

基于域与样例平衡的多源迁移学习方法

季鼎承, 蒋亦樟, 王士同

(江南大学数字媒体学院, 江苏无锡 214122)

摘 要: 针对如何有效使用多源域的决策知识去预测目标域样例标签的问题, 提出一种平衡域与样例信息的多源迁移学习算法. 为实现上述目的, 本文提出了一种基于域与样例平衡的多源迁移学习方法 (Multi-source Transfer Learning by Balancing both Domains and Instances, MTL-BDI). 该方法的基本思想是将域层面和样例层面的双加权平衡项嵌入到迁移学习的原始目标函数中, 然后利用交替优化技术对提出的目标函数进行有效求解. 在文本和图像数据集上的大量实验表明, 该方法在分类精度方面确实优于现有的多源迁移学习方法 MCC-SVM (Multiple Convex Combination of SVM)、A-SVM (Adaptive SVM)、Multi-KMM (Multiple Kernel Mean Matching) 和 DAM (Domain Adaptation Machine).

关键词: 迁移学习; 多源域适应; 模糊 C 均值聚类; 实例加权; Universum 学习; 支持向量回归; 交替优化

中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2019)03-0692-08

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2019.03.025

Multi-source Transfer Learning Method by Balancing both the Domains and Instances

Ji Ding-cheng, Jiang Yi-zhang, Wang Shi-tong

(School of Digital Media, Jiangnan University, Wuxi, Jiangsu 214122, China)

Abstract: When transfer learning attempts to leverage the decision knowledge effectively from multiple source domains to predict the labels of instances accurately in target domain, it should consider how to well balance source and target domains, and their instances in both domains. In this paper, a novel multi-source transfer learning method called mlti-source transfer learning by balancing both domains and instances (MTL-BDI) is proposed to achieve the above goal. The basic idea of the proposed method is to embed the doubly weighted domain-level and instance-level balance term into the original objective function of transfer learning and then solve the proposed objective function effectively by using the alternating optimization technique. Extensive experiments on text and image datasets indicate that the proposed method indeed outperforms several existing multi-source transfer learning methods MCC-SVM (Multiple Convex Combination of SVM), A-SVM (Adaptive SVM), Multi-KMM (Multiple Kernel Mean Matching) and DAM (Domain Adaptation Machine) in the sense of classification accuracy on target domain.

Key words: transfer learning; multi-source domain adaptation; fuzzy C-means clustering; instance weighting; universum learning; support vector regression; alternative optimization

1 引言

单源迁移学习在单一源域和目标域相似但不同的学习场景中取得了许多成果. 然而, 现实中对目标数据的学习, 往往存在多种、多样的源域数据, 如果仅利用单个源域信息来学习, 不但限制了学习算法的性能, 而且造成了学习资源的浪费. 因此, 近来一些迁移学习方法^[1-3]提出利用多源域的多样化数据来学习鲁棒性更

好的分类器. 参考文献[4]的综述, 可将多源迁移学习方法分成两类. 第一类方法考虑将不同源域学得的基础分类器进行加权组合, 代表算法有: MCC-SVM^[5]和 A-SVM^[6]. 第二类方法进一步考虑了对无标目标数据信息的利用, 代表算法有: CP-MDA (Conditional Probability based Multi-source Domain Adaptation)^[7]、DAM^[8]及 DSM (Domain Selection Machine)^[2]. 与单源迁移学习方法相比, 上述的多源迁移学习方法都只考虑了域层面的迁

移不平衡性,而忽略了样例层面的迁移不平衡性.这样做忽视了对于域层面信息和样例层面信息的同时利用,导致迁移信息的利用不充分.

为解决两种层面迁移不平衡问题,本文提出一种平衡域与样例信息的多源迁移算法 MTL-BDI. 在样例层面,采用一种新的加权策略:针对目标域,首先从各源域分别训练出分类器(源分类器),并对目标数据进行预测.然后,定义类分布下的分类置信度,置信度越高,表明目标样例与源域越近,则使用相应的源分类器能更准确地预测目标样例类别.对于各个源域,将所有的源域样例作为 Universum 数据^[9]来引导目标分类面的确定.利用上面定义的分类置信度来衡量 Universum 数据距离目标分类面的接近程度,给予接近目标分类面的 Universum 数据较高的权值,以便于获取最佳的分类平面.在域层面,提出一种软加权指数,通过对每个源域按各自的迁移重要性来进行加权,实现与目标域的合理适配,该指数与模糊聚类算法 FCM(Fuzzy C-Means)^[10]的模糊指数类似,能有效控制目标函数凹凸性.综上,本文将这两种加权方法统称为双加权策略.进一步地,通过将双加权策略与基于 ε -不敏感损失的稀疏正则项相结合,可提高算法的预测效率,并避免了对大规模矩阵求逆.最后,通过在文本和图像分类数据集上的实验结果,一定程度上反应了本文工作的有效性.

2 多源迁移学习方法

2.1 问题定义

本文考察多源迁移学习问题,形式化多个源域和一个目标域的多源迁移模型,其中源域数据完全标注,目标域数据少量标注.给定 M 个源域,其中第 r 个源域为 $D_{S_r} = \left\{ (\mathbf{x}_i^{S_r}, y_i^{S_r}) \right\}_{i=1}^{N_{S_r}}$, $\mathbf{x}_i^{S_r}$ 和 $y_i^{S_r}$ 分别为该源域中第 i 个样例及标签.全体源域样例总数为 $N_S = \sum_{r=1}^M N_{S_r}$.类似地,目标域 D_G 由无标样例 $D_{G_u} = \left\{ \mathbf{x}_i^G \right\}_{i=1}^{N_{G_u}}$ 和标签样例 $D_{G_l} = \left\{ \mathbf{x}_i^G, y_i^G \right\}_{i=N_{G_u}+1}^{N_{G_u}+N_{G_l}}$ 构成,共有样例 $N_G = N_{G_u} + N_{G_l}$ 个.源域及目标域的样例总数为 $N = N_G + N_S$.记 M 个源分类器 $\{f^{S_r}(\mathbf{x})\}_{r=1}^M$ 分别由各源域训练得到,可为任意的有监督或迁移学习分类器.目标分类器为 $f^G(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b$,其中 \mathbf{w} 和 b 为分类器参数, $\phi(\mathbf{x})$ 是从原特征空间映射到再生核希尔伯特空间的函数.

2.2 学习模型

基于双加权策略、Universum 数据、 ε -不敏感损失及结构风险最小化,本文提出 MTL-BDI 算法.该算法的目标函数为:

$$\begin{aligned} \min J_m(\boldsymbol{\gamma}, f^G(\mathbf{x}), \mathbf{w}, b) \\ = \sum_{r=1}^M (\gamma_r)^m \sum_{i=1}^{N_{G_u}} \rho_i^{G_r} (f^G(\mathbf{x}_i^G) - f^{S_r}(\mathbf{x}_i^G))^2 \\ + \lambda_A \sum_{r=1}^M \sum_{i=1}^{N_{S_r}} \rho_i^{S_r} (f^G(\mathbf{x}_i^{S_r}))^2 + \lambda_B \sum_{i=N_{G_u}+1}^{N_G} (f^G(\mathbf{x}_i^G) - y_i^G)^2 \\ + \lambda_C \mathbf{w}^T \mathbf{w} + \lambda_D \left[\sum_{i=1}^{N_G} \ell_\varepsilon(\mathbf{w}^T \phi(\mathbf{x}_i^G) + b - f^G(\mathbf{x}_i^G)) \right. \\ \left. + \sum_{r=1}^M \sum_{i=1}^{N_{S_r}} \ell_\varepsilon(\mathbf{w}^T \phi(\mathbf{x}_i^{S_r}) + b - f^G(\mathbf{x}_i^{S_r})) \right] \quad (1) \end{aligned}$$

其中, $\boldsymbol{\gamma} = [\gamma_1, \dots, \gamma_M]^T$, $m(>1)$ 为加权指数, $\lambda_A, \lambda_B, \lambda_C$ 和 λ_D 为非负正则化参数.

下面详细给出式(1)每一项的具体含义.

式(1)中的第1项是关于目标域无标样例的双加权损失.其中, $(\gamma_r)^m$ 用于度量第 r 个源域关于目标域的迁移重要性,并满足下列约束:

$$\sum_{r=1}^M \gamma_r = 1, \quad \gamma_r > 0 \quad (2)$$

显然, $0 < \gamma_r < 1$ ($r = 1, \dots, M$). 加权指数 m 用于控制目标函数的凹凸性及各源域的迁移重要性程度.类似 FCM, 当 $m > 1$ 时, MTL-BDI 算法可采用交替优化策略来求解.

$f^G(\mathbf{x}_i^G)$ 和 $f^{S_r}(\mathbf{x}_i^G)$ 分别表示目标分类器和第 r 个源分类器对第 i 个目标样例的预测.相应地, $\rho_i^{G_r}$ 用于度量 $f^{S_r}(\mathbf{x}_i^G)$ 对 $f^G(\mathbf{x}_i^G)$ 逼近准确性的置信度,即对 $(f^G(\mathbf{x}_i^G) - f^{S_r}(\mathbf{x}_i^G))$ 进行加权. $\rho_i^{G_r}$ 的具体定义如下:

$$\rho_i^{G_r} = \tau_{G_r}^{(c)} e^{-\sigma_{G_r} z_{G_r,i}^{(c)}} / v_{G_r} \quad (3)$$

其中, σ_{G_r} 为尺度因子, v_{G_r} 为标准化因子.记 $D_{S_r}^{(c)} = \{\mathbf{x}_i^{S_r} : y_i^{S_r} = c \wedge (\mathbf{x}_i^{S_r}, y_i^{S_r}) \in D_{S_r}\}$ 表示第 r 个源域中类标为 c 的样例集合, $D_{G_r}^{(c)} = \{\mathbf{x}_i^G : f^{S_r}(\mathbf{x}_i^G) = c \wedge \mathbf{x}_i^G \in D_{G_u}\}$ 表示采用第 r 个源分类器预测类别为 c 的目标域无标样例集合.将 $D_{S_r}^{(c)}$ 看作一类, $D_{G_r}^{(c)}$ 看作另一类,使用两者进行训练,可得到分类超平面 $H(D_{S_r}^{(c)}, D_{G_r}^{(c)})$.定义 $z_{G_r,i}^{(c)}$ 为 $D_{G_r}^{(c)}$ 中目标样例 \mathbf{x}_i^G 到超平面 $H(D_{S_r}^{(c)}, D_{G_r}^{(c)})$ 的符号距离.当符号距离 $z_{G_r,i}^{(c)}$ 越小时,目标样例 \mathbf{x}_i^G 越接近第 r 个源域,第 r 个源分类器越能准确预测该目标样例,则分类置信度权重 $\rho_i^{G_r}$ 越大.为了缓解源分类器生成的目标样例类别 $f^{S_r}(\mathbf{x}_i^G)$ 不均衡问题,定义因子 $\tau_{G_r}^{(c)} = N_{G_u} / |D_{G_r}^{(c)}|$.

式(1)中的第2项是关于源域样例的 Universum 正则项,本文采用与目标样例分布不同但相关的各源域样例作为 Universum 数据. Universum 数据可用于增强分类面的区分度,它既与所考虑的分类问题相关但又不属于当中的任一类别.为了能最大程度区分各类数据,设定 Universum 数据的类别为 0.文献[11]指出不是所有的 Universum 数据都能正确引导分类面的确定,提出

Universe 数据中靠近分类面附近的样例对分类性能会有帮助. 对于本文的迁移学习场景, Universe 数据中越接近目标域的样例, 越靠近目标分类面, 越能正确引导分类面的确立. Universe 样例 $\mathbf{x}_i^{S_r}$ 接近目标域的程度, 可通过权重 $\rho_i^{S_r}$ 来衡量. 借鉴式(3) $\rho_i^{G_r}$ 的定义, 给出 $\rho_i^{S_r}$ 的定义:

$$\rho_i^{S_r} = \tau_{S_r}^{(c)} e^{-\sigma_{S_r} z_{S_r,i}^{(c)}} / v_{S_r} \quad (4)$$

其中, σ_{S_r} 为尺度因子, v_{S_r} 为标准化因子, $\tau_{S_r}^{(c)} = N_{S_r} / |D_{S_r}^{(c)}|$, $z_{S_r,i}^{(c)}$ 为 $D_{S_r}^{(c)}$ 中样例 $\mathbf{x}_i^{S_r}$ 到超平面 $H(D_{S_r}^{(c)}, D_{G_r}^{(c)})$ 的符号距离.

式(1)中的第3项是关于目标域标签样例的正则项. 使用目标分类器对目标域标签样例 D_{G_r} 的预测 $f^G(\mathbf{x}_i^G)$ 来逼近标签样例对应的真实类标 y_i^G .

式(1)中的第4项是 L2 正则项, 用于控制目标分类器 $f^G(\mathbf{x})$ 的复杂度, 并预防 $f^G(\mathbf{x})$ 过拟合.

式(1)中的第5项为 ε -不敏感损失正则项, 其中

$$\ell_\varepsilon(t) = \begin{cases} |t| - \varepsilon, & \text{if } |t| > \varepsilon \\ 0, & \text{otherwise} \end{cases} \quad \text{用于增强目标分类器的稀疏表示, 以及避免对大型逆矩阵进行数值求解. 其中, 第5项中的首项表示目标样例的 } \varepsilon\text{-不敏感损失, 末项表示 Universe 数据的 } \varepsilon\text{-不敏感损失.}$$

式(1)中的 γ 是对域信息迁移重要性程度的衡量, $\rho_i^{G_r}$ 是对目标样例迁移重要性程度的衡量, $\rho_i^{S_r}$ 是对 Universe 样例迁移重要性程度的衡量, 本文在 MTL-BDI 算法中将通过交替优化策略来对域与样例的迁移信息进行适配平衡.

2.3 优化算法

本节提出 MTL-BDI 算法来对式(1)求解. 令 Φ_G 为 $[\phi(\mathbf{x}_1^G), \dots, \phi(\mathbf{x}_{N_G}^G)]^T$, Φ_{S_r} 为 $[\phi(\mathbf{x}_1^{S_r}), \dots, \phi(\mathbf{x}_{N_{S_r}}^{S_r})]^T$, $\Phi = [\Phi_G^T, \Phi_{S_1}^T, \dots, \Phi_{S_M}^T]^T = [\Phi_1, \dots, \Phi_N]^T$. 定义 $K = \Phi\Phi^T$ 为 $N \times N$ 规模的核矩阵. 由于 ℓ_ε 是非平滑的, 对式(1)引入长度为 N 的拉格朗日乘子 α, α^*, η 和 η^* , 可得下面的优化问题:

$$\begin{aligned} \min J_m(\gamma, f^G, \mathbf{w}, \mathbf{b}, \xi, \xi^*, \alpha, \alpha^*, \eta, \eta^*) = & \\ \sum_{r=1}^M (\gamma_r)^m (f_u^G - f_{G_r}^G)^T P_{G_r} (f_u^G - f_{G_r}^G) & \\ + \lambda_A \sum_{r=1}^M (f_{S_r}^G)^T P_{S_r} (f_{S_r}^G) & \\ + \lambda_B \|f_l^G - y_l^G\|^2 + \lambda_C \|\mathbf{w}\|^2 + \lambda_D \mathbf{1}_N^T (\xi + \xi^*) & \\ - \alpha^T (\varepsilon + \xi + f^G - \Phi \mathbf{w} - \mathbf{b}) - \eta^T \xi & \\ - \alpha^{*T} (\varepsilon + \xi^* - f^G + \Phi \mathbf{w} + \mathbf{b}) - \eta^{*T} \xi^* & \\ \text{s. t. } \sum_{r=1}^M \gamma_r = 1, \quad \gamma_r > 0 & \end{aligned} \quad (5)$$

其中, $\mathbf{b} = b \times \mathbf{1}_N$, $\mathbf{1}_N$ 为全1列向量, ε 为偏差, ξ 和 ξ^* 为松弛变量, 且长度均为 N . 预测向量 $f_u^G, f_l^G, f_{S_1}^G, f_{S_2}^G, \dots, f_{S_M}^G$ 和 f^G

分别表示 $[f^G(\mathbf{x}_1^G), \dots, f^G(\mathbf{x}_{N_G}^G)]^T, [f^G(\mathbf{x}_{N_G+1}^G), \dots, f^G(\mathbf{x}_{N_G}^G)]^T, [f^G(\mathbf{x}_1^{S_r}), \dots, f^G(\mathbf{x}_{N_{S_r}}^{S_r})]^T, [f^{S_r}(\mathbf{x}_1^G), \dots, f^{S_r}(\mathbf{x}_{N_G}^G)]^T, [y_{N_G+1}^G, \dots, y_{N_G}^G]^T$ 和 $[(f_u^G)^T, (f_l^G)^T, (f_{S_1}^G)^T, \dots, (f_{S_M}^G)^T]^T$. 权值向量 ρ^G 和 ρ^{S_r} 分别为 $[\rho_1^G, \dots, \rho_{N_G}^G]^T$ 和 $[\rho_1^{S_r}, \dots, \rho_{N_{S_r}}^{S_r}]^T$. 并且, P_{G_r} 和 P_{S_r} 分别表示 $\text{diag}(\rho^{G_r})$ 和 $\text{diag}(\rho^{S_r})$.

定义 $\mathbf{0}_N$ 为长度 N 的全0列向量, $\mathbf{1}_{N_{G_l}}$ 为长度 N_{G_l} 的全1列向量, $\mathbf{0}_N$ 为长度 N 的全0列向量, \tilde{y}^G 为 $[(f_u^G)^T, (y_l^G)^T, \mathbf{0}_{N_s}^T]^T$, ρ^S 为 $[(\rho^{S_1})^T, (\rho^{S_2})^T, \dots, (\rho^{S_M})^T]^T$, $\rho^r = [(\rho^{G_r})^T, \lambda_B / \sum_{r=1}^M (\gamma_r)^m \mathbf{1}_{N_{G_l}}^T, \lambda_B / \sum_{r=1}^M (\gamma_r)^m (\rho^S)^T]^T$, q_i 为 $\sum_{r=1}^M [(\gamma_r)^m \rho_i^r \tilde{y}_i^G] / \sum_{i=1}^M [(\gamma_i)^m \rho_i^i]$, v_i 为 $1 / \sum_{r=1}^M [2(\gamma_r)^m \rho_i^r]$, \mathbf{q} 为 $[q_1, \dots, q_N]^T$, \mathbf{V} 为 $\text{diag}(v_1, \dots, v_N)$. 结合上面的符号定义, 下面建立式(1)优化问题的必要条件, 然后由此条件给出 MTL-BDI 算法:

定理1 对于任意的目标域或源域样例 $\{\mathbf{x}_i\}_{i=1}^N$, 当且仅当:

$$\gamma_r = 1 / \sum_{i=1}^M \left[\frac{(f_u^G - f_{G_r}^G)^T P_{G_r} (f_u^G - f_{G_r}^G)}{(f_u^G - f_{G_r}^G)^T P_{G_r} (f_u^G - f_{G_r}^G)} \right]^{1/(m-1)} \quad (6)$$

$$\text{和 } f^G(\mathbf{x}) = \sum_{i: \alpha_i^* - \alpha_i \neq 0} (\alpha_i^* - \alpha_i) K(\mathbf{x}_i, \mathbf{x}) / (2\lambda_C) + b \quad (7)$$

$$\mathbf{w} = \Phi^T (\alpha^* - \alpha) / (2\lambda_C) \quad (8)$$

$$b = \mathbf{1}_N^T (f^G + \varepsilon - \Phi \mathbf{w}) / N \quad (9)$$

其中, α 和 α^* 须满足

$$\begin{aligned} \min_{\alpha, \alpha^*} (\alpha - \alpha^*)^T (\mathbf{V}/2 + \mathbf{K}/(4\lambda_C)) (\alpha - \alpha^*) & \\ + \mathbf{q}^T (\alpha - \alpha^*) + \varepsilon^T (\alpha + \alpha^*) & \\ \text{s. t. } \begin{cases} (\alpha - \alpha^*)^T \mathbf{1}_N = 0 \\ \mathbf{0}_N \leq \alpha, \alpha^* \leq \lambda_D \mathbf{1}_N \end{cases} & \end{aligned} \quad (10)$$

时, $\gamma, f^G(\mathbf{x}), \mathbf{w}$ 和 b 为式(1)的局部最优解.

证明 为证明式(6)成立, 假设 $f^G(\mathbf{x}), \mathbf{w}$ 和 b 不变, 则问题变为在约束条件 $\sum_{r=1}^M \gamma_r = 1$ 下, $J_m(\gamma, f^G(\mathbf{x}), \mathbf{w}, b)$ 关于 γ 的最优化问题. 引入拉格朗日乘子 μ 后, 本问题等价于下面的无约束最小化问题:

$$\begin{aligned} \min J_m(\gamma, \mu) = \sum_{r=1}^M (\gamma_r)^m (f_u^G - f_{G_r}^G)^T P_{G_r} (f_u^G - f_{G_r}^G) & \\ + \mu \left(\sum_{r=1}^M \gamma_r - 1 \right) + \lambda_A \sum_{r=1}^M (f_{S_r}^G)^T P_{S_r} (f_{S_r}^G) & \\ + \lambda_B \|f_l^G - y_l^G\|^2 + \lambda_C \|\mathbf{w}\|^2 & \\ + \lambda_D \sum_{i=1}^N \ell_\varepsilon(\mathbf{w}^T \Phi_i + b - f_i^G) & \end{aligned} \quad (11)$$

对式(11)求关于 γ 和 μ 的一阶偏导并置0, 可得式(6).

为证明式(7)~(9)成立,假设 γ 不变,则式(1)优化问题等价于下面的无约束优化问题:

$$\begin{aligned} \min J_m(\mathbf{f}^G, \mathbf{w}, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\xi}^*, \boldsymbol{\alpha}, \boldsymbol{\alpha}^*, \boldsymbol{\eta}, \boldsymbol{\eta}^*) = & \\ \sum_{r=1}^M (\gamma_r)^m (\mathbf{f}^G - \tilde{\mathbf{y}}^{G_r})^T \mathbf{P}_r (\mathbf{f}^G - \tilde{\mathbf{y}}^{G_r}) & \\ + \lambda_c \|\mathbf{w}\|^2 + \lambda_D \mathbf{1}_N^T (\boldsymbol{\xi} + \boldsymbol{\xi}^*) & \\ - \boldsymbol{\alpha}^T (\boldsymbol{\varepsilon} + \boldsymbol{\xi} + \mathbf{f}^G - \Phi \mathbf{w} - \mathbf{b}) - \boldsymbol{\eta}^T \boldsymbol{\xi} & \\ - \boldsymbol{\alpha}^{*T} (\boldsymbol{\varepsilon} + \boldsymbol{\xi}^* - \mathbf{f}^G + \Phi \mathbf{w} + \mathbf{b}) - \boldsymbol{\eta}^{*T} \boldsymbol{\xi}^* & \end{aligned} \quad (12)$$

其中, \mathbf{P}_r 为 $\text{diag}(\rho^r)$. 对式(12)求关于 \mathbf{f}^G 、 \mathbf{w} 、 \mathbf{b} 、 $\boldsymbol{\xi}$ 和 $\boldsymbol{\xi}^*$ 的一阶偏导并置 0, 可得式(7)~(10). 综上所述, 定理 1 证毕.

如果将式(10)中的 $(V/2 + K/(4\lambda_c))$ 部分看作核矩阵, 则与经典的支持向量回归机(SVR)对偶问题的形式一致, 因此可采用经典的 SVR 工具包(如 LIBSVM^[12])来解决式(10)的优化问题. 值得注意的是, 如果不使用 ε -不敏感损失正则项, 在假设 γ 不变的条件下, 式(1)问题等价于:

$$\begin{aligned} \min J_m(\mathbf{w}, \mathbf{b}) = & \sum_{r=1}^M (\gamma_r)^m (\Phi \mathbf{w} + \mathbf{b} - \tilde{\mathbf{y}}^{G_r})^T \\ & \mathbf{P}_r (\Phi \mathbf{w} + \mathbf{b} - \tilde{\mathbf{y}}^{G_r}) + \lambda_c \|\mathbf{w}\|^2 \end{aligned} \quad (13)$$

对上式求关于 \mathbf{w} 和 \mathbf{b} 的一阶偏导并置 0, 可得式(13)优化问题对应的解:

$$\begin{cases} \mathbf{w} = \left(\sum_{r=1}^M (\gamma_r)^m \Phi^T \mathbf{P}_r \Phi + \lambda_c \mathbf{I} \right)^{-1} \\ \quad \cdot \left[\sum_{r=1}^M (\gamma_r)^m \Phi^T \mathbf{P}_r (\tilde{\mathbf{y}}^{G_r} - \mathbf{b}) \right] \\ \mathbf{b} = \left(\sum_{r=1}^M (\gamma_r)^m \mathbf{P}_r \right)^{-1} \left[\sum_{r=1}^M (\gamma_r)^m \mathbf{P}_r \tilde{\mathbf{y}}^{G_r} \right] - \Phi \mathbf{w} \end{cases} \quad (14)$$

显然式(14)的解是非稀疏的, 并且参数 \mathbf{w} 、 \mathbf{b} 的计算涉及到矩阵求逆. 当矩阵规模较大时, 使用式(14)来求解是不可取的, 利用 LU 分解计算式(14)的逆矩阵问题需要时间复杂度 $O(N^3)$. 通过定理 1 可知, 利用 LIBSVM 计算式(10)的优化问题需要时间复杂度 $O(N^{2.3})$. 因此, 采用 ε -不敏感损失的目标函数可以降低计算量.

MTL-BDI 算法是建立在必要条件(6)、(7)、(8)和(9)基础上的. 具体流程见算法 1.

算法 1 MTL-BDI

输入: 给定 M 个源域 $D_{S_r} = \{(\mathbf{x}_i^{S_r}, \mathbf{y}_i^{S_r})\}_{i=1}^{N_{S_r}}, r=1 \cdots M$ 及目标域中无标样例 $D_{G_u} = \{\mathbf{x}_i^G\}_{i=1}^{N_{G_u}}$, 标签样例 $D_{G_l} = \{\mathbf{x}_i^G, \mathbf{y}_i^G\}_{i=N_{G_u}+1}^{N_{G_l}}$.

输出: 输出目标分类器 $f^G(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + \mathbf{b}$.

Step 1 对每个源域 D_{S_r} , 用 D_{S_r} 去训练源分类器 $f^{S_r}(\mathbf{x})$, 得到无标样例 D_{G_u} 关于 $f^{S_r}(\mathbf{x})$ 的预测 $\mathbf{f}_{G_u}^{S_r}$.

Step 2 对 D_{G_u} 和 $\mathbf{f}_{G_u}^{S_r}$, 由式(3)计算 ρ^{S_r} .

Step 3 对每个源域 D_{S_r} , 由式(4)计算 ρ^{S_r} .

Step 4 选择核函数 $K(\mathbf{x}_i, \mathbf{x}_j)$ 并计算核矩阵 \mathbf{K} 为 $\{K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)\}_{i=1, j=1}^{N \times N}$.

Step 5 设定 $m \in (1, \infty)$, 初始化 $\gamma^{(0)}$ 使其满足 $\sum_{r=1}^M \gamma_r(0) = 1$ 约束.

Step 6 当迭代次数为 $l(l=0, 1, 2, \cdots)$ 时, 通过式(10)和 SVR 工具包(如 LIBSVM)求解参数 $\mathbf{f}^{G(l)}$ 、 $\mathbf{w}^{(l)}$ 和 $\mathbf{b}^{(l)}$.

Step 7 通过式(6)将 $\gamma^{(l)} = \{\gamma_r^{(l)}\}_{r=1}^M$ 更新为 $\gamma^{(l+1)} = \{\gamma_r^{(l+1)}\}_{r=1}^M$.

Step 8 若 $\|\gamma^{(l+1)} - \gamma^{(l)}\| < \zeta$ (ζ 为一极小常数, 本文取 10^{-6}), 则停止算法; 否则, 令 $l = l + 1$, 返回 Step 6.

2.4 算法收敛性分析

本节证明: 假设 \mathbf{b} 不变的情况下, 通过交替计算, 目标函数 $J_m(\gamma, \mathbf{f}^G, \mathbf{w})$ 将减小或保持不变, 即 $J_m(\gamma^{(l+1)}, \mathbf{f}^{G(l+1)}, \mathbf{w}^{(l+1)}) \leq J_m(\gamma^{(l)}, \mathbf{f}^{G(l)}, \mathbf{w}^{(l)})$. 为了证明该性质, 下面先引入引理 1、引理 2 和引理 3.

引理 1 令 $\phi(\gamma) = J_m(\gamma, \mathbf{f}^G, \mathbf{w})$, 其中 \mathbf{f}^G 和 \mathbf{w} 不变, 则 γ 为 $\phi(\gamma)$ 局部最优解的充要条件是, γ 由式(6)计算得到.

证明 必要性已在定理 1 中证完. 下面证明充分性. 先根据式(6)计算 γ , 然后计算 $\phi(\gamma)$ 的 $M \times M$ 阶海森矩阵 $\mathbf{U}_\gamma = [u_{t,r}(\gamma)]$. 由式(1), 有:

$$\begin{aligned} u_{t,r}(\gamma) &= \partial^2 [\phi(\gamma) / \partial \gamma_r] / \partial \gamma_t \\ &= \begin{cases} m(m-1)(\gamma_t)^{m-2} (\mathbf{f}_u^G - \mathbf{f}_{G_r}^G)^T \mathbf{P}_{G_r} (\mathbf{f}_u^G - \mathbf{f}_{G_r}^G), & t=r \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (15)$$

式(15)中, 海森矩阵 \mathbf{U}_γ 是正定的, 从而说明式(6)是 $\phi(\gamma)$ 局部最优解的充分条件.

同理可得引理 2 和引理 3.

引理 2 令 $h(\mathbf{f}^G) = J_m(\gamma, \mathbf{f}^G, \mathbf{w})$, 其中 γ 和 \mathbf{w} 不变, 则 \mathbf{f}^G 为 $h(\mathbf{f}^G)$ 局部最优解的充要条件是, \mathbf{f}^G 是由式(7)计算得到的.

引理 3 令 $\Omega(\mathbf{w}) = J_m(\gamma, \mathbf{f}^G, \mathbf{w})$, 其中 γ 和 \mathbf{f}^G 不变, 则 \mathbf{w} 为 $\Omega(\mathbf{w})$ 局部最优解的充要条件是, \mathbf{w} 是由式(8)计算得到的.

定理 2 令 $\gamma^{(l)} \mathbf{f}^{G(l)}$ 和 $\mathbf{w}^{(l)}$ 是通过算法 Step 6 和 Step 7 计算得到的序列, 则对于所有 $l=0, 1, 2, \cdots$, 有:

$$J_m(\gamma^{(l+1)}, \mathbf{f}^{G(l+1)}, \mathbf{w}^{(l+1)}) \leq J_m(\gamma^{(l)}, \mathbf{f}^{G(l)}, \mathbf{w}^{(l)}) \quad (16)$$

证明 由引理 1、引理 2 和引理 3 可得:

$$J_m(\gamma^{(l+1)}, f^{G(l+1)}, w^{(l+1)}) \leq J_m(\gamma^{(l+1)}, f^{G(l+1)}, w^{(l)}) \quad (17)$$

$$J_m(\gamma^{(l+1)}, f^{G(l+1)}, w^{(l)}) \leq J_m(\gamma^{(l+1)}, f^{G(l)}, w^{(l)}) \quad (18)$$

$$J_m(\gamma^{(l+1)}, f^{G(l)}, w^{(l)}) \leq J_m(\gamma^{(l)}, f^{G(l)}, w^{(l)}) \quad (19)$$

合并式(17)、(18)和(19),即可得式(16).

因此,假设 b 不变的情况下,目标函数 J_m 在迭代过程中能保持单调递减,并且有下界 0.

3 实验结果与分析

3.1 实验设置

为了评估所提出的 MTL-BDI 算法的性能,本文在文本集 Twenty Newsgroups^[8]和 Email Spam^[8]和图像集 PIE^[13]、COIL20^[14]、Office + Caltech^[15],共 5 个多源迁移学习数据集上进行实验.限于文章篇幅,本文从全部数据集的 17 个多源迁移任务中选取 9 个任务给出实验结果,并进行分析.

对于 Twenty Newsgroups 和 Email Spam 数据集,本文采用与文献[8]相同的构造方法. Twenty Newsgroups 数据集中 2 个任务组具体为 rec vs sci(NG-T1)、comp vs rec(NG-T2).在 Email Spam 数据集中,将个人收件箱的 3 个数据集作为源域,将公共收件箱的数据集作为目标域.对于文本集,每次实验随机选取 20 个目标样例作为目标域标签样例,即每类选取 10 个标签样例.

对于 Office + Caltech 数据集,实验中选取 3 个子集作为源域,剩余的子集作为目标域,构造 2 个多源迁移任务 $CDW \rightarrow A(OC-T1)$ 、 $ADW \rightarrow C(OC-T2)$.每次实验随机选取 80 个目标样例作为目标域标签样例,即每类选取 8 个标签样例.实验采用与文献[14]相同方式来对 COIL20 数据集的进行预处理.将该数据集划分为四个不相交的子集: $C1$ 、 $C2$ 、 $C3$ 、 $C4$.选取 3 个子集作为源域,剩余的子集作为目标域,构造 2 个多源迁移任务 $C2C3C4 \rightarrow C1(COIL-T1)$ 、 $C1C3C4 \rightarrow C2(COIL-T2)$.每次实验随机选取 60 个目标样例作为目标域标签样例,即每类选取 3 个标签样例.实验采用与文献[13]相同方式来对 PIE 数据集的进行预处理,选取 PIE 数据集中 5 个不同的人脸朝向作为子集,即 $P1(C05)$ 、 $P2(C07)$ 、 $P3(C09)$ 、 $P4(C27)$ 、 $P5(C29)$.使用 4 个子集作为源域,剩余的子集作为目标域,构造 2 个多源迁移任务 $P2P3P4P5 \rightarrow P1(PIE-T1)$ 、 $P1P3P4P5 \rightarrow P2(PIE-T2)$.每次实验随机选取 204 个目标样例作为目标域标签样例,即每类选取 3 个标签样例.

在本文实验中,将 MTL-BDI 算法与监督学习算法支持向量机(SVM)^[12],单源迁移学习算法 FR(Feature Replication)^[16],以及 5 个多源迁移学习算法 MCC-SVM^[5]、A-SVM^[6]、Multi-KMM^[5]、DAM^[8]进行对比.参考文献[5,8,13,14,17],对实验中各算法的参数进行网格寻优. SVM

采用 LIBSVM 工具包实现,正则项参数通过遍历 $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于 FR,正则项参数通过遍历 $\delta \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于 MCC-SVM,其正则项参数通过遍历 $\lambda \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于 Multi-KMM,其参数 α 通过遍历 $\{0.01, 0.1, 1, 10, 100\}$ 来设置,参数 λ 通过遍历 $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于 DAM,其参数 λ_c 通过遍历 $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置,参数 λ_L 和 λ_D 通过遍历 $\{0.01, 0.1, 1, 10, 100\}$ 来设置, MMD 中的参数 β 根据实验数据集的不同以满足最佳结果来设定.对于 A-SVM,其参数通过遍历 $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于本文算法 MTL-BDI 在实验中的参数设置如下,固定 $m=2$ 、 $\lambda_A=1$ 、 $\lambda_B=1$ 、 $\lambda_C=1$ 和 $\varepsilon=0.1$,参数 λ_D 通过遍历 $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$ 来设置.对于上述核方法,线性核被用于文本数据集,高斯核被用于图像数据集.根据文献[13],高斯核的带宽参数设定为所有训练数据间欧氏距离平均值的倒数.实验采用测试数据(即目标域无标数据)准确率作为评价指标.

3.2 实验结果

本节给出学习算法 MTL-BDI 以及 6 种基准算法在关于文本、图像的 9 个多源迁移学习任务上的准确率.在实验中,对于每个算法在各任务上重复运行 10 次,并在表 1 公布它们的均值及标准差.其中,MTL-BDI 在 Twenty Newsgroups 数据集中使用 SVM 作为源分类器,在 Email Spam、Office + Caltech 和 PIE 数据集中使用 MCC-SVM 作为源分类器,在 COIL 数据集中使用 A-SVM 作为源分类器.

首先,根据表 1 的结果,可以发现所有的迁移学习方法性能都优于非迁移的经典 SVM 分类器,这说明了知识迁移对于文本和图像数据的跨领域识别的有效性.此外,由于传统的监督学习算法,如 SVM,其模型的构造假设是不同域的数据具有相同的分布,这使得此类算法在用于迁移学习环境时效果并不理想.而本文提出的 MTL-BDI 算法考虑到了利用源域数据作为 Universum 数据并兼顾平衡了域层面和样例层面的迁移信息.因此,可以看到 MTL-BDI 算法在 9 个迁移任务中均显著优于 SVM 算法.

其次,从所有实验的均值来看,多源迁移学习算法中的三个(即 Multi-KMM, DAM 和 MTL-BDI)优于单源迁移学习算法 FR.这是因为多源迁移学习方法考虑到了对每个源域进行有效适配,进而避免了对各源域进行蛮力的迁移.

最后,与多源迁移学习方法相比,MTL-BDI 算法在绝大多数数据集上优于其他的多源学习方法.这也从侧面验证了软加权指数 m 可以获取更为理想的权重,

进而更好地评估出各源域对目标域的迁移重要性.此外,可以看到如果不考虑软加权指数 m 和样例层面的加权 ρ' 时,式(1)中的第一项与 DAM 算法中所提出的域依赖正则项是相类似的.而表 4 的实验结果表明,MTL-BDI 算法在 9 个多源迁移任务中有 8 个优于 DAM 算法,这说明双加权策略使得 MTL-BDI 算法在处理多源迁移学习任务时更为有效.而与其它多源迁移学习方法相比,MTL-BDI 算法在绝大多数任务上也取得的最佳的实验结果,这进一步说明了多源迁移学习中对域与样例信息进行平衡的重要性.

表 1 实验数据集的平均分类准确率及标准差 (%)

任务	SVM	FR	MCC-SVM	Multi-KMM	DAM	A-SVM	MTL-BDI
NG-T1	91.95 (0.06)	91.19 (1.54)	89.16 (0.84)	93.42 (0.25)	89.74 (0.23)	92.72 (0.34)	94.75 (0.26)
NG-T2	87.79 (0.10)	93.79 (0.99)	92.03 (0.91)	95.58 (0.12)	95.39 (0.11)	90.00 (0.59)	98.07 (0.20)
Email Spam	61.02 (0.04)	75.26 (1.34)	69.29 (1.61)	69.31 (1.28)	75.11 (1.98)	72.65 (1.13)	79.21 (2.31)
OC-T1	49.63 (1.35)	55.87 (1.26)	50.73 (1.19)	53.79 (0.95)	56.34 (1.21)	55.55 (1.62)	58.56 (1.42)
OC-T2	41.37 (0.97)	43.39 (1.18)	42.87 (1.22)	47.03 (1.55)	46.79 (1.37)	45.88 (1.48)	50.26 (1.72)
COIL-T1	78.82 (2.48)	81.65 (2.06)	84.35 (2.16)	96.20 (2.02)	89.83 (1.77)	88.82 (1.48)	96.35 (2.14)
COIL-T2	79.62 (1.95)	82.28 (2.55)	84.50 (1.51)	93.32 (1.81)	87.73 (2.18)	90.09 (1.38)	94.35 (1.29)
PIE-T1	51.30 (2.67)	83.62 (2.30)	57.54 (0.53)	83.20 (0.93)	77.16 (1.29)	73.37 (1.33)	71.67 (1.32)
PIE-T2	63.15 (2.49)	82.76 (2.70)	73.81 (1.18)	84.31 (1.80)	78.42 (2.19)	75.44 (1.71)	86.72 (1.38)
平均值	67.18	76.65	71.59	79.57	77.39	76.06	81.10

3.3 算法运行性能分析

若设定算法 1 的源分类器和分类超平面 $H(D_{S_r}^{(c)}, D_{G_c}^{(c)})$ 均为支持向量机,则算法 1 中 Step 1 的时间复杂度为 $O(\sum_{r=1}^M N_{S_r}^{2.3})$, Step 2 和 Step 3 的时间复杂度为 $O(\sum_{r=1}^M \sum_{c=1}^C (|D_{S_r}^{(c)}| + |D_{G_c}^{(c)}|)^{2.3})$, Step 4 的时间复杂度为 $O(N^2)$, Step 5 的时间复杂度为 $O(M)$. 设定 MTL-BDI 算法最终迭代次数为 t 、软加权指数 m 取 2,则 Step 6 至 Step8 的时间复杂度为 $O(t(N^{2.3} + M^2N + M))$. 由于与样例总数 N 相比,源域的个数 M 较小.因此, $O(tN^{2.3})$ 主要决定了算法的总体时间复杂度.也就是说,两个主要因素即最大迭代次数 t 和运算数据的规模 N ,决定了 MTL-BDI 方法的时间复杂度.

时间复杂性实验验证了上面的分析.本文分别从

Twenty Newsgroups、Email Spam、Office + Caltech、COIL 和 PIE 上随机选择一个任务运行所有算法,来分析时间复杂度,实验结果如表 2 所示.可以看出,MTL-BDI 算法在规模较大的目标域数据集上表现出相当长的运行时间,例如 NG(T1)、Email Spam 和 PIE(T1).与绝大多数基准算法相比,本文的算法 MTL-BDI 时间复杂度的结果并不显得突出,这与本文算法需要同时兼顾域与样例迁移信息的适配、花费时间对全部 Universum 数据进行计算以及所采用的交替优化策略需要多次迭代才收敛有关.

表 2 实验数据的平均运行时间及标准差(单位:s)

任务	SVM	FR	MCC-SVM	Multi-KMM	DAM	A-SVM	MTL-BDI
NG (T1)	2.57 (0.23)	4.78 (0.04)	2.86 (0.13)	9.23 (0.04)	6.97 (0.15)	4.37 (0.10)	188.46 (14.75)
Email Spam	5.49 (0.07)	19.76 (0.30)	6.50 (0.23)	25.71 (1.32)	19.16 (0.74)	12.68 (0.80)	451.46 (16.91)
OC (T1)	0.51 (0.02)	0.76 (0.02)	0.41 (0.03)	0.92 (0.02)	7.84 (0.04)	0.69 (0.03)	15.73 (1.06)
COIL (T1)	0.43 (0.10)	0.64 (0.06)	0.33 (0.04)	0.53 (0.01)	0.67 (0.03)	0.60 (0.05)	6.78 (0.77)
PIE (T1)	12.30 (3.63)	22.99 (0.97)	14.03 (0.10)	181.72 (23.17)	34.90 (1.65)	28.43 (1.27)	201.52 (10.13)

3.4 收敛性分析

为了验证算法的实际收敛性,本文分别从 Twenty Newsgroups 数据集、Office + Caltech 数据集和 PIE 数据集中选取一个分类任务,实验后得到目标函数的收敛趋势图,如图 1 所示.可以看到随着迭代次数的增加,目标函数值在迭代 10 次后呈现平稳趋势不再明显变化,并收敛到局部最优解.

3.5 参数敏感性分析

对于 MTL-BDI 算法,本文分别从 Twenty Newsgroups 数据集、Office + Caltech 数据集和 PIE 数据集选取一个分类任务,观察参数 λ_A 、 λ_C 、 λ_D 和 m 的敏感性,如图 2 ~ 5 所示.可以看出, $\lambda_A \in [1, 10]$, $\lambda_C \in [0.1, 5]$, $\lambda_D \in [1, 100]$ 及 $m \in [1.5, 2.5]$ 较为合理.

3.6 源域选择对算法性能的影响

这里实验性地考察域层面上的加权对算法性能的影响.具体地,本文采用文本数据集 Twenty Newsgroups 和图像数据集 Office + Caltech 的全部任务来进行实验.在实验中,先运行一遍 MTL-BDI 算法,公布各源域的权值,然后剔除掉权值最小的源域.对剩下的源域再执行一遍 MTL-BDI 算法,将这次的精度与之前的精度进行了对比,如表 3 所示.可以看到,进行实验的 7 个任务中,有 5 个任务在通过源域剔除后,精度不同程度的得到了提升,一定程度说明了源域选择对算法精度是重要的.因此,如何在本文算法的基础上进一步设计出具有源域选择机制的加权策略,将是本文下一阶段的研究方向.

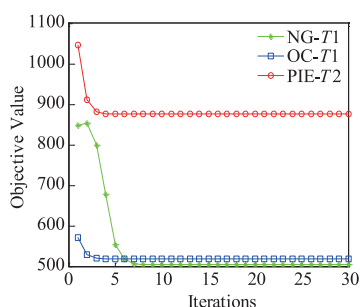


图1 目标函数收敛性

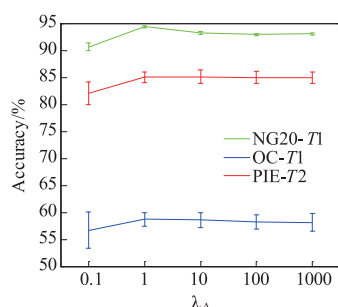
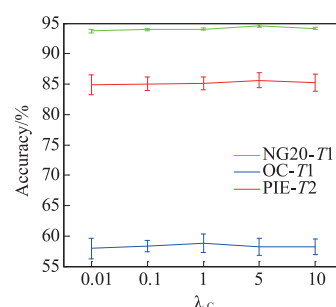
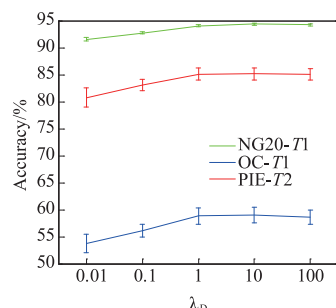
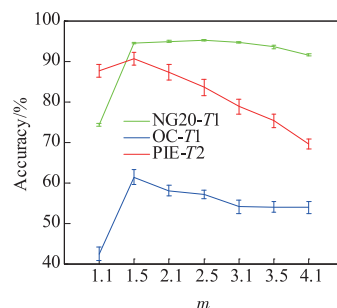
图2 参数 λ_A 的敏感性分析图3 参数 λ_C 的敏感性分析图4 参数 λ_D 的敏感性分析图5 参数 m 的敏感性分析

表3 源域选择对算法性能的影响

任务	NG-T1	NG-T2	NG-T3	OC-T1	OC-T2	OC-T3	OC-T4
源域1 权重	0.1249	0.0784	0.1666	0.4044	0.3594	0.0474	0.2478
源域2 权重	0.1312	0.1122	0.1513	0.2662	0.2897	0.0512	0.2377
源域3 权重	0.7439	0.8093	0.6821	0.3294	0.3509	0.9013	0.5145
原始精度	94.68	97.76	94.54	58.43	50.91	79.22	75.81
源域剔除后精度	94.47	97.55	95.84	60.93	54.55	81.82	76.74

4 结论

本文就多源迁移学习中存在的域层面迁移不平衡性和样例层面迁移不平衡性这两个方面进行了讨论. 首先, 在样例层面提出一种新的加权策略, 分别对目标数据和源域数据进行加权迁移适配. 其次, 在域层面提出一种软加权指数, 通过交替优化策略对每个源域按各自的迁移重要性进行加权, 实现与目标域的适配. 进一步, 这两种加权方法与基于 ε -不敏感损失的稀疏正则项相结合, 得出一种新的多源迁移算法 MTL-BDI. 该算法的优势是考察了多源迁移学习中多种必要的学习准则, 并可采用包括监督模型和迁移模型在内的任何类型的分类器作为源分类器. 在 9 个文本、图像多源迁移任务的系统实验证明了本文方法相对于其他方法的优势.

参考文献

- [1] Luo P, Zhuang F Z, Xiong H, et al. Transfer learning from multiple source domains via consensus regularization[A]. Proceedings of the ACM International Conference on Information and Knowledge Management [C]. USA: ACM, 2008. 103 – 112.
- [2] Duan L, Xu D, Chang S F. Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach[A]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition [C]. USA: IEEE, 2012. 1338 – 1345.
- [3] Yao Y, Doretto G. Boosting for transfer learning with multiple sources[A]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition [C]. USA: IEEE, 2010. 1855 – 1862.
- [4] Sun S L, Shi H L, Wu Y B. A survey of multi-source domain adaptation[J]. Information Fusion, 2015, 24 (C): 84 – 92.
- [5] Schweikert G, Widmer C, Schölkopf B, et al. An empirical analysis of domain adaptation algorithms for genomic sequence analysis[A]. Proceedings of the Neural Information Processing Systems Conference [C]. USA: Curran Associates, 2009. 1433 – 1440.
- [6] Yang J, Yan R, Hauptmann A G. Cross-domain video concept detection using adaptive SVMs[A]. Proceedings of the ACM International Conference on Multimedia [C].

- USA:ACM,2007. 188 – 197.
- [7] Chattopadhyay R, Sun Q, Fan W, et al. Multi-source domain adaptation and its application to early detection of fatigue [A]. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining [C]. USA:ACM, 2011. 717 – 725.
- [8] Duan L, Xu D, Tsang I W. Domain adaptation from multiple sources: A domain-dependent regularization approach [J]. IEEE Transaction on Neural Networks and Learning System, 2012, 23(3): 504 – 518.
- [9] Weston J, Collobert R, Sinz F, et al. Inference with the universum [A]. Proceedings of the International Conference on Machine Learning [C]. USA:ACM, 2006. 1009 – 1016.
- [10] Bezdek J C. Pattern Recognition with Fuzzy Objective Function Algorithms [M]. New York:Plenum, 1981. 43 – 93.
- [11] Chen S, Zhang C S. Selecting informative universum sample for semi-supervised learning [A]. Proceedings of the International Joint Conference on Artificial Intelligence [C]. USA:Morgan Kaufmann, 2009. 1016 – 1021.
- [12] Chang C C, Lin C J. LIBSVM: A library for support vector machines [J]. ACM Transactions on Intelligent Systems and Technology, 2011, 2(3): 1 – 27.
- [13] Long M, Wang J, Ding G, et al. Adaptation regularization: a general framework for transfer learning [J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(5): 1076 – 1089.
- [14] Long M, Wang J, Ding G, et al. Transfer feature learning with joint distribution adaptation [A]. Proceedings of the IEEE International Conference on Computer Vision [C]. USA:IEEE, 2013. 2200 – 2207.
- [15] Gong B, Shi Y, Sha F, et al. Geodesic flow kernel for unsupervised domain adaptation [A]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition [C]. USA:IEEE, 2012. 2066 – 2073.
- [16] Daumé H III. Frustratingly easy domain adaptation [A]. Proceedings of the Annual Meeting of the Association for Computational Linguistics [C]. USA: ACL, 2007. 256 – 263.
- [17] Li S, Song S J, Huang G. Prediction reweighting for domain adaptation [J]. IEEE Transactions on Neural Networks and Learning Systems, 2017, 28(7): 1682 – 1695.
- [18] 张景祥, 王士同. 基于共同决策方向矢量的多源迁移及其快速学习方法 [J]. 电子学报, 2015, 43(7): 1349 – 1355.
- Zhang Jing-xiang, Wang Shi-tong. Common-decision-vector based multiple source transfer learning classification and its fast learning method [J]. Acta Electronica Sinica, 2015, 43(7): 1349 – 1355. (in Chinese)
- [19] 蒋亦樟, 邓赵红, 王士同. 0 阶 L2 型 TSK 迁移学习模糊系统 [J]. 电子学报, 2013, 41(5): 897 – 904.
- Jiang Yi-zhang, Deng Zhao-hong, Wang Shi-tong. 0-order-l2-norm-takagi-sugeno-kang type transfer learning fuzzy system [J]. Acta Electronica Sinica, 2013, 41(5): 897 – 904. (in Chinese)
- [20] 倪彤光, 王士同, 应文豪, 邓赵红. 迁移组概率学习机 [J]. 电子学报, 2013, 41(11): 2207 – 2215.
- Ni Tong-guang, Wang Shi-tong, Ying Wen-hao, Deng Zhao-hong. Transfer group probabilities based learning machine [J]. Acta Electronica Sinica, 2013, 41(11): 2207 – 2215. (in Chinese)

作者简介



季鼎承 男, 1992 年生于江苏淮安. 现为江南大学大学数字媒体学院硕士研究生. 主要研究方向为人工智能.
E-mail: know4thysself@ gmail. com



蒋亦樟 (通讯作者) 男, 1988 年生于江苏无锡. 现为江南大学大学数字媒体学院副教授、硕士生导师. 主要研究方向为人工智能、模式识别、模糊系统.
E-mail: jyz0512@ 163. com



王士同 男, 1964 年生于江苏扬州. 现为江南大学大学数字媒体学院教授、博士生导师. 主要研究方向为人工智能、模式识别和生物信息学.
E-mail: wxwangst@ aliyun. com