Scientific Journal Impact Factor (SJIF): 1.711

e-ISSN: 2349-9745
p-ISSN: 2393-8161

International Journal of Modern Trends in Engineering and Research
www.ijmter.com

A Survey on Big Data Mining Challenges

Irin Ani John[1], Tintu Alphonsa Thomas[2]

[1,2]*Department Of Computer Science And Engineering*
Amal Jyothi College Of Engineering
Kanjirappally , Kottayam, India

Abstract— Big Data is the new technology or science to make the well informed decision in business or any other science discipline with huge volume of data from new sources of heterogeneous data. . Such new sources include blogs, online media, social network, sensor network, image data and other forms of data which vary in volume, structure, format and other factors. Big Data applications are increasingly adopted in all science and engineering domains, including space science, biomedical sciences and astronomic and deep space studies. The major challenges of big data mining are in data accessing and processing, data privacy and mining algorithms. This paper includes the information about what is big data, data mining with big data, the challenges in big data mining and what are the currently available solutions to meet those challenges.

Keywords—Big Data, Big Data Mining Algorithms, MPI, MapReduce, Dryad.

I. INTRODUCTION

Recent years have witnessed a dramatic increase in our ability to collect data from different sensors, instruments, in different formats from independent or connected applications. This data volume has outgrown our ability to process, analyse, store and understand these datasets. Consider the internet data, the web pages indexed by Google were around one million in 1998, but quickly reached 1 billion in 2000 and exceeded 1 trillion in 2008. This rapid expansion is accelerated by the dramatic increase in acceptance of social networking application such as Facebook, Twitter, etc, that allows user to create contents freely and amplify the already huge web volume.

Furthermore with mobile phones becoming the sensory gateway to get real time data in people from different aspects, the vast amount of data that mobile carried can potentially process to improve our daily life has significantly outpaced our past call data record based processing for billing purposes only. People and devices (from home, to cars, to buses, railway stations and airport) are all loosely connected. Trillions of such connected components will generate a huge data ocean, and valuable information must be discovered from the data to help improve quality of life and make our world a better place.

II. CHARACTERISTICS OF BIG DATA

The major characteristics can be defined by HACE theorem:[1] Big Data starts with large-volume ***h***eterogeneous, ***a***utonomous sources with distributed and decentralized control and seeks to explore ***c***omplex and ***e***volving relationships among data.

These characteristics make it an extreme challenge for discovering useful knowledge from the Big Data. The heterogeneous characteristics of the Big Data is because different information collectors prefer their own schemata or protocols for data recording, and the nature of different applications

also results in diverse data representations. Being autonomous, each data source is able to generate and collect information without involving (or relying on) any centralised control. While the volume of the data grows, the complexity and the relationships underneath the data will also increases exponentially.

III. CHALLENGES OF BIG DATA

With the development of Internet services, indexes and queried contents were rapidly growing. Therefore, search engine companies had to face the challenges of handling such big data. Google created GFS[3] and MapReduce programming models[2] to cope with the challenges brought about by data management and analysis at the Internet scale. Some obstacles[4][5][6] in the development of big data applications are :

A. *Data representation:* Many datasets have certain levels of heterogeneity in type, structure, semantics, organization, granularity, and accessibility. Data representation aims to make data more meaningful for computer analysis and user interpretation. Efficient data representation shall reflect data structure, class, and type, as well as integrated technologies, so as to enable efficient operations on different datasets.

B. *Redundancy reduction and data compression:* Generally, there is a high level of redundancy in datasets. Redundancy reduction and data compression is effective to reduce the indirect cost of the entire system on the premise that the potential values of the data are not affected. For example, most data generated by sensor networks are highly redundant, which may be filtered and compressed at orders of magnitude.

C. *Data life cycle management:* Compared with the relatively slow advances of storage systems, pervasive sensing and computing are generating data at unprecedented rates and scales. We are confronted with a lot of pressing challenges, one of which is that the current storage system could not support such massive data.

D. *Analytical mechanism:* The analytical system of big data shall process masses of heterogeneous data within a limited time. However, traditional RDBMSs are strictly designed with a lack of scalability and expandability, which could not meet the performance requirements. Non-relational databases have shown their unique advantages in the processing of unstructured data and started to become mainstream in big data analysis. Even so, there are still some problems of non-relational database in their performance and particular applications. We shall find a compromising solution between RDBMSs and non-relational databases. For example, some enterprises have utilized a mixed database architecture that integrates the advantages of both types of databases.

E. *Data Confidentiality:* Most big data service providers or owners at present could not effectively maintain and analyze such huge datasets because of their limited capacity. They must rely on professionals or tools to analyze such data. These data contains details of the lowest granularity and some sensitive information such as credit card numbers. Therefore, analysis of big data may be delivered to a third party for processing only when proper preventive measures are taken to protect such sensitive data, to ensure its safety.

F. *Expendability and scalability:* The analytical system of big data must support present and future datasets. The analytical algorithm must be able to process increasingly expanding and more complex datasets.

The major challenges[1] of data mining with big data are in Big data mining platform,in information sharing and data privacy and Mining algorithms.

A. Big Data Mining Platform

The mining platform is needed to have efficient access to the data and computing processors.For Big Data mining, because data scale is far beyond the capacity that a single personal computer (PC) can handle, a typical Big Data processing framework will rely on cluster computers with a high-performance computing platform, with a data mining task being deployed by running some parallel programming tools, such as MapReduce or Enterprise Control Language (ECL),DryadLINQ[7], on a large number of computing nodes.

B. Information Sharing and Data Privacy

A real-world concern is that Big Data applications are related to sensitive information, such as banking transactions and medical records. Simple data exchanges or transmissions do not resolve privacy concerns. To protect privacy, two common approaches are to 1) restrict access to the data, such as adding certification or access control to the data entries, so sensitive information is accessible by a limited group of users only, and 2) anonymize data fields such that sensitive information cannot be pinpointed to an individual record[8]. The most common k-anonymity privacy measure is to ensure that each individual in the database must be indistinguishable from k-1 others. Common anonymization approaches are to use suppression, generalization, perturbation, and permutation to generate an altered version of the data.

C. Mining Algorithms

As Big Data applications are featured with autonomous sources and decentralized controls, aggregating distributed data sources to a centralized site for mining is systematically prohibitive due to the potential transmission cost and privacy concerns. A Big Data mining system has to enable an information exchange and fusion mechanism to ensure that all distributed sites can work together to achieve a global optimization goal. Spare, uncertain, and incomplete data are defining features for Big Data applications. The rise of Big Data is driven by the rapid increasing of complex data and their changes in volumes and in nature. Inspired by the above challenges, many data mining methods have been developed to find interesting knowledge from Big Data with complex relationships and dynamically changing volumes. This has also spawned new computer architectures for real-time data-intensive processing, such as the open source Apache Hadoop[9] project that runs on high-performance clusters.

IV. BIG DATA ANALYTIC METHODS

At present, the main processing methods of big data are shown as follows.

A. *Bloom Filter*: Bloom Filter consists of a series of Hash functions. The principle of Bloom Filter is to store Hash values of data other than data itself by utilising a bit array, which is in essence a bitmap index that uses Hash functions to conduct lossy compression storage of data. It has such advantages as high space efficiency and high query speed, but also has some disadvantages in misrecognition and deletion.

B. *Hashing*: it is a method that essentially transforms data into shorter fixed-length numerical values or index values. Hashing has such advantages as rapid reading, writing, and high query speed, but it is hard to find a sound Hash function.

C. *Index*: index is always an effective method to reduce the expense of disk reading and writing, and improve insertion, deletion, modification, and query speeds in both traditional relational databases that manage structured data, and other technologies that manage semi-structured and unstructured data. However, index has a disadvantage that it has the additional cost for storing index files which should be maintained dynamically when data is updated.

D. *Triel*: also called trie tree, a variant of Hash Tree. It is mainly applied to rapid retrieval and word frequency statistics. The main idea of Triel is to utilise common prefixes of character strings to reduce comparison on character strings to the greatest extent, so as to improve query efficiency.

E. *Parallel Computing*: compared to traditional serial computing, parallel computing refers to simultaneously utilising several computing resources to complete a computation task. Its basic idea is to decompose a problem and assign them to several separate processes to be independently completed, so as to achieve coprocessing. Presently, some classic parallel computing models include MPI (Message Passing Interface), MapReduce, and Dryad .

Although the parallel computing systems or tools, such as MapReduce or Dryad, are useful for big data analysis, they are low levels tools that are hard to learn and use. Therefore, some high-level parallel programming tools or languages are being developed based on these systems. Such high-level languages include Sawzall, Pig, and Hive used for MapReduce, as well as Scope and DryadLINQ used for Dryad.

MapReduce: MapReduce is a simple but powerful programming model for large-scale computing using a large number of clusters of commercial PCs to achieve automatic parallel processing and distribution. In MapReduce, computing model only has two functions, i.e., Map and Reduce, both of which are programmed by users. The Map function processes input key-value pairs and generates intermediate key-value pairs. Then,MapReduce will combine all the intermediate values related to the same key and transmit them to the Reduce function, which further compress the value set into a smaller set. MapReduce has the advantage that it avoids the complicated steps for developing parallel applications, e.g., data scheduling, fault-tolerance, and inter-node communications. The user only needs to program the two functions to develop a parallel application. The initial MapReduce framework did not support multiple datasets in a task, which has been mitigated by some recent enhancements.

Over the past decades, programmers are familiar with the advanced declarative language of SQL, often used in a relational database, for task description and dataset analysis. However, the succinct MapReduce framework only provides two nontransparent functions, which cannot cover all the common operations. Therefore, programmers have to spend time on programming the basic functions, which are typically hard to be maintained and reused. In order to improve the programming efficiency, some advanced language systems have been proposed, e.g., Sawzall of Google, Pig Latin of Yahoo, Hive of Facebook, and Scope of Microsoft.

Dryad: Dryad is a general-purpose distributed execution engine for processing parallel applications of coarse-grained data. The operational structure of Dryad is a directed acyclic graph, in which vertexes represent programs and edges represent data channels. Dryad executes operations on the vertexes in clusters and transmits data via data channels, including documents, TCP connections, and shared-memory FIFO. During operation, resources in a logic operation graph are automatically map to physical resources.

The operation structure of Dryad is coordinated by a central program called job manager, which can be executed in clusters or workstations through network. A job manager consists of two parts: 1) application codes which are used to build a job communication graph, and 2) program library codes that are used to arrange available resources. All kinds of data are directly transmitted between vertexes. Therefore, the job manager is only responsible for decision-making, which does not obstruct any data transmission.

In Dryad, application developers can flexibly choose any directed acyclic graph to describe the communication modes of the application and express data transmission mechanisms. In addition, Dryad allows vertexes to use any amount of input and output data, while MapReduce supports only one input and output set. DryadLINQ is the advanced language of Dryad and is used to integrate the aforementioned SQL-like language execution environment.

V. CONCLUSION

This paper surveys various challenges in the storage and processing of big data. The major areas of challenges are big data mining platform, information sharing and privacy, and mining algorithms. The analysis of different big data mining platforms includes analysis about MPI, MapReduce and Dryad. We regard Big Data as an emerging trend and the need for Big Data mining is arising in all science and engineering domains.

REFERENCES

[1] X.Wu, X.Zhu, G.Wu, and Wei Ding, "Data Mining with Big Data," *IEEE Trans. Knowledge and Data Eng.*, vol. 26, no.1, Jan 2014.
[2] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Cluster," *Google Inc.*
[3] Ghemawat S, Gobioff H, Leung S-T (2003) The google file system. In: ACM SIGOPS Operating Systems Review, vol 37. ACM, pp 29–43
[4] Labrinidis A, Jagadish HV (2012) Challenges and opportunities with big data. Proc VLDB Endowment 5(12):2032–2033.
[5] Chaudhuri S, Dayal U, Narasayya V (2011) An overview of business intelligence technology. Commun ACM 54(8):88–98.
[6] Agrawal D, Bernstein P, Bertino E, Davidson S, Dayal U, FranklinM, Gehrke J, Haas L, Halevy A, Han J et al (2012) Challenges and opportunities with big data. A community white paper developed by leading researches across the United States.
[7] Isard M, Budiu M, Yu Y, Birrell A, Fetterly D (2007) Dryad: distributed data-parallel programs from sequential building blocks.ACM SIGOPS Oper Syst Rev 41(3):59–72.
[8] G. Cormode and D. Srivastava, "Anonymized Data: Generation, Models, Usage," Proc. ACM SIGMOD Int'l Conf. Management Data, pp. 1015-1018, 2009.
[9] Mark Kerzner and Sujee Maniyam, " Hadoop Illuminated," eBook.

