Big Data (MapReduce vs Spark)

K Sachith Rangana

Table of content

Introduction for Apache Spark

- Powerful distributed computing framework.
- Enables processing of large datasets with in-memory computing with fault tolerance.
- Supports batch processing, real-time stream processing, machine learning and graph processing.
- Offers speed, flexibility and power for tackling diverse data processing challenges.

Introduction to MapReduce

- A program model for processing big data.
- Two phases: Map phase (data processing) and Reduce phase (aggregation)
- Introduced by google for large-scale data processing and popularized by apache hadoop.
- Splits data into small chunks and processes them in parallel.
- Handles node failure by redistributing tasks to other nodes.
- Scale horizontally to handle large datasets across distributed environments.

Demo

With Apache spark and MapReduce

MapReduce vs Apache Spark Comparison

MapReduce

Performance: process data sequentially, leading to slower processing time.

Ease of Use: require explicit management of job flow serialization and deserialization.

Fault tolerance: completely relies on hadoop HDFS.

Scalability: scales horizontally.

Apache Spark

Performance: perform in-memory computing, which result in faster processing speed.

Ease of Use: provides higher-level abstraction like DataFrame api

Fault tolerance: achieve through minimizing data replication with improving efficiency.

Scalability: offers better scalability due to its in-memory computing capabilities and optimized execution model

Conclusion

- MapRuduce and Apache spark are both powerful frameworks for data processing each with its strengths and weaknesses.
- While MapReduce is well-established and integrated with hadoop ecosystem tools, Apache Spark offers superior performance, ease of use, fault tolerance and scalability.

References

- Solution github url: https://github.com/sachithr7/248367H_SachithR_BigData_Assignment
- Demo outcomes document: https://github.com/sachithr7/248367H_SachithR_BigData_Assignment/blob/main/UoM_MapReduce-vs-Spark/248367H%20-%20Rangana%20K.S.%20-%20Big%20Data%20Assignment%20Analysis.pdf
- Solution outcome jupyter notebook:

  https://github.com/sachithr7/248367H_SachithR_BigData_Assignment/blob/main/UoM_MapReduce-vs-Spark/Fully_completed_final_solution_with_shell_scripting_spark_and_map_reduce/output/Rangana%20K.S.%20248367H%20-%20Big%20Data%20Assignment.ipynb