HZ BOOKS 华章IT

[PACKT] PUBLISHING

数据分析师和数据科学家的便捷参考指南

系统讲解利用Hadoop和Spark及其生态系统里的一系列工具进行大数据分析的方法及实践

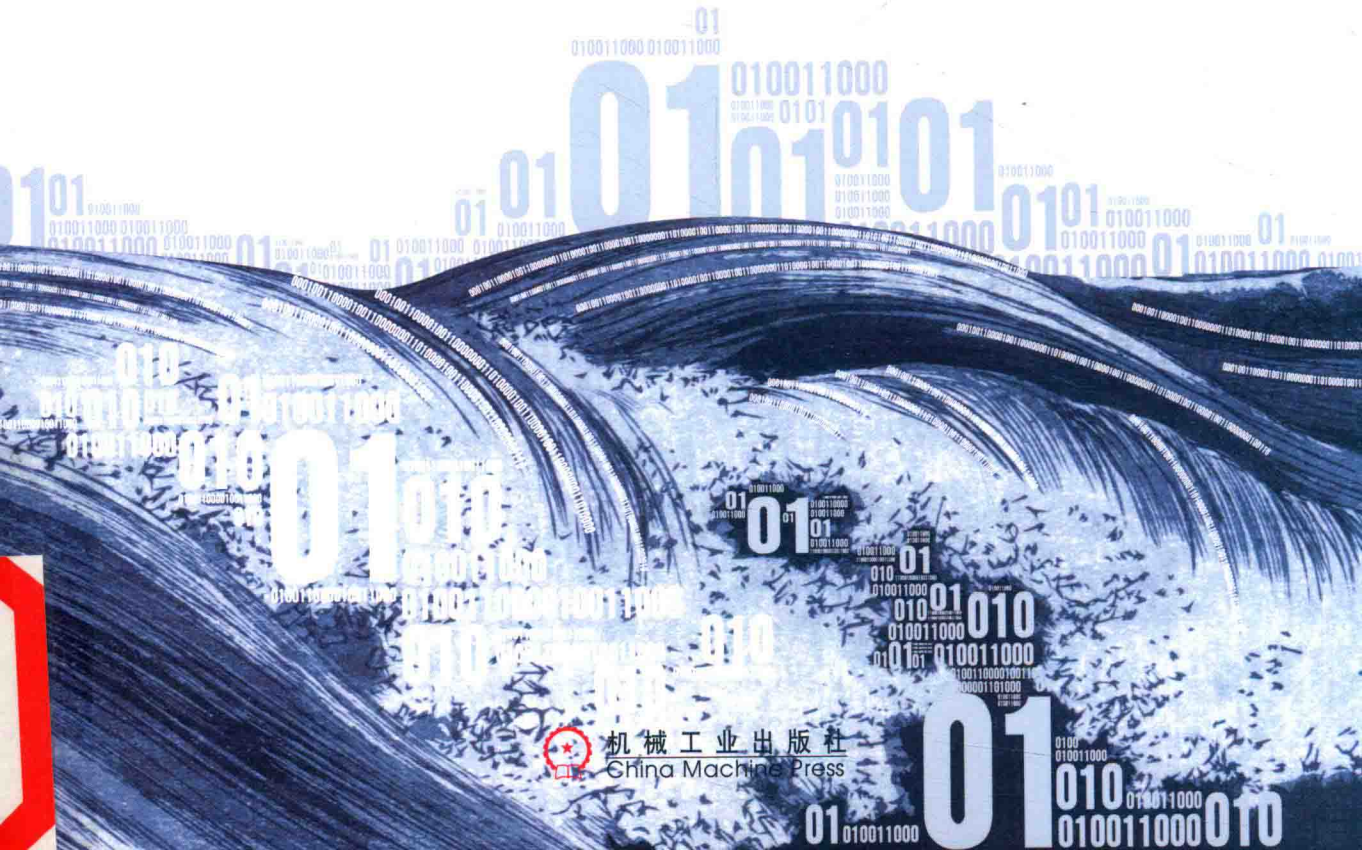
大数据技术丛书

Big Data Analytics

Spark与Hadoop大数据分析

[美] 文卡特·安卡姆（Venkat Ankam） 著

吴今朝 译



机械工业出版社
China Machine Press

大数据技术丛书

Big Data Analytics

Spark与Hadoop大数据分析

[美] 文卡特·安卡姆（Venkat Ankam） 著

吴今朝 译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

Spark 与 Hadoop 大数据分析 /（美）文卡特·安卡姆（Venkat Ankam）著；吴今朝译．
—北京：机械工业出版社，2017.6
（大数据技术丛书）
书名原文：Big Data Analytics

ISBN 978-7-111-56941-1

I. S… II. ① 文… ② 吴… III. 数据处理软件 IV. TP274

中国版本图书馆 CIP 数据核字（2017）第 117319 号

本书版权登记号：图字：01-2016-8647

Spark 与 Hadoop 大数据分析

出版发行：机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码：100037）
责任编辑：缪 杰
责任校对：殷 虹
印 刷：北京诚信伟业印刷有限公司
版 次：2017 年 7 月第 1 版第 1 次印刷
开 本：186mm × 240mm 1/16
印 张：15.5
书 号：ISBN 978-7-111-56941-1
定 价：59.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换
客服热线：（010）88379426 88361066
投稿热线：（010）88379604
购书热线：（010）68326294 88379649 68995259
读者信箱：hzit@hzbook.com

The Translator's Words 译 者 序

目前，大数据已经成了一个热点，各个专业领域都在利用大数据技术解决复杂的业务问题。与此同时，很多业务人员和技术人员对大数据技术还不太了解，觉得大数据技术背后的技术平台很复杂、很深奥。而本书就可以让读者循序渐进地熟悉目前主流的大数据技术平台。

本书比较系统地讲解了利用 Hadoop 和 Spark 及其生态系统里的一系列工具进行大数据分析的方法，并配套了详细的示例，是一本比较实用的参考书。

大家可以按照本书的内容循序渐进地学习。本书的难度并不大，绝大部分内容都配有详细的说明和实践步骤。偶尔需要补充一些背景知识，我会尽力用译者注的形式进行说明，希望对读者能有所帮助。

我们可以看到，Hadoop 和 Spark 实际上并不是相互竞争的关系，二者可以搭配使用，相互补充，为大数据分析人员提供一个全面和统一的技术框架，使之能够专注于业务需求，而无须在技术实现层面花费太多精力。

本书的定位主要是在大数据技术平台的搭建和配置方面。虽然原书书名是《 Big Data Analytics 》，但本书的核心内容是大数据分析的基础架构及实施方法，而不是大数据的分析方法，比如书中对于示例中用到的机器学习模型只有比较简略的讲解。

从这个角度来说，本书比较适合大数据分析的技术支持人员，以及对机器学习算法和模型已有一定造诣，希望学习利用最新的技术平台进行分析的独立研究者。

如果读者对机器学习的算法和模型感兴趣，可以参考我之前翻译的《预测分析：R 语言实现》（书号是：978-7-111-55354-0 ），该书比较深入地讲解了机器学习常用的一些模型，并且有详细的示例帮助读者进行练习。

和以往一样，我在 GitHub 上为本书开通了一个讨论区，网址是 https://github.com/coderLMN/BigDataAnalytics/issues。如果读者在阅读中遇到问题或错误，欢迎来这里提出，更欢迎参与讨论。谢谢！

根据我之前的经验，这样的讨论区对于不少读者来说是很实用的。他们不仅能提出问题、参与讨论，也可以提出自己的观点和实现方法，让自己、译者、其他读者都能从中获益。

在此我要感谢贾立恒等读者在讨论中给我带来的启发。另外，他们在学习过程中表现出来的认真和严谨对我也是一种无声的督促，让我在翻译的过程中不敢懈怠，时刻提醒自己要对翻译出来的文字负责。

最后，我要感谢我的家人，他们对我的翻译工作给予了极大的耐心和理解，让我能专心地从事这项我热爱的工作。

吴今朝

Preface 前　言

本书讲解了 Apache Spark 和 Hadoop 的基础知识，以及如何通过简单的方式将它们与最常用的工具和技术集成在一起。所有 Spark 组件（Spark Core、Spark SQL、DataFrame、Dataset、Conventional Streaming、Structured Streaming、MLlib、GraphX 和 Hadoop 核心组件）、HDFS、MapReduce 和 Yarn 都在 Spark + Hadoop 集群的实现示例中进行了深入的探讨。

大数据分析行业正在从 MapReduce 转向 Spark。因此，本书深入讨论了 Spark 相比 MapReduce 的优势，从而揭示出它在内存计算速度方面的好处。我们还会讲解如何运用 DataFrame API、Data Sources API，以及新的 Dataset API 来构建大数据分析应用程序。书中还讲解了如何使用 Spark Streaming 配合 Apache Kafka 和 HBase 进行实时数据分析，以帮助构建流式应用程序（streaming application）。新的结构化流（Structured Streaming）概念会通过物联网（Internet of Things，IOT）用例来解释。在本书中，机器学习技术会使用 MLLib、机器学习流水线和 SparkR 来实现；图分析则会利用 Spark 的 GraphX 和 GraphFrames 组件包来进行。

本书还介绍了基于 Web 的笔记本（如 Jupyter 和 Apache Zeppelin）和数据流工具 Apache NiFi，它们用于分析和可视化数据，以及利用 Livy 服务器把 Spark 作为一个服务提供给用户。

本书包含的内容

第 1 章从宏观的角度讲解了大数据分析的概念，并介绍了在 Apache Hadoop 和 Apache Spark 平台上使用的工具和技术，以及一些最常见的用例。

第 2 章介绍了 Hadoop 和 Spark 平台的基础知识。该章还讲解了 Spark 与 MapReduce 有何不同，以及 Hadoop 平台上的 Spark 有哪些优点。随后介绍如何安装集群，以及如何设置分析所需的工具。

第 3 章介绍了 Spark 的更深层概念，例如 Spark Core 内部结构、如何使用键值对

RDD、Spark 程序的生命周期、如何构建 Spark 应用程序、如何持久化和缓存 RDD，以及如何使用 Spark 资源管理器（Standalone、Yarn 和 Mesos）。

第 4 章涵盖了 Data Sources API、DataFrames API 和新的 Dataset API。本章会特别重点地讲解 DataFrame API 的用途，以及如何对具有内置数据源（CSV、Json、Parquet、ORC、JDBC 和 Hive）和外部数据源（如 Avro、Xml 和 Pandas）的 DataFrame API 进行分析。Spark-on-HBase 连接器部分解释了如何使用 DataFrame 分析 Spark 中的 HBase 数据。该章还讲解了如何使用 Spark SQL 作为分布式 SQL 引擎。

第 5 章讲解了实时分析的含义，以及 Spark Streaming 与 Storm、trident、Flink 和 Samza 等其他实时引擎的不同之处。其中描述了具有输入数据源和输出存储的 Spark Streaming 的架构，涵盖无状态和有状态的流处理，以及使用基于接收器的方法和直接方法，把 Kafka 作为数据源，把 HBase 作为存储。该章还讲解了应用程序在驱动进程（Driver）或执行进程（Executor）出现故障的情况下，有关 Spark 流的容错概念。结构化流（Structured Streaming）的概念会通过一个物联网（IOT）的用例来解释。

第 6 章用 Jupyter、Zeppelin 和 Hue 等工具介绍了基于 Web 的笔记本。该章还介绍了 Livy REST 服务器，它用于把 Spark 构建为服务，并在多个用户之间共享 Spark RDD。该章还介绍了 Apache NiFi，它可以利用 Spark 和 Hadoop 构建数据流。

第 7 章旨在更深入地讲解利用 Spark 和 Hadoop 来实现数据科学中用到的机器学习技术。该章介绍了 Spark 使用的机器学习算法，包括垃圾邮件的检测、实现和构建机器学习流水线（machine learning pipeline）的方法，还讲解了使用 H2O 和 Hivemall 实现机器学习的方法。

第 8 章详细介绍了协同过滤技术，并解释了如何使用 Spark 和 Mahout 构建实时推荐引擎。

第 9 章介绍了图处理、GraphX 与 Giraph 的区别，以及 GraphX 的各种图运算，如创建图、计数、过滤、度、三元组、修改、连接、属性变换、顶点 RDD 和边 RDD 运算等。它还通过一个航班分析用例讲解了 GraphX 算法，如三角计数和连通分量。该章还介绍了基于 DataFrame 的新 GraphFrames 组件，用来解释模式发现（motif finding）这样的一些概念。

第 10 章讲解了 R 语言和 SparkR 之间的差异，以及如何开始通过 shell 脚本在 local、standalone 和 Yarn 模式下使用 SparkR 。该章还介绍了如何把 SparkR 与 RStudio、DataFrame、机器学习算法，以及 Apache Zeppelin 配套使用。

学习本书所需的资源

为了方便入门，本书中的实践练习会在 Cloudera、Hortonworks、MapR 或预构建的

Spark for Hadoop 的虚拟机（VM）上演示。同样的练习也可以在更大的集群上运行。

在你的笔记本电脑上使用虚拟机的必要条件有：

- 内存：8 GB 及以上
- CPU：至少 2 个虚拟 CPU
- 必须为 Windows 或 Linux 操作系统安装最新版本的 VMWare player 或 Oracle VirtualBox
- Mac 上需要安装最新版本的 Oracle VirtualBox 或 VMWare Fusion
- 需要在 BIOS 里启用虚拟化
- 浏览器：推荐使用 Chrome 25+、IE 9+、Safari 6+ 或 Firefox 18+（HDP Sandbox 无法在 IE 10 上运行）
- Putty
- WinScP

在本书的各章中会使用 Python 和 Scala 编程语言，其中 Python 的侧重程度更高。我们假设读者具备 Java、Scala、Python、SQL 或 R 语言的初级编程背景，并具有基本的 Linux 经验。如果读者在 Hadoop 平台上的大数据环境中有一些工作经验，就能比较快捷地开始构建 Spark 应用程序。

本书的目标读者

虽然本书主要是针对数据分析师和数据科学家编写的，但它也会对架构师、程序员和大数据从业者有所帮助。

对于数据分析师：本书可以用作数据分析人员在 Spark 和 Hadoop 上开发分析应用程序的参考指南。

对于数据科学家：本书可以用作在 Spark 和 Hadoop 上构建数据产品的参考指南。

对于架构师：本书提供了一个完整生态系统的概述和一些大数据分析应用程序的示例，有助于构建大数据分析的解决方案。

对于程序员：本书讲解了用 Scala 和 Python 语言构建大数据应用程序所使用的 API 和技术。

对于大数据从业者：本书有助于了解新的范例和技术，并做出正确的决定。

下载示例代码

你可以从 http://www.packtpub.com 的账户下载此书的示例代码文件。如果你是通过其

他渠道购买了此书，可以访问 http://www.packtpub.com/support 并注册，以便将文件直接发送给你。

你可以通过以下步骤下载代码文件：

（1）使用你的电子邮件地址和密码登录或注册 Packt 网站。

（2）将鼠标指针悬停在网页顶部的 SUPPORT 选项卡上。

（3）点击 Code Downloads & Errata。

（4）在 Search 输入框里输入本书的书名。

（5）选择你要下载代码文件的图书。

（6）从你购买此书的下拉菜单中选择要下载的代码。

（7）点击 Code Download。

你也可以通过点击 Packt 出版社网站上该图书对应网页上的 Code Files 按钮来下载代码文件。这个页面可以通过在搜索框中输入图书的名称来访问。请注意，你需要登录到你的 Packt 账户。

下载文件后，请确保你使用以下软件的最新版本来解压缩或提取文件夹：

- WinRAR / 7-Zip for Windows
- Zipeg / iZip / UnRarX for Mac
- 7-Zip / PeaZip for Linux

该书配套的代码也托管在 GitHub 上，网址为 https://github.com/PacktPublishing/big-data-analytics。在 https://github.com/PacktPublishing/ 上还有其他代码库，里面有丰富的书籍和视频分类。去看一下吧！[⊖]

下载本书的彩色图像

本书还提供了一个 PDF 文件，其中包含本书中使用的截图 / 图表的彩色图像。这些彩色图像会帮助你更好地了解输出的变化。你可以从 http://www.packtpub.com/sites/default/files/downloads/BigDataAnalyticsWithSparkAndHadoop_ColorImages.pdf 下载此文件。[⊜]

⊖ 推荐直接访问 GitHub 上的代码，无需登录账号，比较简单直接。也可以注册一个账号，这样遇到问题还可以在那里的 issues 里直接向作者提出。——译者注

⊜ 这个文件很有用，强烈推荐下载，因为书是黑白的，里面有的标记可能看不太清楚。——译者注

Contents 目　录

第 1 章 Chapter 1

从宏观视角看大数据分析

本书的目标是让你熟悉 Apache Spark 用到的工具和技术，重点介绍 Hadoop 平台上使用的 Hadoop 部署和工具。大多数 Spark 的生产环境会采用 Hadoop 集群，用户在集成 Spark 和 Hadoop 配套的各种工具时会遇到很多挑战。本书将讲解 **Hadoop 分布式文件系统**（Hadoop Distributed File System，HDFS）和**另一种资源协商器**（Yet Another Resource Negotiator，YARN）面临的集成挑战，以及 Spark 和 Hadoop 使用的各种工具。本书还会讨论所有 Spark 组件——Spark Core、Spark SQL、DataFrame、Dataset、Spark Streaming、Structured Streaming、MLlib、GraphX 和 SparkR，以及它与分析组件（如 Jupyter、Zeppelin、Hive、HBase）及数据流工具（例如 NiFi）的集成。此外，本书还会通过使用 MLlib 的一个实时推荐系统示例来帮助我们理解数据科学技术。

在本章，我们会从比较宏观的角度来介绍大数据分析，并尝试了解在 Apache Hadoop 和 Apache Spark 平台上使用的工具和技术。

大数据分析是分析大数据的过程，它可以提取过去、当前和未来的统计数据，以及用于改进业务决策的内在规律性。

大数据分析大致可分为两大类：数据分析和数据科学，它们是相互关联的学科。本章会解释数据分析与数据科学之间的差异。数据分析和数据科学在当前行业里的定义会随着它们的应用案例的不同而不同，但让我们尝试理解它们分别能够完成什么工作。

数据分析侧重于数据的收集和解释，通常侧重于过去和现在的统计。而另一方面，数据科学通过进行探索性分析，可以根据过去和现在的数据所识别的模型来产生推荐，重点关注于未来。

图 1-1 解释了数据分析和数据科学在时间和实现的价值方面的差异。图中还显示了它们解决的典型问题和使用的工具及技术。数据分析主要有两种类型的分析：描述性分析和诊断性分析。数据科学也有两种类型的分析：预测性分析和规范性分析。数据科学和数据分析的具体情况如图 1-1 所示。

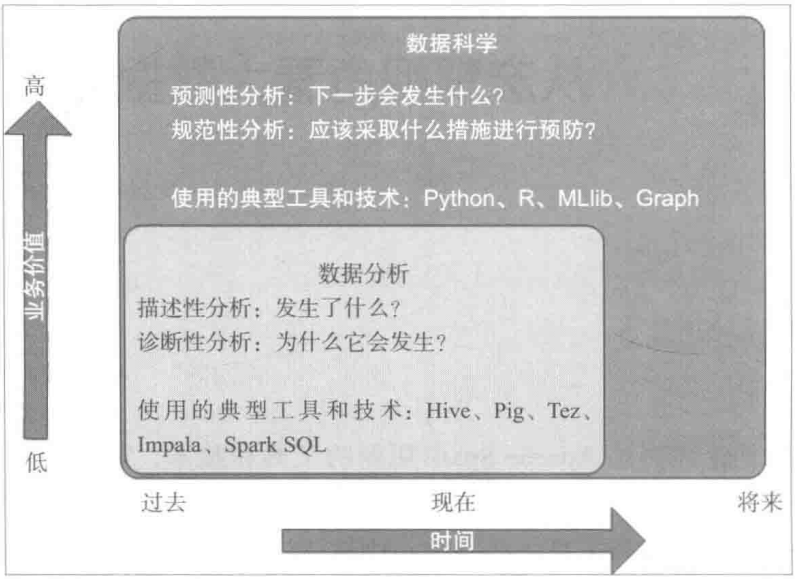


图 1-1 数据分析与数据科学

两者之间在过程、工具、技术、技能和输出方面的差异见下表：

	数据分析	数据科学
角度	向后看	向前看
工作性质	报表和优化	探索、发现、调查和视觉化
输出	报表和仪表盘	数据产品
采用的典型工具	Hive、Impala、Spark SQL 和 HBase	MLlib 和 Mahout
采用的典型技术	ETL 和探索性分析	预测分析和情感分析
所需的典型技能	数据工程、SQL 和编程	统计学、机器学习和编程

本章要讨论的主题如下：

- 大数据分析以及 Hadoop 和 Spark 在其中承担的角色
- 大数据科学以及 Hadoop 和 Spark 在其中承担的角色
- 相关的工具和技术
- 真实环境下的用例

1.1 大数据分析以及 Hadoop 和 Spark 在其中承担的角色

传统的数据分析使用**关系型数据库管理系统**（Relational Database Management System，RDBMS）的数据库来创建数据仓库和数据集市，以便使用商业智能工具进行分析。RDBMS 数据库采用的是**写时模式**（Schema-on-Write）的方法，而这种方法有许多缺点。

传统数据仓库的设计思想是用于**提取、转换和加载**（Extract, Transform, and Load，ETL）数据，据此回答与用户需求直接相关的一组预先定义的问题。这些预先定义的问题是利用 SQL 查询来回答的。一旦数据以易于访问的（consumable）格式进行转换和加载，用户就可以通过各种工具和应用程序访问它，从而生成报告和仪表板。但是，以易于访问的格式创建数据需要几个步骤，如下所示：

（1）确定预先定义的问题。

（2）从数据源系统识别和收集数据。

（3）创建 ETL 流水线，把数据以易于访问的格式加载到分析型数据库里。

如果有了新的问题，系统就需要识别和添加新的数据源并创建新的 ETL 流水线。这涉及数据库中的模式更改，实施工作通常会持续 1 ~ 6 个月。这是一个很重大的约束，迫使数据分析人员只能在预定义的范围内进行操作。

将数据转换为易于访问的格式通常会导致丢失原始 / 原子数据，而这些数据可能含有我们正在寻找的答案的结论或线索。

处理结构化和非结构化数据是传统数据仓库系统中的另一个挑战。有效地存储和处理大型二进制图像或视频也总是有挑战性的。

大数据分析是不使用关系数据库的；相反，它通常借助 Hive 和 HBase 在 Hadoop 平台上使用**读取模式**（Schema-on-Read，SOR）方法 。这种方法有许多优点。图 1-2 比较了 Schema-on-Write 和 Schema-on-Read 的场景。

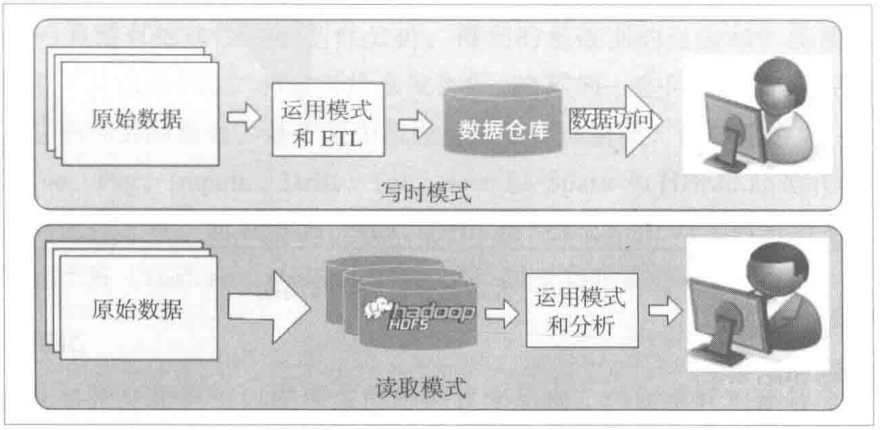


图 1-2 写时模式和读取模式的对比