[1.]Shafqat-ul-AHSAAN, [2.]Ashish Kumar MOURYA, [3.]Abdul Majid FAROOQI

PREDICTIVE ANALYTICS AND MODELING OF BIG DATA THROUGH MUTUAL CONTRACTION OF MAP-REDUCE AND R-PROGRAMMING LIBRARIES

[1-3.]Department of Computer Science & Engineering, Jamia Hamdard, New Delhi, INDIA

Abstract: The generation of data from every corner of the world forced the data scientists to think over it, how to handle such voluminous data while processing and storing. Therefore, to tackle this gigantic data called Big Data, its analytics has become very important. The development of innovative tools and algorithms is the need of the hour for the academic world, research study, and IT industry. The uncontrolled and continuous expansion of data sources generates heterogeneous data at a speed of light over the internet including Tweets, Facebook posts/likes, Blogs, news, articles, YouTube videos, website clicks, etc. Big data becomes a new challenge for research communities to extract meaningful information for commercial as well as personal use. There are numerous open source programming platform available free of cost for processing big data such as Hadoop, MapReduce, Hive, Flink, Spark, etc. Hadoop is an open source, distributed computing machine used for big data analytics. Map reduce is one of the most important and useful processing tools written in java language. This tool processes the large-scale data through distributed mode. It converts the large data inputs in small chunks and distributes them on different machines that are interconnected with each other in the form of a cluster. On the other hand, 'R' is another freely available statistical tool that offers a set of different types of libraries for statistical data mining. In this paper, we have presented architecture that allows coordination among MapReduce and R Libraries. This architecture will promote building predictive analytics combined with performance and flexibility for data science as it helps in exporting R libraries and process through MapReduce. The main objective of this paper is to provide in-depth analysis and relative evaluation of most up-to-date tools and models used for big data analytics.

Keywords: Big Data analysis, Data streaming, MapReduce, R libraries, Hadoop, Hive, Flink, Spark

INTRODUCTION

The sudden increase of information that is being generated online by means of social media, internet, and worldwide communications has increasingly rendered data-driven learning. A new study revealed that over 4 million queries are being received by Google every minute, e-mails' sent by users reaches the limit of 200 million messages, 72 hours of videos are uploaded by YouTube users, 2 million chunks of content are shared over Facebook, and 277,000 Tweets are generated every minute on Twitter, Whatsapp users share 3,47,222 photos, Instagram users post 2,16,000 new photos every minute [1], [2]. The present age is the age of Big Data, where data is growing on a large scale than ever before. According to the Computer World, 70% to 80% of data is considered to be in the unstructured form in organizations [3]. The data, which derives from social media, form 80% of the data globally and report for 90% of Big Data. As stated by the International Data Corporations (IDC) annual digital universe study [4], the data are being produced too rapidly and by the estimation of 2020, it would touch the range of 44 zettabytes which would be ten times larger than it was in 2013[5].

With the amount of data growing swiftly on a large scale, there may arise a situation when conventional analytical methods lack the ability to process such voluminous data and therefore we require advanced algorithms and techniques in order to extract data values that best aligns with the user interests, which finally became the key to introduce a new technology to the world called Big Data [3].

We are aware of the fact that the data storage capacities are increasing day-by-day; secondly, we lack the tools that are as powerful as to handle such massive data. Big data analytics is gaining focus from every field of research particularly from IT industry because of its unbeatable processing power in major areas like healthcare, business firms, social media, education, banking [1], etc.

Conventional means of processing and evaluation of data mostly depend on restricted data set organized in a structured form. Such tools and technologies are unsuccessful to put in any value in big data aspects [6]. Hence, more powerful machines and innovative techniques are compulsory to process the data and in fact, the generation of data on a large scale is the point of departure for the emergence and intensification of Big Data. Gigantic and multifarious data is out of the capability of traditional data warehousing tools to process.

As the technology and services seemed to have progressed at a pace, it leads to the generation and extraction of such giant sum of data from several sources that can be heterogeneous. The need for Big Data emerges from major companies like Google and Facebook [7]. The data that is generated while using Facebook or Google is mostly of unstructured form and it seems laborious to process data that contains billions of records of millions of people. Therefore, Big Data can be stated as the quantity of data that is far-fetching from the potential of technology to pile up, handle and process in the most powerful and substantial way.

BIG DATA CHARACTERISTICS

— Volume

The massive quantity of data that is derived every second constitutes the volume of Big Data [6,8]. There are multiple numbers of sources that play a key role in producing this vast portion of data like social media, surveillance cameras, sensor data,

weather data, phone records, online transactions, etc. We are living in an age where data is generated in petabytes and zettabytes. This sudden boom in the production of data that is too large to store and analyze requires advanced tools and techniques that open the way for Big Data. To handle such voluminous data is really a big challenge for the data scientists [9].

— Velocity

Velocity is defined by how rapidly the new data is being generated. As we see how messages on social media go viral within no time, millions of photos are being uploaded by Facebook users each and every second, it takes milliseconds for the business systems to analyze social networking websites to gather message that set off the verdict to purchase or sell shares[6,8]. Big Data streaming processing method makes it possible to examine the data while it is emanated, in need of ever storing it into the database.

— Variety

Variety focuses on different forms of data like music, pictures, text, e-mails, medical records and images, weather records and log files, etc. generated from multiple sources. This means that the data produced belongs to different categories consisting of raw, unstructured, structured and semi-structured data which looks very difficult to deal with [9].

— Veracity

Veracity denotes the meaningfulness or value of data.

— Value

Value focuses on the analytics and statistical methods, knowledge extraction and decision-making [6,8]. The data that is generated and it is not analyzed and processed then it is nothing other than garbage.

— Validity

Validity and Veracity are not the same but have a similar concept. Validity means the accuracy of data for the intended usage. Veracity leads to validity if the data is properly understood, it means that we have to check properly and appropriately whether the dataset is valid for a particular application or not [9].

— Volatility

Volatility refers to the period for which we have to store the data. If volatility is not in place then a lot of storage space is wasted in storing data that is no more required, for instance a commerce company keeps the purchase history of a customer for 1 year only as after 1 year the warranty on the purchased item expires so there is no reason to store such data [9].

THE SLANT OF BIG DATA ANALYTICS

— Data identification and collection

In this phase, multiple forms of a large number of data sources are recognized on the basis of the scope of the problem. More is the number of data resources more are the chances of discovery of hidden associations and patterns among data. Tools are required to encapsulate keywords, facts, and figures from these varied data sources as shown in Figure 1.

— Data storage

The data taken from various types of data sources are composed of structured and unstructured data and it has to be stored in databases/ data warehouse for future use. Traditional databases are not capable of handling such voluminous data; hence we require more powerful databases that can accommodate Big Data like NoSQL. There are innovative and influential models and databases that have been developed and maintained by organizations like Apache, Oracle, Facebook, Google, etc. that permit interpretive tools to obtain and perform processing of data from these data warehouses.

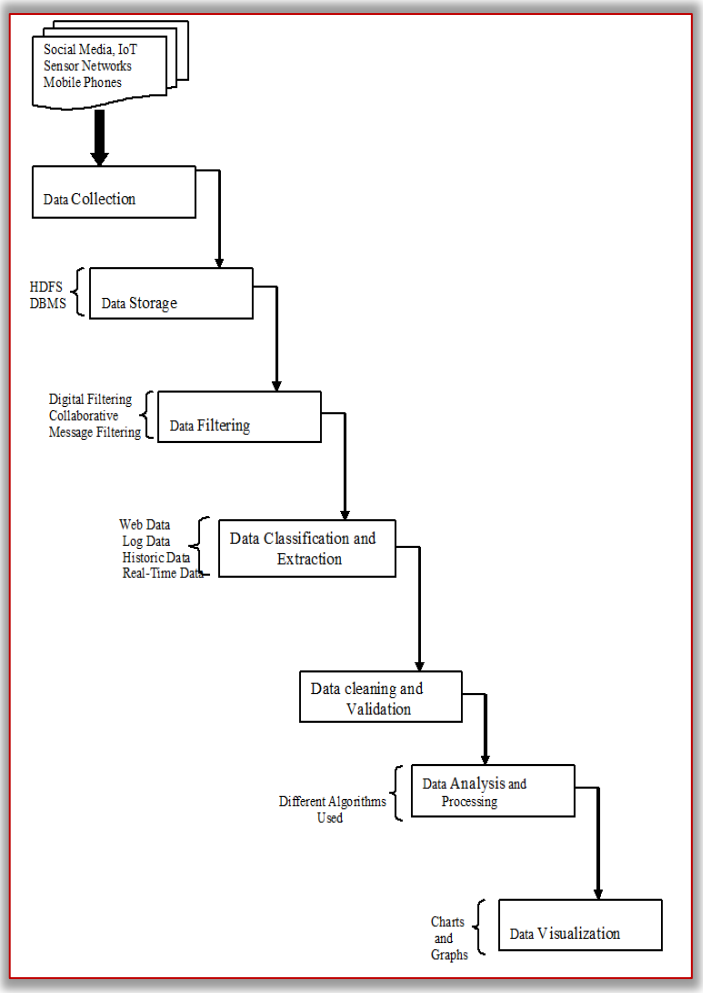

Figure 1. Data Analysis Process

— Data filtering and noise elimination

This phase plays a very important role in data analytics; the main objective of the concerned phase is to remove the redundant data, null and inconsistent data from the collected information. However, the data that has cleaned after the process of filtration might be beneficial in another context or analysis [10].

— Data classification and extraction

The data that is generated after the process of filtration goes under the classification. In this phase, the data is classified on the basis of the domain and the data that is out of a particular domain is extracted and converted into a common data format that can be used for analytics using different analytical tools [11].

By means of extraction, the data that is relevant or similar are also mined in order to reduce the data volume that is to be submitted to the analytics engine.

— Data cleansing, validation, and aggregation

This stage is used to apply validation rules on the basis of the business case. Validation rules authenticate that the data entered by the user meets the principles specified before the record is

saved. A validation rule comprises expression or formula that estimates the data in one or more fields. Even though, it may be complicated at times to put into use validation checks to the mined data due to intricacy. Aggregation is employed to merge compound data sets into smaller numbers based on common fields. This makes data processing further simpler.

— **Data analysis and processing**

This stage is responsible for actual data mining and analysis to ascertain inimitable and unknown patterns for making business decisions. The techniques used for data analytics may be different on the basis of the business case i.e. confirmatory, predictive, diagnostic or descriptive, exploratory and prescriptive [11].

— **Data visualization**

Under this phase, the results obtained from the analysis are represented into charts or graphical form so that it becomes easy to understand for the viewers.

BIG DATA ANALYTICS TOOLS

The main objective of big data analytics is to employ the most innovative and highly developed analytic tools and techniques in addition to gigantic, multiple forms of datasets like structured or unstructured, in the range of terabytes to zettabytes.

Big data comes into play for processing of voluminous data sets that are out of range from the processing, capturing and managing the potential of conventional relational databases. However, Big Data analytics tools make use of artificial intelligence, data mining and new techniques for data analysis. Some of the most important analytics tools are summarized as under:

— **Hive**

Apache Hive is an open source software project used for data query and analysis, built on top of Apache Hadoop. Although, the hive was very popular from the beginning as Facebook was developing it. These days, we on a regular basis execute millions of jobs over the Hadoop/Hive cluster having thousands of clients for a large number of applications ranging from easy summarization tasks to big commerce intelligence, support Facebook product features and machine learning applications [12].

As in the case of conventional databases, Hive also stores data in the form of tables, where each table is composed of multiple rows and each row is made of a specific number of columns. At this time, the following data types are supported:

1. Integers – big int (8 bytes), int (4 bytes), smallest (2 bytes), tiny int (1 byte). All integer types are signed.
2. Floating point numbers – float (single precision), double (double precision)
3. Strings
4. Associative arrays – map
5. Lists – list
6. Structs – struct.

Hive provides an SQL-like interface to query data that is stored in a variety of databases and file systems that amalgamate with Hadoop. The main components of the Hive are mentioned below:

» **Metascore:** The component is used to store the system directory and metadata about tables, columns, partitions, etc.
» **Driver:** It is responsible for handling and managing the hive query language statements as it moves from one phase to other through the hive.
» **Query Compiler:** After the query submission, it is query compiler which compiles HiveQL statements into a directed acyclic graph of map/reduce tasks.
» **Execution Engine:** The output from query compiler is provided as input to execution engine in the proper order of dependency.
» **Hive Server:** It is the component that makes the interface available to the user. It contains a JDBC/ODBC server by means of combining Hive with other applications.
» **Client components:** Client components include Command Line Interface (CLI), the web UI and JDBC/ODBC, driver.
» **Extensibility Interfaces:** If the user wants to make use of functions that are not available in the metastore, this component allows the user to define their own functions.

— **Apache Spark**

Spark works on Hadoop MapReduce algorithms provides a computing framework that is distributive in nature. It is efficient as it uses Memory Computing where the intermediate and output results can be stored in memory. Spark is particularly used for iterative applications like Machine Learning and Data Mining. Spark is based on the concept Resilient Distributed Datasets (RDD), which is a set of components that work in a parallel fashion with fault tolerant feature and permits users to unambiguously to store data in memory [13].

RDD is read-only data sets, loaded with an enormous set of operators to manipulate the data. Spark offers high-level APIs in python, scala and R and an engine that allows optimization. It provides a set of higher level tools like spark SQL for SQL, sparks streaming for streaming data, GraphX is used for graph processing and MLlib for machine learning. Spark SQL is like an SQL language that process queries admitted by the user. Spark Streaming is a computing model to process real-time data. It provides an API that allows integration of batch, streaming and interactive query applications. There is a parallel computation API called GraphX that is used for Spark charts and graph processing.

Spark supports a machine learning library that is scalable in nature called MLLib (Machine Learning library). The performances of Machine Learning algorithms are more efficient than Map-Reduce. MLLib includes the core algorithms primarly used for Machine Learning, such as clustering, collaborative filtering, dimensionality reduction, classification, regression and supports Sparse Matrix.

— **Apache Storm**

In December 2010, an idea strike to the mind of Nathan Marz, who thought if there exists a processing system that works on real-time data in order to save a lot of storage that is needed to store the data. The output of this idea came into the form of a new project that is called a storm.

Apache Storm enables software developers to build distributed systems that perform the processing of real-time data at a faster rate. Apache Storm is considered to be highly scalable, simple to use, and offers low delay with guaranteed high data processing. The architecture of the storm is very simple in order to build applications [14].

Apache Storm, on 17 September 2014 becomes the part of Apache family. Apache storm is an efficient tool that offers a couple of key attributes such as easy to use, fast as it processes millions of records in seconds, fault tolerant means processes data without any disturbance if a node fails to operate, the operation is performed by some other node in the cluster, reliability, and scalability which means processes the data in a parallel fashion over a number of machines that are connected with each other in order to share data.

— Map-Reduce

Map-Reduce is a programming mode, used to refine for massive data files with the implementation of coordinated and disbursed algorithms on a cluster. Map-Reduce programming structure is sparked by the Map ( ) and the Reduce ( ) function. In Map ( ) step, the Master Node or the Name Node accepts the input file and partition it into minor sub-problems, these sub-problems are then assigned to Slave Nodes or Data Nodes.

The Slave Nodes may further divide the problem into sub-subproblems. The Slave Node then handles these smaller problems and responds to the Master Node to which it is connected. In the Reduce ( ) step, the Master Node receives the result and combines them together to turn out the final result to the original problem that it has to solve [15].

R PROGRAMMING

There are various programming platforms available for processing the data and extract useful information for commercial and personal use. R programming is an important statistical programming interface available for gathering information. It is also open source, so users have no need to pay any license fee for personal use.

However, if somebody wants to use for commercial purpose then he/she is required to purchase its commercial version. R programming offers a wide range of packages and libraries for processing statistical data on a large scale. There are a variety of other related programming interfaces such as Weka and MATLAB that offers support for multiple statically operations, R-programming also supports matrix arithmetic.

Data structures of r-programming include vectors, matrices, arrays, data frames (alike tables in a relational database) and lists [16], arrays are stored in column-major order. R's extensible object system includes objects for (among others): regression models, time-series and geospatial coordinates. The scalar data type was never a data structure of R. Instead, a scalar is represented as a vector with length one [17-18]. There is a couple of libraries available in R, that supports Map-Reduce framework such as rHDFS, rmr, and rhbase.

— rHDFS

The R programming provides fundamental connectivity to the Hadoop Distributed File System through rhdfs library. The rhdfs library worked as an interface between R programming and Hadoop Distributed File System, which allows the client to access and process HDFS from the R programming interface. It can be used to browse, read, write, and modify files stored in HDFS.

The first function of rhdfs is Manipulation [19]. The users can write hdfs.copy, hdfs.move, hdfs.rename, hdfs.delete, hdfs.rm, hdfs.del, hdfs.chown, hdfs.put, hdfs.get commands as per the need and the format of data plus domain of data. In order to Read or Write files through rhdfs library, it has different commands such as hdfs.file, hdfs.write, hdfs.close, hdfs.flush, hdfs.read, hdfs.seek, hdfs.tell, hdfs.line.reader, hdfs.read.text.file.

The Directories handling commands are hdfs.dircreate, hdfs.mkdir.

— rmr

The rmr library provides Map-Reduce functionalities in R programming. The Hadoop clients may write Map-Reduce programs in R programming in a more productive and more elegant way. It provides a great statistical working environment for researchers. The R Programming clients may access big data analysis techniques by using Map-Reduce programming functionality on its console.

The rmr library must not be seen as data streaming, even it can be used as a streaming architecture. The Client can perform Hadoop streaming through R programming without any of those libraries since the language support stdin and stdout access [20].

— rhbase

The rhbase library provides connectivity functionalities in R programming. There is a wide range of versions available in R programming with rhbase library. The library comes with convenient functions to browse, and manipulate (read, write, and modify) files stored in the Hadoop Distributed File System.

WORKING PRINCIPLE OF R-LIBRARIES WITH MAP-REDUCE

Map-Reduce is an open source software that provides a platform for processing huge volumes of heterogeneous data in a most efficient way and produces striking results. It works on a distributed computing platform and supports Java programming language.

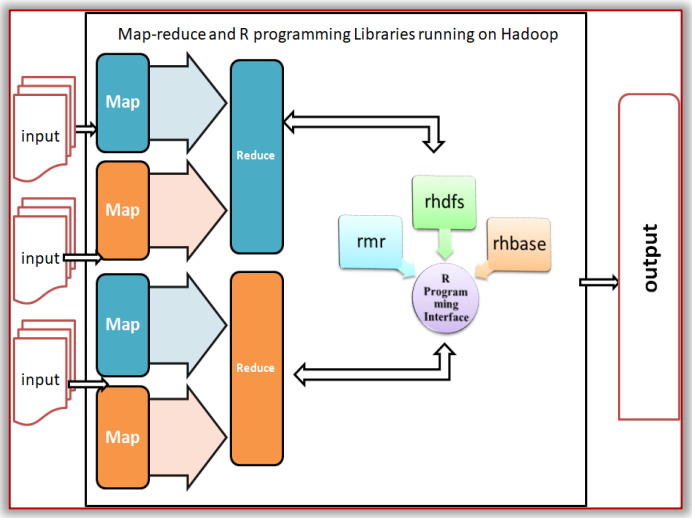

Figure 1. The architecture of Map-educe and R Programming libraries

Hadoop is proficient to run Map-Reduce programs that are written in diverse languages: Java, weka, Python, and C++. All of the programs of map-reduce are parallel in nature. It is a combination of Map and Reduces and the working principle of Map-Reduce is covered in previous Section.

The inputs given to the Hadoop platform are divided into various fixed size job. These jobs have also assigned as mapping or map function. The next phase of Map-Reduce consumes the yield of the Map function. Now the main task is to merge the relevant records from Map function output. In proposed architecture (Figure 1), the

related chunks have clubbed together along with their respective occurrence.

The outputs from the shambling phase are aggregated. Now Reduce function combines values from shambling phase and returns a single output value. Reduce function does not work on the perception of data locality. The resultant value of every Map job is fed to the Reducer. Map resultant values have been transferred to the machine where Reduce task is executing. Disparate the Map job, the output of the Reducer function has to be stored in the Hadoop File System.

The main function of R-Hadoop file system also collaborates to process the data. All of these three libraries run independently to manipulate the partitioned job and then produce consolidate output. The data manipulation power of R is competent and the turnaround time of R-programming language is really amazing as compared to other data manipulation platforms.

CONCLUSION

Big data analytics has boosted the IT industry as it has proven to be a very important tool to mine valuable patterns and unknown correlations of the potential consumer market, client preferences, buying attributes and a lot of other information from intricate data sources. The existing data processing techniques are not capable to handle this massive, varied and complex data. Nowadays e-commerce and digital markets have become hot areas which play a key role in the generation of Big Data and are gaining so much popularity that the commerce industry depends on online transactions and services to a great extent.

In this paper, we have presented architecture that allows coordination among Map-Reduce and R libraries. This architecture will promote building predictive analytics combined with performance and flexibility for data science as it helps in exporting R libraries and process through Map-Reduce. The main objective of this paper is to provide in-depth analysis and relative evaluation of most up-to-date tools and models used for big data analytics.

References:

[1] Jaseena, K.U. and David, J.M., 2014. Issues, challenges, and solutions: big data mining. CS & IT-CSCP, 4(13), pp.131-140.

[2] Lawal, Z., Zakari, R., Shuaibu, M. and Bala, A., 2016. A review: Issues and Challenges in Big Data from Analytic and Storage perspectives. International Journal of Engineering and Computer Science, 5(3), pp.15947-15961.

[3] Holzinger, A., Stocker, C., Ofner, B., Prohaska, G., Brabenetz, A. and Hofmann-Wellenhof, R., 2013. Combining HCI, natural language processing, and knowledge discovery-potential of IBM content analytics as an assistive technology in the biomedical field. In Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data (pp. 13-24). Springer, Berlin, Heidelberg.

[4] Landset, S., Khoshgoftaar, T.M., Richter, A.N. and Hasanin, T., 2015. A survey of open source tools for machine learning with big data in the Hadoop ecosystem. Journal of Big Data, 2(1), p.24.

[5] Jin, X., Wah, B.W., Cheng, X. and Wang, Y., 2015. Significance and challenges of big data research. BigDataResearch, 2(2), pp.59-64.

[6] Torre-Bastida, A.I., Del Ser, J., Laña, I., Ilardia, M., Bilbao, M.N. and Campos-Cordobés, S., 2018. Big Data for transportation and mobility: recent advances, trends and challenges. IET Intelligent Transport Systems, 12(8), pp.742-755.

[7] Kaisler, S., Armour, F., Espinosa, J.A. and Money, W., 2013, January. Big data: Issues and challenges moving forward. In 2013 46th Hawaii International Conference on System Sciences (pp. 995-1004). IEEE.

[8] Saqqi, M.K. and Jain, S., 2018. A survey towards an integration of big data analytics to big insights for value-creation. Information Processing & Management, 54(5), pp.758-790.

[9] Khan, N., Yaqoob, I., Hashem, I.A.T., Inayat, Z., Ali, M., Kamaleldin, W., Alam, M., Shiraz, M. and Gani, A., 2014. Big data: survey, technologies, opportunities, and challenges. The Scientific World Journal, 2014.

[10] Perroud, B., 2013. A hybrid approach to enabling real-time queries to end-users. Software Developer's Journal, 33, p.40.

[11] Elgendy, N. and Elragal, A., 2014, July. Big data analytics: a literature review paper. In Industrial Conference on Data Mining (pp. 214-227). Springer, Cham.

[12] Sangroya, A. and Singhal, R., 2015, December. Performance assurance model for hiveql on large data volume. In 2015 IEEE 22nd International Conference on High Performance Computing Workshops (pp. 26-33). IEEE.

[13] Fu, J., Sun, J. and Wang, K., 2016, December. Spark–a big data processing platform for machine learning. In 2016 International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII) (pp. 48-51). IEEE.

[14] Iqbal, M.H. and Soomro, T.R., 2015. Big data analysis: Apache storm perspective. International journal of computer trends and technology, 19(1), pp.9-14.

[15] Dean, J. and Ghemawat, S., 2008. MapReduce: simplified data processing on large clusters. Communications of the ACM, 51(1), pp.107-113.

[16] Dalgaard, Peter (2002). Introductory Statistics with R. New York, Berlin, Heidelberg: Springer-Verlag. pp. 10–18, 34

[17] An Introduction to R, Section 5.1: Arrays. Retrieved in 2010-03 from https://cran.r-project.org/doc/manuals/R-intro.html#Arrays.

[18] Ihaka, Ross; Gentlman, Robert (Sep 1996). "R: A Language for Data Analysis and Graphics" (PDF). Journal of Computational and Graphical Statistics. American Statistical Association. 5 (3): 299–314

[19] http://cran.r-project.org/web/packages/rJava/index

[20] http://www.adaltas.com/en/2012/07/19/hadoop-and-r-is-rhadoop/

ISSN: 2067-3809