

Devanshu Suriana
RC-23, 1032210755

Panel C, BDT - Batch & Interactive

BDT Lab - Assignment 5

Problem statement: Perform Data Analysis using Map-Reduce in Hadoop / PySpark

Objectives: ① To learn concepts of Map Reduce. ② To learn how to do analysis in Hadoop.

Theory: ① Introduction to Map Reduce

Map Reduce is a programming paradigm that enables massive scalability across hundreds or thousands of servers in a Hadoop cluster. As the processing component, Map Reduce is the heart of Apache Hadoop. The term "Map Reduce" refers to two different tasks that Hadoop programs perform.

Map Reduce programming offers several benefits to help you gain valuable insights from your big data:

- Scalability
- Flexibility
- Speed

Simple

② Working of Map Reduce with example

Assume you have 5 files, and each file contains

2 columns (a key and value in Hadoop terms) that represent a city and the corresponding temperature recorded in that city for the various measurement days. The city is the key and the temp. is the value. For eg: (Toronto, 20)
 Out of all the data collected, you want to find the max temperature for each city across the data files.

Using MapReduce (you can break this into 5 map tasks, for eg the results produced one mapper task for the above data would look like: (Toronto, 20) (Whitby, 25) (New York, 22) (Rome, 33)

Assume the other 4 mappers yield this:

(T, 18) (W, 27) (NY, 32) (R, 37) (T, 32) (W, 20)
 (NY, 33) (R, 38) (T, 22) (W, 19) (NY, 20) (R, 31)
 (W, 22) (NY, 19) (R, 30)

All five of these output streams would be fed into the reduce tasks which combine the input result and yield a single value for each city.

The final answer would be as follows:

(Toronto, 32) (Whitby, 27) (New York, 33) (Rome, 38)

Platform: 64-bit Open Source Linux/windows.

Conclusion: Hence, I learned the Map Reduce concept applying on data set in Hadoop environment.

FAQ's

Ans. 1) DFS (Distributed File System):

- Storage system: Hadoop's primary storage for large files, spread across multiple machines.
- Master slave setup: One master (NameNode) manages meta data, multiple slaves (Data Nodes) store data.
- Replication For Reliability: Data is replicated for fault tolerance.

YARN (Yet Another Resource Negotiator):

- Resource Manager: Manager and scheduler resource in a Hadoop cluster.
- Separates Processing Engines: Allows diffⁿ processing engine to run on the same cluster.
- Scalable: Can handle large clusters with 1000s of nodes.

Ans 2) Shuffling in Map Reduce is the process of transferring the data from mappers to reducers, involving partitioning, and intermediate key-value pairs across the cluster.

Sorting in MapReduce refers to arrangement of key-value pairs during the shuffling phase, ensuring that each reducer receives data with keys in a sorted order for efficient processing.

Ans 3) 1) Scalability 2) Fault Tolerance 3) Parallel Processing 4) Cost - Effective 5) Flexibility 6) Ease of Programming

- 1) Scalability
- 2) Fault Tolerance
- 3) Parallel Processing
- 4) Cost - Effective
- 5) Flexibility
- 6) Ease of Programming

Scalability is the ability of a system to handle an increasing amount of work or to be enlarged to accommodate that work. Fault Tolerance is the ability of a system to continue to operate in the event of a failure of one or more components. Parallel Processing is the simultaneous execution of multiple tasks or processes. Cost - Effective means that the system provides good performance at a low cost. Flexibility is the ability of a system to adapt to changing requirements. Ease of Programming is the ability of a system to be programmed easily.

When a system is scaled, it must be able to handle the increased load without a significant increase in cost. Fault Tolerance is achieved by having redundant components that can take over in the event of a failure. Parallel Processing is achieved by having multiple processors that can execute tasks simultaneously. Cost - Effectiveness is achieved by using low-cost components and optimizing the system architecture. Flexibility is achieved by having a modular architecture that allows for easy expansion and modification. Ease of Programming is achieved by using high-level programming languages and providing a rich set of development tools.

~~Scalability~~ is the ability of a system to handle an increasing amount of work or to be enlarged to accommodate that work. Fault Tolerance is the ability of a system to continue to operate in the event of a failure of one or more components. Parallel Processing is the simultaneous execution of multiple tasks or processes. Cost - Effectiveness is achieved by using low-cost components and optimizing the system architecture. Flexibility is achieved by having a modular architecture that allows for easy expansion and modification. Ease of Programming is achieved by using high-level programming languages and providing a rich set of development tools.

Scalability is the ability of a system to handle an increasing amount of work or to be enlarged to accommodate that work. Fault Tolerance is the ability of a system to continue to operate in the event of a failure of one or more components. Parallel Processing is the simultaneous execution of multiple tasks or processes. Cost - Effectiveness is achieved by using low-cost components and optimizing the system architecture. Flexibility is achieved by having a modular architecture that allows for easy expansion and modification. Ease of Programming is achieved by using high-level programming languages and providing a rich set of development tools.