International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 15 (2017) pp. 5181-5186
© Research India Publications. http://www.ripublication.com

Computation and Management of Bigdata Using Cloud Map Table Framework

Sheikh Ikhlaq [1] and Bright Keswani [2]

[1]Research Scholar, Department of Computer Sciences, Suresh Gyan Vihar University, Jaipur, India.

[1]Orcid Id: 0000-0002-7175-6214

[2]Associate Professor, Department of Computer Sciences, Suresh Gyan Vihar University, Jaipur, India.

Abstract

In this paper a solution termed as CMT (Cloud Map Table) based on implementing Data warehouse technology into the Cloud based BigData processing framework which will not only prove better Data management solution but will provide Agility and multidimensionality feature to the data processing ability. Since the BigData technologies are still in their initial phase this research is based on exploratory design. Text study and latest approaches were studied. Vast study of Data warehousing, Algorithms was done to find out the optimal approach to implement both for the design. This study found that Data Management has always been an issue while dealing with BigData which creates problems with other fronts like Data Virtualization, Security, Network Intrusion, Query Support, Scalability, Cost benefits and ease of use. The novelty of this study led to design an optimal solution for Data Management problem which will directly affect the other issues also. This will prove healthy for AI, Data mining, machine learning and neural networks, till now new techniques on BigData technologies were not available go in hands with pervious methods of data mining, because of scattered heterogeneous data this was not possible in present BigData processing technologies. With Proposed design for Bigdata computation scattered and heterogeneous data is managed so well that all previous methods of data mining can be implemented on it with the scope to adjust with upcoming methods.

Keywords: BigData, Data warehousing, Data Mining, Agility, Data Management

INTRODUCTION

As of now there are various technologies to process BigData which include MapReduce, Hadoop, NoSQL, HPCC, Mobile and Cloud. The main Problem to implement these technologies is that of cost and scalability. Another major problem being faced by these technologies is that they don't support any a proper Data Management. Data is scattered all around, which makes it very difficult to process and attain desired results with agility. Processing of this data with respect to Data Mining and machine learning is not achievable as no present technologies work with technologies of Big Data processing. The Cost and the Scalability issue is well handled by Cloud based technology. As data, here is heterogeneous in nature and can be stored or processed by any DBMS Technology, NoSQL provides solution for it. Huge heaps of data are well processed by Hadoop. The only issue that remains to be solved is we need to have a data management technique in Cloud for the processing Big Data in a cheap and effective manner. The effect of Data Management problem is that it gives rise to many other problems like Data Virtualization, Network Intrusion, Security, Ease of use, and Query Support. As already mentioned BigData technologies don't provide any support for Multidimensional view of data which means that there is no support to create or find relations, co-relation and patterns with existing technologies to do so [1].

One of the best ways to manage Big Data can be found in the concept of Data Warehousing. The Data is generally not in the correct format to support a decision-making process in a Business. One of the main goals of implementing a data warehouse is to turn the wealth of data into information that can be used in the daily decision making process. A data warehouse integrates large amounts of enterprise data from various and independent data sources consisting of operational databases into a common repository for querying and analysing. Data warehousing will gain critical importance in the presence of data mining and generating several types of analytical reports which are usually not available in the original processing systems. A data warehouse is a collection of data copied from other systems and assembled into one place. Once assembled it is made available to end users, who can use it to support a plethora of different kinds of business decision support and information collection activities [2]. W.H. Inmon (1993), in his landmark work Building the data Warehouse, offers the following definition of a data warehouse: "A data warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management’s decision making process” [3]. Here we can customize the Data warehouse in such a way that it processes the data then splits the data so that we can compute or process the data per the need. Splitting of data is same as Hadoop does to process it by the machine or infrastructure that has larger capacity of processing more that the split size of data. The reason behind the splitting is that when we have huge amounts of data to process and we don’t have such a processing capability we must split the data and process it otherwise it will be garbage. According, to the Algorithms, given a function to compute on n inputs the *Divide –and-Conquer* strategy suggests splitting the input into k distinct subsets, *1< k<=n,* yielding k sub problems. These sub Problems must be solved, and then a method must be found to combine sub solutions into a solution of the hole. If the sub problems are relatively large, then Divide-and-conquer strategy can possibly be reapplied. Often the sub problems resulting from the divide and conquer design are of the same type as that of original problem. For those cases the reapplication the Divide- and-Conquer principle is naturally expressed by a recursive algorithm. Now smaller and smaller sub Problems of the same kind are generated until eventually sub Problems that are small enough to be solved without splitting are produced. already mentioned BigData technologies don’t provide any support for Multidimensional view of data which means that there is no support to create or find relations, co-relation and patterns with existing technologies to do so [1].

One of the best ways to manage Big Data can be found in the concept of Data Warehousing. The Data is generally not in the correct format to support a decision-making process in a Business. One of the main goals of implementing a data warehouse is to turn the wealth of data into information that can be used in the daily decision making process. A data warehouse integrates large amounts of enterprise data from various and independent data sources consisting of operational databases into a common repository for querying and analysing. Data warehousing will gain critical importance in the presence of data mining and generating several types of analytical reports which are usually not available in the original processing systems. A data warehouse is a collection of data copied from other systems and assembled into one place. Once assembled it is made available to end users, who can use it to support a plethora of different kinds of business decision support and information collection activities [2]. W.H. Inmon (1993), in his landmark work Building the data Warehouse, offers the following definition of a data warehouse: “A data warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of data in support of management’s decision making process” [3]. Here we can customize the Data warehouse in such a way that it processes the data then splits the data so that we can compute or process the data per the need. Splitting of data is same as Hadoop does to process it by the machine or infrastructure that has larger capacity of processing more that the split size of data. The reason behind the splitting is that when we have huge amounts of data to process and we don’t have such a processing capability we must split the data and process it otherwise it will be garbage. According, to the Algorithms, given a function to compute on n inputs the *Divide –and-Conquer* strategy suggests splitting the input into k distinct subsets, *1< k<=n,* yielding k sub problems. These sub Problems must be solved, and then a method must be found to combine sub solutions into a solution of the hole. If the sub problems are relatively large, then Divide-and-conquer strategy can possibly be reapplied. Often the sub problems resulting from the divide and conquer design are of the same type as that of original problem. For those cases the reapplication the Divide- and-Conquer principle is naturally expressed by a recursive algorithm. Now smaller and smaller sub Problems of the same kind are generated until eventually sub Problems that are small enough to be solved without splitting are produced.

```
Algorithm DAndC (P)
{
If Small (P) then return S(P);

Else
{ Divide P into smaller instances P1, P2,….., Pk , K>= 1;

Apply DAndC to each of these problems;
Return Combine (DAndC (P1), DAndC(P2) ……DAndC(Pk));
}
}
```

Control Abstraction for Divide and Conquer

Here DAndC is the name of the Algorithm,

P is the problem to be solved,

Small (*P*) is the Boolean-Valued function that determines whether the input size is small enough that the answer can be computed without splitting. If this is so, the function S is invoked. Otherwise the problem p is divided into smaller problems. The sub problems *P*1, *P*2… *P*k is solved by recursive application of DAndC. Combine is the function that determines the solution to P using the solutions to the K Sub Problems. If the size of P is n and the sizes of the k sub Problems are $n1$, $n2\ldots$ nk, respectively, then computing time of DAndC is described by the recursive relation

$$T(n)=g(n) \quad n \text{ small otherwise divided.}$$

$$T(n1) + T(n2) + \ldots\ldots\ldots\ldots + T(n\text{k}) + f(n)$$

Where $T(n)$ is the time for DAndC on any input of size n and $g(n)$ is the time to compute the answer directly for small inputs. The function $f(n)$ is the time for dividing P and then combining the solutions to sub problems. For divide and conquer based algorithms that produce sub problems of the same type as the original problem, it is very natural to first describe such algorithms using recursion.

The complexities of many divide and conquer algorithms is given by recursive relation of the form:

$$T(n) = T(1) \quad n = 1$$

$$aT(n/b) + f(n) \; n > 1$$

Where a and b are known constants. We assume that $T(1)$ is known and n is the power of b (i.e., $n = \text{b}^{\text{k}})^{\text{i}}$ [4].

As most of the work on Data Mountains, is of searching and sorting which means more sorted the data is more refined results will be. All this is in agile manner. Implementing Customized Data warehouse with Divide-and- conquer strategy on cloud will give as a desired solution to the data management problem and its other effects as discussed earlier. Also, it will prove boon to Data mining. Hence this study provides a solution based on mentioned methodology termed as CMT (Cloud Map Table).

THE CMT FRAMEWORK

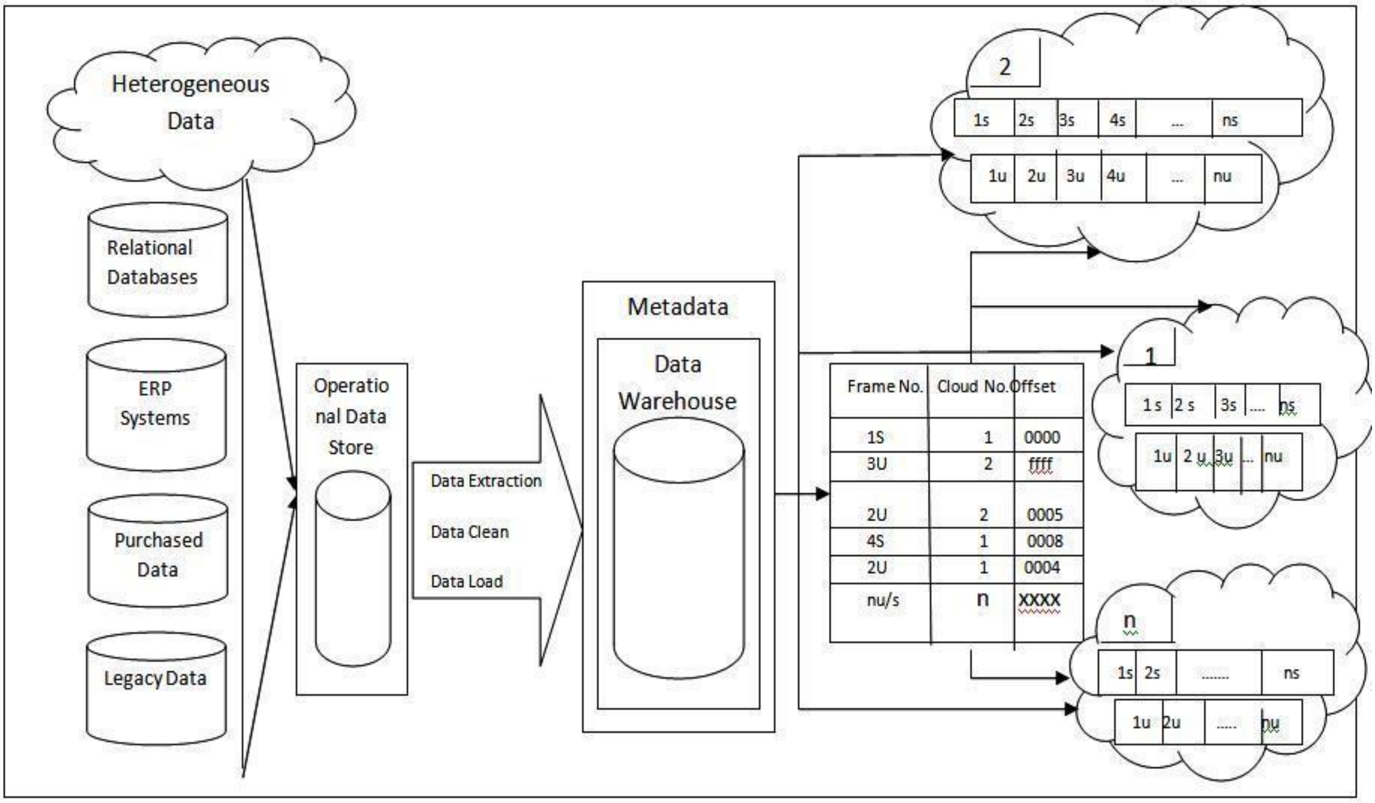
Customized Data warehouse with CMT in cloud has been designed to empower Researchers or Data Scientists while presenting them with a tool enable research involving Data Mining, Extraction due to effective Data Management, to perform very easily. With the Knowledge that CMT would be used by Researchers and Data Scientists as a tool, we designed CMT with the following goals in mind.

1. Store Data in a very efficient manner for the use in various BigData Processing.
2. Allow Management of data using Customized Data Warehouse by splitting data in terms of type i.e. whether structured or unstructured.
3. Present users with Interface in which data that can be viewed in multiple Dimensions.
4. Use existing Techniques of Data Mining on this alienated and managed data.
5. Improve data retrieval and processing rate i.e. improve agility.
6. Make full use of cloud technologies for cheaper resources.
7. Provide improvement in BigData processing on various fronts like Data visualization, Cost, Ease of use, Security, Query Support and network intrusion.

CMT BASED COMPUTATION

Keeping Fig.1 in focus, we can understand the concept of CMT. As of now no major work had been done on Data Management issue in all major technologies of BigData Processing which include Hadoop, MapReduce and Cloud. In All the said technologies Data was scattered which had major impact on various fronts like Data Visualization, Scalability, Query Support, Security, Network Intrusion, Agility and Cost.

In CMT based computation, we have a concept of Divide-and-Conquer, as we know to process these huge mountains of Data for wealth otherwise if left unexplored will be nothing but garbage. To process such huge amount of data we need such processing capabilities, which we do possess but come higher costs and have same problem of Data management as discussed earlier portions. So, what we do is that we use cheap technology available i.e. Cloud and then to deal with its data management issue we use customized data warehouse, as Data warehouses are well known for data management. In this Customized data warehouse, we introduce a three Colum table termed as CMT (Cloud Map Table). Here Data warehouse does not put the refined data into the traditional databases but into cloud. Cloud has two sections, in one section data that is structured is stored while as data that is unstructured in stored in another portion.



Frame No.	Cloud No.	Offset
1S	1	0000
3U	2	ffff
2U	2	0005
4S	1	0008
2U	1	0004
nu/s	n	xxxx

Figure 1: Customized Data Warehouse with Implemented CMT in Cloud for Bigdata Computation

This means that the data is being divided in such a way that it is easy to manage and then process it. Since data can be of variable sizes we make the portions into the cloud of variable length(s) termed as frame (s) with alphanumeric values. Number of clouds can be increased in accordance to the need; each cloud has a numeric entry /value termed as cloud number. While Storing Data on cloud(s) some entries are made into the CMT (Three Colum table), these entries include, Cloum1 Specifies the frame number with Alphabets U or S assigned to the numeric value, here U means Unstructured i.e. data is in unstructured portion and S means that the data is in Structured portion, Colum 2 specifies the Cloud number, and Colum 3 specifies the offset, this means that there will be no size limit to the frame, offset will determine the displacement from the base address, helping in locating the particular frame.

WORKING OF CMT

Seeking the CMT we come to know that data is very well managed. This means that whenever we need to process the data we can easily locate the data using the CMT. Once the Data is found it can be processed in accordance to the need. If data is Structured, it can be stored in either MRDB (Multi Relational Database) or MDDB (Multidimensional Database). As in Case with Unstructured data which is Stored in U Section of the Cloud, Data can be stored in any schema i.e. schema can be made in accordance to the need i.e. On the Fly concept, will be used. Thus, NoSQL comes to play.

Since we have upgraded the Infrastructure as a service (IaaS) aspect of cloud Computing rest aspects which include Software as services (SaaS), Platform as service (PaaS) and Hardware as service (HaaS) need no to be altered in their working as they are flexible enough to work with the change in (IaaS) [5].

Now users who are well versed with the working of Data Warehouses can easily use our tool to manage their data, view them multidimensionaly, process them and attain results in an agile and secure manner. Having a close on the working of CMT it means that we can visualize the data in a very efficient manner, have a better query support, Secure transactions, ease of use, and highly scalable that too with a very minimal cost. We can use both Traditional techniques of data mining which we couldn’t earlier with latest drill technologies like Apache drill and Google Dermel with Adhoc Query Support [6][7]. In CMT we have an option of frame replacement which will use same replacement algorithms, if needed, as Page replacement algorithms to keep the CMT lighter i.e. if the Frame gets replaced then CMT entry will be deleted.

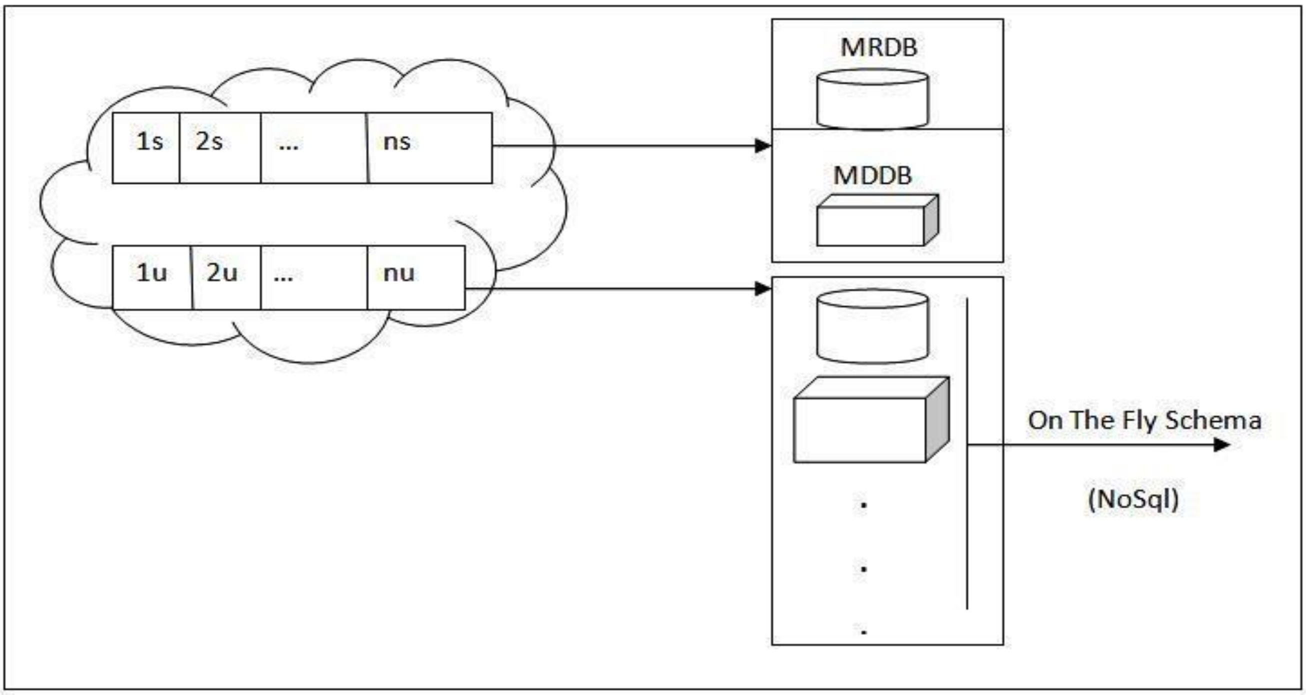


Figure 2: Inside Look of Frame

Fig.2 gives us an inside look at the frame to understand the working of CMT and frame design. Also, to mention Hadoop may be also used which will now use more refined data and perform Map and Reduce function. Here user can process the data without knowing the location of data, same is if we use cloud technology and implement Map and Reduce through it. It is worth mentioning that other components of Hadoop framework have compatibility to be implemented in this design, as when needed.

AREAS OF PERFORMANCE EVALVATION FOR CMT

CBT (Cloud Based Table) works on one of the issues i.e. Data management of data in the cloud. This makes Cloud a choice to be used for Big Data processing. Data management has led to the solution of various problems associated with the various technologies and the cloud itself. Our Performance Study with Various technologies is based on various parameters and the results are calculated, when CBT is applied to the cloud.

Scalability: As the size of the computation increases so does the size of Hadoop cluster increases which means data will be more scattered and the loss of master will be the loss of whole infrastructure. But when it comes to other technologies like High Performance Computing Architecture (HPCC), Cloud technology and the mobile based agent, the growth of computation hardly matters but the data scattering remains the issue. Since CBT is applied to cloud the issue of data scattering is sorted out and now the data is more organized.

Storage: It is evident that storage is a problem with Hadoop and mobile based technology but in cloud and High Performance Computing Architecture (HPCC) storage is not an issue. The only issue with these two in terms of storage was of how to store this data perfectly or in an organized manner. With the introduction of CBT to cloud this problem is entirely solved and cloud is the king of storage.

Fault Tolerance: Hadoop is the best survivor of fault tolerance as the data is spread over the nodes and there is the redundancy of data. Mobile Agent technology performs modestly when it comes to fault tolerance. HPCC is good with fault tolerance. Now as the CBT is applied to the cloud fault tolerance is dealt in a very effective manner as data can be made redundant with proper management which makes retrieval very easy and effective.

Virtualization: With Big Sheets and the tools associated with it: Tag Cloud, Pie Chart, Map, Heat Map and Bar Chart, Hadoop did well with data virtualization, these tools can be used with cloud. Also As the data is more organized in Cloud using CBT which makes it to provide better virtualization as we can have visuals in multidimensionality also we can drill down deep into the data, thus making data mining an easy job with existing technologies.

Cost: When it comes to cost all the existing technologies fail and this becomes the reason why majority is not using them. Cloud is already proving to be a boon for such majority and now computation of BigData being done by cloud will outperform the existing technologies soon.

Agility: Agility which is the need of an hour and can't be provided by the existing technologies. But With the introduction of CBT in Cloud we have the power of data agility, attaining results in both the worst-case scenarios and the best scenarios with best possible retrieval times [8].

Data Management: CBT with cloud is the only technology that works on data management in an effective manner. No other technologies have an upper hand when it comes to data management. It is due to data management and the solutions to other problems that CBT with Cloud will become a choice.

Ease of use: All the technologies need professional to use them but as we have implemented CBT to cloud which gives it an interface of the thing that was being used by everyone, starting from low level management to high level management. This means that this technology needs no special people to handle them.

Query Execution/Support: In CBT data warehouse queries, can be carried out in split execution which will give the efficient results. Complex Queries can be split and better results can be attained.

Network Intrusion/Security: There are various security challenges with possible Solutions that other technologies follow, which can be followed by cloud. Challenges and solutions and their solutions include: a) Network level with File encryption and Network encryption as the solution. b) Authentication level with logging as the Solution c) Data level with software format and Node maintenance, Node authentication as the solutions d) Generic level with honey pot nodes, rigorous testing of MapReduce Jobs, layered framework of assuring cloud, Third party secure data publication to cloud, Access control as the solutions [9]. There are many cryptographic techniques/methods which can be applied: a) Homomorphic encryption b) Verifiable Computation c) Multiparty Computation. Scope for Multiparty Computation is huge [10]

CONCLUSION

With the introduction of CBT in cloud data management becomes efficient as both structured and unstructured data are put and sorted in order in cloud. Cloud is divided into portion named as frames known as Cloud frames which hold the data both structured and unstructured. Cloud frame entries are put into a table know as cloud based table. These entries are then used to retrieve the data i.e. searching is done from here. Implementing warehouse with cloud and CBT help go give multidimensional view to the data stored. It helps to perform ETL (Extraction, Transformation, Loading). This helps the data mining to be performed with the existing technologies. Data Agility becomes achievable with CBT. Implementation of CBT for various issues can be designed and implemented. Cloud Frame management will be the future scope of work.

REFERENCES

[1] Ikhlaq, S., & Keswani, B. 2016.”Computation of BigData in Hadoop and Cloud Environment,” IOSR Journal of Engineering, 6(1), pp.31-39.

[2] Berson, A., & Stephen J, S., 1997, “Data Warehousing, Data Mining , And OLAP,” McGraw-Hill, Newyork

[3] Immon, W., 1996,”Building the Data Warehouse,” John Wiley.

[4] Elmis, H., Sartjaj, S., & Sanguthevar, R., 2011, “Fundamentals Of Computer Algorithms”, Universities Press Limited, India.

[5] Armbrust, M., Fox, A., Griffith, R., Joseph, A., Katz, R., Konwinski, A., Zaharia, M., 2010,”A view Of Cloud Computing,” Communications Of the ACM,53(4), pp.50-58.

[6] “Drill Introduction”, 2017, https://drill.apache.org/docs/drill-introduction/.

[7] “Introducing JSON”, 2016, http://www.json.org

[8] Ikhlaq, S., & Keswani, B., 2016,” Enhanced Approach for BigData Agility,” International Journal of Advanced research in Computer Science and Software Engineering, 6(1), pp. 579-584.

[9] Inukollu V, N., Arsi, S., & Ravuri S, R., 2014, “Security Issues Associated With Big Data In Cloud Computing,” International Journal of Network Security & Its Applications (IJNSA), 6(3).

[10] Yakoubov S, Gadepally, 2014,”A Survey Of Cryptographic Approaches To Securing Big- Data Analytics,” In The Cloud. Proceedings of High Performance Extreme Computing Conference (HPEC), Waltham, Massachusetts, pp. 1-7.