BigDataBench：开源的大数据系统评测基准

詹剑锋[1] 高婉铃[1),2)] 王 磊[1] 李经伟[3)] 魏 凯[4)]
罗纯杰[1)] 韩 锐[1)] 田昕晖[1),2)] 姜春宇[4)]

[1)](中国科学院计算技术研究所 北京 100190)
[2)](中国科学院大学 北京 100190)
[3)](北京尖峰新锐信息科技研究院 北京 100081)
[4)](中国信息通信研究院 北京 100191)

摘 要 大数据系统的蓬勃发展催生了大数据基准测试的研究，如何公正地评价不同的大数据系统以及怎样根据需求选取合适的系统成为了热点问题.然而，应用领域的广泛性、数据类型的多样性和数据操作的复杂性使得大数据基准测试集的设计面临很大的挑战.现有的相关基准测试工作要么针对某一类特定的应用或软件栈，要么根据流行度主观地选择大数据负载，难以全面覆盖大数据的多样性和复杂性.针对现有工作的不足，文中讨论大数据评测基准需要满足的需求，并研制了一个跨系统、体系结构、数据管理 3 个领域的大数据基准测试开源程序集——BigDataBench.它覆盖 5 个典型的应用领域(搜索引擎、电子商务、社交网络、多媒体、生物信息学)，包含结构化、半结构化、非结构化的数据类型，涵盖离线分析、交互式分析、在线服务、NoSQL 这 4 种负载类型.目前包含 14 个真实数据集、3 种类型的数据生成工具以及 33 个负载的不同软件栈实现.BigDataBench 已广泛应用到学术界和工业界中，应用案例包括负载分析、体系结构设计、系统优化等.基于 BigDataBench，中国信息通信研究院联合中国科学院计算技术研究所、华为等国内外知名公司和科研机构共同制定了国内首个工业标准的大数据平台性能评测标准.

关键词 大数据；基准测试；工业标准；测试方法；数据生成；应用案例
中图法分类号 TP311 **DOI 号** 10.11897/SP.J.1016.2016.00196

BigDataBench: An Open-source Big Data Benchmark Suite

ZHAN Jian-Feng[1)] GAO Wan-Ling[1),2)] WANG Lei[1)] LI Jing-Wei[3)] WEI Kai[4)]
LUO Chun-Jie[1)] HAN Rui[1)] TIAN Xin-Hui[1),2)] JIANG Chun-Yu[4)]

[1)](*Institute of Computing Technology, Chinese Academy of Sciences, Beijing* 100190)
[2)](*University of Chinese Academy of Sciences, Beijing* 100190)
[3)](*Beijing Academy of Frontier Science & Technology, Beijing* 100081)
[4)](*Chinese Academy of Information and Communications, Beijing* 100191)

Abstract Booming big data sparks tremendous outpouring of interest in storing and processing these data, and consequently a variety of big data systems emerge, giving rise to great pressure on big data benchmarking. However, complexity and diversity of big data raise great challenges in big data benchmarking. Most of the related benchmark efforts either target at specific application domains and software stacks, or choose workloads subjectively according to so-called popularity, thus fail to cover the diversity and complexity of big data. In this paper, we discuss the requirements for big data benchmarking and present our open source big data benchmark suite—BigDataBench, which is a multi-discipline research and engineering effort, i.e. system, architecture, and data

收稿日期：2015-07-13；d 在线出版日期：2015-12-16.本课题得到国家自然科学基金重点项目(61432006)资助.**詹剑锋**，男，1976 年生，博士，研究员，博士生导师，中国计算机学会(CCF)高级会员，主要研究领域为操作系统、数据管理、基准测试、分布式系统.E-mail：zhanjianfeng@ict.ac.cn.**高婉铃**，女，1989 年生，博士研究生，主要研究方向为大数据基准测试、大数据分析.**王 磊**，男，1976 年生，博士研究生，高级工程师，主要研究方向为基准测试、分布式系统.**李经伟**，男，1990 年生，学士，主要研究方向为大数据基准测试.**魏 凯**，男，1981 年生，高级工程师，主要研究方向为大数据技术和标准.**罗纯杰**，男，1987 年生，工程师，主要研究方向为大数据基准测试、大数据学习.**韩 锐**，男，1985 年生，博士，主要研究方向为云计算、基准测试.**田昕晖**，男，1989 年生，博士研究生，中国计算机学会(CCF)会员，主要研究方向为大数据基准测试、分布式系统.**姜春宇**，男，1987 年生，工程师，主要研究方向为大数据基准测试.

management. BigDataBench adopts an iterative and incremental methodology, not only covering five representative application domains, but also containing diverse data models and workload types. Currently, it includes 14 real-world data sets, scalable data generation tools for 3 kinds of data types, and 33 workloads implemented using competitive technologies. BigDataBench has been used both in academia and industry, with typical use cases of workload characterization, architecture design and system optimization. Based on BigDataBench, Chinese Academy of Information and Communications releases China's first industry-standard big data benchmark suite together with ICT, CAS, Huawei and other well-known companies and research institutions.

Keywords　big data; benchmarks; industry standard; benchmarking methodology; data generation; use cases

1 引　言

数据的爆炸式增长掀起了大数据的研究热潮，越来越多的应用领域涉及到大数据的处理和存储，其所蕴藏的科学价值和商业价值逐渐体现. 为了挖掘大数据中隐藏的知识，各种大数据系统应运而生. 目前，一系列关于大数据处理和存储的开源项目被发布，如 Hadoop MapReduce①、Hive②、Cloudera Impala③、NoSQL 数据库④、Spark[1]、GraphLab[2]等. 如何客观地评价众多的大数据系统以及从中选择适合自身需求的系统成为学术界和工业界普遍关心的问题，大数据工业界和研究社区迫切需要一套公认的大数据评测基准.

Lord Kelvin⑤ 阐述了评测对于设计和优化的重要性. 在数据库评测基准还未发展成熟的 20 世纪 80 年代初期，各大厂商根据自定义的有偏颇的标准对自家的产品进行市场推广，使得数据库领域一度处于争夺利益的混乱时期. 可见推出公认的评测基准作为系统设计的依据和产品优劣的评价标准是非常有必要的. 大数据的特性[3]（海量、多样性、高速性）使得大数据在系统、应用和数据 3 个层次体现了与传统模式的差别[4]，决定了现有的评测基准不能满足需求，也决定了设计大数据领域评测基准的高度挑战性. 首先，大数据的海量特性使得评测基准需要能够提供多样化、大规模和真实的数据集，然而大数据中蕴藏的巨大商业价值令其成为重要的私有财产. 另外，即使大数据持有者愿意公开其数据，如何传输千万亿字节（Petabyte）甚至更大规模的数据也是一个很大的难题；其次，大数据的多样性表现为应用领域的多样性、数据的多样性、负载的多样性以及软件栈的多样性，这些都直接加剧了构建评测基准的复杂度；最后，大数据系统的快速迭代更新，需要大数据评测基准能够与时俱进，适应大数据系统的快速演变，这同样具有极大的挑战.

如表 1 所示，现有的相关基准不能完整的涵盖大数据的特性. 部分评测基准[5-9]针对特定的应用领域或者软件栈，如 BigBench[5]的目标评测系统是数据库管理系统（DBMS）和 MapReduce 系统，另外有些评测基准没有提供可扩展的数据集，如 CALDA[7]、YCSB（Yahoo! Cloud Serving Benchmark）[8]等，或者负载选取缺乏合理的依据，如 CloudSuite[10]仅仅根据流行度选取大数据负载，缺乏对大数据多样性和复杂性的全面覆盖. 针对现有工作的不足，本文研制了一个跨系统、体系结构、数据管理 3 个领域的大数据基准测试开源程序集——BigDataBench. 为了适应大数据系统快速演变和更新的发展特性，我们采取一种增量式和迭代式的构建方法. 首先，我们使用公认的标准[11]选取互联网服务中最重要的大数据应用领域，同时调研新兴的大数据应用领域，并最终确定 5 个领域：搜索引擎、电子商务、社交网络、多媒体和生物信息学. 其次，针对所选择的应用领域，对数据和负载进行多个维度的全面分析，抽象出领域内典型算法中频繁出现的基本操作单元，并基于此构建评测规范. 它不仅包含丰富的数据类型（如结构化、半结构化和非结构化数据）和数据语义（如文本、表、图和多媒体），而且涵盖多样的负载类型（如离线分析类负载、在线服务类负载、交互式分析类负载和 NoSQL 负载）. 然后，根据不同领域的数据和负载特性，我们定制针对不同领域的基准测试规范，从而指导基准测试的实现. 最后，鉴于软件栈对负载

① https://hadoop.apache.org/
② https://hive.apache.org/
③ http://www.cloudera.com/content/cloudera/en/products-and-services/cdh/impala.html
④ http://nosql-database.org/
⑤ http://zapatopi.net/kelvin/quotes/

的行为特征有很大的影响[12-13],单一的软件栈不能满足大数据系统横向比较的需求,我们基于多种软件栈实现负载,例如,对于离线分析类负载,我们提供Hadoop、Spark、MPI 这 3 种实现方式.同样地,考虑到真实的大数据集不易获取,我们提供了基于真实的小数据集进行建模并保持真实数据特征的大数据生成工具,包含文本、图和表数据生成工具.另外,为了满足不同的评测需求,我们进一步提供多租户混合负载版本[14]和 BigDataBench 子集[12].其中,多租户混合负载版本[14]支持符合真实工作日志动态特征的大数据负载重放,BigDataBench 子集[12]是为系统和体系结构研究者提供的典型负载集合,主要目的是为了减少 BigDataBench 中大量负载给系统和体系结构研究带来的巨大评测开销.BigDataBench 经历了 4 个版本的迭代更新,目前发布的 BigDataBench 3.1 版本是 BigDataBench 2.0[15]的显著升级版,总共包含 14 个真实数据集和 33 个负载的不同软件栈实现.

表 1 相关评测基准对比

评测基准	规范	应用领域	负载类型	负载	基于真实数据的可扩展数据集	多种软件栈的实现	多租户混合负载	子集	模拟器版本
BigDataBench	有	5	4	33	6(可扩展)	是	是	是	是
BigBench	有	1	3	10	3	否	否	否	否
CloudSuite	无	N/A	2	8	3	否	否	否	是
HiBench	无	N/A	2	10	3	是	否	否	否
CALDA	有	N/A	1	5	N/A	是	否	否	否
YCSB	有	N/A	1	6	N/A	是	否	否	否
LinkBench	有	N/A	1	10	1	是	否	否	否
AMP Benchmarks	有	N/A	1	4	N/A	是	否	否	否

本文第 2 节阐述大数据评测基准的需求;第 3 节介绍大数据评测基准相关工作;第 4 节描述 BigDataBench 构造方法以及所包含的数据集和负载;第 5 节论述 BigDataBench 的使用方法和适用范围;第 6 节基于 3 个典型用例描述 BigDataBench 的应用场景;最后第 7 节对全文进行总结.

2 大数据评测基准需求

Gray[16]认为:特定领域的评测基准应选择典型应用,并满足领域内应用的多样性.据此他进一步提出了一套成功的评测基准需要满足的 4 个条件:系统相关性、可移植性、可扩展性和简单.其中,系统相关性是指能够评测领域相关的系统性能,包括系统的峰值性能,性价比等;可移植性是指评测基准能够移植到不同的平台上,易于在不同的系统和架构上实现;可扩展性是指能够适应不同的系统规模;简单是指评测基准易于理解,评测结果具有可靠性.

参考 Gray 提出的 4 条标准,并结合大数据海量、高速、多样的特性,我们提出了针对大数据领域的评测基准需要满足的需求.

(1)可代表性.大数据领域具有非常广的覆盖范围,信息时代的来临使得越来越多的应用领域涉及到大数据的处理和存储,因此一个完整而全面的评测基准不可能一蹴而就.如何尽可能提高负载覆盖度又不失评测的简易性是很大的挑战,这也就要求评测基准具有领域代表性.我们认为大数据领域的代表性主要体现在 3 个方面:① 代表性负载.众所周知,目前应用领域极其繁多,领域之间有一定的共有特性,但每个领域有其独特性,因此应用领域和负载的代表性在一定程度上也就决定了评测基准的代表性;② 代表性数据.大数据领域与传统数据库等领域的一个显著区别即是数据类型多元化,传统的结构化数据不再占据主导地位,半结构化和非结构化数据爆炸性增长,因此评测基准不能忽略复杂而多样的数据类型;③ 代表性软件栈.数据迅猛增长催生了众多的大数据处理和存储系统,然而不同的软件栈对大数据负载的行为特征具有很大的影响[12-13],因此大数据评测基准需要涵盖代表性软件栈.

(2)可移植性.大数据评测基准不仅需要能够纵向地评测大数据系统,而且需要能够对不同的系统进行横向的对比.这就要求相同的负载能够提供不同的实现方式,评测基准能够便利地移植到其他平台.为了使不同的实现方式具有公平的可比性,针对不同平台的实现,需要具有相同的输入和输出,以及相同的算法处理逻辑.如今,一系列针对大数据处理和存储的开源产品被发布,例如 MapReduce、Spark 等,所以在评测基准的实现过程中需要考虑基于这些不同的软件栈的实现.

(3)可扩展性.大数据评测基准需要提供可扩展的数据集和负载.大数据的一个显著特征即是数据量大,单一节点的存储已逐步转变成分布式存储,因此评测基准所提供的数据和负载需要适应不同规

模的平台.然而如今大多数的大数据持有者视数据为重要的商业机密,因而能够提供符合真实数据特性的可扩展数据集是大数据评测基准重要而基本的需求.

(4)可理解性.评测基准需要具有简易性,易于理解,并易于部署和评测,同时评测结果能够指导系统的评价、改进和优化.然而,大数据系统本身非常复杂.仅仅从简单性的角度来选择典型负载,可能会使基准程序丧失代表性.因此,我们用可理解性来取代原有的简单性需要.可理解性有3点含义:能从基本操作单元和负载模式的角度理解典型负载;评测结果需要简单直观,评测人员能够根据负载的特性分析结果的合理性并判断系统的瓶颈或者优劣;评测结果需要具有稳定性,其结果必须是可靠的并且可重现的.

3　相关工作

随着大数据关注热度的持续升温,大数据评测基准吸引了学术界和工业界的广泛研究,国内外涌现了大量相关研究工作.

BigBench[5]是一个针对大数据离线分析的端到端的大数据评测基准,基于TPC-DS构建,并在此模型基础上加入了半结构化和非结构化数据类型.围绕上述3种不同的数据类型,BigBench提供了一系列的查询负载.尽管BigBench包含了丰富的数据类型,但是其目标评测系统是DBMS系统和MapReduce系统,对大数据环境中繁多的软件栈缺少全面的覆盖.

CloudSuite[10]是一套评测Scale-out负载的基准,并根据流行度选取了8个负载,包含使用Hadoop框架运行机器学习任务的数据分析负载,数据缓存负载,依赖雅虎云服务标准测试程序的数据服务负载,图分析负载,流媒体负载,资源需求高且耗时的软件测试负载,Web搜索和服务负载.

HiBench[6]是Intel提出的一套评测MapReduce应用的评测基准,主要包含微基准测试程序,如计数排序等,机器学习相关负载,如分类聚类等,服务类负载,如索引等,和HDFS基准测试.HiBench-4.0加入了对Spark软件栈的支持.

CALDA[7]是评测Hadoop系统和RDBMS系统的测试基准.其所提供的负载是一系列的数据分析任务,主要包含选择操作、查询操作、聚集操作等.其评测指标包含构建索引时间、查询时间、系统加载时间等.

YCSB[8]是评测NoSQL系统的测试基准,主要关注云服务系统的性能和可扩展性.其主要针对NoSQL数据库进行压力测试,测试数据库在并发读取、写入、更新等操作时的行为,如吞吐量等.YCSB提供了一组核心的负载,被称为核心包,其中包含5个负载:频繁更新、频繁读取、小范围、只读和读取最新.

AMP Benchmarks① 是UC Berkeley AMP Lab所提出的一个针对实时分析类应用的大数据评测基准,它主要评测不同规模数据集下一系列关系查询的系统响应时间.其输入数据集包含非结构化的HTML文档和两张SQL表.该团队使用其评测了Redshift、Hive、Shark、Imapla、Stinger/Tez等系统,并报告了实验结果.

LinkBench[9]是针对社交图谱数据库的一个可定制和可扩展的评测基准,该基准根据真实的社交网络应用Facebook开发,实时地查询、更新图数据库,评测指标主要包含延迟、吞吐率等.

BSMA(Benchmark for Analytical Queries over Social Media Data)[17]是一个针对社交媒体分析应用的测试基准.它提供了一个真实的社交网络数据集和符合一定数据分布规律的数据生成工具.同时,它提供了一系列的查询负载,包括图操作、热点查询、时间线查询等.

CloudBM[18]是中国人民大学提出的评测云数据管理系统的测试基准.它以电信业务为应用背景,提供了一组性能评价指标,基于实际应用的数据操作类型和相应的工作流.

综上所述,现有的相关测试基准要么针对特定的应用领域或软件栈,要么缺乏负载选取的合理依据,使得所选取的数据集和负载具有一定的局限性.

4　大数据基准测试方法(Methodology)

本节介绍BigDataBench构造方法.我们从构造方法、测试规范、负载和数据集这几个方面进行介绍.构造方法主要介绍如何构造BigDataBench;测试规范介绍BigDataBench的使用场景;负载和数据集具体介绍根据使用场景选取的负载和数据集.

4.1　构造方法

基于大数据评测基准需要满足的需求,我们提出了BigDataBench构造方法.图1宏观地描述了BigDataBench的构造方法.它主要包含5个步骤:

① https://amplab.cs.berkeley.edu/benchmark/

(1)广泛调研大数据应用领域,并选取典型或者新兴的领域;(2)对选取的应用领域中涉及的数据集和负载进行深入的分析,抽取频繁出现的基本操作单元;(3)针对每一个选取的应用领域提出测试基准规范;(4)提供基于真实数据的数据生成工具和基于多种软件栈实现的负载;(5)根据不同的评测需求提供多租户混合负载版本和 BigDataBench 子集.

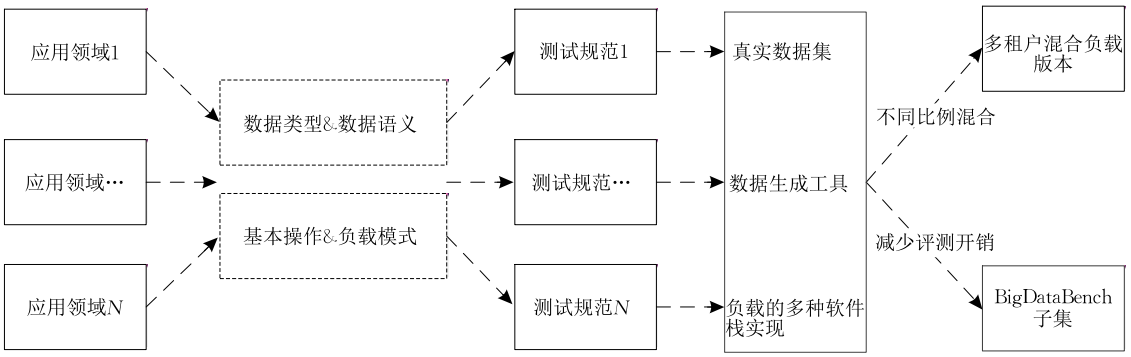

图 1 BigDataBench 构造方法

Seltzer 等人[19]指出为得到真实应用的性能数据必须采用基于应用的评测基准. Chen 等人[20]同样提出评测基准应该与真实的应用领域相关联,并且能够反映真实的计算需求. 受此启发,BigDataBench 构造方法基于具体的应用领域,涵盖领域中真实的数据集和典型的负载. 首先,我们调研大数据领域并且根据公认的指标选取典型和重要的领域. 互联网服务是大数据中的一个典型类别,我们根据公认的标准(页面访问量和每日访问人数)进行筛选,发现搜索引擎、电子商务和社交网络占据互联网服务中 80% 的访问量①. 毋容置疑,搜索引擎、电子商务和社交网络这 3 类应用领域是互联网服务中最重要的应用场景. 鉴于数据的增长渗透到各行各业,我们同样调研了新兴的大数据应用领域,发现多媒体领域②和生物信息学领域③占据了非常重要的地位. 综上所述,我们总共选取了 5 个大数据应用领域:搜索引擎、电子商务、社交网络、多媒体和生物信息学. 针对选取的应用领域,我们深入地分析其中的数据集和负载,主要包含两个方面:(1)数据类型,如结构化、半结构化、非结构化数据和数据语义,如文本、图、表、多媒体数据等;(2)探索领域内常用算法中频繁出现的基本操作单元[21-22]. 然后,为每一个应用领域提出特定的测试规范,从而指导评测基准的实现.

在 BigDataBench 的具体实现中,我们结合数据的类型选取多样的真实数据集,并在此基础上提取真实数据的特性,构造符合特性的数据生成工具. 在负载选择方面,我们全面考虑各种复杂的负载类型,权衡各类型负载所占据的比例,包含离线分析、交互式分析、在线服务和 NoSQL 这 4 种类型. 不仅如此,为了方便各个大数据系统的对比,我们对于每一个负载提供了多种实现方式.

为了满足大数据评测中的特定需求,我们提供了基于 BigDataBench 的两个不同的版本. 一是多租户混合负载版本④:真实的数据中心一般包含多个租户共享基础设施资源,因而提供支持多租户、混合负载场景的评测基准是非常有必要的. 多租户混合负载版本提供了符合真实工作日志动态特征的大数据负载重放,主要考虑了两类典型的负载类型,包含长期运行的服务类负载和短期运行的分析类负载. 混合负载测试工具主要包含 3 个模块:用户接口模块、负载运行日志与真实负载匹配模块、多租户负载生成模块. 其中,用户接口模块与用户交互,接收具体评测需求,如评测场景中的机器和负载类型等,然后根据评测需求选取相应的负载运行轨迹. 负载运行日志与真实负载匹配模块主要是抽取日志中的负载行为特征,进一步采取回归和聚类的方法与真实的负载进行匹配,从而生成负载重放脚本来指导混合负载的生成. 服务类负载的生成主要决定于请求发送频率、请求的序列和请求内容 3 个因素,分析类负载生成主要决定于任务提交时间、负载类型和输入数据 3 个因素. 多租户负载生成模块以负载重放脚本为依据,提取租户信息并构建能够生成服务类和分析类混合负载的多租户框架. 目前,我们主要基于搜狗用户查询日志⑤生成服务类负载,以

① http://www.alexa.com/topsites/global;0
② www.oldcolony.us/wp-content/uploads/2014/11/whatis-bigdata-DKB-v2.pdf
③ www.ddbj.nig.ac.jp/breakdown_stats/dbgrowth-e.html#dbgrowth-graph
④ http://prof.ict.ac.cn/BigDataBench/multi-tenancyversion/
⑤ http://www.sogou.com/labs/dl/q-e.htm

及基于谷歌集群负载日志[①]生成分析类负载. 二是BigDataBench 子集[②]：大数据负载繁多，且实现方式多样，为系统和体系结构研究带来了巨大的开销，尤其是以模拟器为运行环境的研究，因而从大量的负载中选取特征明显并且差异显著的负载能够缓解评测的压力. BigDataBench 子集分析了不同负载及实现，选取了 45 个系统和体系结构层次的指标，通过主成分分析（PCA[23]）和聚类分析（*K*-Means[24]）选出了典型负载的精简集合.

4.2　测试规范

针对所选的 5 个应用领域，我们制定了不同的测试规范，模拟 5 个不同的应用场景并选取其中重要而典型的负载，从而指导大数据评测基准的实现. 在本节中，我们以搜索引擎为例，阐述模拟的场景以及定义的规范.

如图 2 所示，BigDataBench 中的搜索引擎包含两个场景：普通搜索和垂直搜索. 普通搜索是对互联网上爬取的所有网页建立索引，而垂直搜索只对和主题相关的页面建立索引，因此能够反馈更符合搜索主题的页面.

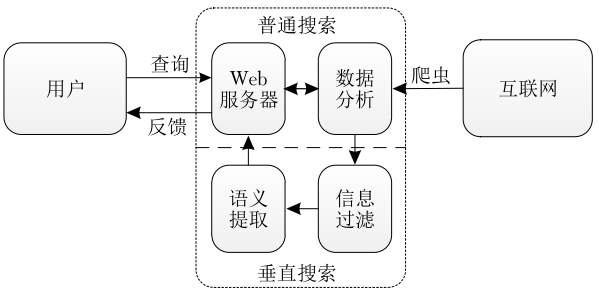

图 2　搜索引擎抽象

图 3 描述了搜索引擎的处理细节. 其输入数据主要是网页，在 BigDataBench 的实现中，网页是由数据生成工具所生成. 在网页的处理流程中，搜索引擎首先解析网页的内容并理清网络图的架构. 其次，索引器根据解析出的内容建立索引，PageRank 程序根据图的链入链出结构计算网页的重要程度. 另外，统计程序提取内容的关键词并对网页进行分类. 当用户输入一个查询请求，搜索引擎读取索引并根据网页的重要程度将其反馈给用户. 整个过程涉及到的关键负载如下：

（1）解析. 提取网页的内容和外部链接. 解析是获取网页页面之后的第一个步骤，通过正则表达式匹配 HTML 标签得到.

（2）统计. 计算网页内容中每个词的词频，从而提取描述网页主题的关键词.

（3）分类. 根据网页关键词进行分类，将网页划分入不同的主题.

（4）索引. 建立单词到文档号的映射，当搜索到单词时，能够检索到包含单词的文档.

（5）PageRank. 根据网页的外部链接图迭代计算网页的重要性.

（6）搜索请求. 在线网络搜索服务器.

（7）排序. 根据网页的重要性以及与搜索词的相近程度对结果进行排序.

（8）推荐. 根据搜索的日志文件对搜索关键词进行推荐.

（9）过滤. 针对垂直搜索引擎的主题，过滤与主题无关的页面，仅保留与主题相关的内容.

（10）语义提取. 提取网页的语义信息.

（11）数据存取. 读取、写入或者扫描提取的语义信息.

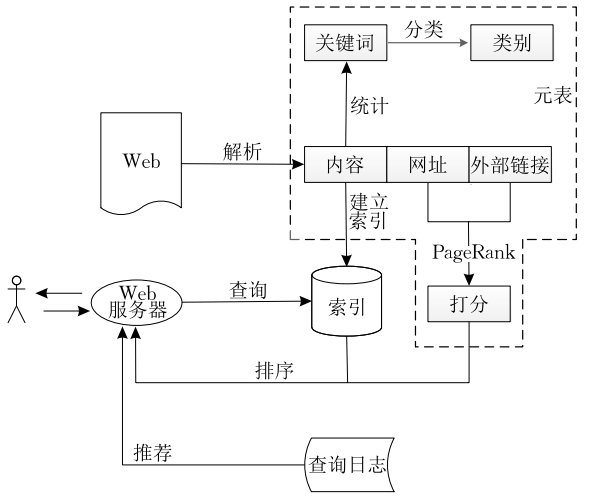

图 3　搜索引擎细节描述

4.3　真实数据集和数据生成工具

大数据处理中包含丰富的数据类型，据统计，80％的数据增长来自于非结构化数据[③]，如文档、图片、音视频等. 作为大数据系统的评测基准，需要能够覆盖多样的数据类型和数据语义，并满足数据集的可扩展性. 本小节介绍我们所选择的真实数据集以及相应的数据生成工具.

4.3.1　真实数据集

通过调研 5 个选取的应用领域，我们收集了

① http://code.google.com/p/googleclusterdata/
② http://prof.ict.ac.cn/BigDataBench/simulatorversion/
③ http://www.oldcolony.us/wp-content/uploads/2014/11/whatisbigd ata-DKB-v2.pdf

14 个真实的数据集.其代表性主要体现在:(1)数据来源于可靠而权威的机构或组织,如维基百科、美国国家生物技术信息中心(NCBI)等,和知名的企业,如谷歌、亚马逊等;(2)在各自所属的应用领域内,数据作为常用的数据源已被多次引用并用于评测,如 ImageNet 是目前图像识别最大的数据库,其相关论文已被引用 1600 多次;(3)结合大数据领域的数据特性,我们考虑了多样的数据模式,涵盖了结构化、半结构化和非结构化数据类型,包含了文本、图、表、图像和音视频数据语义.表 2 列举了我们选取的真实数据及相应的数据生成工具,每个数据集的详细介绍如下.

表 2 真实数据集

标号	数据集	数据规模	数据类型	数据生成工具
1	维基百科条目	4 300 000 篇英文文章	非结构化文本	文本生成工具
2	亚马逊电影评论	7 911 684 条评论	半结构化文本	文本生成工具
3	谷歌 Web 图	875 713 个结点,5 105 039 条边	非结构化图	图生成工具
4	Facebook 社交图谱	4039 个结点,88 234 条边	非结构化图	图生成工具
5	电子商务交易数据	表 1:38 658 行 4 列;表 2:242 735 行 6 列	结构化表	表生成工具
6	ProfSearch 科研人简历	278 956 份简历	半结构化表	表生成工具
7	ImageNet	21 841 个非空集合,14 197 122 张图片	非结构化图片	待开发
8	英语广播音频	约 8000 个来自 VOA、BBC 等的音频文件	非结构化音频	待开发
9	DVD 输入流	110 个输入流数据,分辨率 704×480	非结构化视频	待开发
10	图像场景描述数据集	39 个图像场景描述文件	非结构化文本	待开发
11	基因序列数据	4 种不同规模的序列数据(21 MB~7.1 GB)	非结构化文本	待开发
12	基因组装数据	4 种不同规模的组装数据(100 MB~13 GB)	非结构化文本	待开发
13	SoGou 数据集	语料库和查询数据	非结构化文本	待开发
14	MNIST	60 000 个训练数据和 10 000 个测试数据的手写体数据库	非结构化图片	待开发

维基百科条目[①]是非结构化文本数据集,具有 4 300 000 篇英文文章,其覆盖的主题包含艺术、地理、历史、数学、自然、科技等.

亚马逊电影评论[②]数据集是半结构化文本数据集,包含 1997 年 8 月至 2012 年 10 月期间 253 059 个用户对 889 176 部电影的 7 911 684 条评论.

谷歌 Web 图[③]是非结构化图数据集,描述了 875 713 个网页之间 5 105 039 个链接的图结构.

Facebook 社交图谱[④]是非结构化图数据集,包含 4039 个节点和 88 234 条边,其中每一个节点表示一个用户,每一条边表示其所连接的两个用户之间的朋友关系.

电子商务交易数据是结构化表数据,此数据集来自电子商务网站并且经过了匿名处理.它主要包含订单和订单项目两张表.

ProfSearch 科研人简历是半结构化表数据.该数据集来自我们自己开发的针对科研人员的垂直搜索引擎,总共包含从大学和研究机构的约 20 000 000 个网页中自动提取的 278 956 份简历.

ImageNet[25]是非结构化图片数据集,它根据 WordNet 层次结构进行分类,目前包含其中的 21 841 个同义词集合,每个集合平均超过 500 张图片,现总共有 14 197 122 张图片.

英语广播音频[⑤]是非结构化音频数据集,包含来自美国之音(VOA)、英国广播公司(BBC)、有线电视新闻网(CNN)、国际广播电台(CRI)和美国国家公共电台(NPR)的约 8000 个音频文件.

DVD 输入流[⑥]是非结构化视频数据集,包含 110 个分辨率为 704×480 的输入流数据.

图像场景描述数据集[⑦]是非结构化文本数据集,包含 39 个图像场景描述文件,主要从几何坐标、视角、光线、阴影等多个角度对场景进行描述.

基因序列数据[⑧]是非结构化文本数据集,包含 4 种不同的基因数据,规模从 20MB 至 7 GB,基因片段(Reads)的数量从 101 617 条至 31 257 852 条.

基因组装数据[⑨]是非结构化文本数据集,包含 4 种不同规模的组装数据,从 100 MB~13 GB.

SoGou 数据集[⑩]是非结构化文本数据,包含来自 SoGou 实验室的语料库和查询数据.基于语料库我们得到了 4.98 GB 的索引数据.

MNIST[⑪] 是非结构化图片数据集.它是一个手写体数据库,提供了训练集和测试集,其中训练集包含 60 000 个手写体数据,测试集包含 10 000 个手写体数据.

4.3.2 数据生成工具

大数据不仅获取难度大,而且下载开销大,因此我们提供了能够体现大数据真实性、多样性、速率和

① http://en.wikipedia.org
② http://snap.stanford.edu/data/web-Amazon.html
③ http://snap.stanford.edu/data/web-Google.html
④ http://snap.stanford.edu/data/egonets-Facebook.html
⑤ http://www.tingvoa.com
⑥ ftp://ftp.tek.com/tv/test/streams/Element/index.html/
⑦ http://jedi.ks.uiuc.edu/johns/raytracer/
⑧ http://ccl.cse.nd.edu/software/sand/
⑨ http://hgdownload.cse.ucsc.edu/goldenPath/hg19/bigZips/
⑩ http://www.sogou.com/labs/
⑪ http://yann.lecun.com/exdb/mnist/

规模的大数据生成工具[26-27]（BDGS）. 通过对真实数据的建模分析，BDGS 能够快速生成保持真实数据特征的指定规模数据集. 目前该工具包含文本、表和图数据生成工具. 针对多媒体数据和基因数据的生成工具还待进一步开发.

图 4 描述了 BDGS 的构造方法. 数据生成工具总共包含 4 个步骤：(1) 数据筛选，即选取代表性的真实数据集和相应的建模方法或工具；(2) 原始数据处理，即对真实数据采样并建模，提取数据的特性；(3) 数据生成，即通过参数控制数据规模和并行度；(4) 格式转化，即根据负载的输入需求转换生成数据的格式.

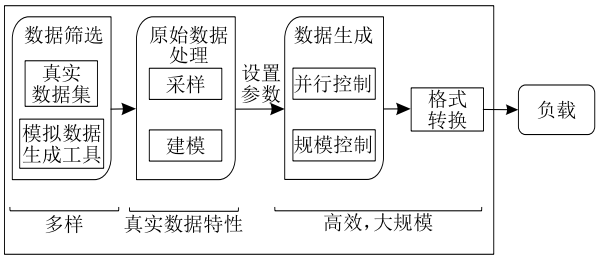

图 4　BDGS 构造方法

图 5 是围绕构造方法设计 BDGS 的总体架构图，它总共包含 3 个模块：数据处理生成模块、并行控制模块和格式转化模块. 数据处理生成模块首先完成构造方法中第 2 个步骤，即原始数据处理，然后并行控制模块设置规模和并行度，将任务分发，按照参数控制器的输入参数生成指定规模的大数据，最后格式转化模块根据负载的输入需求进行格式转化. 在原始数据的建模过程中，文本数据生成工具基于统计学方法，采用隐含狄利克雷分布模型[28]（LDA）进行建模；图数据生成工具采用 Kronecker[29] 模型进行建模；表数据生成工具基于元数据和业务规则进行建模. 我们以文本数据生成为例，阐述 BDGS 如何对真实数据进行建模并生成保持真实数据特性的指定规模文本数据集.

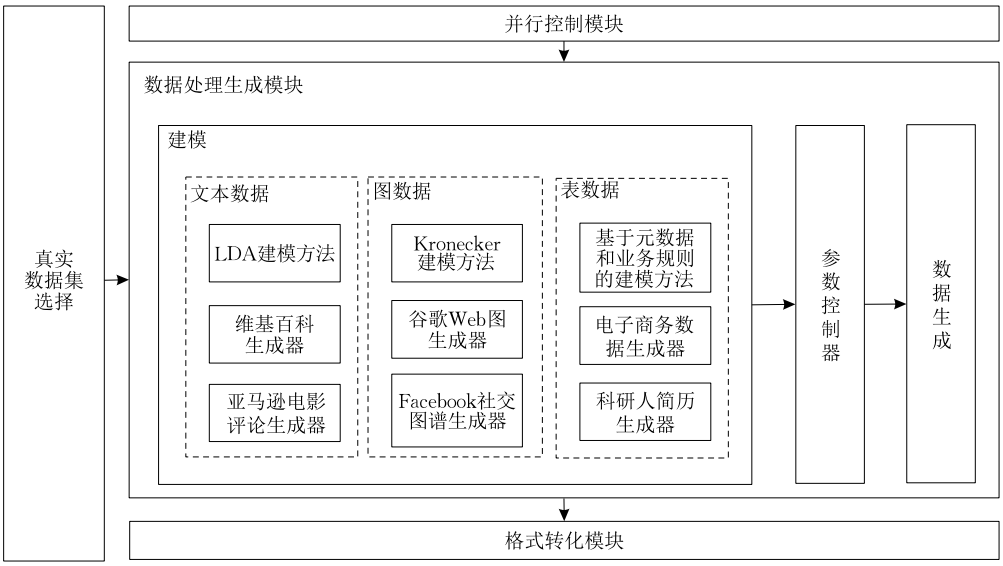

图 5　BDGS 架构

LDA 模型采用词袋的方法，通过将每篇文档转换成一个词频向量进行建模. 对于每一篇文档，LDA 定义了从主题分布中抽取一个主题，从该主题对应的单词分布中抽取一个单词的过程. LDA 模型不仅考虑了词项，而且考虑了隐含的主题，提取了文档的语义特征. 因而相对于传统的依据某个概率分布从词典中取词的数据生成方法，LDA 模型更多地保留了真实数据特征.

图 6 描述了 LDA 主题模型. 在 LDA 模型中①，主题 z 和单词 w 都符合多项式分布，其中主题多项式分布的参数 $\boldsymbol{\theta}$ 符合狄里克雷分布，此狄里克雷分布的参数为 $\boldsymbol{\alpha}$；单词多项式分布的参数为 $\boldsymbol{\beta}$. LDA 建模过程主要根据真实语料库训练得出参数 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$，进而确定整个模型，并生成文档. $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 包含的信息为：$\boldsymbol{\alpha}$ 是分布 $P(\boldsymbol{\theta}|\boldsymbol{\alpha})$ 中需要的参数，即狄里克雷分布参数，用以生成文档主题分布 $\boldsymbol{\theta}$，$\boldsymbol{\beta}$ 是单词多项式分布 $P(\boldsymbol{w}_n|\boldsymbol{z}_n,\boldsymbol{\beta})$ 中需要的参数，用以生成主题对应的单词. 使用训练出的 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 生成文本数据的过程如下：

首先，通过参数为 $\boldsymbol{\alpha}$ 的的狄利克雷分布生成文档主题分布 $\boldsymbol{\theta}$，然后，以 $\boldsymbol{\theta}$ 作为参数，通过多项式分布生成 N 个主题，最后，对每个主题，通过参数 $\boldsymbol{\beta}$ 的多项式分布，生成对应的单词.

① 后文描述中向量用黑体字母表示.

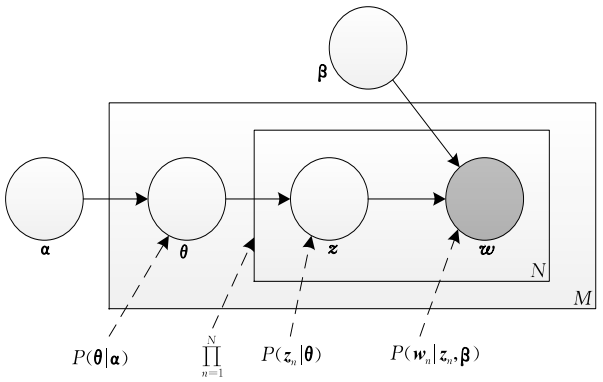

图 6 LDA 主题模型[28]

图 7 描述了文本数据处理和生成的整个流程，总共包含 5 个步骤：(1)针对所选的真实数据集，进行数据预处理，即清洗原始数据，去掉无用的字符；(2)对预处理之后的数据进行分词处理，统计词项并生成数据词典. 词典的生成主要包含是否去掉停用词，如何筛选词项两个方面. 对于微基准测试程序，如 Sort、Grep 等，数据词典的生成方式并不会对它们的运行行为产生较大的影响，然而对于索引构建和文本分类等负载，词典的生成方式会影响负载的运行行为. 因此我们在生成数据词典时根据不同负载的特性来决定处理方式，如是否去掉停用词等；(3)根据预处理后的数据和数据词典生成原始数据集的词频矩阵；(4)将上一步得到的词频矩阵作为输入，训练 LDA 模型，得到 $\boldsymbol{\alpha}$ 和 $\boldsymbol{\beta}$ 参数. 我们采用开源的 LDA-C① 程序对输入数据进行建模；(5)根据模型参数和控制参数(并行度控制、规模控制)生成指定规模的文本大数据.

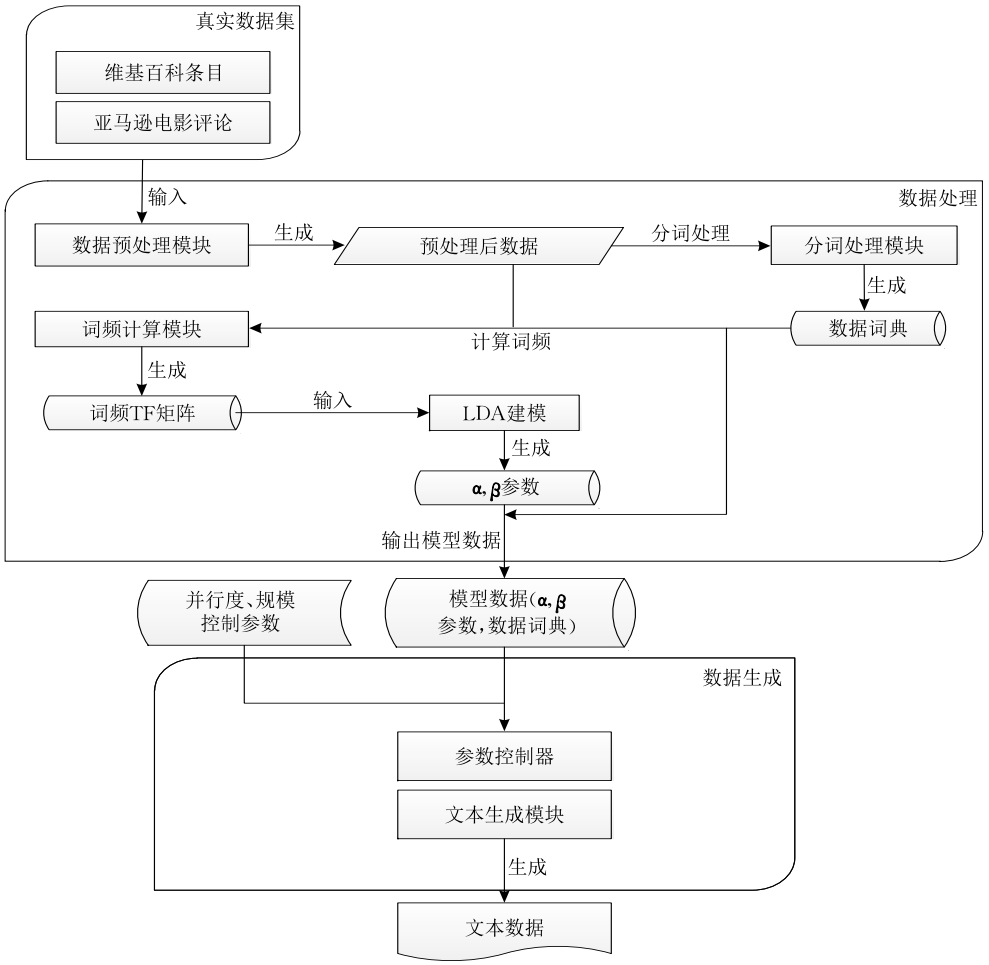

图 7 文本数据处理生成流程图

4.4 负 载

考虑到大数据评测基准的 4 个基本需求，我们总共选取了 33 个来自 5 个应用领域的负载，选取原因及代表性体现主要描述如下：(1)它们是 5 个典型应用领域中的基本并且重要的负载. 如测试规范所述，通过模拟 5 个应用场景，我们挖掘场景中必不可少的负载. 例如，对于电子商务而言，应用于商品推荐的协同过滤负载(Collaborative Filtering)、应用

① http://www.cs.princeton.edu/~blei/lda-c/

于商品分类的 Naïve Bayes 负载和应用于数据库查询的交互式分析类负载是其中关键的负载；(2) 它们涵盖了大数据处理中多样的负载类型，如离线分析、在线服务等；(3) 它们能够处理不同的数据类型，能够以大数据场景中复杂的数据模式作为输入；(4) 它们使用不同的软件栈实现，能够用于评测不同的软件栈. 表 3 从应用领域、负载名称、负载类型、使用数据集、软件栈 5 个维度描述了 BigDataBench 包含的负载. 对应于上述搜索引擎测试规范的描述，我们同样以搜索引擎为例介绍怎样通过测试规范选取典型的负载. 在 4.2 节搜索引擎测试规范中，我们定义了 11 个负载，目前我们提供了其中 7 个负载的实现，包含解析(Grep)、统计(WordCount)、索引(Index)、PageRank、搜索请求(Nutch Server)、排序(Sort)、数据存取(Read，Write，Scan). 我们的评测基准测试规范总共包含 42 个负载，目前我们实现了其中的 33 个，另外的 9 个负载会在未来工作中实现.

表 3　负　载

应用领域	负载名称	负载类型	使用数据集	软件栈
搜索引擎	Grep	离线分析	维基百科条目	Hadoop，Spark，MPI
	WordCount	离线分析	维基百科条目	Hadoop，Spark，MPI
	Index	离线分析	维基百科条目	Hadoop，Spark，MPI
	PageRank	离线分析	谷歌 Web 图	Hadoop，Spark，MPI
	Nutch Server	在线服务	SoGou 数据集	Nutch
	Sort	离线分析	维基百科条目	Hadoop，Spark，MPI
	Read	NoSQL	ProfSearch 科研人简历	HBase，MySQL
	Write	NoSQL	ProfSearch 科研人简历	HBase，MySQL
	Scan	NoSQL	ProfSearch 科研人简历	HBase，MySQL
社交网络	Connected Component	离线分析	Facebook 社交图谱	Hadoop，Spark，MPI
	K-means	离线分析	Facebook 社交图谱	Hadoop，Spark，MPI
	BFS	离线分析	Graph500 生成工具	MPI
电子商务	选择查询	交互式分析	电子商务交易数据	Hive，Shark，Impala
	聚集查询	交互式分析	电子商务交易数据	Hive，Shark，Impala
	连接查询	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Collaborative Filtering	离线分析	亚马逊电影评论	Hadoop，Spark，MPI
	Naïve Bayes	离线分析	亚马逊电影评论	Hadoop，Spark，MPI
	Project	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Filter	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Cross Product	交互式分析	电子商务交易数据	Hive，Shark，Impala
	OrderBy	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Union	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Difference	交互式分析	电子商务交易数据	Hive，Shark，Impala
	Aggregation	交互式分析	电子商务交易数据	Hive，Shark，Impala
多媒体	MPEG	离线分析	DVD 输入流	Libc
	SIFT	离线分析	ImageNet	MPI
	DBN	离线分析	MNIST	MPI
	语音识别	离线分析	英语广播音频	MPI
	光线跟踪	离线分析	图像场景描述数据集	MPI
	图像分割	离线分析	ImageNet	MPI
	人脸检测	离线分析	ImageNet	MPI
生物信息学	SAND	离线分析	基因序列数据	Work Queue
	BLAST	离线分析	基因组装数据	MPI

所有的 33 个负载均来自于选取的 5 个应用领域，由于软件栈对负载的行为特征有很大的影响[12-13]，所以我们提供负载的多种软件栈实现. 例如，对于离线分析类负载，我们提供 Hadoop、Spark、MPI 这 3 种实现；对于交互式分析类负载，我们提供 Hive、Shark、Impala 这 3 种实现. 另外，对于不同的负载，我们优先选取目前主流的及其适合的软件栈进行实现. 在未来的工作中，我们会加入更多的实现方式，例如多媒体负载的不同软件栈实现. 针对不同的评测目标，用户可自行选择部分或者全部负载进行评测. 每个负载的详细介绍描述如下：

(1) 搜索引擎

Grep 负载用于获取网页之后的第一步--解析网页，从中提取搜索的字符串.

WordCount 负载统计词频信息，从而发现网页中的关键字，便于之后的分类处理.

Index 负载是对网页内容建立索引,通过对内容的分词和停用词去除等处理,对词元进行索引.

PageRank 负载用于给所有的网页打分,通过网页之间的链接关系迭代计算网页的重要程度.

Nutch Server 负载提供在线搜索的服务,对搜索请求进行查询反馈.

Sort 负载是对内容进行排序,如对网页的打分数据进行排序从而决定页面的显示顺序.

Read、Write、Scan 负载表示从数据库中读取、写入或者顺序读取记录.

(2)社交网络

Connected Component 负载提取社交网络图中的连通分支,从而发现不同的社区.

K-means 负载通过分析节点的相似性(距离)对社交网络图进行聚类分析.

BFS 负载对图进行广度优先搜索,从某一个节点开始,广度优先遍历整个图.

(3)电子商务

选择查询负载用于找出一个订单中订购数超过 100 的物品.

聚集查询负载用于计算每个物品的销售总量.

连接查询负载用于计算物品在给定时间段内被每个用户所购买的数量.

Collaborative Filtering 负载主要用于物品推荐,根据相似性推荐用户可能感兴趣的物品,主要为矩阵计算.

Naïve Bayes 负载利用贝叶斯定理预测类别未知的文档最可能属于的类别,主要用于分类.

Project、Filter、Cross Product、OrderBy、Union、Difference、Aggregation 是交互式分析中基本的操作.

(4)多媒体

MPEG 负载[30]是按照 MPEG-2 的标准对视频进行编码解码.

SIFT 负载[31]用于检测图像中对旋转、光亮、缩放等保持不变的局部性特征.

DBN 负载[32]是对深度信念网络的实现,用于手写体识别.

语音识别负载①通过建立声学模型和语言模型将音频信息翻译成文本信息.

光线跟踪负载[33]根据图像场景描述文件的参数,跟踪光线生成 3D 图像.

图像分割负载[34]通过对每个像素加标签将图像分割成多个子区域.

人脸检测负载[35]根据脸部特征判断图像中是否有人脸以及检测人脸的位置和尺寸.

(5)生物信息学

SAND 负载[36]实现基因数据的组装,将很多个基因碎片组装成原始的基因序列.

BLAST 负载[37]实现基因序列的对比,将基因片段与数据库中的数据进行对比以检测相似性.

5 BigDataBench 使用方法和适用范围

本节介绍 BigDataBench 的使用方法,包括使用原则和使用步骤,并讨论 BigDataBench 的适用范围.

5.1 使用方法

BigDataBench 从 5 个典型的应用场景出发,选取了 14 个真实数据集和 33 个负载.针对不同的评测需求,具有不同的评测方法.以下主要分为 5 个基本的步骤来介绍其使用方法.

(1)根据评测需求选取合适的负载和数据集

BigDataBench 提供了多种类型的数据集和负载,用户可根据其具体的评测需求进行针对性的选择.例如,对于关注评测交互式数据分析系统的用户而言,他们只需要选取交互式分析类负载和电子商务交易数据;而对于进行体系结构研究的用户而言,我们提供了 BigDataBench 子集用以减少评测开销.

(2)部署集群环境并下载安装 BigDataBench

针对用户选取的系统或者软件栈部署集群环境,下载 BigDataBench 并进行配置.

(3)使用数据生成工具 BDGS 生成指定规模数据集

根据集群的规模和数据集的类型,选取相应的文本、图或者表数据生成工具,并生成指定规模的数据集.

(4)运行负载并调优

运行第一步选取的大数据负载,并根据实验结果进行调优,如更改配置参数等.

(5)采集实验数据并分析得到评测结果

根据评测的指标和评测的目的采集实验数据,如负载运行时间、吞吐率、系统层行为特征等.分析和对比实验数据得出评测结果.

由于篇幅限制,本文仅介绍使用的一般步骤,具体的数据生成和负载运行等使用方法请参照我们提供的用户手册②.

① http://cmusphinx.sourceforge.net/

② http://prof.ict.ac.cn/BigDataBench/wp-content/uploads/2014/12/BigDataBench-handbook-6-12-16.pdf

5.2　适用范围

BigDataBench 是跨系统、体系结构和数据管理 3 个研究领域的大数据基准测试程序集. 适用于大数据系统评测、处理器体系结构设计、负载分析、数据管理系统评测等研究工作.

6　BigDataBench 应用案例

如上一节所述，大数据评测基准 BigDataBench 具有广泛的应用，本节从 3 个典型案例分别介绍 BigDataBench 在大数据系统评测（此案例主要评测系统开销）、处理器体系结构设计和负载分析中的应用. 针对每一个案例，我们主要从两个方面进行介绍，一是案例本身的工作内容，二是在案例中如何使用 BigDataBench 以及得出的结论.

6.1　云数据安全

剑桥大学计算机实验室（The Computer Laboratory）设计了一个 Hadoop MapReduce 框架下保护敏感数据的系统——MrLazy[38]. 它总共包含 4 个子系统：来源跟踪、来源重建、字段级跟踪分析和标签生成. 来源跟踪主要获取记录（Record）的来源信息，即跟踪输出记录具体是由哪些输入记录处理得到，并保存输入记录和输出记录间的链接关系. MapReduce 具有 Map 和 Reduce 两个阶段，来源跟踪子系统在 Map 阶段获取中间记录到输入记录的链接关系，在 Reduce 阶段获取输出记录到中间记录的链接关系. 来源重建将前两部分的链接关系根据中间记录进行连接，得到输入记录和输出记录的直接链接关系. 字段级跟踪分析通过分析二进制文件判断执行过程中是否使用敏感字段产生输出. 输出记录的标签是以对应输入记录的标签作为函数输入所得到. 图 8 描述了 MrLazy 的输出标签生成过程.

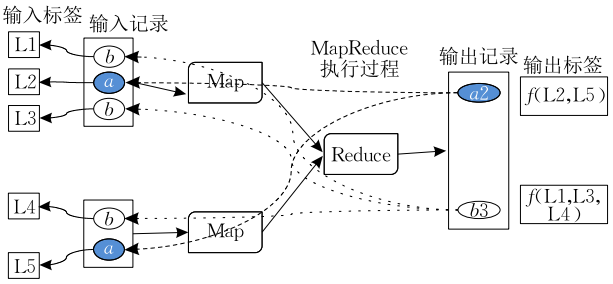

图 8　MrLazy 输出标签生成过程

由于 BigDataBench 提供了代表性的 Hadoop 负载以及相应的数据生成工具，论文选取 BigDataBench 作为测试负载，主要选用了交互式分析中的 Join 负载和电子商务交易数据. 他们使用表数据生成工具生成了 120 GB 的数据，并在 7 个节点组成的集群上运行所选的负载，用以评测 MrLazy 系统的开销. 实验发现从时间开销方面，来源跟踪部分大约增加了 19% 的运行时间，来源重建部分占任务实际运行时间的 18%；从空间开销方面，执行完来源重建之后生成的链接数据文件大小是输出数据的 50%，是输入数据的 5%.

6.2　体系结构设计

Zou 等人[39]介绍了多核领域中 3 种主流的异构集成技术：单指令集异构多核处理器体系结构设计、CPU 和 GPU 异构计算、异构混合内存架构. 其中，单指令集异构多核设计是指所有的核使用同一套指令集但是具有不同的计算资源，如高性能强核和低功耗弱核的组合；CPU 和 GPU 异构计算能够综合 CPU 低延迟和 GPU 高吞吐量的优势；异构混合内存架构是指传统的 DRAM/SRAM 技术和新兴的非易失性内存（NVM）的集成，如相变存储器（PCM）[40].

论文选取了 BigDataBench 中的 6 个大数据负载（K-means，Aggregation，PageRank，Sort，选择查询，连接查询），然后抽取负载的内存轨迹并使用 NVMain[41] 重放，从而研究不同内存混合类型下负载的延迟和功耗特征. 论文中讨论了 4 种不同的配置：单纯使用 DRAM、单纯使用 PCM、DRAM 和 PCM 结合使用、DRAM 作为缓存. 实现发现从延迟角度，单纯使用 PCM 延迟最大，而单纯使用 DRAM 延迟最小；从功耗角度，单纯使用 DRAM 和使用 DRAM 作为缓存这两种配置具有更高的功耗，单纯使用 PCM 的功耗最低.

6.3　负载分析

负载分析主要包含负载的系统层和体系结构层行为，如 CPU 利用率、访存行为、网络传输、指令特征、流水线行为、TLB（Translation Lookaside Buffer）行为、缓存行为等. 这些行为影响着系统和体系结构的设计，深入理解负载的行为特征能够为系统部件设计和硬件配置提供指导，对系统的性能分析和调优同样具有非常重要的作用[42].

BigDataBench 包含多样的大数据负载. 从负载类型的角度包含在线服务类、离线分析类、交互类、NoSQL 等；从负载特征的角度包含计算密集型、I/O 密集型、混合型. 目前已有很多的相关工作使用 BigDataBench 进行负载分析. Jiang 等人[13]探索内存计算框架 Spark 的负载行为特性，并从系统层面

和微体系结构层面发现 Spark 负载与 Hadoop 负载和传统的 HPC 负载均有较大的差异. Pan 等人[43]分析数据中心大数据负载的 I/O 特性,并发现内存的增加能够有效降低 I/O 请求数,以及 HDFS 和 MapReduce 具有不同的 I/O 行为. Jia 等人[44]系统地分析了数据中心分析类负载的微体系结构层的行为特征,并与传统的负载如 SPEC CPU2006①、HPCC[45]、SPEC WEB2005② 以及 CloudSuite 中的 Scale-out 服务类负载进行对比,并发现数据分析类负载和服务类负载在指令执行和流水线停顿方面都具有较大的差异,其中,流水线停顿是影响流水线效率的一个关键因素,它是指在指令流水线的执行过程中,由于指令之间的相关性等原因,使得流水线被迫停顿一个或者多个时钟周期,以保证程序的正确执行. 从指令执行角度,数据分析类负载每时钟周期执行的指令数(IPC)比服务类负载多;从流水线停顿角度,分析类负载和服务类负载均有较大的前端停顿,然而分析类负载主要是乱序执行阶段的停顿,服务类负载主要是进入乱序执行之前的停顿.

7 经验与总结

到目前为止,BigDataBench 总共经历了 4 个版本的迭代更新. 图 9 描述了其整个发展历程. 2013 年,我们发布了第一个版本,主要由 3 个评测基准组成,包含 BigDataBench 1.0、DCBench 1.0 和 CloudRank 1.0. 之后,我们将前 3 个评测基准进行合并并发布了 BigDataBench 2.0 版本,它来自搜索引擎、社交网络、电子商务等 3 个应用场景,总共包含 6 个真实数据集和 19 个大数据负载,并具有能够保持真实数据特性的大数据生成工具. 第 3 个版本增加了数据集和负载,包含 6 个真实数据集、2 个合成的数据集以及 32 个负载. 其主要的更新在于关注大数据负载中频繁出现的基本操作.

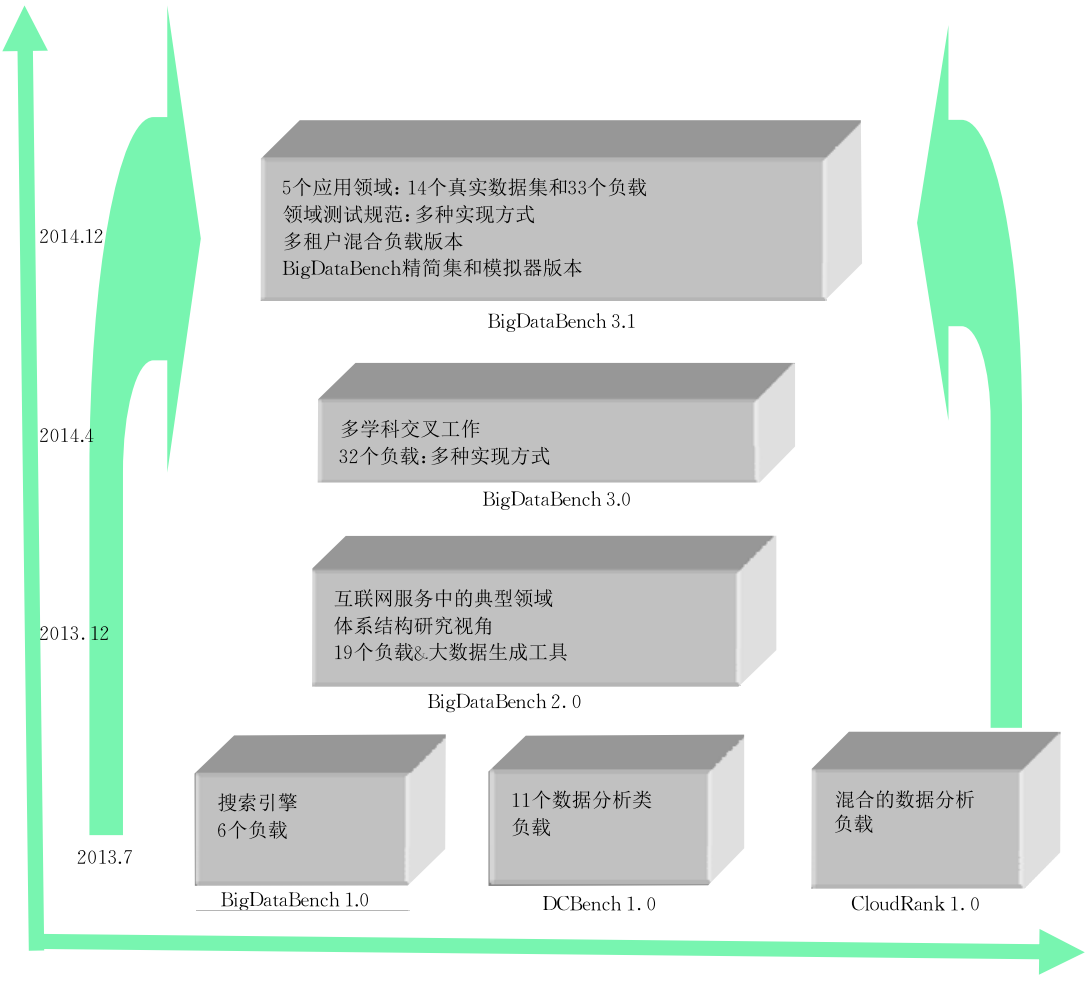

图 9 BigDataBench 发展历程

BigDataBench 3.1 版本较之前的版本是一个显著升级的版本,它加入了新兴的应用领域——多媒

① https://www.spec.org/cpu2006/
② https://www.spec.org/web2005/

体和生物信息学.通过调研选取的5个应用领域,分析领域中广泛使用的具有代表性的算法,统计出算法中频繁出现的基本操作,进一步针对各个领域制定测试规范,模拟应用场景,并按照测试规范选取典型的负载,从而指导评测基准实现.BigDataBench 3.1包含14个真实的数据集、33个负载实现和采用多样化的实现方式.在此基础上提供两个针对不同应用需求的版本,一是符合数据中心真实工作日志动态特征的多租户混合负载版本,二是降低体系结构研究开销的BigDataBench子集.

在之后的工作中,我们会加入更多的典型领域,进一步开发和优化数据生成工具及相应负载实现,同时加入更多新兴的大数据处理和存储系统,如Flink①、Cassandra②、Mongodb③等.

BigDataBench设计方法紧密联系工业界,深度调研真实的应用领域和实际的评测需求,目前已广泛应用在学术界和工业界中,并在此基础上完成了国内首个④工业标准的大数据平台性能评测标准⑤.

参　考　文　献

[1] Zaharia M, et al. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing//Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. San Jose, USA, 2012: 2-2

[2] Low Y, Bickson D, Gonzalez J, et al. Distributed GraphLab: A framework for machine learning and data mining in the cloud. Proceedings of the VLDB Endowment, 2012, 5(8): 716-727

[3] Graham-Rowe D, Goldston D, Doctorow C, et al. Big data: Science in the petabyte era. Nature, 2008, 455(7209): 8-9

[4] Jin Che-Qing, Qian Wei-Ning, Zhou Min-Qi, et al. Benchmarking data management systems: From traditional database to emergent big data. Chinese Journal of Computers, 2015, 38(1): 18-34(in Chinese)
(金澈清, 钱卫宁, 周敏奇等. 数据管理系统评测基准: 从传统数据库到新兴大数据. 计算机学报, 2015, 38(1): 18-34)

[5] Ghazal A, Rabl T, Hu M, et al. BigBench: Towards an industry standard benchmark for big data analytics//Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data. New York, USA, 2013: 1197-1208

[6] Huang S, Huang J, Dai J Q, et al. The HiBench benchmark suite:Characterization of the MapReduce-based data analysis//Proceedings of the ICDE Workshops on Information & Software as Services. LongBeach, USA, 2010: 41-51

[7] Pavlo A, Paulson E, Rasin A, et al. A comparison of approaches to large-scale data analysis//Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data. Providence, USA, 2009:165-178

[8] Coper B, Silberstein A, Tam E, et al. Benchmarking cloud serving systems with YCSB//Proceedings of the 1st ACM Symposium on Cloud Computing. Indianapolis, USA, 2010: 143-154

[9] Armstrong T G, Ponnekanti V, Borthakur D, Callaghan M. LinkBench: A database benchmark based on the Facebook social graph//Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data. New York, USA, 2013: 1185-1196

[10] Ferdman M, Adileh A, Kocberber O, et al. Clearing the clouds: A study of emerging scale-out workloads on modern hardware. ACM SIGPLAN Notices, 2012, 47(4): 37-48

[11] Burby J, Atchison S. Actionable Web Analytics: Using Data to Make Smart Business Decisions. New York, USA: John Wiley & Sons, 2007

[12] Jia Z, Zhan J, Wang L, et al. Characterizing and subsetting big data workloads//Proceedings of the IEEE International Symposium on Workload Characterization(IISWC). Raleigh, USA, 2014: 191-201

[13] Jiang T, Zhang Q, Hou R, et al. Understanding the behavior of in-memory computing workloads//Proceedings of the IEEE International Symposium on Workload Characterization (IISWC). Raleigh, USA, 2014: 22-30

[14] Han R, Zhan S, Shao C, et al. BigDataBench-MT: A benchmark tool for generating realistic mixed data center workloads. Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China: Technical Report arXiv: 1504.02205, 2015

[15] Wang L, Zhan J, Luo C, et al. BigDataBench: A big data benchmark suite from internet services//Proceedings of the IEEE 20th International Symposium on High Performance Computer Architecture (HPCA). Orlando, USA, 2014: 488-499

[16] Gray J. Benchmark Handbook for Database and Transaction System. 2nd Edition. San Francisco: Morgan Kaufmann Publishers, 1933

[17] Xia Fan, Li Ye, Yu Chengcheng, et al. BSMA: A benchmark for analytical queries over social media data. Proceedings of the VLDB Endowment, 2014, 7(13): 1573-1576

[18] Liu Bing-Bing, Meng Xiao-Feng, Shi Ying-Jie. CloudBM: A benchmark for cloud data management systems. Journal of Frontiers of Computer Science and Technology, 2012, 6(6): 504-512(in Chinese)

① https://flink.apache.org/
② http://cassandra.apache.org/
③ https://www.mongodb.org/
④ http://www.cnii.com.cn/Bigdata/2014-12/26/content_1505248.htm
⑤ http://prof.ict.ac.cn/BigDataBench/industry-standard-benchmarks/

(刘兵兵，孟小峰，史英杰. CloudBM：云数据管理系统测试基准. 计算机科学与探索，2012，6(6)：504-512)

[19] Seltzer M，Krinsky D，Smith K，et al. The case for application-specific benchmarking//Proceedings of the IEEE 7th Workshop on Hot Topics in Operating Systems. Rio Rico，USA，1999：102-107

[20] Chen Y，Raab F，Katz R. From TPC-C to big data benchmarks：A functional workload model//Rabl T，Poess M，Baru C，Jacobsen H-A eds. Specifying Big Data Benchmarks. Springer Berlin Heidelberg，2014：28-43

[21] Zhu Y，Zhan J，Weng C，et al. BigOP：Generating comprehensive big data workloads as a benchmarking framework//Proceedings of the Database Systems for Advanced Applications. Bali，Indonesia，2014：483-492

[22] Gao W，Luo C，Zhan J，et al. Identifying dwarfs workloads in big data analytics. Institute of Computing Technology，Chinese Academy of Sciences，Beijing，China：Technical Report arXiv：1505.06872，2015

[23] Jolliffe I. Principal Component Analysis. New York，USA：John Wiley & Sons，2002

[24] MacQueen J. Some methods for classification and analysis of multi-variate observations//Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability. California，USA，1967：281-297

[25] Deng J，Dong W，Socher R，et al. ImageNet：A large-scale hierarchical image database//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Miami，USA，2009：248-255

[26] Ming Z，Luo C，Gao W，et al. BDGS：A scalable big data generator suite in big data benchmarking//Rabl T，Raghunath N，Poess M，et al，eds. Advancing Big Data Benchmarks. New York，USA：Springer International Publishing，2014：138-154

[27] Ming Zi-Jian. On Methods and Tools of Generating Big Data [M. S. dissertation]. University of Chinese Academy of Sciences，Beijing，2014(in Chinese)
(明子鉴. 基准大数据生成方法与工具研究[硕士学位论文]. 中国科学院大学，北京，2014)

[28] Blei D M，Ng A Y，Jordan M I. Latent Dirichlet allocation. The Journal of Machine Learning Research，2003，3：993-1022

[29] Leskovec J，Chakrabarti D，Kleinberg J，et al. Kronecker graphs：an approach to modeling networks. The Journal of Machine Learning Research，2010，11：985-1042

[30] Li M L，Sasanka R，Adve S V，et al. The ALPBench benchmark suite for complex multimedia applications//Proceedings of the IEEE International Workload Characterization Symposium. Austin，USA，2005：34-45

[31] Lowe D G. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision，2004，60(2)：91-110

[32] Hinton G E，Osindero S，Teh Y W. A fast learning algorithm for deep belief nets. Neural Computation，2006，18(7)：1527-1554

[33] Stone J E. An Efficient Library for Parallel Ray Tracing and Animation[M. S. dissertation]. University of Missouri-Rolla，Missouri，1998

[34] Felzenszwalb P F，Huttenlocher D P. Ecient graph-based image segmentation. International Journal of Computer Vision，2004，59(2)：167-181

[35] Uricar M，Franc V，Hlavac V. Detector of facial landmarks learned by the structured output SVM//Proceedings of the 7th International Conference on Computer Vision Theory and Applications. Rome，Italy，2012：547-556

[36] Thrasher A，Musgrave Z，Kachmark B，et al. Scaling up genome annotation with MAKER and work queue. International Journal of Bioinformatics Research and Applications，2014，10(4-5)：447-460

[37] Darling A，Carey L，Feng W. The design，implementation，and evaluation of mpiBLAST//Proceedings of the Cluster World. San Jose，Canada，2003：13-15

[38] Akoush S，Carata L，Sohan R，et al. MrLazy：Lazy runtime label propagation for MapReduce//Proceedings of the 6th USENIX Workshop on Hot Topics in Cloud Computing. Philadelphia，USA，2014

[39] Zou Q，Poremba M，He R，et al. Heterogeneous architecture design with emerging 3D and non-volatile memory technologies //Proceedings of the 20th Asia and South Pacific Design Automation Conference. Tokyo，Japan，2015：785-790

[40] Lee B C，Ipek E，Mutlu O，et al. Architecting phase change memory as a scalable DRAM alternative. ACM SIGARCH Computer Architecture News，2009，37(3)：2-13

[41] Poremba M，Xie Y. NVMain：An architectural-level main memory simulator for emerging non-volatile memories//Proceedings of the IEEE Computer Society Annual Symposium on VLSI Technology and Circuits. Honolulu，USA，2012：392-397

[42] Cheveresan R，Ramsay M，Feucht C，et al. Characteristics of workloads used in high performance and technical computing //Proceedings of the ACM 21st Annual International Conference on Supercomputing. Seattle，USA，2007：73-82

[43] Pan F，Yue Y，Xiong J，et al. I/O characterization of big data workloads in data centers//Proceedings of the 4th Workshop on Big Data Benchmarks，Performance Optimization，and Emerging Hardware. Salt Lake City，USA，2014：85-97

[44] Jia Z，Wang L，Zhan J，et al. Characterizing data analysis workloads in data centers//Proceedings of the IEEE International Symposium on Workload Characterization (IISWC). Portland，USA，2013：66-76

[45] Luszczek P R，Bailey D H，Dongarra J J，et al. The HPC Challenge (HPCC) benchmark suite//Proceedings of the 2006 ACM/IEEE Conference on Supercomputing. Tampa，USA，2006：213

ZHAN Jian-Feng, born in 1976, Ph. D., professor, Ph. D. supervisor. His main research interests include operating systems, data management, benchmarks, parallel and distributed systems.

GAO Wan-Ling, born in 1989, Ph. D. candidate. Her research interests include big data benchmark, big data analytics.

WANG Lei, born in 1976, Ph. D. candidate, senior engineer. His research interests include benchmark, parallel and distributed systems.

LI Jing-Wei, born in 1990, bachelor. His research interest is big data benchmark.

WEI Kai, born in 1981, senior engineer. His research interests include big data technology and industry standards.

LUO Chun-Jie, born in 1987, engineer. His research interests include big data benchmarking, big data learning.

HAN Rui, born in 1985, Ph. D. His research interests include cloud computing, parallel and distributed systems, and big data benchmark.

TIAN Xin-Hui, born in 1989, Ph. D. candidate. His research interests include big data benchmark, parallel and distributed systems.

JIANG Chun-Yu, born in 1987, engineer. His research interest is big data benchmark.

Background

With booming big data and corresponding systems, big data benchmarking becomes more and more important, as it provides yardsticks for measuring, comparing, and evaluating various big data systems. Many researchers from academia and industry are committed to explore the way to define a successful big data benchmark. However, the complexity, diversity and rapid evolution of big data systems make big data benchmarking very challenging. Existing big data benchmarking efforts focus on specific application types or software stacks, thus fail to cover diversity of workloads and real data sets. Different from previous work, we employ an incremental and iterative approach, covering not only broad application domains but also diverse workloads with different implementations and different real-world data sets. Following the above methodology, we propose benchmark specifications for five important application domains, including search engine, social networks, and electronic commerce, multimedia and bioinformatics, and release an open source big data benchmark suite—BigDataBench.

This work is supported by the Major Program of National Natural Science Foundation of China (Grant No. 61432006).