Big Data Optimization: A Review

Stuti Mehla[1],

Dr. Suneet Kumar[2]

M.M Institute of Engineering and Technology,
Mullana, Ambala

Abstract

Big data is the large amount of data which is being generated from the emails, social networking conversations, videos, images, mobile phones, logs, online transactions, search queries etc. Big data has become a vital part of our daily life. As the amount of data is quite large storing it into traditional database software and using the regular tools for information retrieval has become a difficult task. Also it is difficult to capture, share, store, manage, form, visualize and analyze via traditional database software tools. So, there is need of optimized methods to manage this large amount of data. This paper focuses on the literature review of various methods of optimization in Big Data.

Keywords: *Big Data, Optimization, Optimization methods*

Introduction

According to Gartner "Big data is high-volume, high-velocity and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation."[1]

Big data refers to data volumes in the range of exabytes (1018) and beyond. Such volumes go beyond the capability of existing on-line storage space systems and processing systems. In future the rate at which the data and information is being created and collected is reaching to a range of exebyte/year, which will approach to zettabyte per year within a few years. The one bid aspect of big data is volume; the other aspects are value, velocity, variety and complexity. The term volume is the size of the data set, velocity indicates the speed of data in and out, and variety describes the range of data types and sources. It is right to say that Big Data will revolutionize many fields, including business, the scientific research, public administration, and so on[2].

We're clearly in the era of big data. Big data is characterized by millions of structured and unstructured data streams (high velocity), petabytes of historical data (high volume), and heterogeneous data types (high variety). Twitter produces an average of 6,000 tweets per second; however, the number expands to more than 140,000 during certain events (New Year's Eve, natural disasters, and so on). Such data explosions have led to the next grand challenge in computing: the big data problem, which is defined as the practice of collecting complex datasets so large that they're difficult to store, process, and interpret manually or using traditional data management applications (such as Microsoft Excel, relational databases, and data ware housing technologies).[3]

With diversified data needs, such as scientific experiments, telescopes, sensor networks, and high throughput instruments, the datasets increase at exponential rate as demonstrated in Fig. 1.

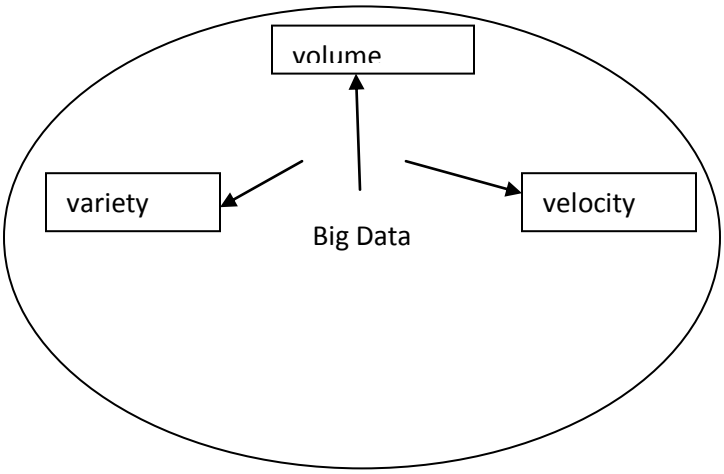

Figure 1.The three Vs of big data

The off-the-shelf methods and technologies that are used to store and analyze the data are not efficient enough to handle this large volume of data. The challenges start from data capture and data creation to data analysis and data visualization. In many instances, science is lagging behind the real world in the capability of discovering the precious information from massive volume of data. Based on precious knowledge, we need to develop and create new techniques and technologies to excavate Big Data and benefit our specified purposes. Big Data has changed the way that we adopt in doing businesses, managements and researches. Data-intensive science especially in data-intensive computing is coming into the world that aims to provide the tools that we need to handle the Big Data problems.

Processing Big Data: Analytics Challenges

However, in spite of the enormous possible of obtainable big data processing platforms, designing, developing, and implementing an most favorable big data scheduling platform that can assurance presentation (minimize response time or latency, maximize throughput) and fault tolerance (maximize availability or reliability) constraints at the same time is challenging, owing to several complexities and uncertainties. The first complexity is resource contention and interference. To minimize infrastructure cost, multiple big data processing frameworks are often hosted on shared cluster computing infrastructures. Sharing cluster resources among heterogeneous big data processing frameworks can also save the huge data migration costs involved in dataflow pipelines. However, such scenarios lead to resource contention and interference as co-located big data processing frameworks will compete for resources and interfere with each other's performance, making it extremely hard to meet performance requirements for real-time decision-making applications such as disaster management, stock purchasing, credit card fraud detection, online patient heart rate monitoring, and traffic management. Although these applications require short response times, current big data application scheduling platforms such as Apache Yarn10 and Mesos12 can't guarantee performance because of resource contention, lack of workload prioritization intelligence, and lack of coordinated scheduling capability across multiple big data processing frameworks. Big data processing frameworks must also deal with heterogeneous dataflows (for example, static, streaming, and transactional), heterogeneous data processing semantics (batch processing in Hadoop, continuous stream processing in Storm, and transaction processing in MySQL and Cassandra), and heterogeneous data types (such as unstructured data from Twitter, structured data from traditional SQL databases, and image data from video cameras) governed by varying data volume, data velocity, and query types[2,3,4].

To guarantee performance, scheduling platforms need to be able to predict the demands and behaviors of underlying frameworks so they can intelligently distribute and prioritize workloads. Further, it's not clear how such priorities can be preserved across multiple frameworks because dataflows are processed across a distributed platform. Third, big data processing frameworks must deal with uncertain resource needs. The Big data processing platforms usually extent distributed and heterogeneous software frameworks. These frameworks require heterogeneous and dynamic allocation and configuration of datacenter resources (for example, number and speed of CPUs; storage, cache, and RAM size; and network I/O bandwidth) to accommodate workload changes and to guarantee investigative outcome within an adequate delay. Determining an optimal resource configuration for big data processing frameworks is extremely hard because different big data applications have different performance constraints and complexity (3Vs). Current scheduling platforms, such as Apache Yarn and Mesos, entail considerable manual effort, where an administrator has to know in advance how many resources to allocate to each framework without over provisioning the available resource pool. Further, it's extremely hard to define and aggregate performance constraints of multiple frameworks to get a holistic view of end-to-end performance. Lack of robustness is another complexity. Big data scheduling platforms such as 10 Omega,11 YARN and Mesos12 can't handle fears arising from breakdown of datacenter resources, data overloading, malicious attacks, and network link congestion. Most of these scheduling platforms implement a simple failure model, in which a CPU resource instance hosting a big data processing framework (NoSQL or Hadoop, for example) is reconfigured (or restarts, fires a new instance, and so on) and doesn't respond to a certain number of network probes. Such reconfiguration is done without understanding the underlying causes of failures, such as disk failure, processor overload, malicious data, or malicious queries. Addressing these challenges requires careful consideration of numerous design and performance optimization tasks when developing robust and fault-tolerant big data processing solutions for those applications requiring real-time decision making such as disaster management, stock purchase, credit card fraud detection, and traffic management.

Literature Review

The algorithm proposed by the [5] is a parallelizable decomposition algorithm. It is a hybrid random/deterministic algorithm which aims to reduce the sum of possibly non smooth separable convex function and a possible non convex differentiable function. The framework proposed has the following features: a) it is capable of dealing non separable non-convex functions ; b) it is capable to use approximate solutions; c) it allows convergence of different types of variables ; d) it is parallel; and e) it can incorporate both first order or higher order information.

To identify the potential performance issues of Map Reduce programs a framework is proposed by [6]. It evaluates the performance by correlating the performance metrics from different layers. It uses predefined patterns to find out the potential issues. To reflect the improvement of the proposed model Terasort benchmark running on a 10-node Power7R2 cluster as a real case is used. The given framework is simple and effective.

The authors [7] have given methods for storage optimization and data loading in Hadoop cluster.
In order to gain better compression ratio to boost IO throughput the compression algorithms like ORC nad LZMA are implemented with HBase and Hadoop. The proposed optimization strategy is based on 8 factors. The main aim is to have an optimization strategy for better big data loading, which also include low serialization, and shuffle, decrease middle data landing, record split and byte array-oriented. The experimental results reveal that the method is much better for the big data load.

The six research challenges identified by [8] for research in big data are : i) Data privacy **(**Challenge 1) : Need of redefining the abstraction levels for auditing and access control for data platforms; ii) Approximate Results(Challenge 2): develop a querying procedure for estimated outcomes to make possible an order of magnitude quicker as compared to conventional query implementation; iii) Data Exploration To Enable Deep Analytics(Challenge 3): construct a atmosphere to facilitate data exploration for profound analytics; iv) Enterprise Data Enrichment With Web And Social Media(Challenge 4): recognize services that particular list of entities and their characteristics, returns enhancement of entities based on information in web and social media with suitably high accuracy and evoke; v) Query Optimization(Challenge 5): query optimization for data parallel platforms should be reorganized; vi) Performance Isolation For Multi-Tenancy (Challenge 6): identify a representation of presentation SLAs for multi-tenant data systems. Resource allocation techniques should be developed to support multi-tenancy.

The methods of big data processing from application and system aspects are given in [9]. MapReduce optimization methods and application areas are reviewed in the literature. The future issues and challenges in the big data and cloud computing are explored.

Conclusion

The big data processing and analysis is largely affected by the data load performance. The maximum time of big data process is consumed by the data processing and analysis. The methods of optimization are applied in various fields like engineering, economics, science etc. A lot of research work has been done to scale up the large--extent optimization by mutual co-evolutionary algorithms. Big data applications like ITSs and WSNs also need real time optimization. The other methods of optimization are data reduction and parallelization. The present big data loading algorithms and methods offer some improvement, but for a variety of big data environment, it is still not adequate for sufficient throughput. Further work will be done in this direction.

References

1. Beyer, Mark. "Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data". Gartner. Archived from the original on 10 July 2011. Retrieved 13 July 2011
2. Douglas and Laney, "The importance of 'big data': A definition," 2008.
3. Big data: science in the petabyte era, *Nature 455 (7209):1*, 2008.
4. X. Zhou, J. Lu, C. Li, and X. Du, "Big data challenge in the management perspective," *Communications of the CCF*, vol. 8, pp. 16–20, 2012.
5. Amir Daneshmand, Francisco Facchinei, Vyacheslav Kungurtsev, and Gesualdo Scutari ; Flexible Selective Parallel Algorithms for Big Data Optimization
6. Yan Li #Kun Wang #Qi Guo #Xin Li #Xiaochen Zhang xGuancheng Chen #Tao Liu #Jian ;Breaking the Boundary for Whole-System Performance Optimization of Big Data ;Li978-1-4799-1235-3/13/$31.00 ©2013 IEEE
7. Liping Zhang, Qi Chen, Kai Miao; A Compatible LZMA ORC-based Optimization for High Performance Big Data Load , IEEE International Congress on Big Data 2014
8. What Next? A Half-Dozen Data Management Research Goals for Big Data and the Cloud ; Surajit Chaudhuri ; May 21–23, 2012, Scottsdale, Arizona, USA. Copyright 2012 ACM
9 Big Data Processing in Cloud Computing Environments, Changqing Ji∗†, Yu Li‡, Wenming Qiu‡, Uchechukwu Awada‡, Keqiu Li; 1087-4089/12 $26.00 © 2012 IEEE

10 Bae B.J et.al."An Intrusive Analyzer for Hadoop Systems Based on Wireless Sensor Networks" Hindawi Publishing Corporation, International Journal of Distributed Sensor Networks Volume 2014, Article ID 196040.
11 GilLee J, Kang M "Geospatial Big Data: Challenges and Opportunities" 2214-5796/ 2015 Elsevier .
12 Jagadish V.H " Big Data and Science: Myths and Reality" 2214-5796/©2015 Elsevier .
13 X.Jin et.al. "Significance and Challenges of Big Data Research" 2214-5796/ 2015 Elsevier.
14 Huang .T et. Al." Promises and Challenges of Big Data Computing in Health Sciences" 2214-5796/ 2015 Elsevier.
15 Barkhordari, M. "ScaDiPaSi, An Effective Scalable and Distributable MapReduce-Base Method to Find Patient Similarity on Huge Healthcare Networks" 2214-5796/ 2015 Elsevier.