Computer Science and Application 计算机科学与应用, 2022, 12(3), 582-589
Published Online March 2022 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2022.123059

Hans汉斯

基于MapReduce的大数据并行分析与处理

张昕晨，王雅君，程胜明，冷峻宇，刘小奇

大连工业大学，机械工程与自动化学院，辽宁 大连

收稿日期：2022年2月13日；录用日期：2022年3月9日；发布日期：2022年3月16日

摘　要

针对传统分布式数据库架构存储和管理企业产品相关的大数据资源效率不高等问题，研究企业产品海量数据资源处理与并行分析计算，提出在Hadoop平台基础上基于MapReduce并行架构模型的数据并行分析与数据处理方法。通过对数据的优化存储布局，在MapReduce并行框架基础上，采用多通道数据融合特征提取技术实现产品大数据信息的提取和并行分析计算，提高了数据资源管理效率。实际验证表明和标准Hadoop方案比较，多通道数据融合并行特征提取算法执行时间为其34.8%，实现了产品大数据资源高效的组织和管理。

关键词

数据资源，并行处理，特征提取，MapReduce

Parallel Analysis and Processing of Big Data Based on MapReduce

Xinchen Zhang, Yajun Wang, Shengming Cheng, Junyu Leng, Xiaoqi Liu

School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian Liaoning

Received: Feb. 13th, 2022; accepted: Mar. 9th, 2022; published: Mar. 16th, 2022

Abstract

Aiming at the low efficiency of traditional distributed database architecture to store and manage big data resources related to enterprise products, the processing and parallel analysis and calculation of massive data resources of enterprise products are studied, and a parallel data analysis and data processing method based on MapReduce parallel architecture model based on Hadoop platform is proposed. By optimizing the storage layout of data, based on MapReduce parallel framework, multi-channel data fusion feature extraction technology is used to realize product big data information extraction and parallel analysis and calculation, improving the efficiency of data resource management. Actual verification shows that compared with the standard Hadoop

文章引用: 张昕晨, 王雅君, 程胜明, 冷峻宇, 刘小奇. 基于 MapReduce 的大数据并行分析与处理[J]. 计算机科学与应用, 2022, 12(3): 582-589. DOI: 10.12677/csa.2022.123059

scheme, the execution time of the multi-channel data fusion parallel feature extraction algorithm is 34.8%, which realizes the efficient organization and management of product big data resources.

Keywords

Data Resource, Data Parallel Processing, Feature Extraction, MapReduce

1. 引言

计算机的发展和网络通信技术日趋成熟，数据规模的增长在给人们带来便利生活的同时也让从大量数据中汲取有用信息变得困难，如何高效安全地对这些数据资源进行快速地访问、挖掘和分析是目前需要面对的重要问题[1] [2]。工业企业随着其规模的快速增长，纷纷加强了过程状态监测技术在产品生产、流通过程中的推广和应用，增加了智能监测设备的使用数量，同时这些智能监测设备中获取和传输的数据种类增多，各类数据数量呈几何级增长[3] [4] [5]。其中与产品相关的数据资源包含生产车间监测视频图像及产品相关数据及文档、物料跟踪数据、加工数据、生产流通数据等，其存在着数据资源规模大、种类多、来源不同且分散分布的特点[6] [7]。

本文针对上述数据资源管理与处理中存在的问题，在 Hadoop 平台下对产品海量数据资源应用 Hadoop MapReduce 框架进行海量数据并行分析与处理方法的研究，对 Hadoop 平台的数据划分策略、数据块规格调整方法进行了研究。通过对数据的优化存储布局，在 Hadoop MapReduce 并行框架基础上，实现产品海量数据的多源数据信息融合特征的提取和并行分析计算。

2. 数据融合特征并行提取研究

2.1. 多通道数据融合特征并行提取算法

通过对数据的优化存储布局[8]，在 Hadoop MapReduce 并行框架基础上，实现产品海量数据的多源数据信息融合特征的提取和并行分析计算。学术界采用多种方法在不同应用背景下对数据序列间的动态相互关系进行评价[9]。文献[10]提出多规格多变量样本熵(Multiscale Multivariate Entropy, MSMVE)分析方法，对多通道时间序列的动态相互关系根据其内在非线性耦合特征从复杂度、互预测性和长时相关性等多个角度进行评价。MSMVE 分析方法目前已应用在物理、生理等学科多种领域中[11]，具有其潜在的理论意义和实际应用价值。

以 6 通道同步采集的质量检测数据应用 MSMVE 算法进行数据融合特征提取为例，算法运行速度随着数据量增加而降低。为提高海量数据量时算法的数据融合特征提取效率，设计基于 Hadoop 平台 CMCHA 算法的并行化多规格多变量样本熵 MSMVE 算法。

同步采集的 6 通道过程信息独立存储在 6 个文件中，为进行数据并行分析，将数据分段上传存储至标准 Hadoop 平台的 HDFS，每段数据带有时间戳，随机分布到多个数据节点上。在一次 MSMVE 计算任务中使用的多通道信号，由于未考虑数据相关性，在数据分布存储时可能被分配到不同的节点，所以并行化 MSMVE 算法采用的计算模式为：数据过滤在映射过程完成，并将各段信息通过网络发送给约减端，求解在约减过程完成，数据分布存储和数据并行计算的过程如图 1 所示。

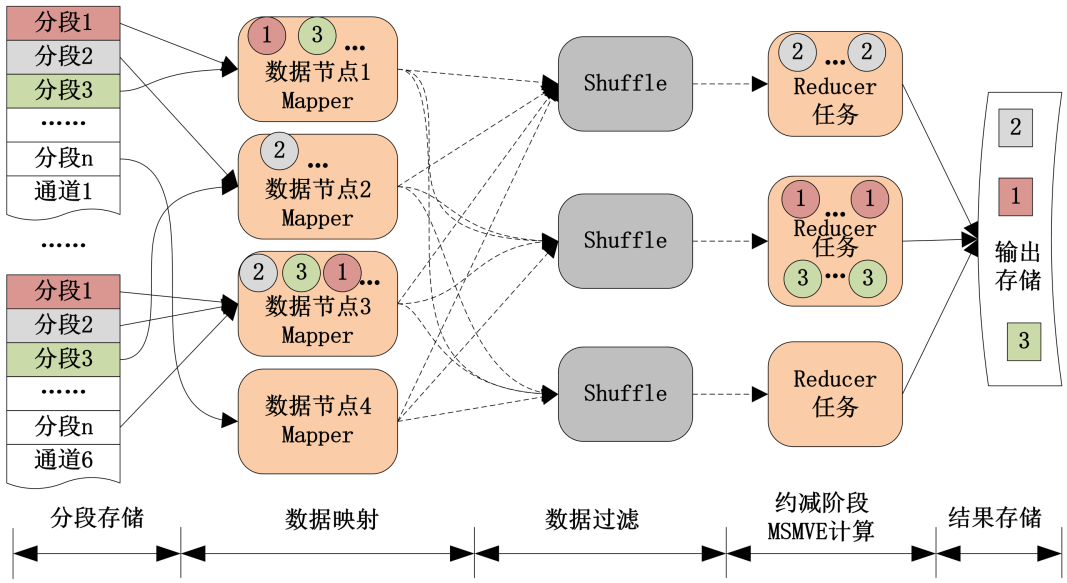

Figure 1. Data distribution on standard Hadoop platform and data feature extraction on reduce process
图 1. 标准 Hadoop 平台的数据分布及约减端数据特征提取流程

特征提取过程中首先将各通道文件均划分成图 1 中的分段 1，分段 2，…，分段 n，这些数据分段分布存储在多个数据节点。按相同时间戳在数据过滤阶段同步采集信号片段(图中用相同编号的小圆表示)，这些信号片段会在约减阶段的同一个 MSMVE 的计算任务中使用。MSMVE 计算结果用带编号的小方格表示并输出到 HDFS 中保存。

对上述多通道数据采用 CMCHA 算法重新进行数据分布优化，此时将采集时间戳作为关键字并考虑数据的时间相关性，计算 Hash 存储位置。通过数据优化分布聚集同步数据，并在映射阶段进行 MSMVE 并行计算，基于 Hadoop 的数据优化分布和映射端 MSMVE 并行计算过程如图 2 所示。

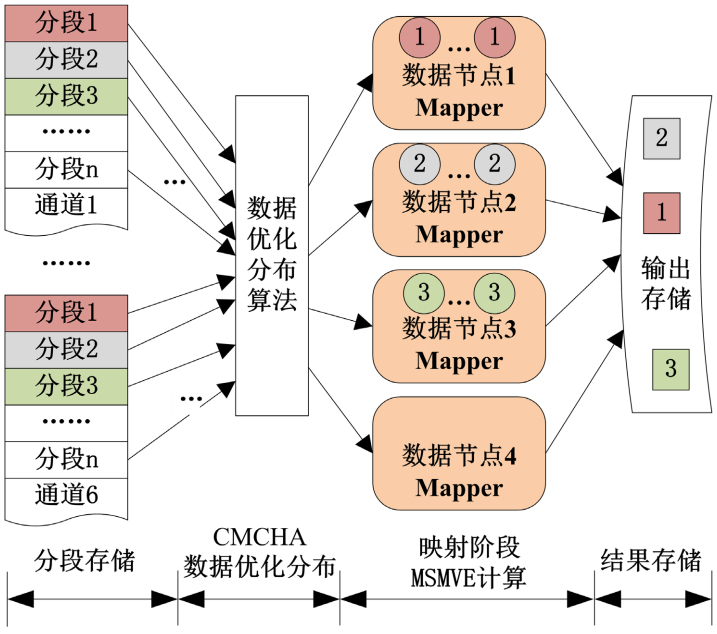

Figure 2. Data distribution optimization with CMCHA and map feature extraction
图 2. 数据布局的优化及映射端数据特征提取流程

基于 CMCHA 算法的映射端 MSMVE 并行特征提取算法流程：1) 根据 MSMVE 计算任务时间，通过过滤数据优化数据分布，将不满足计算时间条件的数据消除；2) 以数据采集时间戳为主连接组键来标记每条数据记录；3) 按连接组键划分数据记录，具有相同属性值的记录为一组，调用 MSMVE 算法，执行计算任务；4) 将 MSMVE 计算结果输出到分布式文件系统。

MSMVE 算法执行步骤：

1) 设 p 维(通道)时间序列为 $\{x_{k,i}\}_{i=1}^{N}$，$k=1,2,\cdots,p$，每维时间序列有 N 个点。ε 为预先给定的尺度因子，对其构建时间序列 $\{y_{k,j}^{\varepsilon}\}$ 如下

$$y_{k,j}^{\varepsilon}=\frac{1}{\varepsilon}\sum_{i=(j-1)\varepsilon+1}^{j\varepsilon}x_{k,i}\ ,\quad k=1,2,\cdots,p\ ,\quad 1\le j\le\frac{N}{\varepsilon} \tag{1}$$

该事件序列具有多变量粗粒化等特征，当尺度因子 $\varepsilon=1$ 时，$\{y_{k,j}^{\varepsilon}\}$ 就是原始时间序列。

2) 根据多变量时间序列模型[12]，预先设定参数嵌入矢量 $M=[m_1,m_2,\cdots,m_p]$，时间延迟向量 $\tau=[\tau_1,\tau_2,\cdots,\tau_p]$，利用序列 $\{y_{k,j}^{\varepsilon}\}$ 构建 $(N-n)$ 个复合延迟向量 $Y(i)\left(m=\sum_{k=1}^{p}m_k\right)$，那么

$$Y_m(i)=\left[y_{1,i},y_{1,i+\tau_1},\cdots,y_{1,i+(m_1-1)\tau_1},y_{2,i},y_{2,i+\tau_2},\cdots,y_{2,i+(m_2-1)\tau_2},\cdots,y_{p,i},y_{p,i+\tau_p},\cdots,y_{p,i+(m_p-1)\tau_p}\right]$$

$$i=1,2,\cdots,N-n\ ;\quad n=\max\{M\}\times\max\{\tau\} \tag{2}$$

3) 定义 $Y_m(i)$ 和 $Y_m(j)$ 间的距离

$$d[Y_m(i),Y_m(j)]=\max_{l=1,2,\cdots,mn}\{|x(i+l-1)-x(j+l-1)|\} \tag{3}$$

4) 给定的阀值 r，对每个 i 值计算事件 P_i：$d[Y_m(i),Y_m(j)]\le r(j\ne i)$ 出现的概率计算公式为 $B_i^m(r)=P_i/(N-n-1)$，表示所有 $Y_m(j)(i\ne j)$ 和 $Y_m(i)$ 的关联程度，也表示时间序列 $\{Y_m(j)\}$ 的规律性。

5) 求概率 $B^m(r)$ 所有 i 的平局值

$$B^m(r)=\frac{1}{N-n}\sum_{i=1}^{N-n}B_i^m(r) \tag{4}$$

6) 扩展 2)中 m 为 $m+1$，重复 3)到 5)得到 $B^{m+1}(r)$。

7) 计算多规格多变量样本熵

$$MSMVE(M,\tau,r,N)=-\ln\frac{B^{m+1}(r)}{B^m(r)} \tag{5}$$

2.2. 算法性能分析

按固定尺寸规格对每个数据文件进行数据分块，在未进行优化的标准 Hadoop 平台上，随机选择数据节点按照数据块进行存储分布在不同数据节点上，在对这些数据进行分析和应用 MapReduce 框架进行并行计算的过程中，为减少各数据节点之间的数据通信，根据跟踪过程数据间的相互关系，CMCHA 算法进行数据分布优化并聚集相关数据。在未进行数据存储优化时，并行计算分析程序运行时的数据节点间数据通信量分析如下。

首先做如下假设：

假设 1：Hadoop 集群规模为 N，并将并行分析任务分解，形成 M 个子任务 Map_{ij} (其中 i 为任务序号，$1\le i\le M$，j 为所在节点序号，$1\le j\le N$)；

假设 2：单个数据块规格不变，为 d 字节，数据节点 j 上子任务 Map_{ij} 在执行时使用 a_i 个数据块，R 个数据副本。

MapReduce 的任务分配器在程序运行过程中会尽可能确保数据的本地性，因此 Map_{ij} 所需的数据块至少会有 1 个数据块在节点 j，在 Hadoop 平台，相同数据节点不会被分配放置同一个数据块的多个副本，这样子任务 Map_{ij} 所需的数据块只有 1 个在本地的概率为

$$p_{i,1}=\frac{a_i P_{N-1}^{R-1} R\left(P_{N-1}^{R}\right)^{a_i-1}}{\left(P_N^R\right)^{a_i}}=\frac{a_i R}{N^{a_i}\left(N-R+1\right)^{a_i-1}} \tag{6}$$

此时需要拉取 a_i-1 个数据块；计算 Map_{ij} 所需的数据块中有 k 个在本地的概率为

$$p_{i,k}=\frac{C_{a_i}^{k}\left(P_{N-1}^{R-1}\right)^{k} R^k\left(P_{N-1}^{R}\right)^{a_i-k}}{\left(P_N^R\right)^{a_i}}=\frac{C_{a_i}^{k} R^k}{N^{a_i}\left(N-R+1\right)^{a_i-k}} \tag{7}$$

则需要拉取的数据块数量为 a_i-k； Map_{ij} 所需要的 a_i 个数据块全部在本地的概率为 $p_{i,all}=\left(R/N\right)^{a_i}$，则不需要拉取其他数据节点上的数据，数据通信量为 0。

综上，可将 Map_{ij} 执行时的数据通信量即数据块数量 D_i 表示为概率平均值：

$$\begin{aligned} D_i &= p_{i,1}\left(a-1\right)+p_{i,2}\left(a-2\right)+\cdots+p_{i,k}\left(a-k\right)+\cdots+p_{i,a_i-1} \\ &=\sum_{k=1}^{a_i}\left[p_{i,k}\left(a-k\right)\right]=\sum_{k=1}^{a_i}\left[\frac{C_{a_i}^{k} R^k}{N^{a_i}\left(N-R+1\right)^{a_i-k}}\left(a_i-k\right)\right] \end{aligned} \tag{8}$$

执行任务时总的数据通信量表示为：

$$D=\sum_{i=1}^{M}\sum_{k=1}^{a_i}\left[\frac{C_{a_i}^{k} R^k}{N^{a_i}\left(N-R+1\right)^{a_i-k}}\left(a_i-k\right)d\right] \tag{9}$$

子任务 Map_{ij} 在数据通信过程中从不同的数据节点拉取数据的通信网络带宽不同，在 Hadoop 集群网络拓扑结构中存在 3 种数据节点间通信的网络带宽，分别为 c_1，c_2，c_3，对应的节点位置分别为同机架不同节点，同机房不同机架节点和不同机房节点。当子任务 Map_{ij} 执行时所需数据块不在本地时也存在 3 种情况，其概率分别为 p_1，p_2，p_3，对应于数据块与子任务 Map_{ij} 在同机架，在同机房不同机架和在不同机房三种位置情况，计算数据节点间通信网络带宽为 $C_{avr}=\sum_{j=1}^{3} p_i c_i$，为以上三种情况的平均值。并行算法任务执行时间为

$$T=\frac{D}{C_{avr}}=\frac{\sum_{i=1}^{M}\sum_{k=1}^{a_i}\left[\frac{C_{a_i}^{k} R^k}{N^{a_i}\left(N-R+1\right)^{a_i-k}}\left(a_i-k\right)d\right]}{\sum_{j=1}^{3} p_i c_i} \tag{10}$$

从式(10)可以看出，Hadoop 集群节点数量 N、任务分解的子任务数量 M，数据存放的副本数量 R、数据块规格 d、数据本地性概率 p_i 和节点间的通信带宽 c_i 是影响并行算法执行性能的主要因素。当数据的分布随机时，相关数据的聚集性随着 Hadoop 集群节点数量 N 的增加而变差，随之而来的是数据通信量增加，算法执行效率降低。

数据本地性概率通过数据的多副本策略可以得到提高，从而提高算法运行效率，但数据的副本数 R 受系统数据容量等性能的限制不能设置过大，通常设定为超过 3 个副本。由执行任务时总的数据通信量

计算式(9)可以看出，数据通信量 D 与数据块规格成正比，而从前述分析中可知数据块规格与数据传输率成正比，因此通常应综合考虑数据传输率、负载平衡等多种因素设定数据块规格。因此，在 Hadoop 集群节点数量 N 不断增长，而又不能随意调节数据存放的副本数量和数据块规格时，要提高数据并行算法任务执行性能的有效途径是通过数据存储分布优化算法，按照规则聚集海洋食品跟踪过程海量数据资源的相关数据，使数据处理在本地进行。

3. 验证分析

利用文献[8]中的实验平台和存储数据，为测试数据存储分布优化后 MSMVE 并行特征提取算法的性能，将前述针对产品大数据连接算法和基于标准 Hadoop 平台的连接算法进行分析比对验证。测试使用的数据集独立存储在 6 个文件中，单个文件含有 81,920 个采样点，大小约为 6.5 MB，算法是针对海量数据而提出的，但是由于受样本数量限制，现有数据数量无法满足算法性能测试需求，为确保验证算法在处理海量数据时的性能，人为加大数据量，复制现有数据集，使单个文件大小增加到 650 MB，6 个文件数据集整体达到 3900 MB。

1) 数据上传运行时间变化趋势

采用 CHCMA 算法和随机选择数据分布策略，分别将上述数据集从本地上传至 Hadoop 的分布式文件系统，以验证数据优化存储策略对数据上传速度的影响。数据上传过程中增加数据集规模，见文件数量从 1 个文件逐步增加到 6 个文件，这样文件大小从 650 M 增加到 3900 M，数据上传运行时间变化趋势采用图 3 的折线图表示。从图 3 的实验数据表明，随着数据规模的逐步增长，数据传输率保持不变，传输时间均为线性增加。数据优化存储策略对上传效率影响较小，上传时间比随机分布情况下运行时间略有加长，数据传输率下降微小。CHCMA 算法和标准 Hadoop 平台下数据传输率平均值分别为 20.1 M/s 和 22.1 M/s。数据传输率略有下降主要是因为标准 Hadoop 平台采用的是随机分布策略，而优化数据布局需要额外的处理时间进行数据节点的选择。

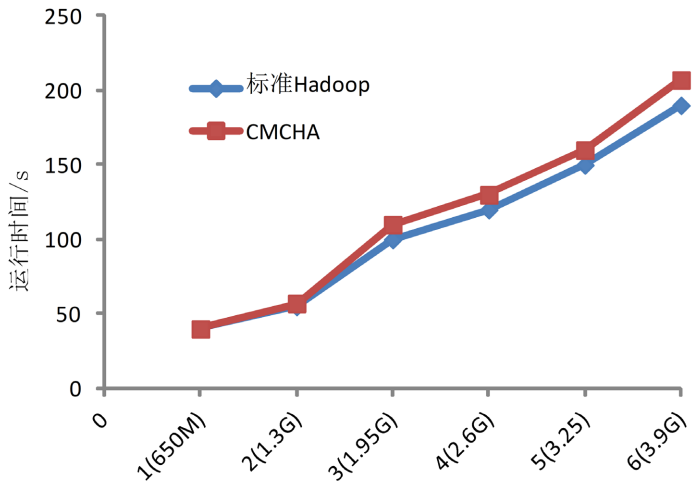

Figure 3. Execution time and its variation trend of data upload
图 3. 两种算法下数据上传运行时间变化趋势比较

2) MSMVE 并行计算时间变化趋势

主要验证基于 CHCMA 的映射端 MSMVE 并行计算时间变化趋势。进行过程数据采集，选取其中 5210 个采样点作为样本数据长度，设置多尺度因子 ε 分别为 8 和 15，嵌入维数向量为 $M[2,2,2,2,2,2]$，时间延迟向量 $\tau[1,1,1,1,1,1]$，阈值参数 $r=0.45$，计算 MSMVE。数据集共包含约 3900 MB 的 1600 条样本数据。

增加实验样本数据量，从 200 条递增到 1600 条，对应的运行时间变动采用图 4 的曲线表示。根据图中时间变化趋势可知，随着数据规模的增长 MSMVE 的求解运行时间逐步平缓增长，数据处理效率增加，这表明 MSMVE 并行特征提取算法适合处理较大规模的数据。在映射过程中完成 MSMVE 计算过程，网络通信带宽对整体运行时间的影响基本很小，MSMVE 并行算法性能稳定。

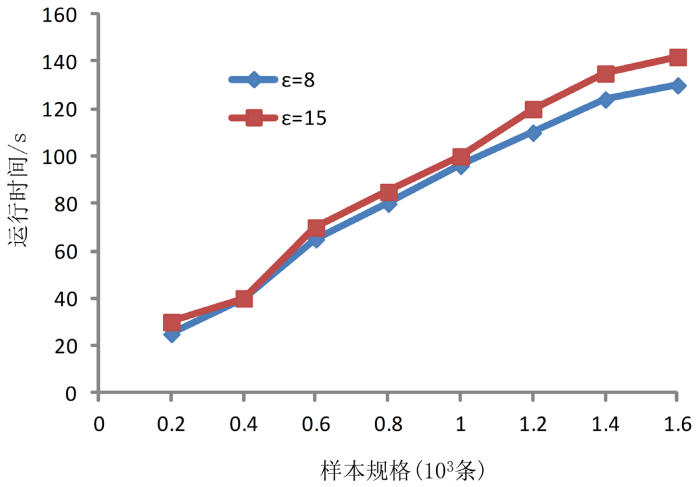

Figure 4. Execution time of Map-MSMVE
图 4. MSMVE 并行计算时间变化趋势

3) 特征提取算法运行时间对比

验证过程选取由 200 条增加到 1600 条的不同规模样本数据集，进行两种特征提取算法运行时间的对比，由于 CMCHA 的数据存储优化分布省去了许多数据传输和任务约减过程，基于 CMCHA 的特征提取算法的运行时间比基于标准 Hadoop 的算法运行时间节约了 75%左右，如图 5 所示。

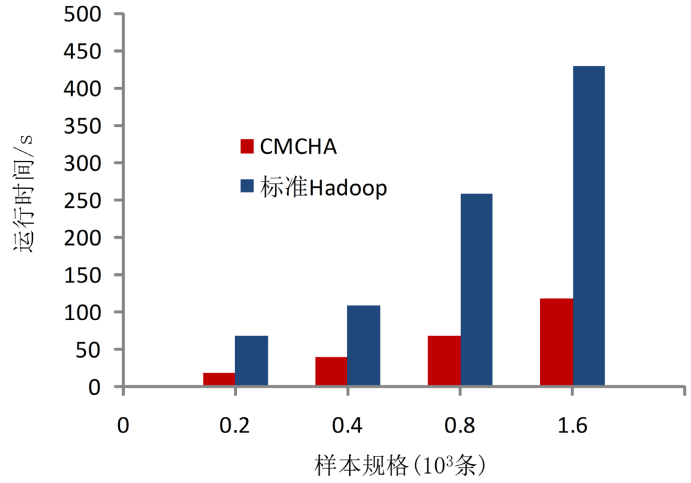

Figure 5. Execution time comparison based on 2 algorithms
图 5. 特征提取算法运行时间比较

4. 结论

针对种类多容量大的产品大数据资源，本文研究了基于 Hadoop 平台的海量数据组织与管理方法，采用分布式、分层结构的存储优化和并行处理等技术，提出了多副本一致性 Hash 数据存储算法，基于该算

法设计实现了 Hadoop 平台下基于 MapReduce 并行编程框架实现多通道数据融合特征并行提取算法。通过测试证明通过数据的存储分布优化，算例的运行速度明显加快，和标准 Hadoop 方案比较，多通道数据融合并行特征提取算法执行时间为其 34.8%。

参考文献

[1] 张国华, 叶苗, 王自然, 周婷婷. 大数据 Hadoop 框架核心技术对比与实现[J]. 实验室研究与探索, 2021, 40(2): 145-148+176.

[2] 王艳, 蒋义然, 刘永立. 基于 Hadoop 的大数据处理技术及发展[J]. 信息记录材料, 2020, 21(11): 146-147.

[3] 李善青, 郑彦宁, 赵辉, 等. 大数据背景下科学元数据的重要问题研究[J]. 科技管理研究, 2019, 18(1): 184-188.

[4] 李联辉, 尹冠飞, 莫蓉. 面向航空发动机装配过程的信息追溯与过程监控[J]. 计算机集成制造系统, 2018, 22(12): 2986-3000.

[5] 李青, 冯丹, 梅正朋. 飞机使用寿命周期构型管理和追溯[J]. 计算机集成制造系统, 2016, 22(2): 476-481.

[6] 程学旗, 靳小龙, 王元卓, 等. 大数据系统和分析技术综述[J]. 软件学报, 2014, 25(9): 1889-1908.

[7] 孟小峰, 慈祥. 大数据管理: 概念、技术与挑战[J]. 计算机研究与发展, 2013, 50(1): 146-149.

[8] 王耐东, 王雅君, 张昕晨, 等. 基于 Hadoop 的产品大数据分布式存储优化[J]. 计算机科学与应用, 2021, 11(5): 1503-1511.

[9] 李鹏, 刘澄玉, 李丽萍, 等. 多尺度多变量模糊熵分析[J]. 物理学报, 2013, 62(12): 120512-12020.

[10] Ahmed, M.U. and Mandic, D.P. (2012) Multivariate Multiscale Entropy Analysis. *IEEE Signal Processing Letters*, **19**, 91-94. https://doi.org/10.1109/LSP.2011.2180713

[11] Morabito, F.C., Labate, D., La Foresta, F., *et al*. (2012) Multivariate Multi-Scale Permutation Entropy for Complexity Analysis of Alzheimer's Disease EEG. *Entropy*, **14,** 1186-1202. https://doi.org/10.3390/e14071186

[12] Cao, L., Mees, A. and Judd, K. (1998) Dynamics from Multivariate Time Series. *Physica D*: *Nonlinear Phenomena*, **121**, 75-88. https://doi.org/10.1016/S0167-2789(98)00151-1