Difference Between Hadoop and Spark

Talha Hanif Butt, L18-1864, FAST-NUCES, Lahore

October 15, 2018

With multiple big data frameworks available on the market, choosing the right one is a challenge. A classic approach of comparing the pros and cons of each platform is unlikely to help, as businesses should consider each framework from the perspective of their particular needs. Facing multiple Hadoop MapReduce vs. Apache Spark requests, our big data consulting practitioners compare two leading frameworks to answer a burning question: which option to choose Hadoop MapReduce or Spark.

Both Hadoop and Spark are open source projects by Apache Software Foundation and both are the flagship products in big data analytics. Hadoop has been leading the big data market for more than 5 years. According to our recent market research, Hadoops installed base amounts to 50,000+ customers, while Spark boasts 10,000+ installations only. However, Sparks popularity skyrocketed in 2013 to overcome Hadoop in only a year. A new installation growth rate (2016/2017) shows that the trend is still ongoing. Spark is outperforming Hadoop with 47 percent vs. 14 percent correspondingly.

1 Tasks Hadoop MapReduce is good for:

1.1 Linear processing of huge data sets

Hadoop MapReduce allows parallel processing of huge amounts of data. It breaks a large chunk into smaller ones to be processed separately on different data nodes and automatically gathers the results across the multiple nodes to return a single result. In case the resulting dataset is larger than available RAM, Hadoop MapReduce may outperform Spark.

1.2 Economical solution, if no immediate results are expected

For instance, if data processing can be done during night hours, it makes sense to consider using Hadoop MapReduce.

2 Tasks Spark is good for:

2.1 Fast data processing

In-memory processing makes Spark faster than Hadoop MapReduce up to 100 times for data in RAM and up to 10 times for data in storage.

2.2 Iterative processing

If the task is to process data again and again Spark defeats Hadoop MapReduce. Sparks Resilient Distributed Datasets (RDDs) enable multiple map operations in memory, while Hadoop MapReduce has to write interim results to a disk.

2.3 Near real-time processing

If a business needs immediate insights, then they should opt for Spark and its in-memory processing.

2.4 Graph processing

Sparks computational model is good for iterative computations that are typical in graph processing. And Apache Spark has GraphX an API for graph computation.

2.5 Machine learning

Spark has MLlib a built-in machine learning library, while Hadoop needs a third-party to provide it. MLlib has out-of-the-box algorithms that also run in memory.

2.6 Joining datasets

Due to its speed, Spark can create all combinations faster, though Hadoop may be better if joining of very large data sets that requires a lot of shuffling and sorting is needed.

3 Examples of practical applications

3.1 Customer segmentation

Analyzing customer behavior and identifying segments of customers that demonstrate similar behavior patterns will help businesses to understand customer preferences and create a unique customer experience.

3.2 Risk management

Forecasting different possible scenarios can help managers to make right decisions by choosing non-risky options.

3.3 Real-time fraud detection

After the system is trained on historical data with the help of machine-learning algorithms, it can use these findings to identify or predict an anomaly in real time that may signal of a possible fraud.

3.4 Industrial big data analysis

Its also about detecting and predicting anomalies, but in this case, these anomalies are related to machinery breakdowns. A properly configured system collects the data from sensors to detect pre-failure conditions.