

控制与决策

Control and Decision

基于地标特征和元学习方法推荐最适用优化算法

崔建双, 吕玥, 徐子涵

引用本文:

崔建双, 吕, 徐子涵. 基于地标特征和元学习方法推荐最适用优化算法[J]. *控制与决策*, 2021, 36(5): 1223–1231.

在线阅读 View online: <https://doi.org/10.13195/j.kzyjc.2019.0993>

您可能感兴趣的其他文章

Articles you may be interested in

基于动态行为选择的和声搜索算法

Harmony search algorithm based on dynamic behavior selection

控制与决策. 2021, 36(3): 577–588 <https://doi.org/10.13195/j.kzyjc.2019.0597>

带不相关并行机和有限缓冲MHFS调度的混合启发式算法

Hybrid heuristic algorithm for multi-stage hybrid flow shop scheduling with unrelated parallel machines and finite buffers

控制与决策. 2021, 36(3): 565–576 <https://doi.org/10.13195/j.kzyjc.2019.0835>

融合稀疏编码与深度学习的草图特征表示

A feature representation of sketch based on fusion of sparse coding and deep learning

控制与决策. 2021, 36(3): 699–704 <https://doi.org/10.13195/j.kzyjc.2019.0941>

基于KPCA和G-G聚类的多元时间序列模糊分段

Fuzzy segmentation of multivariate time series with KPCA and G-G clustering

控制与决策. 2021, 36(1): 115–124 <https://doi.org/10.13195/j.kzyjc.2019.0849>

参数未知的离散系统Q-学习优化状态估计与控制

Q-learning optimal state estimation and control for discrete systems with unknown parameters

控制与决策. 2020, 35(12): 2889–2897 <https://doi.org/10.13195/j.kzyjc.2019.0180>

基于地标特征和元学习方法推荐最适用优化算法

崔建双[†], 吕 玥, 徐子涵

(北京科技大学 东凌经济管理学院, 北京 100083)

摘 要: 设计并实证研究一种基于地标特征和元学习方法推荐最佳优化算法的实现框架. 地标特征摒弃了传统的问题简单特征、统计特征和信息理论特征复杂的提取过程, 通过简化运行算法并仅以算法的相对性能表现作为问题特征集. 在此基础上, 利用元学习方法训练建模并针对新问题作出算法推荐. 为验证推荐效果, 以多模式资源约束的项目调度问题(MRCPSP)为优化对象, 以人工蜂群、蚁群、粒子群和禁忌搜索4种元启发式算法作为推荐对象, 分别使用人工神经网络、 k 最近邻、决策树以及随机森林4种元学习方法建立推荐元模型. 计算结果表明, 多种元学习方法均指向相近的推荐准确率, 平均稳定在70%以上, 最高可达95%. 基于地标特征和元学习方法实现优化算法推荐是一个值得进一步探讨的新方向.

关键词: 地标特征; 元学习; 算法推荐; 元特征; 准确性; 分类器; 元启发式优化算法

中图分类号: TP181

文献标志码: A

DOI: 10.13195/j.kzyjc.2019.0993

开放科学(资源服务)标识码(OSID):



引用格式: 崔建双, 吕玥, 徐子涵. 基于地标特征和元学习方法推荐最适用优化算法[J]. 控制与决策, 2021, 36(5): 1223-1231.

Recommending best suitable metaheuristic based on landmarking feature and meta-learning approach

CUI Jian-shuang[†], LYU Yue, XU Zi-han

(Dolinks School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China)

Abstract: This paper presents and empirically studies an implementation framework for recommending the best suitable optimization algorithm based on landmarking features and meta-learning approaches. The landmarking abandons the traditional feature extraction techniques and/or approach. The landmarking features are obtained using the simplified algorithm on the problem and using only the relative performance of the algorithm as the feature dataset. On this basis, meta-learning approaches are applied to train the metamodel and make algorithm recommendations for new problems. In order to verify the effect, a set of multi-mode resource constrained project scheduling problem (MRCPSP) is selected as the objective. Four meta-heuristic algorithms, namely artificial bee colony, ant colony system, particle swarm optimization and tabu search, are selected as the recommended algorithms. The four meta-learning approaches, namely artificial neural network, k -nearest neighbourhood, decision tree and random forest, are used to generate the recommended meta-model. The empirical study shows that all the prediction results point to similar recommendation accuracy, with an average stabilised around 70% and a maximum at 95%. The optimization algorithm recommendation based on the landmarking and meta-learning approach is a new direction worthy of further exploration.

Keywords: landmarking feature; meta-learning; algorithm recommendation; meta-feature; accuracy; classifier; meta-heuristic algorithm

0 引 言

元启发式算法在求解具有NP难特征的大规模组合优化问题领域获得了广泛的应用. 这类算法虽不能确保最优解, 但大多具备全局收敛性, 能够以较

高的效率在可接受的时空条件下获得问题的近优解^[1]. 然而, 正如无免费午餐定理指出的那样: 对于广泛的问题域, 没有普遍适用的“最佳”算法能够胜过所有其他算法^[2]. 反过来说, 每一个具体问题都存在

收稿日期: 2019-07-12; 修回日期: 2019-12-10.

基金项目: 国家自然科学基金项目(71871017).

责任编辑: 刘向杰.

[†]通讯作者. E-mail: cuijs@manage.ustb.edu.cn.

着算法“偏好”,使得某个算法在该问题上的表现最佳. 算法推荐就是要从众多可用算法中选择最适用当前问题的算法,从而有效地提升解决问题的效率和效果.

事实上,算法推荐问题存在于许多研究领域,但表述方式不尽相同. 即使在同一研究领域也有不同的侧重点. 在机器学习领域,人们提出了元学习(meta-learning)的概念^[3],强调对各种机器学习算法在分类或回归问题上的表现进行学习,推荐最佳算法以实现更准确的数据分类或回归^[4]. 在优化算法研究领域,伴随着各种启发式和元启发式优化算法的大量涌现,算法推荐已成为继算法定制和算法改进,提升算法应用效果的一条重要途径^[5]. 为了达到理想的优化效果,减少因算法定制带来的高成本,研究者们已着手开展跨算法乃至跨领域的交叉融合技术,提出了各种算法推荐方法和技术,出现了诸如超启发式(hyper-heuristics)^[6]、元学习(meta-learning)^[7]、集合策略(ensemble strategies)^[8]、岛模型(island models)^[9]、算法合成(portfolios)^[10]、自适应算子选择(adaptive operator selection)^[11]等研究分支. 其中,基于元学习的算法推荐方法和技术,借鉴了机器学习领域多年成功经验,有望在优化算法推荐问题的研究上取得突破. Smith-Miles曾断言:源于机器学习领域的元学习方法对分类算法的推荐技术完全可以推广至优化算法领域^[12].

从目前公开文献来看,基于元学习方法实现优化算法的推荐已经取得了令人鼓舞的成果. 文献[7]提出并实证了一种优化算法智能推荐系统. 该系统利用支持向量机实现了4种元启发式优化算法的分类推荐,平均准确率可达90%以上. 基于同样的目的,文献[5]利用前馈型神经网络技术成功实现了多种元启发式优化算法的推荐,平均准确率达到80%以上. 文献[13]引入了一种基于元学习的元启发式方法以解决带时间窗的车辆路径问题. 通过定义一组表征不同路径问题算例的元特征,使用多层感知分类器,结合包装器元特征选择方法,预测出最适合于给定问题算例的元启发式算法. 文献[14]使用基于标签排名算法的元学习方法来求解旅行商问题. 实验结果验证了元学习器可以准确预测不同场景下元启发式的排名. 文献[4]提出了一种基于元学习的自适应算法推荐模型,通过将问题特征映射到算法性能来自适应地选择算法. 实验采用基准函数和具有各种性质的现实问题,揭示了在解决方案准确性、排

名和成功率方面对各种测试问题的可扩展性和有效性. 文献[15]根据黑盒连续优化问题的要求提出了基于景观分析方法的顺序、邻域结构和计算复杂性的分类,讨论了算法选择框架的应用及其与算法组合、混合元启发式和超启发式之间的关系. 文献[16]利用知识发现方法理解问题结构与启发式有效性之间的关系,使用决策树选择最佳启发式,其准确性超过97%. 除了依据问题特征自动推荐最佳优化算法之外,基于元学习的方法还有助于解释问题特征与算法效果之间的关系,探究算法效率成因和自适应条件,有望从根本上揭示算法表现与问题自身固有属性之间的规律,解释一个给定算法更适宜解决什么类型的问题以及为什么的问题.

值得注意的是,在相关研究文献中,人们不约而同地提出了一个共性问题,即元特征(meta-feature)的恰当定义对算法推荐准确性的重要影响. 从广义的角度来看,凡是有助于反映所要解决问题的属性和算法特点的数据都可称为元特征. 然而,如何提取元特征?提取哪些元特征?每个元特征对推荐准确度的影响程度如何?迄今仍是悬而未决的一系列问题.

本文提出并验证一种仅基于地标特征推荐最佳优化算法的实现方法. 据作者所知,迄今为止尚未见到类似公开文献. 地标特征是元学习中不常见的元特征提取技术^[17],摒弃了传统的问题固有属性和统计测量特征的复杂提取过程,通过直接提取优化算法自身性能表现来定位训练数据的位置(标号). 具体而言,就是使用优化算法的简化版本快速获得各训练算例的目标值,并将其偏差作为地标特征集,同时,以取得最小目标值的算法作为最适用优化算法,给予分类标号;然后使用元学习方法训练建模,用于对新问题算例作出算法推荐. 为了验证效果,本文以典型的组合优化问题——多模式资源约束的项目调度问题(MRCPSP)为优化对象^[18],选用人工蜂群(artificial bee colony, ABC)、蚁群(ant colony system, ACS)、粒子群(particle swarm optimization, PSO)和禁忌搜索(taboo search, TB)4种元启发式算法作为推荐对象分别使用人工神经网络(artificial neural network, ANN)、 k 最近邻(k -nearest neighbor, k -NN)、决策树(decision tree, DT)以及随机森林(random forests, RF)^[19]4种元学习算法建模以验证推荐效果.

相对于其他特征提取方法^[5,7],本文的研究成果验证了如下观点成立:仅基于地标特征进行算法推荐是可行的,效果是可接受的. 其优势在于元特征

提取过程简单,不需要对问题的深入介入,可以大大降低特征提取时间和空间复杂度. 在没有显著降低推荐准确率的前提下,简化了以往提取问题特征的繁杂过程. 实验结果表明,4种元学习方法均指向相近的推荐准确率,平均稳定在70%以上,最高可达95%. 这一方面说明地标特征的鲁棒性,另一方面说明基于地标特征实现优化算法更高推荐准确率是一个值得进一步探讨的新课题.

1 算法推荐的相关问题

关于算法推荐(选择),Rice^[20]最早曾给出一个概念框架. 设 P 是问题的集合, A 是可用算法的集合, F 是问题可测特征的集合, Y 是算法性能测度的集合. 形式化地表述为:对于给定问题实例 $x \in P$,若已知相关特征 $f(x) \in F$,算法推荐就是要找到算法 $\alpha \in A$,令映射 $S(f(x)) \rightarrow A : \max\{y(\alpha(x))\} \in Y$.

当给定问题和推荐原则后,准确地推荐最适用于问题的算法可大大提升解决问题的绩效^[12,21],达到事半功倍的目的. 最直接的算法推荐方法无疑是“试错法”,最可靠的算法推荐方法则是依据专家经验,然而,前者过于消耗时间和资源,后者却难以获得推广. 因此,有必要寻求更高效的方法和途径. 其中,基于元学习的算法推荐方法值得深入研究. 元学习思想源于机器学习领域提升学习模型的泛化能力和稳定性的需求^[22]. 在机器学习领域,人们从大量数据的训练中寻找规律,进行分类并作出行为预测. 许多优秀的机器学习算法,如神经网络、支持向量机、 k 近邻、决策树、随机森林等是人工智能领域认知科学的结晶. 20世纪90年代中期,算法研究者逐渐意识到此领域“没有免费的午餐”^[2],提出了元学习的概念,强调对机器学习算法在分类问题上的表现进行学习来寻求最佳学习算法. 元学习系统接收一组元实例输入,获取其中的元知识并用其构建元学习(分类)器(meta-learner)^[23-24].

借鉴机器学习领域的元学习方法来实现优化算法的推荐,通过对优化算法性能表现的学习来推荐优化算法是一种“离线”学习模式^[3]. 主要实现步骤如下:1)提取实例集元特征;2)在实例集上执行候选算法 (a_1, \dots, a_n) ,获得各算法在各算例上的性能表现并作出优劣排序;3)元学习方法建模元分类器;4)将元分类器用于新实例算法推荐. 由于算法存在“偏好”^[2],通过建立元分类器寻求算法表现与问题特征之间的“偏好”关系,从而将算法推荐问题转化为分类问题. 把符合某一类特征的算例归为一类并向其推

荐已被证明是“好”的算法. 其中,特征来自于算例数据集,分类目标号则来自于对算法性能表现的排序. 机器学习领域对分类效果的最常用的评价指标是准确率(命中率),定义为正确分类的数据集样本数与样本总数之比,大小在0~1之间,值越大准确率越高.

2 基于地标特征实现优化算法推荐

2.1 问题特征与地标特征

特征是算法相对性能的良好预测因子. 特征间接地表征数据集并在数据集样本上表征算法的性能^[25]. 基于元学习方法实现算法推荐,一项不可或缺的任务就是提取数据集特征. 高质量的数据集特征在提供数据信息的基础上,可对算法的性能差别作出合理的解释. 就目前人们对元学习方法的认知程度,主要提取如下几种类型的特征:1)源自问题实例的简单特征、统计特征和信息论特征^[26]. 例如,变量维度、节点个数、标准差、偏度、峰度、相关系数、平均熵等^[5,7]. 2)景观特征^[27-28]. 通过引入某种形式的拓扑结构来理解搜索空间的结构以及对算法行为的影响. 常见景观特征包括适应度距离相关性、变量相互依赖、噪声和适应度分布、鲁棒性和粗糙度等. 3)基于模型的特征^[24]. 将决策树的不同属性用作特征值. 例如叶子数、节点数等. 4)地标特征或基准特征^[17]. 地标的一般含义是特定地点的标记或边界. 在机器学习领域,地标给出的是分类器专业知识空间问题定位的标志点. 地标关系不是计算绝对性能,而是有效地捕捉相对于彼此的表现^[29]. 基于地标的元学习方法假定分类器对问题的表现揭示了有关该问题性质的信息,这些信息可以通过运行在该问题上的分类器进行收集. 对于优化问题而言,优化算法就是分类器或者称为地标学习器(landmarkers)^[30]. 可以通过快速估计算法性能来获得算法与问题的匹配程度;可以在全体数据集实例空间运行算法的标准版本或者在数据集部分样本上运行简化版本来获得子采样地标. 此处,基准的概念源自机器学习分类算法中寻求相同的归纳偏置. 在推荐优化算法过程中,要求对被推荐的算法具有相同的评判标准,例如迭代次数相同、执行时间相同或可行解数量相同等.

在算法推荐研究领域,人们对于哪些特征能够反映问题本质性的区别仍无定论. 大多数研究文献采用的是基于统计和信息论类型的特征^[4,5,7,14-16]. 获得这类特征最大的难点在于特征的准确定义和高成本的计算复杂度^[31]. 文献[5,7]曾花费了大量精力和时

间定义并提取MRCPS的简单特征和统计特征,虽然是以直接方式获得数据集的特征,但提取的特征是否与算法性能关联紧密却不得而知,仍需要花费大量的时间逐一甄别这些特征的价值.在各种特征的反复提取和验证过程中,本文注意到提取特征,特别是提取与问题密切相关的特征是一项极其繁琐且棘手的问题,迄今为止没有简单规律可循.此外,一些问题的专属特征对于另一些问题可能并不十分明显.即使提取了各种类型的问题特征,但仍然难以确保所提取的特征确实全面反映了问题关联算法的本质.在此过程中,本文敏锐地注意到,地标特征无论从推荐准确率还是从提取环节都具有令人刮目相看的效果.

2.2 基于地标特征推荐优化算法的实现过程

基于地标特征推荐优化算法的实现过程如图1所示.通过执行优化算法的简化版本来获得地标元特征以及优化算法性能的相对排序,组成元数据集.下面具体说明图1中各模块的实现过程.

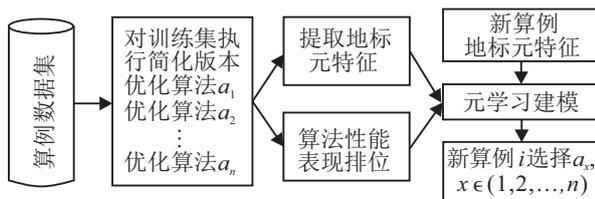


图1 基于地标元特征推荐优化算法的实现框架

算例数据集是指类型相同但规模或参数不同的一组问题算例.大多数此类问题都有包含成百上千算例的标杆算例库,可供算法研究者下载测试自己的算法并作出比较.例如资源约束的项目调度问题、作业车间调度问题(jobshop scheduling problem, JSP)、车辆路径问题(vehicle routing problem, VRP)、旅行商问题(travelling salesman problem, TSP)等.对训练集执行简化版本优化算法 a_1, a_2, \dots, a_n ,一般通过减少种群规模、迭代时间或次数等使得算法以较少的运行资源在较短时间内快速获得性能表现.提取地标元特征是让简化算法 a_1, a_2, \dots, a_n 运行在每一个训练算例的数据集上,以得到各算法在不同算例上的目标值偏差.根据目标值偏差对算法性能表现进行排位,偏差小的排位靠前.地标元特征结合排位即构成元数据集,之后利用元学习建模得到元分类器.当新算例出现时首先提取新算例地标元特征,然后利用元分类器推荐最佳优化算法.

本文用到的元学习方法包括ANN、 k -NN、DT和RF,均是有监督机器学习算法,均可利用地标特征和算法标号组成的元数据集训练得到元分类器,实现分

类预测^[19].

3 实验过程及结果分析

3.1 实验环境设置

采用Matlab (R2015b)编程实现.从项目调度问题库(project scheduling problem library, PSPLIB)^[32]选取108个MRCPS实例作为数据集.这些算例规模相同且特征很接近,实验结果更能展示推荐效果.选用ABC、ACS、PSO和TB四种元启发式优化算法参与推荐.为了确保公平无偏,采取如下几项措施:1)使用同一台计算机(CPU core i7, 2.6 G, 64位);2)提取地标特征时各优化算法平均运行5次取均值;3)各优化算法采用相同的编解码方式;4)种群数量均设为20.各优化算法参数如下:1)ABC:蜜蜂角色变换上限为10,下一代蜂群选择采用轮盘赌方式;2)ACS: $\rho = 0.2, \alpha = 0.1, \beta = 1, Q = 40$,残留信息素更新采用蚁周模型;3)PSO: $c_1 = 1.2, c_2 = 0.8$,惯性权重 $\omega = 0.98 - 0.3 \times \text{迭代次数}/\text{最大迭代次数}$,取标准粒子群算法公式^[33];4)TB:禁忌对象2-opt邻域交换,邻域解最大数量为20,无条件破禁,禁忌表长为 $2\sqrt{1.3 \times n}$ (n 是问题规模).

各优化算法的性能表现以与已知目标函数值偏差的大小进行排序,偏差最小者最佳,标号为1,其余从2~4依大小排序进行编号,实验过程仅推荐标号为1的最佳算法.

3.2 获取算例集地标特征

实验初始阶段,本文曾设计了与优化算法搜索过程相关的一些特征,如平均改进值、最大和最小改进值等,但效果不佳.随后,本文确定以目标值偏差作为地标特征.首先分别编程实现4种优化算法,然后运行这4种优化算法的简化版本.根据优化算法的简化程度获得4组108行4列地标特征,分别是:1)执行100次可行解;2)执行600次可行解;3)执行1000次随机迭代;4)执行固定时长60s.

这些地标特征加上各优化算法排序编号,便构成了关于MRCPS的地标元特征矩阵,是下一步元学习建模必要的元知识数据集.

3.3 元学习建模

使用ANN、 k -NN、DT和RF四种元学习算法分别建模以验证优化算法的推荐效果.为了确保通用性和公平性,建模时均通过抽样法随机选取88个训练算例和20个测试算例,并采用10折交叉验证以防止过拟合.4种机器学习算法的推荐准确率均为运算10

次后取均值,下面具体说明实现过程.

3.3.1 ANN推荐过程

ANN是对人脑神经网络进行抽象并建立的简化模型,主要由输入层、隐藏层以及输出层构成. 关键参数包括隐藏层数量和激活函数. Matlab提供的神经网络工具箱包含一组训练和预测函数. 前馈神经网络函数 Feedforwardnet() 采用 hiddenSize 和 trainFcn 作为参数. 使用双循环遍历择优两个参数,即取 hiddenSize = 10, 15 或 20; trainFcn = radbas, logsig 或 tansig 三者之一. ANN的标号值由四位二进制值组成,算法固定编号为 ABC = 1, ACS = 2, PSO = 3, TB = 4. 当前实例的最佳元启发式算法标记为1,其余标记为0. 如0100表示ACS是当前实例的最佳算法,而0001表示TB是最佳算法. 以此类推. ANN平均预测准确率保持在70%以上,最高准确率在84%左右.

3.3.2 k-NN推荐过程

k-NN基于某种距离的度量寻找训练集中与目标最靠近的k个算例,并基于k个最近邻算例的类型,使用“投票法”来对新算例进行分类. 如果特征空间中k个最邻近的样本中的大多数属于某个类别,则样本也属于此类别. 基于该原则,当应用k-NN进行分类推荐时,训练算例的标号值是1、2、3和4之一,分别表示按默认顺序排列的算法. 例如,1表示算例使用ABC,4表示使用TB,依此类推.

本文分别取 $k = 2, 3, 4, 5, 7$ 进行筛选测试. 为了防止过拟合,从训练集中做10次随机取样交叉验证. 测试结果表明 $k = 3$ 效果最好,最终选用3-NN进行分类预测.

3.3.3 DT推荐过程

DT由结点、分枝和叶子组成. 沿决策树从上到下遍历的过程中,每个结点都会遇到一个测试,对每个结点上问题的不同的测试输出导致不同的分枝. 通过构造决策树来发现数据中蕴涵的分类规则. 构造精度高、规模小的决策树是核心目标^[34].

DT推荐过程分两步进行:1)利用训练数据集来构造初始决策树;2)使用决策树剪枝技术构造剪枝精度高、规模小的决策树,对步骤1)中生成的决策树进行校验修正,将那些影响准确性的分枝剪除.

用DT算法来预测准确率时,将第1列置为算例顺序号,第2列置为最佳标号值,其余各列为算例地标特征值. 本文选用目前较为流行的C4.5算法,这是一种改进的ID3算法,其产生的分类规则易于理解,准确率较高. 用信息增益率来选择属性,克服了用信

息增益偏向选择取值多的属性不足^[35].

3.3.4 RF随机森林预测过程

RF包含多颗决策树. 每颗决策树都是一个分类器,当输入一个样本之后,每颗决策树都会得到相应的输出分类结果,将其中出现最多的分类结果作为最终分类结果. RF属于集成学习中的套袋(bagging)方法. 首先从原始数据集中随机抽取n个训练算例,共进行k轮抽取,得到k个训练集;然后使用DT获得k颗相对独立的决策树;最后由投票表决产生分类结果. RF输入数据的格式与DT的相同.

3.4 实验结果分析

3.4.1 推荐准确率

分别独立使用4组地标特征(100次可行解、600次可行解、1000次随机迭代和固定时长60s)进行算法推荐. 按照3.3节所述,分别将4组地标特征矩阵和标号向量代入4种元学习算法进行训练建模,并利用获取的模型对测试算例数据进行模拟推荐. 表1列出了最差(wst)、平均(avg)、最好(best)推荐准确率. 准确率定义为

$$HR = \frac{\text{Hit_number}}{\text{Total_number}} = \frac{\text{测试算例中预测正确的标号数量}}{\text{全部测试算例数量}}$$

图2分别给出了这4种元学习方法推荐准确率折线图.

由表1数据可见,最高推荐准确率94%发生在使用地标特征“固定时长60s”和RF元学习算法的情况. 此时的最差和平均推荐准确率分别是87.50%和90.65%,横向比“100次可行解”的情况普遍高出20个百分点;纵向比ANN、k-NN等算法也高出近10个百分点. 分析其原因有如下两点:一是实测获取“固定时长60s”的地标特征(3700s,每个算例平均34s)所花费的时间是“100次可行解”地标特征(240s,每个算例平均2s)的15倍多,说明较长时间的优化算法运行能够提取含有更多信息的地标特征,所付出的代价是计算时间的增加;二是RF算法是一种在决策树算法基础上提高分类性能的多决策树技术,理论上应该比单一分类算法具有更好的效果,所付出的代价也是计算时间上的增加.

进一步分析看,k-NN算法表现出较平均的预测准确率,其准确率标准差仅为0.02. 分析其原因主要是该算法对异常值不敏感,不具有显式学习过程,不需要进行训练或者说训练开销几乎为零,在得到测试样本后可直接根据距离作出分类.

表1 基于地标特征的最小/最大/平均推荐准确率 %

元学习算法		100次可行解	600次可行解	1000次迭代	固定时长60s	平均值
ANN	最差(wst)	65.00	64.00	65.50	71.00	72.62
	平均(avg)	72.40	69.65	71.90	78.00	
	最好(best)	77.00	75.00	78.00	84.00	
k-NN	最差(wst)	71.67	68.33	73.24	72.59	72.67
	平均(avg)	72.95	69.64	74.12	74.01	
	最好(best)	74.17	70.93	75.00	75.37	
DT	最差(wst)	61.00	67.50	75.00	79.50	74.94
	平均(avg)	64.35	73.40	78.80	83.25	
	最好(best)	68.50	79.50	81.50	87.00	
RF	最差(wst)	67.00	72.00	79.50	87.50	80.82
	平均(avg)	69.90	78.40	84.35	90.65	
	最好(best)	74.00	83.00	89.50	94.00	
平均值		69.83	72.61	77.20	81.41	

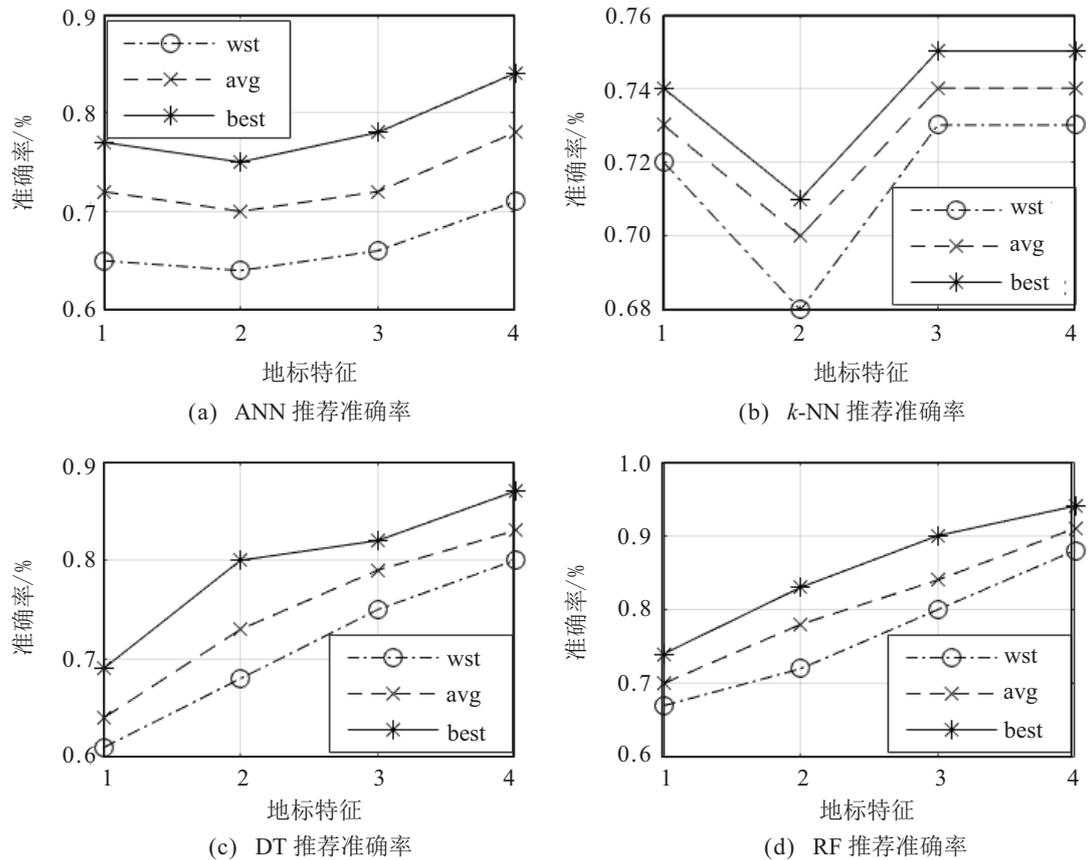


图2 4种元学习算法最差/平均/最好推荐准确率

从实验结果来看,多组独立获取的地标特征均取得了相近的推荐准确率,总体平均推荐准确率绝大多数超过70%,说明基于地标特征的优化算法推荐方法初步取得了成功。值得注意的是,除了地标特征之外,结合其他特征(例如与算法搜索过程相关的一些特征)是否有助于提升推荐准确率?是否提取的特征越多推荐效果越好?除了地标特征之外,能否通过增

加一些问题相关的简单特征或统计特征来提升推荐准确率?为此,本文设计提取了改进跨度、终止时步数、平均改进值、最大改进值和最小改进值5个特征,期望能得到好的效果。表2列出了基于这些特征分别使用4种元学习方法初步获得的推荐准确率,显然,推荐效果较差,因此本文不予采纳,留待进一步查证分析。

表2 其他特征下的推荐准确率 %

元学习方法	准确率				
ANN	42.00	43.50	43.50	44.00	42.00
k-NN	59.26	56.48	57.75	59.81	37.41
DT	48.50	47.50	50.50	48.50	43.00
RF	34.00	39.00	38.00	47.50	36.00
均值	46.00	47.00	47.00	50.00	39.60

表3 20个未见算例的推荐准确率 %

地标特征	ANN	k-NN	DT	RF
100次可行解	61.50	75.00	71.50	81.00
600次可行解	70.00	60.00	85.00	84.50
1000次迭代	59.50	45.00	86.00	88.50
固定时长	92.50	90.00	90.50	95.50

3.4.2 对未见算例的实证分析

为了实证上述元学习方法的效果,从PSPLIB算例库^[32]随机选取20个未见算例.这些算例没有参与前述训练建模环节,仅需要提取其地标特征并根据地标特征向其推荐最适用优化算法.表3是根据提取的4组地标特征,采用4种元学习方法获得的推荐准确率(取10次)均值.

针对这20个算例,加大优化算法的运行时间和迭代次数,记录目标值的计算结果并观察是否符合推荐结果.表4是20个未见算例经180s运行时间计

算后得到的实际结果.与推荐结果进行比较,在20个算例中有两个显示推荐错误(序号2和16),其余均为正确,推荐准确率90%符合预期.根据推荐算法计算的平均偏差是0.32,均小于各单独算法的偏差.最佳算法的平均偏差是0.24,二者仅相差0.08.图3给出了推荐算法与全体算法目标偏差均值结果的比较,显然,推荐算法的目标值偏差远远低于全体算法的偏差均值.图4是推荐算法与各单独算法目标值偏差的比较,可见除两个算例(2和16)外,所推荐的算法均获得了最小偏差.

表4 20个未见实例的推荐结果与实际最佳算法比较

%

序号	目标值偏差(执行时长60s)					推荐算法偏差	推荐结果	目标值偏差(执行时长180s)				最佳算法偏差	实际结果	是否准确
	ABC	ACS	PSO	TB	ABC			ACS	PSO	TB				
	1	9	4.4	2.4	2			2	4	8	3.6			
2	6	4.4	2.4	1.8	1.8	4	5	2.8	0.4	1.2	0.4	3	×	
3	4	2.4	1	0.4	0.4	4	4.4	1.6	1.6	0.4	0.4	4	√	
4	3	1.8	1.2	0.2	0.2	4	2.8	1	0.2	0	0	4	√	
5	2.8	1	1.6	0	0	4	1.6	0.8	0.8	0	0	4	√	
6	2.2	1	0.6	0	0	4	1.6	1	0.6	0	0	4	√	
7	6.6	2.4	2	1.2	1.2	4	5.6	2.2	1	0.8	0.8	4	√	
8	4.2	6.2	1.4	0	0	4	4.2	4.4	1.2	0	0	4	√	
9	2.4	0	0	0	0	2	1.8	0	0	0	0	2	√	
10	2.8	1.2	1	0	0	4	2	1	0.6	0	0	4	√	
11	0.6	0	0	0	0	2	0.6	0	0	0	0	2	√	
12	0.6	0	0	0	0	2	0.2	0	0	0	0	2	√	
13	1.6	0	0	0	0	2	0	0	0	0	0	2	√	
14	0.2	0	0	0	0	2	0	0	0	0	0	2	√	
15	5	2.6	0.8	1.2	1.2	3	4	2.4	0.4	0.4	0.4	3	√	
16	7	5.2	2.8	1.6	1.6	4	7.2	4.4	0.8	1.6	0.8	3	×	
17	3	1.8	1	0	0	4	3	1.8	1.6	0	0	4	√	
18	2.4	0	0	0	0	2	0.4	0	0	0	0	2	√	
19	1.2	0	0	0	0	2	0.6	0	0	0	0	2	√	
20	4	1.2	0.2	0	0	4	4	0.6	0.1	0	0	4	√	
均值	3.43	1.78	0.92	0.4	0.42	均值	2.85	1.38	0.57	0.3	0.24			

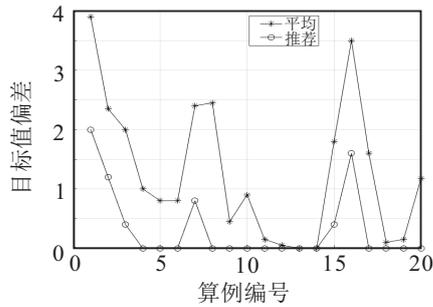


图3 推荐算法与全体算法目标偏差均值结果的比较

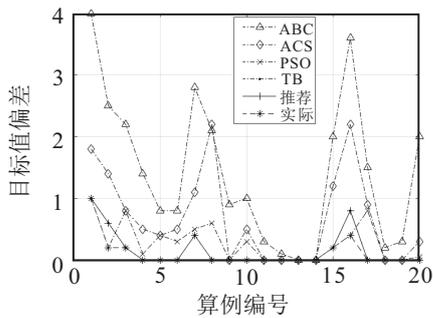


图4 推荐算法与各单独算法目标值偏差的比较

4 结论

在算法研究领域,算法选择与推荐是一项极具挑战性的前沿课题.其目标是从众多可用算法中选出最适用于当前问题的算法,实现问题-算法的“精准对接”,以提升解决问题的效果和效率.“试错法”和专家推荐法各有难以克服的弊端,而基于元学习的推荐方法值得更深入地探讨.据以往研究成果^[4,5,7,14-16],基于元学习推荐的方法最大的难点在于问题特征的提取,尤其是对于具有离散整数型特点的组合优化问题,统计特征和解空间的拓扑结构难以把握.为此,本文提出仅基于地标特征来表征问题数据集属性这一新思路,这是在全类问题特征提取的反复实验中观察到的结果并利用MRCPS数据集进行了验证.以往虽然也有文献提及与优化算法相关的地标特征^[14],但仅限于与其他问题特征的结合,更缺少具体的实证分析和详细说明.

本文创新点主要有两个:一是尝试把机器学习领域元学习方法引入到优化算法的选择;二是实现了仅基于地标特征进行优化算法的选择,并基于元学习方法获得了较高的推荐准确率.与传统的问题简单特征、统计特征和信息理论特征的复杂提取过程相比,地标特征通过运行优化算法的简化版本获得,既不需要对被优化问题的深度介入,也不需要分析搜索空间的拓扑结构.在推荐准确率没有明显降低的前提下简单易行,时空复杂度大大降低.

实验结果表明,基于地标特征和元学习方法实现优化算法的推荐取得了良好的效果.特别是采用RF

元学习方法,推荐准确率平均达到或超过90%.后续工作将进一步探讨更多的特征表现形式和提取方法,目的是如何以更小的提取代价获得更准确的推荐效果.

参考文献(References)

- [1] Boussaid I, Lepagnot J, Siarry P. A survey on optimization metaheuristics[J]. Information Science, 2013, 237: 82-117.
- [2] Wolpert D H, Macready W G. No free lunch theorems for optimization[J]. IEEE Transactions on Evolutionary Computation, 1997, 1: 67-82.
- [3] Vilalta R, Drissi Y. A perspective view and survey of meta-learning[J]. Artificial Intelligence Review, 2002, 18(2): 77-95.
- [4] Chu X H, Cai F L, Cui C, et al. Adaptive recommendation model using meta-learning for population-based algorithms[J]. Information Sciences, 2019, 476: 192-210.
- [5] 崔建双, 刘晓婵, 杨美华, 等. 基于元学习推荐的优化算法自动选择框架与实证分析[J]. 计算机应用, 2017, 37(4): 1105-1110.
(Cui J S, Liu X C, Yang M H, et al. Meta-learning based optimization algorithm selection framework and its empirical study[J]. Journal of Computer Applications, 2017, 37(4): 1105-1110.)
- [6] Burke E K, Gendreau M. Hyper-heuristics: A survey of the state of the art[J]. Journal of the Operational Research Society, 2013, 64(12): 1695-1724.
- [7] 崔建双, 车梦然. 基于多分类支持向量机的优化算法智能推荐系统与实证分析[J]. 计算机工程与科学, 2019, 41(1): 153-160.
(Cui J S, Che M R. An intelligent recommendation system for optimization algorithms based on multi-classification support vector machine and its empirical analysis[J]. Computer Engineering & Science, 2019, 41(1): 153-160.)
- [8] Wu G H, Mallipeddi R, Suganthan P N. Ensemble strategies for population-based optimization algorithms—A survey[J]. Swarm and Evolutionary Computation, 2019, 44: 695-711.
- [9] Skolicki Z, De Jong K. Improving evolutionary algorithms with multi-representation island models[C]. Parallel Problem Solving from Nature-PPSN VIII, Lecture Notes in Computer Science. Berlin, Heidelberg: Springer 2004: 420-429.
- [10] Xu L, Hutter F, Hoos H H, et al. SATzilla: Portfolio-based algorithm selection for SAT[J]. Journal of Artificial Intelligence Research, 2008, 32: 565-606.
- [11] Li K, Fialho A, Kwong S, et al. Adaptive operator

- selection with bandits for a multiobjective evolutionary algorithm based on decomposition[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(1): 114-130.
- [12] Smith-Miles K A. Cross-disciplinary perspectives on meta-learning for algorithm selection[J]. *ACM Computing Surveys*, 2009, 41(1): 1-25.
- [13] Gutierrez-Rodríguez A E, Conant-Pablos S E, Ortiz-Bayliss J C, et al. Selecting meta-heuristics for solving vehicle routing problems with time windows via meta-learning[J]. *Expert Systems with Applications*, 2019, 118(15): 470-481.
- [14] Kanda J, de Carvalho A, Hruschka E, et al. Meta-learning to select the best meta-heuristic for the traveling salesman problem: A comparison of meta-features[J]. *Neurocomputing*, 2016, 205: 393-406.
- [15] Muñoz M A, Sun Y, Kirley M, et al. Algorithm selection for black-box continuous optimization problems: A survey on methods and challenges[J]. *Information Sciences*, 2015, 317: 224-245.
- [16] Smith-Miles K A, James R J W, Giffin J W, et al. A knowledge discovery approach to understanding relationships between scheduling problem structure and heuristic performance[C]. *Learning and Intelligent Optimization, LION 2009, Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer, 2009: 89-103.
- [17] Balte A, Pise N, Kulkarni P. Meta-learning with landmarking: A survey[J]. *International Journal of Computer Applications*, 2014, 105(8): 47-51.
- [18] van Peteghem V, Vanhoucke M. An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances[J]. *European Journal of Operational Research*, 2014, 235(1): 62-72.
- [19] Harrington P. *Machine learning in action*[M]. New York: Manning Publication, 2013: 1-55.
- [20] Rice J R. The algorithm selection problem[C]. *Advances in Computers*. Amsterdam: Elsevier, 1976: 65-118.
- [21] Reyes L C, Santillán C G, Ortega J P, et al. Algorithm selection: From meta-learning to hyper-heuristics[J]. *Intelligent Systems*, 2012: 77-102.
- [22] Brazdil P, Giraud-Carrier C, Soares C, et al. *Metalearning applications to data mining*[M]. Berlin: Springer-Verlag, 2009.
- [23] Ferrari D G, De Castro L N. Clustering algorithm selection by meta-learning systems: A new distance-based problem characterization and ranking combination methods[J]. *Information Science*, 2015, 301: 181-194.
- [24] Ricardo V, Christophe G C, Pavel B, et al. Using meta-learning to support data mining[J]. *International Journal of Computer Science and Applications*. 2004, 1(1): 31-45.
- [25] Ho T K, Basu M. Complexity measures of supervised classification problems[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(3): 289-300.
- [26] Brazdil P, Soares C, Vilalta R, et al. A new data characterization for selecting clustering algorithms using meta-learning[J]. *Information Sciences*, 2018, 477: 203-219.
- [27] Malan K M, Engelbrecht A P. A survey of techniques for characterizing fitness landscapes and some possible ways forward[J]. *Information Sciences*, 2013, 241:148-163.
- [28] Czogalla J, Fink A. Fitness landscape analysis for the resource constrained project scheduling problem[C]. *3th International Conference on Learning and Intelligent Optimization*. Trento, 2009: 14-18.
- [29] Reif M, Shafait F, Goldstein M. Automatic classifier selection for non-experts[J]. *Pattern Analysis and Applications*, 2014, 17(1): 83-96.
- [30] Bensusan H, Girard-Carrier C. Discovering task neighborhoods through landmark learning performance[C]. *Principles of Data Mining and Knowledge Discovery*. Lyon: Springer, 2002: 325-330.
- [31] 曾子林, 张宏军, 张睿, 等. 基于元学习思想的算法选择问题综述[J]. *控制与决策*, 2014, 29(6): 961-968. (Zeng Z L, Zhang H J, Zhang R, et al. Summary of algorithm selection problem based on meta-learning[J]. *Control and Decision*, 2014, 29(6): 961-968.)
- [32] Kolisch R, Sprecher A. PSPLIB-A project scheduling problem library[J]. *European Journal of Operational Research*, 1997, 96(1): 205-216.
- [33] Jarbouli B, Damak N, Siarry P, et al. A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems[J]. *Applied Mathematics and Computation*, 2008, 195: 299-308.
- [34] Grabczewski K. *Meta-learning in techniques of decision tree induction*[M]. New York: Springer, 2014: 18-25.
- [35] Quinlan J R. *Induction of decision trees*[J]. *Machine Learning*, 1986, 1(1): 81-106.

作者简介

崔建双(1961—), 男, 副教授, 博士, 从事智能优化算法、项目优化调度等研究, E-mail: cuijs@manage.ustb.edu.cn;

吕玥(1996—), 女, 硕士生, 从事智能优化算法、项目优化调度的研究, E-mail: lvyue87@126.com;

徐子涵(1995—), 女, 硕士生, 从智能优化算法、项目优化调度的研究, E-mail: julia1995@126.com.

(责任编辑: 孙艺红)