This is the Pre-Published Version.
The following publication W. Hang, S. Liang, K. Choi, F. Chung and S. Wang, "Selective Transfer Classification Learning With Classification-Error-Based Consensus Regularization," in IEEE Transactions on Emerging Topics in Computational Intelligence is available at https://doi.org/10.1109/TETCI.2019.2892762

# Selective Transfer Classification Learning With Classification-Error-Based Consensus Regularization

***Abstract*:** Transfer learning methods are conventionally conducted by utilizing abundant labeled data in the source domain to build an accurate classifier for the target domain with scarce labeled data. However, most current transfer learning methods assume that all the source data are relevant to target domain, which may induce negative learning effect when the assumption becomes invalid as in many practical scenarios. To tackle this issue, the key is to identify the correlated source data and the corresponding weights. In this paper, we make use of the least square-support vector machine (LS-SVM) framework for identifying the correlated data and their weights from source domain. By keeping the consistency between the distributions of the classification errors of both the source and target domains, we first propose the classification-error-based consensus regularization (CCR), which can guarantee the performance improvement of the target classifier. Based on this approach, a novel selective transfer learning method (CSTL) is then developed to autonomously and quickly choose the correlated source data and the weights to exploit the transferred knowledge by solving the LS-SVM based objective function. This method minimizes the leave-one-out cross-validation error despite scarce target training data. The advantages of the CSTL are demonstrated by evaluating its performance on public text and image datasets and comparing it with that of the state-of-the-art transfer learning methods.

***Keywords***: Selective transfer learning, Classification error, Least square-support vector machine (LS-SVM), Leave-one-out cross-validation

## 1. Introduction

A vast amount of data is always desirable for mining and extracting useful information. The data can be acquired readily in common and conventional settings. However, for newly emerging situations, the availability of data can be very limited. For example, in image recognition applications, a wide variety of pictures of ordinary cars can be obtained from the internet for locating a car in the traffic; while the pictures of illegally modified cars, with markedly different exterior, are scarce that makes accurate recognition a difficult task. Another example is the recognition of web document. While billions of webpages are available from the internet for categorization, for documents in newly create websites, the data features or distributions can be very different which make it difficult to categorize them given the amount of such documents is few.

To address this issue, researches have been conducted to leverage the abundance of

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

existing data in a domain to deal with the problem in another domain that is of certain degree of similarity but the data are scarce. This approach is known as transfer learning [1], and the two domains are referred to as the source and target domains respectively. Most of the transfer learning methods attempt to improve the classification performance of target domain by exploiting the shared knowledge structure between the source and target domains from three main aspects: 1) *what to transfer*: depending on the problem, the transferred knowledge from the source domain can be categorized into instance, feature representation and model parameters; 2) *how to transfer*: the transferred knowledge, encoded by methods like boosting [2], is propagated from the source to the target domain, and extracted as the supervision information for the target domain; and 3) *when to transfer*: Since the data collected in the past are not always correlated to the target data, it is critical to determine when to transfer for brute force transfer may induce negative effect [1], e.g., degrade the classification performance.

Transfer learning has been proved to be promising in real-world applications including text categorization [3-4], sentiment analysis [5], image classification [6], video summarization [7] and collaborative filtering [8]. Despite the success, most current transfer learning methods [10-20] assume that all the source data are wholly relevant to the target domain so that the data of the entire source domain can be leveraged to exploit the shared knowledge structure. This assumption is not valid in many real-world applications.

If the source data that are related to the target domain, namely, *correlated data*, and the weights of these data in the source domain can be identified to exploit the common knowledge structure shared between the two domains,, the resulting transferred knowledge will be more favorable for facilitating the construction of the target classifier. This is analogous to the theory of adaptive control of thought (ACT) in cognitive psychology [21]. The ACT framework considers that the process of human cognition develops progressively. When encountering a new situation, humans retrieve previous information of related situations for making interference in order to learn about the new information. The idea of ACT is illustrated in Fig. 1, where the red and green apples are retrieved from the source domain according to the characteristics (shape and color) of a few target objects; similarly, for the recognition of a newly seen chicken. This ability enables humans to make reference to related information in the past and perform inductive inference on a new situation even with only a small amount of information.

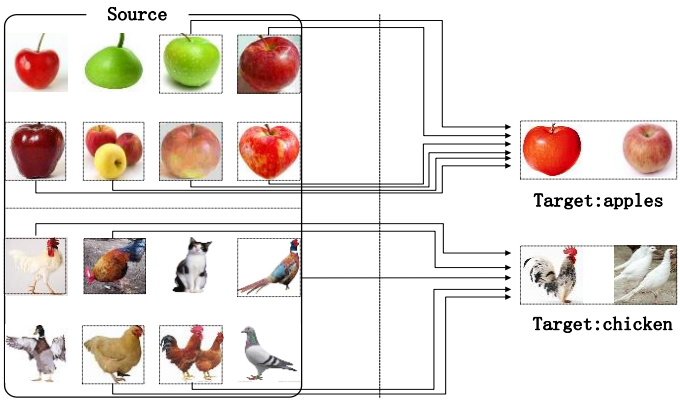



Fig. 1 Illustration of the theory of ACT: learning new objects of *apple* and *chicken.*

Based on this idea, we propose the classification-error-based consensus regularization (CCR) as the bridge for propagating the useful information from the source domain to the target domain by requiring a consistency in the distribution of the classification errors between the two domains. The classification selective transfer learning (CSTL) is then proposed to automatically identify the correlated data in the source domain and their weights to facilitate the modeling of the target domain using the classical LS-SVM framework [9]. The major contributions of this work include:

1) The novel CCR is proposed to achieve classification selective transfer learning, which minimizes the disagreement of the LS-SVM based classifiers between the source and target domains, as measured by the CCR term, and leads to the performance improvement in the target domain.

2) The proposed CSTL approach selectively leverages the correlated source data by maintaining the consistency in the distribution of the classification errors between the source domain and the target domain.

3) The fast leave-one-out cross-validation strategy is developed to accelerate selective sampling for the determination of the correlated source data and their weights.

4) Since the correlated source data are only considered in the knowledge transfer, the decision making process for future data can be speeded up as CSTL reduces the number of support vectors.

The rest of the paper is organized as follows. Section 2 discusses the related works. Section 3 describes the mathematical notations involved in the study and presents the proposed CCR approach, where the performance improvement of the target classifier is shown theoretically. Section 4 discusses the proposed CSTL method in detail. Section 5 presents the experiments conducted to evaluate the performance of CSTL and compares it with that of the state-of-the-art approaches. Section 6 gives an overall discussion and concludes the paper with possible avenues for future research.

## 2. Related Works

In 2005, the Broad Agency Announcement (BAA) 05-29 of the Defense Advanced Research Projects Agency (DARPA)'s Information Processing Technology Office (IPTO), defines transfer learning as the ability of a system to recognize and apply knowledge and skills learned in previous tasks to novel tasks. Transfer learning aims to exploit the useful transferred knowledge from the source domain and propagate them as the supervision information to the target domain. It is in analogy with the process of human cognition.

~~However, one major computational problem of existing transfer learning methods is that the transferring skills should be done in which and how many data samples from the source domain. Not all the data collected or learned in the past are relevant to target domain to the same extent. The main limitation of most current transfer learning methods is that they assume all the source data could be leveraged to exploit the shared knowledge structure for target domain. In many real scenarios, preserving the correlated source data with their weights for target domain is more important to make learning models more effective.~~

Two primary transfer learning methods concerning selective knowledge transfer based on support vector machine are Adapting SVM (ASVM) [16] and Selective Transfer Machine (STM) [22]. ASVM aims to minimize both the classification error over the training examples and the discrepancy between the adapted and the original classifier. It also provides a selective sampling strategy based on the loss minimization principle to seed the most informative examples for classifier adaptation. Despite the success [16], the sample selection strategy based on the minimization of the expected risk requires re-training in order to update the estimate of the expected loss on the sample set of the last iteration, making the method inefficient and inaccurate. On the other hand, STM reassigns the weights by reducing the difference between the means of the source and target domains by the Kernel Mean Matching (KMM) [23] method. As pointed out in [22], the limited number of target samples lead to unreliability in the estimated weights for the source samples and thus unsatisfactory performance of STM. Another shortcoming of STM is the high computational cost of re-weighting incurred, i.e., cubic time of the number of target samples. These works give rise to two open questions: (i) Are there any other reliable methods for incorporating prior knowledge? (ii) Is it easier to learn new target objects which are similar to some the objects in the source domain? These questions motivates us to develop a new knowledge transfer method that is able to autonomously identify the most correlated objects and their weights.

## 3. Basic Concepts and the Proposed Regularization

In this section, we introduce the proposed regularization approach CCR and show theoretically that by minimizing the disagreement of the classifiers, as measured by CCR, between the source and target domains, the performance of the target classifier can be improved. Before introducing the CCR, the notations used in the paper are firstly described.

### *3.1. Mathematical Notations*

The mathematical notations and definitions used in this paper are introduced as follows. $\boldsymbol{\eta}=[\eta_1,\eta_2,....\eta_N]^T \in \mathrm{R}^N$ denotes a column vector. $\mathbf{Q}\in \mathrm{R}^{M\times N}$ denotes a matrix with $\mathbf{Q}_{ij}$ corresponding to its $(i,j)$ element. Let $\|\boldsymbol{\eta}\|_p := \left(\sum_{i=1}^{N}|\eta_i|^p\right)^{1/p}$ denote the *p-norm* of the vector $\boldsymbol{\eta}\in \mathrm{R}^N$. Suppose $D_S=\left\{\left(\mathbf{x}_{S_1},y_{S_1}\right),\left(\mathbf{x}_{S_2},y_{S_2}\right),....,\left(\mathbf{x}_{S_{N_S}},y_{S_{N_S}}\right)\right\}$ is a source domain with $N_S$ samples , $\mathbf{x}_{S_i}\in \mathbf{X}_S$ is the data instance and $y_{S_i}\in \mathbf{Y}_S$ is the corresponding class label. Similarly, we denote the target domain data as $D_T=\left\{\left(\mathbf{x}_{T_1},y_{T_1}\right),\left(\mathbf{x}_{T_2},y_{T_2}\right),....,\left(\mathbf{x}_{T_{N_T}},y_{T_{N_T}}\right)\right\}$, where $\mathbf{x}_{T_i}\in \mathbf{X}_T$ is input data and $y_{T_i}\in \mathbf{Y}_T$ is the corresponding output. Without loss of generality, we consider a binary classification problem with the labels $\mathbf{Y}\in\{-1,1\}$ in this paper.

### *3.2. Classification-Error-based Consensus Regularization*

A standard binary classification learning system with least square loss function can be formulated as an optimization problem with the aim of finding the following decision function $f$ in the hypothesis space of function H

$$\begin{aligned}\min_{f}\quad & \Phi(f)+C\sum_{i=1}^{N}\xi_i^2 \\ \text{s.t.}\quad & y_i=f(\mathbf{x}_i)+\xi_i,\quad \forall i=1,2,...,N.\end{aligned}, \tag{1}$$

where $\xi_i$ is the classification error of the data vector $\mathbf{x}_i$ corresponding to the classifier $f$, and $\Phi(f)$ is the regularization term to avoid overfitting and guarantee good generalization performance. The coefficient $C>0$ is a trade-off parameter.
In the hypothesis space H , all the linear models have the following form

$$f(\mathbf{x})=\mathbf{w}^T\phi(\mathbf{x})+b\,, \tag{2}$$

where $\phi(\mathbf{x})$ is a feature mapping function. The kernel functional can be expressed as $K(\mathbf{x},\mathbf{x}')=\phi(\mathbf{x})^T\phi(\mathbf{x}')$. We also set the regularization term $\Phi(f)$ to be $\frac{1}{2}\|\mathbf{w}\|^2$, so that the classification error can be represented as

$$\xi_i=y_i-f(\mathbf{x}_i)=y_i-\left(\mathbf{w}^T\phi(\mathbf{x}_i)+b\right). \tag{3}$$

Here, $\xi_i, i=1,\ldots,N$ measure the classification errors of all the objects.

Classical inductive transfer learning methods focus on how much knowledge can be eventually transferred from the source domain $D_S$. However, in many real-world scenarios,

not all objects collected in the source domain have the same, or very similar distributions, as that in the target domain. If all the source data are used, it will induce negative transfer effect. So, it is desirable to design a robust selective knowledge transfer method which can automatically pick out the correlated source data and their weight. In this paper, we develop a fast cross-validation based selective sampling method based on the LS-SVM framework.

Human beings can always sum up the error in the process of previous learning and use it to guide the inference of new target. By simulating the process of human cognition in ACT framework, we design a novel classification-error-based regularization term to identify how many objects could be transferred from source domain through keeping consistency between the distribution of classification errors of source domain and that of target domain.

***Definition 1:*** (*Classification-Error-based Consensus Regularization*) Suppose the classification errors $\xi_{S_1}, \xi_{S_2}, ...., \xi_{S_{N_S}}$ of the source data and the classification errors $\xi_{T_1}, \xi_{T_2}, ...., \xi_{T_{N_T}}$ of the target data have been already found by Eq. (3), then the CCR term can be formulated as

$$\Delta = \sum_{i=1}^{N_T} \sum_{j=1}^{N_S} \left( \xi_{T_i} - \xi_{S_j} \right)^2 . \tag{4}$$

***Remark 1:*** By using the Parzen window estimation [24] of the Gaussian type, the densities of the classification errors in the source and target domains can be approximated respectively as

$$P_S(\xi) = \frac{1}{N_S(\sqrt{2\pi}\sigma_S)^d} \sum_{i=1}^{N_S} e^{-\frac{\left\| \xi - \xi_{S_i} \right\|^2}{2\sigma_S^2}} \quad \text{and} \tag{5}$$

$$P_T(\xi) = \frac{1}{N_T(\sqrt{2\pi}\sigma_T)^d} \sum_{j=1}^{N_T} e^{-\frac{\left\| \xi - \xi_{T_j} \right\|^2}{2\sigma_T^2}} , \tag{6}$$

where $\sigma$ denotes the Gaussian kernel bandwidth. According to Eq. (5) and Eq. (6), $\int_{D_S, D_T} (P_S(\xi) - P_T(\xi))^2 d\xi$ should be as small as possible considering the shared knowledge structure between the two domains.

$$\begin{aligned} &\int_{D_S, D_T} (P_S(\xi) - P_T(\xi))^2 d\xi \\ &= \int_{D_S} (P_S(\xi))^2 d\xi - 2\int_{D_S, D_T} P_S(\xi) P_T(\xi) d\xi + \int_{D_T} (P_T(\xi))^2 d\xi \end{aligned} . \tag{7}$$

Without any prior knowledge, we can reasonably assume $\int_{D_S} P_S(\xi) d\xi = 1$, thus

$$\int_{D_S} (P_S(\xi))^2 d\xi = \int_{D_S} P_S(\xi) P_S(\xi) d\xi = E[P_S(\xi)] \approx \frac{1}{N_S} . \tag{8}$$

Similarly, $\int_{D_T}(P_T(\xi))^2 d\xi \approx 1/N_T$ . So Eq. (7) can be reformulated as

$$\begin{aligned} & E[P_S(\xi)] - 2\int_{D_S,D_T} P_S(\xi)P_T(\xi)d\xi + E[P_T(\xi)] \\ & \approx \frac{1}{N_S} - 2\int_{D_S,D_T} P_S(\xi)P_T(\xi)d\xi + \frac{1}{N_T} \end{aligned}, \tag{9}$$

such that $\Delta_1 = \int_{D_S,D_T} P_S(\xi)P_T(\xi)d\xi$ should be as large as possible. According to [25], the following relationship holds

$$\int G\left(\xi,\xi_{T_j},\sigma_T\right)G\left(\xi,\xi_{S_i},\sigma_S\right)d\xi = G\left(\xi_{T_j}-\xi_{S_i},\sigma_T+\sigma_S\right), \tag{10}$$

where the Gaussian distribution function $G(\xi,\xi_i,\sigma) = \frac{1}{(\sqrt{2\pi}\sigma)^d}\exp\left(-\frac{\|\xi-\xi_i\|^2}{2\sigma^2}\right)$. Thus we have

$$\Delta_1 = \frac{1}{N_S N_T}\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\frac{1}{(\sqrt{2\pi}(\sigma_T+\sigma_S))^d}e^{-\frac{\left\|\xi_{T_i}-\xi_{S_j}\right\|^2}{2(\sigma_T+\sigma_S)^2}}. \tag{11}$$

Since maximizing $e^{-x^2}$ is equivalent to minimizing $x^2$, in order to minimize $\int_{D_S,D_T}(P_S(\xi)-P_T(\xi))^2 d\xi$ (i.e., maximize $\Delta_1$) between the source and target domains, we $\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left\|\xi_{T_i}-\xi_{S_j}\right\|^2 \Big/ 2(\sigma_T+\sigma_S)^2$ should be as small as possible, i.e., we should minimize the term $\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left(\xi_{T_i}-\xi_{S_j}\right)^2$.

***Theorem 1****:* Minimizing the disagreement between the LS-SVM based classifiers, as measured by the CCR term in Eq. (4), of the source and the target can improve the performance of the target classifier.

***Proof****:* The proof is in Appendix.

The CCR for transfer learning will be covered in detail in the following section. It has the following merits: 1) to the best of our knowledge, this is the first endeavor that consensus regularization based on classification error is considered as the knowledge transfer regularization term in LS-SVM based transfer learning research; 2) based on the CCR, the prediction for future target samples can be readily achieved with the proposed CSTL; 3) the fast leave-one-out cross-validation strategy can be developed for tuning the transfer process autonomously for selective transfer learning and simultaneously identifying the correlated objects and their weights from the source domain.

## 4. The Proposed CSTL Method

To avoid negative knowledge transfer caused by the portion of source data that are irrelevant to the target, we propose the CSTL method based on the CCR. Within the LS-SVM framework, CSTL yield an elegant formulation for identifying the correlated source data which have minimal discrepancy in classification error distribution with respect to the target data. Besides, to properly scale the importance of the correlated source data, different weighting factors $\eta_j$ are introduced to the different classification errors $\xi_{S_j}$ in Eq. (4). In this case, these weights are regarded as the learning parameters and can be chosen by the leave-one-out cross-validation strategy. Consequently, the identified correlated source data and their weights can be leveraged to explore the transferred knowledge for the construction of the target classifier. The overall framework of the CSTL system is illustrated in Fig. 2.

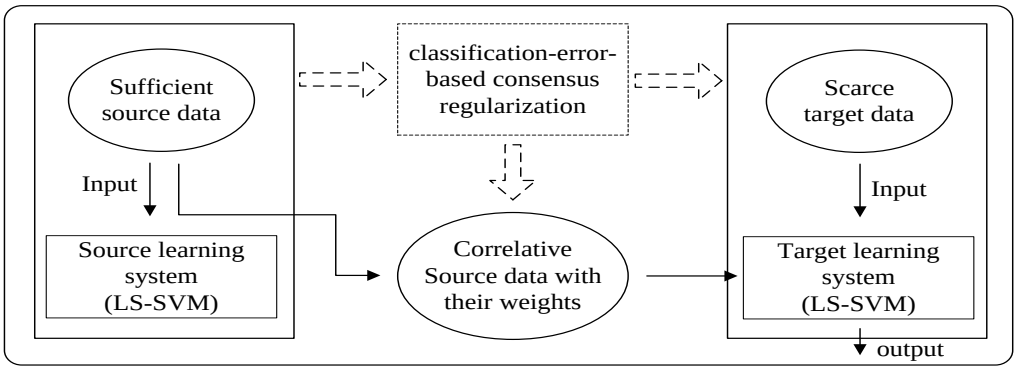



Fig. 2 Overall framework of the CSTL system.

### *4.1. LS-SVM-based Selective Transfer Learning*

Instead of equally weighing the classification errors $\xi_{S_j}$, $j=1,2,....,N_S$, in Eq. (4), the CSTL method is developed by linearly combining the classification errors of the source data, i.e., $\sum_{j=1}^{N_S}\eta_j\xi_{S_j}$. The objective function based on the LS-SVM framework can be formulated as

$$\begin{aligned}\min_{\mathbf{w},b}\quad & \frac{1}{2}\|\mathbf{w}\|^2+\frac{C}{2}\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left(\xi_{T_i}-\eta_j\xi_{S_j}\right)^2 \\ \text{s.t.}\quad & \mathrm{y}_i=w^T\phi(\mathbf{x}_i)+b+\xi_i,\quad \forall i=1,2,...,N_T,\forall j=1,2,...,N_S.\end{aligned}, \tag{12}$$

where the weighting parameter $\eta_j$ can be used to determine whether the *j*th source data can be leveraged for transfer learning or not. Moreover, the larger the value of $\eta_j$, the more important the *j*th source data. It can be seen form Eq. (12) that the proposed objective

function aims to select the correlated source data and the corresponding weight parameter $\eta_j$ by minimizing the difference between the two distributions of classification errors. The corresponding Lagrangian $L$ of Eq. (12) is formulated as

$$L = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{2}\sum_{i=1}^{N_T}\sum_{j=1}^{N_S}\left(\xi_{T_i} - \eta_j\xi_{S_j}\right)^2 - \sum_{i=1}^{N_T}\alpha_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b + \xi_{T_i} - y_i)\,, \tag{13}$$

where $\alpha_i$ is the Lagrangian multiplier. By taking the derivative of $L$ with respect to $\alpha_i$, $\mathbf{w}$, $b$ and $\xi_{T_i}$ respectively and set them to 0, we obtain

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N_T}\alpha_i\phi(\mathbf{x}_i), \tag{14}$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N_T}\alpha_i = 0\,, \tag{15}$$

$$\frac{\partial L}{\partial \xi_{T_i}} = 0 \Rightarrow \xi_{T_i} = \frac{\alpha_i}{C\cdot N_S} + \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\,, \tag{16}$$

$$\frac{\partial L}{\partial \alpha_i} = 0 \Rightarrow \mathbf{w}^T\phi(\mathbf{x}_i) + b + \xi_{T_i} = y_i\,. \tag{17}$$

By combining the formulations Eq. (14), Eq. (15) and Eq. (16) with Eq. (17), we have

$$\sum_{i=1}^{N_T}\alpha_i\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_k) + b + \frac{\alpha_k}{C\cdot N_S} = y_k - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\,. \tag{18}$$

Based on the kernel trick, we can rewrite the system of linear equations in the form of a matrix, i.e.,

$$\begin{bmatrix} \boldsymbol{K} + \frac{1}{C}\boldsymbol{\Lambda} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix}\begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\mathbf{1} \\ 0 \end{bmatrix}, \tag{19}$$

where $\boldsymbol{y}$ is the label vector of the known target data, i.e. $\boldsymbol{y} = \left[y_1, y_2, ...., y_{N_T}\right]^T$. Moreover, $\boldsymbol{\Lambda} = diag\left\{N_S^{-1}, N_S^{-1}, ...., N_S^{-1}\right\}$. Then the model parameters can be calculated by matrix inversion

$$\begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \mathbf{Q}\begin{bmatrix} \boldsymbol{y} - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\mathbf{1} \\ 0 \end{bmatrix}, \tag{20}$$

where $\mathbf{Q}$ equals to the inverse of the matrix $\mathbf{H}$, and $\mathbf{H}$ is the first term on the left in Eq. (19). Besides, the pre-trained classification errors of the source domain can be computed in advance using LS-SVM.

### *4.2. Fast Selective Sampling for Source Domain*

Finding the optimal weight vector $\boldsymbol{\eta}$ is the key to identify the correlated subset of the source data and correctly weight these source data. Here, a fast leave-one-out cross validation method is exploited to find the optimal $\boldsymbol{\eta}$. Within the LS-SVM framework, the

leave-one-out prediction of the scarcely available target training data in Eq. (12) can be formulated in a closed form at negligible additional computational cost [26]. This fast selective sampling method is discussed as follows.

***Theorem 2:*** By reformulating Eq. (20) as $\left[\boldsymbol{\alpha}'^T, b'\right]^T = \mathbf{Q}\left[\boldsymbol{y}^T, 0\right]^T$, $\left[\boldsymbol{\alpha}''^T_j, b''_j\right]^T = \mathbf{Q}\left[\xi_{S_j}\mathbf{1}^T, 0\right]^T$ and $\boldsymbol{\alpha} = \boldsymbol{\alpha}' - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\boldsymbol{\alpha}''_j$, then the prediction of a sample $\mathbf{x}_{T_i}$ (i.e., $\tilde{y}_i$, $i = 1,2,....,N_T$), when it is removed from the training set of target domain, can be represented as

$$\tilde{y}_i = y_i - \frac{\alpha'_i}{\mathbf{Q}_{ii}} + \frac{\boldsymbol{\eta}^T\boldsymbol{\xi}_S}{\mathbf{Q}_{ii}}, \tag{21}$$

where $\boldsymbol{\eta} \in \mathbb{R}^J$ is a vector containing all the values of $\eta_j$; and $\boldsymbol{\xi}_S$ is also a vector containing the classification errors, denoted as $\left[\xi_{S_1}, \xi_{S_2}, ...., \xi_{S_{N_S}}\right]^T$, of all the samples in the source domain.

***Proof:*** By isolating the first row and the first column, we decompose the matrix $\mathbf{H}$ into the block representation as

$$\mathbf{H} = \begin{bmatrix} \boldsymbol{K} + \frac{1}{C}\boldsymbol{\Lambda} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} = \begin{bmatrix} h_{11} & \mathbf{h}_1^T \\ \mathbf{h}_1 & \mathbf{H}_{(-1)} \end{bmatrix}. \tag{22}$$

Let $\boldsymbol{\alpha}_{(-i)}$ and $b_{(-i)}$ denote the model parameter in the $i$th iteration of the leave-one-out cross validation procedure. Thus, in the first iteration, i.e., excluding the first training sample in target domain, we have

$$\begin{bmatrix} \boldsymbol{\alpha}_{(-1)} \\ b_{(-1)} \end{bmatrix} = \mathbf{Q}_{(-1)}\left(\boldsymbol{y}_{(-1)} - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\mathbf{1}_{(-1)}\right), \tag{23}$$

Where $\mathbf{Q}_{(-1)} = \mathbf{H}_{(-1)}^{-1}$, and $\boldsymbol{y}_{(-1)} = \left[y_2, y_3, ...., y_{N_T}, 0\right]$. According to Eq. (20), we can obtain the leave-one-out prediction of the first training sample as follows,

$$\begin{aligned} \tilde{y}_1 &= \mathbf{h}_1^T\begin{bmatrix} \boldsymbol{\alpha}_{(-1)} \\ b_{(-1)} \end{bmatrix} + \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j} \\ &= \mathbf{h}_1^T\mathbf{Q}_{(-1)}\left(\mathbf{y}_{(-1)} - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\mathbf{1}_{(-1)}\right) + \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j} \end{aligned}. \tag{24}$$

In Eq. (19), the last $N_T$ equations in the system can be represented as $\left[\mathbf{h}_1 \quad \mathbf{H}_{(-1)}\right]\left[\boldsymbol{\alpha}^T \quad b\right]^T = \left(\mathbf{y}_{(-1)} - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j}\mathbf{1}_{(-1)}\right)$. Thus, Eq. (24) can be reformulated as

$$\begin{aligned}\tilde{y}_1 &= \mathbf{h}_1^T\mathbf{Q}_{(-1)}\left[\mathbf{h}_1 \quad \mathbf{H}_{(-1)}\right]\left[\alpha_1,...,\alpha_{N_T},b\right]^T + \frac{1}{N_S}\sum\nolimits_{j=1}^{N_S}\eta_j\xi_{S_j} \\ &= \mathbf{h}_1^T\mathbf{Q}_{(-1)}\mathbf{h}_1\alpha_1 + \mathbf{h}_1^T\left[\alpha_2,...,\alpha_{N_T},b\right]^T + \frac{1}{N_S}\sum\nolimits_{j=1}^{N_S}\eta_j\xi_{S_j}\end{aligned}. \tag{25}$$

In Eq. (19), the first equation in the system is $y_1 - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\xi_{S_j} = h_{11}\alpha_1 + \mathbf{h}_1^T\left[\alpha_2,\alpha_3,....,\alpha_{N_T},b\right]^T$, so we obtain $\tilde{y}_1 = y_1 - \alpha_1\left(h_{11} - \mathbf{h}_1^T\mathbf{Q}_{(-1)}\mathbf{h}_1\right)$. Using the block matrix inversion lemma and $\mathbf{Q} = \mathbf{H}^{-1}$, we obtain

$$\mathbf{Q} = \begin{bmatrix} \nu^{-1} & -\nu^{-1}\mathbf{h}_1\mathbf{Q}_{(-1)} \\ \mathbf{Q}_{(-1)} + \nu^{-1}\mathbf{Q}_{(-1)}\mathbf{h}_1^T\mathbf{h}_1\mathbf{Q}_{(-1)} & -\nu^{-1}\mathbf{Q}_{(-1)}\mathbf{h}_1^T \end{bmatrix}, \tag{26}$$

where $\nu = h_{11} - \mathbf{h}_1^T\mathbf{Q}_{(-1)}\mathbf{h}_1$.

It can be seen that the system in Eq. (19) is insensitive to permutations of the order of the linear equations, so the leave-one-out prediction of the *i*th training sample can be formulated as

$$\tilde{y}_i = y_i - \alpha_i/\mathbf{Q}_{ii}\ . \tag{27}$$

Using the equations $\left[\boldsymbol{\alpha}'^T,b'\right]^T = \mathbf{Q}\left[\mathbf{y}^T,0\right]^T$, $\left[\boldsymbol{\alpha}_j''^T,b_j''\right]^T = \mathbf{Q}\left[\xi_{S_j}\mathbf{1}^T,0\right]^T$ and $\boldsymbol{\alpha} = \boldsymbol{\alpha}' - \frac{1}{N_S}\sum_{j=1}^{N_S}\eta_j\boldsymbol{\alpha}_j''$, Eq. (27) can be reformulated as $\tilde{y}_i = y_i - \frac{\alpha_i'}{\mathbf{Q}_{ii}} + \frac{1}{N_S}\cdot\frac{\boldsymbol{\eta}^T\boldsymbol{\xi}_S}{\mathbf{Q}_{ii}}$. It is worth noting that $1/N_S$ is a constant which can be absorbed into the vector $\boldsymbol{\eta}$ and Theorem 2 is proved.

***Definition 2***. (*Loss Function*) Define the loss function $l(\cdot,\cdot)$ as

$$l\left(\tilde{y}_i,y_i\right) = \left|1 - \tilde{y}_i y_i\right|_+ = \left|y_i\frac{\alpha_i' - \boldsymbol{\eta}^T\boldsymbol{\xi}_S}{\mathbf{Q}_{ii}}\right|_+, \tag{28}$$

where $|x|_+ = \max\{0,x\}$.

***Remark 2:*** It can be seen from Eq. (21) that the model parameter $\boldsymbol{\alpha}$ depends linearly on the weight vector $\boldsymbol{\eta}$. Thus, if all the weights $\eta_j$, $j = 1,2,....,N_S$, are chosen, the learning model can be formulated. Obviously, the optimal values of $\eta_j$ can yield positive values for $\tilde{y}_i y_i$ for each training sample $\mathbf{x}_{T_i}$ available in the target. However, it is a non-convex formulation to directly maximize the sum of the signs of those quantities. The proposed convex loss function in Definition 2 is the strict upper bound to the leave-one-out misclassification loss

so that the predictions $\tilde{y}_i$ ($i=1,2,....,N_T$) have an absolute value equal or greater than 1, and are of the same sign as $y_i$.

Overall, the problem of optimizing the weight vector $\boldsymbol{\eta}$ can be formulated as

$$\min_{\eta} \sum_{i=1}^{N} l(\tilde{y}_i, y_i), \quad \text{s.t. } \|\boldsymbol{\eta}\|_p \le 1, \ \eta_j \ge 0, \tag{29}$$

where the *p-norm* constraint is used as the regularization form on vector $\boldsymbol{\eta}$. This approach alleviates the overfitting problem that may occur when the number of source data is large compared to that of the available target training data.

In this paper, we categorize the *p-norm* constraint into three types according to the value of *p*. First, when *p*=1, the $L_1$-norm constraint is applied. This constraint sums all the absolute values of the vector elements, i.e., $\|\boldsymbol{\eta}\|_1 = \text{sum}(|\eta_1|, |\eta_2|, \ldots, |\eta_N|)$. According to [27], the optimization problem is easy to implement and can yield sparse solutions. Second, when *p*=2, $L_2$-norm constraint, i.e. the Euclidean norm $\|\cdot\|_2$, is applied. The optimization problem using $L_2$-norm constrains can be implemented through a projected sub-gradient descent algorithm. The pseudo-code is shown in Algorithm 1. Third, when $p=\infty$, $L_\infty$-norm constraint is applied, which returns the maximum absolute value of all the vector elements, i.e., $\|\boldsymbol{\eta}\|_\infty = \max\{|\eta_1|, |\eta_2|, \ldots, |\eta_N|\}$.

---

Algorithm 1: Projected Sub-gradient Descent Algorithm

---

Input: $\boldsymbol{\alpha}'$, $\boldsymbol{\alpha}''$ and $\xi_S$

Initialize: $\boldsymbol{\eta} \leftarrow 0$ and $t \leftarrow 1$

repeat

$$\tilde{y}_i = y_i - \frac{\alpha_i'}{\mathbf{Q}_{ii}} + \frac{\boldsymbol{\eta}^T \boldsymbol{\xi}_S}{\mathbf{Q}_{ii}} \quad \forall \text{ i=1,2,...}N_T$$

$$d_i \leftarrow \mathbf{1}\{\tilde{y}_i y_i > 0\}, \quad \forall \text{ i=1,2,...}N_T$$

$$\boldsymbol{\eta} \leftarrow \boldsymbol{\eta} - \frac{1}{\sqrt{t}} \sum_{i=1}^{N_T} d_i y_i \frac{\boldsymbol{\xi}_S}{N_S \cdot \mathbf{Q}_{ii}}$$

if $\|\boldsymbol{\eta}\|_2 > 1$, then

$$\boldsymbol{\eta} \leftarrow \boldsymbol{\eta} / \|\boldsymbol{\eta}\|_2$$

end if

$$\boldsymbol{\eta} \leftarrow \max(\boldsymbol{\eta}, \mathbf{0})$$

$$t \leftarrow t+1$$

Until convergence

Output: $\boldsymbol{\eta}$

---

*4.3. Decision Function*

The idea of the proposed CSTL is to make the target classification errors close to the linear combination of the prior known source classification errors, i.e., $\sum_{j=1}^{N_S} \eta_j \xi_{S_j}$, so that the new value of $\boldsymbol{\alpha}$ in Eq. (20) can be obtained with the solution of $\boldsymbol{\eta}$ and the optimal solution can be formulated as

$$\mathbf{w} = \sum_{i=1}^{N_T} \alpha_i \phi(\mathbf{x}_i). \tag{30}$$

When it is used for the classification of further target data, we have

$$f(\mathbf{z}) = \mathbf{w}^T \phi(\mathbf{z}) = \sum_{i=1}^{N_T} \alpha_i \phi(\mathbf{x}_i)^T \phi(\mathbf{z}). \tag{31}$$

*4.4. Computational Complexity*

The computational complexity of the proposed CSTL is considered from two aspects, training and prediction. In training, it is given by $\mathrm{O}\left(N_T^3 + t_{\max} N_S N_T\right)$, where $N_T$ is the number of the training samples in the target domain, $N_S$ is the number of objects in the source domain, and $t_{\max}$ is the maximum number of iterations in Algorithm 1. The first term concerns the evaluation of the matrix $\mathbf{Q}$ due to training, the complexity of which is given by that of a plain SVM and in the worst case is $\mathrm{O}\left(N_T^3\right)$ [28]. The second term concerns the computational complexity due to the computation of $\boldsymbol{\eta}$. The complexity of a one iteration of the projected sub-gradient descent algorithm (i.e., Algorithm 1) is $\mathrm{O}(N_S N_T)$, so the total computational complexity is $\mathrm{O}\left(t_{\max} N_S N_T\right)$.

In prediction, unlike the traditional transfer learning methods which take the entire source data into account for training on the target data, CSTL only consider the correlated source objects and their weights. Hence, CSTL involves less number of support vectors from the source objects in the decision function of the target domain and therefore the testing (prediction) time is a shorter and is more appropriate applications that require real-time prediction on future data in the target domain.

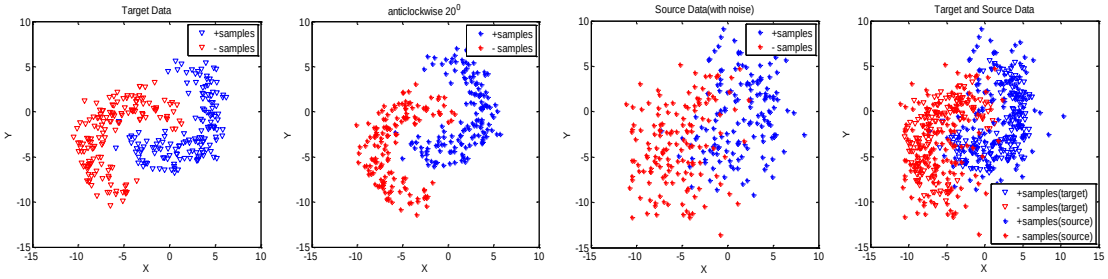
## 5. Experimental Results

In this section, we present the experiments conducted on both synthetic datasets and real-world image and text datasets to evaluate the effectiveness of the proposed CSTL.

*4.1. Data Preparation*

*4.1.1. Synthetic Datasets*

A synthetic target data set was constructed which was composed of 300 samples according to a bi-dimensional pattern of two intertwining images of the moon that were associated with two specific information classes – 150 positive and 150 non-positive samples each. Specifically, the source data were generated by first rotating anticlockwise the original target data by $10^0$, $20^0$, $30^0$ and $40^0$ respectively, and then introducing Gaussian noise (mean = 0, variance = 2) to each data point after the rotations. Fig. 3(a) shows an example of the target datasets. The data were then rotated anticlockwise by 20° in Fig. 3(b), followed by the introduction of Gaussian noise in Fig. 3(c) to obtain the source data. Fig. 3(d) shows the combination of the source and target data, from which we can see that some suitable source data could be selected as auxiliary knowledge for transfer learning for the case when a rotation of 20° was applied. The distribution of the source and target data were different distributions due to the rotation and noise applied. In particular, a greater rotation will result in a more difficult transfer learning problem, which can also be seen from the resulting values of Jensen–Shannon scatter (DJS) [29].




(a) (b) (c) (d)

Fig. 3 The target data in (a) was rotated by 20° in (b) and then modified with Gaussian noise in (c) to obtain the source data. The source and target data were combined in (d).

Table 1. Details of the Image Datasets

| Tasks | Caltech-256 | | | | Tasks | Digital Images | | | |
|---|---|---|---|---|---|---|---|---|---|
| | **Source datasets** | | **Target datasets** | | | **Source datasets** | | **Target datasets** | |
| | **Positive class** | **Negative class** | **Positive class** | **Negative class** | | **Positive class** | **Negative class** | **Positive class** | **Negative class** |
| Task 1 | *Fire-truck* | *Snowmobiles* | *Bulldozer* | *Moto-bikes* | Task 7 | *USPS7* | *USPS9* | *MNIST7* | *MNIST9* |
| Task 2 | *School-bus* | *Snowmobiles* | *Bulldozer* | *Moto-bikes* | Task 8 | *MNIST7* | *MNIST9* | *USPS7* | *USPS9* |
| Task 3 | *Car-side* | *Snowmobiles* | *Bulldozer* | *Moto-bikes* | Task 9 | *USPS4* | *USPS9* | *MNIST4* | *MNIST9* |
| Task 4 | *Fire-truck* | *Moto-bikes* | *Bulldozer* | *Snowmobiles* | Task 10 | *MNIST4* | *MNIST9* | *USPS4* | *USPS9* |
| Task 5 | *School-bus* | *Moto-bikes* | *Bulldozer* | *Snowmobiles* | Task 11 | *USPS0* | *USPS6* | *MNIST0* | *MNIST6* |

| Task 6 | *Car-side* | *Moto-bikes* | *Bulldozer* | *Snowmobiles* | Task 12 | *MNIST0* | *MNIST6* | *USPS0* | *USPS6* |
|---|---|---|---|---|---|---|---|---|---|

*4.1.2. Image Datasets*

The three image datasets *Caltech-256* [30], *USPS* and *MNIST* [32] that are broadly adopted in computer vision literature were employed in the experiment. The *Caltech-256* dataset contains 30607 images of objects belonging to 256 categories. In this paper, we adopt all the pictures of the vehicle class which contains 5 sub-categories: *fire-truck, school-bus, car-side, Moto-bike and snow-mobile*. The pre-computed features [30] were downloaded and the PHOG shape descriptors [31] were selected. The images of *Moto-bike* and *snow-mobile* were used as the negative samples while the others as the positive. Table 1 gives the details about of the experimental datasets.

*USPS* and *MNIST* are digital datasets. The *USPS* dataset contains 7,291 training images and 2,007 test images with a size of 16×16 pixels. The *MNIST* dataset contains 60,000 training images and 10,000 test images with a size of 28×28 pixels. Although *USPS* and *MNIST* have different distributions, both of them share 10 common semantic classes, each corresponding to one digit. Some deceptive sub-categories were selected for the current learning task, e.g. the manuscript digits "7" and "9". With reference to the work in [32], 1,800 images were randomly sampled from *USPS* to form the source domain and 2,000 images were randomly sampled from *MNIST* to form the target domain. Then, the source domain and the target domain were swapped to form another dataset. All the images were re-scaled to the size of 16×16 pixels, transformed into gray scale, and represented by a 256-dimensional vector. This ensures that all the source and target data share the same feature space. The details of the experimental datasets can be found in the Table 1.

*4.1.3. Text Datasets*

Email spam filtering is widely used for evaluating and benchmarking transfer learning methods [33]. The email spam filtering datasets [34] contains one public email set and three email subsets, i.e., User1, User2 and User3, which are identified respectively by three different users. The public email set has 4,000 emails. Each subset contains 2,500 emails and is divided equally in quantity into two specific classes – spam and non-spam emails. In the experiment, we constructed three datasets by using the public email set to form the source domain, and each subset respectively as the target domains. Then the source and target in these three datasets are swapped to obtain another three datasets. In addition, the word-frequency feature [34] of the emails is adopted. The details of the datasets can be found in Table 2.

Table 2. Details of the Text Datasets

| Tasks | Source datasets | Target datasets | Tasks | Source datasets | Target datasets |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| Task 1 | Public | User1 | Task 4 | User2 | Public |
| Task 2 | User1 | Public | Task 5 | Public | User3 |
| Task 3 | Public | User2 | Task 6 | User3 | Public |

*4.2. Experimental Setup*

*4.2.1. Baseline Methods*

Six methods were considered in the experiments. They were (1) Least Square Support Vector Machine (LS-SVM) [9]; (2) Adaptive SVM (ASVM) [16]; (3) Cross-Domain SVM (CDSVM) [18]; (4) Boosting for Transfer Learning (TrAdaBoost) [2]; (5) Selective Transfer Machine (STM) [22]; and (6) the proposed CSTL in this paper.

*4.2.2. Implementation Details*

For LS-SVM, we simply regarded the labeled target data as the training samples so that the effectiveness of the auxiliary knowledge could be evaluated easily. The other five methods, i.e. ASVM, CDSVM, TrAdaBoost, STM and CSTL, can be casted into the transfer learning category. For STM, the weights of the labeled target samples were set to 1, and combined with the re-weighted source samples to form the new training sets [22]. Although samples were selected from the source domain for both ASVM and CSTL, the selected samples were weighted in CSTL which enables it to outperform ASVM or at least achieving comparable performance. This will be demonstrated in the experiments. Each transfer learning method was run in an inductive way, i.e., some labeled data in the target domain were required for creating an objective predictive model for use in the target domain. The number of the labeled target data was increased in the subsequent steps until the classification result converged. Each method was executed 10 times and the labeled samples were extracted randomly at each time. The average performance was then reported.

The Gaussian kernel $K(\mathbf{x}_1,\mathbf{x}_2)=\exp\left(-\gamma\|\mathbf{x}_1-\mathbf{x}_2\|^2\right)$ was used on both the source and target domains for all the experiments. Under this experimental setting, it was impossible to automatically tune the optimal parameters for the target classifier using the 5-fold cross validation. The trade-off parameter $C$ was set by searching the value from the set $\{1e\text{-}5,1e\text{-}4,1e\text{-}3,1e\text{-}2,1e\text{-}1,1e0,1e1,1e2,1e3,1e4,1e5\}$. The Gaussian kernel parameter $\gamma$ was set by searching the value from the set $\{1e\text{-}5,1e\text{-}4,1e\text{-}3,1e\text{-}2,1e\text{-}1,1e0,1e1,1e2,1e3,1e4,1e5\}$. To fully define the proposed CSTL, it is necessary to choose the $p$ value in the constraint in Eq. (29). Besides, we computed the LS-SVM based classification error of the source domain in

advance. According to [22], we set the parameter $B=1000$, $\varepsilon=\sqrt{N_T-1}/\sqrt{N_T}$ for STM. We obtained the results empirically for $p=1,2,\infty$, denoted as $CSTL_1$, $CSTL_2$ and $CSLT_\infty$, on the synthetic datasets. In this paper, CSTL is referred to as $CSTL_2$. All the algorithms were implemented using MATLAB on a computer with Intel Core i3-3240 3.40 GHz CPU, 4GB RAM and a 32-bit operation system.

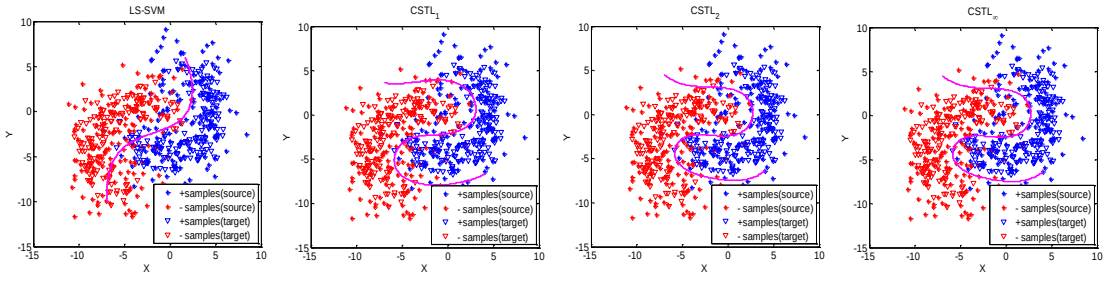
### *4.3. Experimental Results*

In this section, the performance of the CSTL is compared with that of the other five methods in terms of classification accuracy.

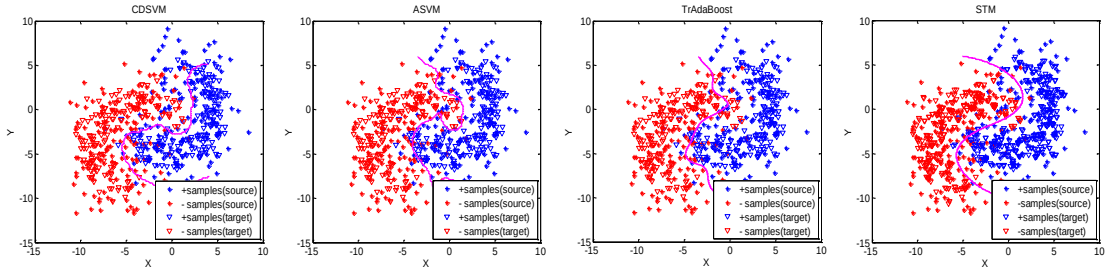
#### *4.3.1. Cross-Domain Synthetic Data Classification*

Figs. 4, 5 and 6 show the classification accuracy and the running time of the six methods on the synthetic data sets. Fig. 4 shows the learning performance when the source data were generated by rotating the target data 20° anticlockwise and only 30 labeled target samples were considered. It can be seen that CSTL can achieve perfect separation of the classes even if the rotation angles ranged from 10° to 40°, as shown in Fig. 6. The following results are obtained:

(1) In terms of the data-generation way in the above, not all data points of the source domain shown in Fig.4 were suitable for knowledge transfer, and hence the transfer learning methods that used the entire source knowledge were adversely affected by the noise. Obviously, the classification results of CSTL, as shown in Fig. 4(b), 4(c) and 4(d), were better than that of the other five methods according to the 5-fold cross validation on the training data. The success was attributed to the ability of CSTL in identifying correlated data of the source domain and leveraging them to enhance the classification performance for the data in the target domain.




(a) (b) (c) (d)




(e) (f) (g) (h)

Fig.4 Classification accuracy of the six methods on source data obtained by rotating the target data anticlockwise by 20°. Thirty labeled training samples in the training data were used.

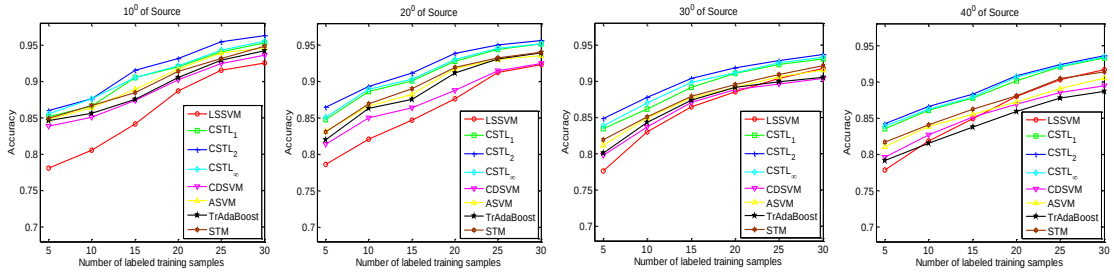

(a) (b)

Fig.5 Running time of the six methods on the anticlockwise $20^0$ rotation of target data as source data with 30 labeled training samples being used in target data.

(2) CSTL can automatically identify the relevance of each source object for transfer learning to the target data. The vector $\boldsymbol{\eta}$ reflects the importance of all the source objects. In Figs. 4(b), 4(c) and 4(d), the number of nonzero entries in vector $\boldsymbol{\eta}$ are 158, 142 and 142 respectively for $\text{CSTL}_1$, $\text{CSTL}_2$ and $\text{CSTL}_\infty$. It can be seen from the experiment that the number of correlated source objects accounted for about fifty percent of the source data.

(3) Fig. 5(a) shows the running time of the six methods on the datasets where the source data were obtained by rotating the target data 20° anticlockwise and considering only 30 labeled target samples. Here, the running time was defined as the training time plus the testing time for all the unlabeled data in the target domain. Among the five transfer learning methods, the timing performance of CSTL was the best. Even CSTL needs to compute the weights of the selected source samples, the total running time is still less than other transfer learning methods, which is attributed to the lesser amount of source objects involved in the training. Accordingly, the number of support vectors from the source objects that are involved in the decision function is also smaller, which implies that CSTL has shorter testing time (see Fig.5(b)), particularly when real-time prediction is required for the future or unlabeled objects in the target domain.

(4) Fig. 6 shows the classification accuracy for data obtained by rotations at four different angles. The classification accuracy of CDSVM, ASVM, TrAdaBoost and STM decrease dramatically, which was due to the increasing difference between the source and target domains. Satisfactory performance could still be achieved with the proposed CSTL because of its ability to select relevant and useful source objects and assign them with appropriate weights to preserve the consistency in the distribution of the classification errors in the two domains.




(a) (b) (c) (d)

Fig. 6 Classification accuracy for data obtained rotations at four different angles.

*4.3.2. Cross-Domain Image Classification*

It can be seen from the experimental results on the synthetic datasets that the choice of $\eta$ did not produce significance difference among the three versions of CSTL. Among them, the one with $p$ =2 showed slightly better performance. Hence, the 2-norm constraint was used for the experiments to be discussed in this section.

Fig. 7 shows the classification accuracies of the six methods on the image dataset. The following observations can be made.

(1) Figs. 7(a)-7(f) show the average classification performance on the *Caltech-256* dataset, and Figs. 7(g)-7(l) on the digital image dataset. The proposed CSTL demonstrated very promising results in most cases. The major limitation of CDSVM, ASVM and TrAdaBoost is that these methods consider all the source objects, or take selected samples into account but without considering their relatively importance. Therefore, they cannot generalize the model for the target model appropriately. For STM, the weights re-assignment by reducing the difference between the means of the source and target domains may lead to inaccuracy, due to the need of a larger dataset to estimate the importance of the weights by the Kernel Mean Matching method adopted in STM [22, 35].

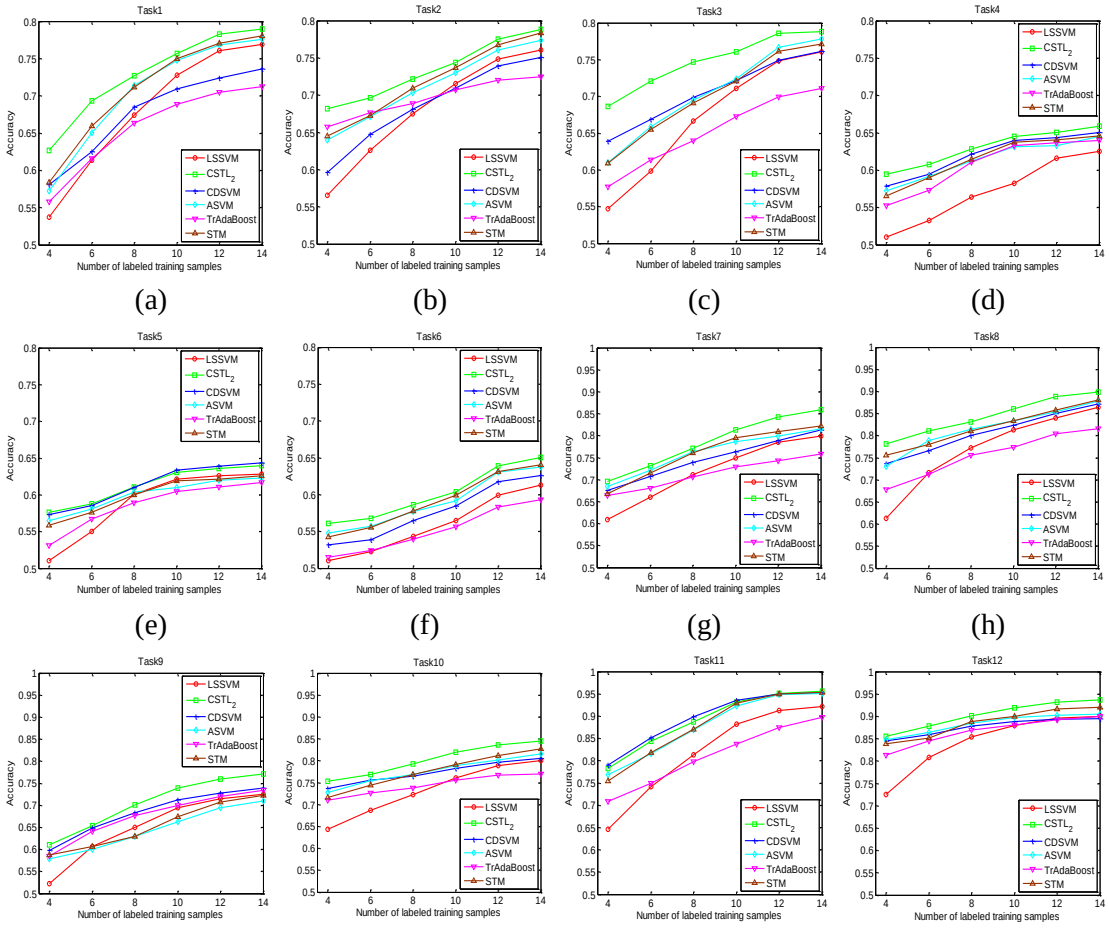



Fig.7 Classification accuracy obtained by six methods on the two image datasets.

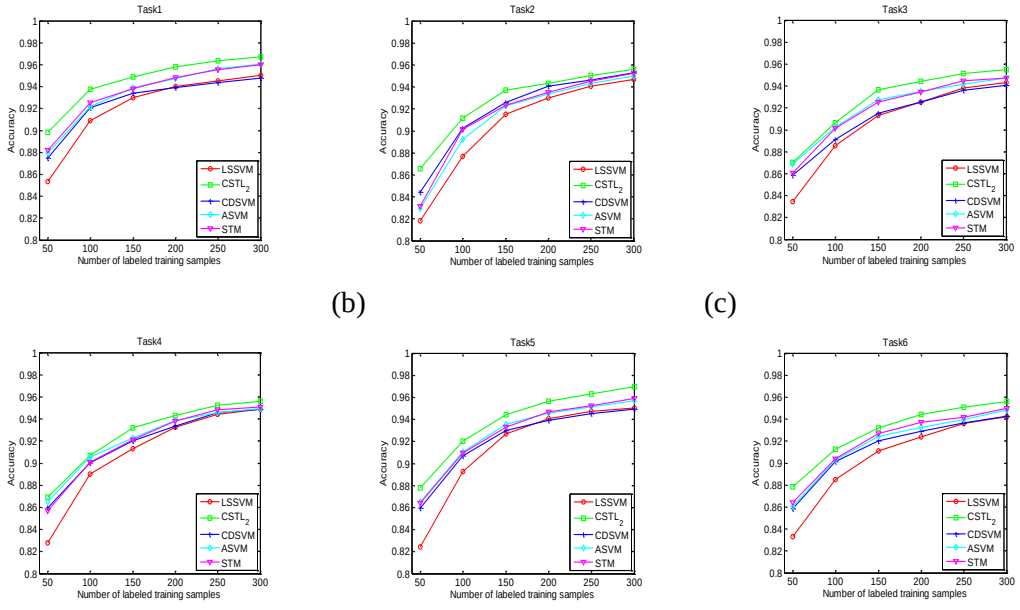

Fig.8 Running time obtained by six methods on the two image datasets.

For the proposed CSTL, with CCR, the correlated data and their weights in the source domain were determined by the fast unbiased leave-one-out cross-validation strategy, which did not introduce negative transfer learning effect and produced better generalized result. These experimental results show that considering only the related source objects can improve classification performance.

(2) Among the transfer learning methods, TrAdaBoost always showed the worst classification performance because it was sensitive to the quality (or KL-divergence) of the data distribution [2]. Additionally, most transfer learning methods outperformed LS-SVM when the number of labeled training data was small. With the increase in the number of labeled training data, the classification performance would degrade below that of the LS-SVM if the source knowledge was still fully exploited in the training process of the classification for the unlabeled target data.

(3) It can be seen from Figs. 7(a)-7(l) that, with increasing number of labeled training samples, the classification accuracy of LS-SVM would improve and exceed the transfer learning methods. This is potentially due to the abundant supply of training data of the target domain that enables LS-SVM to better supervise the inferences with the unlabeled target data. For the proposed CSTL, it leveraged specifically the useful information and resulted in performance improvement for future inferences. The selection of related samples through CCR can avoid overfitting effectively.

(4) The histogram in Fig. 8(a) shows the average running time of the six methods on the *Caltech-256* image dataset for Task 6 in Table 1, with respect to the increasing number of labeled training samples. Similarly, Fig. 8(b) illustrates the average running time of the six methods for Task 12 in Table 1. From these two figures, we can see that except LS-SVM, CSTL exhibited shorter running time than the other four transfer learning methods.




(a) (b) (c)

(d) (e) (f)

Fig. 9 Classification accuracy of five methods on text datasets.

Table 3. Classification accuracy of five methods on text datasets with different umber of labeled samples

| Methods \ No. of labeled samples | 50 | 100 | 150 | 200 | 250 | 300 |
|---|---|---|---|---|---|---|
| **LSSVM** | 0.8318 | 0.8898 | 0.9182 | 0.9319 | 0.9417 | 0.9469 |
| **CDSVM** | 0.8591 | 0.9038 | 0.9242 | 0.9344 | 0.9421 | 0.9469 |
| **ASVM** | 0.8612 | 0.9058 | 0.9286 | 0.9388 | 0.9463 | 0.9519 |
| **STM** | 0.8595 | 0.9096 | 0.9278 | 0.9399 | 0.9472 | 0.9531 |
| **CSTL$_2$** | **0.8765** | **0.9158** | **0.9383** | **0.9481** | **0.9552** | **0.9602** |

*4.3.3. Cross-Domain Text Classification*

Boosting for transfer learning is an extension of the Adaboost learning framework. It is based on a learning mechanism which combines the source and target samples and then iteratively decreases the weights of the source data in order to reduce the impact on the learning process. According to [2], the number of iterations required is at least 50 for text datasets, which is computationally expensive and takes 14,347 seconds to complete for Task 1 in Table 2 when the number of iterations is one. Hence, we will not include TrAdaBoost method in the experiment.

The classification results on text datasets with different number of labeled samples are shown in Fig. 9. Table 3 shows the average classification accuracy for the 6 tasks in Table 2. From these results, we have the following observations.

(1) In most cases, CSTL outperforms other methods in terms of the classification accuracy. This promising result is mainly attributed to consideration of only selected knowledge for transfer learning based on CCR.

(2) It is found that CDSVM achieved better classification accuracy than LS-SVM with scarce labeled target data, and this advantage quickly diminished with the increasing labeled target samples. A major limitation of CDSVM is that it regards the data in the source domain exhibits a homogenous distribution. According to Fig. 9 and Table 3, ASVM and STM could achieve better results than CDSVM in most cases, which is attributed to the selective strategy. The classification performance of CSTL was more outstanding due to the ability to accurately select useful source objects as well as the assignment of different weights for the correlated objects to facilitate the construction of the target classifier.

## 5. Conclusions

In this paper, we have proposed the general framework CSTL with CCR to improve the performance of transfer learning. CSTL is effective in achieving the transfer classification learning with the capability of selecting correlated objects with weights from the source domain through the fast leave-one-out cross-validation strategy. An important advantage of CSTL is that it can alleviate the shortcomings in most existing transfer learning methods which exploit the entire source domain. The results of our experiments demonstrated the

effectiveness of CSTL for classification problems with respect to other existing transfer learning methods. Moreover, the weights assigned to different selected objects from the source domain were proved to be advantageous.

While classification performance of CSTL proposed in this paper is encouraging, some issues remain to be studied. For example, the classification errors of the source domain in the LS-SVM framework must be given in advance, which may lead to inaccurate distribution of the source data when the number of labeled training data is scarce. This issue and the related ones will be investigated in our future work.

**Acknowledgements**

This work was supported by the Research Grants Council of Hong Kong SAR (PolyU 52040/16E), the Hong Kong Polytechnic University (G-UA68), the National Natural Science Foundation of China (61300151, 2015NSFC), and the Natural Science Foundation of Jiangsu Province (BK20130155, BK20130161).

**Appendix**

In the following, we show theoretically that minimizing the disagreement between the source and target classifiers $f_S$ and $f_T$, denoted by the probability $P(f_S \neq f_T)$ and measured by the classification-error-based diversity penalization term can improve the performance of the target classifiers. We cast the classifiers $f_S$ and $f_T$ in the LS-SVM framework. Let $y_S, y_T \in \{-1,1\}$ be the labels from the source and target domain respectively. First, the three definitions are given.

***Definition A.1***. (*Nontrivial Target Classifier*) If the target classifier $f_T$ satisfies

$$P(f_T = l \mid f_T = l) > 1/2 \quad \text{or} \quad P(f_T \neq l \mid f_T = l) \leq 1/2, \tag{A.1}$$

where $l \in \{-1,+1\}$. Thus, the classifier $f_T$ is a nontrivial classifier for the target domain.

***Definition A.2***. (*Nonperfect Target Classifier*) If the target classifier $f_T$ returns a prediction accuracy of less than 100 percent on the target domain, the classifier $f_T$ is a nonperfect classifier for the target domain.

Similarly, the conditions defined in definitions A.1 and A.2 also hold for the source classifiers $f_S$.

***Definition A.3***. (*Conditional Independence of Source and Target Classifiers*) The conditional independence of source and target classifiers $f_S$ and $f_T$ is shown as

$$P(f_S = l \mid f_T = l', y_S = y) = P(f_S = l \mid y_S = y), \quad P(f_T = l \mid f_S = l', y_T = y) = P(f_T = l \mid y_T = y), \tag{A.2}$$

where $l, l', y \in \{-1,+1\}$.

According to Definitions A.1, A.2 and A.3, the following conclusion can be drawn: if the conditional independent assumption is satisfied, the misclassification errors of the nontrivial and nonperfect classifiers $f_T$ on the target domain have a strict upper bound, i.e., the disagreement $P(f_S \neq f_T)$. This will be shown theoretically as follows.

The classification error of $f_T$ on the target domain and the disagreement between $f_T$ and $f_S$ can be respectively represented as

$$\begin{aligned} P(f_T \neq y_T) &= P(f_T = 1, y_T = -1) + P(f_T = -1, y_T = 1) \\ &= P(f_T = 1, f_S = -1, y_T = -1) + P(f_T = 1, f_S = 1, y_T = -1) \\ &\quad + P(f_T = -1, f_S = -1, y_T = 1) + P(f_T = -1, f_S = 1, y_T = 1) \end{aligned}, \tag{A.3}$$

$$\begin{aligned} P(f_T \neq f_S) &= P(f_T = 1, f_S = -1) + P(f_T = -1, f_S = 1) \\ &= P(f_T = 1, f_S = -1, y_T = -1) + P(f_T = 1, f_S = -1, y_T = 1) \\ &\quad + P(f_T = -1, f_S = 1, y_T = -1) + P(f_T = -1, f_S = 1, y_T = 1) \end{aligned}, \tag{A.4}$$

To prove the inequality $P(f_T \neq y_T) < P(f_T \neq f_S)$, we should first validate the following inequality,

$$\begin{aligned} &P(f_T = 1, f_S = 1, y_T = -1) + P(f_T = -1, f_S = -1, y_T = 1) \\ &< P(f_T = 1, f_S = -1, y_T = 1) + P(f_T = -1, f_S = 1, y_T = -1) \end{aligned}. \tag{A.5}$$

According to the Bayes principle and Eq. (A.2), Eq. (A.5) can be formulated as

$$\begin{aligned} &P(f_T = 1 | y_T = -1)P(f_S = 1, y_T = -1) + P(f_T = -1 | y_T = 1)P(f_S = -1, y_T = 1) \\ &< P(f_T = 1 | y_T = 1)P(f_S = -1, y_T = 1) + P(f_T = -1 | y_T = -1)P(f_S = 1, y_T = -1) \end{aligned}. \tag{A.6}$$

By combining definitions A.1 and A.2, we have

$$\begin{aligned} &P(f_T = 1 | y_T = -1) < P(f_T = -1 | y_T = -1), \\ &P(f_T = -1 | y_T = 1) < P(f_T = 1 | y_T = 1), \\ &P(f_T = -1 | y_T = 1) > 0, \\ &P(f_T = 1, y_T = -1) > 0. \end{aligned} \tag{A.7}$$

Thus, Eq. (A.6) holds and we subsequently achieve

$$P(f_T \neq y_T) < P(f_T \neq f_S). \tag{A.8}$$

This shows that the disagreement $P(f_S \neq f_T)$ between the classifiers $f_S$ and $f_T$ in Eq. (1) upper bounded by the misclassification errors for the classifiers $f_T$. In other words, we should try to minimize the disagreement $P(f_S \neq f_T)$ between the two domains in order to reduce the classification errors and improve the learning performance of the target domain. Since the LS-SVM framework is adopted in both the source and target domains, this goal can be achieved by: 1) controlling the disagreement in $\mathbf{w}$ and $b$ between the two domains, which will result in the model-parameter-based consensus regularization terms that are commonly adopted by most transfer learning methods [6, 13-16]; 2) controlling the disagreement of classification errors between the two domains. In terms of the equation constraints (see Eq. (3)) used in LS-SVM, the control of the disagreement in $\mathbf{w}$ and $b$ between the source and target domains actually equivalent to the control the disagreement of classification errors between the two domains. Obviously, Eq. (4) can improve the performance of the target classifier.

**References**


[1] S. J. Pan and Q. Yang, “A survey on transfer learning,” IEEE Trans. Knowl. Data Eng., vol. 22, no. 10, pp. 1345–1359, 2010.

[2] W. Dai, Q. Yang, G. Xue, et al, “Boosting for transfer learning,” Proc. 24th Int. Conf. Mach. Learning, pp.193-200, Jun.2007.

[3] F. Z. Zhuang, P. Luo, Z. Y. Shen, et al, “Mining distinction and commonality across multiple domains using generative model for text classification,” IEEE Trans. Knowl. Data Eng., vol. 24, no.11, pp.2025-2039, 2012.

[4] F. Z. Zhuang, P. Luo, C.Y. Du, et al, “Triplex Transfer Learning: Exploiting Both Shared and Distinct Concepts for Text Classification,” IEEE Trans. Cybern., vol. 44, no.7, pp. 1191 – 1203, 2014.

[5] S. J. Pan, X. Ni, J.-T. Sun, et al, “Cross-domain sentiment classification via spectral feature alignment,” Proc. 19th Int. Conf. World Wide Web, pp.751-760, 2010.

[6] Y. Zhu, Y. Chen, Z. Lu, et al, “Heterogeneous transfer learning for image classification,” Proc. 25th AAAI Conf. Artif. Intell., 2011.

[7] L. Li, K. Zhou, G.-R. Xue, et al, “Video summarization via transferrable structured learning,” Proc. 20th Int. Conf. World Wide Web, pp.287-296, 2011.

[8] B. Li, Q. Yang, and X. Xue, “Transfer learning for collaborative filtering via a rating-matrix generative model,” Proc. 26th Int. Conf. Mach. Learning, vol.31, pp.617-624, 2009.

[9] J. Suykens, T. V. Gestel, J. D. Brabanter, et al, “Least Squares Support Vector Machines,” Singapore: World Scientific, vol. 4, 2002.

[10] E. Bart and S. Ullman, “Cross-generalization: Learning novel classes from a single example by feature replacement,” in Proc. IEEE computer Society Conf. Computer Vision and Pattern Recognition, vol.1, pp.672-679, 2005.

[11] Z.H. Deng, Fu-Lai Chung and Shitong Wang, “Knowledge-Leverage-Based TSK Fuzzy System Modeling,” IEEE Trans. Neural Net. And Learning Syst., vol. 24, no. 8, pp. 1200–1212, 2013.

[12] J. Tao, S. Wen, W. Hu, “L1-norm locally linear representation regularization multi-source adaptation learning,” Neural Networks, vol.69, pp.80-98, 2015.

[13] W. Bian, D. C. Tao, Y. Rui, “Cross-Domain Human Action Recognition”, IEEE Trans. Syst., Man, Cybern, Part B: Cybern., vol. 42, no.2, pp. 298-307, 2012.

[14] S. Chun-Wei, I.W. Tsang, and O. Yew-Soon, “Transfer Ordinal Label Learning,” IEEE Trans. Neural Net. And Learning Syst., vol. 24, no. 11, pp. 1863–1876, 2013.

[15] M. Stark, M. Goesele, and B. Schiele, “A shape-based object class model for knowledge transfer,” Proc. Int. Conf. Comput. Vision, pp.373-380, 2009.

[16] J. Yang, R. Yan, and A. G. Hauptmann, “Adapting SVM classifiers to data with shifted distributions,” Proc. 7th IEEE Int. Conf. Data Mining Workshops, pp.69-76, 2007.

[17] Y. Aytar and A. Zisserman, “Tabula rasa: Model transfer for object category detection,”

Proc. Int. Conf. Computer Vision, pp. 2252-2259, 2011.
[18] W. Jiang, E. Zavesky, S. F. Chang, et al, “Cross-domain learning methods for high-level visual concept classification,” Proc. 15th IEEE Int. Conf. Image Processing, pp.161-164, 2008.
[19] Y. Zhang and D. Yeung, “Transfer metric learning by learning task relationships,” Proc. ACM SIGKDD Int. Conf. Knowledge discovery data mining, pp. 1199-1208, 2010.
[20] W. Y. Dai, O. Jin, G. R. Xue, et al, “Eigen Transfer: a unified framework for transfer learning,” Proc. Int. Conf. Mach. Learning, pp.193-200, 2009.
[21] J. R. Anderson, Cognitive Psychology and Its Applications (seventh edition). New York: Freeman, 2010.
[22] W.S. Chu, F.D.L. Torre, J.F. Cohn, “Selective transfer machine for personalized facial action unit detection,” Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on., pp. 3515-3522, 2013.
[23] A. Gretton, A. Smola, J. Huang, et al, “Covariate shift by kernel mean matching,” Dataset shift in machine learning, vol.3, no.4, pp.5, 2009.
[24] E. Parzen, "On Estimation of a Probability Density Function and Mode," Annals of Math. Statistics, vol. 33, pp. 1065-1076, Sept. 1962.
[25] Z. H. Deng, Fu-Lai Chung, Shitong Wang, “FRSDE: Fast reduced set density estimator using minimal enclosing ball approximation,” Pattern Recognition, vo. 41, no. 4, pp.1363-1372, 2008.
[26] G. C. Cawley, “Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs,” Proc. Int. Joint Conf. Neural Network., pp. 1661–1668, 2006.
[27] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, “Efficient projections onto the l1-ball for learning in high dimensions,” in International Conference on Machine Learning (ICML), 2008.
[28] D. Hush, P. Kelly and C. Scovel, et al, “QP algorithms with guaranteed accuracy and run time for support vector machines,” Journal of Machine Learning Research, vol.7, pp.733-769, May 2006.
[29] L. Bruzzone and M. Marconcini, “Domain Adaptation Problems: A DASVM Classification Technique and a Circular Validation Strategy,” IEEE Trans. Pattern Anal. Mach. Intell., vol.32, no.5, pp.770-787, 2010.
[30] P. Gehler and S. Nowozin, “On feature combination for multiclass object classification,” Proc. Int. Conf. Computer Vision, pp.221-228, 2009.
[31] A. Bosch, A. Zisserman, and X. Munoz, “Representing shape with a spatial pyramid kernel,” Proc. 6th ACM Int. Conf. Image and video retrieval, pp.401-408, 2007.
[32] M. S. Long, J. M. Wang, G. G. Ding, et al, “Transfer Learning with Graph Co-Regularization,” IEEE Trans. Knowl. Data Eng., vol.26, no.7, pp.1805-1818, 2014.
[33] J. W. Tao, K. F. L. Chung, and S. T. Wang, “On minimum distribution discrepancy support vector machine for domain adaptation,” Pattern Recognition, vol.45, no.11, pp. 3962-3984, 2012.

[34] S. Bickel, “ECML-PKDD Discovery Challenge 2006 Overview,” In Proc. ECML/PKDD Discovery Challenge Workshop, pp.1-9, 2008.
[35] M.Sugiyama, M.Kawanabe, “Machine learning in non-stationary environments,” The MIT Press, 2012.