Spark与Hadoop的对比

那么Spark与我们之前学习过的Hadoop生态圈有什么区别呢？这一节我们将比较一下Spark和Hadoop，Spark和Hadoop同属于Apache生态系统的两兄弟。

Spark已经取代Hadoop成为最活跃的开源大数据项目。但是，在选择大数据框架时，企业不能因此就厚此薄彼。近日，著名大数据专家Bernard Marr在一篇文章中分析了Spark和Hadoop的异同。

Hadoop和Spark均是大数据框架，都提供了一些执行常见大数据任务的工具。但确切地说，它们所执行的任务并不相同，彼此也并不排斥。虽然在特定的情况下，Spark据称要比Hadoop快100倍，但它本身没有一个分布式存储系统。而分布式存储是如今许多大数据项目的基础。它可以将PB级的数据集存储在几乎无限数量的普通计算机的硬盘上，并提供了良好的可扩展性，只需要随着数据集的增大增加硬盘。因此，Spark需要一个第三方的分布式存储。也正是因为这个原因，许多大数据项目都将Spark安装在Hadoop之上。这样，Spark的高级分析应用程序就可以使用存储在HDFS中的数据了。

与Hadoop相比，Spark真正的优势在于速度。Spark的大部分操作都是在内存中，而Hadoop的MapReduce系统会在每次操作之后将所有数据写回到物理存储介质上。这是为了确保在出现问题时能够完全恢复，但Spark的弹性分布式数据存储也能实现这一点。

spark是对MapReduce计算模型的改进，可以说没有HDFS，MapReduce，就没有spark，Spark可以使用YARN作为它的集群管理器，并且可以处理HDFS的数据。这对于已经部署Hadoop集群的用户特别重要，毕竟不需要做任何的数据迁移就可以使用Spark的强大处理能力。另外，在高级数据处理（如实时流处理和机器学习）方面，Spark的功能要胜过Hadoop。这一点连同其速度优势是Spark越来越受欢迎的真正原因。实时处理意味着可以在数据捕获的瞬间将其提交给分析型应用程序，并立即获得反馈。在各种各样的大数据应用程序中，这种处理的用途越来越多，比如，零售商使用的推荐引擎、制造业中的工业机械性能监控。Spark平台的速度和流数据处理能力也非常适合机器学习算法。这类算法可以自我学习和改进，直到找到问题的理想解决方案。这种技术是最先进制造系统（如预测零件何时损坏）和无人驾驶汽车的核心。Spark有自己的机器学习库MLib，而Hadoop系统则需要借助第三方机器学习库，如Apache Mahout。实际上，虽然Spark和Hadoop存在一些功能上的重叠，但它们都不是商业产品，并不存在真正的竞争关系，而通过为这类免费系统提供技术支持赢利的公司往往同时提供两种服务。例如，Cloudera就既提供Spark服务也提供Hadoop服务，并会根据客户的需要提供最合适的建议。

虽然Spark发展迅速，但它尚处于起步阶段，安全和技术支持基础设施方还不发达。Spark在开源社区活跃度的上升，表明企业用户正在寻找已存储数据的创新用法。

首先我们来看一下两者框架的区别：在Hadoop平台中，MapRedcue由Map和Reduce两个阶段，并通过shuffle将两个阶段连接起来的。

但是套用MapReduce模型解决问题，不得不将问题分解为若干个有依赖关系的子问题，每个子问题对应一个MapReduce作业，最终所有这些作业形成一个DAG。而Spark是通用的DAG框架，可以将多个有依赖关系的作业转换为一个大的DAG。

核心思想是将Map和Reduce两个操作进一步拆分为多个元操作，这些元操作可以灵活组合，产生新的操作，并经过一些控制程序组装后形成一个大的DAG作业。

其次中间结果的存储方式也有很大差别：在DAG中，由于有多个MapReduce作业组成，每个作业都会从HDFS上读取一次数据和写一次数据（默认写三份），即使这些MapReduce作业产生的数据是中间数据也需要写HDFS。

这种表达作业依赖关系的方式比较低效，会浪费大量不必要的磁盘和网络I/O，根本原因是作业之间产生的数据不是直接流动的，而是借助HDFS作为共享数据存储系统。在Spark中，使用内存（内存不够使用本地磁盘）替代了使用HDFS存储中间结果。

对于迭代运算效率更高。

在操作模型上，Hadoop只提供了Map和Reduce两种操作，所有的作业都得转换成Map和Reduce的操作，而Spark提供很多种的数据集操作类型

比如Transformations 包括map, filter, flatMap, sample, groupByKey, reduceByKey, union, join, cogroup, mapValues,sort,partionBy等多种操作类型，还提供actions操作包括Count,collect, reduce, lookup, save等多种。

这些多种多样的数据集操作类型，给开发上层应用的用户提供了方便。

编程模型上：Hadoop就是唯一的Data Shuffle一种模式，spark用户可以命名，物化，控制中间结果的存储、分区等，编程方式更灵活，Hadoop无法缓存数据集，spark的60%内存用来缓存弹性分布式数据集（RDD），对于缓存后的rdd进行操作，节省IO，效率高。Hadoop适合离线大规模分析处理，在只有map操作或者只有一次reduce操作的场景下，Spark比Hadoop的优势不明显)

□ 对于迭代计算比Hadoop有更大的优势，spark使用scala语言，更简洁高效

 spark对机器学习算法，图计算能力有很好的支持。总的来说，Spark采用更先进的架构，使得灵活性、易用性、性能等方面都比Hadoop更有优势，有取代Hadoop的趋势，但其稳定性有待进一步提高。