HZ BOOKS
华章教育

智能科学与技术丛书

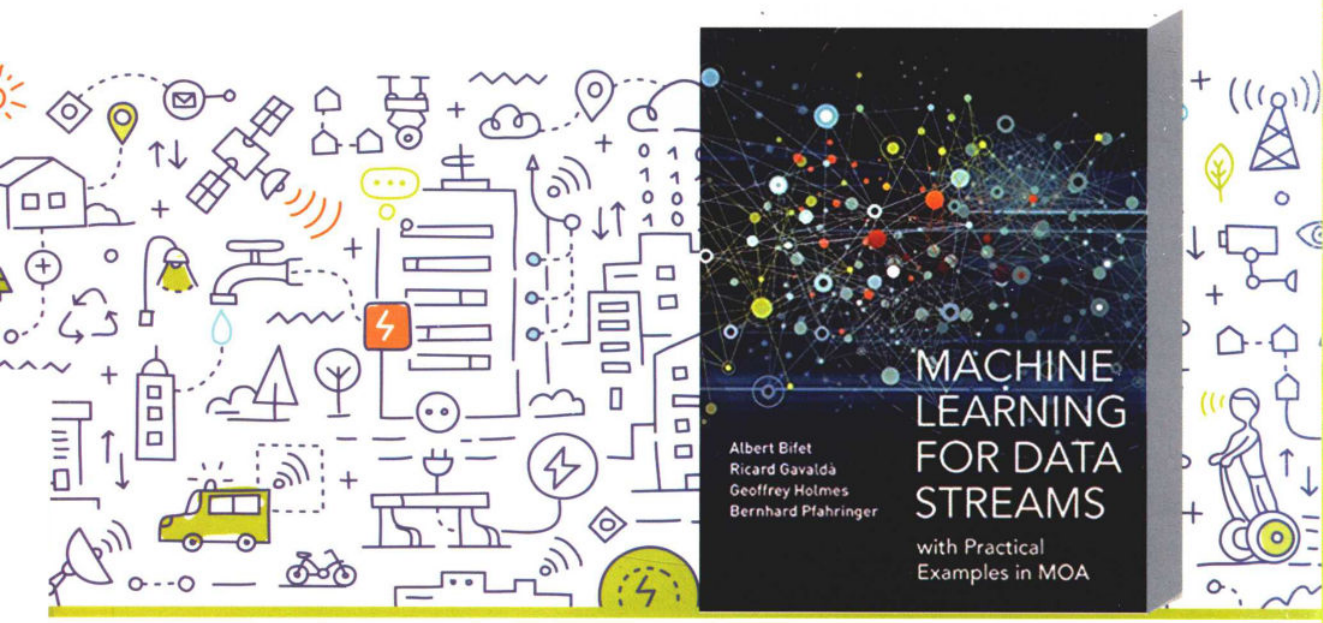
Machine Learning for Data Streams with Practical Examples in MOA

数据流机器学习

MOA 实例

[法] 阿尔伯特·比费特（Albert Bifet）
[西班牙] 理查德·戈华达（Richard Gavaldà）
[新西兰] 杰弗里·福尔摩斯（Geoffrey Holmes）
[新西兰] 伯恩哈德·普法林格（Bernhard Pfahringer）
◎ 著

陈瑶 姚毓夏 ◎ 译



机械工业出版社
China Machine Press

智能科学与技术丛书

Machine Learning for Data Streams with Practical Examples in MOA

数据流机器学习

MOA实例

[法] 阿尔伯特·比费特（Albert Bifet）
[西班牙] 理查德·戈华达（Richard Gavaldà）
[新西兰] 杰弗里·福尔摩斯（Geoffrey Holmes）
[新西兰] 伯恩哈德·普法林格（Bernhard Pfahringer）
◎ 著

陈瑶 姚毓夏 ◎ 译

机械工业出版社
China Machine Press

图书在版编目（CIP）数据

数据流机器学习：MOA 实例 /（法）阿尔伯特·比费特（Albert Bifet）等著；陈瑶，姚毓夏译 . —北京：机械工业出版社，2020.1
（智能科学与技术丛书）
书名原文：Machine Learning for Data Streams: with Practical Examples in MOA

ISBN 978-7-111-64139-1

I. 数… II. ① 阿… ② 陈… ③ 姚… III. ① 数据处理 ② 机器学习 IV. ① TP274 ② TP181

中国版本图书馆 CIP 数据核字（2019）第 254430 号

本书版权登记号：图字 01-2018-8096

本书首先简要介绍了机器学习的主题，包括大数据挖掘、数据流挖掘的基本方法，以及一个简单的 MOA 示例。接下来针对 sketch 技术、分类、集成方法、回归、聚类和频繁模式挖掘进行了更详细的讨论。本书最后讨论了 MOA 软件，涵盖 MOA 图形用户界面、命令行、MOA API 的使用以及 MOA 中新方法的开发。对于那些想要使用数据流挖掘的读者、数据流挖掘的研究人员，以及想要为 MOA 创建新算法的程序员来说，本书将是一个重要的参考指南。

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：梁华杰　　责任校对：殷　虹
印　　刷：北京市荣盛彩色印刷有限公司　　版　　次：2020 年 1 月第 1 版第 1 次印刷
开　　本：185mm × 260mm　1/16　　印　　张：13.25
书　　号：ISBN 978-7-111-64139-1　　定　　价：79.00 元

客服电话：（010）88361066　88379833　68326294　　投稿热线：（010）88379604
华章网站：www.hzbook.com　　读者信箱：hzjsj@hzbook.com

机器学习和数据挖掘早已是计算机领域中的热门话题，这两个领域中的优秀作品也屡见不鲜。本书的独特之处在于从数据流的角度详细介绍了机器学习模型，同时结合数据流生成和分析的开源软件 MOA，为数据流挖掘从业者和学者提供了易于上手实践的理论和工具。本书的每一章不仅分析了基础数学背景和传统机器学习的算法或模型，同时为了加深读者的横向理解，还专门为此对比了相关的数据流模型。除了引用最前沿的文献和广泛使用的模型，本书的可贵之处在于，呼吁业界多加考虑使用那些尚未普及却效果不错的新算法；对于新算法和常用算法，本书对模型进行了多方面的比较；正如前言中所说，本书面向有一定编程基础、机器学习基础或数据挖掘基础的读者。对于其他背景的读者，请参考前言中的阅读方法。

本书由陈瑶负责全书翻译内容审核，姚毓夏负责翻译本书大部分章节。

每一本书的背后都关联着大同小异的故事。翻译过程中，我们克服了跨国时差的合作挑战，且至今素未谋面。翻译开始之初，从联系出版社到最终交稿，经历了各种曲折，这也让我们更加坚定了初衷。在这里，我们想要号召愿意参与翻译工作但还在观望的译者，联系我们、加入我们，这里有计算机各个领域的专家和翻译小组。每一个贡献出自己个人时间的译者，除了能在翻译中学习所翻译书籍的内容之外，还可以找到自己想要深耕的领域，默默地在中国技术发展史上留下或深或浅的印记。

最后感谢北京华章图文信息有限公司的刘锋老师在翻译过程中给予的悉心帮助和指导！

译者

2019 年 9 月

前　言

Machine Learning for Data Streams: with Practical Examples in MOA

实时数据流分析正变得越来越常见，已经成为从现实中获取有用信息的标准手段。数据流分析可以让组织迅速应对问题并探测未来趋势，从而提升自身的运营效率。本书将充分展现数据流挖掘领域中常用的算法和技术，并且详细地介绍 MOA 软件：一个包含这些算法和技术的开源框架。

本书展示的数据流挖掘领域中的算法，面向以下 3 类特定的读者群体：

第一类读者希望在实际中应用数据流挖掘。他们有数据挖掘基础，但往往没有很强的算法或编程背景，比如管理、商务智能和市场营销领域的学生和教授。本书充分考虑了这些读者的需求，提供了 MOA 的上手教程，以任务为导向而不是以算法为导向。

第二类读者是那些在数据流挖掘领域的研究者和创新者。他们需要详细地了解算法和评估方法，才能更恰当地运用现有的算法，评估其性能，并把算法融入应用，甚至是创造新的算法。这群读者往往是计算机或数据科学专业的高年级本科生、研究生和博士，以及创新型开发者。

第三类读者想要在 MOA 中加入新的算法，从而对这个开源项目做出贡献。这些读者需要理解 MOA 的类结构和创建学习任务的方法。

为了充分满足这 3 类读者群体的需求，本书分成三个部分。第一部分简单地介绍大数据流挖掘，包含三章。前两章介绍大数据流及其基本挖掘方法。后一章是 MOA 上手指南，读者可以作为参考，自行探索 MOA。

本书的第二部分更详细地展现了数据流挖掘中的常见问题和重要算法。由于涉及知识面广阔，本书优先讲解 MOA 中已涵盖的算法。第 4 章提到了 sketch 技术，本书认为数据流挖掘领域人员很有必要对该技术加以了解。大部分章节含有一套练习题或 MOA 上手教程，或两者兼具。

第三部分全篇讲解 MOA，从用户界面开始，到命令行和 API，最后讲解如何实现新方法。

综上所述，第一类读者应该阅读第一部分，有时间可以把第 11 章和第 12 章作为延伸阅读，以了解 MOA 的软件体系和其他可选参数。

第二类读者也应该阅读第一部分，然后至少应该阅读 4.1 节至 4.3 节（如果对 sketch 数据结构有兴趣，应该反复阅读第 4 章）、第 5 章和第 6 章。可以根据兴趣，自

行选读第 7 章到第 10 章。如果需要在实际中运用 MOA，还应该继续阅读第 11 章至第 14 章。

第三类读者在阅读以上部分的基础上，还应该阅读第 15 章。

本书的网站地址是 https://mitpress.mit.edu/books/data-stream-mining，其中会持续更新书中的内容和一些辅助资源，包括讲稿、习题、软件上手教程和其他阅读材料。欢迎各位读者阅读参考并提出建议。

过去十年中，已经出现了好几本关于数据流挖掘的书。由 Garofalakis、Gehrke 和 Rastogi 编写的 *Data Stream Management—Processing Lligh-Speed Data Streams* [118]，还有 Aggarwal 编写的 *Data Streams—Models and Algorithms* [4]，这两本书涵盖的内容与本书部分相同，但是这些书往往从大数据库社区的角度出发，而不是数据流挖掘社区或者机器学习社区。

Gama 编写的 *Knowledge Discovery from Data Streams* [110] 一书从数据流挖掘及机器学习社区的角度讲解算法，但是并没有提到 MOA 的开发和评估框架。该书展现了算法的伪代码，其中有些算法已经在 MOA 中实现了。因此，读者可以考虑在阅读本书的同时参考 Gama 的书。

为了跟上数据流挖掘领域日新月异的变化，我们极力推荐以下会议的论文和报告：KDD（数据库知识发现）、ICDM（国际数据挖掘会议）、SAC（应用计算研讨会）。这些会议都设有数据流挖掘的分会场。此外，还有 ECML PKDD（机器学习及数据库知识发掘原理和实践欧洲会议）、SDM（数据挖掘 SIAM 会议），以及 IEEE DSAA（电气电子工程师协会的数据科学和高级分析会议）。

在杂志方面，至今为止还没有一本数据流挖掘的专门杂志，而刊登过数据流挖掘文章的杂志又数不胜数，在此就不一一列举了。

致谢

我们向所有为这本书和相关软件做出了贡献的人员表示感谢。我们希望罗列所有贡献者的名字，但难免由于姓名不详（如审阅者）、名单过长（比如 MOA 共同作者、通过提问和指出 bug 而改进 MOA 的学生，及直接贡献代码的人）以及我们的疏忽，而难以列全，希望得到读者的谅解。在此，我们事先向未被本书提及的数据流挖掘作者致歉。由于篇幅限制和选择不够明智而没能提及你的作品，希望得到理解。

我们要感谢麻省理工学院出版社（MIT Press），特别要感谢 Marie Lufkin Lee、Christine Bridget Savage 和 Kathleen Hensle 的协助。

在此特别告知，本书的灵感来自于具有奠基意义的 WEKA 机器学习工具。

本书共同作者 Ricard Gavaldà 的撰写部分由加泰罗尼亚政府（Generalitat de Catalunya）的 MACDA 项目（SGR2014-0890）和西班牙政府经济竞争部门（MINECO）的 APCOM 项目（TIN2014-57226）提供部分赞助。

译者序
前言

第一部分　概述

第 1 章　简介 …… 2
1.1　大数据 …… 2
1.1.1　工具：开源革命 …… 4
1.1.2　大数据带来的挑战 …… 4
1.2　实时分析 …… 6
1.2.1　数据流 …… 6
1.2.2　时间和内存 …… 6
1.2.3　应用一览 …… 6
1.3　关于本书 …… 7

第 2 章　大数据流挖掘 …… 8
2.1　算法 …… 8
2.2　分类算法 …… 9
2.2.1　如何在数据流中评估分类器 …10
2.2.2　多数类分类器 …… 11
2.2.3　无变化分类器 …… 11
2.2.4　惰性分类器 …… 11
2.2.5　朴素贝叶斯分类器 …… 12
2.2.6　决策树分类器 …… 12
2.2.7　集成分类器 …… 13
2.3　回归算法 …… 13
2.4　聚类算法 …… 14
2.5　频繁模式挖掘 …… 14

第 3 章　MOA 的实际操作介绍 …… 16
3.1　入门开始 …… 16
3.2　分类模型的图形用户界面 …… 18
3.3　用命令行操作 …… 23

第二部分　数据流挖掘

第 4 章　数据流和 Sketch 数据结构 …… 26
4.1　背景知识：近似算法 …… 27
4.2　集中不等式 …… 28
4.3　取样 …… 30
4.4　统计总数 …… 31
4.5　去重统计 …… 32
4.5.1　线性计数 …… 33
4.5.2　科恩对数计数器 …… 33
4.5.3　Flajolet-Martin 计数器和 HyperLogLog 算法 …… 34
4.5.4　应用：图论的计算距离函数 …… 36
4.5.5　讨论：对数与线性 …… 37
4.6　频率问题 …… 37
4.6.1　SpaceSaving sketch …… 38
4.6.2　CM-Sketch 算法 …… 40
4.6.3　CountSketch 算法 …… 42
4.6.4　时刻计算 …… 44
4.7　滑动窗口的指数矩形图 …… 45
4.8　分布式 sketch 计算的可合并性 …47
4.9　一些技术方面的讨论和其他资料 …… 48
4.9.1　哈希函数 …… 48
4.9.2　创建 (ε, δ) 近似算法 …… 49
4.9.3　其他 sketch 技术 …… 49
4.10　练习 …… 50

第 5 章　处理变化 …… 52
5.1　数据流中变化的定义 …… 52

5.2 评估器……56
5.2.1 滑动窗口和线性评估器……57
5.2.2 指数加权移动平均评估器……57
5.2.3 单维度卡尔曼滤波器……58
5.3 变化探测……58
5.3.1 评估变化探测……59
5.3.2 CUSUM 测试和 Page-Hinkley 测试……59
5.3.3 统计测试……60
5.3.4 漂移探测法……61
5.3.5 自适应滑动窗口算法……62
5.4 与其他 Sketch 和多维数据结合……64
5.5 练习……64

第 6 章 分类……66
6.1 分类器评估……67
6.1.1 误差估算……68
6.1.2 分布评估……69
6.1.3 性能的评估测量……70
6.1.4 统计显著性……72
6.1.5 测量挖掘成本……73
6.2 基线分类器……73
6.2.1 多数类……73
6.2.2 无变化分类器……74
6.2.3 朴素贝叶斯……74
6.2.4 多项式朴素贝叶斯……77
6.3 决策树……78
6.3.1 估算切分标准……79
6.3.2 Hoeffding 决策树……80
6.3.3 CVFDT……82
6.3.4 VFDTc 和 UFFT……83
6.3.5 Hoeffding 适应树……84
6.4 处理数字属性……85
6.4.1 VFML……85
6.4.2 穷举二叉树……86
6.4.3 Greenwald 和 Khanna 的分位数摘要……86
6.4.4 高斯近似……87
6.5 感知器模型……88
6.6 惰性学习……89
6.7 多标签分类器……89
6.8 主动学习……91
6.8.1 随机策略……92
6.8.2 固定不确定策略……93
6.8.3 可变不确定策略……93
6.8.4 随机不确定策略……94
6.9 概念演变……94
6.10 MOA 实战操作……95

第 7 章 集成方法……99
7.1 准确率加权集成……99
7.2 加权多数算法……100
7.3 堆叠算法……102
7.4 装袋算法……102
7.4.1 在线装袋算法……103
7.4.2 装袋算法如何应对数据流变化……103
7.4.3 杠杆装袋算法……103
7.5 提升算法……104
7.6 Hoeffding 树集成算法……105
7.6.1 Hoeffding 选项树算法……105
7.6.2 随机森林算法……105
7.6.3 有限的 Hoeffding 树的感知器堆叠……106
7.6.4 自适应大小的 Hoeffding 树算法……107
7.7 重复性概念……107
7.8 MOA 实战操作……108

第 8 章 回归……110
8.1 什么是回归……110
8.2 如何评估回归……111

8.3 感知器学习 ····· 112
8.4 惰性学习 ····· 112
8.5 决策树学习 ····· 112
8.6 决策规则 ····· 113
8.7 MOA 中的回归 ····· 114

第 9 章 聚类 ····· 115

9.1 聚类的评估方法 ····· 116
9.2 *k*-means 算法 ····· 117
9.3 BIRCH、BICO 和 CluStream ····· 118
9.4 基于密度的方法：DBSCAN 和 Den-Stream ····· 120
9.5 ClusTree ····· 121
9.6 StreamKM++：核心集 ····· 122
9.7 延伸阅读 ····· 123
9.8 MOA 实战操作 ····· 124

第 10 章 频繁模式挖掘 ····· 127

10.1 什么是模式挖掘 ····· 127
10.1.1 模式的定义和例子 ····· 127
10.1.2 频繁模式挖掘的批量算法 ····· 129
10.1.3 闭合模式和最大模式 ····· 131
10.2 数据流中频繁模式挖掘的方法 ····· 131
10.3 如何在数据流中进行频繁项集挖掘 ····· 134
10.3.1 简化为高频繁项 ····· 134
10.3.2 Moment 算法 ····· 135
10.3.3 频繁模式数据流算法 ····· 135
10.3.4 IncMine 算法 ····· 136
10.4 数据流的频繁子图挖掘 ····· 137
10.4.1 WinGraphMiner 框架 ····· 138
10.4.2 AdaGraphMiner 框架 ····· 139
10.5 延伸阅读 ····· 140
10.6 练习 ····· 141

第三部分 MOA 软件

第 11 章 MOA 及其软件体系 ····· 144

11.1 MOA 架构 ····· 145
11.2 安装 ····· 145
11.3 MOA 的近期发展 ····· 145
11.4 MOA 扩展包 ····· 146
11.5 ADAMS 优化 ····· 147
11.6 MEKA 优化 ····· 149
11.7 OpenML 环境 ····· 150
11.8 StreamDM 软件 ····· 150
11.9 Streams 工具 ····· 151
11.10 Apache SAMOA 流媒体 ML 库 ····· 151

第 12 章 图形用户界面 ····· 154

12.1 初识图形用户界面 ····· 154
12.2 分类和回归 ····· 154
12.2.1 主要任务一览 ····· 156
12.2.2 数据源和数据生成器 ····· 157
12.2.3 贝叶斯分类器一览 ····· 160
12.2.4 决策树一览 ····· 160
12.2.5 元分类器（集成）一览 ····· 161
12.2.6 函数分类器一览 ····· 162
12.2.7 漂移分类器一览 ····· 162
12.2.8 主动学习分类器 ····· 163
12.3 聚类 ····· 163
12.3.1 数据源和数据生成器 ····· 163
12.3.2 数据流聚类算法一览 ····· 163
12.3.3 如何进行可视化和数据分析 ····· 164

第 13 章 用命令行操作 ····· 166

13.1 给分类和回归创建学习任务 ····· 166
13.2 给分类和回归创建评估任务 ····· 167

13.3 给分类和回归创建学习与评估任务……167
13.4 两种分类器的对比……168

第 14 章 调用 API……170

14.1 MOA 对象……170
14.2 选项……170
14.3 示例：先序评估……173

第 15 章 在 MOA 中开发新的方法……175

15.1 MOA 中的主要类……175
15.2 创建新的分类器……176
15.3 编译分类器……183
15.4 MOA 中的良好编程方法……183

参考文献……185

第一部分

Machine Learning for Data Streams: with Practical Examples in MOA

概　述

第 1 章

Machine Learning for Data Streams: with Practical Examples in MOA

简　介

当今世界，每一天人们通过各种各样的电子终端制造海量的数据，这些数据有不同的形式，并且来自于一些独立的或关联的应用。我们现有的数据处理、分析、存储和理解能力，在这股大数据的洪流面前显得力不从心。社交网络应用诞生普及以来，用户可以随心所欲地发布内容，这更加速了数据的快速增长，让本已拥有海量数据的互联网变得更为庞大。

不仅如此，手机里的感应器正从我们身上实时读取各个方面的数据。一部手机可以处理的数据量远远不止通话记录这么简单，毕竟通话记录的发明只是为了方便结账。可以预见的是物联网（IoT）会把数据规模提升到一个前所未有的高度。到时候，任何人和任何机器（不论是家用咖啡机还是轿车和公共汽车，不论是在火车站还是在机场）都有着松散的联系。数以万亿计的相连物体无疑会产生巨大的信息海洋，而我们必须大海捞针，去发现有价值的信息，从而提升生活质量，让世界变得更好。例如，每天早上起床后，为了最优化通勤时间，信息处理系统需要综合处理交通、天气、建筑、警察管制和你的日程安排信息，并在有限的时间里进行深度优化。

为了处理多到让人难以置信的数据，我们需要快捷高效、合理利用资源的实时处理方法。

1.1　大数据

用一个具体的数据大小来定义“大数据”是没有意义的，哪怕用拍字节 (PB，相当于一千兆字节) 也不够。比较有意义的定义是大数据通常太大而难以用常规算法和技术来管理，尤其是当我们要从中提取知识的时候。

二十年前人们还在为吉字节（GB）量的数据挣扎，而写这本书的时候纠结的单位已经变成了表 1-1 中的太字节（TB）和拍字节（PB）。毫无疑问二十几年后，我们纠结

的数据单位会变成表格更下面的几行。

表 1-1 存储单位换算表（单位：字节）

存储单位	十进制	二进制
千字节（KB）	10^3	2^{10}
兆字节（MB）	10^6	2^{20}
吉字节（GB）	10^9	2^{30}
太字节（TB）	10^{12}	2^{40}
拍字节（PB）	10^{15}	2^{50}
艾字节（EB）	10^{18}	2^{60}
泽字节（ZB）	10^{21}	2^{70}
尧字节（YB）	10^{24}	2^{80}

2001 年，在 Gartner 工作的分析师 Doug Laney[154] 用 3 个 V 特性定义了大数据管理：

- 数据容量（volume）：数据量前所未有且持续增长，但是我们能处理的数据量相对而言并没有增加。
- 数据种类（variety）：数据种类繁多，有文字、传感器数据、音频、视频、图片等，我们要从所有这些数据中提取信息。
- 数据运动（velocity）：数据源源不断，我们想从中实时获取有用的信息。

而后其他 V 特性又被陆续添加进来：

- 数据可变性（variability）：数据结构或者说用户解释数据的方法，一直在变化。
- 数据价值（value）：数据有用之处仅仅在于其能导向更佳的决策并最终赢得优势。
- 数据可靠性（validity and veracity）：有些数据不完全可靠，必须要控制这些不确定性。

Gartner 公司 [200] 在 2012 年把大数据的定义总结为“体量庞大、高速变动和种类繁多的信息资产，需要采用经济型和创新型的信息处理方式，以增强信息洞察及决策的能力。”

大数据的应用应该让人们获得更好的服务、更佳的消费体验和更高的健康质量：

- 商业：个性化体验和客户流失检测。
- 科技：把处理时间的单位从小时级降低到秒级。
- 健康：挖掘医疗记录和基因数据，以监控病情、提升健康水平。
- 智慧城市：专注发展可持续经济和高质量生活，合理有效地利用自然资源。

举一个大数据挖掘的应用案例，我们来看一下 Global Pulse 是如何工作的 [236]。Global Pulse 是一个联合国的倡议组织，旨在利用大数据改善发展中国家人民的生活质

量。该组织由大数据创新实验室构成，其大数据挖掘策略如下：

1. 研究新的方法和技术来分析实时电子数据，尽早检测出潜在的漏洞。
2. 组装一个免费、开源的技术工具套件，来分析实时数据并分享研究假设。
3. 建立综合的全球 Pulse 实验室网络，从国家层面试行数据挖掘策略。

大数据挖掘的改革并不仅限于工业化国家，因为手机在发展国家也逐渐普及开来。全球超过 50 亿部的手机中，大约 80% 都源自发展中国家。

1.1.1　工具：开源革命

大数据的现象本质上和开源软件革命息息相关。大公司比如雅虎、推特、领英、谷歌和 Facebook 都从开源项目中受益，并且对其做出贡献，例如：

- Apache Hadoop[16] 是一个基于 MapReduce 编程模型和 Hadoop 分布式文件系统（HDFS）的平台，用于运行数据密集型的分布式应用。用户可以在 Hadoop 上快速开发应用，在计算机集群上并行处理海量数据。
- Apache Hadoop 的相关项目[260]：Apache Pig、Apache Hive、Apache HBase、Apache ZooKeeper、Apache Cassandra、Cascading、Scribe 和 Apache Mahout[17] 都是主要基于 Hadoop，具有拓展性的机器学习和数据挖掘开源软件。
- Apache Spark[253] 是一个运行在 Hadoop 架构上，专为大规模数据处理而设计的数据处理引擎。Spark 提供了大量的库，包括 SQL、DataFrames、MLlib for machine learning、GraphX 和 Spark Streaming。开发者可以在同一个应用中无缝组合使用这些库。
- Apache Flink[62] 是一个流式的数据流执行引擎，为数据流的分布式计算提供了数据分布、数据通信和容错机制。基于流执行引擎，Flink 提供了几个易于开发应用的 API。如果说 Apache Spark 的 Spark Streaming 是个可以用微批次数据来模拟流处理的批处理引擎，那么 Apache Flink 就是可以做到批处理的流处理引擎。
- Apache Storm[168] 是一个分布式数据流处理系统，同 Apache S4 及 Apache Samza 类似。
- TensorFlow[1] 是一个用机器学习和深度神经网络的开源包。

1.1.2　大数据带来的挑战

由于数据的本质：庞大、多样、变化[128]，大数据的管理和分析在未来仍有诸多挑

战。接下来几年，研究者和从业者需要处理的部分挑战如下：

- 架构分析。目前尚不清楚如何搭建最优化的架构分析系统，用于同时处理历史数据和实时数据。第一个架构是 Nathan Martz 提出的 Lambda 架构 [169]。Lambda 架构划成三层：批处理层、服务层和速度层，可以在任何数据上实时运行任意功能。它在同一个系统里整合了 Hadoop 和 Storm，分别用于批处理层和速度层的计算。一个更近期的方案是由领英的 Kreps 提出的 Kappa 架构 [152]。它简化了 Lambda 架构，删去了批量处理系统。
- 评估。有效的评估方法是得出重要的统计结论，并且避免概率的陷阱。如果“多重假设问题”没有处理好，很容易像 Efron 说的那样 [95]，一下子在大数据集和成千上万亟待解答的问题上出错。在进行数据评估时，更重要的是避免陷入纸上谈兵的误区，即只注重技术上的衡量标准，比如错误率和速度，而忽视了对现实的影响。Wagstaff 曾讨论过 [242]，想要驳倒那些觉得大数据徒有虚名的人，唯一的办法就是定期发布达到挑战性问题的合理标准的应用，就像他的论文里解释的那样。
- 分布式挖掘。许多数据挖掘技术都在分布式挖掘上也有一定用途。为了开发出这些技术的分布式版本，需要更多实验研究和理论分析。
- 数据变化。数据可能随时变化，因此大数据挖掘技术要注重灵活应变，有时还要能明确侦测到变化。正是这种需求促进了本书的许多数据流挖掘技术的开发 [110]。
- 数据压缩。存储空间的大小和大数据处理息息相关。节省空间有两个主要的途径：压缩，无损于信息；或者取样，选择具有代表性的数据。压缩耗时更多而需要的空间更少，相当于化时间为空间。取样虽然有损信息，但是可以节省数量级的空间。比如 Feldman 等人 [99] 就用核集（coreset）简化了大数据的问题。核集是数据集的一个小子集，能够可靠地估算原本的数据。
- 数据可视化。大数据分析还有一个主要问题就是如何可视化结果，其挑战在于要用易于理解的方法表现大量数据里的信息。就像 *The Human Face of Big Data* [228] 这本书所说，大数据可视化需要新的技术和框架来呈现故事。
- 隐藏的大数据。大部分原本有用的数据实际上都没发挥作用，因为它们没加标签、基于文件或者非结构化。2012 年 IDC 对大数据的研究解释道，如果能加上标签和分析，2012 年里有 23%（632 艾字节）的数字世界能够被用于大数据。然而那时只有 3% 可能有用的数据被加上了标签，被分析的数据就更少了。这个数字这几年可能还在下降。开放数据和语义网运动的出现让我们意识到了这个问题，并且改善了情况。

1.2　实时分析

一个大数据的著名例子是实时分析。对一个组织来说，重要的不仅是立即获得查询结果，更是根据刚刚产生的数据进行查询。

1.2.1　数据流

数据流是一个用于支持实时分析的抽象的算法概念。数据流是指一系列的数据项，可以是无限的。每一个数据项都有时间戳，所以也就有了时间顺序。数据项接踵而至，而我们想要建立并维护这些实时数据项的模型，比如模式或者预测者。处理数据流的时候，在算法方面有两个主要挑战：数据流数据庞大并且流动速度快，而我们需要从中实时提取信息。这意味着通常需要接受近似的解决方案，以便节省时间和内存；另一个挑战是数据会演变，所以我们建立的模型要能适应数据里的变化。

1.2.2　时间和内存

准确度、时间和内存是数据流挖掘的三个主要维度：我们希望得到用最少时间和最小总内存，获取最高准确度的方法。之后我们会展示，只要把时间和内存合并到单一成本测量，就完全有可能把评估降维到二维任务。另外要注意的还有，与传统的数据挖掘类似，因为高速数据流无法缓冲，所以处理单个数据项的时间和总时间是相关的。

1.2.3　应用一览

产生数据流的场景有很多，这里我们举几个例子：

- 传感器数据和物联网：每天越来越多的传感器用于工业中的过程监控和质量改善。城市也开始部署庞大的传感器网络，用于监控人流的移动，检查道路和桥梁的健康情况、市内交通和人口的重要常数（vital constant）等。
- 远程通信：远程通信公司有大量的手机通话记录。现在，手机通话和位置也变成了需要实时处理的大数据来源。
- 社交媒体：在社交网站比如 Facebook、推特、领英和 Instagram 上，用户持续产生互动和贡献的数据。随之产生了两个需要实时数据分析的问题：话题社群的发现和情感分析。
- 市场和电子商务：销售行业正在实时收集大量交易数据，分析其背后价值，并