大规模数据挖掘的并行算法优化与性能评估

张海英　哈丽旦·塔什
新疆轻工职业技术学院

摘　要：并行计算在大规模数据挖掘中可提升计算效率并降低资源消耗，本研究围绕数据挖掘算法的并行化，分析主流计算框架，探讨任务划分、数据存储与通信优化策略，并评估计算复杂度、扩展性及资源利用率。结果表明，合理的任务分配提升负载均衡性，高效的数据存储优化访问效率，低延迟的通信机制改善计算性能，从而增强数据挖掘能力。研究结论可为高效并行计算框架构建及数据挖掘优化提供参考。

关键词：并行算法优化；大规模数据挖掘；性能评估

引言

大规模数据挖掘在金融分析、医疗诊断、智能制造等领域具有广泛应用，高效计算能力决定系统性能。传统数据挖掘算法受计算复杂度、存储开销和执行效率限制，难以满足超大规模数据处理需求。并行计算架构的发展推动算法优化，分布式存储与多核处理框架显著提升计算效率。任务划分、负载均衡、数据存储与通信机制的优化直接影响算法扩展性与稳定性，性能评估需结合计算复杂度分析、资源利用率测评及实验验证，以增强数据挖掘能力。

1 并行数据挖掘算法概述

1.1 并行计算框架对比

并行计算框架决定数据挖掘任务的计算效率，不同架构在数据存储、任务调度与计算资源管理方面存在差异。共享内存架构依赖单机多核计算环境，数据访问速度较快但计算规模受物理资源限制，分布式计算架构基于多节点协同处理，计算能力可扩展但受数据传输机制影响较大。任务调度模式直接影响计算效率，静态调度在任务开始前分配计算资源，适用于负载均衡性较高的计算环境，动态调度根据计算节点状态调整任务分配，提高计算资源利用率。存储管理方式决定数据访问效率，集中式存储数据一致性较高但访问吞吐量受限，分布式存储通过数据分片减少访问冲突，提高计算吞吐量但增加一致性维护成本。计算资源分配方式影响任务执行效率，任务并行模式将计算任务划分后分配至多个计算单元独立执行，数据并行模式通过划分数据集并行处理同一计算任务，混合模式结合任务并行与数据并行，提高计算资源的整体利用率。

1.2 并行计算模式分析

并行计算模式决定数据挖掘算法的执行方式，不同模式适用于不同的数据特征与计算需求。任务划分方式影响计算负载均衡性，细粒度划分减少单任务计算时间但通信开销较大，粗粒度划分降低通信开销但可能导致负载不均衡。

数据传输方式决定计算节点间的数据交互效率，消息传递模式依赖进程间通信进行数据交换，适用于计算任务之间数据共享较少的环境，共享内存模式通过内存映射技术实现数据交互，适用于计算任务间数据共享需求较高的计算场景。计算同步策略影响算法执行顺序，全局同步策略在所有计算任务完成后进行数据聚合，适用于计算任务之间存在强依赖关系的计算模式，局部同步策略在部分计算任务完成后进行数据更新，适用于计算任务之间依赖关系较弱的并行计算模式，提高计算吞吐量。

计算任务的并行化优化涉及负载均衡与计算复杂度控制，假设计算任务总量为 f(n)，计算节点数量为 p，单节点的计算复杂度可表示为：

$$O\left(\frac{f(n)}{p}\right)$$

任务划分策略优化目标为最小化计算负载不均衡程度，设计算节点 i 的计算时间为 T_i，负载均衡目标函数表达为：

$$\max(T_i) - \min(T_i) \to 0$$

该优化策略减少计算节点的等待时间，提高数据挖掘任务的整体计算效率。

2 并行算法优化策略

2.1 任务划分与负载均衡

任务划分是并行计算中影响计算效率的核心环节，合理的划分方式能够降低计算节点间的负载差异，提高计算资源利用率。划分策略主要包括静态划分和动态划分，静态划分在任务执行前分配计算资源，适用于计算任务规模稳定、负载均衡性较好的场景，动态划分在计算过程中根据计算节点的实时负载调整任务分配，适用于计算任务规模不均或计算资源动态变化的环境。

任务划分的粒度影响计算节点的计算开销，细粒度划分减少单个计算节点的计算负担，但会导致任务间通信开销增加，粗粒度划分减少通信开销，但可能导致计算负载不均衡。任务粒度的选择依赖计算任务的复杂度和数据分布特性，适当的划分策略可以优化计算吞吐率，提高并行算法的执行效率。调度策略直接决定负载均衡性，集中调度策略依赖中央调度单元进行任务分配，全局控制任务执行流程，适用于任务依赖性强的计算模式，分布式调度策略由各计算节点独立管理任务分配，适用于计算任务相对独立的环境。负载均衡优化策略可分为静态负载均衡和动态负载均衡，静态负载均衡在任务执行前进行均衡任务分配，适用于计算任务执行时间可预测的情况，动态负载均衡根据计算节点的实时负载调整任务执行顺序，适用于计算负载变化较大的计算场景。

执行时间的不均衡性影响整体计算吞吐量，假设计算任务集合 T 被划分为 p 个计算节点，每个节点 i 的计算任务量为 t_i，计算任务的均衡性优化目标表达为：

$$\sum_{i=1}^{p}\left|t_i-\frac{T}{p}\right|\rightarrow \min$$

该优化目标减少计算任务的负载偏差，使计算节点的任务执行时间尽可能接近理论均值，避免部分计算节点长时间计算而其他计算节点资源空闲的问题。计算任务的不均衡性会导致整体计算时间由最慢的计算节点决定，影响并行计算的吞吐率，优化任务划分策略能够减少计算节点的执行时间差异，提高任务调度的均衡性，使得所有计算节点能够同步完成计算任务，缩短数据挖掘任务的总执行时间，提高并行计算的整体计算效率。

任务迁移策略可提升计算资源利用率，在计算任务执行过程中，当计算节点出现负载不均的情况，系统可根据计算节点的实时负载信息调整任务分配方式，将高负载计算节点的部分计算任务迁移至低负载计算节点，减少计算任务的执行时间偏差，提高计算资源的整体利用率。任务迁移的成本依赖于任务大小、数据传输带宽以及计算节点之间的通信延迟，合理的任务迁移策略需要平衡任务迁移开销与负载均衡的收益。

2.2 数据存储与计算节点通信

数据存储方式决定数据访问效率，不同存储架构影响计算任务的数据交互速度。集中式存储依赖单一存储节点管理数据，存储一致性较高但访问延迟受存储节点负载影响较大，分布式存储通过数据分片机制减少存储访问延迟，提高并行计算的存储吞吐量。数据传输优化策略依赖数据访问模式，分块数据传输模式将大规模数据集拆分为多个数据块，提高数据流的并行吞吐量，流式数据传输模式基于数据流逐步加载数据，提高计算任务的连续性。计算节点通信机制影响并行计算效率，消息传递模式依赖进程间通信进行数据交换，适用于计算任务间数据共享需求较低的计算模式，共享内存模式通过直接内存访问减少数据交换开销，适用于数据交互频繁的计算环境。

通信延迟优化策略依赖数据压缩与传输优化，设计算节点 i 与计算节点 j 之间的数据交换量为 d_{ij}，传输带宽为 B，数据压缩率为 α，数据传输时间优化目标表达为：

$$\sum_{i=1}^{p}\sum_{j=1}^{p}\frac{d_{ij}\cdot(1-\alpha)}{B}\rightarrow \min$$

该优化目标减少数据传输延迟，提高计算节点之间的数据交换速率。在大规模数据挖掘任务中，计算节点需要频繁进行数据交互，高速数据传输能够减少计算节点之间的等待时间，提高数据吞吐量，避免因通信延迟导致的计算资源空闲，提高计算任务的执行效率。通过优化数据压缩率和数据分块策略，可以减少计算节点间的通信负担，优化计算节点的负载均衡性，提高数据挖掘任务的整体计算性能。

3 并行算法性能评估

3.1 计算复杂度与扩展性分析

计算复杂度是衡量并行数据挖掘算法效率的核心指标，不同算法的时间复杂度和空间复杂度决定计算资源的消耗水平。并行算法的计算复杂度由任务划分策略、通信开销和数据存储方式决定，总计算时间可分解为计算任务执行时

间、数据传输时间和存储访问时间。

计算任务执行时间受任务划分粒度影响，细粒度划分可减少单个计算节点的计算负担，但会增加通信开销，粗粒度划分可降低通信负担，但可能导致计算负载不均衡。设计算任务总量为 T，计算节点数为 p，单节点计算复杂度可表示为

$$O\left(\frac{T}{p}+C_{comm}+C_{mem}\right)$$

其中 C_{comm} 为通信开销，C_{mem} 为存储访问时间，该表达式说明计算复杂度受计算任务规模、通信延迟和存储访问效率影响，优化任务划分策略可降低单节点计算负载，提高计算吞吐率。

扩展性是衡量并行算法计算能力随计算节点数增加而提升的程度，理想扩展模型中计算速度应随计算节点数呈线性增长，但实际计算过程中受通信开销、数据存储访问延迟和负载均衡影响，计算效率无法无限扩展。设单节点执行时间为 T_s，p 个计算节点的总执行时间 T_p 可表示为：

$$T_p=\frac{T_s}{p}+C_{comm}(p)+C_{mem}(p)$$

其中 C_{comm} (p) 与 C_{mem} (p) 随计算节点数增长而增加。优化通信延迟和存储访问策略可提升算法扩展性，使计算时间更接近理想状态，提高数据挖掘任务的并行计算效率。

3.2 负载均衡与资源利用率评估

负载均衡影响计算节点的计算资源利用率，任务划分策略决定负载均衡性，计算资源的动态调度优化计算吞吐量，任务调度不均衡会导致部分计算节点长时间处于高负载运行状态，而其他计算节点资源未充分利用，计算任务完成时间由最慢的计算节点决定。

计算资源利用率衡量计算资源的有效使用程度，不同任务划分策略影响计算资源的使用效率。设计算节点 i 的计算时间为 T_i，计算资源的利用率可表示为：

$$U=\frac{\sum_{i=1}^{p}T_i}{p\cdot\max(T_i)}$$

该表达式描述计算任务在各个计算节点上的均衡性，提高任务均衡性可提升计算资源利用率，使计算任务的执行时间尽可能接近理论均值，避免计算节点资源空闲或计算任务过载问题。计算节点的负载均衡影响并行计算的整体吞吐率，任务迁移策略可动态调整计算任务，使计算负载在计算节点之间均匀分布，负载均衡优化策略可减少任务执行时间的偏差，提高计算任务的整体执行效率，使计算资源利用率达到最优状态。计算资源的调度优化需综合考虑计算任务的特性，合理的任务划分策略可减少负载不均衡带来的性能损失，提高计算节点的资源利用率。优化任务划分、存储访问和数据通信策略可提升计算负载均衡性，使并行算法在大规模数据挖掘任务中实现更高效的计算能力。

4 结语

大规模数据挖掘的计算效率受并行算法优化策略影响，合理的计算框架选择、任务划分方式和数据存储管理决定计算资源的利用率和负载均衡性。优化任务划分策略可降低计算负载不均衡，提高计算吞吐率，改进数据传输和存储管理可减少通信延迟，提升整体计算性能。计算复杂度与扩展性分析表明，优化计算资源调度可提高计算效率，负载均衡评估结果说明，优化任务分配可减少计算资源浪费，提升数据挖掘任务的执行能力。结合计算复杂度、扩展性与负载均衡优化，可进一步提升并行数据挖掘的计算性能，推动并行计算技术在大规模数据处理中的应用。

参考文献：

[1] 石杰 . 基于云计算环境的数据挖掘算法研究 [J]. 电子技术与软件工程 ,2023(4):233-236.

[2] 陈叶旺 , 曹海露 , 陈谊 , 等 . 面向大规模数据的 DBSCAN 加速算法综述 [J]. 计算机研究与发展 ,2023,60(9):2028-2047.

[3] 程红阳 , 叶青 . 基于数据挖掘的学习效果评估算法设计 [J]. 电子设计工程 ,2022,30(19):15-18+25.

基金项目：课题名称：基于关联规则的数据挖掘算法研究，编号：2024HT603。