

【电子与信息科学 / Electronics and Information Science】

基于DDQN的否定约束规则迁移

秦建斌, 杜玉琪, 林毅斌

深圳大学计算机与软件学院, 广东深圳 518060

摘要: 传统的数据清洗方法因需专家手动定义数据质量规则, 不仅复杂且耗时巨大, 且清洗后的数据可能不能被重复利用, 降低了数据清洗的质量和效率. 为此, 提出基于双重深度Q网络的否定约束迁移(double deep Q-network for denial constraints transfer, DDQN-DCT)算法. 该算法设计否定约束(denial constraint, DC)的相似性度量方法, 并结合相似性和DC的简洁性和覆盖度, 使用双重深度Q网络(double deep Q-network, DDQN)对DC规则中的谓词进行修改从而实现DC迁移, 目标是使迁移后的规则与原规则具有最大的相似性, 以保留原有规则的信息. 基于DDQN-DCT进一步设计了DDQN-DCT+算法, 把DDQN的动作选择策略划为增加和删除2个阶段, 并通过对比实验证明DDQN-DCT+在DC简洁性上有更好的表现. 通过与暴力依赖约束迁移(brute-force dependency constraint transfer, BFDC)、DDQN-DCT+和结构扩大/缩小(structure expansion/reduction, SER)方法进行对比实验, 发现DDQN-DCT算法在规则相似度上相较于BFDC平均提升约10%, 相较于DDQN-DCT+平均提升约10.6%, 相较于SER平均提升约16.4%. DDQN-DCT能够有效地将源域的规则迁移到相似的目标域数据上.

关键词: 计算机技术; 规则迁移; 否定约束; 相似度量; 强化学习; 数据清洗

中图分类号: TP399; TP181

文献标志码: A

DOI: 10.3724/SP.J.1249.2025.02242

A DDQN framework for denial constraints transfer

QIN Jianbin, DU Yuqi, and LIN Yibin

College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, Guangdong Province, P. R. China

Abstract: Traditional data cleaning methods rely on experts to manually define data quality rules, making the process complex and time-consuming. Moreover, the cleaned data may not be reusable, reducing the efficiency and quality of data cleaning. To address this issue, we propose a double deep Q-network for denial constraints transfer (DDQN-DCT) algorithm. This algorithm designs a similarity measurement method for denial constraints (DCs) and, by considering the simplicity and coverage of DCs, modifies the predicates in DC rules using double deep Q-network (DDQN) to achieve DC transfer. The goal is to make the transferred rules as similar as possible to the original rules, thereby retaining the information of original rules. Building upon DDQN-DCT, we further design an enhanced algorithm, DDQN-DCT+, which separates the action selection strategy of DDQN into two stages: addition and deletion. Comparative experiments demonstrate that DDQN-DCT+ performs better in terms of DC simplicity. In comparison with algorithms of brute-force dependency constraint transfer (BFDC), DDQN-DCT+, and structure expansion/reduction (SER), the DDQN-DCT algorithm achieves an improvement of average rule similarity by approximately 10% over BFDC, 10.6% over DDQN-DCT+, and 16.4% over SER. The experiments show that DDQN-DCT can effectively transfer rules from the source domain to similar target domain data.

Key words: computer technology; rule transfer; denial constraints; similarity metric; reinforcement learning; data cleaning

Received: 2024-09-09; **Accepted:** 2024-12-10; **Online (CNKI):** 2025-01-22

Foundation: National Key R & D Program of China (2021YFB3301500)

Corresponding author: Professor QIN Jianbin (qinjianbin@szu.edu.cn)

Citation: QIN Jianbin, DU Yuqi, LIN Yibin. A DDQN framework for denial constraints transfer [J]. Journal of Shenzhen University Science and Engineering, 2025, 42(2): 242-248. (in Chinese)



数据清洗是数据处理过程中查找和纠正数据错误的关键步骤, 包括检查数据一致性和处理无效值或缺失值^[1-2], 可确保实验研究的准确性. 研究表明, 数据科学家将80%的时间花在了数据清洗任务上^[3]. 传统的数据清洗方法需先人工定义数据质量规则, 再依据数据质量规则清洗脏数据. 然而, 规则的发现和制定需耗费大量人力, 且因其中的一致性约束、函数依赖^[4]和否定约束^[5](denial constraint, DC)未被复用, 导致数据清洗效率低. 尽管已有学者探索出了挖掘数据质量规则的方法^[6-7], 然而在实际应用中, 采用这些方法提取出的规则不仅数量庞大且质量参差不齐. 若数据来源和结构发生变化, 还可能令现有规则失效. 目前关于数据质量规则的研究主要集中于规则挖掘^[8-10]. 规则可以以决策树、结构化查询语言(structured query language, SQL)和逻辑表达式等形式存储, 并使用机器学习和本体图^[11-12]的技术来融入领域知识.

规则迁移的研究主要分为两个方向: ① 跨不同领域调整规则以保持规则的相关性和有效性; ② 在动态环境中优化规则以保持其准确性. 在跨领域调整规则方面, MINVIELLE等^[13]提出了结构扩大/缩小(structure expansion/reduction, SER)和结构迁移(structure transfer, STRUT)两种不同的基于决策树的迁移方法. 然而, 这些规则有时缺乏明确的意义. JAROVSK等^[14]提出的GOLDRUSH系统针对欺诈检测, 通过在不同环境间平滑地调整规则来增强规则共享, 并使用本体图来优化规则, 但该算法针对的是单一规则场景.

在对动态环境中规则优化的研究中, 用于欺诈检测的交互式规则完善算法(interactive rule refinement system for fraud detection, RUDOLF)^[15]是一个能够通过领域专家互动更新规则, 以适应变化的条件. TABEBORDBAR等^[16]提出一种基于特征的规则适应算法, 利用贝叶斯多臂老虎机算法来确定规则的最佳变更. ZHAO等^[17]开发了一种利用强化学习创建和演化规则的方法. LI等^[18]提出了一种从宽泛规则开始, 通过添加具体条件逐步缩小范围的方法. 基于规则的增量学习器(rule-based incremental learner, RILL)是一个针对流数据的基于概念漂移的规则生成算法^[19]. LIU等^[20]使用数据库溯源技术来修改提取规则, 以消除提取结果中的假阳性. OMRAN等^[21]探索了如何在知识图谱中迁移规则. 但是, 以上方法考虑的都是如何优化现有的规则挖

掘方法, 并没有考虑到对现有规则的运用.

本研究提出基于双重深度Q网络^[22]的否定约束迁移(double deep Q-network framework for denial constraints transfer, DDQN-DCT)算法, 利用强化学习^[23]对DC规则进行迁移, 提高了规则的迁移前后的相似度.

1 基于DDQN的DC迁移

首先定义DC迁移的目标函数.

定义1(规则迁移目标函数) 给定源数据 S 及其否定约束规则 R 和目标数据 T , R 可以经过一系列修改使得 T 中的元组对满足这些规则的数量在用户给定的阈值 g 内, 并且修改后的规则 R' 与原规则 R 具有最大相似度.

$$\varphi' = \underset{\varphi'}{\operatorname{argmax}} S(\varphi, \varphi') \quad (1)$$

$$\text{st. match}(\varphi') \leq g \quad (2)$$

其中, φ' 为迁移后的目标数据否定约束规则; $S(\varphi, \varphi')$ 为迁移前源数据的规则 φ 和迁移后规则 φ' 的相似度, 式(1)和式(2)的目标是找到能够使得单个迁移前的否定约束规则 φ 经过修改后与 φ 相似度最大的 φ' ; $\text{match}(\varphi')$ 为 T 中满足当前规则的元组对数量与 T 中元组对总数的比值; g 为 φ' 在目标数据中必须满足的元组对的阈值, $0 \leq g < 1$, $g = 0$ 表示 φ' 是一个严格的DC, 即目标数据中的所有元组对都必须遵守该DC.

1.1 DDQN-DCT算法

图1(a)展示了基于双深度Q网络(double deep Q-network, DDQN)的DC迁移算法的计算流程. DDQN-DCT算法将源域的规则迁移到目标域中, 对不符合目标域数据分布的规则使用预训练的智能体, 对规则进行修改.

将获得的规则与原始规则之间的相似性作为DDQN奖励函数的一部分, 确保迁移后的规则与原始规则具有最大的相似性. 相似性计算如图1(b)所示. 其中, $r(A_i, A_j)$ 为属性 A_i 和属性 A_j 的相关性; W_{attrs} 为属性集的权重; W_{ops} 为操作符的权重; 相似性指数 J_s 为杰卡德距离(Jaccard distance)^[24]. 首先, 计算源域数据中属性间的相关性和目标域数据中属性间的相关性, 以确定反映属性相关性差异的属性权重; 然后将结合属性权重与操作符权重计算谓词权重; 最后应用基于原始规则和迁移后规则的属性组合得出相似性指数 J_s , 利用 J_s 确保迁移后的规则

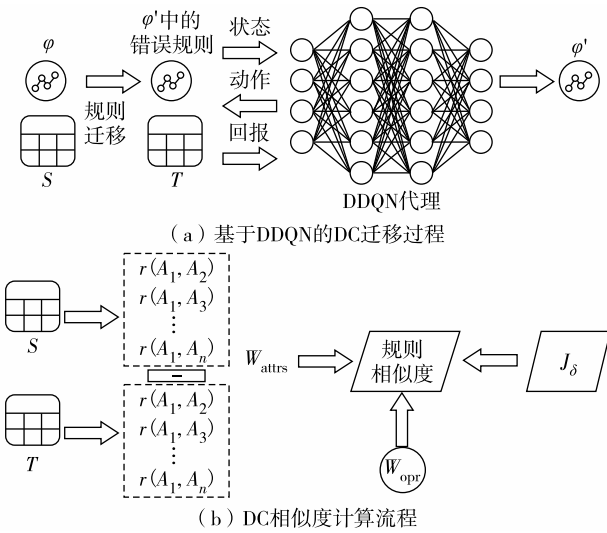


图1 DDQN-DCT算法流程

Fig. 1 The DDQN-DCT algorithm framework includes two main processes. (a) DDQN-based migration process for DCs and (b) DCs similarity calculation process.

与原始规则紧密对应。

1.2 DDQN环境

1) 状态. DC规则由多个谓词组成, 这些谓词属于目标数据 T 的谓词空间 D . 谓词空间使用 BELLMAN^[25] 进行构建. 为提高DC规则的适用性, 从 D 中提取新的谓词来替代或增强规则中的现有谓词, 以确保规则在近似度范围内满足所有元组对. 规则状态由 D 中的每个谓词的选择状态定义. 因此, 状态的长度等于 $|D|$. 在这种表示方法中, 构成规则的谓词通过将其在状态空间中的相应位置设置为1来表示, 而未选择的谓词则标记为0.

2) 动作. 本研究定义 $2^{|D|}$ 种不同的动作. 每个谓词都有添加和删除两种动作, 一旦智能体执行某个动作, 状态就会发生转移.

3) 奖励. 在智能体训练过程中, 提供奖励以促进有效学习至关重要. 本研究基于规则相似性设置奖励, 即

$$E = 0.5 \times S(\varphi, \varphi') + 0.3 \times C(\varphi) + 0.2 \times f_{\text{suc}}(\varphi) \quad (3)$$

其中, C 为规则覆盖度函数; f_{suc} 为计算规则简洁度函数. 函数奖励结合了相似性与简洁性和覆盖度指标. 规则迁移的关键在于如何在保留原有语义的基础上, 利用规则迁移前后的相似性作为评价奖励指标最大化地保留原有的语义. 每个指标都赋予相等的权重, 以最大化迁移规则与原始规则之间的相似性, 同时保持简洁性和覆盖度. DC的简洁性和覆

盖度也是评估规则质量的重要因素. DC的简洁性、覆盖度计算方法以及它们的权重比例在文献[5]中有描述, 这里不再赘述.

在训练设置中, 智能体每次迭代都是从随机生成的规则开始. 智能体使用 ε -贪婪策略选择动作, 以平衡探索和利用. 当智能体选择无效动作时, 会按概率选择下一个动作, 直到找到有效动作.

考虑到规则迁移训练时经验池中通常存在负样本主导的场景, 此时智能体从经验池中抽取的经验大多倾向于负面结果. 因此, 采用优先经验回放策略^[26], 允许智能体根据加权偏好来提取经验, 增加智能体获得有效经验的概率.

1.3 DC相似性度量

DC规则反映了元组对中各属性之间的关系, 两数据集间属性相关性的差异越大, 与这些属性关联的谓词需要修改的可能性就越高. 因此, 本研究利用属性相关性的差异对DC的相似性进行度量.

1) 属性权重. 设源数据具有属性集合 $A = \{A_1, A_2, \dots, A_n\}$, 目标数据具有属性集合 $A' = \{A'_1, A'_2, \dots, A'_n\}$. 每个属性与其他属性之间的相关性为 $r(A_i, A_j)$, $i \neq j$, $i, j = 0, 1, \dots, n$. A_i 的权重 W_{A_i} 是 A_i 与其他属性的属性对之和除以总属性数量, 即

$$W_{A_i} = \frac{1}{n} \sum_{j=0, j \neq i}^n W(A_i, A_j) \quad (4)$$

$$W(A_i, A_j) =$$

$$\frac{\min \left\{ \left| r_s(A_i, A_j) - r_t(A_i, A_j) \right| \right\}}{\left| r_s(A_i, A_j) - r_t(A_i, A_j) \right|} \quad (5)$$

属性的权重越小, 该属性需要修改的可能性就越高.

$$r(A_i, A_j) = \frac{\sum_{k=1}^m (A_{i_k} - \bar{A}_i)(A_{j_k} - \bar{A}_j)}{\sqrt{\sum_{k=1}^m (A_{i_k} - \bar{A}_i)^2} \sqrt{\sum_{k=1}^m (A_{j_k} - \bar{A}_j)^2}} \quad (6)$$

根据属性对的数据类型, 属性对 $\langle A_i, A_j \rangle$ 的相关性 $r(A_i, A_j)$ 有以下3种计算方法:

① 数值-数值型属性对采用式(6)来衡量属性间的相关性. 其中, A_{i_k} 和 A_{j_k} 分别为属性 i 和属性 j 的第 k 个值; $\bar{A}_i$ 和 $\bar{A}_j$ 分别为属性 i 和属性 j 的均值.

② 字符-字符型属性对需先构建属性对的交叉表, 再根据观测频数和期望频数计算卡方统计量, 并以卡方统计量的 p 值作为属性对相关性的衡量指标.

③ 字符-数值型属性对先将字符型属性转化为整型编码, 再采用式(6)计算其皮尔逊相关系数。

2) 操作符权重: 谓词操作中不同符号的选择性不同^[27], 因此, 本研究为运算符分配特定的优先级, 操作符“≠”在谓词中通常能满足大部分的元组对, 具有较低的选择行, 因此具有较高优先级, 相反操作符“=”具有较高的选择行, 因此具有较低的优先级, 操作符“>”和操作符“<”的选择性相同, 操作符“≤”和操作符“≥”选择性相同。因此, 给定操作符优先级函数为

$$P(o) = \begin{cases} 1, & o \text{ 为操作符“} \neq \text{”} \\ 2, & o \text{ 为操作符“} > \text{”或“} < \text{”} \\ 3, & o \text{ 为操作符“} \geq \text{”或“} \leq \text{”} \\ 4, & o \text{ 为操作符“} = \text{”} \end{cases} \quad (7)$$

则每个操作符权重计算公式为

$$W_o = \frac{P(o)}{\max(P(o) | o \in O)} \quad (8)$$

其中, O 为所有操作符的集合。 o 优先级越高, W_o 值越小, 被选几率越大。

3) 谓词权重. 谓词 d 的权重定义为属性权重和运算符权重的线性加权和, 即

$$W_d = \alpha W_{d_o} + (1 - \alpha) W_{d_A} \quad (9)$$

其中, W_{d_o} 为谓词空间里任意谓词 d 中所有运算符的权重之和; W_{d_A} 为 d 中所有属性的权重之和; 加权因子 $\alpha \in (0, 1)$ 。本研究主要关注属性间相关性与谓词的关系, 因此设 $\alpha = 0.3$ 以提高属性权重的影响。

4) DC规则相似度. 给定DC规则 r_1 和 r_2 , 除了根据谓词权重选择合适的谓词, 规则的相似度还和属性集合的差异有关。本研究认为, 两个规则越相似, 它们的共同元素越多, 因此, 结合杰卡德距离^[24]来共同定义两个规则的相似性。若 r_1 和 r_2 的谓词权重之和 $W_{d_{r_1}}$ 和 $W_{d_{r_2}}$ 的差异为0, 且属性集合一致, 则两个规则的DC距离为0, DC的相似度为1。 r_1 和 r_2 的DC距离和DC相似度计算公式为

$$d(W_{d_{r_1}}, W_{d_{r_2}}) = \frac{|W_{d_{r_1}} - W_{d_{r_2}}|}{\max(W_{d_{r_1}}, W_{d_{r_2}})} \quad (10)$$

$$S(r_1, r_2) = \omega \times \left[1 - d(W_{d_{r_1}}, W_{d_{r_2}}) \right] + (1 - \omega) \times J_\delta(r_1, A, r_2, A') \quad (11)$$

其中, ω 为加权因子, $\omega \in (0, 1)$; $r_1.A$ 为 r_1 的属性集; $r_2.A'$ 为 r_2 的属性集; $d(W_{d_{r_1}}, W_{d_{r_2}})$ 为规则距离计

算函数; $J_\delta(r_1, A, r_2, A')$ 为 $r_1.A$ 和 $r_2.A'$ 的Jaccard距离; $\max(W_{d_{r_1}}, W_{d_{r_2}})$ 为选择谓词权重之和的最大值, 旨在计算两条规则之间的谓词差异率。本研究更加关注属性集合差异对规则相似度的影响, 因此设 $\omega = 0.4$ 。

1.4 DDQN-DCT+算法

对DDQN-DCT算法进行优化, 进而提出DDQN-DCT+算法。

在DDQN网络中, 对动作的选择策略进行进一步细化, 把动作选择策略划为两个阶段, 智能体在动作选择过程中, 会优先考虑增加谓词的操作。一旦当前状态下的谓词组合能够使满足的元组对数目不超过用户设定的阈值, 智能体便转变其动作选择策略, 专注于执行删除谓词的动作, 以此来减少谓词冗余。在整个过程中, 智能体调整其动作选择策略是一种动态的过程, 这一过程依赖于当前谓词组合所满足的元组对的集合大小。

2 实验与分析

实验在Ubuntu 20.04服务器上进行, CPU为Intel(R) Xeon(R) Gold 6326 CPU @ 2.90 GHz, 共含有32个核心, 服务器内存大小为128 Gbyte。

2.1 数据集

实验所用数据集均来自UC Irvine Machine Learning Repository和Kaggle数据平台(<https://www.kaggle.com/>)。选择其中的wine、mushroom、cancer、cardiopathy和retail数据集, 并将每个数据集拆分为源数据和目标数据, 如表1。

表1 数据集介绍

Table 1 The introduction of the datasets

数据集	元组数	属性数	谓词数	源域数据	目标域数据
wine	6 497	13	70	白酒	红酒
mushroom	8 124	22	42	可食用	有毒
cancer	569	32	180	恶性	良性
cardiopathy	68 205	17	84	患有心脏病	未患心脏病
retail	1 000	14	76	有客户流失	无客户流失

2.2 对比算法

使用DDQN-DCT、DDQN-DCT+、SER^[17]和暴力依赖约束迁移(brute-force dependency constraint transfer, BFDC)算法分别将规则从源域数据中迁移

至目标域数据中,并对比迁移后规则的简洁性、覆盖度以及与原规则的相似度.其中, SER 是一种基于决策树的转移算法.该算法寻找决策树的结构更改的局部最优解,把 DC 细化到目标数据中并把归纳 DC 的进行泛化,完成 DC 迁移.实验将该算法设为基准算法; DDQN-DCT+ 是 DDQN-DCT 的优化版本,采用了两阶段的动作选择策略,优先考虑增加谓词的操作,然后再考虑删除谓词的操作; BFDC 是直接将源数据规则迁移到目标数据上,在目标数据谓词空间进行搜索,对规则进行各种组合,再将找到的首个匹配结果作为该规则的迁移结果返回的启发式算法.

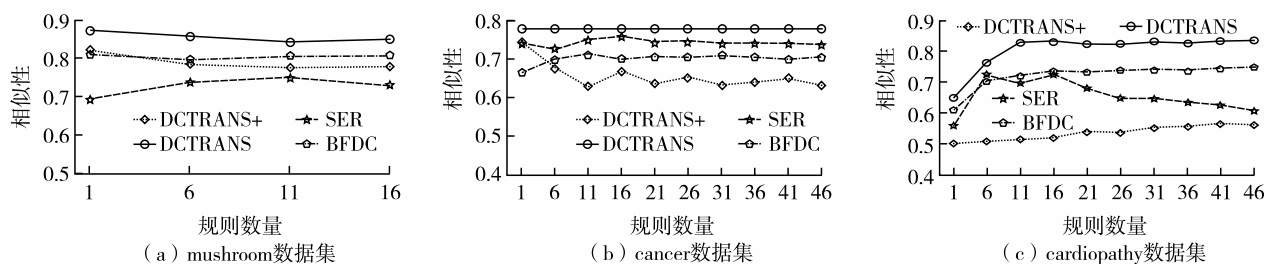


图2 Mushroom、cancer 和 cardiopathy 数据集上不同算法转移规则与原始规则之间的相似性曲线

Fig. 2 The similarity between transferred and original rules for different algorithms on dataset of (a) mushroom, (b) cancer, and (c) cardiopathy.

表2给出了4种算法在5个数据集上得到的规则相似度、简洁度和覆盖度的平均表现.由表2可见, DDQN-DCT算法在相似度指标上表现最优,而在简洁度指标上, DDQN-DCT+算法的表现要优于其他算法,这是因为其动作选择策略使结果更偏向于得到简洁度高的规则; SER算法得到的规则覆盖度较高,这可能得益于其利用决策树迁移的优势,保留了部分相似结构,对规则进行增加谓词的修改,使得规则覆盖度提升; BFDC算法在简洁度和覆盖度的表现上都较差,因为它是根据原规则的谓词组合来随机匹配,并不进行谓词去冗余的操作; DDQN-DCT算法在相似性上表现出色,在简洁性和覆盖率

2.3 评估指标

采用原规则与迁移后规则的相似度以及规则的简洁性和覆盖度,多角度全方面评估所衍生规则的质量和算法的性能.

2.4 结果分析

图2给出了在 mushroom、cancer 和 cardiopathy 数据集上,不同规则数量下,4种算法的迁移规则的相似性.由图2可见, DDQN-DCT算法在3个数据集上的相似性最高达87%,均优于其他算法, DDQN-DCT+算法的表现较差,原因可能是该算法优先考虑规则的简洁性.极简规则可能在去除不必要的复杂性时损失了一些关键信息,影响了规则的相似性.

方面仅次于 DDQN-DCT+ 和 SER 算法,这表明 DDQN-DCT算法在生成目标数据的规则时,能够有效保留原始规则的信息,同时保持规则的可解释性和全面性,在相似性、简洁性和覆盖率之间取得了平衡.

图3展示了属性数量如何影响不同算法在规则迁移中的运行时间,尤其是在属性较为丰富的癌症数据集上的表现.随着属性数量的增加, SER 和 BFDC 算法的运行时间显著增长,这反映了它们在处理更大谓词空间时所需的计算资源增加.与此相比, DDQN-DCT 和 DDQN-DCT+ 算法在属性数量增加时表现出了更好的稳定性,运行时间相对平缓.

表2 不同数据集上4种算法的性能表现

Table 2 Comparison of metrics for different methods

算法	wine 数据集			mushroom 数据集			retail 数据集			cancer 数据集			cardiopathy 数据集		
	相似度	简洁度	覆盖度	相似度	简洁度	覆盖度	相似度	简洁度	覆盖度	相似度	简洁度	覆盖度	相似度	简洁度	覆盖度
SER	0.67	0.30	0.47	0.72	0.43	0.65	0.51	0.43	0.58	0.74	0.43	0.60	0.62	0.51	0.77
BFDC	0.50	0.41	0.42	0.81	0.40	0.59	0.81	0.32	0.59	0.71	0.44	0.60	0.75	0.36	0.62
DDQN-DCT+	0.71	0.70	0.41	0.78	0.60	0.63	0.53	0.71	0.60	0.63	0.59	0.57	0.56	0.71	0.39
DDQN-DCT	0.74	0.59	0.43	0.85	0.55	0.61	0.87	0.46	0.51	0.78	0.69	0.56	0.84	0.40	0.48

注: 灰底数据为该指标下的最优值.

由于DDQN-DCT+产生了更多的重复规则,因此时间消耗没有显著增加.总体而言,DDQN-DCT算法展现了灵活性和稳定性,在不同属性数量下维持了一致的运行时间.

DDQN-DCT算法不仅在规则度量中优于对比算法,还在近似DC迁移方面表现出色.图4(a)展示了相似度随近似度变化的趋势,在这5个数据集上,相似度随近似度的增加呈先降后升的变化过程.在低近似度时,相似度下降,可能是由于算法需要精确匹配原始规则和目标域的数据,而这在某些情况下可能较为困难.然而,随着近似度的进一步增加,算法可更宽泛地匹配规则,允许更多的变

化,这时相似度又逐渐提升.这意味着在高近似度下,DDQN-DCT算法能够在较小程度上修改原始规则,使其更适合目标域的数据.

图4(b)展示了随着近似度的变化,在不同数据集上DDQN-DCT算法的运行时间变化.总体来看,DDQN-DCT算法在时间上的表现相对平稳,尽管在蘑菇数据集上,算法的运行时间在低近似度时略有增加.这可能反映了在低近似度下,为寻找与目标域严格匹配的谓词,算法需要进行更多的搜索,这自然会增加运行时间.与此同时,其他数据集的时间性能变化不大,表明DDQN-DCT算法在不同的数据集上具有稳定的时间性能,尤其是在那些复杂度较低的数据集上.

结 语

提出一个新颖的基于深度强化学习的否定约束的迁移框架DDQN-DCT.基于DC迁移问题构建了DDQN的迁移环境,设计状态动作空间以及回报值的变化策略,并基于DC的特性设计了DC相似性度量.在多个数据集上进行的实验结果表明,DDQN-DCT算法与基准算法的对比证明了DDQN-DCT算法的高效率和有效性.DDQN-DCT算法在DC迁移中,能使得到的DC相似度达到87.5%,并且能够在覆盖度和简洁度之间进行有效地平衡.

未来可探索集成更多先进深度学习的架构,或结合注意力机制以更精准地捕捉和映射源域与目标域之间的内在规律和特征关联,也可结合其他形式的领域适应性技术,对跨域数据的相关性进行更深层次的理解与建模,提出更通用、更便捷的迁移方法.

基金项目:国家重点研发计划资助项目(2021YFB3301500)

作者简介:秦建斌(qinjianbin@szu.edu.cn),深圳大学教授、博士生导师.研究方向:数据库系统、高维向量查询和数据治理.

引文:秦建斌,杜玉琪,林毅斌.基于DDQN的否定约束规则迁移[J].深圳大学学报理工版,2025,42(2):242-248.

参考文献 / References:

- [1] CHU Xu, ILYAS I F, PAPOTTI P. Holistic data cleaning: putting violations into context [C]// The 29th International Conference on Data Engineering. Piscataway, USA: IEEE, 2013: 458-469.
- [2] REKATSINAS T, CHU Xu, ILYAS I F, et al. HoloClean: holistic data repairs with probabilistic inference [J]. Proceedings of the VLDB Endowment, 2017, 10(11): 1190-

<http://journal.szu.edu.cn>

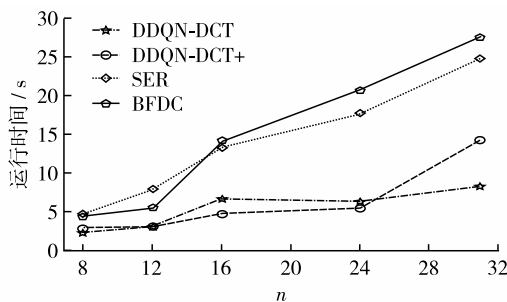
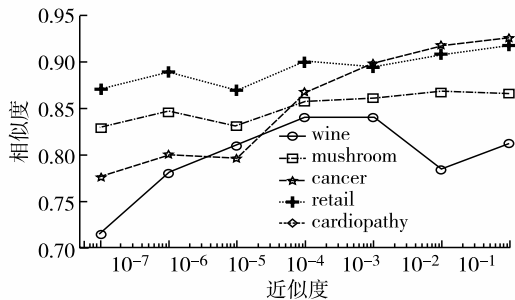
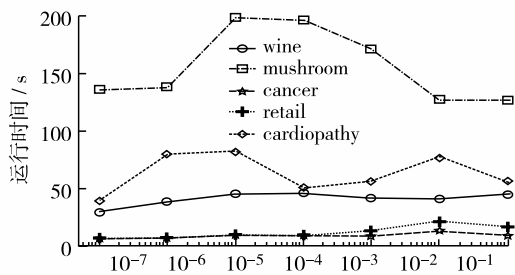


图3 Cancer数据集上不同算法的属性数量对运行时间的影响

Fig. 3 Runtime of various methods when using different number of attributes on the cancer dataset.



(a) 相似性



(b) 运行时间

图4 DDQN-DCT算法在不同近似条件下的相似性和运行时间

Fig. 4 (a) Similarity and (b) time efficiency at various approximations by DDQN-DCT.

- 1201.
- [3] DATA C B. Most time-consuming, least enjoyable data science task, survey says [J]. *Forbes*. (2016-03-23) [2024-08-01]. <https://www.forbes.com/sites/gilpress/2016/03/23/data-preparation-most-time-consuming-least-enjoyable-data-science-task-survey-says/>.
 - [4] BOHANNON P, FAN Wenfei, GEERTS F, et al. Conditional functional dependencies for data cleaning [C]// The 23rd International Conference on Data Engineering. Piscataway, USA: IEEE, 2007: 746-755.
 - [5] CHU Xu, ILYAS I F, PAPOTTI P. Discovering denial constraints [J]. *Proceedings of the VLDB Endowment*, 2013, 6(13): 1498-1509.
 - [6] CHU Xu, ILYAS I F, PAPOTTI P, et al. RuleMiner: data quality rules discovery [C]// The 30th International Conference on Data Engineering. Piscataway, USA: IEEE, 2014: 1222-1225.
 - [7] LIU Jixue, LI Jiuyong, LIU Chengfei, et al. Discover dependencies from data: a review [J]. *IEEE Transactions on Knowledge and Data Engineering*, 2012, 24(2): 251-264.
 - [8] AGRAWAL R, SRIKANT R. Fast algorithms for mining association rules in large databases [C]// *Proceedings of the 20th International Conference on Very Large Data Bases*. San Francisco, USA: Morgan Kaufmann Publishers Inc., 1994: 487-499.
 - [9] LIU Bing, HSU W, MA Yiming. Integrating classification and association rule mining [C]// *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining*. Palo Alto, USA: AAAI Press, 1998: 80-86.
 - [10] CHITICARIU L, LI Yunyao, REISS F R. Rule-based information extraction is dead! Long live rule-based information extraction systems! [C]// *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, USA: ACL, 2013: 827-832.
 - [11] SUCHANEK F M, KASNECI G, WEIKUM G. Yago: a core of semantic knowledge [C]// *Proceedings of the 16th International Conference on World Wide Web*. New York, USA: ACM, 2007: 697-706.
 - [12] DBpedia. Global and unified access to knowledge graphs [EB/OL]. [2024-09-05]. <https://www.dbpedia.org/>.
 - [13] MINVIELLE L, ATIQ M, PEIGNIER S, et al. Transfer learning on decision tree with class imbalance [C]// The 31st International Conference on Tools with Artificial Intelligence. Piscataway, USA: IEEE, 2019: 1003-1010.
 - [14] JAROVSKY A, MILO T, NOVGORODOV S, et al. Rule sharing for fraud detection via adaptation [C]// The 34th International Conference on Data Engineering. Piscataway, USA: IEEE, 2018: 125-136.
 - [15] MILO T, NOVGORODOV S, TAN W C. Interactive rule refinement for fraud detection [C]// *Proceedings of the 21st International Conference on Extending Database Technology*. Konstanz, Germany: OpenProceedings, 2018: 265-276.
 - [16] TABEBORDBAR A, BEHESHTI A, BENATALLAH B, et al. Feature-based and adaptive rule adaptation in dynamic environments [J]. *Data Science and Engineering*, 2020, 5(3): 207-223.
 - [17] ZHAO Tianqi, ZHANG Wei, ZHAO Haiyan, et al. A reinforcement learning-based framework for the generation and evolution of adaptation rules [C]// *International Conference on Autonomic Computing*. Piscataway, USA: IEEE, 2017: 103-112.
 - [18] LI Huan, LI Xin, HU Dawei, et al. Adaptation rule learning for case-based reasoning [J]. *Concurrency and Computation Practice and Experience*, 2009, 21(5): 673-689.
 - [19] DECKERT M, STEFANOWSKI J. RILL: algorithm for learning rules from streaming data with concept drift [C]// *Foundations of Intelligent Systems*. Cham: Springer International Publishing, 2014: 20-29.
 - [20] LIU Bin, CHITICARIU L, CHU V, et al. Automatic rule refinement for information extraction [J]. *Proceedings of the VLDB Endowment*, 2010, 3(1/2): 588-597.
 - [21] OMRAN P G, WANG Zhe, WANG Kewen. Knowledge graph rule mining via transfer learning [C]// *Advances in Knowledge Discovery and Data Mining*. Cham: Springer International Publishing, 2019: 489-500.
 - [22] VAN HASSELT H, GUEZ A, SILVER D. Deep reinforcement learning with double Q-learning [C]// *Proceedings of the 13th AAAI Conference on Artificial Intelligence*. Palo Alto, USA: AAAI Press, 2016: 2094-2100.
 - [23] LI Yuxi. Deep reinforcement learning: an overview [EB/OL]. (2017-01-25) [2018-11-26]. <https://arxiv.org/abs/1701.07274>.
 - [24] Anon. Jaccard index [EB/OL]. [2024-09-05]. https://en.wikipedia.org/wiki/Jaccard_index.
 - [25] DASU T, JOHNSON T, MUTHUKRISHNAN S, et al. Mining database structure; or, how to build a data quality browser [C]// *Proceedings of the 2002 ACM SIGMOD International Conference on Management of Data*. New York, USA: ACM, 2002: 240-251.
 - [26] SCHAUL T, QUAN J, ANTONOGLOU I, et al. Prioritized experience replay [EB/OL]. (2015-11-18) [2014-02-25]. <https://arxiv.org/abs/1511.05952>.
 - [27] PENA E H M, De ALMEIDA E C, NAUMANN F. Discovery of approximate (and exact) denial constraints [J]. *Proceedings of the VLDB Endowment*, 2019, 13(3): 266-278.

【中文责编：英子；英文责编：木柯】