRAMMING ON BIG DATA

[1]B Swathi, [2]Dr.Meghna Dubey
Research Scholar Mewar University,Chittorgarh,Rajasthan
Research Supervisor Mewar University,Chittorgarh,Rajasthan

Abstract:- The purpose of the system to cut down network traffic cost. Map out a novel intermediate data participant schema. The map reduce type simplifies the large scale data deal with product group even though many times effort have been made to maximize the execution of map reduce work. They ignore the network effects which conclude. A fundamental part in implementation update. A hash capacity made use of session middle of the topology among minimizes the task even so is note movement valued in network topology. At last board reproduced outcome that shows at the proposed algorithm can together minimize network cost under both offline and online cases.

1. Introduction

MapReduce programming paradigm has changed the way programs are written. This is the new model of programming that runs in distributed environment. Hadoop is best example for distributed programming framework which is open source and supports MapReduce programming. This kind of programming is suitable for processing big data or voluminous data which grows exponentially. Enterprises like Yahoo, Facebook and Google are using MapReduce programming to handle data of their increasing number of clients. There are many applications that run using MapReduce programming. They include applications related to cyber security, bio informatics and machine learning to mention few.

MapReduce programming divides a computation into two phases namely map and reduce. Map phase may have several map tasks and reduce phase may have several reduce tasks. All map tasks are executed in parallel. Map task converts original input into intermediate data in the form of key/value pairs. These are stored in local machine and then organized into multiple parts so as to give to reduce tasks. In the reduce phase, each reduce task takes data from the partitioned data and in order to generate final result. There is another phase between these two known as shuffle phase. This phase is responsible for ordering data produced by map phase and the data is partitioned and transformed appropriately to give to reduce tasks. The resulting network traffic is caused by both map and reduce tasks that run in parallel in each phase. This leads to a serious constraint in the big data applications. In most of the applications data shuffling causes around 58.6 percent of traffic [4]. Therefore shuffle-heavy in MapReduce can lead to 30-40 percent performance overhead [15] which needs to be addressed. By default a hash function is used to shuffle intermediate data [16] with respect to Hadoop. This leads to huge amount of network traffic. The rationale behind this is that the shuffle phase ignores network topology and data size of each key. Ke et al. [2], of late, proposed a solution for this problem. They defined two algorithms to handle both traffic aware partitioning and data aggregation for dynamically coping with runtime situations. They considered data partition and aggregation for reducing network traffic and performance overhead.

Their results revealed difference between real traffic cost and the optimal traffic cost. This clearly indicates that the work of Ke et al. [2] can be enhanced. In this thesis we proposed alternative algorithms for both distributed algorithm that deals with huge traffic related to big data applications and an online algorithm that provides real time feedback to improve data partition and aggregation dynamically. A simulation environment is used to build a prototype application that shows proof of the concept. The results of the new algorithms are compared with that of traditional hash function and aggregation methods and traffic-aware partition and aggregation methods of Ke et al. The results are expected to improve performance of MapReduce programming as the traffic aware partition and aggregation in MapReduce is implemented with a novel approach for big data applications.

2. Literature Review

This section provides review of literature related MapReduce programming paradigm in distributed environment and issues related to that. Singh et al. [1] identified network traffic as a problem in MapReduce programming as it results in huge traffic. They found that traffic analysis for MapReduce where huge amount of network traffic is generated needs further research. They proposed and implemented an analytical framework for intrusion detection. They employed Random Forests for mining data and provide protection to MapReduce programming.

MapReduce programming generates huge traffic in shuffle phase also. Ke et al. [2] focused more on this kind of traffic and proposed two algorithms to deal with it. The first algorithm is meant for distributed traffic aware partition while the second algorithm focus on the placement of aggregator appropriately in order to reduce network traffic. Their effort was to reduce network traffic in MapReduce programming effectively. Towards this end, they proposed above said algorithms and implemented. This work is close to the work of this research. In the proposed research work, the author is going to provide alternative algorithms for further reducing the gap between real traffic and optimal traffic expected.

Kambatla et al. [3] focused on trends in big data analytics. They studied how the traffic is growing from time to time in case of MapReduce programming in the real world. They found that data is growing exponentially and there is need for big data analytics to provide sophisticated business intelligence. In this context, they also found in their study that data – driven models need to consider the traffic which is generated. They tried to characterize underlying hardware and software to assess their role in traffic and data analytics. Sankar and Murali [4] focused on big data analytics in case of mobile cellular networks. They made a survey of big data analysis with respect to cellular networks. They focused on the ways and means to reduce network traffic in the process of building big data analytics.

Ranjan et al. [5] focused on something known as Internet of Things (IoT) and Cloud of Things (CoT) that is related to big data and big data analytics. Cloud computing and IoT go hand in hand as the data produced by IoT is stored in cloud infrastructure. The CoT is the vision provided by IoT as it makes use of cloud as much as possible. They threw light into the traffic issues with respect to these technologies. They tried to identify traffic hotlines and found that they are the traffic intensive spots that need further optimization.

Dolberg et al. [6] studied two problems with respect to big data applications. They are known as flow scheduling and network configuration. They explored a tool known as OpenFlow controller for each major traffic flow. They studied the problem of traffic in distributed environment and emphasized the need for traffic aware networking. In case of MapReduce programming they envisaged the necessity of traffic aware approaches in map and reduce tasks. They provided guidelines to monitor traffic and handle with gracefully in the distributed programming paradigm such as MapReduce.

Zhang et al. [7] focused on tagged MapReduce programming. In this approach a tag is associated with map and reduces tasks. The tag is related to sensitivity of each key/value pair. Thus the processing of data is done as per the sensitivity level. This will enable secure computations and fine-grained data flow in MapReduce programming. They efforts led to the security of MapReduce computations. They could achieve secure computations in case of hybrid clouds where the services are rendered by a private cloud and a public cloud. Inter-cloud data traffic is managed with traffic aware approaches as underlying mechanisms in MapReduce programming. Software defined networking is the phenomenon which deals with controlling the network by using some external program or setup.

This kind of research is made by Cui et al. [8] to handle big data traffic in distributed programming paradigm like MapReduce. Their solution includes traffic engineering, secure communications, cross layer design, and handling security attacks. They proposed the possibility of SDN based intra and inter-data centre networks and the control of network traffic over there.

Terzi et al. [9] threw light into big data eco system, its security and privacy issues. The focused on security of big data in terms of monitoring and auditing, anonyization, key management, cloud security and Hadoop security. They emphasized big data safety and security. They also found the need for encrypting network traffic for secure computations in big data environment. Their work is related to finding security and privacy issues and overcoming those using different approaches in big data eco systems. In this proposal, the need for further research in the area of distributed programming paradigm for reducing network traffic and improving performance of MapReduce programming framework.

3. Resceach Methodology

Statement of the Research Problem

Research is simply the process of finding solutions to a problem after a thorough study and analysis of the situational factors.

The research topic on which the current research is being done demonstratesSecureand Traffic Aware MapReduce Programming on Big Data

Research Design

In MapReduce programming input data is processed by converting that into thousands of key/value pairs. Each key is associated with a set of variables. When such data is causing huge amount of network traffic, we considered it as a large-scale optimization problem. This problem is little addressed in the existing literature. HoweverKe et al. [2] focused on the traffic aware partitioning and aggregation to reduce network traffic and improve performance of MapReduce programming. However, their solution can be improved further as there is gap between the actual traffic and expected optimized traffic. This section provides the proposed methodology to take the optimization problem further to enhance performance and reduce network traffic. We proposed two alternative algorithms for distributed data partitioning which is traffic aware and data aggregation. The two algorithms will strive to work together to have joint optimization of MapRedcue programming in terms of reducing network traffic and performance overhead.

The proposed algorithms are used to further optimize traffic aware data partitioning and data aggregation in distributed environment. Real network traffic traces are used to have experiments. The traces are collected by the procedure followed in [2]. A cluster containing five virtual machines where each VM has 1 MB RAM and 2 GHz processing capabilities is used for experiments. The network topology is based on three-tier architecture. They are access tier, aggregation tier and core tier. Rack VMs are used which are connected using access tier

which is made up of Ethernet switches. There exists hop distances between mappers and reducers. Total number of physical machines is considered 10. Distance between any two machines is considered a value in the range 1-60. Maximum number of aggregators is set to 4. The data size of key/value pairs is 1-100. Figure 1 shows general MapReduce programming paradigm.

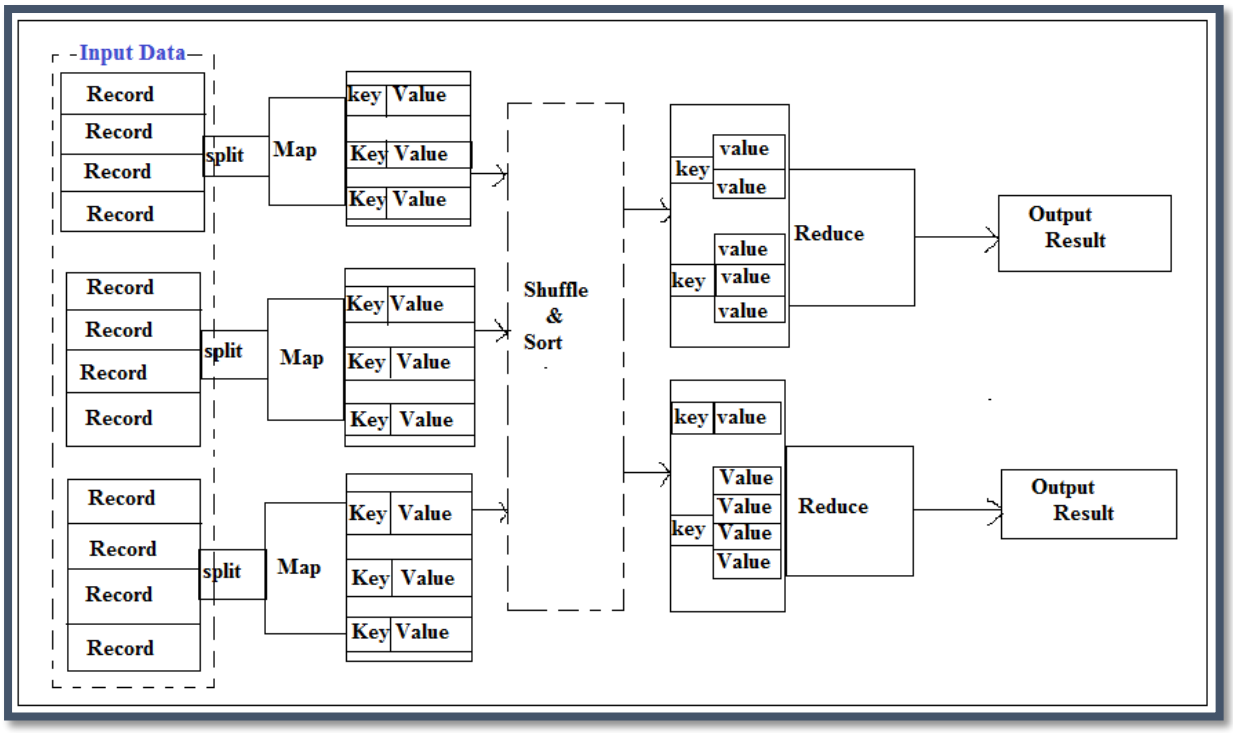

Figure 1: Execution of jobs in MapReduce programming paradigm

As shown in Figure 1, it is evident that the Map phase produced intermediate results in the form of key/value pairs. These results are then subjected to shuffle phase and the output is set to reduce phase. The reduce phase generates final output which is stored in the distributed file system.

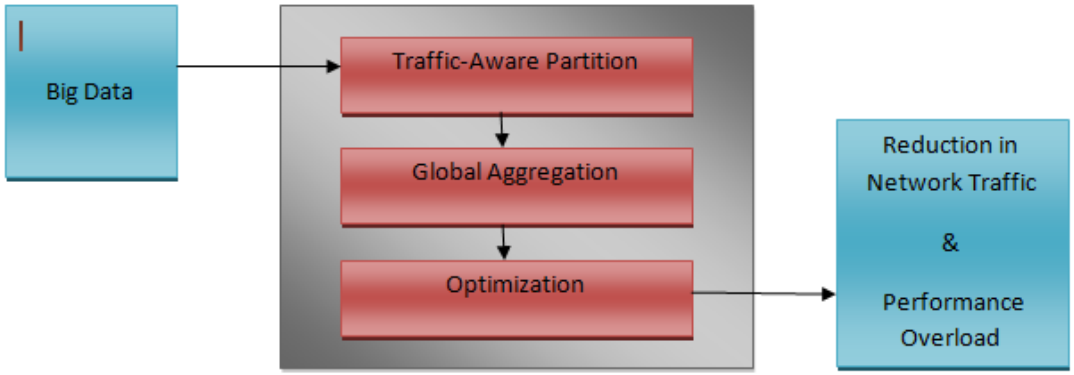

Figure 2: Overview of the proposed approach

As shown in Figure 2, it is evident that the proposed methodology includes defining and implementing two alternative algorithms for distributed data partitioning and data aggregation to reduce the network traffic further and improve MapReduce performance in distributed environment. The given big data is subjected to Map tasks and Reduce tasks. In the process there are proposed algorithms implemented for traffic-aware partitioning and global aggregation in distributed environment for optimization of MapReduce programming. The expected result of the proposed methodology includes reduction of network traffic and reduction of performance overhead.

CONCLUSIONS

The joint optimization of intermediate data partition and aggregation in Map Reduce to minimize network traffic cost for big data applications. Map Reduce to decrease network traffic cost for big data applications. We programming distributed algorithm to clear a problem on multiple machine in additional. We extended our algorithm to deal with map reduce job in a online manner. The simulation results demonstrate that our proposals can effectively reduce network traffic cost under various network settings.

References

1. KamaldeepSingh ,Sharath Chandra Guntuku , Abhishek Thakur and ChittaranjanHota . (2014). Big Data Analytics framework for Peer-to-Peer Botnet detection using Random Forests. *Computer Systems*. p1-10.
2. HuanKe, Peng Song Guo and MinyiGuo. (2015). On Traffic-Aware Partition and Aggregation in MapReduce for Big Data Applications. *IEEE*. p1-12.
3. KarthikKambatla, GiorgosKollias ,Vipin Kumar and AnanthGramaa. (2014). Trends in big data analytics. *Computer Systems*. p1-13.
4. AthiraSankar and SoumyaMurali. (2016). A Survey on Big Data Analytics in Mobile Cellular Networks. *IJESC*. 6 (11), p1-2.
5. RAJIV RANJAN, LIZHE WANG,PREM PRAKASH JAYARAMAN,KARAN MITRA and DIMITRIOS GEORGAKOPOULOS. (2017). Editorial Special issue on Big Data and Cloud of Things (CoT). *IEEE*. p1-3.
6. LautaroDolberg, Jer^ome Francois, Shihabur Chowdhury, Reaz Ahmed, RaoufBoutaba and Thomas Engel. (2016). Network Conguration and Flow Scheduling for Big Data Applications. *IEEE*. p1-21.
7. Chunwang Zhang, Ee-Chien Chang and Roland H.C. Yap. (2014). Tagged-MapReduce: A General Framework for Secure Computing with Mixed-Sensitivity Data on Hybrid Clouds. *IEEE/ACM*. p1-10.
8. Laizhong Cui, F. Richard Yu and Qiao Yan. (2016). When Big Data Meets Software-Defined Networking: SDN for Big Data and Big Data for SDN. *IEEE*. p1-9.
9. RamazanTerzi, DuyguSinanc and SerefSagiroglu. (2015). A survey on security and privacy issues in big data. *ICITST*. p1-7.
10. Y. Wang, W. Wang, C. Ma, and D. Meng, "Zput: A speedy data uploading approach for the hadoop distributed file system," in Cluster Computing (CLUSTER), 2013 IEEE International Conference on. IEEE, 2013, pp. 1–5.
11. F. Ahmad, S. Lee, M. Thottethodi, and T. Vijaykumar, "Mapreduce with communication overlap," pp. 608–620, 2013.
12. H.-c. Yang, A. Dasdan, R.-L. Hsiao, and D. S. Parker, "Map-reduce-merge: simplified relational data processing on large clus-ters," in Proceedings of the 2007 ACM SIGMOD international con-ference on Management of data. ACM, 2007, pp. 1029–1040.