PUNE INSTITUTE OF COMPUTER TECHNOLOGY

Department of Computer Engineering

(2022- 2023)

DSBDAL

Batch: - K2

Case Study

On

Health care systems with Hadoop Ecosystem Components

Submitted

by

31202 – Amrit Singh

Guided by

Prof. Kimaya Urane

Table of Contents: -

1. Problem Statement

Write a case study to process data-driven for Health care systems with Hadoop Ecosystem components as shown.

- HDFS: Hadoop Distributed File System
- YARN: Yet Another Resource Negotiator
- MapReduce: Programming-based Data Processing
- Spark: In-Memory data processing
- PIG, HIVE: Query based processing of data services
- HBase: NoSQL Database (Provides real-time reads and writes)
- Mahout, Spark MLLib: (Provides analytical tools) Machine Learning algorithm libraries Solar, Lucene: Searching and Indexing

2. Motivation

➢ The analysis and prediction of future health conditions are still developing. The data which is exerted in a little amount has risen dramatically from a few bytes to terabytes, not only has the storage increased but also the dataset maintenance

➢ The traditional method of using data mining and diagnosis tools is difficult, therefore the need for big data tools and techniques arises.

3. Scope

Everyone likes that their work must be completed in less amount of time and the calculator on the salesforce cloud will help users/customers to do that. Hence, many users will use such applications which help them to achieve their goals in less time.

4. Objective

By performing this case study, we shall be able to:

- ➢ HDFS: Hadoop Distributed File System
- ➢ YARN: Yet Another Resource Negotiator
- ➢ MapReduce: Programming based Data Processing
- ➢ Spark: In-Memory data processing
- ➢ PIG, HIVE: Query based processing of data services
- ➢ HBase: NoSQL Database (Provides real-time reads and writes)
- ➢ Mahout, Spark MLLib: (Provides analytical tools) Machine Learning algorithm libraries
- ➢ Solar, Lucene: Searching and Indexing above components of Hadoop ecosystem in Health care system.

5. Outcomes

By performing this case study, student will be able to understand following components of Hadoop Ecosystem:

- HDFS
- YARN
- MapReduce
- Spark
- PIG, HIVE
- HBase
- Mahout, Spark MLLib
- Solar, Lucene: Searching and Indexing.

6. Software and Hardware Requirements:

Software:

- Windows 10 OS, 64 bits
- Hadoop

Hardware:

- Processor: Intel i-5 8th gen
- Manufacturer: Acer Nitro 7
- Ram: 8 GB/ 16GB Optane memory

7. Theory

1. HDFS (Hadoop Distributed File System):

The **Hadoop Distributed File System (HDFS)** is the essential information stockpiling framework utilized by Hadoop applications. It comprises NameNode, The Master and DataNodes. The Slave design to execute a disseminated record framework called Hadoop Distributed File System to get to information crosswise over exceedingly adaptable Hadoop Clusters in an effective way. Hadoop Framework in total consists of 5 daemon processes namely:

1. **NameNode**: NameNode is utilized to store the Metadata (data about the area, size of files/blocks) for HDFS. The Metadata could be put away on RAM or Hard-Disk. There will dependably be just a single NameNode in a cluster. The only way that the Hadoop framework can fail is when the NameNode will crash.

2. **Secondary NameNode**: It is used as a backup for NameNode. It holds practically the same data as that of NameNode. On the off chance that NameNode falls flat, this one comes into the picture.

3. **DataNode**: The actual user files or data is stored on DataNode. The number of DataNode depends on your data size and can be increased with the need. The DataNode communicates to NameNode in a definite interval of time.

4. **Job Tracker**: NameNode and DataNodes store points of interest and genuine information on HDFS. This information is likewise required to process according to users' prerequisites. A Developer writes a code to process the information.

5. **Task Tracker**: The Jobs taken by Job Trackers are in genuinely performed by Task trackers. Each DataNode will have one task tracker. Task trackers communicate with Job trackers to send statuses of the undertaken job status.

HDFS bolsters the quick exchange of information between Masters and Slaves as it is combined with MapReduce, an automatic system for information handling and accessing information at a higher rate. When HDFS takes in information, it separates the data into partitioned squares and appropriates them to various nodes making the system effective via parallel processing.

2. MapReduce:

The map Reduction algorithm contains two important tasks, namely Map and Reduce.

- Mapping – Attained by Mapper Class

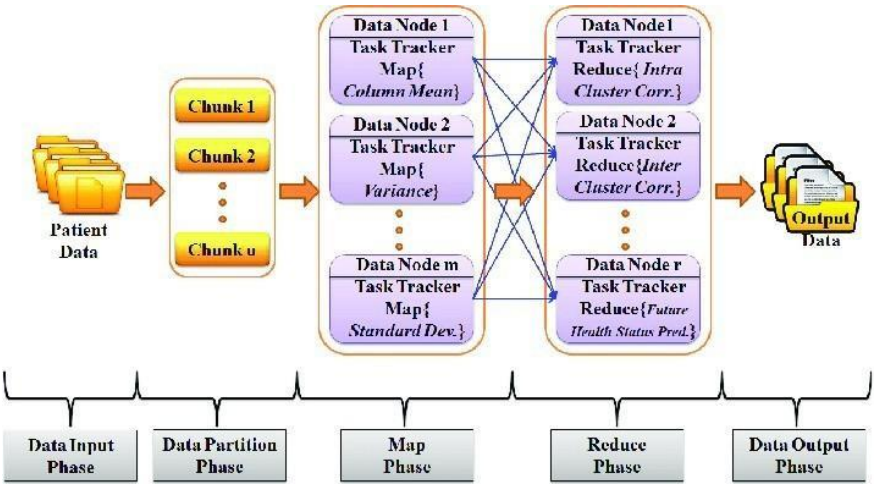
- Reduction – Attained by Reducer Class.

MapReduce utilizes different numerical calculations to separate an errand into little parts and dole out them to various frameworks. MapReduce calculation helps in sending the Map and Reduce errands to proper servers in a bunch. The tasks are executed in parallel in all the different nodes and finally, the result is returned to the user.

The Healthcare Industry uses it primarily for the following

- Data Warehouse Optimization
- Patient Analysis
- Predictive Maintenance

Hadoop uses the MapReduce algorithm to create tasks, called jobs which can be executed independently on different clusters (DataNodes) while the result is fetched back to a single node (NameNode) for output.



As can be seen above, the system will group together the items having the same key. Finally, the system provides the requested output.

Patient's data is stored in a centralized repository which makes the system cost effective by reducing number of storage warehouses as well as eliminates any sort of data redundancy, which leads to the system being consistent as well.

3. YARN (Yet Another Resource Negotiator):

Hadoop YARN (Yet Another Resource Negotiator) is a Hadoop ecosystem component that provides the resource management. Yarn is also one of the most important components of the Hadoop Ecosystem. YARN is called as the operating system of Hadoop as it is responsible for managing and monitoring workloads. It allows multiple data processing engines such as real-time streaming and batch processing to handle data stored on a single platform. Contribute to society and human well-being.

YARN has been projected as a data operating system for Hadoop. The main features of YARN are:

- **Flexibility** – Enables other purpose-built data processing models beyond MapReduce (batch), such as interactive and streaming. Due to this feature of YARN, other applications can also be run along with Map Reduce programs in Hadoop2.

- **Efficiency** – As many applications run on the same cluster, Hence, the efficiency of Hadoop increases without much effect on the quality of service.

- **Shared** – Provides a stable, reliable, secure foundation and shared operational services across multiple workloads. Additional programming models such as graph processing and iterative modeling are now possible for data processing.

4. HIVE:

The Hadoop ecosystem component, Apache Hive, is an open-source data warehouse system for querying and analyzing large datasets stored in Hadoop files. Hive does three main functions: *data summarization, query, and analysis*.

Hive use a language called HiveQL (HQL), which is similar to SQL. HiveQL automatically translates SQL-like queries into MapReduce jobs which will execute on Hadoop. Main parts of Hive are:

- **Metastore** – It stores the metadata.
- **Driver** – Manage the lifecycle of a HiveQL statement.
- **Query compiler** – Compiles HiveQL into Directed Acyclic Graph(DAG).
- **Hive server** – Provide a thrift interface and JDBC/ODBC server.

5. Pig:

Apache Pig is a high-level language platform for analyzing and querying huge dataset that are stored in HDFS. Pig as a component of Hadoop Ecosystem uses ***PigLatin*** language. It is very similar to SQL. It loads the data, applies the required filters and dumps the data in the required format. For Programs execution, pig requires Java runtime environment.

Features of Apache Pig:

- **Extensibility –** For carrying out special purpose processing, users can create their own function.
- **Optimization opportunities –** Pig allows the system to optimize automatic execution. This allows the user to pay attention to semantics instead of efficiency.
- **Handles all kinds of data –** Pig analyzes both structured as well as unstructured.

6. HBase:

Apache HBase is a Hadoop ecosystem component which is a distributed database that was designed to store structured data in tables that could have billions of row and millions of columns. HBase is scalable, distributed, and NoSQL database that is built on top of HDFS. HBase, provide real-time access to read or write data in HDFS.

Components of Hbase

There are two HBase Components namely- HBase Master and RegionServer.

HBase Master

- ➢ It is not part of the actual data storage but negotiates load balancing across all RegionServer.
- ➢ Maintain and monitor the Hadoop cluster**.**
- ➢ Performs administration (interface for creating, updating and deleting tables.)
- ➢ Controls the failover.
- ➢ HMaster handles DDL operation.

RegionServer

It is the worker node which handles read, writes, updates and delete requests from clients. Region server process runs on every node in Hadoop cluster. Region server runs on HDFS DateNode.

7. Mahout:

Mahout is open source framework for creating scalable machine learning algorithm and data mining library. Once data is stored in Hadoop HDFS, mahout provides the data science tools to automatically find meaningful patterns in

those big data sets.

Algorithms of Mahout are:

- **Clustering –** Here it takes the item in particular class and organizes them into naturally occurring groups, such that item belonging to the same group are similar to each other.
- **Collaborative filtering –** It mines user behavior and makes product recommendations (e.g. Amazon recommendations)
- **Classifications –** It learns from existing categorization and then assigns unclassified items to the best category.
- **Frequent pattern mining –** It analyzes items in a group (e.g. items in a shopping cart or terms in query session) and then identifies which items typically appear together.

Solr, Lucene (searching and indexing):-

Apache Lucene is a high-performance and full-featured text search engine library written entirely in Java from the Apache Software Foundation. It is a technology suitable for nearly any application that requires full-text search, especially in a cross-platform environment.

Lucene offers powerful features like scalable and high-performance indexing of the documents and search capability through a simple API. It utilizes powerful, accurate and efficient search algorithms written in Java.

Most importantly, it is a cross-platform solution. Lucene provides search over documents; where a document is essentially a collection of fields. A field consists of a field name that is a string and one or more field values. Lucene does not in any way constrain document structures. Fields are constrained to store only one kind of data, either binary, numeric, or text data. There are two ways to store text data: string fields store the entire item as one string; text fields store the data as a series of *tokens*. Lucene provides many ways to break a piece

of text into tokens as well as hooks that allow you to write custom tokenizers. Lucene has a highly expressive search API that takes a search query and returns a set of documents ranked by relevancy with documents most similar to the query having the highest score.

8. Conclusion:

We have covered all the Hadoop Ecosystem Components in detail. Hence these Hadoop ecosystem components empower Hadoop functionality on the Healthcare system.