

<<Mahout实战>>

图书基本信息

书名：<<Mahout实战>>

13位ISBN编号：9787115347220

10位ISBN编号：7115347220

出版时间：2014-3

出版时间：人民邮电出版社

作者：[美] Sean Owen,[美] Robin Anil,[美] Ted Dunning,[美] Ellen Friedman

译者：王 斌,韩冀中,万 吉

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Mahout实战>>

内容概要

<<Mahout实战>>

作者简介

Sean Owen

现为大数据公司Cloudera数据产品总监，Myrrix创始人，曾任Apache Mahout项目管理委员会委员、谷歌高级软件工程师，是Mobile Web和Taste框架（现属于Mahout项目）的主力开发者。

Owen拥有哈佛大学计算机专业学士学位。

Robin Anil

谷歌公司负责地图与广告方向的软件工程师，Apache Mahout项目管理委员会委员，为Mahout开发了贝叶斯分类器和频繁模式挖掘实现，曾经在雅虎公司任高级软件工程师。

Ted Dunning

MapR Technologies公司首席应用架构师，Apache Mahout和Zookeeper项目管理委员会成员，为Mahout聚类、分类、矩阵分解算法做出了贡献，曾任DeepDyve公司CTO及多家公司首席科学家。

Ellen Friedman

Apache Mahout项目代码提交者，生物化学博士学位，经验丰富的科技作家，作品涵盖计算机、分子生物学、医学和地球科学。

<<Mahout实战>>

书籍目录

第1章 初识Mahout	1
1.1 Mahout的故事	1
1.2 Mahout的机器学习主题	2
1.2.1 推荐引擎	2
1.2.2 聚类	3
1.2.3 分类	4
1.3 利用Mahout和Hadoop处理大规模数据	4
1.4 安装Mahout	6
1.4.1 Java和IDE	6
1.4.2 安装Maven	7
1.4.3 安装Mahout	7
1.4.4 安装Hadoop	8
1.5 小结	8
第一部分 推荐	
第2章 推荐系统	10
2.1 推荐的定义	10
2.2 运行第一个推荐引擎	11
2.2.1 创建输入	11
2.2.2 创建一个推荐程序	13
2.2.3 分析输出	14
2.3 评估一个推荐程序	14
2.3.1 训练数据与评分	15
2.3.2 运行RecommenderEvaluator	15
2.3.3 评估结果	16
2.4 评估查准率与查全率	17
2.4.1 运行RecommenderIRStats-Evaluator	17
2.4.2 查准率和查全率的问题	19
2.5 评估GroupLens数据集	19
2.5.1 提取推荐程序的输入	19
2.5.2 体验其他推荐程序	20
2.6 小结	20
第3章 推荐数据的表示	21
3.1 偏好数据的表示	21
3.1.1 Preference对象	21
3.1.2 PreferenceArray及其实现	22
3.1.3 改善聚合的性能	23
3.1.4 FastByIDMap和FastIDSet	23
3.2 内存级DataModel	24
3.2.1 GenericDataModel	24
3.2.2 基于文件的数据	25
3.2.3 可刷新组件	25
3.2.4 更新文件	26
3.2.5 基于数据库的数据	26
3.2.6 JDBC和MySQL	27
3.2.7 通过JNDI进行配置	27

<<Mahout实战>>

3.2.8	利用程序进行配置	28
3.3	无偏好值的处理	29
3.3.1	何时忽略值	29
3.3.2	无偏好值时的内存级表示	30
3.3.3	选择兼容的实现	31
3.4	小结	33
第4章	进行推荐	34
4.1	理解基于用户的推荐	34
4.1.1	推荐何时会出错	34
4.1.2	推荐何时是正确的	35
4.2	探索基于用户的推荐程序	36
4.2.1	算法	36
4.2.2	基于GenericUserBased-Recommender实现算法	36
4.2.3	尝试GroupLens数据集	37
4.2.4	探究用户邻域	38
4.2.5	固定大小的邻域	39
4.2.6	基于阈值的邻域	39
4.3	探索相似性度量	40
4.3.1	基于皮尔逊相关系数的相似度	40
4.3.2	皮尔逊相关系数存在的问题	42
4.3.3	引入权重	42
4.3.4	基于欧氏距离定义相似度	43
4.3.5	采用余弦相似性度量	43
4.3.6	采用斯皮尔曼相关系数基于相对排名定义相似度	44
4.3.7	忽略偏好值基于谷本系数计算相似度	45
4.3.8	基于对数似然比更好地计算相似度	46
4.3.9	推测偏好值	47
4.4	基于物品的推荐	47
4.4.1	算法	48
4.4.2	探究基于物品的推荐程序	49
4.5	Slope-one推荐算法	50
4.5.1	算法	50
4.5.2	Slope-one实践	51
4.5.3	DiffStorage和内存考虑	52
4.5.4	离线计算量的分配	53
4.6	最新以及试验性质的推荐算法	53
4.6.1	基于奇异值分解的推荐算法	53
4.6.2	基于线性插值物品的推荐算法	54
4.6.3	基于聚类的推荐算法	55
4.7	对比其他推荐算法	56
4.7.1	为Mahout引入基于内容的技术	56
4.7.2	深入理解基于内容的推荐算法	57
4.8	对比基于模型的推荐算法	57
4.9	小结	57
第5章	让推荐程序实用化	59
5.1	分析来自约会网站的样本数据	59
5.2	找到一个有效的推荐程序	61

<<Mahout实战>>

5.2.1	基于用户的推荐程序	61
5.2.2	基于物品的推荐程序	62
5.2.3	slope-one推荐程序	63
5.2.4	评估查准率和查全率	63
5.2.5	评估性能	64
5.3	引入特定域的信息	65
5.3.1	采用一个定制的物品相似性度量	65
5.3.2	基于内容进行推荐	66
5.3.3	利用IDRescorer修改推荐结果	66
5.3.4	在IDRescorer中引入性别	67
5.3.5	封装一个定制的推荐程序	69
5.4	为匿名用户做推荐	71
5.4.1	利用PlusAnonymousUser-DataModel处理临时用户	71
5.4.2	聚合匿名用户	73
5.5	创建一个支持Web访问的推荐程序	73
5.5.1	封装WAR文件	74
5.5.2	测试部署	74
5.6	更新和监控推荐程序	75
5.7	小结	76
第6章	分布式推荐	78
6.1	分析Wikipedia数据集	78
6.1.1	挑战规模	79
6.1.2	分布式计算的优缺点	80
6.2	设计一个基于物品的分布式推荐算法	81
6.2.1	构建共现矩阵	81
6.2.2	计算用户向量	82
6.2.3	生成推荐结果	82
6.2.4	解读结果	83
6.2.5	分布式实现	83
6.3	基于MapReduce实现分布式算法	83
6.3.1	MapReduce简介	84
6.3.2	向MapReduce转换：生成用户向量	84
6.3.3	向MapReduce转换：计算共现关系	85
6.3.4	向MapReduce转换：重新思考矩阵乘	87
6.3.5	向MapReduce转换：通过部分乘积计算矩阵乘	87
6.3.6	向MapReduce转换：形成推荐	90
6.4	在Hadoop上运行MapReduce	91
6.4.1	安装Hadoop	92
6.4.2	在Hadoop上执行推荐	92
6.4.3	配置mapper和reducer	94
6.5	伪分布式推荐程序	94
6.6	深入理解推荐	95
6.6.1	在云上运行程序	95
6.6.2	考虑推荐的非传统用法	97
6.7	小结	97
第二部分	聚类	
第7章	聚类介绍	100

<<Mahout实战>>

7.1	聚类的基本概念	100
7.2	项目相似性度量	102
7.3	Hello World : 运行一个简单的聚类示例	103
7.3.1	生成输入数据	103
7.3.2	使用Mahout聚类	104
7.3.3	分析输出结果	107
7.4	探究距离测度	108
7.4.1	欧氏距离测度	108
7.4.2	平方欧氏距离测度	108
7.4.3	曼哈顿距离测度	108
7.4.4	余弦距离测度	109
7.4.5	谷本距离测度	110
7.4.6	加权距离测度	110
7.5	在简单示例上使用各种距离测度	111
7.6	小结	111
第8章	聚类数据的表示	112
8.1	向量可视化	113
8.1.1	将数据转换为向量	113
8.1.2	准备Mahout所用的向量	115
8.2	将文本文档表示为向量	116
8.2.1	使用TF-IDF改进加权	117
8.2.2	通过n-gram搭配词考察单词的依赖性	118
8.3	从文档中生成向量	119
8.4	基于归一化改善向量的质量	123
8.5	小结	124
第9章	Mahout中的聚类算法	125
9.1	k-means聚类	125
9.1.1	关于k-means你需要了解的	126
9.1.2	运行k-means聚类	127
9.1.3	通过canopy聚类寻找最佳k值	134
9.1.4	案例学习：使用k-means对新闻聚类	138
9.2	超越k-means: 聚类技术概览	141
9.2.1	不同类型的聚类问题	141
9.2.2	不同的聚类方法	143
9.3	模糊k-means聚类	145
9.3.1	运行模糊k-means聚类	145
9.3.2	多模糊会过度吗	147
9.3.3	案例学习：用模糊k-means对新闻进行聚类	148
9.4	基于模型的聚类	149
9.4.1	k-means的不足	149
9.4.2	狄利克雷聚类	150
9.4.3	基于模型的聚类示例	151
9.5	用LDA进行话题建模	154
9.5.1	理解LDA	155
9.5.2	对比TF-IDF与LDA	156
9.5.3	LDA参数调优	156
9.5.4	案例学习：寻找新闻文档中的话题	156

<<Mahout实战>>

9.5.5	话题模型的应用	158
9.6	小结	158
第10章	评估并改善聚类质量	160
10.1	检查聚类输出	160
10.2	分析聚类输出	162
10.2.1	距离测度与特征选择	163
10.2.2	簇间与簇内距离	163
10.2.3	簇的混合与重叠	166
10.3	改善聚类质量	166
10.3.1	改进文档向量生成过程	166
10.3.2	编写自定义距离测度	169
10.4	小结	171
第11章	将聚类用于生产环境	172
11.1	Hadoop下运行聚类算法的快速入门	172
11.1.1	在本地Hadoop集群上运行聚类算法	173
11.1.2	定制Hadoop配置	174
11.2	聚类性能调优	176
11.2.1	在计算密集型操作中避免性能缺陷	176
11.2.2	在I/O密集型操作中避免性能缺陷	178
11.3	批聚类及在线聚类	178
11.3.1	案例分析：在线新闻聚类	179
11.3.2	案例分析：对维基百科文章聚类	180
11.4	小结	181
第12章	聚类的实际应用	182
12.1	发现Twitter上的相似用户	182
12.1.1	数据预处理及特征加权	183
12.1.2	避免特征选择中的常见陷阱	184
12.2	为Last.fm上的艺术家推荐标签	187
12.2.1	利用共现信息进行标签推荐	187
12.2.2	构建Last.fm艺术家词典	188
12.2.3	将Last.fm标签转换成以艺术家为特征的向量	190
12.2.4	在Last.fm数据上运行k-means算法	191
12.3	分析Stack Overflow数据集	193
12.3.1	解析Stack Overflow数据集	193
12.3.2	在Stack Overflow中发现聚类问题	193
12.4	小结	194
第三部分	分类	
第13章	分类	198
13.1	为什么用Mahout做分类	198
13.2	分类系统基础	199
13.2.1	分类、推荐和聚类的区别	201
13.2.2	分类的应用	201
13.3	分类的工作原理	202
13.3.1	模型	203
13.3.2	训练、测试与生产	203
13.3.3	预测变量与目标变量	204
13.3.4	记录、字段和值	205

<<Mahout实战>>

- 13.3.5 预测变量值的4种类型 205
- 13.3.6 有监督学习与无监督学习 207
- 13.4 典型分类项目的工作流 207
 - 13.4.1 第一阶段工作流：训练分类模型 208
 - 13.4.2 第二阶段工作流：评估分类模型 212
 - 13.4.3 第三阶段工作流：在生产中使用模型 212
- 13.5 循序渐进的简单分类示例 213
 - 13.5.1 数据和挑战 213
 - 13.5.2 训练一个模型来寻找颜色填充：初步设想 214
 - 13.5.3 选择一个学习算法来训练模型 215
 - 13.5.4 改进填充颜色分类器的性能 217
- 13.6 小结 221
- 第14章 训练分类器 222
 - 14.1 提取特征以构建分类器 222
 - 14.2 原始数据的预处理 224
 - 14.2.1 原始数据的转换 224
 - 14.2.2 一个计算营销的例子 225
 - 14.3 将可分类数据转换为向量 226
 - 14.3.1 用向量表示数据 226
 - 14.3.2 用Mahout API做特征散列 228
 - 14.4 用SGD对20 Newsgroups数据集进行分类 231
 - 14.4.1 开始：数据集预览 231
 - 14.4.2 20 Newsgroups数据特征的解析和词条化 234
 - 14.4.3 20 Newsgroups数据的训练代码 234
 - 14.5 选择训练分类器的算法 238
 - 14.5.1 非并行但仍很强大的算法：SGD和SVM 239
 - 14.5.2 朴素分类器的力量：朴素贝叶斯及补充朴素贝叶斯 239
 - 14.5.3 精密结构的力量：随机森林算法 240
 - 14.6 用朴素贝叶斯对20 Newsgroups数据分类 241
 - 14.6.1 开始：为朴素贝叶斯提取数据 241
 - 14.6.2 训练朴素贝叶斯分类器 242
 - 14.6.3 测试朴素贝叶斯模型 242
 - 14.7 小结 244
- 第15章 分类器评估及调优 245
 - 15.1 Mahout中的分类器评估 245
 - 15.1.1 获取即时反馈 246
 - 15.1.2 确定分类“好”的含义 246
 - 15.1.3 认识不同的错误代价 247
 - 15.2 分类器评估API 247
 - 15.2.1 计算AUC 248
 - 15.2.2 计算混淆矩阵和熵矩阵 250
 - 15.2.3 计算平均对数似然 252
 - 15.2.4 模型剖析 253
 - 15.2.5 20 Newsgroups语料上SGD分类器的性能指标计算 254
 - 15.3 分类器性能下降时的处理 257
 - 15.3.1 目标泄漏 258
 - 15.3.2 特征提取崩溃 260

<<Mahout实战>>

15.4	分类器性能调优	262
15.4.1	问题调整	262
15.4.2	分类器调优	265
15.5	小结	267
第16章	分类器部署	268
16.1	巨型分类系统的部署过程	268
16.1.1	理解问题	269
16.1.2	根据需要优化特征提取过程	269
16.1.3	根据需要优化向量编码	269
16.1.4	部署可扩展的分类器服务	270
16.2	确定规模和速度需求	270
16.2.1	多大才算大	270
16.2.2	在规模和速度之间折中	272
16.3	对大型系统构建训练流水线	273
16.3.1	获取并保留大规模数据	274
16.3.2	非规范化及下采样	275
16.3.3	训练中的陷阱	276
16.3.4	快速读取数据并对其进行编码	278
16.4	集成Mahout分类器	282
16.4.1	提前计划：集成中的关键问题	283
16.4.2	模型序列化	287
16.5	案例：一个基于Thrift的分类服务器	288
16.5.1	运行分类服务器	292
16.5.2	访问分类器服务	294
16.6	小结	296
第17章	案例分析——Shop It To Me	297
17.1	Shop It To Me选择Mahout的原因	297
17.1.1	Shop It To Me公司简介	298
17.1.2	Shop It To Me需要分类系统的原因	298
17.1.3	对Mahout向外扩展	298
17.2	邮件交易系统的一般结构	299
17.3	训练模型	301
17.3.1	定义分类项目的目标	301
17.3.2	按时间划分	303
17.3.3	避免目标泄漏	303
17.3.4	调整学习算法	303
17.3.5	特征向量编码	304
17.4	加速分类过程	306
17.4.1	特征向量的线性组合	307
17.4.2	模型得分的线性扩展	308
17.5	小结	310
附录A	JVM调优	311
附录B	Mahout数学基础	313
附录C	相关资源	318
索引		320

<<Mahout实战>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>