© 2016 IJSRSET | Volume 2 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Effective Distribution of Large Scale Datasets Clustering Based on Map Reduce

Vignesh Kumar V, Yuvaraj R, Anusha C

Department of Computer Science and Engineering Dhanalakshmi College of Engineering, Chennai, Tamilnadu, India

ABSTRACT

Big data is a broad term for data sets so large or complex that traditional data processing applications are inadequate. Challenge sinclude analysis, capture, data curation,search, sharing, storage, transfer, visualization, querying andinformation privacy. The term often refers simply to the use of predictive analytics or certain other advanced methods to extract value from data, and seldom to a particular size of data set. Accuracy in big data may lead to more confident decision making, and better decisions can result in greater operational efficiency, cost reduction and reduced risk .Data that are generated from variety of sources with massive volumes, high rates, and different data structure are collectively known as Big Data. MapReduce framework was built as a parallel distributed programming model to process such large-scale datasets effectively and efficiently. Big Data software analysis solutions were implemented on MapReduce framework, describing their datasets structures and how they were implemented with MongoDB as NoSQL Database. NoSQL encompasses a wide variety of different database technologies that were developed in response to the demands presented in building modern applications. MongoDB stores data using a flexible document data model. Documents contain one or more fields, including arrays, binary data and sub-documents. Thus, the demand for building a service stack to distribute, manage, and process massive data sets has risen drastically. In this paper, we investigate the Big Data Broadcasting problem for a single source node to broadcast a big chunk of data to a set of nodes with the objective of minimizing the maximum completion time. Big-data computing is a new critical challenge for the ICT industry. Engineers and researchers are dealing with data sets of petabyte scale in the cloud computing paradigm.

Keywords: Clustering, Datasets, Map Reduce, Big Data, ICT, MongoDB, NoSQL Database, ERP, LSBT, LHC

I. INTRODUCTION

A. Scope of the Project

- The use of Big Data within Organization processes allows efficiencies in terms of cost, productivity, and innovation.
- This process does not come without its flaws.
- Data analysis often requires multiple parts of Organization to work in collaboration and create new and innovative processes to deliver the desired outcome.

Problems in Existing System

- Organizations today are confronted with the challenge and the opportunity of data growing at unprecedented rates.
- This data comes from numerous sources – ERP systems, Data Warehouses, Website logs, Web Services, Social Media, Mobile devices, Sensors, etc. - in various forms - Structured, Semi-structured and Unstructured.
- “Big Data” is the catch all phrase for this rapidly changing field. Big Data analytics has the potential to provide great insights and opportunities to organizations in the areas of consumer behavior, marketing, fraud detection and customer service.

IJSRSET1622136 | Received : 26 March 2016 | Accepted : 05 April 2016 | March-April 2016 [(2)2: 505-508]

- With the right technical architecture, true real-time decisions are enabled providing organizations with heightened agility.
- While most organizations recognize the importance and benefits of Big Data analytics, there are challenges arising from the nature of Big Data and limitations of existing technologies that need to be considered.

Disadvantages

- LSBT stores extremely large files containing record-oriented data.
- It does not split large data files.
- The size of the files and the number of replications are not configurable.

II. METHODS AND MATERIAL

A. Proposed System

- In this project, enterprise system has a centralized server to store and process data.
- The following illustration depicts a schematic view of a traditional enterprise system. Traditional model is certainly not suitable to process huge volumes of scalable data and cannot be accommodated by standard database servers.
- Moreover, the centralized system creates too much of a bottleneck while processing multiple files simultaneously.
- The data will be stored and retrieved from database within the group of organization. In all the branch they can store and retrieve the data.
- Here the data will stored in the common database and the values will be retrieved using MapReduce algorithm.

Advantages:

- MapReduce algorithm stores files containing record-oriented data.
- It splits large data files into chunks of 64 MB, and replicates the chunk across three different nodes in the cluster.
- The size of the chunks and the number of replications are configurable.

B. Future Work

- We plan to examine whether the techniques used in HM can be extended to support unstructured data.
- The main challenge is still the limitation of storage. Similar to the relational case, we need a layout plan to guide us which part of data should be maintained to maximize the performance. The first step is to discover query patterns.
- Three most popular workloads on unstructured data are keyword based queries, data mining tasks and machine learning tasks.

C. Technical Terms

- **Information and Communication Technology(ICT)**

CT (information and communications technology - or technologies) is an umbrella term that includes any communication device or application, encompassing: radio, television, cellular phones, computer and network hardware and software, satellite systems and so on, as well as the various services and applications associated with them, such as videoconferencing and distance learning.

- **Large Hadron Collider (LHC)**

The Large Hadron Collider (LHC) is the world's largest and most powerful particle collider, the largest, most complex experimental facility ever built, and the largest single machine in the world.

- **Big Data Management**

The goal of big data management is to ensure a high level of data quality and accessibility for business intelligence and big data analytics applications. Corporations, government agencies and other organizations employ big data management strategies to help them contend with fast-growing pools of data, typically involving many terabytesor even petabytes of information saved in a variety of file formats. Effective big data management helps companies locate valuable information in large sets of unstructured data and semi-structured data from a variety of sources, including call detail records, system logs and social media sites.

- **Sorting**

Sorting in MapReduce is originally intended for sorting of the emitted key-value pairs by key, but there exist techniques that leverage the implementation specifics to achieve sorting by values.

III. RESULTS AND DISCUSSION

ALGORITHM DETAILS

The MapReduce algorithm contains two important tasks, namely Map and Reduce.

- The map task is done by means of Mapper Class
- The reduce task is done by means of Reducer Class.

Mapper class takes the input, tokenizes it, maps and sorts it. The output of Mapper class is used as input by Reducer class, which in turn searches matching pairs and reduces them. MapReduce implements various mathematical algorithms to divide a task into small parts and assign them to multiple systems. In technical terms, MapReduce algorithm helps in sending the Map & Reduce tasks to appropriate servers in a cluster.

These mathematical algorithms may include the following

- Sorting
- Searching
- Indexing
- TF-IDF

Sorting

Sorting is one of the basic MapReduce algorithms to process and analyze data. MapReduce implements sorting algorithm to automatically sort the output key-value pairs from the mapper by their keys.

- Sorting methods are implemented in the mapper class itself.
- In the Shuffle and Sort phase, after tokenizing the values in the mapper class, the **Context** class (user-defined class) collects the matching valued keys as a collection.
- To collect similar key-value pairs (intermediate keys), the Mapper class takes the help of **RawComparator** class to sort the key-value pairs.
- The set of intermediate key-value pairs for a given Reducer is automatically sorted by Hadoop to form key-values (K2, {V2, V2, …}) before they are presented to the Reducer.

Searching

Searching plays an important role in MapReduce algorithm. It helps in the combiner phase (optional) and in the Reducer phase.

Indexing

Normally indexing is used to point to a particular data and its address. It performs batch indexing on the input files for a particular Mapper.

The indexing technique that is normally used in MapReduce is known as**inverted index.** Search engines like Google and Bing use inverted indexing technique.

TF-ID TF-IDF is a text processing algorithm which is short for Term Frequency − Inverse Document Frequency. It is one of the common web analysis algorithms. Here, the term 'frequency' refers to the number of times a term appears in a document.

IV. CONCLUSION

The promise and potential of big data needs to be matched by a considered approach to collection, storage, licensing and use. Without a well thought through data strategy, remedies for misuse may be hard to find. Traditional copyright protection is unlikely to assist and contract and confidential information remedies are likely to be far more significant. If competitors benefit from observing publically available data, remedies may be particularly very difficult to find. As a result, the battle against scrapers of data will often be more a battle of technologies (such as IP address blocking) than successful assertion of legal remedies.

V. REFERENCES

[1] R. E. Bryant, R. H. Katz, and E. D. Lazowska, “Big-data computing: Creating revolutionary break throughs in commerce, science, and society,” In Computing Research Initiatives for the 21st Century., 2008.

[2] A. Szalay and J. Gray, “2020 computing: Science in an exponential world,” Nature 440, 413-414, March, 2006.

[3] G. Brumfiel, “High-energy physics: Down the petabyte highway,” Nature 469, 282-283 January, 2011.

[4] J. Dean and S. Ghemawat, “Mapreduce: Simplified data processing on large clusters,” Proc. of Operating Systems Design and Implementation (OSDI), 2004.

[5] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Bur- rows, T. Chandra, A. Fikes, , and R. E. Gruber, “Bigtable: A distributed storage system for structured data,” Proc. of Operating Systems Design and Implementation (OSDI), 2006.

[6] W. D. Hillis and G. L. Steele, Jr., “Data parallel algorithms,” Commu- nications of the ACM, vol. 29, pp. 1170–1183, December 1986.

[7] U. Rencuzogullari and S. Dwarkadas, “Dynamic adaptation to available resources for parallel computing in an autonomous network of workstations,” Proc. of ACM SIGPLAN PPoPP, 2001.

[8] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica, “Man- aging data transfers in computer clusters with orchestra,” Proc. of ACM Special Interest Group on Data Communication (SIGCOMM), pp. 98–109, 2011.

[9] D. Nukarapu, B. Tang, L. Wang, and S. Lu, “Data replication in data intensive scientific applications with performance guarantee,” IEEE Transactions on Parallel and Distributed Systems, aug. 2011.

[10] C. Peng, M. Kim, Z. Zhang, and H. Lei, “Vdn: Virtual machine image distribution network for cloud data centers,” Proc. of IEEE International Conference on Computer Communications (INFOCOM), 2012.

[11] S. Khuller and Y.-A. Kim, “Broadcasting in heterogeneous networks,” Algorithmica, vol. 48, no. 1, Mar. 2007.

[12] J. Mundinger, R. Weber, and G. Weiss, “Optimal scheduling of peer-to- peer file dissemination,” Journal of Scheduling, vol. 11, no. 2, 2008.