面向成本收益协同优化的高质量数据准备方法

崔文萱[1]，丁小欧[1]，程海峰[1]，钱泽凯[1]，李勇[1]，王宏志[1]，王晨[2]
1. 哈尔滨工业大学，黑龙江 哈尔滨 150001；
2. 清华大学大数据系统软件国家工程实验室，北京 100084

摘要

数据治理是保障数据质量与安全、促进数据高效流通与价值释放的关键基础。然而目前实际机器学习（machine learning，ML）模型训练场景下，存在数据准备环节资源利用效率低、成本不可控及预算约束下优化不足的问题。针对以上问题，构建了一种面向成本收益协同优化的数据准备方法。采用数学规划方法，将数据预处理、清洗、转换等操作统一纳入成本建模与决策框架，设计面向整体管道的成本收益分析规划方法。实验结果表明，该方法在有限预算条件下可实现更优的成本效益权衡，提升数据准备过程的整体性能与性价比。研究认为，将成本收益分析和优化的思想纳入到数据准备优化策略，有助于实现资源的高效配置，并为复杂数据处理场景中的成本控制与性能优化提供参考。

关键词

数据治理；数据准备；资源调度；成本收益

中图分类号：　　文献标志码：A　　doi：10.11959/j.issn.2096–0271.2026xxx

A cost-benefit collaborative optimization method for high-quality data preparation

CUI Wenxuan[1], DING Xiaoou[1], CHENG Haifeng[1], QIAN Zekai[1], LI Yong[1], WANG Hongzhi[1], WANG Chen[2]
1. School of Computer Science and Technology, Harbin Institute of Technology, Harbin；
2. National Engineering Research Center for Big Data Software, Tsinghua University

Abstract

Data governance is a fundamental basis for ensuring data quality and security, promoting efficient data circulation, and realizing data value. However, in practical machine learning (ML) model training scenarios, data preparation is still constrained by low resource utilization efficiency, uncontrollable costs, and insufficient optimization under budget constraints. To address these problems, a data preparation method oriented toward cost-benefit collaborative optimization was proposed. Mathematical programming was adopted to incorporate data preprocessing, cleaning,

transformation, and other operations into a unified cost modeling and decision-making framework, and a cost-benefit analysis and planning method for the overall pipeline was designed. Experimental results showed that the proposed method achieved a better trade-off between cost and performance under limited budget conditions, thereby improving the overall performance and cost-effectiveness of the data preparation process. The results indicate that incorporating cost-benefit analysis and optimization into data preparation strategies facilitates efficient resource allocation and provides useful support for cost control and performance optimization in complex data processing scenarios.

Key words
data governance, data preparation, resource scheduling, cost–benefit

0 引言

数据治理是保障数据质量与安全、促进数据高效流通与价值释放的关键基础[1][2]。随着数据要素流通和数据社会化进程不断深化，数据治理已逐渐从组织内部的数据质量管理扩展到跨组织、跨领域的数据流通安全治理与价值释放过程[3]。在此背景下，数据准备在机器学习等数据驱动应用中的重要性愈发凸显，尤其体现在对数据质量的系统性保障与优化上。作为数据驱动技术的典型代表，机器学习模型的训练效果对训练数据的质量有很强的依赖性。 而真实数据通常存在格式不一致、字段缺失、异常值和噪声等多种数据质量问题，因此需要针对各类问题展开数据清洗与准备。随着ML的快速应用发展，自动化数据准备工具逐渐替代部分人工操作，显著提升了处理效率。然而目前实际模型训练场景下，数据准备仍耗费较高成本：一方面，许多环节仍依赖人工处理，导致时间与人力开销[4]；另一方面，不同自动化算法在处理效果、资源消耗和适用场景上存在差异，因此算法选择、组合与决策过程本身也会产生一定的额外开销。因此，如何有效控制数据准备成本仍是实际ML场景中的不容忽视的问题。

数据准备涵盖规则、统计、机器学习及深度学习等多类实现方法，不同方法在处理效果与适用场景上存在差异，因此需要进行合理的选择与编排。现有研究已提出多种自动化管道优化框架。Learn2Clean[5]面向标准化、特征选择及多类清洗操作进行强化学习决策；HAIPipe[6]结合人工经验与人工智能（artificial intelligence，AI）生成实现协同编排；DiffPrep[7]将离散管道选择转化为可微优化问题；CtxPipe[8]进一步利用上下文信息提升管道决策质量。相关方法对比如表1所示。

但在追求下游模型性能的同时，我们还需考虑数据准备过程的成本消耗。低预算场景下，我们不能盲目选择最高效果的数据准备管道，还应考虑各类方法的性价比。现有管道优化方法虽然能够通过策略设计提升数据准备效果或效率，但对成本收益关系的系统分析仍较不足。少数方法虽考虑成本，其重点也多在于降低管道搜索和构建阶段的算法开销，而非针对清洗算子自身的执行成本及资源投入进行优化。现有研究主要存在以下两方面局限：

（1）聚焦单独任务，缺乏对投入成本与模型收益之间关系的整体性分析。如表1提到的BoostClean[9]HoloClean[10]等方法主要关注特定清洗算法的效果提升，对成本的考虑多局限于算法自身的运行开销，

表1 数据准备优化策略对比总结

数据准备优化策略		模型性能考量	成本考量	优化依据	优化目标	功能
数据管道组合优化	Learn2Clean[5]	↔↔↔	↔	数据质量指标	性能提升	算子决策
	HAIPipe[6]	↔↔↔	↔↔	人类-AI 协作	性能提升	管道编排
	Diffprep[7]	↔↔↔	↔	可微梯度优化	性能提升	流程优化
	Ctxpipe[8]	↔↔↔	↔	上下文感知	性能提升	流程优化
数据清洗方法	BoostClean[9]	↔↔	↔↔	错误检测	算法鲁棒	错误修复
	HoloClean[10]	↔↔	↔	概率推理	数据质量	统计清洗
	Uniclean[11]	↔↔↔	↔↔	算法集成	性能提升	按需清洗
	ActiveClean[12]	↔↔	↔↔	主动学习	性能提升	主动清洗

较少分析该算法对完整数据准备流程及最终下游模型性能的影响。因此，有必要对各数据准备环节的资源投入与最终收益进行统一评估，为具体算法的选择和组合提供依据。

（2）缺乏成本收益视角下的数据准备方法调配和资源分配策略的研究。目前VChecker[13]、CrowdGame[14]等众包研究已对成本收益问题进行了探索，CBAClean也通过综合清洗效果与执行成本推荐具有成本效益的清洗方案[15]。但这些方法主要面向特定任务或具体清洗方案，对完整数据准备管道的算子组合、执行程度和资源分配仍缺少系统建模。现有管道研究通常更关注理想条件下的性能提升，忽视了实际预算约束下的成本收益平衡。

针对上述问题，本文将成本收益优化引入ML数据准备过程，在保证训练数据质量和下游模型性能的同时，对数据准备成本进行统一管控与分配，提出面向用户需求的自动化数据准备优化框架。该框架充分考虑实际场景中的资源限制，通过联合优化算子组合和资源投入，寻找低成本、高收益的数据准备方案。

对数据准备全局进行有效的成本收益优化存在诸多挑战，需要克服以下**技术难点：**

1、传统成本指标与预算量化之间的差异。现有研究常以数据量、特征维度等技术指标衡量数据准备开销，这样的指标虽能反映资源消耗，但难以直接与实际的预算需求挂钩。因此想要将实际成本预算限制纳入优化条件，必须将以技术指标为主的成本计算根据现实世界的成本量纲进行转换，这大幅提升了成本评估的复杂程度。

2、成本控制需要与模型性能追求的兼顾。以往的数据准备管道研究不考虑成本问题，只需充分地执行清洗方法，尽可能达到更佳模型效果，但这样可能造成过高的数据准备开销。同样的，若一味压缩成本，也可能造成数据质量和下游模型效果大幅下降。因此，如何在满足用户需求的前提下，实现成本和收益的合理兼顾，是本文需要解决的关键难点。

3、复杂组合空间优化算法的复杂度控制。若数据准备包含K类环节，每类包含M种方法和P种参数配置，则搜索空间高达 $O((M\times P)^K\times K!)$，直接进行全量搜索会导致严重的组合爆炸问题。同时本文在传统管道搜索基础上引入成本的规划分配，进一步提升了优化问题的复杂程度。因此如何设计高效的剪枝机制和近似算法，在控制计算开销的同时获得较优的数据准备方案，是本文一个重大的技术挑战。

针对以上量纲转换、效益均衡及搜索空间复杂度等方面的核心瓶颈，本文通过

理论创新与算法优化提出了一套系统性的解决方案，主要贡献和创新点主要围绕以下几个方面：

1、提出面向成本收益协同优化的数据准备整体框架（cost-benefit-aware data preparation，CBAprep）。该框架将成本建模、收益预测、组合搜索与资源分配统一纳入优化流程，采用统计回归方法评估数据准备成本，并利用梯度提升树（gradient boosting decision tree，GBDT）预测下游模型收益，从而在预算约束下联合优化数据准备效果与资源消耗，弥补现有方法侧重性能提升、缺乏成本收益分析的不足。

2、提出数据准备算子组合的资源分配算法（cost-benefit-aware resource optimization，CARO）。与以往研究追求理想状态下的数据准备方法的充分执行不同，本文聚焦解决数据准备方法的成本收益优化问题。CARO通过引入任务执行程度变量，将方法部分执行纳入优化空间，并根据预算约束自适应分配资源。通过结合蒙特卡洛采样与梯度引导优化，将资源分配问题由高复杂度搜索转化为关于任务数量线性增长的近似求解过程，算法的时间复杂度从的$O(n^2)$降低至$O(n)$，实现复杂资源分配问题的高效求解。

3、设计方法组合与资源分配的分层联动搜索方法。不同于传统仅针对方法组合进行搜索优化的研究，本文联合考虑方法组合决策与资源分配两类优化变量，在候选方法组合筛选基础上进一步进行资源投入优化，使数据准备过程同时兼顾组合选择与资源分配。该机制结合规则约束减少无效搜索空间，并通过粗粒度组合筛选与细粒度资源规划协同提升整体优化效率，显著提升了搜索效率。

4、实验表明，在预算受限场景下，CBAprep在效果与鲁棒性方面展现出明显优势。在5个真实数据集上，相比4种先进数据准备基线，CBAprep在有限预算约束下平均获得更优性能收益，在低预算场景下准确率最高提升超过6%。

1 相关工作

近年来ML模型训练数据准备过程中的成本问题得到了一定关注，现有工作主要从自动化算法研究和管道策略优化两个方面来控制数据准备的成本。其中自动化清洗主要适用于缺失值填充、异常值检测、数据一致性检查等具体数据处理环节，利用概率模型、深度学习等技术自动完成实现特定的清洗目标。研究者们通过对算法不断改进优化来提升数据清洗的效果和效率。管道策略优化方面的研究则是追求通过近似计算和先验筛选快速获得最佳管道编排，实现源数据到下游模型“端到端”的自动化流程。

1.1 数据清洗自动化算法研究

数据清洗自动化方面的研究主要集中在对训练数据的不一致性、异常值、缺失值、重复项等问题进行自动化处理。传统数据清洗方法包括基于完整性约束的清洗、基于规则的清洗、基于统计的清洗、基于深度学习的清洗等[2][16]。例如均值填补、回归插补等统计技术通过数据分布特征对缺失值进行处理[17]-[20]。近年来，研究逐渐转向先进ML技术。HoloClean[10]利用概率图模型、自动化规则和监督学习检测并修复数据错误，减少人工干预；ZeroER[21]利用对抗生成网络实现零样本重复记录识别；DemandClean[22]从数据真实性与多样性的

权衡出发，构建多目标学习框架。

此外针对不同领域的不同数据清洗需求，研究者们还提出了适用于不同领域、具有独特性的自动化清洗方法。针对机器学习模型训练场景，ActiveClean[12]面向凸优化模型，将数据清洗嵌入梯度下降过程，并优先处理高损失样本。自动化算法评估方面，BoostClean[错误!未找到引用源。]采用集成学习方法从预定义方法库中筛选最优清洗策略。类似地，RLClean[23]进一步利用深度强化学习实现清洗策略自动决策与优化。此外，针对高维、不平衡等复杂数据场景，已有研究通过特征重要性评估和搜索策略降低冗余特征带来的建模与清洗成本[24]。

1.2 自动化ML数据准备管道优化策略

随着各类自动化数据处理工具的涌现，为了在ML数据准备阶段实现高效的数据管道编排，研究者们开展了自动化ML数据准备管道优化策略的相关研究[25]。

其中一类方法将管道搜索视为复杂组合优化问题，采用机器学习、强化学习、元特征和代理模型等技术降低搜索开销[26]。例如Learn2Clean[5]框架通过研究各类方法之间的约束关系缩减搜索空间，并设计了基于Q学习算法的强化学习管道搜索机制。

另一类方法将传统离散管道搜索转化为连续优化问题，通过可微架构搜索降低开销。例如HAIPipe[6]结合人工编排与AI编排进行管道优化。DiffPrep[7]则将离散搜索空间转化为连续可微空间，利用梯度下降进行搜索，整个过程仅需执行一次ML模型训练。Ctxpipe[8]捕获原始数据和转换数据的语义，并通过门控上下文插件控制模型训练与推理，识别高质量特征管道。

上述方法以自动化方式替代部分人工操作，降低了数据准备成本。然而，复杂参数和高维数据仍可能带来较大资源开销，且现有研究主要从算法设计角度控制时空成本，缺乏对算子成本与下游模型收益的统一量化。为此，本文进一步研究数据准备算子成本与模型收益的平衡，并据此优化方法组合与资源分配。

2 问题概述

本文首先针对各类机器数据清洗方法的成本，以及这些方法组合和资源分配的不同策略对下游ML模型质量指标的影响展开研究。在这些成本收益研究的基础上，本文提出一种面向成本收益协同优化的数据准备方法。以用户预算有限的情况为例，CBAprep能够在预算范围内为用户提供最高的模型性能收益。该问题的形式化定义如下：

给定下游模型M_T和原始数据集D，用户预算B，所有可能的数据准备方法组合$\phi \in \Phi$，数据准备策略的资源分配方案$\alpha \in A$。其中(ϕ,α)表示α资源分配方案下的ϕ流程。本文用函数$C(\phi,\alpha)$来表征成本，用质量函数$Q(\phi,\alpha)$评估方案(ϕ,α)对下游模型M_T的性能指标的影响。本研究致力于在搜索空间Φ中找到一个特定的方案对方案$\phi\star$，并计算得出该方案的一个资源分配方案$\alpha\star$，满足：

$$\phi\star,\alpha\star=\underset{\phi\in\Phi,\ \alpha\in A_\phi}{\arg\max}\ Q(\phi,\alpha)\ s.t.C(\phi,\alpha)\le B\#(1)$$

更直观的来讲，本文主要解决以下问题。给定原始数据集D、所有可能的数据准备策略组成的搜索空间Φ、模型性能评估指标Q、用户提供的预算B：从Φ中寻找成本小于预算B的一个ϕ，从而得到使性

能指标 Q 最大化的训练数据 $D'=\phi(D)$。CBAprep的核心目标是在在成本收益约束范围内寻求优化目标最大化的准备方法组合和资源分配方式。我们根据各类用户需求将优化问题建模为数学规划问题，通过求解规划问题实现整体数据准备流程的优化。

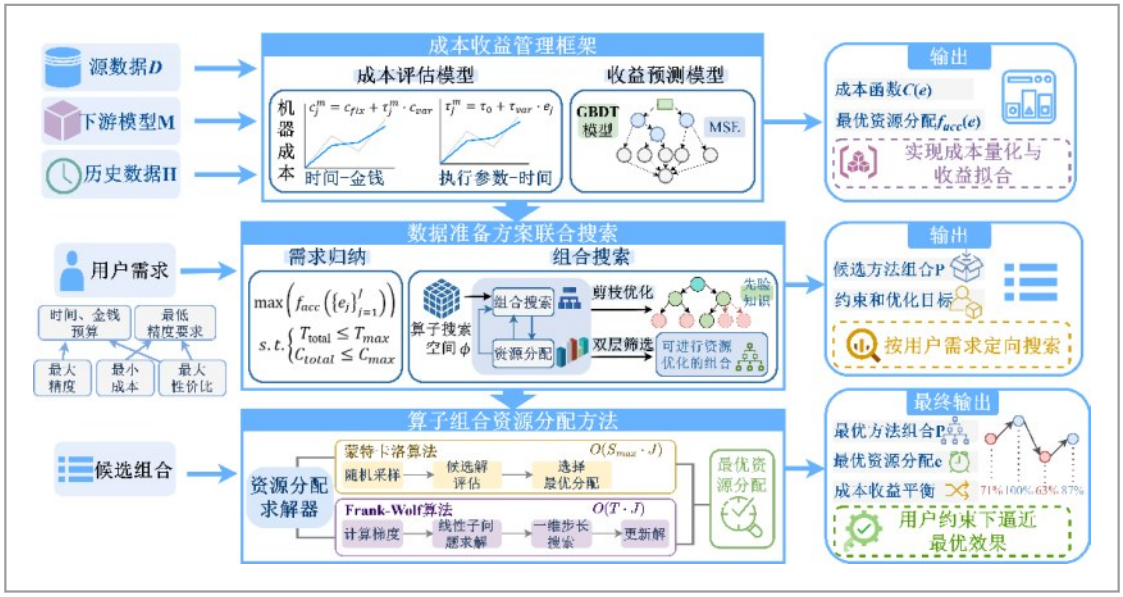

图1 CBAprep整体架构图

本研究的技术架构如图1所示，以分类任务作为下游任务，考虑预处理、数据清洗、特征工程等操作，收集用户具体需求指标，在数据准备方法组合空间中进行决策，并对具体组合进行资源优化分配，从而提供满足用户需求的最优数据准备方案。

3 CBAprep核心方法

3.1 基于成本收益需求的数学规划模型

本文首先在框架中对用户的需求进行归纳和分类为数学规划问题，如图2所示。

我们将用户对于数据准备过程在成本收益方面的需求分为以下几类：

（1）综合考虑成本和收益，选择能够在预算范围内提供最佳性能提升的方案。目标，是最大化收益成本比，同时满足总成本约束 $\sum_j c_j x_j \le B$ 。

（2）针对特定的精度要求，选择能够达到该精度的最低成本方案。设目标精度为 A，二进制变量 $\boldsymbol{x}$ 用来表示方法选择向量，$\boldsymbol{e}$ 表示方法的执行程度向量，f_{acc} 是方法选择和资源分配到下游模型准确度的映射，目标是最小化总成本 $\sum_j c_j x_j$，同时满足精度约束 $f_{acc}(\boldsymbol{x},\boldsymbol{e}) \ge A$。

（3）在预算有限的情况下，优先选择性价比高的方法。设总预算为 B，每个方法 j 的成本为 c_j，收益为 r_j，二进制变量 x_j

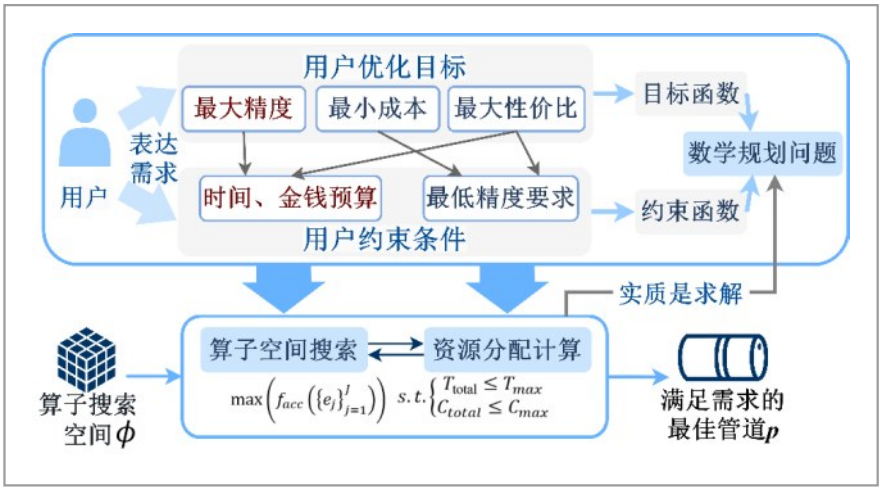

图 2　用户需求分类示意图

用来表示是否选择方法j。目标是最小化总成本$\sum_j c_j x_j \le B$，同时最大化总收益$\sum_j r_j x_j$。

基于以上三类需求对应优化目标最大精度、最小成本、最大性价比三类。在优化目标和约束条件下，我们将用户需求表达为多变量的数学规划问题。下文我们以基于预算约束的数据准备为例进行说明。

为了全面反映数据准备各类任务的成本，对于数据处理任务j，三种不同需求的优化目标不同。对于成本优先模式，我们的优化目标：

$$\min\left(C_{total}\right)\#(2)$$

对于精度优先模式，我们有优化目标：

$$\max\left(f_{acc}(j)\right)\#(3)$$

对于平衡模式，我们有优化目标：

$$\max\left(\frac{f_{acc}(j)}{C_{total}}\right)\#(4)$$

其中：

$$C_{total}=\sum_{j\in J}\left(c_j\right)\#(5)$$

对以上模式进行总结可以得到表2。

表 2　数据准备用户需求模式总结

类别	处理步骤	约束条件	公式编号
成本优先模式	最小化总成本	达到最低精度要求A	(2)
精度优先模式	最大化模型准确度	时间/金钱成本 <= B	(3)
平衡模式	最大化收益成本比	满足资源与精度的双向边界	(4)

成本收益分析部分的核心是在一定约束条件下得到实现优化目标的解。这些约束条件由数据准备框架中的用户引导机制中获得，使得最终优化结果充分符合用户要求。其数学表达如下：

$$\max\left(\frac{f_{acc}(j)}{C_{total}}\right) \quad s.t.\begin{cases} C_{total} \le C_{max} \\ f_{acc}(j) \ge A_{min} \end{cases} \#(6)$$

本方法通过深入分析用户需求并将其抽象为特定的需求范式，将其转化为具象的数学规划问题。后续算法的核心本质就是在组合决策与资源分配的框架下，实现对该规划模型及其衍生问题的有效求解。

3.2 数据准备成本收益管理框架

本文从时间开销视角展开对数据准备成本的分析和计算，并以下游模型的性能提升作为方法的收益依据，在此基础上构建统一的成本收益管理机制，对各类数据准备任务进行成本管控与全局收益评估。

1 成本计算方法 MachineCost

本文对机器成本进行了归纳计算，我们根据不同方法的机器处理逻辑分为以下三类，用于确定各类数据准备方法的成本调整空间和计算基准，如表3所示。

（1）规则映射处理类。该类方法可在局部数据上拟合规则，但必须将规则应用于全量数据，否则可能造成量纲失衡或语义错位。其资源投入可通过控制拟合数据规模、候选特征规模或搜索空间大小进行调节，因此部分执行主要表现为“部分数据拟合、全量规则应用”。

（2）样本独立处理类。包括缺失值处理、离群值检测和重复数据去除等。该类操作以样本或记录为基本对象，不同样本间相对独立，可通过控制处理样本比例调节成本，且处理时间通常与数据量近似线性相关。

（3）全局结构处理类。包括数据集划分、数据转换等。该类操作会改变数据集整体结构，难以按样本比例拆分，其成本在特定数据集和参数配置下通常较固定，因此本文将其作为固定成本项，不参与连续执行程度优化。

表3 数据准备方法成本计算类别划分

类别	处理步骤
规则映射处理类	特征编码,特征选择、特征提取、数据规范化
样本独立处理类	重复数据去除、缺失值处理、离群值检测
全局结构处理类	数据转换、数据集划分

基于以上对数据准备方法成本计算的分类总结，本文进一步对执行程度变量的适用边界展开形式化说明。对于第j个数据准备算子，执行程度$e_j \in [0,1]$表示该算子在可调资源维度上的投入比例。对于规则映射处理类和样本独立处理类算子，e_j可对应参与拟合的数据比例、处理样本比例、候选特征比例或搜索迭代比例。而对于全局结构处理类算子，本文设置$e_j = 1$，将其作为固定成本项纳入预算约束。

在此基础上，对于支持执行程度调节的算子，其机器成本可近似表示为执行程度的函数：

$$C_j(e_j) = a_j e_j + b_j \#(7)$$

其中，a_j和b_j为通过历史运行数据拟合得

到的成本参数，e_j表示第j个算子的执行程度。对于不支持执行程度调节的全局结构类算子，其成本记为固定值C_j。因此，给定数据准备方案P和执行程度向量$\boldsymbol{e}$，总成本可表示为：

$$C(P,\boldsymbol{e})=\sum_{j\in\mathcal{J}_a}C_j(e_j)+\sum_{j\in\mathcal{J}_f}C_j \#(8)$$

其中，$\mathcal{J}_a$表示支持执行程度调节的算子集合，$\mathcal{J}_f$表示固定执行算子集合。MachineCost采取基于上述算子执行逻辑分类的成本建模方式，从而实现对算子成本的灵活刻画。

2 数据准备收益评估模型

本文以下游模型性能衡量数据准备收益，并可根据任务类型选择不同指标：对分类任务采用准确率（accuracy）和F1分数（F1-score），回归任务采用平均绝对误差（mean absolute error，MAE）、均方根误差（root mean square error，RMSE）或决定系数（coefficient of determination，R^2），深度学习及大模型任务可采用验证集损失、Benchmark得分或人工偏好评价。因此，CBAprep的收益模型并不依赖特定下游模型，而是将数据准备策略到下游性能的映射视为一个可学习的预测问题。

以分类任务为例，本文以下游分类模型的准确率作为衡量数据准备收益的指标。因此本文针对分类任务训练收益评估模型，模型输入为方法组合的0-1决策变量及浮点型执行程度，输出为预测准确率这一过程面临成本和收益关系非线性、输入数据维度过高的问题。考虑到成本收益关系具有非线性且输入维度较高，本文采用GBDT模型进行收益预测，如图3所示。GBDT能够通过树结构刻画复杂非线性关系，并可通过限制树深度和叶节点数控制模型复杂度。

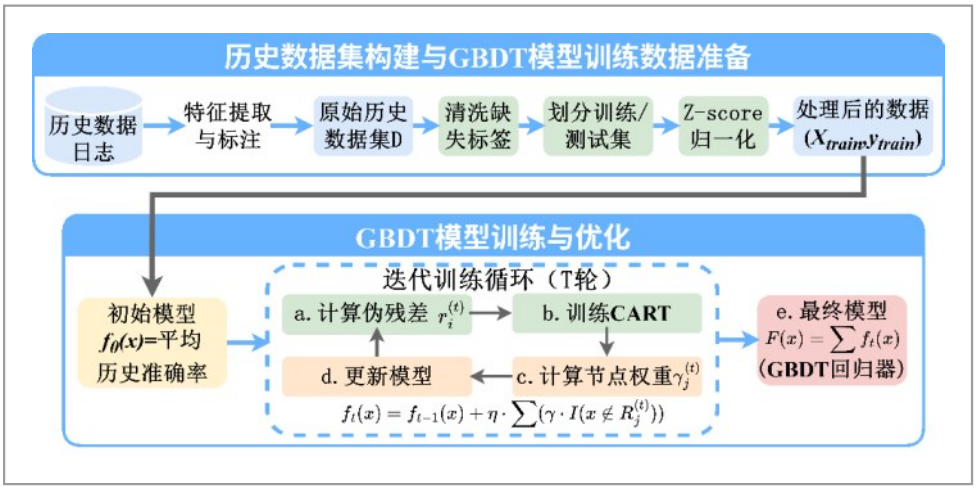

图3 基于GBDT的下游模型准确率预测模型流程

基于GBDT的下游模型准确率预测模型算法伪代码如下：

GBDT训练阶段复杂度时间复杂度为$O(T\cdot(d\cdot M\cdot h+2^h+M))$，由样本量$M$、树

算法 基于 GBDT 的下游模型准确率预测模型

输入：训练数据集 $D=\{(\boldsymbol{\phi}_i,y_i)\}_{i=1}^{n}$，其中 $\boldsymbol{\phi}_i=[\boldsymbol{z}_i,\boldsymbol{e}_i]$ 为数据准备算子决策向量（0-1变量及执行程度浮点数），$y_i\in[0,1]$ 为对应的下游模型实际准确率指标；

损失函数 $\mathcal{L}$：衡量预测精度与实际收益之间偏差的函数；

迭代次数 T、**学习率** η、**最大树深** d_{max}：控制模型复杂度，防止维度爆炸；

输出：最终的收益预测集成模型 $F_{acc}^{(T)}(\boldsymbol{\phi})$

1 $F_{acc}^{(0)}(\boldsymbol{\phi})\leftarrow\arg\min_{\gamma}\sum_{i=1}^{n}\mathcal{L}(y_i,\gamma)$

//计算历史样本准确率的初始平均水平作为预测起点

2 for t from 1 to T do

//展开 T 轮迭代，每轮构建一棵新的决策树

3 $r_{acc,i}^{(t)}\leftarrow-\left[\frac{\partial\mathcal{L}(y_i,F_{acc}(\boldsymbol{\phi}_i))}{\partial F_{acc}(\boldsymbol{\phi}_i)}\right]_{F_{acc}=F_{acc}^{(t-1)}},\forall i\in\{1,\ldots,n\}$

//计算准确率残差

4 $h^{(t)}\leftarrow$ 构建深度为 d_{max} 的 CART 树拟合样本 $\{(\boldsymbol{\phi}_i,r_{acc,i}^{(t)})\}_{i=1}^{n}$

//构建决策树

5 $\gamma_{acc,j}^{(t)}\leftarrow\arg\min_{\gamma}\sum_{\boldsymbol{\phi}_i\in R_j^{(t)}}\mathcal{L}(y_i,F_{acc}^{(t-1)}(\boldsymbol{\phi}_i)+\gamma),\forall j\in\{1,\ldots,J^{(t)}\}$

//计算使损失函数最小化的权重

6 $F_{acc}^{(t)}(\boldsymbol{\phi})\leftarrow F_{acc}^{(t-1)}(\boldsymbol{\phi})+\eta\cdot\sum_{j=1}^{J^{(t)}}\gamma_{acc,j}^{(t)}\mathbb{I}(\boldsymbol{\phi}\in R_j^{(t)})$

//将新生成的树以学习率 η 累加到原有模型中

7 end for

8 return $F_{acc}^{(T)}(\boldsymbol{\phi})$

//返回最终的集成模型

数量 T、树深度 h 和特征维数 d 共同决定。其中，特征标准化复杂度为 $O(d\cdot M)$，单棵树构建中分裂点查找占 $O(d\cdot M\cdot h)$，树生长因深度限制仅需 $O(2^h)$，残差计算每轮 $O(M)$。

本文首先利用线性回归模型量化数据准备方法的执行成本，再基于历史运行记录构建GBDT模型预测下游任务收益，从而实现对资源消耗与性能增益的协同评估。作为CBAprep的关键组件，MachineCost与GBDT收益模型的有效性将在4.4节通过成本估计和收益预测实验进行验证。

3.3 数据准备算子组合的资源分配算法CARO

数据准备算子组合资源分配优化实际上就是将给定方案的代价降低至预算范围内，并在这个过程中尽量保留最大收益。对于支持渐进式资源投入的数据准备算子，执行程度 $\boldsymbol{e}_j$ 表示其资源投入比例；对于不支持部分执行的全局结构类算子，其执行程度固定为1，并作为固定成本项计入预算。

具体的，给定数据准备方案 P、预算 B、成本函数 $C(P,\boldsymbol{e})$ 和收益预测函数 $F(P,\boldsymbol{e})$，资源分配问题可表示为：

$$\max_{\boldsymbol{e}}F(P,\boldsymbol{e})\#(9)s.t.C(P,\boldsymbol{e})\le B,0\le e_j\le 1,j\in\mathcal{J}_a$$
$$e_j=1,j\in\mathcal{J}_f$$

其中 $\boldsymbol{e}=(e_1,e_2,\ldots,e_J)$，$\mathcal{J}_a$ 表示支持执行程度调节的算子集合，$\mathcal{J}_f$ 表示固定执行算子

集合。因此，只需求解以上固定方法组合的预算资源规划问题，就能实现预算范围内高代价方法组合的成本优化。求解上述问题面临高维连续搜索困难、高价值资源投入方向难以动态识别的双重挑战。

（1）基于蒙特卡洛采样的资源优化方法

蒙特卡洛方法通过在可行域内随机采样执行程度向量，将复杂优化问题转化为有限候选解评估问题，适用于高维资源分配，且无需目标函数的显式梯度，对非凸、非线性收益函数具有较强适应性。本文在约束可行域内随机采样执行程度向量 $\boldsymbol{e}^{(1)},\boldsymbol{e}^{(2)},\ldots,\boldsymbol{e}^{(S_{max})}$，其中 S_{max} 为采样次数，并计算各采样方案的预算可行性及收益：

$$R^{(s)}=f_{acc}\left(\boldsymbol{e}^{(s)}\right)\#(10)$$

选择收益最大的可行解：

$$\boldsymbol{e}^{*}=\arg\max_{s}\ R^{(s)}\#(11)$$

若每次采样需计算 J 个任务，则整体时间复杂度为 $O(S_{\max}\cdot J)$，关于任务数线性增长，可通过调节 S_{max} 平衡搜索精度与计算开销。本文主要采用蒙特卡洛近似求解，并以Frank-Wolfe方法作为收益函数可微时的替代路径。

（2）基于 Frank-Wolfe 思想的资源优化方法

对资源优化问题进行求解，本质上就是求解约束问题，由于我们用e表征任务数量占比、投入时间占比等比例关系，因此 $T_{\text{total}}\le T_{max}$ 和 $C_{total}\le C_{max}$ 对 $\boldsymbol{e}_j$ 来说是线性约束，可以用Frank-Wolfe 算法思想来对预算资源规划问题进行求解。具体伪代码如下所示：

传统搜索方法在 $[0,1]^J$ 空间维度内随任务数 J 指数增长，Frank-Wolfe 算法通过将非线性优化分解为序列线性问题，每次迭代仅需在可行域顶点间搜索， $O(1/\epsilon)$ 次迭代后能够达到 ϵ-最优解，解决了搜索效率问题。

算法 数据准备方案资源消耗优化方法

输入：方法组合 $\phi=\{x_1,x_2,\ldots,x_J\}$，准确度函数 $f_{acc}(\boldsymbol{e})$，$\boldsymbol{e}\in[0,1]^J$，最大迭代次数 K，精度参数 ϵ

输出：近似解 $\boldsymbol{e}^{*}$

$\boldsymbol{e}\leftarrow\boldsymbol{e}^{(0)}$

//设置初始可行点

for k from 0 to $K-1$

$\boldsymbol{g}^{(k)}\leftarrow\nabla f_{acc}(\boldsymbol{e}^{(k)})$

//计算梯度，确定可行方向

$\boldsymbol{s}^{(k)}\leftarrow\arg\max_{s^{(k)}\in[0,1]^J}\left\langle\boldsymbol{g}^{(k)},\ \boldsymbol{s}\right\rangle$

//解决线性优化子问题，得到可行域顶点 $\boldsymbol{s}^{(k)}$

$\boldsymbol{d}^{(k)}\leftarrow\left\langle\boldsymbol{g}^{(k)},\ \boldsymbol{e}^{(k)}-\boldsymbol{s}^{(k)}\right\rangle$

//沿着下降方向作一维步长搜索

if $\boldsymbol{d}^{(k)}\le\epsilon$

return $\boldsymbol{e}^{(k)}$

//满足收敛条件，返回结构

$\gamma_k=\arg\max_{\gamma\in[0,1]}f_{acc}(\boldsymbol{e}^{(k)}+\gamma(\boldsymbol{s}^{(k)}-\boldsymbol{e}^{(k)}))$

$\boldsymbol{e}^{(k+1)}\leftarrow\boldsymbol{e}^{(k)}+\gamma(\boldsymbol{s}^{(k)}-\boldsymbol{e}^{(k)})$

//不满足收敛条件，继续迭代

end for

return $\boldsymbol{e}^{(K)}$

面向数据准备成本收益协同优化任务，本文构建了资源约束下的实现数据准备收益最大化方法，引入用户需求分析机制，通过成本量化与基于 GBDT 的收益拟合，配合双层搜索策略，实现了数据准备算子组合与资源投入的联合优化。

4 实验设置与结果分析

4.1 数据集

我们主要针对一系列分类任务展开实验，选用数据集与对应下游任务场景的具体细节如下：

① Google Play Store Apps(简称

Google)：Kaggle 真实数据集，包含 Google Play 商店应用信息，存在字段缺失、噪声和类别不平衡等问题。任务是预测应用评分是否高于 4.0，可用于商店推荐。

② House Prices（简称 House）：Kaggle 真实数据集，包含面积、位置、建造年份和房间数量等房屋属性，存在缺失值、异常值、特征冗余和类别不平衡等问题。任务是预测房价是否高于中位数，可用于房地产估价或贷款风险评估。

③Shuttle：加州大学欧文分校机器学习数据集库（University of California, Irvine Machine Learning Repository，UCI）数据集，包含航天飞机飞行过程中的温度、压力和振动等传感器读数，存在噪声、字段缺失和类别不平衡等问题。任务是判断航天飞机状态是否异常，用于健康监测和故障预警。

④Chess：UCI 数据集，包含国际象棋游戏的位置和移动信息，存在字段缺失、噪声和类别不平衡等问题。任务是预测游戏是否获胜，可用于国际象棋 AI 训练和策略分析。

⑤Uscensus：UCI 数据集，包含美国人口普查中的年龄、教育程度和职业等信息，存在字段缺失、噪声和类别不平衡等问题。任务是预测个人年收入是否高于 5 万美元，可用于市场分析或政策优化。

上述数据集的属性数量和数据量信息如下表所示。

表 4　实验数据集相关信息

数据集	属性数量	数据量
Google Play Store Apps	13	10.8k
Jungle chess	7	4.5k
House Prices	81	1460
Shuttle	10	58 k
Uscensus	15	3.2k

4.2 对比模型和评价指标

实验中选取分类任务作为下游任务进行初步实验，选用逻辑回归（logistic regression，LR）、线性判别分析（linear discriminant analysis，LDA）、朴素贝叶斯（naive Bayes，NB）以及分类与回归树（classification and regression tree，CART）作为下游分类模型，以验证方法对不同模型结构的适用性。数据按照 7∶3 划分为训练集和测试集，测试集在管道搜索过程中保持不可见。考虑到各基线支持的模型存在差异，统一采用 LR 完成管道方法对比。

实验指标方面，本文采用 Performance、Time、Cost/Budget 和 Gain/Cost 四项核心指标，用来展现数据准备方案的收益、成本、实际预算比和成本收益效率。指标的含义和来源如表 5 所示，其中 Performance 采用 accuracy 和 F1-score 两类具体指标衡量。

本文参与对比的基线方法有以下几种：

① Learn2Clean(简称 L2C)：基于经典

表 5　实验指标含义与来源

指标	含义	来源
Performance	模型性能	预设的模型性能衡量指标
Prep_Time	执行时间	数据准备过程的整体时间
Cost/Budget	实际预算比	实际数据准备成本与给定预算的比值
Gain/Cost	成本收益效率	模型性能提升与实际数据准备成本的比值

强化学习（Q-Learning）在预定义动作空间中寻找最优数据清洗路径。

HAIpipe-AI(简称 HAI-AI): 应用 HAIPipe 提出的基于深度强化学习的 AI 管道生成算法, 结合深度强化学习与启发式搜索生成数据准备管道。

DiffPrep: 将预处理过程微分化，通过梯度下降联合优化预处理方案与模型参数。

④ Ctxpipe: 通过语义嵌入获取数据上下文特征，并通过深度强化学习进行算子决策。

本文进一步引入预算约束随机搜索（random-budget search，RBS）、贪心背包方法（greedy knapsack，GK）和均匀预算分配（uniform-budget allocation，UBA）三种预算感知基线：

① RBS：在预算约束下随机采样算子组合及执行程度，并选择验证集性能最优的可行方案。

GK：按单位成本收益比排序，并采用背包式贪心策略选择算子。

UBA：在算子组合确定后，将预算均匀分配给各可调算子。

4.3 实验设置

为了展现本研究在数据准备管道成本收益规划方面的优势，本文将 CBAprep 方法与其他数据准备管道生成方法 Learn2Clean、DiffPrep 、Ctxpipe、HAIpipe-AI 进行对比：在不同运行时间约束下，各管道搜索框架生成的最优管道能达到的模型性能。

设定一个整体的数据准备管道运行时间预算 B_{exec}，分别取 300ms、500ms、700ms、1000ms。收集在时间预算 B_{exec} 约束下，各类数据管道生成方法下游模型在验证集上的性能指标，包括 accuracy 和 F1-score。

对于本文 CBAprep 框架，直接输入 B_{exec} 作为约束进行管道生成即可。而对于其他绝大部分基线方法，它们不具备运行时间预算约束规划模块。因此本文在它们生成的最优管道基础上，如果生成管道运行时间超出时间预算，就对管道内容进行裁剪以满足时间预算要求。如果管道的运行时间最终无法满足 B_{exec}，则判定在该生成方法无法交付可用结果（表格中用“-”来表示）。

为避免非预算感知基线通过管道裁剪带来的实验偏差，本文进一步与 RBS、GK 和 UBA 三种预算感知基线进行比较。所有方法均在相同预算约束下搜索数据准备方案，并比较其平均下游模型性能、实际预算比和单位成本收益。

4.4 实验结果与分析

4.4.1 管道搜索结果在有限预算下的表现对比

根据表 6，在多数预算受限场景下，CBAprep 均能生成满足预算要求的可用管道，并取得较优的准确率。当时间预算降至 300 ms 时，多数缺乏预算感知机制的基线方法无法生成有效管道，而 CBAprep 在所有预算设置下均能稳定输出可用方案，体现出较强的资源受限适应性。以 Shuttle 数据集为例，CBAprep 在 300 ms 预算下的准确率达到 0.966，已接近多数基线方法在 1000 ms 预算下的性能，说明其能够优先组合高性价比算子，在有限预算内获得较高收益。随着预算由 300 ms 增加至 1000 ms，CBAprep 的性能总体平稳提升，表明其可根据资源变化调整算子组合和分配方案。相比之下，部分基线方法因

表 6 不同管道在不同预算约束下的效果对比

数据集	B_{exec}	L2C	HAI-AI	DiffPrep	Ctxpipe	CBAprep
Google	300ms	–	–	0.545/0.356	0.525/0.474	**0.545/0.478**
	500ms	**0.600/0.591**	0.550/0.358	0.549/0.547	0.595/0.581	0.556/0.539
	700ms	**0.600/0.591**	0.550/0.358	0.549/0.547	0.595/0.581	0.586/0.554
	1000ms	**0.633/0.633**	0.621/0.613	0.593/0.574	0.595/0.581	0.628/0.614
Jungle	300ms	–	–	–	**0.674/0.641**	0.662/0.645
	500ms	0.604/0.600	0.681/0.652	0.670/0.484	0.674/0.641	**0.681/0.674**
	700ms	0.673/0.671	0.683/0.651	0.670/0.484	0.674/0.641	**0.687/0.521**
	1000ms	0.680/0.656	**0.760/0.733**	0.680/0.552	0.674/0.641	0.741/0.728
House	300ms	–	–	–	0.818/0.813	**0.898/0.898**
	500ms	0.877/0.865	0.901/0.900	0.895/0.878	0.818/0.813	**0.903/0.900**
	700ms	0.915/0.912	0.913/0.911	0.915/0.910	0.818/0.813	**0.915/0.899**
	1000ms	**0.928/0.928**	0.920/0.919	**0.928/0.928**	0.818/0.813	0.928/0.918
Shuttle	300ms	0.900/0.892	0.922/0.910	–	0.934/0.927	**0.966/0.405**
	500ms	0.900/0.892	0.930/0.923	0.954/0.891	**0.974/0.972**	0.970/0.632
	700ms	0.966/0.965	0.929/0.921	0.979/0.907	0.974/0.972	**0.981/0.606**
	1000ms	0.966/0.965	0.951/0.947	0.990/0.479	**1.000/1.000**	0.982/0.567
Uscensus	300ms	–	–	0.816/0.695	0.785/0.756	**0.858/0.731**
	500ms	0.796/0.759	0.800/0.682	0.816/0.695	0.813/0.797	**0.871/0.690**
	700ms	0.796/0.759	0.802/0.686	0.849/0.780	0.813/0.797	**0.875/0.700**
	1000ms	0.854/0.849	0.802/0.686	0.849/0.780	0.813/0.797	**0.874/0.723**

缺乏成本收益规划，在预算不足时需要裁剪管道，导致性能明显下降。例如 Learn2Clean 在 Shuttle 数据集上于 700 ms 和 1000 ms 时准确率为 0.966，而在

300 ms和500 ms时降至0.900。总体来看，CBAprep能够随预算变化提供更精细、稳定的数据准备方案。

4.4.2 预算感知基线对比实验

本文进一步与RBS、GK和UBA三种预算感知基线进行比较，所有方法均在相同预算约束和相同候选规模下搜索数据准备方案，其中RBS从满足预算约束的随机候选中根据验证集真实性能选择最优方案。表7汇总了各数据集和预算设置下的平均结果，包括accuracy、Cost/Budget和Gain/Cost。其中，accuracy和Gain/Cost越高越好；Cost/Budget应优先不超过1，在满足预算约束时越接近1表示预算利用越充分，若均大于1，则越接近1表示超预算程度越小。

表7 预算感知基线平均实验结果对比

数据集	方法	Avg. accuracy	Avg. Cost/Budget	Avg. Gain/Cost
Google	CBAprep	0.5788	**0.7336**	0.0786
	RBS	**0.5889**	1.1010	0.1007
	GK	0.5546	1.1509	**0.1559**
	UBA	0.5761	1.1390	0.1453
Jungle	CBAprep	**0.6928**	0.4040	**0.0734**
	RBS	0.6833	**0.7523**	0.0289
	GK	0.6791	1.2051	−0.1528
	UBA	0.6788	1.4699	−0.1231
House	CBAprep	**0.9110**	**1.1527**	**0.0001**
	RBS	0.8694	1.4392	−0.1391
	GK	0.8671	1.2051	−0.1528
	UBA	0.8738	1.4699	−0.1231
Shuttle	CBAprep	**0.9748**	0.4906	**0.2283**
	RBS	0.9678	**0.8452**	0.1874
	GK	0.8336	0.5325	−0.6606
	UBA	0.6788	0.725	0.0003
USCensus	CBAprep	**0.8695**	**0.8251**	0.1335
	RBS	0.8633	1.1509	**0.1559**
	GK	0.8487	1.101	0.1007
	UBA	0.8661	1.139	0.1453

实验结果表明，CBAprep在多数数据集上取得了更高的平均准确率，并在预算利用和单位成本收益方面表现较为稳定。与RBS相比，CBAprep利用成本估计和收益预测结果减少无效搜索，能够更稳定地找到高收益候选方案。与GK相比，CBAprep不仅考虑单个算子的单位成本收益，还进一步考虑算子组合之间的相互影响和执行程度分配，因此能够避免局部贪心选择带来的次优结果。与UBA相比，CBAprep能够根据进行自适应资源分配，从而取得更好的成本收益效率。

4.4.3 消融试验

本文设置了CBAprep方法的消融实验，用于分析验证系统关键组件在成本收益规划方面的作用，以及探究各类超参数设置对系统整体性能及效果的影响。

1 关键组件作用分析

在以下实验中，本文将与以下几种不同类型的数据清理策略进行比较，包括整体框架的对比实验：

① CBAprep：在算子优化选择基础上进行自适应资源分配。

R_DIS：仅进行算子组合的优化选择。

R_ALL：完全随机产生的数据准备策略。

④ NONE：不进行数据准备。

表8 对比方法组件设置

策略	算子采用	算子优化选择	自适应资源分配
CBAprep	√	√	√
R_DIS	√	√	×
R_ALL	√	×	×
NONE	×	×	×

以上策略都需要满足预算要求。我们对比以上策略的运行时间、运行开销和分类模型准确率。本研究通过将策略CBAprep与R_DIS、R_ALL、NONE不同实验对比，设置时间约束为1000ms，得到实验结果如下：

表9 CBA模块消融实验

数据集	下游模型	实验方法	prep_time	accuracy	F1
Google Play Store Apps	NB	CBAprep	0.702	**0.717**	**0.702**
		R_DIS	0.649	0.627	0.608
		R_ALL	0.575	0.158	0.135
		NONE	**0**	0.175	0.148
	LDA	CBAprep	0.665	**0.678**	**0.667**
		R_DIS	0.547	0.654	0.638
		R_ALL	0.571	0.285	0.257
		NONE	**0**	0.079	0.061
	CART	CBAprep	0.988	**0.559**	**0.539**
		R_DIS	0.722	0.537	0.512
		R_ALL	0.351	0.225	0.198
		NONE	**0**	0.253	0.224

整体来看，CBAprep在NB、LDA和CART模型上均取得最高的准确率与F1值，其中NB准确率达到0.717，说明其能够稳定提升下游模型性能。相比之下，NONE与R_ALL表现较差，R_ALL在NB上的准确率仅为0.158，表明随机选择算子及资源分配容易造成性能不稳定。R_DIS明显优于R_ALL，验证了算子筛选

模块能够获得更合适的算子组合，但仍低于CBAprep，说明精细资源分配具有进一步提升作用。

从成本收益表现看，在LDA模型下，R_DIS比R_ALL减少0.024s执行时间，同时将准确率由0.285提升至0.654，说明算子筛选能够识别高性价比方案。CBAprep仅比R_DIS增加0.118 s，却将准确率进一步提升至0.678，表明资源分配模块能够通过合理配置有限资源获得更优的数据准备效果。

2 超参数设置影响

资源分配计算部分的采样次数影响特定管道的成本与性能，而S_{max}是蒙特卡洛采样的基本参数，对系统整体的管道优化结果产生重要影响。为了分析S_{max}对搜索性能的影响，我们开展相关消融实验。

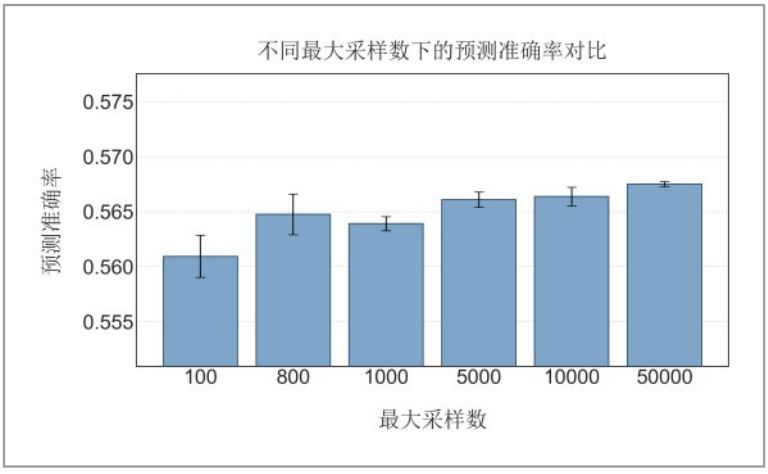

图4 最大采样次数S_{max}对下游模型准确率的影响

如图所示，随着蒙特卡洛采样次数$S_{\max}$增加，预测准确率总体提升并逐渐趋于稳定。当$S_{max}=100$时，准确率约为0.551，且误差波动较大，说明采样不足会导致资源分配估计不稳定。随着$S_{\max}$增至800和1000，准确率明显提高，误差范围缩小，表明增加采样次数能够提升估计可靠性。当$S_{\max}$达到5000以上时，准确率提升幅度逐渐减小，继续增加采样仅带来有限收益和额外开销。因此，本文将$S_{\max}=10000$设为默认值，以平衡搜索效果与计算成本。

4.4 4 成本估计与收益预测模型有效性验证

由于CBAprep的优化决策依赖于MachineCost成本估计模型和GBDT收益评估模型的预测结果，本文进一步对两个模型进行独立验证。

1 成本估计模型MachineCost有效性验证

对于MachineCost成本估计模型，本文采样不同数据准备算子、不同执行程度

和不同数据集规模下的运行配置，记录真实执行时间作为真实成本，并与MachineCost预测结果进行比较。评价指标采用MAE、RMSE、R^2和平均绝对百分比误差（mean absolute percentage error，MAPE）。其中，MAE和RMSE用于衡量预测成本与真实成本之间的绝对误差，MAPE用于衡量相对误差，R^2用于衡量模型对真实成本变化的拟合程度。图5和表10展示了MachineCost成本估计模型的预测结果。

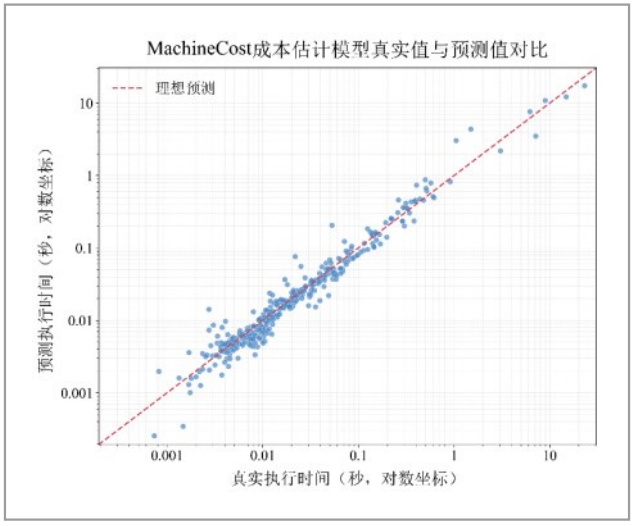

图5 MachineCost成本预测效果

表10 MachineCost成本估计平均结果

数据集	下游模型	MAE	RMSE	R^2	MAPE
全部	LR	0.0685	0.4395	0.9216	25.1122

MachineCost成本估计模型的MAE为0.0685，RMSE为0.4395，R^2达到0.9216，说明该模型能够较好拟合真实执行成本变化趋势，并有效解释不同数据准备方案之间的成本差异。MAPE为25.1122，其相对误差处于可接受范围内，说明MachineCost成本估计具有可靠性。

2 GBDT收益评估模型有效性验证

对于GBDT收益评估模型的有效性验证，本文构造不同的数据准备管道配置，获得对应下游模型真实性能，并将其与GBDT预测收益进行比较。评价指标采用MAE、RMSE、R^2和Spearman相关系数。其中，Spearman相关系数用于衡量预测收益排序与真实收益排序之间的一致性。具体实验结果如图6和表11所示。

根据实验数据，GBDT收益预测模型的MAE为0.0045，RMSE为0.0077，R^2达到0.8352，说明该模型能够较准确地拟

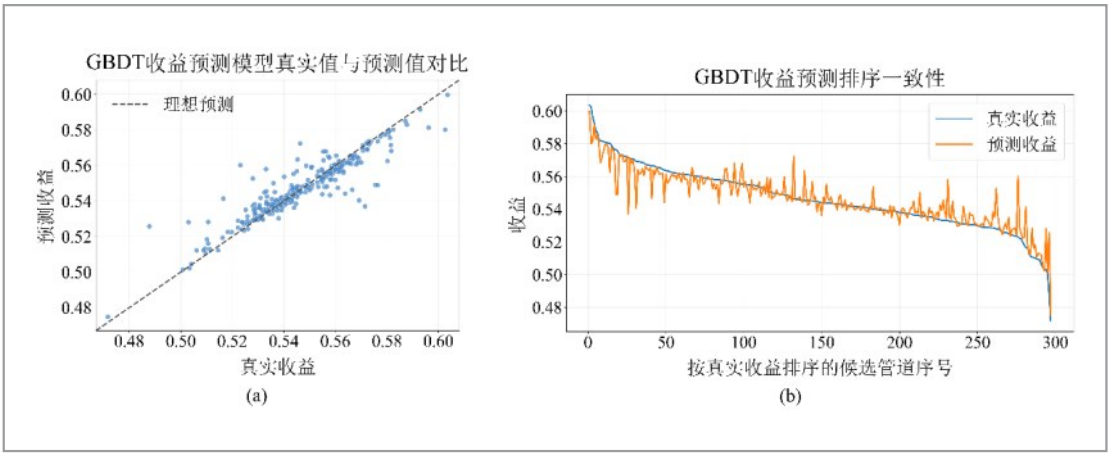

图 6 (a) GBDT 收益预测模型收益预测效果 (b)GBDT 收益预测模型预测收益排序一致性

表 11 GBDT 收益预测平均结果

数据集	下游模型	MAE	RMSE	R^2	Spearman
全部	LR	0.0045	0.0077	0.8352	0.9071

合数据准备策略与下游模型性能之间的非线性关系。进一步地，Spearman 相关系数达到 0.9071，表明预测收益排序与真实收益排序具有较强一致性，能够有效识别高收益候选管道。同时，收益预测模型的有效性也说明 CBAprep 具有一定迁移能力，更换下游模型时，只需替换任务指标并重新训练或微调收益预测模型即可。

4.5 CBAprep 在其他模型类型场景下的可移植性分析

上文实验部分主要以分类任务为代表验证 CBAprep 的效果。但值得注意的是，CBAprep 的核心思想并不依赖于特定分类模型，而是将数据准备策略抽象为成本建模、收益预测和预算约束优化问题。因此，在可移植性方面，CBAprep 可以通过改变收益指标来适用于不同的模型场景。对于回归任务可将收益指标由 Accuracy 替换为 MAE、RMSE 或 R^2；深度学习任务可将训练时间、显存消耗和数据增强成本纳入成本模型，并以验证集性能或损失作为收益指标。

在大模型训练与微调场景中，数据准备过程通常包括数据去重、低质量样本过滤、指令数据构造、安全过滤、人类偏好标注和样本筛选等环节[27][28]。这些环节均存在明显的成本收益权衡关系，CBAprep 可将 Token 消耗、图形处理器计算成本、人工标注及筛选成本纳入成本模型，并以 Benchmark 得分、验证集损失、奖励模型评分或人工偏好胜率衡量收益。需要指出的是，实际大模型场景下数据规模更大、收益评估周期更长、成本构成更复杂，因此如何构建适用于大模型数据准备场景的高精度、低开销成本收益预测模型，将作为本文后续关键研究方向之一。

5 结束语

综上，本文围绕数据准备的成本不可控与资源利用效率不足问题，构建了一个面向成本收益优化的数据准备方法CBAprep。在方法层面，通过建立数据准备过程的成本收益管理框架，实现了从技术代价指标到现实预算量纲的映射，为后续优化提供了量化方法；在优化层面，提出数据准备方案的资源分配优化方法，以线性复杂度实现近似最优的成本分配；在搜索层面，设计方法组合与资源分配的分层联动机制，兼顾组合选择与资源优化，有效压缩组合空间，提升优化效率。在约束成本的情况下，CBAprep在多数数据集上的表现超过其他的基准数据准备方法，展现了开展成本收益分析规划方面的研究在数据准备领域的重要作用，为实现成本收益平衡的高质量数据准备提供了可行路径。

参考文献：

[1] 刘艺, 曹建军, 翁年凤, 等. 数据治理及其发展研究综述[C]//第十二届中国指挥控制大会论文集(上册). 北京: 军事科学出版社, 2024: 170-176.
Liu Y, Cao J J, Weng N F, et al. Research review on data governance and its development[C]//Proceedings of the 12th China Command and Control Conference. Beijing: Military Science Press, 2024: 170-176.

[2] 郝爽, 李国良, 冯建华, 等. 结构化数据清洗技术综述[J]. 清华大学学报(自然科学版), 2018, 58(12): 1037-1050.
Hao S, Li G L, Feng J H, et al. A survey on structured data cleaning technologies [J]. Journal of Tsinghua University (Science and Technology), 2018, 58(12): 1037-1050.

[3] 黄科满, 许多, 杜小勇. 数据社会化视角下的数据流通安全治理: 五位一体框架[J]. 大数据, 2024, 10(6): 5-15.
Huang K M, Xu D, Du X Y. Data circulation security governance from the perspective of data socialization: A five-sphere framework[J]. Big Data Research, 2024, 10(6): 5-15.

[4] 范举, 陈跃国, 杜小勇. 人在回路的数据准备技术研究进展[J]. 大数据, 2019, 5(6): 1-16.
Fan J, Chen Y G, Du X Y. Research progress on human-in-the-loop data preparation technologies[J]. Big Data Research, 2019, 5(6): 1-16.

[5] Berti-Equille L. Learn2Clean: Optimizing the sequence of tasks for web data preparation[C]//Proceedings of the World Wide Web Conference. New York: ACM, 2019: 2580-2586.

[6] Chen S, Tang N, Fan J, et al. HAIPipe: Combining human-generated and machine-generated pipelines for data preparation[J]. Proceedings of the ACM on Management of Data, 2023, 1(1): 1-26.

[7] Li P, Chen Z, Chu X, et al. DiffPrep: Differentiable data preprocessing pipeline search for learning over tabular data [J]. Proceedings of the ACM on Management of Data, 2023, 1(2): 1-26.

[8] Gao H, Cai S, Dinh T T A, et al. CtxPipe: Context-aware data preparation pipeline construction for machine learning [J]. Proceedings of the ACM on Management of Data, 2024, 2(6): 1-27.

[9] Krishnan S, Franklin M J, Goldberg K, et al. BoostClean: Automated error detection and repair for machine learning[R]. arXiv:1711.01299, 2017.

[10] Rekatsinas T, Chu X, Ilyas I F, et al. HoloClean: Holistic data repairs with probabilistic inference[J]. Proceedings of the VLDB Endowment, 2017, 10(11): 1190-1201.

[11] Ding X, Qian Z, Wang H, et al. UniClean:

A multi-signal fusion pipeline for optimizing data cleaning workflow[C]//2025 IEEE 41st International Conference on Data Engineering. Piscataway: IEEE, 2025: 4600-4603.

[12] Krishnan S, Wang J, Wu E, et al. ActiveClean: Interactive data cleaning for statistical modeling[J]. Proceedings of the VLDB Endowment, 2016, 9(12): 948-959.

[13] Zhang H, Chai C, Doan A H, et al. Manually detecting errors for data cleaning using adaptive crowdsourcing strategies [C]//Proceedings of the 23rd International Conference on Extending Database Technology. Konstanz: OpenProceedings.org, 2020: 311-322.

[14] Mao Z, Li X, Guo Z. A game-based framework for crowdsourced data labeling[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. New York: ACM, 2019: 1339-1348.

[15] Ding X, Su H, Qian Z, et al. CBAClean: A comprehensive system for recommending data cleaning solutions through cost-benefit analysis in data quality management[C]//2025 IEEE 41st International Conference on Data Engineering. Piscataway: IEEE, 2025: 4636-4639.

[16] Rahm E, Do H H. Data cleaning: Problems and current approaches[J]. IEEE Data Engineering Bulletin, 2000, 23(4): 3-13.

[17] Ishaq M, Zahir S, Iftikhar L, et al. Machine learning based missing data imputation in categorical datasets[J]. IEEE Access, 2024, 12: 88332-88344.

[18] Lee Y, Park S. High-dimensional missing data imputation via undirected graphical model[J]. Statistics and Computing, 2024, 34(5): 160. Lin W, Tsai C. Missing value imputation: A review and analysis of the literature (2006 - 2017) [J]. Artificial Intelligence Review, 2019, 53(2):1487-1509.

[19] Lin W, Tsai C. Missing value imputation: A review and analysis of the literature (2006—2017) [J]. Artificial Intelligence Review, 2020, 53(2): 1487-1509.

[20] Guo C, Yang W, Liu C, et al. Iterative missing value imputation based on feature importance[J]. Knowledge and Information Systems, 2024, 66(10): 6387-6414.

[21] Wu R, Chaba S, Sawlani S, et al. ZeroER: Entity resolution using zero labeled examples[C]//2020 IEEE International Conference on Data Mining. Piscataway: IEEE, 2020: 1149-1164.

[22] Qian Z, Ding X, Wang C, et al. DemandClean: A multi-objective learning framework for balancing model tolerance to data authenticity and diversity [J]. Proceedings of the VLDB Endowment, 2025, 18(12): 5339-5342.

[23] Peng J, Shen D, Nie T, et al. RLClean: An unsupervised integrated data cleaning framework based on deep reinforcement learning[J]. Information Sciences, 2024, 682: 121281.

[24] 苏璇, 王远军. 面向高维不平衡医学数据的特征选择算法[J]. 小型微型计算机系统, 2024, 45(2): 309-318.
Su X, Wang Y J. Feature selection algorithm for high-dimensional imbalanced medical data[J]. Journal of Chinese Computer Systems, 2024, 45(2): 309-318.

[25] Li P, Rao X, Blase J, et al. CleanML: A study for evaluating the impact of data cleaning on ML classification tasks[C]// 2021 IEEE 37th International Conference on Data Engineering. Piscataway: IEEE, 2021: 13-24.

[26] Wang Q, Guo Y, Yu L, et al. Deep Q-network-based feature selection for

multisourced data cleaning[J]. IEEE Internet of Things Journal, 2021, 8(21): 16153-16164.

[27] Luo J, Wu B, Luo X, et al. A survey on efficient large language model training: From data-centric perspectives[C]// Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics. Stroudsburg: Association for Computational Linguistics, 2025: 30904-30920.

[28] Zha D, Bhat Z P, Lai K H, et al. Data-centric artificial intelligence: A survey [J]. ACM Computing Surveys, 2025, 57 (5): Article 129.

作者简介

丁小欧（1993-），女，哈尔滨工业大学计算学部副教授、博导。主要研究方向为大数据治理、面向人工智能的数据质量管理等。

崔文萱（2004-），女，哈尔滨工业大学计算学部研究生在读。主要研究方向为数据清洗、数据治理。

录用日期: XXXX-XX-XX

通信作者:

基金项目: 国家科技重大专项(No. 2025ZD1607102)；国家自然科学基金面上项目(No. 62572151)；黑龙江省自然科学基金优秀青年项目(No. YQ2024F005)；黑龙江省青年科技人才托举工程项目(No. 2025QNTJ001)

Foundation Items: National Science and Technology Major Project of China (No. 2025ZD1607102), National Natural Science Foundation of China General Program (No. 62572151), Heilongjiang Provincial Natural Science Foundation Excellent Young Scholars Program (No. YQ2024F005), Heilongjiang Young Scientific and Technological Talent Support Program (No. 2025QNTJ001).