|  |
| --- |
| 中南大学 |
| 大数据技术中的Hadoop与Spark对比分析 |
| 寒假读书实践活动报告 |



专业班级：物联网1401班

姓 名：刘莹莹

学 号：0919140124

# 大数据技术中的Hadoop与Spark对比分析

阅读书目：

1、《 Hadoop系统性能优化与功能增强综述 （Performance Optimization and Feature Enhancements of Hadoop System）》作者：董新华、李瑞轩、周湾湾、王聪、薛正元、廖东杰

2、《LUPA开源周刊：大数据技术的回顾与展望 ——写在Hadoop十周年纪念》

3、《 LUPA开源周刊：Ubuntu携手甲骨文 Hadoop喜迎10周年 》

4、《 LUPA开源社区： Spark和Hadoop，孰优孰劣？》

5、吴韶鸿《大数据开源技术发展研究（ModernScience& Technology of Telecommunications）》【期刊论文】-商洛学院学报

6、《Dropbox邵铮:我怎么看Hadoop Summit 2015和Spark Summit 2015？》

## 什么是大数据

大数据（big data）是继云计算和物联网之后IT产业的又一次颠覆性变革，是经济和技术发展的必然结果。维基百科、数据科学家以及一些研究机构和相关厂商均提出过大数据的概念，然而截至目前并未形成统一的定义。而大数据所蕴含的4V特征，即体量大（volume）、式多（variety）、速度快（velocity）、价值密度低（value）,增加了数据管理和信息提取的困和复杂性，导致常规的数据处理方式已经无法适应需求，继续数据处理模式的转变。因此，有关大数据的存储和管理成为时下研究的热点。

云计算和云存储成为大规模数据处理的新型产物，而目前流行的hadoop就是一种实现云存储和云计算的方法，hadoop具有可靠性、高扩展性和高容错性等优点。

什么是大数据？麦肯锡公司的报告《大数据：创新、竞争和生产力的下一个前沿领域》中给出的大数据定义是：大数据指的是规模超过现有数据库工具获取、存储、管理和分析能力的数据集，并同时强调并不是超过某个特定数量级的数据集才是大数据。

国际数据公司（IDC）用四个维度的特征来定义大数据，即数据集的规模（Volume）、数据流动的速度（Velocity）、数据类型的多少（Variety）和数据价值的大小（Value）。

亚马逊的大数据科学家John Rauser的定义比较直接：“超过单台计算机处理能力的数据量则为大数据”。

最后我们来看看维基百科上的大数据定义：“Big data is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications. ”翻译成中文的意思是：大数据指的是数据规模庞大和复杂到难以通过现有的数据库管理工具或者传统的数据处理应用程序进行处理的数据集合。

上述大数据的概念中无一例外都突出了“大”字。从表面上看，数据规模的增长的确为处理数据带来了很大的问题。具体来说，在同样时间内获取与以前相同价值的数 据变得不可为了。换言之，本质问题是数据的价值密度变低了，数据交换速率变慢了，所以催生了很多新型数据处理技术和工具，如Google的GFS和 MapReduce，Apache Hadoop生态系统，美国伯克利大学AMPLab的Spark等；出现了对时间敏感程度不同的计算模式，如批式计算模式、交互式计算模式、流计算模式、 实时计算模式等。计算模式的差异只是决定获取价值的技术不同，取决于上层业务需求的不同。实际上，所谓大数据问题的本质应是数据的资产化和服务化，而挖掘 数据的内在价值是研究大数据的最终目标。

目前大数据开源项目主要集中在计算处理和数据管理领域，其中hadoop和spark主键发展成为可以为大数据应用提供各个层面技术支持的开源生态，受到了业界广泛关注和应用。大数据分析处理流程中所使用的关键技术几乎都源自开源模式，指明的大数据开源项目如分布式计算和存储系统hadoop、基于内存计算的集群计算系统spark，以及多款非关系型数据库NoSQL.

## hadoop

hadoop开源项目采用分布式系统架构，共包含四个模块，基于hadoop项目发展起来的重要开源项目共有十余个，它们共同为大数据应用提供各个层次的技术支持，是目前应用最广、关注度最高的大数据开源技术生态。

Hadoop于2006年1月28日诞生，已有10年，它改变了企业对数据的存储、处理和分析的过程，加速了大数据的发展，形成了自己的极其火爆的技术生态圈，并受到非常广泛的应用。Hadoop是 为了大数据而诞生的，其本身就象征着本世纪工业革命的焦点：业务的数字化转型。十年前，数字化业务仅仅在少数几个行业中得到了应用，例如电子商务和媒体。 从那时起，我们已经看到数字化技术将成为几乎所有行业必不可少的一环。每一个行业都正在致力于围绕其信息系统构建数据驱动型运营模式。像Hadoop之类 的大数据工具可以使各行业能够从他们所产生的数据中获得最大的利益。

## spark

spark 是一款开源的、基于内存计算的集群计算系统，能够对大数据进行快速分析处理。

spark项目最早出现于2010年，由加州伯克利大学AMP实验室开发，采用Scala编程语言

编写，代码十分精简，核心部分只有63个Scala语言文件。AMP实验室开发spark项目，

主要是用其构建大型、低延迟的数据分析、自然语言处理分析、交通路况预测以及为数据

分析服务提速等。

spark项目也可以与hadoop、亚马逊AWS等其他云计算和大数据项目集成使用。spark项目逐渐成为大数据开源计算技术领域的重要组成，其处理速度快，使用简单以及技术生态

丰富等优势，将推动项目生态在未来进一步快速发展。

## Hadoop与Spark的优缺点对比分析

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。

　　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。在Bernard看来，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。

　　Bernard认为，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。在他看来，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。Spark与hadoop MapReduce均为开源集群计算系统，但两者适用的场景并不相同。其中，Spark基于内存计算实现，可以以内存速度进行运算，优化工作负载迭代过程，加快数据分析处理速度；Hadoop MapReduce以批处理方式处理数据，每次启动任务后，需要等待较长时间才能获得加过。在机器学习和数据库查询等数据计算过程中，Spark的处理速度可以达到Hadoop MapReduce 的100倍以上。因此，对于实时要求比较高的计算处理应用，Spark更加适用；对于海量数据分析的非实时计算应用，Hadoop MapReduce更加适用；同时，相比Hadoop MapReduce，Spark代码更加精简，且其API接口能够支持

Java、Scala和Python等常用编程语言，更方便用户使用。

## 发展趋势

关于为什么Spark的发展速度比Hadoop更快，通过以上所列书籍与期刊所讲，有以下原因：

1. Spark非常容易使用。Spark Notebook，Spark与Java/Scala/Python/R的互操作性都做得非常好。而Hadoop的早期用户和社区的主要贡献者都来自于大公司，服务于资深用户。资深用户更关注功能是否完善、系统是否稳定，而易用性就不是主要的考虑因素。

2. Spark是为交互式使用设计的。这体现在聚焦于规模较小的数据处理应用，因而使用内存来加速变得非常重要。这也体现在剔除很多不必要的开销，例如JVM启动时间、polling/heartbeat interval、用来防止出现Self-DDOS的sleep/wait。而Hadoop社区的决定者很多都是大公司。在那里，超大规模的数据计算是最重要的，而几秒钟的启动时间和等待都是无关紧要的。

3. Spark的Committer非常注重发展外部的代码贡献者。一开始，辅导外部的代码贡献者来提交patch可能比Committer自己写code提交patch更慢，但是辅导外部的代码贡献者是一个很好的投资，可以有长期的回报。显然，这个策略在Spark身上非常奏效。

相对来说，Spark的技术比较新，所以运维稳定性、调试等方面不及Hadoop的相关技术。今年Berkeley AMPLAB就专门在USENIX NSDI 2015上发表了一篇文章 Making Sense of Performance in Data Analytics Frameworks 来讲述如何调试Spark的性能问题。

根据目前各专家书籍中提到与讨论，Hadoop与Spark的发展趋势大致如下：

1. Hadoop技术进一步的成熟。Hadoop最近的比较大的进步都是在运维稳定性和性能上的，例如HA（High Availablility）for YARN ResourceManager，Rolling Upgrades，Erasure Coding Support inside HDFS 等等。 相对来说，用户可用的新功能较少。

2. Spark在Machine Learning和Data Science/Statistics用户中的普及非常快。Spark Notebook，MLLib，SparkR 是Spark的几个杀手级的产品。SparkSQL中的DataFrame也是一个非常有效的功能，但SparkSQL在Data Warehouse领域（如ETL，BI等）的前景还有待进一步的观察，因为SparkSQL毕竟是后来者。

3. Spark和Hadoop的生态系统在融合。这点可以参见Hadoop & Spark, Perfect Together。Hadoop和Spark各自都有很多子项目。对于一个大数据的高级用户来说，他/她所做的决定一定不是”我到底用Hadoop还是Spark“，而是"我到底用Hadoop的哪些组件和Spark的哪些组件"。所以，对Hadoop和Spark的各个子项目的了解变得非常重要。

(注：以上内容均根据所列书籍与期刊杂志总结而来，如有错误不当之处，可查找原文)