Big Data Processing Model from Mining Perspective

Sobha Rani Neelam
Department of CSE
Audisankara College of Engg & Tech
Gudur, Nellore dt.
Mobile No: 8985983035
Sobharani15@gmail.com

Abstract— Big data is defined as large amount of data which requires new technologies and architectures. Due to such large size of data it becomes very difficult to perform effective analysis using the existing traditional techniques. Big data due to its various properties like volume, velocity, variety, variability, value and complexity put forward many challenges. Since Big data is a recent upcoming technology in the market which can bring huge benefits to the business organizations, it becomes necessary that various challenges and issues associated in bringing and adapting to this technology are brought into light. This paper introduces the Big data technology along with its importance in the modern world and existing projects which are effective and important in changing the concept of science into big science and society too. The various challenges and issues in adapting and accepting Big data technology, its tools (Hadoop) are also discussed in detail along with the problems Hadoop is facing. The features of the Big Data revolution, and proposes a Big Data processing model, from the data mining perspective. This data-driven model involves demand-driven aggregation of information sources, mining and analysis, user interest modeling, and security and privacy considerations. We analyze the challenging issues in the data-driven model and also in the Big Data revolution.

***Keywords — Big data; Hadoop; Hadoop Distributed File System; MapReduce,* NoSQL, MongoDB.**

I. INTRODUCTION

The data is growing at a huge speed makes it difficult to handle such large amount of data. The main difficulty in handling such large amount of data is because that the volume is increasing rapidly in comparison to the computing resources. Big data can be defined with the following properties

A. Variety

Data is not of single category but also the semi structured data from various resources like web Pages, Web Log Files, social media sites, e-mail, documents, sensor devices data

B. Volume

The Big word in Big data itself defines the volume. At present the data existing is in peta bytes and is supposed to increase to zeta bytes in nearby future.

C. Velocity

Velocity in Big data is a concept which deals with the speed of the data. This characteristic is not being limited to the speed of incoming data but also speed at which the data flows.

D. Variability

Variability considers the inconsistencies of the data flow. Data loads become challenging to be maintained especially with the increase in usage of the social media which generally causes peak in data loads with certain events occurring.

E. Complexity

It is quite an undertaking to link, match, cleanse and transform data across systems coming from various sources. It is also necessary to connect and correlate relationships, hierarchies and multiple data linkages or data can quickly spiral out of control.

F. Value

User can run certain queries against the data stored from the filtered data obtained and can also rank it according to the dimensions they require. These reports help the people to find the business trends so that they can change their strategies accordingly. The designing of such systems would be able to handle such large amount of data efficiently and effectively and to filter the most important data from all the data collected by the organization. In other words it adds value to the business.

The rise of Big Data applications where data collection has grown tremendously and is beyond the ability of commonly used software tools to capture, manage, and process within a “tolerable elapsed time.” The most fundamental challenge for Big Data applications is to explore the large volumes of data and extract useful information or knowledge. The knowledge extraction process has to be very efficient and close to real time because storing all observed data is nearly infeasible. Existing methods can only work in an offline fashion and are incapable of

handling this Big Data scenario in real time. As a result, the unprecedented data volumes require an effective data analysis and prediction platform to achieve fast response and real-time classification for such Big Data. For example, each single human being in a biomedical world can be represented by using simple demographic information such as gender, age, family disease history, and so on.

In the coming sections I have presented the main issues and challenges along with the complete description of the technologies/methods being employed for tackling the storage and processing problems associated with Big Data in the mining platform.

II. RELATED WORK

In paper [1] the issues and challenges in Big data are discussed as the authors begin a collaborative research program into methodologies for Big data analysis and design. In paper [2] the author discusses about the traditional databases and the databases required with Big data concluding that the databases don't solve all aspects of the Big data problem and the machine learning algorithms need to be more robust and easier for unsophisticated users to apply. There is the need to develop a data management ecosystem around these algorithms so that users can manage and evolve their data, enforce consistency properties over it and browse, visualize and understand their algorithm results. In paper [3] architectural considerations for Big data are discussed concluding that despite the different architectures and design decisions, the analytics systems aim for Scale-out, Elasticity and High availability. In paper [4] all the concepts of Big data along with the available market solutions used to handle and explore the unstructured large data are discussed. The observations and the results showed that analytics has become an important part for adding value for the social business. This paper [5] proposes the Scientific Data Infrastructure (SDI) generic architecture model. This model provides a basis for building interoperable data with the help of available modern technologies and the best practices. The authors have shown that the models proposed can be easily implemented with the use of cloud based infrastructure services provisioning model. In paper [6] the author investigates the difference in Big data applications and how they are different from the traditional methods of analytics existing from a long time. In paper [7] authors have done analysis on Flickr, Locr, Facebook and Google+ social media sites? Based on this analysis they have discussed the privacy implications and also geo-tagged social media; an emerging trend in social media sites. The proposed concept in this paper helps users to get informed about the data relevant to them in such large social Big data.

(Tier I), which focuses on low-level data accessing and computing challenges on information sharing and privacy, and Big Data application domains and knowledge form Tier II, which concentrates on high-level semantics, application domain knowledge, and user privacy issues. The outmost circle shows Tier III challenges on actual mining algorithms the human society[12].

The present software tools are unable to capture, manage and process the data within time because of the rise of big data. The challenge for big data applications is to store the large volumes of data and extract useful information or knowledge from the Big data. To get fast response from big data it requires effective data analysis, prediction and classification. The complexity of data will be increased along with the size of the data.

Typical data mining algorithms require all data to be loaded into the main memory. This becomes a technical barrier because moving data across different locations is expensive. For the concept of big data, mining from dig data is very difficult i.e., conversion from quantity to quality. It needs parallel computing infrastructure, one good programming language support and software models for data analysis. Existing data models are key-value stores, big table clones, document databases, and graph data base. Map Reduce is a batch oriented parallel computing models.

III. BIG DATA CHALLENGES AND ISSUES

A. Privacy and Security

It is the most important issue with Big data which is sensitive and includes conceptual, technical as well as legal significance[7]. The personal information of a person when combined with external large data sets leads to the inference of new facts about that person and it's possible that these kinds of facts about the person are secretive and the person might not want the Data Owner to know or any person to know about them.

B. Data Access and Sharing of Information

If data is to be used to make accurate decisions in time it becomes necessary that it should be available in accurate, complete and timely manner. This makes the Data management and governance process bit complex adding the necessity to make Data open and make it available to government agencies in standardized manner with standardized APIs,

metadata and formats thus leading to better decision making, business intelligence and productivity improvements. Expecting sharing of data between companies is awkward because of the need to get an edge in business. Sharing data about their clients and operations threatens the culture of secrecy and competitiveness.

C. Storage and Processing Issues

The storage available is not enough for storing the large amount of data which is being produced by almost everything: Social Media sites are themselves a great contributor along with the sensor devices etc. Terabytes of data will take large amount of time to get uploaded in cloud and thus the cloud issues with Big Data can be categorized into Capacity and Performance issues. The transportation of data from storage point to processing point can be avoided in two ways. One is to process in the storage place only and results can be transferred or transport only that data to computation which is important. But Processing of such large amount of data also takes large amount of time. To find suitable elements whole of data Set needs to be Scanned which is somewhat not possible .Thus Building up indexes right in the beginning while collecting and storing the data is a good practice and reduces processing time considerably.

D. Analytical challenges

Big data brings along with it some huge analytical challenges. The type of analysis to be done on this huge amount of data which can be unstructured, semi structured or structured requires a large number of advance skills. Moreover the type of analysis which is needed to be done on the data depends highly on the results to be obtained i.e. decision making. This can be done by using one using two techniques: either incorporate massive data volumes in analysis or determine upfront which Big data is relevant.

E. Skill Requirement

Since Big data is at its youth and an emerging technology so it needs to attract organizations and youth with diverse new skill sets.

F. Technical Challenges

1) Fault Tolerance: With the incoming of new technologies like Cloud computing and Big data it is always intended that whenever the failure occurs the damage done should be within acceptable threshold rather than beginning the whole task from the scratch. Thus the main task is to reduce the probability of failure to an "acceptable" level. Unfortunately, the more we strive to reduce this probability, the higher the cost. Two methods which seem to increase the fault tolerance in Big data are as: First is to divide the whole computation being done into tasks and assign these tasks to different nodes for computation. In case of any failure, the computation can restart from last checkpoint maintained.

2) Scalability: The processor technology has changed in recent years. Previously data processing systems had to worry about parallelism across nodes in a cluster but now the concern has shifted to parallelism within a single node. In past the techniques which were used to do parallel data processing across data nodes aren't capable of handling intra-node parallelism. This is because of the fact that many more hardware resources such as cache and processor memory channels are shared across a core in a single node.

The scalability issue of Big data has lead towards cloud computing, which now aggregates multiple disparate workloads with varying performance goals into very large clusters. This requires a high level of sharing of resources which is expensive and also brings with it various challenges like how to run and execute various jobs so that we can meet the goal of each workload cost effectively. It also requires dealing with the system failures in an efficient manner which occurs more frequently if operating on large clusters. These factors combined put the concern on how to express the programs, even complex machine learning tasks.There has been a huge shift in the technologies being used.

Hard Disk Drives (HDD) are being replaced by the solid state Drives and Phase Change technology which are not having the same performance between sequential and random data transfer. Thus the kind of storage devices used is again a big question for data storage.

3) Quality of Data: Collection of huge amount of data and its storage comes at a cost. More data if used for decision making or for predictive analysis in business will definitely lead to better results. Big data basically focuses on quality data storage rather than having very large irrelevant data so that better results and conclusions can be drawn.

4) Heterogeneous Data: Unstructured data represents almost every kind of data being produced like social media interactions, to recorded meetings, to handling of PDF documents, fax transfers, to emails and more. Structured data is always organized into highly mechanized and manageable way. It shows well integration with database but unstructured data is completely raw and unorganized. Working with unstructured data is cumbersome and of course costly too. Converting all this unstructured data into structured one is also not feasible. Structured data is the one which is organized in a way so that it can be managed easily. Digging through unstructured data is cumbersome and costly.

IV. TOOLS AND TECHNIQUES AVAILABLE

The following tools and techniques are available:

A. Hadoop

Hadoop is an open source project hosted by Apache Software Foundation. It consists of many small sub projects which belong to the category of infrastructure for distributed computing. Hadoop mainly consists of :

- File System (The Hadoop File System)
- Programming Paradigm (Map Reduce)

The other subprojects provide complementary services or they are building on the core to add higher-level abstractions. There exist many problems in dealing with storage of large amount of data. Though the storage capacities of the drives have increased massively but the rate of reading data from them hasn't shown that considerable improvement. The reading process takes large amount of time and the process of writing is also slower. This time can be reduced by reading from multiple disks at once. Only using one hundredth of a disk may seem wasteful. But if there are one hundred datasets, each of which is one terabyte and providing shared access to them is also a solution. There occur many problems also with using many pieces of hardware as it increases the chances of failure. This can be avoided by Replication. The main problem is of combining the data being read from different devices. Many a methods are available in distributed computing to handle this problem but still it is quite challenging. All the problems discussed are easily handled by Hadoop. The problem of failure is handled by the Hadoop Distributed File System and problem of combining data is handled by Map reduce programming Paradigm. Map Reduce basically reduces the problem of disk reads and writes by providing a programming model dealing in computation with keys and values. Hadoop thus provides: a reliable shared storage and analysis system. The storage is provided by HDFS and analysis by MapReduce.

B. Hadoop Components in detail

1) Hadoop Distributed File System: Hadoop comes with a distributed File System called HDFS, which stands for Hadoop Distributed File System. HDFS is a File System designed for storing very large files with streaming data access patterns, running on clusters on commodity hardware. HDFS block size is much larger than that of normal file system i.e. 64 MB by default. A HDFS cluster has two types of nodes i.e. namenode (the master) and number of datanodes (workers). The name node manages the file system namespace, maintains the file system tree and the metadata for all the files and directories in the tree. The datanode stores and retrieve blocks as per the instructions of clients or the namenode. The data retrieved is reported back to the namenode with lists of blocks that they are storing. Without the namenode it is not possible to access the file. So it becomes very important to make namenode resilient to failure. These are areas where HDFS is not a good fit: Low-latency data access, Lots of small file, multiple writers and arbitrary file modifications.

2) MapReduce: MapReduce has two phases Map and Reduce. Map takes an input pair and produces a set of intermediate key/value pairs. The Mapreduce library groups all intermediate values that are associated with the same intermediate key and passes them to the Reduce function. Reduce function accepts an intermediate key and a set of values for that key. It merges these values to form a smaller set of values. For each Map task and reduce task it stores the state like idle, in-progress, or completed. And the identity of the map reduce library must tolerate the machine failures.

The Map reduce is easy to use for programmers without experience with parallel and distributed systems. Since it hides the details of parallelization, fault tolerance, locality, optimization and load balancing. The advantage with Map reduce is large varieties of problems are easily expressible as MapReduce computations.

MapReduce was not originally developed to perform structured data analysis. HadoopDB is an open source version of Hadoop that can be used without cost. Map and reduce functions written in a general purpose language. Map reduce best meets the fault tolerance and able to operate in heterogeneous environment properties. Map reduce has a flexible query interface but does not create a detailed query execution plan. This plan specifies which nodes will run which tasks in advance. Instead of this it is determined at runtime.

Mapreduce and other batch processing systems cannot process small updates individually as they rely on creating large batches efficiently.

A Map-reduce computation executes as follows: Map tasks are given input from distributed file system. The Map tasks produce a sequence of key-value pairs from the input and this is done according to the code written for Map function. These value generated are collected by master controller and are sorted by key and divided among reduce tasks. The sorting basically assures that the same key values ends with the same reduce tasks. The Reduce tasks combine all the values associated with a key working with one key at a time. Again the combination process depends on the code written for reduce job.

The Master controller process and some number of worker processes at different compute nodes are forked by the user. Worker handles Map

tasks (MAP WORKER) and reduce tasks (REDUCE WORKER) but not both. The Master controller creates some number of Map and reduce tasks which is usually decided by the user program. The tasks are assigned to the worker nodes by the master controller. Track of the status of each Map and Reduce task (idle, executing at a particular Worker or completed) is kept by the Master Process. On the completion of the work assigned the worker process reports to the master and master reassigns it with some task. The failure of a compute node is detected by the master as it periodically pings the worker nodes.

All the Map tasks assigned to that node are restarted even if it had completed and this is due to the fact that the results of that computation would be available on that node only for the reduce tasks. The status of each of these Map tasks is set to idle by Master. These get scheduled by Master on a Worker only when one becomes available. The Master must also inform each Reduce task that the location of its input from that Map task has changed.

C. Comparison of Hadoop Technique with other system Techniques

1) Comparison with HPC and Grid Computing Tools: The approach in HPC and Grid computing includes the distribution of work across a cluster and they are having a common shared File system hosted by SAN. The jobs here are mainly compute intensive and thus it suits well to them unlike as in case of Big data where access to larger volume of data as network bandwidth is the main bottleneck and the compute nodes start becoming idle. Map Reduce component of Hadoop here plays an important role by making use of the Data Locality property where it collocates the data with the compute node itself so that the data access is fast. HPC and Grid Computing basically make use of the API's such as message passing Interface (MPI). Though it provides great control to the user, the user needs to control the mechanism for handling the data flow. On the other hand Map Reduce operates only at the higher level where the data flow is implicit and the programmer just thinks in terms of key and value pairs. Coordination of the jobs on large distributed systems is always challenging. Map Reduce handles this problem easily as it is based on shared-nothing architecture i.e. the tasks are independent of each other. The implementation of Map Reduce itself detects the failed tasks and reschedules them on healthy machines. Thus the order in which the tasks run hardly matters from programmer's point of view. But in case of MPI, an explicit management of check pointing and recovery system needs to be done by the program. This gives more control to the programmer but makes them more difficult to write.

2) Comparison with Volunteer Computing Technique: In Volunteer computing work is broken down into chunks called work units which are sent on computers across the world to be analyzed. After the completion of the analysis the results are sent back to the server and the client is assigned with another work unit. In order to assure accuracy, each work unit is sent to three different machines and the result is accepted if at least two of them match. This concept of Volunteer Computing makes it look like MapReduce. But there exists a big difference between the two the tasks in case of Volunteer Computing are basically CPU intensive. This tasks makes these tasks suited to be distributed across computers as transfer of work unit time is less than the time required for the computation whereas in case of MapReduce is designed to run jobs that last minutes or hours on trusted, dedicated hardware running in a single data center with very high aggregate bandwidth interconnects.

3) Comparison with RDBMS: The traditional database deals with data size in range of Gigabytes as compared to MapReduce dealing in petabytes. The Scaling in case of MapReduce is linear as compared to that of traditional database. In fact the RDBMS differs structurally, in updating, and access techniques from MapReduce.

V. BIG DATA GOOD PRACTICES

 Creating dimensions of all the data being store is a good practice for Big data analytics. It needs to be divided into dimensions and facts.

 all the dimensions should have durable surrogate keys meaning that these keys can't be changed by any business rule and are assigned in sequence or generated by some hashing algorithm ensuring uniqueness.

 Expect to integrate structured and unstructured data as all kind of data is a part of Big data which needs to be analyzed together.

 Generality of the technology is needed to deal with different formats of data. Building technology around key value pairs work.

 Analyzing data sets including identifying information about individuals or organizations privacy is an issue whose importance particularly to consumers is growing as the value of Big data becomes more apparent.

 Data quality needs to be better. Different tasks like filtering, cleansing, pruning, conforming, matching, joining, and diagnosing should be applied at the earliest touch points possible.

 there should be certain limits on the scalability of the data stored.

 Collecting, storing and analyzing data comes at a cost. The decisions taken should be revised to ensure that the organization is considering the right data to produce insights at any given point of time.
 Investment in data quality and metadata is also important as it reduces the processing time.

From years, business persons will make the decisions based on the transactional data which is stored in relational databases. Using data mining the unstructured data like web logs, social media email, sensors and photographs can be mined for useful information. In big data, data types include structured data, unstructured data and tabulation form& semantic association of data.

Analysis of structural data relies on keywords which allow the users to filter the data based on searchable terms. Unstructured data cannot easily be separated into categories are analyzed numerically. NoSQL is often used for storing Big Data. This is a new type of database which is becoming more and more popular among web companies today. The NoSQL solutions provide simpler scalability and improved performance relative to traditional relational databases. These products excel at storing “unstructured data,” and the category includes open source products such as Cassandra, MongoDB, and Redis.

A **NoSQL** (originally referring to "non SQL" or "non relational”) database provides a mechanism for storage and retrieval of data which is modeled in means other than the tabular relations used in relational databases. NoSQL databases are increasingly used in big data and real-time web applications. Many NoSQL stores compromise consistency in favor of availability, partition tolerance, and speed.

A NoSQL database environment is a non-relational and largely distributed database system that enables rapid, ad-hoc organization and analysis of extremely high-volume, disparate data types. NoSQL databases are sometimes referred to as cloud databases, non-relational databases, Big Data databases and a myriad of other terms and were developed in response to the sheer volume of data being generated, stored and analyzed by modern users and their applications.

In general, NoSQL databases have become the first alternative to relational databases, with scalability, availability, and fault tolerance being key deciding factors. A very flexible and schema-less data model, horizontal scalability, distributed architectures, and the use of languages and interfaces that are “not only” SQL typically characterize this technology.

MongoDB is one of several database types to arise in the mid-2000s under the NoSQLbanner. Instead of using tables and rows as in relational databases, MongoDB is built on architecture of collections and documents. Documents comprise sets of key-value pairs and are the basic unit of data in MongoDB. Collections contain sets of documents and function as the equivalent of relational database tables. Like other NoSQL databases, MongoDB supports dynamic schema design, allowing the documents in a collection to have different fields and structures.

The main challenges of Big Data are data variety, volume, analytical workload complexity and agility. Many organizations are struggling to deal with the increasing volumes of data. In order to solve this problem, the organizations need to reduce the amount of data being stored and exploit new storage techniques which can further improve performance and storage utilization.

While the volume of the Big Data increases, so do the complexity and the relationships underneath the data. In an early stage of data centralized information systems, the focus is on finding best feature values to represent each observation. This is similar to using a number of data fields, such as age, gender, income, education background, and so on, to characterize each individual. This type of sample feature representation inherently treats each individual as an independent entity without considering their social connections, which is one of the most important factors of

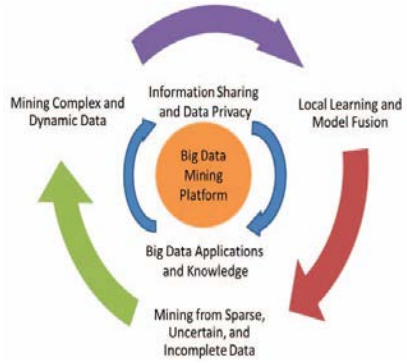

Fig. 1. A Big Data processing framework: The research challenges form a three tier structure and center around the “Big Data mining platform”

Big Data Mining Platform

In typical data mining systems, the mining procedures require computational intensive computing units for data analysis and comparisons. A computing platform is, therefore, needed to have

efficient access to, at least, two types of resources: data and computing processors. For small scale data mining tasks, a single desktop computer, which contains hard disk and CPU processors, is sufficient to fulfill the data mining goals. Common solutions are to rely on parallel computing or collective mining to sample and aggregate data from different sources and then use parallel computing programming (such as the Message Passing Interface) to carry out the mining process. 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), on a large number of computing nodes (i.e., clusters).

Big Data Semantics and Application Knowledge

Semantics and application knowledge in Big Data refer to numerous aspects related to the regulations, policies, user knowledge, and domain information. The two most important issues at this tier include 1) data sharing and privacy; and 2) domain and application knowledge.

Big Data Mining Algorithms

More specifically, the global mining can be featured with a two-step (local mining and global correlation) process, at data, model, and at knowledge levels. At the data level, each local site can calculate the data statistics based on the local data sources and exchange the statistics between sites to achieve a global data distribution view[12]. At the model or pattern level, each site can carry out local mining activities, with respect to the localized data, to discover local patterns. In Big Data, data types include structured data, unstructured data, and semi structured data, and so on. Specifically, there are tabular data (relational databases), text, hyper-text, image, audio and video data, and so on. The existing data models include key-value stores, big table clones, document databases, and graph databases, which are listed in an ascending order of the complexity of these data models. Traditional data models are incapable of handling complex data in the context of Big Data. Currently, there is no acknowledged effective and efficient data model to handle Big Data. However, in the context of Big Data, it is a great challenge to efficiently describe semantic features and to build semantic association models to bridge the semantic gap of various heterogeneous data sources.

VI. CONCLUSION

Currently, Big Data processing mainly depends on parallel programming models like MapReduce, as well as providing a cloud computing platform of Big Data services for the public. MapReduce is a batch-oriented parallel computing model. There is still a certain gap in performance with relational databases. Improving the performance of MapReduce and enhancing the real-time nature of large-scale data processing have received a significant amount of attention, with MapReduce parallel programming being applied to many machine learning and data mining algorithms. Data mining algorithms usually need to scan through the training data for obtaining the statistics to solve or optimize model parameters.Large scale data sets can be divided into several subsets and assigned and various operations are performed and finally the results are summed and the algorithms used for mining can be executed in parallel.

While the term Big Data literally concerns about data volumes, such combined characteristics suggest that Big Data require a “big mind” to consolidate data for maximum values .To explore Big Data, we have analyzed several challenges at the data, model, and system levels. To support Big Data mining, high-performance computing platforms are required, which impose systematic designs to unleash the full power of the Big Data. At the data level, the autonomous information sources and the variety of the data collection environments, often result in data with complicated conditions, such as missing/uncertain values. In other situations, privacy concerns, noise, and errors can be introduced into the data, to produce altered data copies. Developing a safe and sound information sharing protocol is a major challenge.

This requires carefully designed algorithms to analyze model correlations between distributed sites, and fuse decisions from multiple sources to gain a best model out of the Big Data. At the system level, the essential challenge is that a Big Data mining framework needs to consider complex relationships between samples, models, and data sources, along with their evolving changes with time and other possible factors. A system needs to be carefully designed so that unstructured data can be linked through their complex relationships to form useful patterns, and the growth of data volumes and item relationships should help form legitimate patterns to predict the trend and future.

We regard Big Data as an emerging trend and the need for Big Data mining is arising in all science and engineering domains. With Big Data technologies, we will hopefully be able to provide most relevant and most accurate social sensing feedback to better understand our society at real time.

VII. REFERENCES

[1] Stephen Kaisler, Frank Armour, J. Alberto spinosa, William Money,
"Big Data: Issues and Challenges Moving Forward", *IEEE, 46th Hawaii International Conference on System Sciences*, 2013.
[2] Sam Madden, " From Databases to Big Data", *IEEE, Internet Computing*, May-June 2012.
[3] Kapil Bakshi, "Considerations for Big Data: Architecture and Approach", *IEEE , Aerospace Conference*, 2012.
[4] Sachchidanand Singh, Nirmala Singh, "Big Data Analytics", *IEEE, International Conference on Communication, Information & Computing Technology (ICCICT)*, Oct. 19-20, 2012.
[5] Yuri Demchenko, Zhiming Zhao, Paola Grosso, Adianto Wibisono, Cees de Laat, "Addressing Big Data Challenges for Scientific Data Infrastructure", *IEEE , 4th International Conference on Cloud Computing Technology and Science*, 2012.
[6] Martin Courtney, "The Larging-up of Big Data", *IEEE, Engineering & Technology*, September 2012.
[7] Matthew Smith, Christian Szongott, Benjamin Henne, Gabriele von Voigt, "Big Data Privacy Issues in Public Social Media", *IEEE, 6th International Conference on Digital Ecosystems Technologies (DEST)*, 18-20 June 2012.
[8] Avita Katal, Mohammad Wazid, R H Goudar "Big Data: Issues, Challenges, Tools and Good Practices"
[9] World's data will grow by 50X in next decade, IDCstudy predicts ttp://www. computerworld.com /s/article/9217988/World_s_data_will_grow_by_50X _in_next_decade_IDC_study_predicts
[10] The 2011 Digital Universe Study: Extracting Value from Chaos http://www.emc.com/collateral/ demos/microsites/emc-digital-universe-2011/index. htm
[11]. Sachchidanand Singh, Nirmala Singh, "Big Data Analytics" International Conference on Communication, Information & Computing Technology (ICCICT), Oct. 19-20 2012, Mumbai, India
[12] Xindong Wu, Fellow, IEEE, Xingquan Zhu, Senior Member, IEEE, Gong-Qing Wu, and Wei Ding, Senior Member, IEEE, "*Data Mining with Big Data*" ieee transactions on knowledge and data engineering, vol. 26, no. 1, january 2014