金融大数据处理技术

2020-2021 秋季学期

Review

教学目标

- 深入理解大数据处理技术的基本概念、并行计算技术思想、并行计算系统基本架构。
- 学习Hadoop、Spark等大数据处理系统的基本组成和工作原理。
- 学习MapReduce和Spark并行程序设计和基础算法。
- 通过课程实验，熟悉Hadoop、 HBase 、Spark等大数据处理系统的安装和操作管理。
- 通过课程实践，将大数据处理技术应用到实际应用中。

教学目标

- 更深入地掌握大数据处理的基本原理
- 更广泛地了解大数据领域的新兴技术
- 更自信地面对金融科技领域的技术需求

课程性质

- 不是又一门语言/编程课
  - 虽然可能需要自学Java、Python、Scala等语言
- 不是又一门数据挖掘课程
  - 但会讲授和学习使用一些重要的算法和相关工具
- 不是又一门分布并行计算系统课
  - 但要求会操作典型的分布并行计算系统

课程大纲

- Ch.1 大数据处理技术简介
- Ch.2 并行计算和MPI基础编程
- Ch.3 Google MapReduce的基本架构
- Ch.4 Hadoop MapReduce的基本架构
- Ch.5 MapReduce基础编程（I）
- Ch.6 MapReduce基础编程（II）
- Ch.7 MapReduce高级编程
- Ch.8 基于MapReduce的图算法
- Ch.9 MapReduce数据挖掘基础算法（I）
- Ch.10 MapReduce数据挖掘基础算法（II）
- Ch.11 MapReduce数据挖掘基础算法（III）

课程大纲

- Ch.12 HBase基础原理与程序设计
- Ch.13 Hive简介
- Ch.14 Spark简介
- Ch.15 Spark基础编程
- Ch.16 Spark高级编程（I）
- Ch.17 Spark高级编程（II）
- Ch.18 Spark高级编程（III）
- Ch.19 Spark高级数据分析案例
- Ch.20 云计算简介

实验

- Ex.1 MPI编程
- Ex.2 Hadoop安装与运行
- Ex.3 HBase安装与运行
- Ex.4 天猫复购预测（MapReduce和Spark编程）
  - MapReduce基础编程
  - Spark基础编程
  - Hive或Spark QL操作
  - 数据挖掘应用

课程内容

- Ch.1 大数据简介
  - 大数据背景
    - Scale up vs. Scale out
  - 什么是数据？什么是大数据？
  - 大数据的5V特征：Volume，Variety，Velocity，Veracity，Value
  - 大数据的类型
    - 结构特征；获取和处理方式；关联特征
  - 大数据涉及的关键技术
    - 存储，实时处理，高速传输，搜索，数据分析等
    - 新平台，新服务，新传输方案

课程内容

- Ch.2 并行计算和MPI基础编程
  - 提高计算机硬件性能的主要手段
  - 为什么需要并行计算？
  - 并行计算的分类
    - 按数据和指令处理结构；按并行类型；按存储访问构架；按系统类型；按计算特征；按并行程序设计模型/方法
  - MPI并行程序设计的特点
  - MPI通信机制
    - 点对点通信
    - 节点集合通信
    - 用户自定义的复合数据类型传输
  - MPI的不足

课程内容

- Ch.3 Google MapReduce的基本架构
  - MapReduce基本模型和处理思想
    - 抽象模型Map和Reduce
      - map: (k1; v1) → [(k2; v2)]
      - reduce: (k2; [v2]) → [(k3; v3)]
    - MapReduce提供一个统一的计算框架
      - 计算任务的划分和调度
      - 数据的分布存储和划分
      - 处理数据与计算任务的同步
      - 结果数据的收集整理
      - 系统通信、负载平衡、计算性能优化处理
      - 处理系统节点出错检测和失效恢复
    - MapReduce主要设计思想和特征
      - Scale out，not scale up；失效被认为是常态；把处理向数据迁移；大数据集批处理的并行计算；隐藏系统层细节；平滑无缝的可扩展性

课程内容

- Ch.4 Google MapReduce的基本架构
  - Google MapReduce
    - 基本工作原理
    - 失效处理，带宽优化，计算优化
  - GFS
    - 基本设计原则
    - 基本工作原理
  - BigTable基本工作原理
    - 设计目标
    - Data Model
    - 基本构架

课程内容

- Ch.4 Hadoop MapReduce的基本架构
  - Hadoop生态系统
  - HDFS
    - 基本特征
    - 基本构架
    - 数据分布设计及设计要点
  - Hadoop MapReduce
    - 基本构架
    - 主要组件
    - MapReduce v1.0 vs. YARN (v2.0)
    - 容错及优化
  - HBase
    - 逻辑模型
    - 物理存储

课程内容

- Ch.5/6 MapReduce基础编程
  - MapReduce流水线
  - WordCount
  - 矩阵乘法
  - 关系代数运算
  - 排序算法
  - 单词同现
  - 倒排索引
  - 专利文献数据分析

课程内容

- Ch.7 MapReduce高级编程
  - 复合键值对的使用
  - 用户自定义数据类型
  - 用户自定义输入输出格式
  - 用户自定义Partitioner和Combiner
  - 迭代完成MapReduce计算
  - 链式MapReduce任务
  - 多数据源的连接
  - 全局参数/数据文件的传递
  - 其它处理技术

课程内容

- Ch.8 基于MapReduce的图算法
  - 图的表示
  - PageRank的基本设计思想和设计原则
  - 单源最短路径的并行广度优先算法

课程内容

- Ch.9/10/11 MapReduce数据挖掘基础算法
  - K-Means聚类算法
  - KNN最邻近分类算法
  - 朴素贝叶斯分类算法
  - 决策树分类算法
  - 支持向量机分类算法
  - 频繁项集挖掘算法

课程内容

- Ch.12 HBase基础原理与程序设计
  - CAP定理
  - ACID vs. BASE
  - RDBMS vs. NoSQL
  - HBase设计目标和功能特点
  - 数据存储管理方法
  - 基本操作和编程方法

课程内容

- Ch.13 Hive简介
  - RDBMS vs. Hive
  - HBase vs. Hive
  - Hive的体系结构
  - Hive的数据模型
  - Hive QL
    - DDL，DML，QUERY

课程内容

- Ch.14 Spark简介
  - Spark特点
  - Spark vs. Hadoop
  - Spark生态圈
  - Spark的基本构架和组件
  - Spark的技术特点

课程内容

- Ch.15 Spark基础编程
  - Spark安装与运行
  - Spark编程模型
    - RDD的操作
    - RDD的容错
    - RDD的依赖
    - RDD的持久化
  - Spark编程实例
    - WordCount
    - K-Means

课程内容

- Ch.16/17/18 Spark高级编程
  - 键值对操作
  - 共享变量
  - Spark SQL
  - Spark Streaming
  - Spark MLlib/ML
  - GraphX

课程内容

- Ch.19 Spark高级数据分析案例
  - 音乐推荐
    - 协同过滤
  - 基于潜在语义分析算法分析维基百科
    - LSA
    - TF-IDF
    - SVD
  - 基于蒙特卡罗模拟的金融风险评估

课程内容

- Ch.20 云计算简介
  - 什么是云计算？云计算解决什么主要问题？
  - 云计算的主要特点
  - 云计算的分类
    - 按服务层面的分类：IaaS，PaaS，SaaS
    - 按系统类型的分类：公用云，私有云，社区云，混合云
  - 云计算的关键技术
  - 怎样才算是云计算系统？
  - 容器云
  - 云原生
  - 数据湖

教材与参考资料

- 《深入理解大数据——大数据处理与编程实践》，黄宜华，2016，机械工业出版社
- 《Spark快速大数据分析》，Holden Karau等，2015，人民邮电出版社
- 《Spark高级数据分析》，Sandy Ryza等，2018，人民邮电出版社
- 《数据算法 Hadoop/Spark大数据处理技巧》，Mahmoud Parsian，2016，中国电力出版社
- 《Hadoop金融大数据分析》，Rajiv Tiwari，2017，电子工业出版社
- 《云计算》，刘鹏，2010，电子工业出版社

考核方式

- 平时10%
- 实验30%（实验1+2+3:15%，实验4:15%）
- 期末笔试60%

考试题型

- 填空题（20分）：概念
- 简答题（20分）：概念与原理
- 论述题（60分）：分析与设计

谢谢！