ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

BROADCASTING APPROACHES FOR BIG DATA – SURVEY PAPER

Divyadharsini S[#1], Krishneswari k[*2]

[#]*PG scholar & *Head of the Department & Tamilnadu College of Engineering*
Coimabatore, Tamilnadu,India

Abstract -*In the information era, huge amounts of data have become available on hand to decision makers. Big data refers to datasets that are not only big, but also high in variety and velocity, which makes them difficult to handle using traditional tools and techniques. Due to the rapid growth of such data, solutions need to be studied and provided in order to handle and extract value and knowledge from these datasets. Furthermore, decision makers need to be able to gain valuable insights from such varied and rapidly changing data, ranging from daily transactions to customer interactions and social network data. Such value can be provided using big data analytics, which is the application of advanced analytics techniques on big data. This paper aims to analyze some of the different analytics methods and tools which can be applied to big data, as well as the opportunities provided by the application of big data analytics in various decision domains.*

Keywords -*Variety, Velocity, Analytics*

I.INTRODUCTION

Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate to deal with them. Challenges include analysis, capture, data duration, search, sharing, storage, transfer, visualization, querying, updating and information privacy. The term "big data" often refers to the use of predictive analytics, user behavior analytics, or certain other advanced data analytics methods that extract value from data, and seldom to a particular size of data set. There is little doubt that the quantities of data now available are indeed large, but that's not the most likely characteristic of this new data ecosystem.

Big data is really critical to our life and its emerging as one of the most important technologies in modern world. Big data technologies are important in providing more accurate analysis, which may lead to more concrete decision-making resulting in greater operational efficiencies, cost reductions, and reduced risks for the business. The generation of new data has drastically increased. More applications are being built and they are generating more data at extraordinary rate. Then, there is Internet of Things, which has truly brought us into the data age. Its ability to integrate various aspects of Big Data solutions seamlessly such as Streaming.

The flexibility to handle various data formats is available through implying JSON format and extracting requisite information from the data available. Hadoop cluster, Hadoop's

ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

security are also covered by it. As these tools are designed for big data processing, data replication and reliability are provided by the infrastructure, thus enabling the engineers to focus on building the business proposition. Integrating these services with the pipeline will make the system more usable and more versatile.

Analysis of data sets can find new correlations to "spot business trends, prevent diseases, and combat crime and so on". Scientists, business executives, practitioners of medicine, advertising and governments alike regularly meet difficulties with enormous datasets in areas including Internet search, finance, urban informatics, and business informatics. Scientists encounter limitations in e-Science work, including meteorology, genomics, connectomics, complex physics simulations, biology and environmental research.

Data sets grow quickly in part because they are increasing together by cheap and numerous information sensing mobile devices, aerial (remote sensing), software logs, cameras, microphones, radio-frequency identification (RFID) readers and wireless sensor networks. The world's technological per capita capacity to store data has roughly doubled every 40 months since the 1980s; as of 2012, every day 2.5 Exabyte (2.5×10^{18}) of data is generated.[1] One question for large enterprises is determining who should own big-data initiatives that affect the entire organization.

II. RELATED WORK

The surveys based on the application of big data for storing, Map Reduce, monitoring and managing data transfer. This section discusses about various methodology for reducing time complexity in data transfer.

In [1] authors have introduced the advances in digital sensors, communications, computation, and storage have created enormous collections of data, capturing information of value to business, science, government, and society. For example, search engine companies such as Google, Yahoo!, and Microsoft have created an entirely new business by capturing the information freely available on the World Wide Web and providing it to people in useful ways. These companies collect trillions of bytes of data every day and continually add new services such as satellite images, driving directions, and image retrieval. The societal benefits of these services are immeasurable, having transformed how people find and make use of information on a daily basis.

A new form of computer systems, consisting of thousands of "nodes," each having several processors and disks, connected by high-speed local-area networks, has become the chosen hardware configuration for data-intensive computing systems. These clusters provide both the storage capacity for large data sets, and the computing power to organize the data, to analyze it, and to respond to queries about the data from remote users. Compared with traditional high-performance computing (e.g., supercomputers), where the focus is on maximizing the raw computing power of a system, cluster computers are designed to maximize the reliability and efficiency with which they can manage and analyze very large data sets.

In [2] author proposed that Map Reduce is a programming model and an associated implementation for processing and generating large data sets. Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key. Programs written in this functional style are automatically parallelized and executed on a large cluster of commodity machines. The run-time system takes care of the details of partitioning the input data, scheduling the program's execution across a set of machines, handling machine failures, and managing the required inter-machine communication. This allows programmers without any experience with parallel and distributed systems to easily utilize the resources of a large distributed system. Our implementation of Map Reduce runs on a large

ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

cluster of commodity machines and is highly scalable in a typical Map Reduce computation processes many terabytes of data on thousands of machines. Programmers and the system easy to use and also hundreds of Map Reduce pro-grams have been implemented and upwards of one thousand Map Reduce jobs are executed on Google's clusters every day.

In [3] authors introduced big table in distributed storage system for managing structured data that is designed to scale to a very large size petabytes of data across thousands of commodity servers. In some projects used Google store data in big table, including web indexing, Google Earth, and Google Finance. These applications place very different demands on big table, both in terms of data size (from URLs to web pages to satellite imagery) and latency requirements (from backend bulk processing to real-time data serving). Despite these varied demands, big table has successfully provided a flexible, high-performance solution for all of these Google products. In this paper, described about the simple data model provided by big table, which gives clients dynamic control over data layout and format, and also describe the design and implementation of big table.

Recent years big table have designed, implemented, and deployed a distributed storage system for managing structured data at Google called big table. Big table is designed to reliably scale to petabytes of data and thousands of machines. Big table has achieved several goals: wide applicability, scalability, high performance, and high availability. Big table is used by more than sixty Google products and projects, including Google Analytics, Google Finance, Orkut, Personalized Search, Writely, and Google Earth.

In [4] authors introduced the Networks of workstations (NOWs), which are generally composed of autonomous compute elements networked together, are an attractive parallel computing platform since they offer high performance at low cost. They are three primary factors of result based on load imbalances, that is unequal load (compute or communication) assignment to equally-powerful compute no des, unequal resources at compute no des, and multiprogramming. These load imbalances result in idle waiting time on co operating processes that need to synchronize or communicate data.

Additional waiting time may result due to local scheduling decisions in a multi-programmed environment. The combined approach of compile-time analysis, run-time load distribution, and operating system scheduler cooperation for improved utilization of available resources in an autonomous now. The techniques are proposed allow efficient resource utilization by taking into consideration all three causes of load imbalance in addition to locality of access in the process of load distribution. The resulting adaptive load distribution and co operative scheduling system allows applications to take advantage of parallel resources when available by providing better performance than when the loaded resources are not used at all.

In [5] authors proposed that Cluster computing applications like Map Reduce and Dryad transfer massive amounts of data between their computation stages. These transfers can have a significant impact on job performance, accounting for more than 50% of job completion times. Despite this impact, there has been likely little work on optimizing the performance of these data transfers, with networking researchers mainly focusing on per flow traffic management. This limitation by proposing a global management architecture and a set of algorithms that (1) improve the transfer times of common communication patterns, such as broadcast and shuffle, and (2) allow scheduling policies at the transfer level, such as prioritizing a transfer over other transfers.

It can be define a transfer as the set of all flows transporting data between two stages of a job. In frameworks like Map Reduce and Dryad, a stage cannot complete (or sometimes even start) before it receives all the data from the previous stage. Thus, the job running time depends on the

ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

time it takes to complete the entire transfer, rather than the duration of individual flows comprising it.

In [6] authors have introduced Cloud computing centers face the key challenge of provisioning diverse virtual machine instances in an elastic and scalable manner. They have performed an analysis of VM instance traces collected at six production data centers during four months. One key finding is that the number of instances created from the same VM image is relatively small at a given time and thus conventional file-based p2p sharing approaches may not be effective. Based on the understanding that different VM image files often have many common chunks of data, propose a chunk-level Virtual machine image Distribution Network (VDN). Our distribution scheme takes advantage of the hierarchical network topology of data centers to reduce the VM instance provisioning time and also to minimize the overhead of maintaining chunk location information. Evaluation shows that VDN achieves as much as 30–80x speed up for large VM images under heavy traffic.

Cloud computing enables users to access compute resources on demand without the burden of owning, managing, and maintaining the resources. To support Infrastructure as a Service (IaaS), most cloud platforms use virtualized data centers. Typically, a cloud data center maintains a catalog that lists available virtual machine (VM) images. Those images may contain only the large operating system such as Linux Red Hat or Windows, include popular applications such as database management systems, or even be created by users.

In [7] authors investigate about the power efficient broadcast routing problem over heterogeneous wireless ad hoc or sensor networks where network nodes have heterogeneous capability. The network links between pairs of nodes can no longer be modeled as symmetric or bidirectional. It show that, while most previous power efficient algorithms work in this setting with minor modifications, they are not designed to exploit such asymmetric constraints, used suitable algorithm which takes into account of the constraints and yet most power-efficient among all known algorithms.

In many real situations, RF transceivers take on various level of capabilities in terms of maximum transmission range, computational processing, and omnidirectional versus directional antennas, etc. Even transceivers that are supposed to meet certain common specifications may have different capabilities, because internal implementation details will vary from one manufacturer to another. Not to mention such scenario, certain networks such as cellular networks are inherently multi-tiered. In case of wireless sensor networks (WSN), there exist many data-gathering stations which can also serve as gateway nodes to infrastructure networks.

In [8] authors introduced two of the fundamental problems in peer-to-peer (P2P) streaming are as follows: what is the maximum streaming rate that can be sustained for all receivers, and what peering algorithms can achieve close to this maximum? These problems of computing and approaching the P2P streaming capacity are often challenging because of the constraints imposed on overlay topology. In this paper, mainly focus on the limit of P2P streaming rate under node degree bound, i.e., the number of connections a node can maintain is upper bounded. It shows that the streaming capacity problem under node degree bound is NP-Complete in general. Then, for the case of node out-degree bound, through the construction of a "Bubble algorithm", it show that the streaming capacity is at least half of that of a much less restrictive. Then, develop a "Cluster-Tree algorithm" that provides a probabilistic guarantee of achieving a rate close to the maximum rate achieved under no degree bound constraint, when the node degree bound is logarithmic in network size. The activeness of these algorithms in approaching the capacity limit is demonstrated in simulations using uplink bandwidth statistics of Internet hosts. Both analysis and numerical

ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

experiments show that peering in a locally dense and globally sparse manner achieves near-optimal streaming rate if the degree bound is at least logarithmic in network size.

In [9] authors consider the classical problem of broadcasting a large message at an optimal rate in a large scale distributed network. Main approach is the set of participating nodes can be split into two parts: "green" nodes that stay in the open-Internet and "red" nodes that lie behind firewalls or NATs. Two red nodes cannot communicate directly, but rather need to use a green node as a gateway for transmitting a message. In this context, both maximizing the throughput (i.e. the rate at which nodes receive the message) and minimizing the degree at the participating nodes, i.e. the number of TCP connections handle simultaneously. The flow graph using both cyclic and acyclic solutions. In the cyclic case, our main contributions are a closed form formula for the optimal cyclic throughput and the proof that the optimal solution may require arbitrarily large degrees. In the acyclic case, it is possible to achieve the optimal throughput with low degree. Then also prove that a worst case ratio between the optimal acyclic and cyclic throughput and show through simulations that this ratio is on average very close to 1, which makes acyclic solutions efficient both in terms of the throughput and the number of connections.

In [10] an author focused mainly on Network of workstation (NOW) is a cost effective alternative to massively parallel supercomputers. As commercially available off the shelf processors become cheaper and faster, it is now possible to build a PC or workstation cluster that provides high computing power within a limited budget. However, a cluster may consist of different types of processors and this heterogeneity within a cluster complicates the design of efficient collective communication protocols.

This paper shows that a simple heuristic called fastest-node-first (FNF) [3] is very effective in reducing broadcast time for heterogeneous cluster systems. Despite the fact that FNF heuristic fails to give the optimal broadcast time for a general heterogeneous network of workstation, we prove that FNF always gives the optimal broadcast time in several special cases of clusters. Based on these special case results, show that FNF is an approximation algorithm that guarantees a competitive ratio. From these theoretical results we also derive techniques to speed up the branch and bound search for the optimal broadcast schedule in HNOW.

III. CONCLUSION

In big data analytics, the high dimensionality and the streaming nature of the income data aggravate great computational challenges in data mining. Big data grows continually with fresh data are being generated at all times, hence it requires an incremental computation approach which is able to monitor large scale of data dynamically. From this analysis, it is concluded that minimum time complexity of novel pipelining approaches.

REFERENCE

I. Randal E. Bryant, Randy H. Katz and Edward D. Lazowska, Big-Data Computing: Creating revolutionary breakthroughs in commerce, science, and society, 1998.

II. Jeffrey Dean and Sanjay Ghemawat, Mapreduce: Simplied Data Processing on Large Clusters, 1989, vol. 61.

III. Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber, Bigtable: A Distributed Storage System for Structured Data., vol. 20, pp. 569–571, Nov. 1999.

IV. Umit Rencuzogullari, Sandhya Dwarkadas ,Dynamic Adaptation to Available Resources for Parallel Computing in an Autonomous Network of Workstations

ISSN (online): 2250-141X
Vol. 6 Issue 4, October 2016

V. Mosharaf Chowdhur y, Matei Zahar ia, Justin Ma, Michael I. Jordan, Ion Stoica ,Managing Data Transfers in Computer Clusters with Orchestra, Sept. 16, 1997.

VI. Chunyi Peng and Minkyong Kim, Zhe Zhang, Hui Lei ,VDN: Virtual Machine Image Distribution Network for Cloud Data Centers

VII. Intae Kang and Radha Poovendran, Broadcast with Heterogeneous Node Capability

VIII. Shao Liu , Minghua Chen , Sudipta Sengupta , Mung Chiang , Jin Li, and Phil A. Chou ,P2P Streaming Capacity under Node Degree Bound

IX. Olivier Beaumont, Nicolas Bonichon, Lionel Eyraud-Dubois, P. Uznanski ,Broadcasting on Large Scale Heterogeneous Platforms with connectivity artifacts under the Bounded Multi-Port Model

X. Pangfeng Liu ,Broadcast Scheduling Optimization for Heterogeneous Cluster Systems

XI. S. Khuller and Y.-A. Kim, "Broadcasting in heterogeneous networks," Algorithmic, vol. 48, no. 1, pp. 1–21, Mar. 2007.

XII. J. Mundinger, R. Weber, and G. Weiss, "Optimal scheduling of peer-to-peer file dissemination," J. Scheduling, vol. 11, no. 2, pp. 105–120, 2008.

XIII. L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "P2P streaming capacity under node degree bound," IEEE 30th Int. Conf. Dist. Comput. Syst., pp. 587–598, 2010.

XIV. O. Beaumont, L. Eyraud-Dubois, and S. K. Agrawal, "Broadcasting on large scale heterogeneous platforms under the bounded multi-port model," in Proc. IEEE Int. Symp. Parallel Distrib. Process. Symp., 2010, pp. 1–11.

XV. S. M. Hedetniemi, S. T. Hedetniemi, and A. Liestman, "A survey of gossiping and broadcasting in communication networks," Networks, vol. 18, pp. 319–349, 1988.

XVI. P. Liu, "Broadcast scheduling optimization for heterogeneous cluster systems," J. Algorithms, vol. 42, no. 1, pp. 129–136, Jan. 2002.

XVII. K. Wang, J. Li, and L. Pan, "Fast file dissemination in Peer-to-peer networks with upstream bandwidth constraint," Future Generation Comput. Syst., vol. 26, pp. 986–1002, Jul. 2010.

XVIII. K.-S. Goetzmann, T. Harks, M. Klimm, and K. Miller, "Optimal file distribution in peer-to-peer networks," Proc. 22nd Int. Symp. Algorithms Comput. 2011, pp. 210–219.

XIX. M. Deshpande, N. Venkatasubramanian, and S. Mehrotra, "Heuristics for flash-dissemination in heterogeneous networks, Proc.13thInt.Conf.HighPerformanceCompu t" 2006, pp. 607– 618.

XX. B. Cohen, "Incentives build robustness in bit torrent," Proc. ACM Workshop Econ. Peer-to-Peer Syst., 2003, pp. 264–267.