IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-9, Sept 2019 E-ISSN: 2347-2693

Analysis of Techniques to Retrieve Big Database

S. Puri[1*], L. Jain[2], O.P. Gupta[3]

[1,2]Electrical Engineering & Information Technology, College of Agricultural Engineering and Technology, Punjab Agricultural University, Ludhiana, India
[3]Information Technology Section, College of Agricultural Engineering and Technology, Punjab Agricultural University, Ludhiana, India

[*]*Corresponding Author: shivanipuri2013@gmail.com, Tel.: +91-6239763392*

DOI: https://doi.org/10.26438/ijcse/v7i9.6671 | Available online at: www.ijcseonline.org

Accepted: 08/Sept/2019, Published: 30/Sept/2019

***Abstract*—** In today's world there are a large amount of data which need to be processed with big databases. In recent years, increase plethora of companies has adopted different-different types of non-relational database. The goal of this research is to implement techniques to retrieve big database for the big datasets and investigate the performance of the big database techniques on CPU utilization and high-performance computing software. It attempts to use NoSQL database to replace the relational database. In this research mainly focuses on the new technology of NoSQL database i.e. MongoDB, HadoopDB. Performance comparison of two big data techniques is carried out. The result found that Aggregation technique consumes less execution time than MapReduce technique and more efficient with MongoDB database where as MapReduce technique has less efficient with HadoopDB. Aggregation technique also produces fine relevant information results with less CPU utilization. The result also shows that MongoDB has the capability to switch SQL databases as compare to HadoopDB.

***Keywords*—** Big Data, MongoDB, HadoopDB, Aggregation, MapReduce

I. INTRODUCTION

The concept of big data arises into presence with the growing capability to perform different jobs and with the origination of internet and ordinal technologies. It is a difficult task to store the massive data. So big data get a high importance and it is a suitable choice for novel researchers. The era of big data become begin with the increase of data in all fields such as agricultural, bioinformatics and many more with the increasing use of IT. The rapid growth of data is known as data, statistics, and information explosion [1]. The term big data is defined by bulk of data with many collections which is expanding day by day. When there are a number of collection of data that it is difficult to operate and load on a single unit then there is need of database management tools. These tools provide the support to manage the big data [2].

There are many companies, organizations which have bulk of unstructured data but they do not know how to process these bulks of data. Different technologies are introduced to deal with big data. Many companies such as Google, Facebook, Amazon, Twitter etc. have been invested on big data projects [3]. The purpose of this research paper is that the utility of big data collection and analytics has been posed threats to accuracy and access to data. The data might be in any form either structured or unstructured as shown in Figure 1.

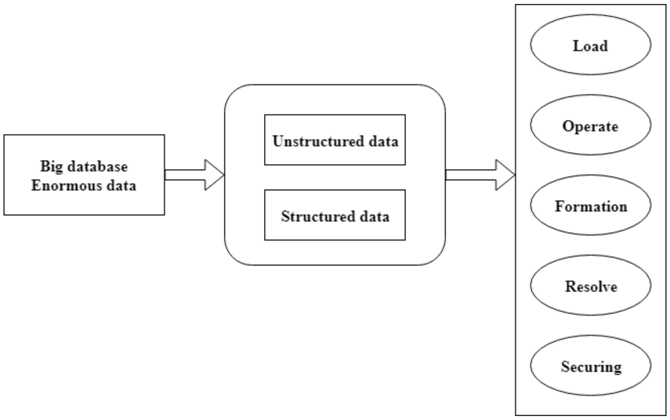

Figure 1. *Fundamental Requirement of Big Data*

Data which is coming from discrete sources has to be evolved, resolve, secure, operate, load such as farmer field, ground sensors, data which is collected by government, IOT (Internet Of Things), large organization covered as big data [4].

Objectives: The motive of this research is to

1. Study of various techniques to analyse big data
2. Performance characterization of Aggregation and MapReduce in big data.

Respite of paper is organized as follows, Section I contains the introduction of the importance of the study with the aims of the study, Section II contain the related work of big databases, Section III contains a brief overview of big database techniques, Section IV explain the methodology of work, Section V describes results and discussion of big database analytics, Section VI concludes research work with future directions.

II. RELATED WORK

2.1 Dean J and Ghemawat S (2008) [5] described the technique that is Map-Reduce for enhancement of extract the big data. For producing the big data and for parallel processing, a model is suggested that is MapReduce which is direct, accessible programming model. The program which is written in this model will spontaneously parallelize and then executed on a large cluster. For managing machine failures, scheduling the executable programs, the run-time system will take care of these all. So that programmers can utilize all the resources of large distributed system easily. With the employment of MapReduce, the bulky data will be accessed on a large cluster. Number of MapReduce jobs is executed on large clusters every single day. It has two main functions. The first function is filtering the group of data or mapping the bunch of data which is referred as mapper and the second function is to shape and reduce the consequence (result) from cluster and produce effective reply of query that is referred as reducer. Their implementation of MapReduce runs on a large cluster of commodity machines. Many MapReduce jobs are executed on Google's clusters every day.

2.2 Dede *et al* (2013) [6] evaluated the parameters that are fault-tolerance, scalability and performance by using MongoDB and HadoopDB and also trying to identifying the accurate environment of software for analysis of data. There are many projects which need to capture, load and process non-static semi-structured data and metadata such as the Materials Project and there are scientific abilities such as the Advanced Light Source (ALS) and Joint Genome Institute. With the growth of semi-structured data within large Internet service providers has led to the creation of NoSQL data stores for scalable indexing and MapReduce for scalable parallel analysis. MapReduce and NoSQL stores have been applied to scientific data. Hadoop, the most popular open source implementation of MapReduce, has been evaluated, utilized and modified for addressing the needs of different scientific analysis problems. ALS and the Materials Project are using MongoDB, a document oriented NoSQL store. However, there is a limited understanding of the performance trade-offs of using these two technologies together.

2.3 Nunan and Domenico (2013) [7] described the term 'big data' with different technologies and marketable trends which help to store and analysis bulk of data that help to produced social networks and mobile devices. They ensured to produce valuable understanding about big data in the commercial trends for gathering new types and volumes which could not be practical in the past. This paper tells about the understanding and development of big data. Many questions had been about the privacy of big data. It was considered that the challenges were raised up for market research.

2.4 Ozarkar and Rajani (2014) presented a new system for querying document which is dynamic query forms. There are thousands of relations and attributes of heterogeneous data which need to be maintaining in the real world. To process this huge database non-trivial assignment and is an exploring area of interest. There are number of queries which are using for database process, but it is not an easy task for those who are not well aware with query language. For ease of user there is a query form which will assist the user to iteratively examine of records. Users can also give response for query enhancement by giving rank to different attributes. If there is negative response from user then it will be removed to improve the query form technique. For this purpose ad_hoc queries can also satisfied by using NoSQL database like MONGODB that maintain dynamic queries.

2.5 Bhosale and gadekar (2014) [9] described the word Big Data and the open source software that is Hadoop. They told about technologies and techniques to process the data sets of huge size with high velocity. Big data can be structured, unstructured or semi-structured. Many sources can generate the data of large size. To produce these large data effective software that is parallelism is used. Big data is a data whose scale, variety so it requires algorithms, analytics to process it and get value and abstract knowledge from it. The core platform for arranging big data and give solutions for different tasks Hadoop is used. It is planned from single node to numbers of machines with fault tolerance.

III. OVERVIEW OF BIG DATABASE TECHNOLOGY

To enhance and improve the performance of retrieving huge data, Aggregation and Map-Reduce and many other techniques are being used. By using these two techniques there are many benefits which are Resulting new business prospects, Data management will become better, Providing visualization of data, Welfares from cloud service provider in the form of speed, ability, and scalability, Develop new analysis methods and capabilities [10]. There are many examples of databases to manage the big databases and make it simple such as HadoopDB, MongoDB, HIVE, APACHE, SciDB, CouchDB and many more.

Big data can be handled by two types of databases that are relational and non-relational databases. For relational databases, developers have to face many difficulties so

developers fluctuating towards non-relational database [11]. These two databases are completely unique from each other.
There are many types of non-relational database such as MongoDB, HadoopDB, Cassandra, HBase, CouchDB, Riak, Redis et cetera [12]. This paper proposes a method which provides the analysis of MongoDB and HadoopDB by using Aggregation and MapReduce technique respectively.

The descriptions of some non-relational databases which help to analyse techniques to retrieve big databases are as follows:

3.1 HadoopDB

It is open source groundwork as Google, Yahoo, use HadoopDB groundwork. It is a software tool which is used for breaking down data into tiny parts such as cluster. HadoopDB has distributed storage that is refer as Map-Reduce. Map reduce is the heart of HadoopDB. The main purpose of HadoopDB is to load and evolve the data. Hadoop is the first optimal choice for big data processing. It is the platform for structuring Big Data, and solves the problem of making it useful for analytics purposes [8]. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance.

3.2 MongoDB

MongoDB is also an open source document oriented database. When there is need of content and user management, it is recommended to use MongoDB. In the MongoDB, there is no need of laying structure of records. There are many features such as rich data model, dynamic schema, data locality, field update, easy for programmers. There are no complex transactions and supports multi-keys as shown below in Figure 2. Sharding is being used by it horizontally. One more kind of database is Sharding. It has main role to divide big data to tinier, speedy, manageable units that are known as shards.

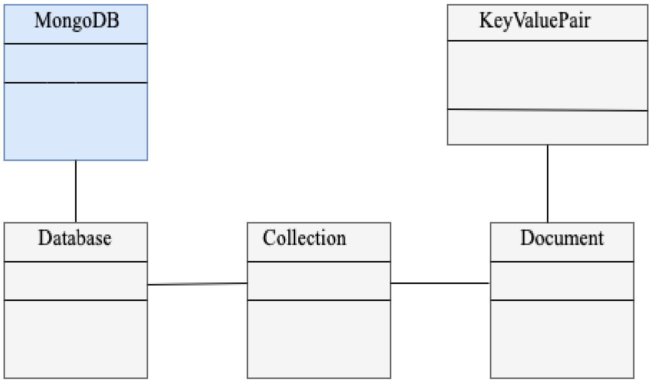

Figure 2. *Structure of MongoDB*

MongoDB facilitates the function of aggregation. The function of aggregation is essential for both SQL and NoSQL database [13]. The term aggregation is defined as data (information) is loaded (retrieved) from many servers and conclusion is clean up in the form of compact (summary). By using the MongoDB database for retrieving the big data in summary form will improve the performance of system.

IV. METHODOLOGY

In this system, a method is proposed to integrate these two types of databases by using graphical user interface tool for evaluating results. For implementation of MongoDB, MongoDB Atlas cloud platform is used and for HadoopDB, in Oracle VM Virtual Box Hive user interface is used. The system makes use of MongoDB and the experimental setup of MongoDB is shown in Figure 3 in which it describes the Opcounters, Network, Connection, and Logical Size of certain cluster. The experimental design and the techniques used along with appropriate statistical methods used clearly along with the MongoDB Atlas.

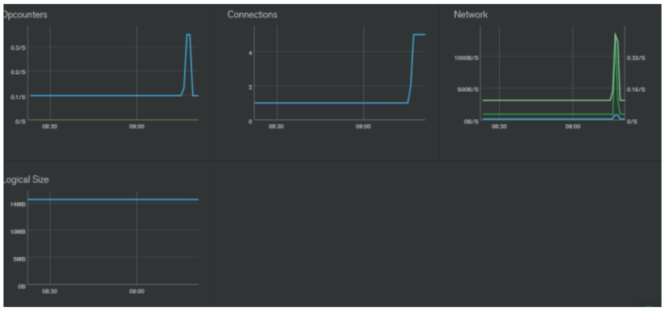

Figure 3. *Online Status of Cluster*

3.1 Aggregation Technique in MongoDB

MongoDB Atlas is an interactive system. It has highly valued and familiar in world major projects for primary output. By using this platform, the implementation of Aggregation technique is carried out to retrieve information as shown in Figure 4. The aggregation technique creates aggregate data of large dataset. The main reason for this, as it was already stated, aggregation is a method which solves the problem into summarizes way. For this reason and depending on the particular application of the database, fewer or greater datasets can be accessed and fetch effective information.

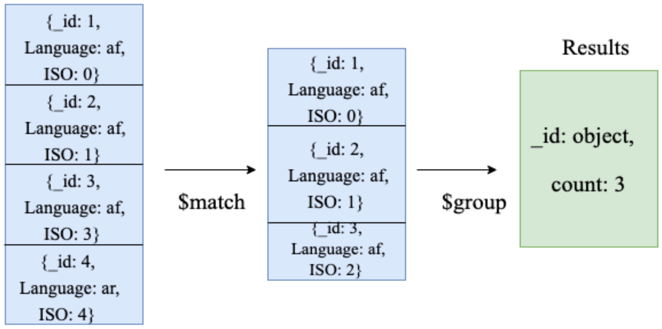
{_id: 1, Language: af, ISO: 0}
{_id: 2, Language: af, ISO: 1}
{_id: 3, Language: af, ISO: 3}
{_id: 4, Language: ar, ISO: 4}
$match
{_id: 1, Language: af, ISO: 0}
{_id: 2, Language: af, ISO: 1}
{_id: 3, Language: af, ISO: 2}
$group
Results
_id: object, count: 3

Figure 4. *Performing Aggregation on Particular Dataset*

3.2 MapReduce Technique in HadoopDB

Apache Hive version 3.1.1 is the user interface used. It is a data warehouse which is built on topmost of HadoopDB [14]. It is implemented on Java based language. The framework of HadoopDB is written in Java programming tool. So there is need of java installation of latest version on system that is java version 1.8.0_201. By using this platform, the implementation of MapReduce technique is conceded out to retrieve information from big dataset as shown in Figure 5.

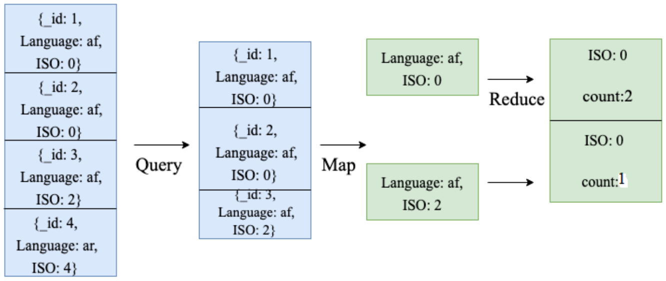
{_id: 1, Language: af, ISO: 0}
{_id: 2, Language: af, ISO: 0}
{_id: 3, Language: af, ISO: 2}
{_id: 4, Language: ar, ISO: 4}
Query
{_id: 1, Language: af, ISO: 0}
{_id: 2, Language: af, ISO: 0}
{_id: 3, Language: af, ISO: 2}
Map
Language: af, ISO: 0
Language: af, ISO: 2
Reduce
ISO: 0 count:2
ISO: 0 count:1

Figure 5. *Performing MapReduce on Particular Dataset*

In our proposed, we make use of standard NoSQL databases i.e. MongoDB and HadoopDB. Due to the need, of bulky data storage we make use one of boosting technology of NoSQL database i.e. MongoDB. MongoDB stores both JSON and CSV structure. Figure 4 and Figure 5 shows the detailed flow of our proposed method.

V. RESULTS AND DISCUSSION

As per the complete review of several papers, a study of big database analysis is prepared between MongoDB and HadoopDB built on their concept and commands used for Aggregation and MapReduce techniques respectively [15].

4.1 Comparison Based on Term/Concept

Table 1 Term/Concept

MongoDB terms/concepts	HadoopDB terms/concept
Collection	Table
JSON Document, Field	HDFS, Column Oriented
Index, Aggregation, Replication	HDFS, MapReduce
Embedded Documents and linking	Table Joins
Schema Less	Schema Less

This table shows the comparison between MongoDB and HadoopDB created on their conception.

4.2 Comparison Based on Schema Statements

As in big database approaches, there are certain schema queries to retrieve effective information [16]. So some schema statements of MongoDB and HadoopDB are as follows:

- **Create Command**

In MongoDB Schema, it is "use database_name and db.CreateCollection(name, option)" whereas in HadoopDB Schema it is "Create database database_name; and Create table table_name[(col_name data_type)] Row format row_format Stored as file_format;"

- **Insert Command**

In MongoDB Schema, it is "db.collection_name.insert (document)" where as in HadoopDB Schema it is "Insert into table_name (col_name data_type);"

- **Drop Command**

In MongoDB Schema, it is "db.dropDatabase() and db.collection_name.drop()" where as in HadoopDB Schema it is "Drop database database_name; and Drop table table_name;"

- **Select Command**

In MongoDB Schema, it is "db.collection_name.find (document)" where as in HadoopDB Schema it is "Select select_expr from table_name;"

- **Delete Command**

In MongoDB Schema, it is "db.collection_name.remove (deletion_criteria)" where as in HadoopDB Schema it is "Drop table table_name;"

Performance Analysis

In this study, implementation and analysis of MongoDB and HadoopDB database have been created. Unstructured datasets of large number of rows are used to implement techniques of big database. The given graph shows the result of implementation. In the database, 100 to 50,000 rows of information have been inserted. The execution time, CPU utilization, efficiency of MongoDB and HadoopDB were recorded in real time as shown below with the help of graph [17].

Two major factors for which MongoDB was preferred over HadoopDB are:

- **Query Speed**

From the graph, we notice that MongoDB spends less execution time than HadoopDB, for large amount of information as shown in table 2. All the readings are taken in real time. MongoDB is much faster than HadoopDB as shown in Figure (Fig.3).

Table 2 Query Speed Comparison

Total Rows	MongoDB Execution Time	HadoopDB Execution Time
More than 100	0.16 (secs)	1.80 (secs)
More than 1000	0.63 (secs)	1.64 (secs)
More than 5000	0.72 (secs)	1.62 (secs)
More than 25000	1.05(secs)	1.50 (secs)
More than 50000	2.48(secs)	2.92 (secs)

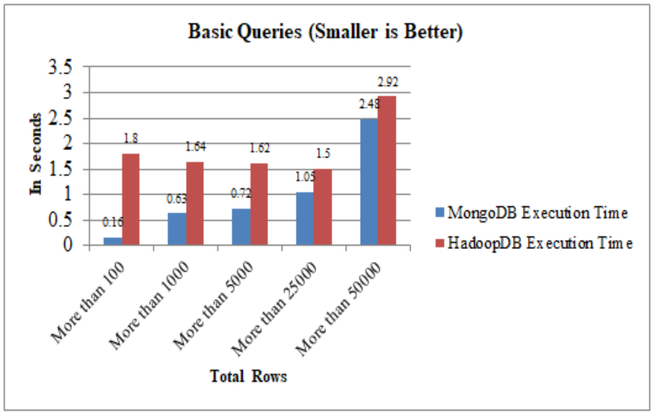

Figure 6. *Query Execution Time for MongoDB and HadoopDB*

- **CPU Utilization**

In Figure 4, it calculates the utilization of CPU to get the information out of the database.

Table 3. CPU Utilization Comparison

Total Rows	Mongo DB CPU Utilization	Hadoop DB CPU Utilization
More than 100	0.9%	3.62%
More than 1000	1.78%	2.91%
More than 5000	1.22%	2.58%
More than 25000	1.66%	1.82%
More than 50000	1.74%	1.61%

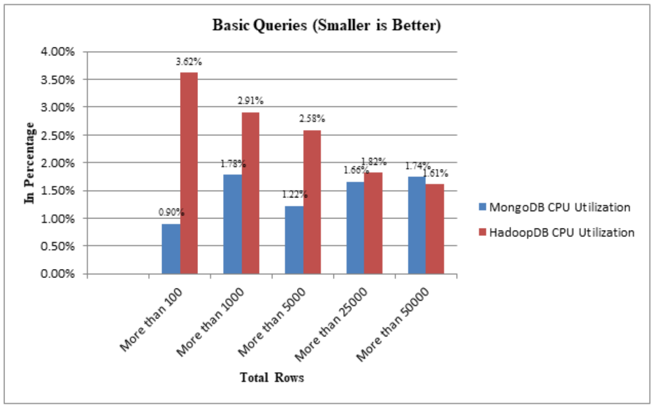

Figure 7. *Query CPU Utilization for MongoDB and HadoopDB*

After implementing the data retrieving two techniques, performance is compared on the basis of parameters like efficiency, execution time and CPU utilization. The experimental results show that Aggregation technique performed on MongoDB approach consumes less execution time and CPU utilization as compared to MapReduce technique. There is no algorithm that can be universally used to solve all problems. Usually, algorithms are designed with certain assumptions and favor some type of biases. All the implementation is done in real time. Thus from the above analysis, it proves that for large amount of data MongoDB is preferred over HadoopDB. Basically MongoDB is planned to substitute RDBMS but on the other hand, HadoopDB helps to increase efficient data in either SQL or NoSQL. Moreover MongoDB is cost-effective as it is particular product while HadoopDB is not because it is a group of software. Table 4 shows the overall comparison of both these technologies.

Table 4. Overall Performance Analysis of Data Retrieving Techniques

Database Approaches	Execution Time	CPU Utilization
MongoDB	1.008 (secs)	0.014%
HadoopDB	1.896 (secs)	0.025%

VI. CONCLUSION

In this paper, performance evaluation of these two techniques is also conducted on the basis of two parameters like execution time, computational time. This paper is about implementing Aggregation Technique on MongoDB database and MapReduce technique on HadoopDB database. It is concluded that HadoopDB would not go away, they are still definitely needed. We can choose MongoDB instead of HadoopDB because of two factors, ease of use and performance, we conclude that if your application is data intensive and stores lots of data, queries lots of data, then you'd better do with MongoDB resources instead of HadoopDB. The system was proposed because MongoDB and HadoopDB have newly come into existence. However, it totally depends on user requirements, as Aggregation technique in MongoDB database consumes less execution time and has better performance than MapReduce technique in HadoopDB database. Aggregation technique also produces fine relevant information results with less CPU utilization.

REFERENCE

[1] Kirti, M Pardeep, *"Database for unstructured, semistructured data- NoSQL"*, International journal of advanced research in computer engineering & technology, Vol. **4**, Issue.**2**, pp**. 466-469, 2015.**

[2] A Ait-Mlouk, F Gharnati, T Agouti, *"Application of big data analysis with decision tree for road accident"*, Indian Journal of Science Technology, Vol. **10**, Issue.**29**, **pp. 1-10, 2017.**

[3] N Rajyaguru, M Vinay, *"A comparative study of big data on mobile computing"*, Indian Journal of Science and Technology, Vol. **10**, Issue.**21**, pp. **1-7, 2017.**

[4] A Kamilaris, A Kartakoullis, B X. F Prenafeta, *"A review on the big data analysis in agriculture"*, Computer and Electronics in Agriculture**,** Vol. **143,** pp. **23-27, 2017.**

[5] Dean J and Ghemawat S (2008) MapReduce: Simplified Data Processing on Large Clusters. 137-150.

[6] Dede E, Govindaraju M, Gunter D, Canon R S, Ramakrishan L (2013) Performance evaluation of a MongoDB and hadoop platform for scientific data analysis. 4th Workshop on Scientific Cloud Computing, ACM, pp. 13-20.

[7] Nunan D, Domenico M D (2013) Market research and the ethics of big data. *International journal of market research,* **55**(4):**505-520.**

[8] Ozarkar K, Rajani R (2014) Optimization technique for efficient dynamic query forms with NoSQL. *International journal of science and research,* **3**(11):**2041-2044.**

[9] Bhosale H S, Gadekar D P (2014) A review paper on big data and hadoop. *International Journal of Scientific and Research Publications*, **4**(10):**1-7.**

[10] A D Arasteh, D Mohammadpur, M Meghdadi, *"MapReduce based implementation of aggregate functions on Cassandra",* International journal of electronics communication and computer technology, Vol. **4**, Issue.**3**, pp**. 604-609, 2014.**

[11] R Zuech, M T Khoshgoftaar and R Wald, *"Intrusion detection and big heterogeneous data a survey"*, Journal of Big Data, Vol.**2**, Issue.**3**, pp**. 2-41, 2015.**

[12] Z Mo, Y Li, *"Research of big data based on the views of technology and application",* American journal of industrial and business management, Vol.**5**, pp. **192-197, 2015.**

[13] V S Thiyagarajan, A Ayyasamy, *"Privacy preserving over big data through Vssfa and Map-Reduce framework in cloud environment",* Indian Journal of Wireless Personal Communication, Vol. **97**, Issue.**4**, pp. **6239-63, 2017.**

[14] K Abouelmehdi, H A Beni and H Khaloufi, *"Big healthcare data: preserving security and privacy",* Journal of Big Data, Vol. **5**, pp. **1-18, 2018.**

[15] M S A Khan, H Jamshed, S Bano, N M Anwar, *"Big data management in connected world of Internet of things",* Indian Journal of Science Technology, Vol. **10**, Issue.**29**, pp. **1-9, 2017.**

[16] V. M A Martin, K David, A Vignesh*, "Big Data and its challenges",* International journal of scientific research in computer science, engineering and information technology, Vol. **3**, Issue.**3**, pp**. 533-538, 2018**.

[17] M Chevalier, M E Malki, A Kopliku, O Teste, R Tournier, *"Implementing Multidimensional Data Warehouses into NoSQL",* ICEIS, Vol. **1,** pp. **172-183, 2015.**

[18] L Kumar, S Rajawat, K Joshi, *"Comparative analysis of NoSQL (MongoDB) with MySQL Database",* International Journal of Modern Trends in Engineering and Research, Vol.**2,** Issue. **5**, pp. **120-127**, **2015.**