A Review of Diabetes Diagnosis and Record Management using HIVE

Bhawna Bajaj[1] Mr. Parikshit[2]
[1]Research Scholar [2]Assistant Professor
[1,2]Doon Valley Institute of Engineering & Technology, Karnal, India

Abstract— Modernizing healthcare industry's move towards processing massive health records, and to access those for analysis and put into action will greatly increases the complexities. Due to the growing unstructured nature of Big Data form health industry, it is necessary to structure and emphasis its size into nominal value with possible solution. Healthcare industry faces many challenges that make us to know the importance to develop the data analytics[1].In this paper we will show that how HIVE (hierarchy of international vengeance and extermination) can process and analyze the diabetic data set with the help of SQL like HIVEQL. With the help of HIVE complexity is reduced. No need to write big and complex programs in java. HIVE structures the data and also queries the data in a very small amount of the time. It can manage only big data. For small data HIVE is not applicable. It supports data loading. This language supports tables, partitions, join, aggregation etc. HIVE also includes a system catalog- Metastore that contains schemas and statistics which are beneficial in data processing, data analyzing, query optimization and query compilation. Analyses of the diabetic data to perform the Outpatient Monitoring and Management of Insulin Dependent Diabetes Mellitus (IDDM) set using HIVE as a warehousing tool resulted in providing an efficient way to cure and care the patients and in deriving some interesting facts such as helping the hospital management to manage the patient records and arrange the medical equipments, staff, labs etc based on the frequency of the arrival of the patients on daily, monthly as well as yearly basis.

Key words: Diabetic Mellitus (DM), Hadoop, Hive, HiveQL, Big Data, Data-Analysis, Partitioning, HDFS, Map Reduce, Serde

I. INTRODUCTION

A. Big Data: Definition:

The term 'Big Data' describes innovative techniques and technologies to capture, store, distribute, manage and analyze petabyte- or larger-sized datasets with high-velocity and different structures. Big data can be structured, unstructured or semi-structured, resulting in incapability of conventional data Management methods. Data is generated from various different sources and can arrive in the system at various rates. In order to process these large amounts of data in an inexpensive and efficient way, parallelism is used. Big Data is a data whose scale, diversity, and complexity require new architecture, techniques, algorithms, and analytics to manage it and extract value and hidden knowledge from it. Hadoop is the core platform for structuring Big Data, and solves the problem of making it useful for analytics purposes. Hadoop is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance.

B. Three Versions of Big Data:

Volume of data: Volume refers to amount of data. Volume of data stored in enterprise repositories have grown from megabytes and gigabytes to petabytes [2].

Variety of data: Different types of data and sources of data. Data variety exploded from structured and legacy data stored in enterprise repositories to unstructured, semi structured, audio, video, XML etc [2].

Velocity of data: Velocity refers to the speed of data processing. For time-sensitive processes such as catching fraud, big data must be used as it streams into your enterprise in order to maximize its value [2].

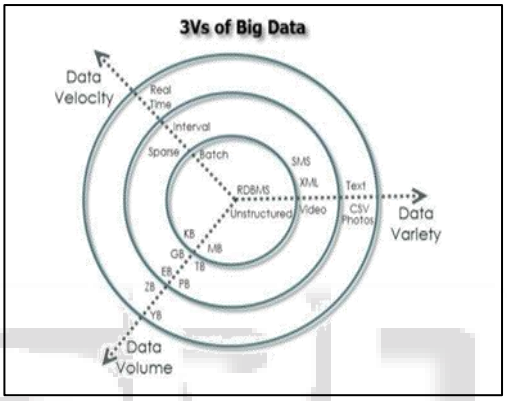

Fig. 1: Three versions Of Big Data

II. PROPOSED SYSTEMS

This section briefly describes the characteristics of Diabetes Data set, Hadoop, Hive and HDFS.

A. Diabetes Data Set:

Diabetic Mellitus (DM) is one of the Non Communicable Diseases (NCD), is a major health hazard in developing countries such as India. The acute nature of DM is associated with long term complications and numerous of health disorders. There are three main types of this disease[3].

Type1 DM results from the body's failure to produce insulin, and presently requires the person to inject insulin. This form is referred as Insulin - Dependent Diabetes Mellitus (IDDM). This can either be due to a) low or absent production of insulin by the beta islet cells of the pancreas subsequent to an auto-immune attack or b) insulin-resistance, typically associated with older age and obesity, which leads to a relative insulin-deficiency even though the insulin levels might be normal.

Type 2 DM results from insulin resistance, a condition in which cells fail to use insulin properly, sometimes combined with an absolute insulin deficiency. This form was previously referred to as Non-Insulin - Dependent Diabetes Mellitus (NIDDM).

The third main form, gestational diabetes occurs when pregnant women without a previous diagnosis of diabetes develop a high blood glucose level. It may precede development of type 2 DM. It was estimated that 61.3 million

people aged 20-79 years live with diabetes at 2011 in India. This number was expected to increase to 101.2 million by 2030.

In this paper, analysis is done on Insulin Dependent Diabetes Mellitus (IDDM) - Type1 Diabetes data set.

1) Characteristics of Diabetes data set

Diabetes files consist of four fields per record. Each field is separated by a tab and each record is separated by a newline [3]

File Names and format:

- Date in MM-DD-YYYY format
- Time in XX:YY format
- Code
- Value

The Code field is deciphered as follows [3]:

33 = Regular insulin dose
34 = NPH insulin dose
35 = UltraLente insulin dose
48 = Unspecified blood glucose measurement
57 = Unspecified blood glucose measurement
58 = Pre-breakfast blood glucose measurement
59 = Post-breakfast blood glucose measurement
60 = Pre-lunch blood glucose measurement
61 = Post-lunch blood glucose measurement
62 = Pre-supper blood glucose measurement
63 = Post-supper blood glucose measurement
64 = Pre-snack blood glucose measurement
65 = Hypoglycemic symptoms
66 = Typical meal ingestion
67 = More-than-usual meal ingestion
68 = Less-than-usual meal ingestion
69 = Typical exercise activity
70 = More-than-usual exercise activity
71 = Less-than-usual exercise activity
72 = Unspecified special event

2) Architecture of Proposed System:

The diabetic data set is given as input to the system, which comprises of hive. The raw data is just a file consisting of comma separated values, for the first time when we look into it, it just looks like a junk of data. But a proper analysis of this data set will reveal some interesting facts. The raw input is given as input to hive, the data set is analysed and partitioned based on different attribute, one is based on diagnosis and curing of patients based on the Blood Glucose (BG) and other is based on the management approach which helps the hospital management to avail the details of the patients and frequency of their occurring on daily, monthly and yearly basis which helps in deciding the no. of medical equipments, staff, machines, lab required as an output which is obtained from hive is well formatted data which helps management in decision making.

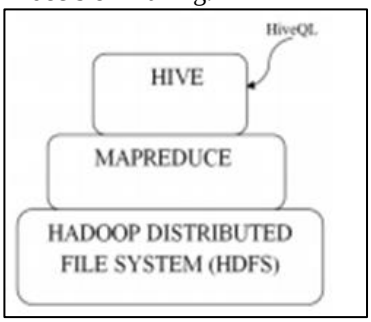

Fig. 2: Figure of metrics

B. HADOOP : Solution For Big Data Processing

Hadoop is a popular open source software framework that allows the distributed processing of large scale data sets. It employs the MapReduce paradigm to divide the computation tasks into parts that can be distributed to a commodity cluster and therefore, provides horizontal scalability. The MapReduce functions of Hadoop uses (key,value) pairs as data format. The input is retrieved in chunks from Hadoop Distributed File System (HDFS) and assigned to one of the mappers that will process data in parallel and produce the (k1,v1) pairs for the reduce step. Then, (k1,v1) pair goes through shuffle phase that assigns the same k1 pairs to the same reducer. The reducers gather the pairs with the same k1 values into groups and perform aggregation operations (see Figure below)[4]. HDFS is the underlying file system of Hadoop. Due to its simplicity, scalability, fault-tolerance and efficiency Hadoop has gained significant support from both industry and academia; however, there are some limitations in terms of its interfaces and performance [10]. Querying the data with Hadoop as in a traditional RDBMS infrastructure is one of the most common problems that Hadoop users face. This affects a majority of users who are not familiar with the internal details of MapReduce jobs to extract information from their data warehouses.

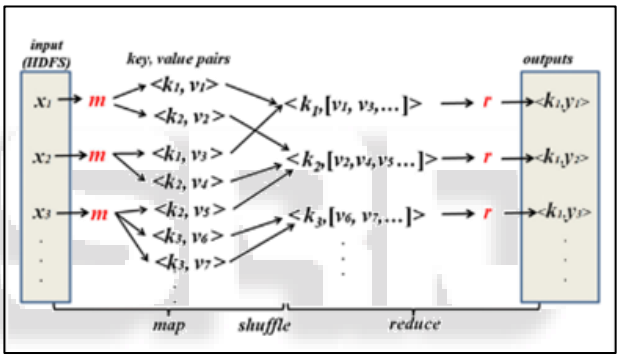

Fig. 3: Map Reduce Tasks

C. HIVE and HIVEQL:

Hadoop Hive is an open source SQL-based distributed warehouse system which is proposed to solve the problems mentioned above by providing an SQL-like abstraction on top of Hadoop framework.[4] Hive is a data warehouse system for Hadoop that facilitates easy data summarization, ad-hoc queries, and the analysis of large datasets stored in Hadoop compatible file systems. Hive provides a mechanism to project structure onto this data and query the data using a SQL-like language called HiveQL. At the same time this language also allows traditional map/reduce programmers to plug in their custom mappers and reducers when it is inconvenient or inefficient to express this logic in HiveQL.

D. Hdfs: Hadoop Distributed File System:

Hadoop includes a fault tolerant storage system called the Hadoop Distributed File System, or HDFS. HDFS is able to store huge amounts of information, scale up incrementally and survive the failure of significant parts of the storage infrastructure without losing data. Hadoop creates clusters of machines and coordinates work among them. Clusters can be built with inexpensive computers. If one fails, Hadoop continues to operate the cluster without losing data or interrupting work, by shifting work to the remaining machines in the cluster. [5] In HDFS(hadoop distributed file

system), data is stored into blocks of data. The default size of each data block is 128 MB[6]. Also same blocks of data are replicated across multiple nodes to provide reliability with help of data nodes and also to increase the performance during map reduce jobs. The name node is a master node which give instructions and directs the data nodes. Map reduce provides mechanism that enable access to each nodes in the cluster. HIVE provides the reliability to create and query data on a large scale with familiar SQL based language called HIVEQL. Data is loaded in batch and then queries are executed to answer strategic business questions. Tables are organized through HIVE. When a table is created through hive a directory is created in hdfs on each node that represents the data for the table are created on each of the nodes and the metadata keeps track of where the files that make up each table are located. These files are located in directory with name of the tables in HDFS in the /user/hive/warehouse folder by default.

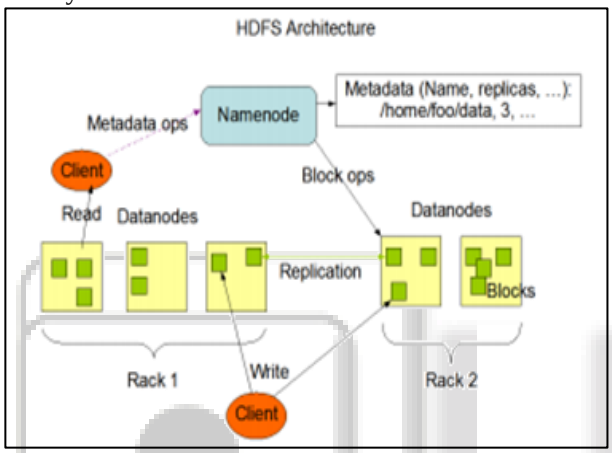

Fig. 4: HDFS Architecture

E. Serde:

Serialization/deserialization(Se/de):- Hive can take an implementation of the Serde java interface provided by the user and associate it to a table or partition. As a result custom data formats can easily be interrupted and queried form. The default SERDe implementation in hive is called the LazySerDe [6]. It deserializes rows into internal objects lazily so that the cost of deserialization of a column is incurred only if the columns of the row is needed in some query expression. The lazy SerDe assured that data is stored in the file such that the rows are delimited by CTRL-A(ascii code 1). This SerDe can also be used to read data that uses any oter delimiter character between columns. In the present data set the serde is used to separate the date (mm-yy-dd) delimited with – to month, year and day separately so as to prepare the raw data in such a way that partitioning can be done and data can be easily loaded into the partitioned table.

F. Partitioning In Hive:

A simple query in Hive reads the entire dataset even if we have where clause filter. This becomes a bottleneck for running MapReduce jobs over a large table. We can overcome this issue by implementing partitions in Hive. Hive makes it very easy timplement partitions by using the automatic partition scheme when the table is created.

In Hive's implementation of partitioning, data within a table is split across multiple partitions. Each partition corresponds to a particular value(s) of partition column(s) and is stored as a sub-directory within the table's directory on HDFS. When the table is queried, where applicable, only the required partitions of the table are queried, thereby reducing the I/O and time required by the query.

III. Related Work

In Related work Section, brief information is presented concerning the related work on analysis of data set using HIVE. Many people have developed various prediction models using Hive to predict diabetes. A few of the models developed using data mining are as follows:

Abdulla et al [8] worked on predictive analysis of diabetic treatment using a regression based data mining technique. Using this techniques to diabetes data, they discover patterns using SVM algorithm that identify the best mode of treatment for diabetes across different age [8]. They concluded that drug treatment for patients in the young age group can be delayed whereas; patients in the old age group should be prescribed drug treatment immediately.

The hadoop usage in health care became more important to process the data and to adopt the large scale data management activities. The analytics on the combined compute and storage can promote the cost effectiveness to be gained using hadoop [9].

The soft computing based prediction model was developed for finding the risks accumulated by the diabetic patients. They have experimented with real time clinical data using Genetic Algorithm [10]. The obtained results pertaining to the level of risk which prone to either heart attack or stroke. The novel pre-processing phase with missing value imputation for both numerical and categorical data. A hybrid combination of Classification and Regression Trees (CART) and Genetic Algorithms to impute missing continuous values and Self Organizing Feature Maps (SOFM) to impute categorical values was improved in [11].

V. Sangeetha at [12] used the Pima Indians Diabetes Database for carrying out Prediction and classification of various type of diabetes using classification algorithm and found out that classification algorithm was the best algorithm to classify the data set. A detailed analysis of the Pima diabetic data set was carried out efficiently using of Hive and R.

All the above researchers have been successful in analysing the diabetic data set and developing good prediction models. In this paper, we use the predictive analysis technique in Hadoop/Hive environment to predict the patients number arriving the hospital in a day,month and year and to predict the hospital requirements on the basis on patients arrival and diagnosing patients and prescribing them the Diet and exercise as per the blood glucose value. This system provides efficient way to care and cure the patients at low cost with better outcomes like affordability and availability.

IV. Conclusion

In this paper, we concluded and listed the use cases handled by the hospital by performing the analysis of diabetic data along with Monitoring and Management of Insulin Dependent Diabetes Mellitus (IDDM) patients using HIVE as a warehousing tool resulted in providing an efficient way to cure and care the patients and in deriving some interesting

facts such as helping the hospital management to arrange the medical equipments, staff, labs etc based on the frequency of the arrival of the patients on daily, monthly as well as yearly basis and helps doctor in prescribing immediate treatment required for the patient with extreme low and extreme high blood glucose value checked at any time based on the code value.

Use Cases:

1) Frequency of arrival of patients in a year, in a month or in a day.
2) Year/Month/Day with maximum patient count.
3) Diagnosing patients on the basis of Blood Glucose value.
4) Referring patients for DIET and Exercise on the basis of blood glucose value.
5) Different Blood Glucose value check scheduled for the entire day.
6) Detail of the patient with Maximum Blood Glucose
7) Top 20 patients with highest/lowest glucose level at breakfast/Lunch/Dinner.
8) Top 20 patients in a month/day with highest Blood Glucose.
9) Generating a monthly Blood Glucose report of patients.
10) Person with High InsulinDose , with low DIET and max Exercise prescribed.
11) Person with Low InsulinDose prescribed for proper DIET chart and Exercise.
12) Ordering of patients on the basis of Glucose rate.
13) Average glucose rate found in an year/month/day..
14) Analysing the most busy and most free day for hospital staff.
15) Person with best record in terms of insulin dose check, diet and exercise

REFERENCES

[1] N.M. Saravana kumar, T. Eswari, P. Sampath, S. Lavanya, "Predictive Methodology for Diabetic Data Analysis in Big Data" ,2015.
[2] Dr. Urmila R. Pol, "Big Data Analysis: Comparision of Hadoop MapReduce, Pig and Hive", 2016.
[3] "UCI - Machine Learning Repository - Repo for the DataSets", https://archive.ics.uci.edu/ml/datasets.html, 2007.
[4] Journal of Cloud Computing Advances, Systems and Applications20143:12 DOI: 10.1186/s13677-014-0012-6© Dokeroglu et al.; licensee Springer 2014.
[5] Harshawardhan S. Bhosale, Prof. Devendra P. Gadekar, "A Review Paper on Big Data and Hadoop" , 2014.
[6] Nikita Bhojwania, Vatsal Shah, "HADOOP HIVE Data Warehousing System – A Review", 2015
[7] "Introduction to Hive Partitioning", https://dzone.com/articles/introduction-hives
[8] Abdullah A. Aljumah, Mohammed Gulam Ahamad, Mohammad Khubeb Siddiqui,"Application of data mining: Diabetes health care in young and old patients" in Journal of King Saud University – Computer and Information Sciences (2013) 25, 127–136.
[9] D. Peter Augustine, "Leveraging Big Data analytics and Hadoop in Developing India's Health Care Services", International Journal of Computer Applications, vol 89(16), pp 44-50, 2014.
[10] Sabibullah M, Shanmugasundaram V, Raja Priya K, "Diabetes Patient's Risk through Soft Computing Model", International Journal of Emerging Trends & Technology in Computer Science, vol 2(6), 2013.
[11] H. Bhat, P. G. Rao, S. Krishna, and P. D. Shenoy, "An Efficient Framework for Prediction in Healthcare," Most, Springer-Verlag Berlin Heidelberg , pp. 522-532, 2011.
[12] K. Rajesh, V. Sangeetha, "Application of Data Mining Methods and Techniques for Diabetes Diagnosis" in International Journal of Engineering and Innovative Technology (IJEIT) Vol 2(3), 2012.

