Unstructured Datasets Analysis: Thesaurus Model

Parvathy Gopakumar
Department of Computer Science and Engineering
Mangalam Engineering College
Kottayam,Kerala,India

Neethu Maria John
Department of Computer Science and Engineering
Mangalam Engineering College
Kottayam,Kerala,India

Vinodh P Vijayan
Department of Computer Science and Engineering
Mangalam Engineering College
Kottayam,Kerala,India

Abstract: Mankind has stored more than 295 billion gigabytes (or 295 Exabyte) of data since 1986, as per a report by the University of Southern California. Storing and monitoring this data in widely distributed environments for 24/7 is a huge task for global service organizations. These datasets require high processing power which can't be offered by traditional databases as they are stored in an unstructured format. Although one can use Map Reduce paradigm to solve this problem using java based Hadoop, it cannot provide us with maximum functionality. Drawbacks can be overcome using Hadoop-streaming techniques that allow users to define non-java executable for processing this datasets. This paper proposes a THESAURUS model which allows a faster and easier version of business analysis.

Keywords: Hadoop;MapReduce;HDFS;NoSQL;Hadoop-Streaming

1. INTRODUCTION

Data has never been more important to the business world as it has become a vital asset as valuable as oil and just as difficult to mine, model and manage. The volume and veracity of the datasets that are being stored and analyzed by the business are unforeseeable and the traditional technologies for data management such as relational databases cannot meet the current industry needs. Bigdata technologies play a vital role to address this issue. Early ideas of big data came in 1999 and at present it becomes an unavoidable phenomenon tool through which we manage business and governance. For a layman the idea of Bigdata may relate to images of chaotic giant warehouses over crowded office space with numerous staffs working through huge number of pages and come with boring formal documents under supervision of some old bureaucrat. On the contrary working of Bigdata is simple and well structured, yet exciting enough to pose new challenges and opportunities even to experts of industry. It provides parallel processing of data in hundreds of machines that are distributed geographically. Necessity of Bigdata arises under the obligation of the following:

1. When existing technology is inadequate to perform data analysis.
2. In the case of handling more than 10TB of dataset.
3. Relevant data for an analysis present across multiple data stores which are filed in multiple formats.
4. When steaming data have to be captured, stored and processed for the purpose of analysis.
5. When SQL is inefficient for high level querying.

In today's data centered world Hadoop is considered as the main agent of big data technology due to its open source nature. However as it is a java based ecosystem, it created hurdle for programmer from non-java background. To address this issue it has facilitated a tool, 'Hadoop-Streaming' by enabling flexibility in programming with effective parallel computability.

2. PROBLEM STATEMENT

Why Big data analysis? Well, it helps the organization to harness their transactional data and use it to identify new opportunities in a cost effective and efficient manner. Primary aim of data analysis is to glean actionable logic that helps the business to tackle the competitive environment. This will alert the business for their inevitable future by introducing new products and services in favor of the customers. Unfortunately for the matter of convenience 80% of the business oriented data are stored in an unstructured format. Structured data usually resides in a relational database with predefined structures so converting the data to different models and analyzing them seems mundane. Here the role of Hadoop-Streaming arises which works on a Map and Reduce paradigm by analyzing the unstructured data and presents viable business logic.

The aim of the paper is to:

- Study existing framework employed by industry players.
- Present a new roadmap for efficient and effective approach to Bigdata problems: THESAURUS MODEL

3. BACKGROUND

3.1 Structured Vs Unstructured datasets

The question that encounters a rookie is that why one uses unstructured dataset when there is always a possibility of using structured data. At the outset of computing, the term storage corresponded only plain texts. Now user needs to store richer content than plain text. Rich data type includes pictures, movies, music, x-rays ,etc.It provides superior user experience at the expense of storage space. Such data sets are called unstructured because they contain data that do not fit neatly in a relational database. Industry came up with a third category called semi structured data which resides in a relational database, similar to structured data. However it does not have some organizational property necessary to make them easy to be analyzed.(Eg.XML doc)

3.2 NOSQL Data store

A NOSQL database [4] provides mechanism for storage and retrieval of data which is modeled in contrast to the tabular relations used in relational databases. It become common in the early twenty first century when the industrial requirements triggered a need of database structures that support query languages other than SQL.(called "Not only SQL", non SQL).This is mostly used in big data and real-time applications as it provides simpler design, horizontal scalability and high availability. The most popular NOSQL databases are MongoDB, Apache Cassandra [3], Datastax, Redis.

3.3 Hadoop & Hadoop Streaming

Apache Hadoop [1] is open source software for reliable, scalable and distributed computing. Hadoop framework allows distributed processing of large datasets across low level commodity hardware using simple programming models. This framework is inspired by Google's MapReduce structure in which application is broken down into numerous small parts and each part can be run in any node in the cluster. Hadoop contains two major components - a specific file system called Hadoop Distributed File System (HDFS) and a Map Reduce framework. Hadoop works on divide and conquer principle by implementing Mapper and Reducer in the framework. Mapper function splits the data into records and converts it into (key,value) pairs. Before feeding the output of the Mappers to Reducer an intermediate Sort and Shuffle phase is implemented in the MapReduce framework to reduce the work load at Reducer machine. The sorted (key,value)pair is given into Reducer phase. The Reducer function does the analysis of the given input and the result will be loaded to HDFS(eg.The maximum temperature recorded in a year, positive and negative ratings in a business etc.).The analyst has to develop Mapper and Reducer functions as per the demand of the business logic.

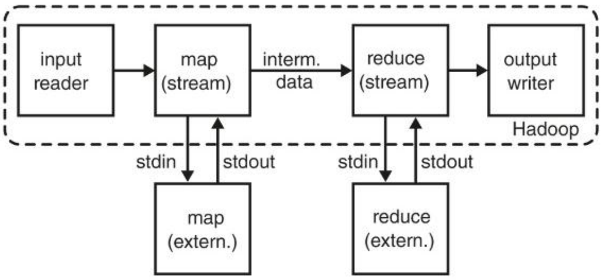

Figure 1 Hadoop-Streaming

Hadoop Streaming (see Figure 1) is an API provided by Hadoop which allows user to write MapReduce functions in languages other than java[2]. Hadoop Streaming uses Unix standard streams as the interface between Hadoop and our MapReduce programs, so the user has the freedom to use any languages (Eg. Python, Ruby, Perl etc.) that can read standard input and write to standard output.

4. ANALYSING UNSTRUCTURED DATASETS USNG HADOOP-STREAMING

Due to the difficulties in analyzing the unstructured data organizations have turned to a number of different software solutions to search and extract prerequisite information. Regardless of the platform used, the analysis must undertake three major steps– data collection, data reduction, data analysis [7][8][9][10](see Figure 2):

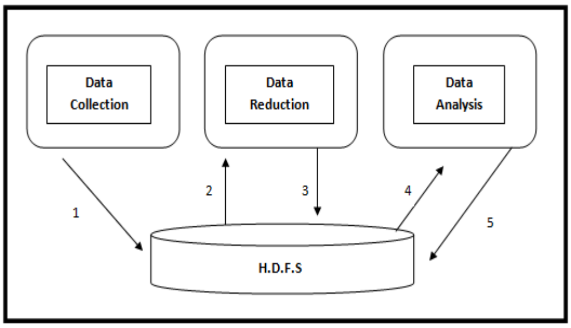

Figure 2 Analyzing Unstructured Dataset

A. Data Collection: In this stage the datasets to be analyzed can be collected through two methods. Firstly, data can be downloaded from different nodes containing the specified records to HDFS. Alternatively it can be done by connecting to the local servers containing the records. The former can be achieved by tools such as Sqoop, Flume and the latter using Apache Spark[6]. In a real time environment the streaming datasets can be accessed using standard public key encryption technique to ensure authenticity.

B. Data Reduction: Once the unstructured dataset got available, analysis process can be launched. It involves cleaning the data, extracting important features from data, removing duplicate items from the datasets, converting data formats, and many more. Huge datasets are minimized into structural and more usable format using series of Mapper and Reducer functions. This is done by projecting the columns of interest and thus converting it in a format which will be adaptable for final processing. Cleaning text is extremely easy using R language, whereas Pig and Hive supports high level abstraction of data preprocessing.

C. Data Analysis: Before the inception of Bigdata technologies collecting, preprocessing and analyzing terabytes of data was considered impossible. But due to the evolution of Hadoop and its supporting framework the data handling and data mining process seems not so tedious. Programmer with the help of Hadoop Streaming API can write the code in any language and work according to the domain of user. In this stage the pre processed data is studied to identify the hidden pattern. Hadoop provides a Mahout tool that implements scalable machine learning algorithms which can be used for collaborative filtering, clustering and classification .The analyzed data then can be visualized according to the requirement of the business using Tableau, Silk, CartoDB, Datawrapper.

Thus the whole process of analysis can be explained in a five step workflow:

1. Collecting the data from alien environment and keep it inside the Hadoop Distributed File System.
2. Apply set of MapReduce tasks to the step one collected data and project the columns of interest based on the user query.
3. Keep the preprocessed data in HDFS for further analysis.
4. Use the preprocessed data for analyzing the pattern of interest.
5. Store the result in HDFS so that with the help of visualization tools user can selectively adopt the method of presentation.

5. MODIFICATION OF EXISTING SYSTEM: THESAURUS MODEL

The underline motivation behind this model is the lack of knowledge base in the existing analysis framework which in turn causes the system to follow some unnecessary repetition. Consider an analysis problem to find the maximum recorded temperature in last 5 years. So the analysis is done by

1. Collecting the data from National Climatic Data Center [5] and store in HDFS.
2. Project the field which contains the temperature data i.e. the column of interest.
3. Store the preprocessed result in HDFS.
4. Find the maximum temperature reported by analyzing the (key, value) pair.
5. Store the final result in HDFS.

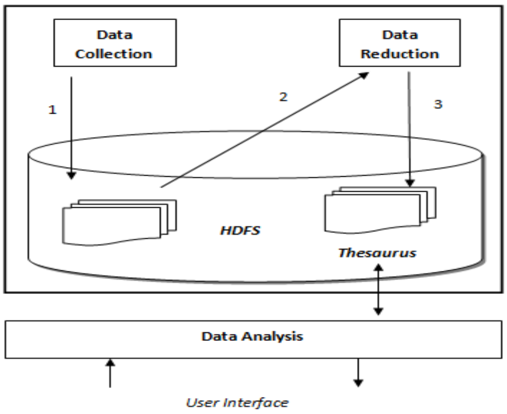

Figure 3 Thesaurus Model

So the maximum temperature of the year is accessed from the file system and can be used for monitoring and reporting purposes. Later if the same analyst needs to find the maximum humidity reported, he has to go through the whole datasets and has to bear the trouble of preprocessing and reducing the data again. This can be avoided by using Thesaurus model. According to this module, minable information are logically arranged and kept in the HDFS so that the future request for the information retrieval can be done in no time. Once the data set is converted into a structural format the schema of the dataset should be specified by the preprocessing programmer so the analyst need not come across the trouble of understanding the newly created data set.. This preprocessed datasets can replace the old datasets so that the unnecessary storage issue is taken care of by the model. The working of the system is specified in two phases, one for collection and preprocessing, and second for analysis. In the first phase the necessary data which can be analyzed are collected and preprocessed. This data is then stored in the thesaurus module in HDFS and made it available for the user to analyze based on the industry needs. Thesaurus not only contains the structured data but also the schema of the data storage. In phase two, the required query can be addressed by referring the schema .Thus analyst need not consider the problems of unstructured data collected by the system. The Figure 3 represents the work flow of Thesaurus model.

1. Collect the data from distributed environment and store in HDFS.
2. Use the stored data for preprocessing.
3. Store the preprocessed data in Thesaurus with a predefined schema. To avoid the storage bottleneck the data that are collected on the first place can be removed as it is no longer necessary.

6. CONCLUSION & FUTURE SCOPE

Mining the inner pattern of business invokes the related trends and interests of the customers. This can be achieved by analysing the streaming datasets generated by the customers in each point of time.Hadoop provides flexible architecture which enables industrialist and even starters to learn and analyse this social changes.Hadoop-Streaming is widely used for sentimental analysis using non-java executables.Also proposed a THESARUS model which works in a time and cost effective manner for analysing these humongous data. Future scope is to enable the efficiency of the system by developing a THESARUS model which is suitable to analyse terabytes of data and returns with the relative experimental results.

7. ACKNOWLEDGMENTS

I would like to thank Almighty, my professors, friends and family members who helped and supported in getting this paper work done with in time.

8. REFERENCES

[1] Apache Hadoop. [Online]. Available: http://hadoop.apache.org

[2] Apache Hadoop-Streaming. [Online]. : http://hadoop-streaming.apache.org

[3] Cassandra wiki, operations. [Online]. Available: http://wiki.apache.org/cassandra/Operations

[4] NOSQL data storage [online]: http://nosql-database.org

[5] National energy research scientific computing center. [Online]. Available: http://www.nersc.gov

[6] Apache Hadoop [Online]: http://spark.apache.org

[7] E. Dede, B. Sendir, P. Kuzlu, J. Weachock, M. Govindaraju, and L. Ramakrishnan, "A processing pipeline for cassandra datasets based on Hadoop streaming," in Proc. IEEE Big Data Conf., Res. Track, Anchorage, AL, USA, 2014, pp. 168–175.

[8] E. Dede, B. Sendir, P. Kuzlu, J. Weachock, M. Govindaraju, L. Ramakrishnan, "Processing Cassandra Datasets with Hadoop-Streaming Based Approaches", *IEEE Transactions on Services Computing*, vol. 9

[9] J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S.-H. Bae, J. Qiu, and G. Fox, "Twister: A runtime for iterative mapreduce," in Proc. 19th ACMInt. Symp. High Perform. Distrib. Comput., 2010, pp. 810–818.

[10] Z. Fadika, E. Dede, M. Govindaraju, and L. Ramakrishnan, "MARIANE: MApReduce implementation adapted for HPC environments," in Proc. 12th IEEE/ACM Int. Conf. Grid Comput., 2011,vol. 0, pp. 1–8.