CS4830: Lab Assignment-4
Shania Mitra CH18B067

Q1. Write a spark code for executing the Hash example provided in slide 14 on Hashing from Lab 1 Presentation, on the public file: 'gs://bdl2022/lab4_dataset.csv'. You would have to find the number of user clicks between 0-6, 6-12, 12-18, and 18-24, as was discussed in the first class.
Code and text file attached

Q2. Provide a brief description of the functionality of the following services:
a. HDFS
Hadoop Distributed File System(HDFS) can store a large quantity of structured as well as unstructured data. HDFS provides reliable storage for data with its unique feature of Data Replication. HDFS is a highly fault-tolerant, reliable, available, scalable, distributed file system.

1. Fault Tolerance

The fault tolerance in Hadoop HDFS is the working strength of a system in unfavourable conditions. It is highly fault-tolerant. Hadoop framework divides data into blocks. After that, it creates multiple copies of blocks on different machines in the cluster.

So, when any machine in the cluster goes down, then a client can easily access their data from the other machine which contains the same copy of data blocks.

2. High Availability

Hadoop HDFS is a highly available file system. In HDFS, data gets replicated among the nodes in the Hadoop cluster by creating a replica of the blocks on the other slaves present in HDFS cluster. So, whenever a user wants to access this data, they can access their data from the slaves which contain its blocks.

At the time of unfavourable situations like a failure of a node, a user can easily access their data from the other nodes. Because duplicate copies of blocks are present on the other nodes in the HDFS cluster.

3. High Reliability

HDFS provides reliable data storage. It can store data in the range of 100s of petabytes. HDFS stores data reliably on a cluster. It divides the data into blocks. Hadoop framework stores these blocks on nodes present in HDFS clusters.

HDFS stores data reliably by creating a replica of each and every block present in the cluster. Hence provides a fault tolerance facility. If the node in the cluster containing data goes down, then a user can easily access that data from the other nodes.

HDFS by default creates 3 replicas of each block containing data present in the nodes. So, data is quickly available to the users. Hence the user does not face the problem of data loss. Thus, HDFS is highly reliable.

4. Replication

Data Replication is a unique feature of HDFS. Replication solves the problem of data loss in an unfavourable condition like hardware failure, crashing of nodes etc. HDFS maintains the process of replication at regular intervals of time.

HDFS also keeps creating replicas of user data on different machines present in the cluster. So, when any node goes down, the user can access the data from other machines. Thus, there is no possibility of losing user data.

5. Scalability

Hadoop HDFS stores data on multiple nodes in the cluster. So, whenever requirements increase you can scale the cluster. Two scalability mechanisms are available in HDFS: Vertical and Horizontal Scalability.

6. Distributed Storage

All the features in HDFS are achieved via distributed storage and replication. HDFS stores data in a distributed manner across the nodes. In Hadoop, data is divided into blocks and stored on the nodes present in the HDFS cluster.

After that HDFS creates the replica of each and every block and store on other nodes. When a single machine in the cluster gets crashed we can easily access our data from the other nodes which contain its replica.

b. Hive

The three important functionalities for which Hive is deployed are data summarization, data analysis, and data query. The query language, exclusively supported by Hive, is HiveQL. This language translates SQL-like queries into MapReduce jobs for deploying them on Hadoop. HiveQL also supports MapReduce scripts that can be plugged into the queries. Hive increases schema design flexibility and also data serialisation and deserialization.

Hive is best suited for batch jobs, rather than working with web log data and append-only data. It cannot work for online transaction processing (OLTP) systems since it does not provide real-time querying for row-level updates.

c. Pig

Apache Pig is an abstraction over MapReduce. It is a tool/platform which is used to analyse larger sets of data representing them as data flows. Pig is generally used with Hadoop; we can perform all the data manipulation operations in Hadoop using Apache Pig.

To write data analysis programs, Pig provides a high-level language known as Pig Latin. This language provides various operators using which programmers can develop their own functions for reading, writing, and processing data.

Apache Pig comes with the following features −

- Rich set of operators − It provides many operators to perform operations like join, sort, filer, etc.
- Ease of programming − Pig Latin is similar to SQL and it is easy to write a Pig script if you are good at SQL.
- Optimization opportunities − The tasks in Apache Pig optimise their execution automatically, so the programmers need to focus only on semantics of the language.
- Extensibility − Using the existing operators, users can develop their own functions to read, process, and write data.
- UDF's − Pig provides the facility to create User-defined Functions in other programming languages such as Java and invoke or embed them in Pig Scripts.
- Handles all kinds of data − Apache Pig analyzes all kinds of data, both structured as well as unstructured. It stores the results in HDFS.

d. Yarn

YARN stands for "*Yet Another Resource Negotiator*". YARN gained popularity because of the following features-

- Scalability: The scheduler in Resource manager of YARN architecture allows Hadoop to extend and manage thousands of nodes and clusters.
- Compatibility: YARN supports the existing map-reduce applications without disruptions thus making it compatible with Hadoop 1.0 as well.
- Cluster Utilisation:Since YARN supports Dynamic utilisation of clusters in Hadoop, which enables optimised Cluster Utilisation.
- Multi-tenancy: It allows multiple engine access thus giving organisations a benefit of multi-tenancy.
