APPROACH FOR URBAN TERRITORY PLANNING BASED BIG DATA

LOUARDI BRADJI, KHAOULA TABET, MOHAMED RIDDA LAOUAR
Department of MI, University of Tebessa, Algeria
bradjilouardi@yahoo.fr

ABSTRACT

Over the last few years, urban data has become more complex for the reason that large amount of data are being available lately, along with the rapid change of technologies and mobile applications and new problems have discovered. Therefore, urban territory planning organizations have believed that urban data analytics tools are really important subject in order to manage a large amount of complex data, which can lead to improve urban territory planning and help urban practice to reach a high level of efficiency and work flow accuracy, if these data analytics tools applied correctly, but the questions are how urban organizations are applying these tools today, and how to think about it's future use? This paper gives a response to this question by proposing an approach that combines big data technologies such as NoSQL systems ,Hadoop and MapReduce. We have used current technologies to implement and validate the proposed approach which offers the ability to handle large data to achieve better decision in urban territory planning domain.

KEYWORDS: Big Data, urban territory planning, NoSQL, Hadoop, Cloud, MapReduce.

1 INTRODUCTION

Big Data is a hot topic in urban territory, health care, biomedical researches. The increased usage of the term "Big Data" in theses researches is indicative of the emerging importance of large-scale data sets, and there is also an increasing awareness of the role that big data can play in these researches [1].

Today's urban organizations are moving from volume-based business into value-based business, which requires an overwork from administrators and urban individual to be more productive and efficient. This will improve urban practice (to provide a safe, organized, and enjoyable home and work life for residents of both new and established towns) [2].

Since urban information systems generate enormous amounts of records every time, it seems the world is reaching the level of data overload. It is obvious now, that in order to process such volumes of data an enormous capacity is required in terms of storage and computing resources. Whereas the growth of capacity is limited by evolution of hardware and technologies, the growth of the data volume is in fact unlimited.

Getting more specific, nowadays many organizations has adopted and broadly use information systems running on technological platforms, many their agendas has become addicted to data.

While big data holds significant promise and advantages for improving urban territory planning, we did not notice in urban territory planning literature, approaches which take into account these advantages especially with the volume of urban data which has become unmanageable and unfathomable. Therefore, to take benefit of new technologies: big data (NoSQL databases, Mapreduce/Hadoop), Cloud, Modern Data Platforms and methodologies, Urban Territory Planning has to do somewhat that it has never done previously. It has to be an early adopter.

To this end, the main objective of this paper is to propose an approach based in big data technologies to develop a richer profile about what is promising and what is not promising to deliver value based urban territory planning.

The reminder of this paper is structured as follows: In sections 2 and 3 we review background information and different definitions, characteristics and types of big data technologies being used for urban informatics. This is followed in Section 4 by a discussion of research approaches and architectures in Urban territory planning that involve the use of big data. Our approach based big data for urban communication planning are described in Section 6. After the description of our case study in Section 7, Section 8 will present the implementation and validation of our approach using the case study. Section 9 concludes this article. Finally, the last Section present the limitations and perspectives of our work.

2 URBAN TERRITORY PLANNING

Urban planning is a branch of architecture that focuses on organizing metropolitan areas. Made up of several different fields, from engineering to social science, this practice was

developed to correct problems caused by cities expanding spontaneously, without planning. At its core, city planning aims to provide a safe, organized, and enjoyable home and work life for residents of both new and established towns. Today, some of the largest concerns of urban planning are building locations, zoning, transportation, and how a town or city looks. Planners also try to eliminate run down areas and prevent their development, as well preserve the natural environment of the area [2].

3 BIG DATA DEFININTION

Big data describe a new generation of technologies and architectures, designed to extract value from large volumes of a wide variety of data by enabling high-velocity capture, discovery and analysis. This world of big data requires a shift in computing architecture so that researchers can handle both the data storage requirements and the heavy server processing needed to analyze large volumes of data in a secure manner. Most of the big data surge is unstructured information and is not typically easy for traditional databases to analyze it [3, 4, 5].

Big data is also characterized by features, known as 5V's. These features are: volume, variety, velocity, variability, and veracity [6]. **Volume** is the most obvious of the three, referring to the size of the data. The massive volumes of data have required scientists to rethink storage and processing paradigms in order to develop the tools needed to properly analyze it [7]. **Velocity** addresses the speed at which data can be received as well as analyzed [8]. **Variety** refers to the issue of disparate and incompatible data formats [9, 10]. **Veracity** (Uncertainty of data (Data Incompleteness)) measure the reliability and trust of data. **Validity** means the data correct and accurate for the intended use. Clearly valid data is key to making the right decisions [11]. Ultimately it is the **Value** (Turn Big data into values else useless (Business Perspective)) that will make Big Data a real force of change in urban territory planning and to derive the significance while making use of Big data [12]. With the advent of changing financing, analytics has taken on new importance [13].

4 BIG DATA TECHNOLOGIES

There are several Big data analytics platforms available. In this section, we discuss recent advances in Big data analytics platforms.

MapReduce is currently the popular paradigm for dealing with Big Data. The landscape is very rich and complex. MapReduce framework represents a pioneering schema for performing Big data analytics. It has been designed for a dedicated platform (such as a cluster). There are three implementations of the MapReduce framework. The first implementation with a proprietary license was developed by Google. The other two implementations: **Hadoop** and Spark are open-source [6].

Hadoop allows us to pull all kinds of structured, semi- structured, and unstructured data into "files" and tag them with keywords [9]. The main design advantage of Hadoop is its fault-tolerance. In fact, Hadoop has been designed with the assumption of failure as a common issue in distributed systems. Therefore, it is robust against failures commonly occur during different phases of execution [14].

The MapReduce storage functionality for storing input, intermediate, and output data is supported by distributed file systems such as **Hadoop Distributed File System** (HDFS) and Google File System (GFS), developed specifically for this framework. Every MapReduce workflow contains three subsequent phases that are Map, Shuffle, and Reduce. In the Map phase, the Map function implemented by the user is executed on the input data across the computational resources. The input data are divided into partitions and stored in a Distributed File System (DFS). Each Map task works on a partition of data from the distributed file system and produces intermediate data that are stored locally on the worker machines. Then, the intermediate data are used by the Reduce phase [6].

For the sake of fault-tolerance, HDFS replicates data blocks in different racks, thus, in case of failure in one rack, the whole process would not fail. A Hadoop cluster includes one master node and one or more worker nodes. The master node includes four components namely, JobTracker, TaskTracker, NameNode, and DataNode. The worker node just includes DataNode and TaskTracker. The JobTracker receives user applications and allocates them to available TaskTracker nodes, while considering data-locality. JobTracker assures about the health of TaskTrackers based on regular heartbeats it receives from them. Although Hadoop is robust against failures in a distributed system, its performance is not the best amongst other available tools because of frequent disk accesses [15].

When executing Map Reduce programs the data is typically stored in HDFS and Hadoop optimizes the data communication by scheduling computations near the data using the data locality information provided by the HDFS file system. Hadoop follows a master node with many client workers approach and uses a global queue for the task scheduling, achieving natural load balancing among the tasks. Hadoop performs data distribution and automatic task partitioning based on the information provided in the master program and based the structure of the data stored in HDFS. The Map Reduce model reduces the data transfer overheads by overlapping data communication with computation when reduce steps are involved. Hadoop performs duplicate execution of slower tasks and handles failures by rerunning of the failed tasks using different workers [6, 10, 16].

The Big-Data ecosystem has also seen the emergence of the **NoSQL** distributed databases such as the Amazon's Dynamo, Cassandra, MongoDB. These emerged mainly due to the limitations (in terms of fault-tolerance and performance) of the HDFS. Some of the NoSQL DBs including Dynamo and Riak are key-value stores, while MongoDB and CouchDB are document stores. The third

category is the columnar databases such as BigTable and Cassandra, with the last category being the graph databases such as Neo4j. the most NoSQL systems sacrifice consistency because of the CAP (Consistency, Availability, and Partition tolerance) theorem [17, 18]. The main characteristics that distinguish the NoSQL model from the traditional RDBMS one are partitioning of data and data replication [19]. NoSQL databases provide an efficient framework to aggregate large volumes of data while relational databases like SQL have a limitation when it comes to data aggregation (Data aggregation becomes impossible on very large volumes of data when it comes to memory and time consumption), which is used for business intelligence and data mining [20].

The big data management challenges include data quality, data streams, dynamically evolving data, data heterogeneity and data modeling, multi-model databases, client and query interfaces, data compression, data encryption, access control and authorization, and deployment on cloud-hosted cluster computers. One task that crosscuts all of the above challenges is identifying a subset of Big Data that has high value. This requires separating the data that is contaminated by spam, noise, and bias from that which is uncontaminated [21].

5 SOME RELATED WORKS

[22] Designed and validated a potential architecture for a disruptive technology called Big Data, and evaluate its impact on the existing Business Intelligence and Geographic Information Systems which is a technology architecture able to manage data for urban sensing, specifically geo-referenced social data, also **[23]** investigated how different database systems can effectively handle the heterogeneous and large amount of data of the Internet of Things on the cloud , in order to meet the increasing demand on load and performance, **[24]**described that HDFS was originally designed for high-latency high-throughput batch analytic systems like MapReduce, and that Hbase improved its suitability for real-time systems low-latency performance, To sum up, we conclude that promising progresses have been made in the area of big data, but much remains to be done. Almost all proposed approaches are evaluated at a limited scale.

6 ARCHITECTURE OF THE PROPOSED APPROACH FOR URBAN COMMUNICATION OPERATORS

Algeria has three telephone operators (Djezzy, Mobilis, Oredoo), each one of those has its characteristics such as information about subscribers (such as identification number (id,number), name, age…etc.), offers and infrastructures. In our subject we try to propose an approach for urban communication operators based big data that integrate the big data technologies with urban territory project, therefore let's focus in the last mentioned characteristic which is the infrastructures (locals and antennas).

In figure 1, we explain the correlation between the communication domain and the urban territory planning.

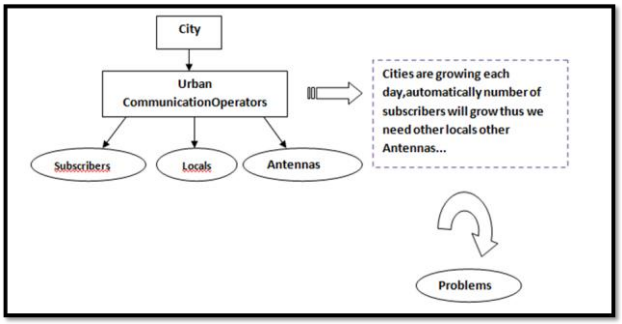


Figure 01: Urban Communication Process

According to figure 1 and our discussion with experts in communication, we resume the problems of communication as follow:

- Traditional RDBMS cannot support the large amount of data.
- Overcrowding of the customers Centers.
- The number of antennas will be insufficient.
- Telephone network congestion, and many other problems.
- ...

Therefore we tried to propose an urban approach based big data to resolve some problems in communication field in order to help operators for making better decision, in a simple way without facing any difficult problem, we can also apply this aproach in other field.

The bellow diagram in figure 2 present a reference architecture of our proposed approach.

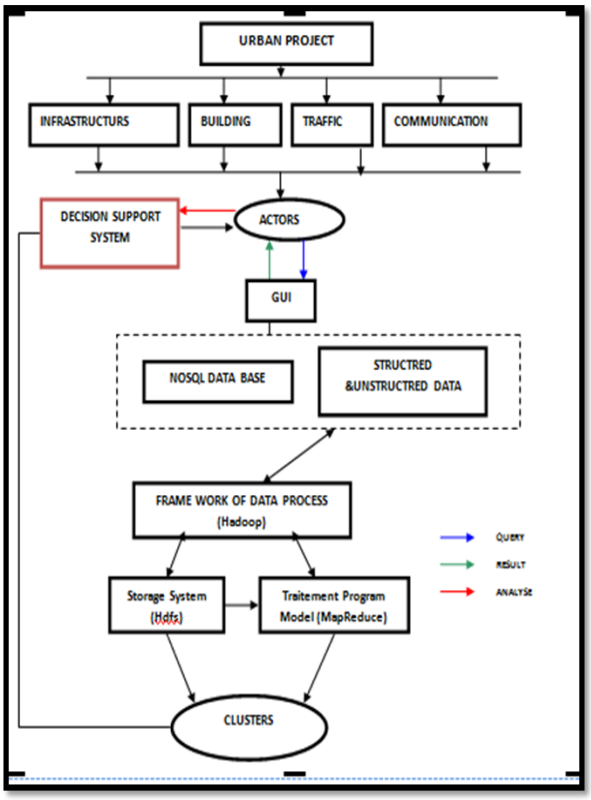


Figure 02: Global Architecture Of Proposed Approach

The principles components of our approach as presented in figure 2 are: Actors, Graphic User Interface (GUI), NoSQL databases, Data process, Program model and Storage system. The next subsection of this section explain in detail the role of each component.

6.1 ACTORE

The actor present the decision maker who's able to : (1) Entry the query, (2) Discuss and analyze results and (3) Make decision.

As it shown in figure 3, the user can be a director, workers, domain expert. Each one of them can use the application according to the situation.

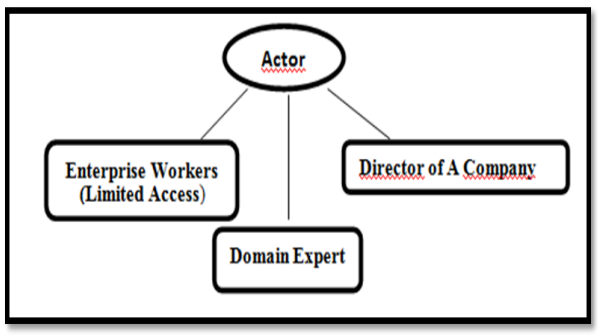


Figure 03: Functions of The Actor

6.2 GUI

The GUI present a computer program that enables a person to communicate with a computer through the use of symbols, visual metaphors, and pointing devices it allows the interaction between user and system. As it presented in figure 4, in our architecture the GUI is composed of two windows CRUD and MapReduce.the **CRUD** offers to the user the ability to **C**reate, **R**esearch, **U**pdate and Delete lines or fields.

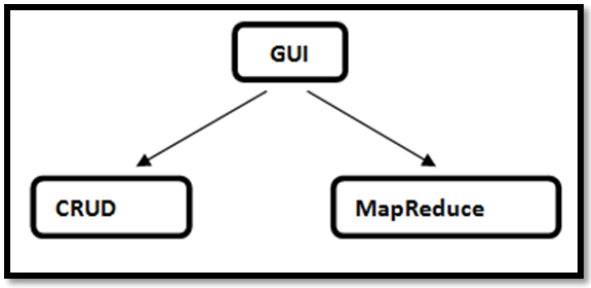


Figure 04: Functions of The GUI

The **MapReduce** allows the user to entry his query and get results to make decision about a situation. Because we'll use the technology of big data we must follow a NoSQL language.

6.3 NOSQL DATABASE

In our work we will try to use the following architecture (see figure 5):

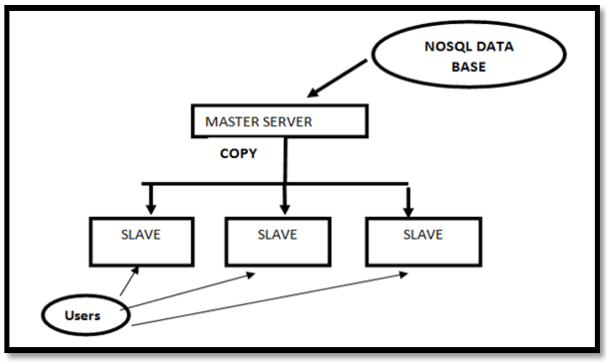


Figure 05: Big Data Architecture

6.4 FRAMEWORK OF DATA PROCESS (HADOOP)

As we have desribed above, Hadoop can process stores of both unstructured and structured data that are extremely large, very complex and changing rapidly. It provides a distributed file system and a framework for the analysis and transformation of very large data sets using the MapReduce , An important characteristic of Hadoop is the

partitioning of data and computation across many (thousands) of hosts, and the execution of application computations in parallel close to their data.

6.5 PROGRAM MODEL (MAPREDUCE)

It is an associated implementation for processing and generating large data sets, we can also say that the MapReduce Simplified Data Processing on Large Clusters , and allows developers to write programs that process massive amounts of unstructured data in parallel across a distributed cluster of processors or stand-alone computers.

6.6 STORAGE SYSTEM (HDFS)

It is designed to store very large data sets reliably, and to stream those data sets at high bandwidth to user applications. In a large cluster, thousands of servers both host directly attached storage and execute user application tasks. By distributing storage and computation across many servers.

6.7 CLUSTER

It is group of independent servers (usually in close proximity to one another) interconnected through a dedicated network to work as one centralized data processing resource, Clusters are capable of performing multiple complex instructions by distributing workload across all connected servers, clustering improves the system's availability to users, A failed server is automatically shut down and its users are switched instantly to the other servers.

6.8 EXPLANATION

The size of the generated data is very big, thus it won't fit on a single computer; for that reason we have distributed it across thousand of nodes ,in this case we will have a faster computation because we've got distributed data plus parallel execution ,we can do things we couldn't possibly do before ,that's the trick behind Hadoop.

Suppose I want to look for an image from thousand of files, the question here how Hadoop work? First of all, Hadoop has to know where the data is ?

The first step is to query the name node to find out all places where the data files located, the graph in figure 6 present the architecture HDFS (Hadoop Distributed Files System) which explain how it works.

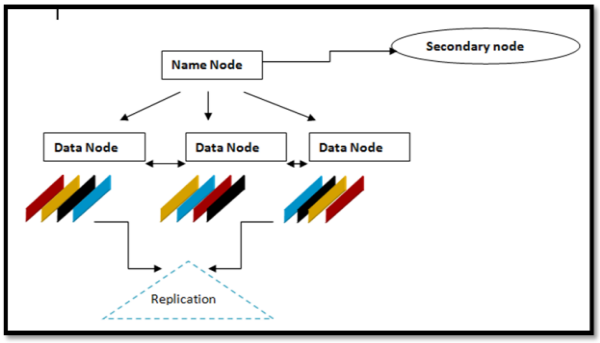


Figure 06: HDFS Architecture

As we we have explain above, the HDFS is composed of Name Node, Secondary Node, and Data Node. **Name Node:** It contains for each data the address of the node where it located. **Secondary Node:** It contains COPIES of data (Replication), if one node falls it doesn't affect result. **Data Node:** It contains the address of clusters which contains data.

Sometimes we need to communicate between nodes, in this case the Hadoop trick is called MapReduce through the Job Tracker. The Job Tracker has a unique role: Schedule the treatment and distribution in clusters and the overall distribution accordingly on the location of the block.

7 CASE STUDY

As we saw in the previous section our architecture can help to make better decision to facilitate the planning of an urban project, it is designated to multiple fields, here we tried to present each component in a simple way to let you all understand the principe of this architecture, therefore, to make it more understandable we tried to apply it in the sector of communication.

Nowadays the massive usage of phone calls, social networks, e-commerce, web pages and smart phones always connected on internet allowed any company to improve and re-moderate the technique, both products and services are now the result of an interactive communication, even implicit ,between customers and companies so in next section we will try to detail the use and the function of each component using a study case (Telephone Operators).

1st Use case: Mobile Subscribers in Tebessa

Line format: PhoneNbr, Full Name (First,Last), Age, Sexe, IdentityPN, IdentityPType, Adress

Operator/Age
Mobilis/Teenager
Mobilis/Adult
Oreedo/Teenager
Oreedo/Adult
Djezzy/Teenager
Djezzy/Adult

2nd use case: Operations

Operator/Utilisation
Mobilis/Call,
Mobilis/SMS
Mobilis/MMS
Mobilis/Internet3G
Oreedo/ Call
Oreedo /SMS
Oreedo /MMS
Oreedo/Internet3G
Djezzy/ Call
Djezzy/SMS
Djezzy/MMS

Line format: PhoneNbr, Full Name (First, Last) , CalledNbr, PhoneNbr SMS , Content_SMS, PhoneNbr_MMS ,Content_MMS, Consulted_Site, Consulted_SM

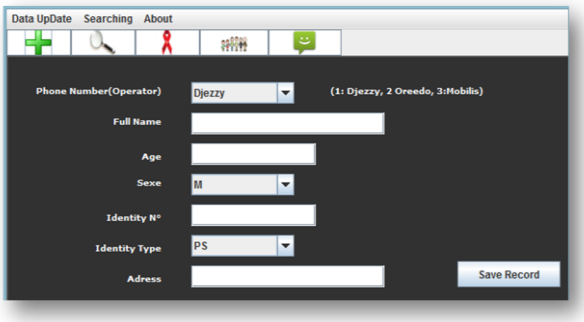
We create a JSON record which identifies the schema as follow:

```
{ "type": "record", "name": "SchemaName",
                "namespace": "avro",
   "fields": [ {"name": "Champ1", "type": "Type1",
                "default":Valeur1},
         {"name": "Champ2", "type": "Type2", "default":"Valeur2"},
         {"name": "Champ3", "type": "Type3", "default":"Valeur3"}
-
 ] }
```

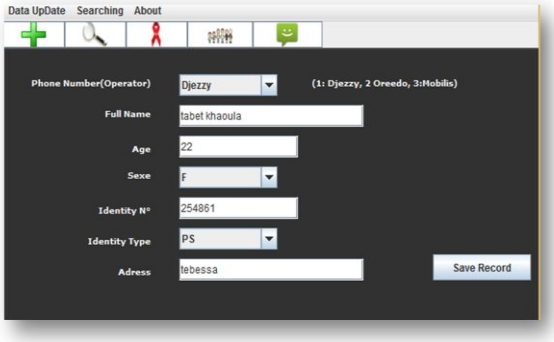
8 VALIDATION

Using simulator Oracle Big Data Lita which is a hight secure platform that offers different workloads to run Hadoop and NoSQL systems. It has many components such as Oracle Enterprise Linux 6.4, Cloudera's Distribution including Apache Hadoop, Oracle SQL Developer 4.0, Oracle JDeveloper 11g. After that we deploy clusters into single data centers. the last step is the demonstration of big data which shows the implementation of the use cases. It held use cases from the data sets used for testing to the data recovery using a data visualization tool and passing by treatment and data structuring.

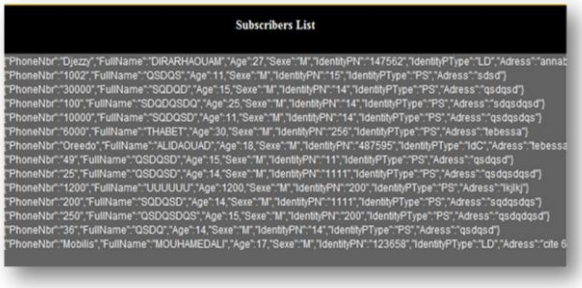
Using the above tools, we have developped application which contains all functions (add, serarch ,graphs.....). this applcation contain the follow window:



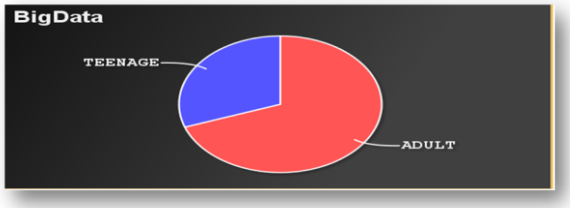
This window shows the execution of the button ""Save Record":



We can also display the list of subscribers using the button ""Subscribers List"".



The most important function is to display graphs (charts) which presents the result of the MapReduce program:



9 CONCLUSIONS

The proposed approach predicts a growth in all use cases associated with 'big data'. It also shows a growing interest

of related technologies discussed in this work, e.g. Hadoop and MapReduce, in-database analytics.

We also argue, that the reference architecture that we propose can help to discuss technologies already existing and newly emerging in the space, to reason about them, categorize them, map them to requirements and functional components and guide in applying them. Therefore, We consider this work successful within its scope and assuming the limitations discussed above.

10 LIMITATIONS & PERSPECTIVES

While the validation step proved the proposed approach relevant and provides good utility, there are still a number of limitations, which should be noted. The main issue in our work is the select of examplary case studies in a concrete project situation to apply and validate the proposed approach. As the reference architecture should be based on best practices and proven concepts from practice, this is definitely an issue, that cannot completely made up from taking those best practices from literature.

REFERENCES

[1] F. Martin-Sanchez, K. Verspoor, "Big Data in Medicine Is Driving Big", IMIA Yearbook of Medical Informatics 2014:14-20.

[2] Web site wiseGEEK.com paper written by Niki Foster 02 Jan 2015.

[3] B. Di Martino, R. Aversa, G. Cretella, A. Esposito, J. Kołodziej, "Big data (lost) in the cloud", Int. J. Big Data Intelligence, Vol. 1, Nos. 1/2, 2014 .

[4] Bo Li, boli"Survey of Recent Research Progress and Issues in Big Data", CSE. Wustlr.edu , 2013.

[5] Fabricio F. Costa, "Big data in biomedicine", Drug Discovery Today _ Volume 00, Number 00 _ November 2013.

[6] Pusala M. K., Salehi M.A., Katukuri J. R., Xie Y., Raghavan V. V., "Massive Data Analysis: Tasks, Tools, Applications and Challenges" . Chapter In book: Big Data Analytics, Publisher: Springer, January 2016.

[7] T. Davenport and J. Dyché, "Big Data in Big Companies", SAS Institute Inc. May 2013.

[8] Hassanien A., · Azar A. T., S. Vaclav, K. Janusz, H. Abawajy "Big Data in Complex Systems: Challenges and Opportunities", Studies in Big Data, Volume 9, Series editor: Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland, springer, 2015.

[9] Laura B. Madsen, "Data-Driven Healthcare: How Analytics and BI are transforming the industry", Published by John Wiley & Sons, Inc., Hoboken, New Jersey, 2014.

[10] S. Landset, Taghi M. Khoshgoftaar, T. Hasanin, "A survey of open source tools for machine learning with big data in the Hadoop ecosystem", Journal of Big Data (2015) 2:24.

[11] J. Trevor, William A. Young, Gary R. Weckman, "Defining, Understanding, and Addressing Big Data." IJBAN 3.2 (2016): 1-32. Web. 16 Jun. 2016.

[12] B. Shankar, P. Mishra, S. Dehuri, E. Kim, G. Wang, "Techniques and Environments for Big Data Analysis: Parallel, Cloud, and Grid Computing", Studies in Big Data, Volume 17, Series editor: Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland, 2016.

[13] S. P. M. Claps , "Bigger Data for Better Healthcare", Sep 2013, IDC Health Insights.

[14] C. Lam, "Hadoop in Action", Manning Publications Co., Greenwich, CT, USA, 2010.

[15] A. Shinnar, D. Cunningham, V. Saraswat, B. Herta. "M3r: Increased performance for in-memory hadoop jobs.", Proceedings of VLDB Endowment, 5(12):1736–1747, Aug 2012.

[16] T. Gunarathne, T. Wu, J. Qiu, G. Fox, "Cloud Computing Paradigms for Pleasingly Parallel Biomedical Applications", HPDC'10, June 20–25, 2010, Chicago, Illinois, USA.

[17] V. Srinivas Agneeswaran, "Big-Data – Theoretical, Engineering and Analytics Perspective", First International Conference, BDA 2012, New Delhi, India, December 24-26, 2012, Proceedings, LNCS 7678, pp. 8–15.

[18] SM. Freire, D. Teodoro, F. Wei-Kleiner, E. Sundvall, D. Karlsson, P. Lambrix, "Comparing the Performance of NoSQL Approaches for Managing Archetype-Based Electronic Health Record Data.", PLoS ONE 11(3): e0150069. doi:10.1371/ journal.pone.0150069 Editor: Kim W Carter, University of Western Australia, 2016.

[19] L. Rocha, F. Vale, E. Cirilo, D. Barbosa, F. Mourao, "Framework for Migrating Relational Datasets to NoSQL", Procedia Computer Science Volume 51, 2015, Pages 2593–2602, ICCS 2015.

[20] C. J. Tauro, B. R. Patil, K. R. Prashanth, "A Comparative Analysis of Different NoSQL Databases on Data Model, Query Model and Replication Model.", In Proceedings of the International Conference on ERCICA.

[21] V. N. Gudivada, D. Rao , V. Raghavan, "Data Management Issues in Big Data Applications", ALLDATA 2015 : The First International Conference on Big Data, Small Data, Linked Data and Open Data.

[22] F. Carini, "Mobility Analysis for Smart Cities: Territorial Intel l igence and Big Data", 2013.

[23] T. A. M. Phan, J. K. Nurminen and M. Di Francesco, "Cloud Databases for Internet-of-Things Data," Internet of Things (iThings), IEEE International Conference on, and Green Computing and Communications, IEEE and Cyber, Physical and Social Computing, Taipei, 2014, pp. 117-124.

[24] T. Harter, D. Borthakur, S. Dong, A. Aiyer, L. Tang, "Analysis of HDFS Under HBase: A Facebook Messages Case Study", Proceedings of the 12th USENIX Conference on File and Storage Technologies (FAST '14), February 17–20, 2014 • Santa Clara, CA USA.