Heterogeneous Domain Adaptation via Nonlinear Matrix Factorization

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, *Member, IEEE*, and Alex C. Kot, *Fellow, IEEE*

***Abstract*— Heterogeneous domain adaptation (HDA) aims to solve the learning problems where the source- and the target-domain data are represented by heterogeneous types of features. The existing HDA approaches based on matrix completion or matrix factorization have proven to be effective to capture shareable information between heterogeneous domains. However, there are two limitations in the existing methods. First, a large number of corresponding data instances between the source domain and the target domain are required to bridge the gap between different domains for performing matrix completion. These corresponding data instances may be difficult to collect in real-world applications due to the limited size of data in the target domain. Second, most existing methods can only capture linear correlations between features and data instances while performing matrix completion for HDA. In this paper, we address these two issues by proposing a new matrix-factorization-based HDA method in a semisupervised manner, where only a few labeled data are required in the target domain without requiring any corresponding data instances between domains. Such labeled data are more practical to obtain compared with cross-domain corresponding data instances. Our proposed algorithm is based on matrix factorization in an approximated reproducing kernel Hilbert space (RKHS), where nonlinear correlations between features and data instances can be exploited to learn heterogeneous features for both the source and the target domains. Extensive experiments are conducted on cross-domain text classification and object recognition, and experimental results demonstrate the superiority of our proposed method compared with the state-of-the-art HDA approaches.**

***Index Terms*— Heterogeneous domain adaptation (HDA), matrix factorization, reproducing kernel Hilbert space (RKHS).**

I. Introduction

In the big data era, the data are easy to collect while labels are still expensive to annotate. Tremendous efforts have been devoted to make the best use of the available out-of-domain labeled data to solve learning problems on a domain of interest. However, due to the difference between domains, a predictive model learned from labeled data on some source domain(s) with standard supervised learning techniques fails to generalize well to the target domain. Domain adaptation or transfer learning has been proposed to address the aforementioned cross-domain learning problems where training data and test data are from different domains [1]. Recently, domain adaptation techniques have been widely applied to many application scenarios, such as indoor WiFi localization [2], sentiment analysis [3], [4], object recognition [5], [6], and so on.

In the literature, many existing domain adaptation approaches focus on cross-domain learning problems of homogeneous features, which are referred to as homogeneous domain adaptation problems [2], [5], [6]. However, there exist many other applications where the source domain and the target domain data are characterized by different sets of features, which are referred to as heterogeneous domain adaptation (HDA) problems. For example, in natural language processing, one may have a lot of linguistic resources and sufficient annotated corpus for a majority language, e.g., English, while only have limited linguistic resources and insufficient annotated corpus for some minority language, e.g., Southeast Asia languages. In this case, it is highly desired if knowledge extracted from the learning tasks of the majority language can be transferred to help the learning tasks for the minority language. In this context, as each data instance, e.g., a document, is represented by heterogeneous features across different languages, HDA techniques are crucial. As another example, in some computer vision problems, one may extract powerful deep features with a well-trained deep learning network in a domain where sufficient labeled data are available for training. However, in some other domains, the training data may be protected by a privacy policy (e.g., EU data protection rule [7]). As a result, one cannot employ deep learning but use handcrafted features to represent the data. In this case, HDA is useful to transfer knowledge from "deep" features to "shallow" features. In some other applications, one domain can be represented by text (e.g., food recipe) and the other can be represented by images (e.g., photographs of food) [8], which also lead to HDA problems.

Generally speaking, HDA can be categorized into two directions. The first direction assumes that there are some corresponding instances or features between the source domain and the target domain as a bridge and thus aims to learn a common subspace by leveraging such correspondences [9]–[11]. The second direction assumes that there are a few labeled data available in the target domain, and aims to learn a common subspace by utilizing the target domain and the source domain labels to bridge a connection between the two domains [12]–[14].

Manuscript received February 21, 2018; revised December 10, 2018; accepted April 19, 2019. This work was supported by the National Research Foundation (NRF)-NSFC, Prime Minister's Office, Singapore under Grant NRF2016NRF-NSFC001-098. The work of S. J. Pan was supported by NTU Singapore Nanyang Assistant Professorship (NAP) under Grant M4081532.020. *(Corresponding author: Haoliang Li.)*

H. Li and A. C. Kot are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: hli016@e.ntu.edu.sg).

S. J. Pan is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798.

S. Wang is with the Department of Computer Science, College of Science and Engineering, Hong Kong.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2019.2913723

Recently, matrix completion-based methods have been proposed for HDA problems [10], [11]. The key idea is that instead of learning, an (or a pair of) explicit feature mapping(s) between feature spaces of the source domain and the target domain, one can directly reconstruct cross-domain heterogeneous features for each instance through matrix completion techniques. For instance, Zhou *et al.* [11] proposed a method named Distribution Matching-based Matrix Completion (DMMC) to encode the distance in distributions between domains into feature reconstruction in order to reduce the distance between domains. However, the existing matrix-completion-based methods have two major problems. First, they require plenty of corresponding data instances between the source domain and the target domain when performing matrix completion, which limits their applications in real-world problems since the corresponding data instances are difficult to collect in some cases. In addition, to make optimization tractable, DMMC adopts the maximum mean discrepancy (MMD) metric [15] with a linear kernel to measure the distance between distributions, rather than a characteristic kernel [16]. As a result, the distance measure between distributions may not be precise.

To address the aforementioned issues, in this paper, we propose a nonlinear matrix factorization method inspired by kernel methods [17] for HDA. To be specific, a new semisupervised distribution-regularized matrix factorization method is proposed to learn a dictionary and new representations for instances by exploiting the correlations between instances and features. Subsequently, the data instances from both the source domain and the target domain are mapped to an approximated reproduced kernel Hilbert space via Random Kitchen Sinks [18] to construct an augmented instance-feature matrix. As all instances are in the RKHS, MMD can be used to provide more accurate distance measure between distributions, which makes knowledge transfer more effective. We conduct extensive experiments on several benchmark data sets on cross-domain text classification and object recognition to verify the effectiveness of our proposed method.

The contributions of our work are summarized as follows.

1) To the best of our knowledge, this is the first work for HDA using nonlinear matrix factorization. Our idea on nonlinear matrix factorization borrows from kernel methods which first map data to a high-dimensional space using an implicit nonlinear feature map induced by a kernel, and then perform linear model on the mapped data. However, as there are a lot of missing values in the augmented matrix, directly conducting matrix factorization based on a characteristic kernel is intractable, and we propose to use the random kitchen sinks technique to construct to an explicit nonlinear feature map that approximates the feature map induced by the radial basis function (RBF) kernel.
2) We propose an effective optimization algorithm to jointly optimize variables in a unified framework.
3) Experimental results on cross-domain object recognition and text classification demonstrate that our proposed method achieves significant improvement in terms of classification accuracy compared with other HDA methods.

The rest of this paper is organized as follows. In Section II, we first provide a brief review on related works regarding HDA. In Section III, we state our problem and introduce some preliminaries which are used in our proposed method. After that, we present our proposed HDA algorithm in the linear matrix factorization manner and further extend our formulation to the nonlinear matrix factorization manner in Section V. We conduct extensive experiments on the tasks of object recognition and text classification in Section VI, and conclude this paper in Section VII.

II. Related Works

Domain adaptation [1] aims to transfer knowledge learned from a domain of rich labeled data to a new target domain of low annotation resource. Traditional domain adaptation methods focus on the problem where the source domain and the target domain are represented by the same type of features or data. For instance, Huang *et al.* [19] and Li *et al.* [20] proposed an instance-weighting method to minimize the distribution discrepancy between the source and target domains. Pan *et al.* [2] proposed the transfer component analysis (TCA) algorithm that aims to project different domain data into a latent space to minimize the distance between distributions. Long *et al.* [21] combined the above-mentioned ideas by proposing transfer joint matching (TJM) by learning a new space where the distribution difference is mingled and reweighting the source domain data that are irrelevant to the target recognition task. Mehrkanoon and Suykens [22] extended the Kernel canonical correlation analysis algorithm to solve HDA problems. Recently, deep learning based domain adaptation methods are also developed and proven to be effective to solve the cross-domain learning problems. For instance, Deep Adaptation Network [23], [24] encodes the MMD criteria into multiple layers of AlexNet [25], where the network parameters as well as the parameters of the RBF kernel are jointly optimized to obtain a suitable distance measurement in the reproduced kernel Hilbert space. Long *et al.* [26] further showed that by introducing addition residual block, it could further improve the performance by learning more domain-invariant feature representation through deep learning. Besides using MMD, adversarial training techniques can also benefit cross-domain learning task [27]–[30], since it has been shown that performing adversarial training to confuse domains is equivalent to minimizing the Jesson–Shannon divergence between two distributions [31].

However, the methods mentioned above cannot be directly applied to HDA problems, as they all assumed that the source and target domains data are represented by the same types of features. HDA aims to tackle the problem where the source and the target domains information are represented by different types of features. Generally, the existing HDA approaches can be categorized into two groups. The first group requires a few labeled data and some unlabeled in the target domain for training, which is referred to as semisupervised HDA. Previous works [32]–[34] focused on aligning heterogenous features in a common latent space for knowledge transfer. Kulis *et al.* [13] proposed to learn a metric for instances from heterogeneous domains. Choo *et al.* [35]

proposed a graph embedding strategy for heterogeneous space alignment. Duan *et al.* [36] proposed the heterogenous feature augmentation (HFA) method to augment homogeneous common features learned by a support vector machine (SVM)-style approach with heterogeneous features. Shi *et al.* [37] developed a spectral embedding approach to learn common feature space between the heterogeneous domains. Zhou *et al.* [14] and Xiao and Guo [38] proposed semisupervised HDA methods for multiclass classification problems by exploiting the error-correcting output coding scheme, respectively. Recently, Chen *et al.* [39] proposed a neural network based transfer learning approach for cross-domain feature adaptation. Tsai *et al.* [40] proposed a landmark selection strategy to align MMD and conditional MMD based on label information. Yan *et al.* [41] proposed to learn a discriminative correlation subspace for HDA, which however, requires the number of labeled source and target domain samples to be the same. More recently, Li *et al.* [42] proposed to learn a suitable MMD with adversarial learning for HDA task.

Another group of HDA approaches does not require labeled data in the target domain but a set of unlabeled correspondences between the source domain and the target domain for training. The existing matrix completion-based approaches [10], [11] fall into this group, which will be briefly reviewed as preliminary in Section III-A. Different from matrix completion based approaches, Dai *et al.* [9] and Prettenhofer and Stein [43] proposed to learn a feature mapping between the heterogeneous features using feature-level correspondences, e.g., word-level translations. It is also worth noting that in our problem setting, though we propose a matrix factorization based method for HDA, we do not require any correspondences between domains. Instead, a few labeled data in the target domain need to be provided in advance for training, which is different from the existing matrix completion-based approaches.

More prior information/assumption can be leveraged to make HDA problems more trackable. Zhuang *et al.* [44] proposed to use multiple domains information based on Probabilistic Latent Semantic Analysis to align text distributions caused by different index words. Yang *et al.* [45] proposed to learn the transferred weights with the aid of co-occurrence data which contain the same set of instances but in different feature spaces. Luo *et al.* [46] proposed to leverage the data from multiple domains to learn high-order statistics in a multitask metric learning manner.

III. Problem Statement and Preliminary

A. Problem Statement

In this paper, we focus on semisupervised HDA problems, where, besides plenty of source-domain labeled data, there are a few labeled data and some unlabeled data in the target domain for training. We denote by $\mathbf{X}_S = [\mathbf{x}_{S_1}^\top, \ldots, \mathbf{x}_{S_{N_S}}^\top]^\top$ the source-domain input matrix with each row being an instance $\mathbf{x}_{S_i} \in \mathbb{R}^{1\times d_S}$, and by $\mathbf{X}_T = [\mathbf{x}_{T_1}^\top, \ldots, \mathbf{x}_{T_{N_T}}^\top]^\top$ the target-domain input matrix with $\mathbf{x}_{T_i} \in \mathbb{R}^{1\times d_T}$. In HDA, $\mathbf{x}_{S_i}$ and $\mathbf{x}_{T_i}$ are represented by heterogeneous features, and thus, in general, $d_S \neq d_T$. Suppose the first N_{T_l} instances in $\mathbf{X}_T$ are labeled, and the rest $N_{T_u} = N_T - N_{T_l}$ are unlabeled. We assume that the two domains share the same set of class labels, which are represented using the one-hot encoding scheme, i.e., $\mathbf{Y}_S \in \{0,1\}^{N_S\times m}$, $\mathbf{Y}_{T_l} \in \{0,1\}^{N_{T_l}\times m}$, and $\mathbf{Y}_{T_u} = \mathbf{0} \in \mathbb{R}^{N_{T_u}\times m}$, where m is the number of classes.

B. Heterogeneous Feature Augmentation

Feature augmentation was introduced in [47] for transfer learning by augmenting the original feature space $\mathbb{R}^d$ to $\mathbb{R}^{3d}$, where the source domain feature $\mathbf{x}_S$ is augmented as $[\mathbf{x}_S, \mathbf{x}_S, \mathbf{0}]$ and the target domain feature $\mathbf{x}_T$ is augmented as $[\mathbf{x}_T, \mathbf{0}, \mathbf{x}_T]$. Here, $\mathbf{0}$ denotes the vector of all zeros with dimension d. As such, the connection between the source domain and the target domain can be established.

Duan *et al.* [36] further extended the idea of feature augmentation to HDA by introducing a common subspace for the source domain and the target domain. In particular, two projection matrices $\mathbf{P}$ and $\mathbf{Q}$ are introduced for the source domain and the target domain, respectively. The common space is then formulated in a feature augmentation manner as $[\mathbf{P}\mathbf{x}_S, \mathbf{x}_S, \mathbf{0}_T]$ and $[\mathbf{Q}\mathbf{x}_T, \mathbf{0}_S, \mathbf{x}_T]$, where $\mathbf{0}_S$ and $\mathbf{0}_T$ denote the vectors of all zeros with the same dimension as $\mathbf{x}_S$ and $\mathbf{x}_T$, respectively. An HDA problem is solved by jointly optimizing $\mathbf{P}$, $\mathbf{Q}$, as well as the parameters of classifier (e.g., SVM). A follow-up semisupervised method was proposed by Li *et al.* [48], where unlabeled target domain data are utilized during the training process. Our proposed method also leverages the advantage of HFA, which has proven to be effective for HDA.

C. Maximum Mean Discrepancy

In our work, we resort to MMD [49], which is a nonparametric criteria to measure the distance between distributions, and has been adopted in various domain adaptation approaches [2], [5], [19], [21], [50]–[53]. The formulation of MMD can be expressed as follows:

$$\mathrm{Dist}(\mathbf{X}_S, \mathbf{X}_T) = \left\| \frac{1}{N_S}\sum_{i=1}^{N_S}\phi(\mathbf{x}_{S_i}) - \frac{1}{N_T}\sum_{j=1}^{N_T}\phi(\mathbf{x}_{T_j}) \right\|_{\mathcal{H}}^2 \quad (1)$$

where $\phi(\cdot)$ denotes a feature mapping induced by a kernel. The MMD distance in (1) can be computed using the kernel trick [49] as follows:

$$\mathrm{Dist}(\mathbf{X}_S, \mathbf{X}_T) = \mathrm{tr}(\mathbf{K}\mathbf{L}) \quad (2)$$

where $\mathbf{K}$ with $\mathbf{K}_{ij} = \langle\phi(\mathbf{x}_i), \phi(\mathbf{x}_j)\rangle$ is the kernel matrix computed on the source-domain data and the target-domain data, which can be written as the following blockwise matrix:

$$\mathbf{K} = \begin{pmatrix} \mathbf{K}_{S,S} & \mathbf{K}_{S,T} \\ \mathbf{K}_{T,S} & \mathbf{K}_{T,T} \end{pmatrix}$$

and $\mathbf{L}_{ij} = \frac{1}{N_S^2}$ if $1 \le i, j \le N_S$, $\mathbf{L}_{ij} = \frac{1}{N_T^2}$ if $N_S + 1 \le i, j \le N_S + N_T$, and otherwise $\mathbf{L}_{ij} = -\frac{1}{N_S N_T}$.

IV. HDA via Matrix Completion

A. Distribution Matching Based on Matrix Completion

Before introducing our proposed method, we briefly revisit the DMMC [11] method for HDA. As presented in Section III-B, HFA introduces zero-padding for missing features. Thus, an intuitive idea is to recover such missing features using matrix factorization or matrix completion for more effective knowledge transfer. Given the source-domain input matrix $\mathbf{X}_S$ and the target-domain input matrix $\mathbf{X}_T$, one can first augment the data by simply padding zeros, which are considered as missing values, to make the dimensions of the data from the two domains identical

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_S & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_T \end{pmatrix} \in \mathbb{R}^{(N_S+N_T)\times(d_S+d_T)}. \tag{3}$$

The goal of matrix completion-based methods is to reconstruct the missing entries in $\mathbf{X}$ via solving

$$\min_{\hat{\mathbf{X}}} \left\| \mathbf{P} \circ (\mathbf{X} - \hat{\mathbf{X}}) \right\|_F^2 + \lambda \Omega(\hat{\mathbf{X}}) \tag{4}$$

where $\hat{\mathbf{X}} \in \mathbb{R}^{(N_S+N_T)\times(d_S+d_T)}$ is the recovered augmented matrix, where each row is an instance with the original features and the learned augmented features, $\mathbf{P}$ is an indicator matrix with $\mathbf{P}_{ij} = 1$ if $\mathbf{X}_{ij}$ is observed, otherwise 0, the operator $\circ$ is the Hadamard product, $\Omega(\hat{\mathbf{X}})$ is a regularization term on $\hat{\mathbf{X}}$, and $\gamma > 0$ is a tradeoff parameter. A commonly used regularization term on is the trace norm $\|\cdot\|_*$, which constrains the rank of $\hat{\mathbf{X}}$.

To ensure the difference between different domains data with the recovered features to be reduced, Zhou *et al.* [11] proposed to encode distribution matching as a constraint in (4), which is casted as the following optimization problem:

$$\begin{aligned} &\min_{\hat{\mathbf{X}}} \ \|\mathbf{P} \circ (\mathbf{X} - \hat{\mathbf{X}})\|_F^2 + \lambda(\|\hat{\mathbf{X}}\|_* + \|\hat{\mathbf{X}}\|_1) \\ &\text{s.t. } \mathcal{K} = \{\hat{\mathbf{X}} | \text{tr}(\mathbf{K}_{\hat{\mathbf{X}}}\mathbf{L}) < \gamma\} \end{aligned} \tag{5}$$

where $\text{tr}(\mathbf{K}_{\hat{\mathbf{X}}}\mathbf{L})$ is the MMD distance based on the recovered augmented matrix $\hat{\mathbf{X}}$. To make the optimization problem in (5) computationally tractable, in [11], a linear kernel is used to compute $\mathbf{K}_{\hat{\mathbf{X}}}$, i.e., $\mathbf{K}_{\hat{\mathbf{X}}} = \hat{\mathbf{X}}\hat{\mathbf{X}}^\top$.

B. Reformulation of Matrix Completion for HDA

By considering that data usually lie on a low-dimensional manifold, we first reformulate (5) by introducing two latent factor matrices $\mathbf{U} \in \mathbb{R}^{(N_S+N_T)\times k}$ and $\mathbf{V} \in \mathbb{R}^{(d_S+d_T)\times k}$

$$\begin{aligned} \min_{\mathbf{U},\mathbf{V}} \ &\|\mathbf{P} \circ (\mathbf{X} - \mathbf{U}\mathbf{V}^\top)\|_F^2 + \mu \ \text{tr}(\mathbf{K}_{\mathbf{U}}\mathbf{L}) \\ &+\lambda\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \end{aligned} \tag{6}$$

where $\mathbf{U}$ is considered as the latent representations for $\mathbf{X}$ and $\mathbf{V}$ is the dictionary. As $\mathbf{U}$ is a representation for $\mathbf{X}$, the kernel matrix $\mathbf{K}_{\mathbf{U}} = \mathbf{U}\mathbf{U}^\top$ is defined on $\mathbf{U}$ instead of $\hat{\mathbf{X}} = \mathbf{U}\mathbf{V}^\top$. In addition, as shown in [54], $\frac{1}{2}(\|\mathbf{U}\|_F + \|\mathbf{V}\|_F)$ is an upper bound of $\|\hat{\mathbf{X}}\|_*$. Therefore, we replace the trace norm by $\|\mathbf{U}\|_F + \|\mathbf{V}\|_F$. Here, it is worth mentioning that we drop the term of L_1-norm in (6) because of the following two considerations: 1) the sparse constraint may not be crucial in some application scenarios and 2) introducing the sparsity constraint makes the optimization procedure much more difficult than necessary.

C. Semisupervised Manifold Alignment

As the values of the antidiagonal blocks of the augmented matrix $\mathbf{X}$ in (3) are all missing, there are no explicit connections between the source domain and the target domain, which may affect the effectiveness of matrix factorization or matrix completion. The existing matrix completion-based approaches [10], [11] make use of some cross-domain instance correspondences as a bridge between domains. However, the cross-domain instance correspondences are difficult to obtain in practice. Instead, we, therefore, assume that only a few target-domain labeled data are given. To make matrix factorization on $\mathbf{X}$ more effective, the target-domain labeled data are utilized. We then propose a semisupervised HDA algorithm by borrowing the idea from [55], aiming to minimize the following objective to encode label information into the latent representation $\mathbf{U}$ of $\mathbf{X}$:

$$\eta \|\mathbf{Q}(\mathbf{U}\mathbf{W} - \mathbf{Y})\|_F^2 + \beta\|\mathbf{W}\|_F^2 + \gamma \ \text{tr}(\mathbf{U}^\top \mathbf{L}_g \mathbf{U}) \tag{7}$$

where $\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_S^\top & \mathbf{Y}_{T_l}^\top & \mathbf{Y}_{T_u}^\top \end{bmatrix}^\top$ is the matrix of labels, $\mathbf{Q}$ is an indicator matrix defined as follows:

$$\mathbf{Q} = \begin{pmatrix} \mathbf{I}_{N_S \times N_S} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \dfrac{N_S}{N_{T_l}} \mathbf{I}_{N_{T_l} \times N_{T_l}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0}_{N_{T_u} \times N_{T_u}} \end{pmatrix}$$

where the zero values indicate unlabeled instances and the nonzero values indicate labeled instances. The weight matrix $\mathbf{W} \in \mathbb{R}^{k\times m}$ can be considered as a linear classifier to map inputs to the corresponding labels, and $\mathbf{L}_g$ is the Laplacian matrix [56] computed using the original input matrix $\mathbf{X}$, defined as $\mathbf{L}_g = \begin{pmatrix} \mathbf{L}_{gS} & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_{gT} \end{pmatrix}$ where $\mathbf{L}_{gS}$ and $\mathbf{L}_{gT}$ are the Laplacian matrices on the source-domain data $\mathbf{X}_S$ and the target-domain data $\mathbf{X}_T$, respectively. In this work, we adopt 0/1 weighting strategy with five nearest neighbors to construct the Laplacian matrices.

The first term in the objective (7) aims to learn a linear classifier in terms of $\mathbf{W}$ to minimize the error on labeled data of both domains. The second term is a regularization term on the complexity of $\mathbf{W}$, and the third term is a manifold regularization term to encode manifold structure for propagating label information from labeled data to unlabeled data. The parameters $\eta > 0$, $\beta > 0$, and $\gamma > 0$ are the tradeoff parameters.

D. Optimization Problem for Semisupervised HDA via Matrix Factorization

By combining (6) and (7), the objective $\mathcal{L}$ of our proposed HDA method can be written as follows:

$$\begin{aligned} \min_{\mathbf{U},\mathbf{V},\mathbf{W}} \ &\|\mathbf{P} \circ (\mathbf{X} - \mathbf{U}\mathbf{V}^\top)\|_F^2 + \lambda\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \\ &+\mu \text{tr}(\mathbf{K}_{\mathbf{U}}\mathbf{L}) + \eta\|\mathbf{Q}(\mathbf{U}\mathbf{W} - \mathbf{Y})\|_F^2 \\ &+\beta\|\mathbf{W}\|_F^2 + \gamma \ \text{tr}(\mathbf{U}^\top \mathbf{L}_g \mathbf{U}). \end{aligned} \tag{8}$$

Algorithm 1 Semisupervised HDA Algorithm via Matrix Factorization

Input: An augmented input matrix $\mathbf{X}$ defined in (3), a label matrix $\mathbf{Y} = \left[\mathbf{Y}_S^\top \ \mathbf{Y}_{T_l}^\top \ \mathbf{Y}_{T_u}^\top\right]^\top$, and the maximum number of iterations, T

Initializations:

1: Initialize matrices $\mathbf{U} \in \mathbb{R}^{(N_S+N_T)\times k}$, $\mathbf{V}_\phi \in \mathbb{R}^{(d_S+d_T)\times k}$ and $\mathbf{W} \in \mathbb{R}^{k\times m}$

while $t \leq T$ **do**

1: update $\mathbf{U}$ based on (9)

2: update $\mathbf{V}$ based on (10)

3: update $\mathbf{W}$ based on (11)

4: $t = t + 1$

end while

Output : $\hat{\mathbf{X}}$ and $\mathbf{W}$

It can be proven that the objective (8) is convex in terms of one variable by fixing the other. Therefore, we develop a gradient-descent-based algorithm to alternatively optimize $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$. To be specific, the update rules regarding $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$ can be computed as follows, respectively, and the overall algorithm is presented in Algorithm 1.

By fixing $\mathbf{V}$ and $\mathbf{W}$, $\mathbf{U}$ is updated through

$$\begin{aligned}\mathbf{U}_{t+1} &= \mathbf{U}_t - \epsilon_t \frac{\partial \mathcal{L}}{\partial \mathbf{U}} \\ &= \mathbf{U}_t - \epsilon_t \big(2\big(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{X}\big)\circ\mathbf{P})\mathbf{V}_t \\ &\quad +2\lambda\mathbf{U}_t + \mu(\mathbf{L}^\top + \mathbf{L})\mathbf{U}_t + \gamma\left(\mathbf{L}_g^\top + \mathbf{L}_g\right)\mathbf{U}_t \\ &\quad +2\eta\,\mathbf{Q}^\top\mathbf{Q}(\mathbf{U}_t\mathbf{W}_t - \mathbf{Y})\mathbf{W}_t^\top\big). \end{aligned} \tag{9}$$

By fixing $\mathbf{U}$ and $\mathbf{W}$, $\mathbf{V}$ is updated through

$$\begin{aligned}\mathbf{V}_{t+1} &= \mathbf{V}_t - \epsilon_t \frac{\partial \mathcal{L}}{\partial \mathbf{V}} \\ &= \mathbf{V}_t - 2\epsilon_t\big(\big(\mathbf{V}_t\mathbf{U}_t^\top - \mathbf{X}^\top\big)\circ\mathbf{P}^\top)\mathbf{U}_t + \lambda\mathbf{V}_t\big). \end{aligned} \tag{10}$$

By fixing $\mathbf{U}$ and $\mathbf{V}$, $\mathbf{W}$ is updated through

$$\begin{aligned}\mathbf{W}_{t+1} &= \mathbf{W}_t - \epsilon_t \frac{\partial \mathcal{L}}{\partial \mathbf{W}} \\ &= \mathbf{W}_t - 2\epsilon_t\big(\eta\mathbf{U}_t^\top\mathbf{Q}^\top\mathbf{Q}(\mathbf{U}_t\mathbf{W}_t - \mathbf{Y}) + \beta\mathbf{W}_t\big). \end{aligned} \tag{11}$$

Here, ϵ_t is the step size at the iteration t, and $\mathbf{U}_t$, $\mathbf{V}_t$, and $\mathbf{W}_t$ are the values of $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$ at the iteration t, respectively.

V. Kernelization of the Proposed HDA Method

As discussed in Section I, there are two limitations of the linear matrix-factorization-based approach presented in the previous section: 1) the augmented heterogeneous features are learned via linear matrix factorization on the original input feature matrix $\mathbf{X}$, which fails to capture nonlinear correlations between instances and features and 2) as a linear kernel is used to compute the MMD distance between domains, it can only measure distance between simple distributions precisely. If the underlying distributions of the source domain data and the target domain data are complex, the distance measured by MMD with the linear kernel may be imprecise. In this section, we extend the linear matrix-factorization-based method to a kernelized version for HDA to overcome the above limitations.

A. Matrix Factorization in an RKHS

On top of (6), we describe our kernelized matrix factorization method in detail. Our key idea is to first map the instances of both the source domain and the target domain, i.e., the augmented input matrix $\mathbf{X}$, to an RKHS, and then perform factorization on the mapped input matrix to learn a common representation and a new dictionary. Specifically, suppose $\phi(\cdot) : \mathbb{R}^{d_S+d_T} \to \mathcal{H}$, where $\mathcal{H}$ is an RKHS with a kernel $k(\cdot,\cdot)$. The construction of $\phi(\cdot)$ will be presented in detail in the next section. We denote by $\Phi = \phi(\mathbf{X}) \in \mathbb{R}^{(N_S+N_T)\times p}$, where p can be $+\infty$, and each row is a mapped instance in $\mathcal{H}$. Our goal is to perform factorization on Φ by solving the following optimization problem:

$$\begin{aligned}\min_{\mathbf{U}_\phi,\mathbf{V}_\phi} &\left\|\mathbf{P}\circ\left(\Phi - \mathbf{U}_\phi\mathbf{V}_\phi^\top\right)\right\|_F^2 + \mu\,\mathrm{tr}(\mathbf{U}_\phi\mathbf{U}_\phi^\top\mathbf{L}) \\ &+\lambda\left(\|\mathbf{U}_\phi\|_F^2 + \|\mathbf{V}_\phi\|_F^2\right) \end{aligned} \tag{12}$$

where $\mathbf{V}_\phi \in \mathbb{R}^{p\times k}$ is the dictionary in the RKHS, and $\mathbf{U}_\phi \in \mathbb{R}^{(N_S+N_T)\times k}$ is the representation for Φ with the dictionary with k being the latent dimensionality. Thus, $\mathbf{U}_\phi\mathbf{U}_\phi^\top$ can be considered as a kernel matrix in the RKHS. The framework of the proposed method is shown in Fig. 1.

B. Explicit Feature Map

On the one hand, we cannot directly optimize (12) due to the infinite dimension of $\mathbf{V}_\phi$, which makes the computation of derivative of the objective function with respect to $\mathbf{V}_\phi$ intractable. On the other hand, it is also not feasible to apply kernel trick based on Φ to induce a kernel matrix $\mathbf{K} = \Phi\Phi^\top$ due to the missing value in Φ. This is because by using kernel trick in this way will lead to inaccurate computation of the matrix $\mathbf{K}$, and thus the learned representation $\mathbf{U}_\phi$ will not be discriminative anymore.

Therefore, we need to approximate Φ with an explicit nonlinear mapping which can still preserve the property of $\phi(\cdot)$ in RKHS. To construct such feature map $\phi(\cdot)$, we adopt the Random Kitchen Sinks method that estimates infinite dimensions of kernel representation by finite dimension vector [18]. Specifically, one can use the property of random Fourier feature to generate approximations of Gaussian RBF kernel, $k(\mathbf{x},\mathbf{x}') = \exp(-(\|\mathbf{x}-\mathbf{x}'\|^2/2\sigma^2))$, by randomly sampling $\mathbf{Z} \in \mathbb{R}^{d\times n}$, and construct the empirical feature map as $\phi(\mathbf{x}) = (1/\sqrt{n})\exp(i[\mathbf{xZ}])$. Here, $\mathbf{x}$ can be either a source-domain instance ($d = d_S$) or a target-domain instance ($d = d_T$), and n is the number of base functions. Since the kernel values in our problem are real, we replace the exponential part by the sinusoidal function as follows:

$$\phi(\mathbf{x}) = \frac{1}{\sqrt{n}}[\cos(\mathbf{xZ})\ \ \sin(\mathbf{xZ})]. \tag{13}$$

In this way, the matrix Φ can be written as

$$\Phi = \begin{pmatrix} \frac{\cos(\mathbf{X}_S\mathbf{Z}_S)}{\sqrt{n}} & \mathbf{0} & \frac{\sin(\mathbf{X}_S\mathbf{Z}_S)}{\sqrt{n}} & \mathbf{0} \\ \mathbf{0} & \frac{\cos(\mathbf{X}_T\mathbf{Z}_T)}{\sqrt{n}} & \mathbf{0} & \frac{\sin(\mathbf{X}_T\mathbf{Z}_T)}{\sqrt{n}} \end{pmatrix} \tag{14}$$

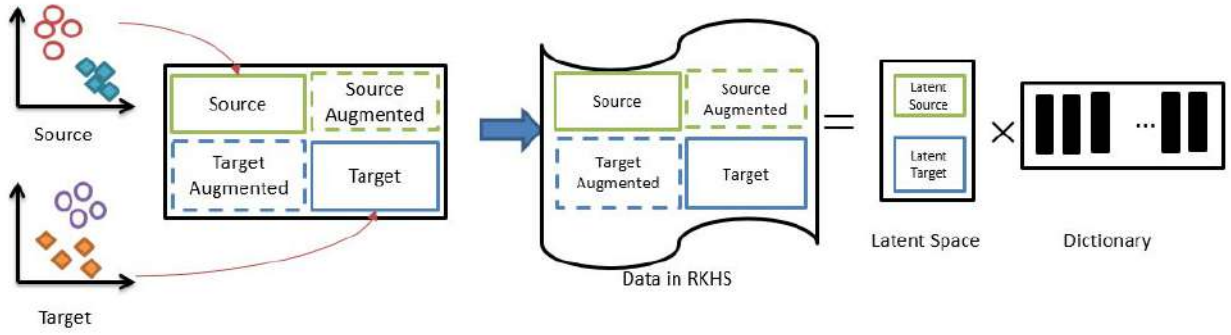


Fig. 1. Proposed algorithm for HDA. We first conduct HFA based on source and target domain data. Then, the augmented feature is mapped to RKHS by Random Kitchen Sinks method. Finally, we propose a novel semisupervised matrix factorization to learn the latent representation of augmented feature and the classifier.

where $\Phi \in \mathbb{R}^{(N_S+N_T)\times(2n+2n)}$, $\mathbf{0}$ denotes missing values, and $\mathbf{Z}_S \in \mathbb{R}^{d_S\times n}$, and $\mathbf{Z}_T \in \mathbb{R}^{d_T\times n}$ denote random matrices of the source domain and the target domain, respectively.

C. Proposed Optimization Problem

By adding the classifier and the regularization terms introduced in (7) to (12), the overall objective function of our proposed kernelized matrix-factorization-based method for HDA is formulated as follows:

$$\begin{aligned}\min_{\mathbf{U}_\phi,\mathbf{V}_\phi,\mathbf{W}} \mathcal{L} = &\left\|\mathbf{P}\circ\left(\Phi-\mathbf{U}_\phi\mathbf{V}_\phi^\top\right)\right\|_F^2+\lambda\|\mathbf{U}_\phi\|_F^2\\ &+\lambda\|\mathbf{V}_\phi\|_F^2+\beta\|\mathbf{W}\|_F^2+\mu\,\mathrm{tr}\left(\mathbf{U}_\phi\mathbf{U}_\phi^\top\mathbf{L}\right)\\ &+\gamma\,\mathrm{tr}\left(\mathbf{U}_\phi^\top\mathbf{L}_g\mathbf{U}_\phi\right)+\eta\|\mathbf{Q}(\mathbf{U}_\phi\mathbf{W}-\mathbf{Y})\|_F^2. \quad (15)\end{aligned}$$

We aim to learn three types of variables $\mathbf{V}_\phi$, $\mathbf{U}_\phi$, and $\mathbf{W}$ by solving the minimization problem in (15). Similar to (8), we propose to use alternating optimization techniques to solve the optimization problem iteratively.

Specifically, at each iteration t, by fixing $\mathbf{V}_\phi$ and $\mathbf{W}$, $\mathbf{U}_\phi$ is updated based on the following gradient descent rule:

$$\begin{aligned}&\mathbf{U}_{\phi_{t+1}}\\ &=\mathbf{U}_{\phi_t}-\epsilon_t\frac{\partial\mathcal{L}}{\partial\mathbf{U}_\phi}\\ &=\mathbf{U}_{\phi_t}-\epsilon_t\big(2\big(\mathbf{U}_{\phi_t}\mathbf{V}_{\phi_t}^\top-\Phi\big)\circ\mathbf{P}\big)\mathbf{V}_{\phi_t}\\ &\quad+2\lambda\mathbf{U}_{\phi_t}+\mu\left(\mathbf{L}^\top+\mathbf{L}\right)\mathbf{U}_{\phi_t}+\gamma\left(\mathbf{L}_g^\top+\mathbf{L}_g\right)\mathbf{U}_{\phi_t}\\ &\quad+2\eta\mathbf{Q}^\top\mathbf{Q}(\mathbf{U}_{\phi_t}\mathbf{W}_t-\mathbf{Y})\mathbf{W}_t^\top\big) \quad (16)\end{aligned}$$

where ϵ_t is the step size at iteration t, and $\mathbf{U}_{\phi_t}$ and $\mathbf{V}_{\phi_t}$ are the values of $\mathbf{U}_\phi$ and $\mathbf{V}_\phi$ at iteration t, respectively. When $\mathbf{U}_\phi$ and $\mathbf{W}$ are fixed, $\mathbf{V}_\phi$ is updated as follows:

$$\begin{aligned}\mathbf{V}_{\phi_{t+1}}&=\mathbf{V}_{\phi_t}-\epsilon_t\frac{\partial\mathcal{L}}{\partial\mathbf{V}_\phi}\\ &=\mathbf{V}_{\phi_t}-2\epsilon_t\big(\big(\mathbf{V}_{\phi_t}\mathbf{U}_{\phi_t}^\top-\Phi^\top\big)\circ\mathbf{P}^\top\big)\mathbf{U}_{\phi_t}+\lambda\mathbf{V}_{\phi_t}\big). \quad (17)\end{aligned}$$

Finally, when $\mathbf{U}_\phi$ and $\mathbf{V}_\phi$ are fixed, $\mathbf{W}$ can be updated as

$$\begin{aligned}\mathbf{W}_{t+1}&=\mathbf{W}_t-\epsilon_t\frac{\partial\mathcal{L}}{\partial\mathbf{W}}\\ &=\mathbf{W}_t-2\epsilon_t\big(\eta\mathbf{U}_{\phi_t}^\top\mathbf{Q}^\top\mathbf{Q}(\mathbf{U}_{\phi_t}\mathbf{W}_t-\mathbf{Y})+\beta\mathbf{W}_t\big). \quad (18)\end{aligned}$$

D. Randomness Reduction via Multiple Feature Maps

To construct the explicit feature map ϕ for generating the augmented matrix Φ in an approximated RKHS, first, we need to generate two random matrices $\mathbf{Z}_S \in \mathbb{R}^{d_S\times n}$ and $\mathbf{Z}_T \in \mathbb{R}^{d_T\times n}$. As a result, the performance of our proposed method may depend on the quality of the two randomly generated matrices. To address the randomness issue caused by $\mathbf{Z}_S$ and $\mathbf{Z}_T$, we borrow the idea from multiple kernel learning (MKL) [57], which aims to learn weights to combine multiple kernels to improve learning performance.

To be specific, at the beginning, we randomly generate h augmented matrices $\{\Phi_i\}_{i=1}^h$ using explicit feature maps. We initialize a weight vector $\boldsymbol{\mu} = [\mu_1, \ldots \mu_h]^\top = [(1/h), \ldots, (1/h)]^\top$, and denote by $\Phi^* = \sum_{i=1}^h \mu_i\Phi_i$ the weighted combination of Φ_i's. By replacing Φ by Φ^* in (12), and solving the optimization problem, we obtain optimal solutions for $\mathbf{U}_\phi$, $\mathbf{V}_\phi$ and $\mathbf{W}$. With the learned $\mathbf{U}_\phi$ and $\mathbf{V}_\phi$, we can update $\boldsymbol{\mu}$ by solving the following optimization problem:

$$\begin{aligned}\min_{\boldsymbol{\mu}}\ &\sum_{(i,j)}\left(\left(\sum_{a=1}^h\mu_a[\Phi_a]_{ij}-(\mathbf{U}_\phi\mathbf{V}_\phi^\top)_{ij}\right)\circ\mathbf{P}_{ij}\right)^2\\ \text{s.t.}\ &\boldsymbol{\mu}\geq\mathbf{0},\ \text{and}\ \mathbf{1}^\top\boldsymbol{\mu}=1.\end{aligned}$$

By defining $\mathbf{f}\in\mathbb{R}^{h\times 1}$, with each element

$$f_a=\sum_{(i,j)}(\Phi_a\circ\mathbf{P})(\Phi_a\circ\mathbf{P})^\top,\ \text{where}\ 1\leq a\leq h$$

and $\boldsymbol{\upsilon}\in\mathbb{R}^{h\times 1}$ with each element $\upsilon_a=\sum_{(i,j)}(\Phi_a\circ\mathbf{P})$, where $1\leq a\leq h$, and $C=\sum_{(i,j)}(\mathbf{U}_\phi\mathbf{V}_\phi^\top)\circ\mathbf{P}$, the optimization problem (19) can be rewritten as

$$\begin{aligned}\min_{\boldsymbol{\mu}}\ &\boldsymbol{\mu}^\top\mathbf{f}\,\mathbf{f}^\top\boldsymbol{\mu}-2C\,\boldsymbol{\upsilon}^\top\boldsymbol{\mu}\\ s.t.\ &\boldsymbol{\mu}\geq\mathbf{0}\\ &\mathbf{1}^\top\boldsymbol{\mu}=1. \quad (19)\end{aligned}$$

We apply quadratic programming to solve the optimization problem. Note that the notation $\sum_{(i,j)}\mathbf{M}$ means the sum of all entries of the matrix $\mathbf{M}$.

After we obtain an updated vector $\boldsymbol{\mu}$, we reconstruct $\Phi^* = \sum_{i=1}^h \mu_i\Phi_i$ and solve the optimization problem (12) again

Algorithm 2 Proposed Kernelized HDA Algorithm

Input: An augmented input matrix $\mathbf{X}$ defined in (3), a label matrix $\mathbf{Y} = \left[\mathbf{Y}_S^\top \ \mathbf{Y}_{T_l}^\top \ \mathbf{Y}_{T_u}^\top\right]^\top$, and the maximum number of iterations, T

Initializations:

1: Randomly generate h pairs of Gaussian matrices $\{(\mathbf{Z}_{S_i}, \mathbf{Z}_{T_i})\}_{i=1}^h$

2: Construct h augmented matrix after feature maps $\{\Phi_i\}_{i=1}^h$based on (13)

3: Initialize matrices $\mathbf{U}_\phi \in \mathbb{R}^{(N_S+N_T)\times k}$, $\mathbf{V}_\phi \in \mathbb{R}^{4n\times k}$, and $\mathbf{W} \in \mathbb{R}^{k\times m}$, set $\boldsymbol{\mu} = [\frac{1}{h}, \frac{1}{h}, ...\frac{1}{h}]^\top \in \mathbb{R}^{h\times 1}$, and $t = 0$

while $t \leq T$ **do**

1: compute $\Phi^* = \sum_i^h \mu_i \Phi_i$

2: update $\mathbf{U}_\phi$ based on (16)

3: update $\mathbf{V}_\phi$ based on (17)

4: update $\mathbf{W}$ based on (18)

5: update $\boldsymbol{\mu}$ based on (19)

6: $t = t + 1$

end while

Output : $\mathbf{U}_\phi$, $\mathbf{V}_\phi$ and $\mathbf{W}$

to obtain updated $\mathbf{U}_\phi$, $\mathbf{V}_\phi$ and $\mathbf{W}$. This procedure is iteratively done until some stopping criterion is met. The overall algorithm of our kernelized matrix-factorization-based HDA method is presented in Algorithm 2.

E. Discussion

Compared with the existing matrix completion-based approaches for HDA [10], [11], the key difference is that our proposed method first maps instances of either the source domain or the target domain with augmented features of missing values to an approximated RKHS and then performs matrix factorization to learn a new dictionary and representations in the RKHS for both the source-domain and the target-domain data. In this way, the nonlinear correlations between instances and features can be captured, which the existing matrix completion-based approaches fail to exploit. In addition, the computed MMD distance in our proposed method is more accurate than that using a linear kernel in [11]. Moreover, we utilize source-domain labeled data and a few target-domain labeled data to align the instances from different domains for feature learning in the RKHS, rather than requiring cross-domain correspondences to be given in advance.

Regarding kernel-based matrix factorization, our formulation shares a similar idea with the kernel sparse coding method proposed by Gao *et al.* [58] at a high level, which aims to learn a dictionary and sparse representations for instances in a kernel space. However, their method cannot be directly applied for HDA as it assumes all the instances are of homogeneous features and does not encode any regularization terms to reduce the difference between domains. Recently, a new kernelized matrix factorization method was proposed for collaborative filtering [59], whose formulation is quite different from ours. Instead of mapping the user-item rating matrix to an RKHS for factorization, their method aims to learn representations of users (rows) and items (columns) in an RKHS such that their inner product in the RKHS equal to the values of the corresponding entries in the rating matrix. Although their method can be directly applied to our problem setting, as will be shown in experiments, its performance is poor on HDA problems as it was originally designed for collaborative filtering, not HDA problems.

VI. Experiment

In this section, we conduct experiments on object recognition with heterogeneous image features and cross-language text classification to verify the effectiveness of our proposed method in the linear and the nonlinear manners, respectively, compared with some state-of-the-art HDA baselines. For the experimental setup, we assume that the source-domain and the target-domain data are obtained from a certain environment. This means that only one source domain and one target domain are considered for evaluation. We use a limited number of target-domain labeled examples for our HDA problems. Note that the target-domain unlabeled examples are used both in training and evaluation. We report the averaged results over ten random splits for each HDA task.

A. Experimental Setup

1) Object Recognition: We follow the setting in [6] and [40] by using images collected from the Amazon data set (A), the digital single-lens reflex camera (DSLR) data set (D), the Webcam data set (W), and the Caltech-256 data set (C), where ten common categories in all these data sets are used for conduct experiments. Examples of the images [6] are shown in Fig. 2. For the HDA setting, we use instances of the DeCAF_6 features [60] with dimension 4096 to construct a source domain and instances of the bag-of-words-based speeded-up robust features (SURF) [5] with dimension 800 to construct a target domain. We randomly select 20 labeled instances per category from the source domain and 3 labeled instances per category from the target domain for training, and the remaining instances in target domain are used for testing.

2) Cross-Language Text Classification: We use Multilingual Reuters Collection data set [61] for cross-language text classification experiment. This data set contains around 11 000 articles with 6 categories in 5 different languages (English, French, German, Italian, and Spanish). All the documents are represented by term frequency–inverse document frequency (TF-IDF) [63]. We randomly select 100 labeled documents per category in source domain and n_{T_l} labeled documents per category in target domain for training, where n_{T_l} varies in $\{5, 10, 15, 20\}$, and randomly select 500 unlabeled documents per category in target domain for testing. We also perform principle component analysis (PCA) on the TF-IDF features with 60% energy preserved in order to keep consistent with baseline methods. We pick up one language for source domain and another for target domain.

3) Baseline Methods: We compare our proposed method with the following baselines: SVM_t, which simply employs labeled data from the target domain to train a model, and some state-of-the-art HDA methods, including semi-supervised HFA (SHFA) [48], semisupervised subspace

Fig. 2. Example images of Amazon, DSLR, Webcam, and Caltech-256 data sets.

coprojection (SCP) [38], cross-domain landmark selection (CDLS) [40], transfer neural trees (TNT) [39], and DMMC [11]. We also consider to compare with a recent proposed kernelized matrix factorization, multikernel matrix factorization (MKMF) [59], to learn features for HDA. The details of each baseline methods are summarized below.

1) *SHFA* [48]: We consider the SHFA as one of our baseline methods. This method learned two transformation mappings based on source domain and target domain and then augmented the feature into a new space. SVM with hinge loss is incorporated to learn a classifier in a semisupervised setting.
2) *SCP* [38]: The SCP method shared a similar idea with our formulation in linear-kernel manner. The difference is that SCP directly learned mapping function based on the original features. Our method jointly learns a latent feature representation as well as a dictionary based on the augmented feature with manifold regularization. To evaluate whether feature augmentation and manifold regularization can further boost the HDA performance, we consider SCP as one of the baselines.
3) *CDLS* [40]: The CDLS method jointly learned mapping function based on Source domain as well as the active samples, which helps to align the feature distribution between the source and target domains. The SVM classifier was trained based on labeled samples to predict the unlabeled target domain samples.
4) *TNT* [39]: The TNT was proposed based on neural networks, which combined stochastic pruning and embedding layer to adapt representative neurons for heterogeneous cross-domain data and preserve the prediction and structural consistency in target domain.
5) *DMMC* [11]: The DMMC was designed for HDA problem where matrix completion was conducted with the regularization of low-rank, sparse, and distribution matching regularization. Our proposed method aims to solve the limitation appeared in DMMC.
6) *MKMF* [59]: The MKMF-based method was originally proposed for collaborative filtering, where the kernelization step was conducted based on latent space and dictionary instead of the feature space. We adopt MKMF as kernelized matrix factorization formulation baseline to evaluate whether our adopted random kitchen sinks method is more effective for HDA problem.

B. Parameter Selection

For our proposed method, there are five tradeoff parameters, namely, μ, γ, η, λ, and β, to influence the impact of distribution matching, graph regularization, discriminative classifier, low-rank regularization, and model regularization, respectively. We tune the tradeoff parameters using both the source-domain and target-domain labeled data. To be more specific, we validate the parameters as follows: μ from $\{10^{-2}, 10^{-1}, \ldots, 10^{2}\}$, γ from $\{10^{-4}, 10^{-3}, \ldots, 1\}$, η from $\{0.01, 0.02, 0.05, 0.1\}$, λ from $\{0.1, 0.2, 0.5, 1\}$, and β from $\{10^{-3}, 10^{-2}, \ldots, 10\}$. We pick the parameter setting with the best validation set classification accuracy based on one experiment and the parameters setting is fixed for the other experiments on object recognition task and text classification task, respectively. The same procedure for parameter selection is also adopted by SCP [38]. For SHFA [48] and CDLS [40], we report the results with the best parameter setting. For the multiple kernel bandwidth for MKMF, we consider Gaussian RBF kernel with $\sigma^2 = \{0.01, 0.1, 1, 10, 100\}$. We set the number of basis functions $n = 5000$ to construct $\{(\mathbf{Z}_{S_i}, \mathbf{Z}_{T_i})\}$ based on the Gaussian distribution with 0 mean and the standard value as $\{0.01, 0.1, 1, 10, 100\}$ for MKL, and the dimension of latent feature $k = 200$.

C. Experimental Results

For object recognition, we report the experimental results by constructing HDA tasks on the same data set and cross data sets. Note that for cross data set HDA, we only construct tasks taking DSLR as the target domain since the number of images in DSLR is much smaller than those in other domains. The results are shown in Table I. From the results, we can see that the SVM_t baseline method performs poorly for all different object recognition tasks. This is reasonable because there are only a few labeled in the target domain for training. By exploiting the labeled training data of heterogeneous features from the source domain, we note that the learning performance in terms of prediction accuracy improvements. Among all the HDA methods, our proposed method consistently achieves the best performance over all the object recognition tasks, which indicates the effectiveness of kernel adaptation compared with linear adaptation. Another possible explanation is that, while other HDA methods focus on either MMD alignment [40] or classifier exploiting [48], we focus on both as regularization. We also investigate linear matrix completion, DMMC, without correspondences. We observe that DMMC performs poorly

TABLE I
CLASSIFICATION ACCURACY (IN %) FOR HETEROGENEOUS OBJECT RECOGNITION ($DeCAF_6$ FOR SOURCE DOMAIN, SURF FOR TARGET DOMAIN)

	Amazon/Amazon	Caltech/Caltech	Webcam/Webcam
SVM_t	34.8	30.1	50.2
DMMC	25.9	20.7	45.2
SCP	39.5	31.0	58.3
SHFA	42.9	30.1	61.6
CDLS	43.7	32.3	60.7
TNT	46.2	31.9	63.1
MKMF	44.1	29.6	62.0
Ours (Linear)	43.4	30.1	60.5
Ours (MKL)	**46.8**	**33.8**	**64.2**

	Amazon/DSLR	Caltech/DSLR	Webcam/DSLR
SVM_t	45.0		
DMMC	40.1	39.3	40.2
SCP	49.6	49.9	47.1
SHFA	55.1	55.4	55.7
CDLS	54.3	54.7	53.4
TNT	57.7	56.2	57.5
MKMF	55.9	53.4	55.2
Ours (Linear)	55.6	55.7	55.7
Ours (MKL)	**59.8**	**60.9**	**59.9**

when there are no cross-domain correspondences, which is consistent with the results reported in [11]. This suggests that to use matrix completion-based method, when there are no correspondences, utilizing target-domain label information is crucial for effective knowledge transfer. Moreover, we also observe that our proposed method outperforms another kernelized matrix factorization method, MKMF [59], which is designed for collaborative filtering And, thus, may not be effective for HDA problems. This will be further investigated in experiments on cross-lingual text classification tasks.

Tables II–VI report the comparison results on the multilingual data set with varying numbers of target-domain labeled data. From the table, we can observe that our proposed method generally outperforms other baseline methods in most cases. DMMC without correspondence achieves a relatively poor performance which is consistent with the results on object recognition. TNT also achieves poor performance, which we conjecture that the tree-based neural network may not be able to tackle dense feature input. Another interesting result we find is that the SCP method performs better on cross-domain text classification tasks compared with the results on object recognition. One possible explanation is that linear kernel adaptation can deal with dense feature (we perform PCA on the text data) alignment better since we can extract more statistics information in dense feature compared with sparse one. Similar to the results on object recognition, MKMF performs much worse than our proposed method on cross-lingual text classification. From the table, we can also observe that the performance of all methods increase when the number of target-domain labeled data increases. Our proposed method achieves larger performance gap when the number of target-domain labeled data is relatively small, e.g., $m = 5$ or $m = 10$.

D. Explicit Mapping Dimension Analysis

As claimed in [18], the performance of random kitchen sinks depends on the dimension of the Gaussian random matrix. To investigate how the dimension of explicit feature map can influence the final performance of our proposed algorithm, we conduct experiments on an object recognition task by taking Webcam with $DeCAF_6$ features as the source domain, and DSLR with SURF as the target domain. The average classification results presented in Fig. 3. Based on the results, we observe that the performance increases when the dimension of explicit feature map becomes larger, and then the performance improvements become less marginal when the dimension of explicit feature map is reasonably large.

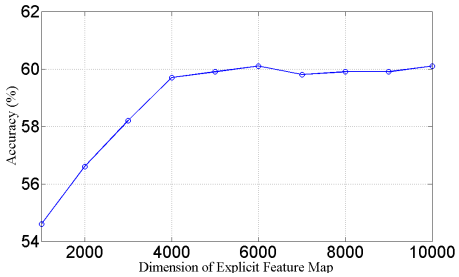
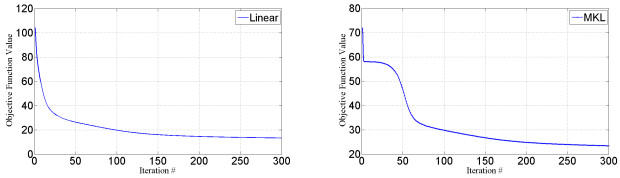


Fig. 3. Dimension of explicit feature map.



(a) (b)

Fig. 4. Convergence curve of our proposed algorithm.

E. Convergence Analysis

We optimize the objective in an iterative manner. Here, we also investigate whether our proposed algorithm can convergence. Fig. 4 shows the convergence curves of our proposed algorithm in both linear and MKL manner based on object recognition task by taking Webcam with $DeCAF_6$ features as the source domain, and DSLR with SURF as the target domain. For each figure, the y-axis denotes the value of the objective function and the x-axis denotes the iteration number. As we observe, our proposed algorithm can converge with sufficient iteration number.

F. Parameter Analysis

To further analyze the parameter sensitivity of our proposed method, we conduct experiments on an object recognition task by taking Webcam with $DeCAF_6$ features as the source domain and DSLR with SURF as the target domain. As discussed in the previous section, we tune the parameter based on the labeled Target-domain data. For this object recognition task, the parameter setting we use for experiments is $\{\mu = 10, \gamma = 10^{-3}, \eta = 10^{-1}, \lambda = 1, \beta = 10^{-2}\}$. We now analyze the sensitivity of one parameter by fixing the values of other parameters. Experimental results are shown in Fig. 5. From the figure, we note that our proposed method is less sensitive to β which is to control the impact of

TABLE II

CLASSIFICATION RESULTS (%) BY CONSIDERING ENGLISH TEXT AS TARGET DOMAIN FOR CROSS-LANGUAGE TEXT CLASSIFICATION

	#Target label=5				#Target label=10				#Target label=15				#Target label=20			
Source	French	Italian	German	Spanish	French	Italian	German	Spanish	French	Italian	German	Spanish	French	Italian	German	Spanish
Target	English				English				English				English			
SVM_t	47.9				58.3				64.0				65.2			
DMMC	42.3	43.0	41.8	42.8	52.7	51.5	52.8	52.4	60.2	61.4	59.5	59.0	61.3	62.1	59.7	61.8
SCP	53.9	54.6	53.7	54.1	64.2	63.8	62.9	64.4	66.3	65.9	66.5	66.1	68.7	69.1	68.2	68.4
SHFA	57.3	57.8	57.4	56.9	66.5	66.2	65.8	66.4	68.2	68.0	67.4	67.2	71.3	70.8	71.1	70.9
CDLS	55.7	55.9	55.6	56.1	65.5	66.0	65.4	65.7	67.4	67.2	68.0	67.8	71.3	70.7	71.2	70.5
TNT	41.2	41.5	40.6	39.8	46.7	46.7	45.3	45.8	47.9	49.6	50.1	48.3	55.5	54.2	52.5	52.7
MKMF	53.1	53.5	54.3	53.0	62.0	61.8	62.9	62.4	66.8	65.6	67.0	65.1	67.6	69.2	68.5	68.2
Ours (Linear)	55.1	54.4	55.2	54.9	64.0	64.1	64.1	63.5	66.0	66.6	65.9	66.5	68.5	69.0	68.6	69.1
Ours (MKL)	**58.8**	**58.4**	**59.2**	**59.1**	**67.1**	**68.0**	**67.2**	**67.3**	**68.4**	**69.6**	**68.9**	**68.5**	**71.5**	**72.2**	**71.7**	**72.0**

TABLE III

CLASSIFICATION RESULTS (%) BY CONSIDERING FRENCH TEXT AS TARGET DOMAIN FOR CROSS-LANGUAGE TEXT CLASSIFICATION

	#Target label=5				#Target label=10				#Target label=15				#Target label=20			
Source	English	Italian	German	Spanish	English	Italian	German	Spanish	English	Italian	German	Spanish	English	Italian	German	Spanish
Target	French				French				French				French			
SVM_t	49.7				62.5				67.2				69.1			
DMMC	46.0	46.2	45.0	45.8	57.4	56.8	57.0	57.9	60.0	60.0	60.1	60.5	63.3	63.1	63.6	63.7
SCP	61.0	60.4	60.0	59.9	68.7	69.0	68.2	68.4	71.1	71.4	71.3	71.0	74.4	74.5	75.0	74.7
SHFA	63.4	61.7	62.3	62.6	70.8	70.4	71.0	70.9	72.0	72.0	71.1	71.2	76.7	76.3	76.5	76.0
CDLS	59.9	59.3	58.8	60.2	71.2	70.5	71.1	70.7	71.8	72.4	72.2	72.1	76.4	75.7	76.2	75.5
TNT	42.4	42.6	41.8	41.8	45.0	46.2	45.5	45.1	48.2	49.0	49.1	47.3	50.6	51.2	51.2	50.9
MKMF	57.2	59.2	58.4	58.0	66.9	68.4	66.7	67.2	70.5	69.4	70.2	70.6	73.0	72.1	72.7	72.3
Ours (Linear)	60.9	61.0	60.5	61.3	68.5	69.1	69.1	68.5	71.8	71.9	70.9	71.8	74.5	74.4	74.8	73.9
Ours (MKL)	**64.4**	**65.0**	**65.0**	**64.6**	**72.4**	**73.0**	**72.2**	**72.7**	**75.3**	**75.0**	**74.8**	**74.5**	**76.8**	**77.4**	**77.8**	**77.1**

TABLE IV

CLASSIFICATION RESULTS (%) BY CONSIDERING GERMAN TEXT AS TARGET DOMAIN FOR CROSS-LANGUAGE TEXT CLASSIFICATION

	#Target label=5				#Target label=10				#Target label=15				#Target label=20			
Source	English	French	Italian	Spanish	English	French	Italian	Spanish	English	French	Italian	Spanish	English	French	Italian	Spanish
Target	German				German				German				German			
SVM_t	49.3				57.9				62.9				65.4			
DMMC	45.4	46.2	45.5	45.9	51.2	50.6	51.3	51.4	55.3	55.6	54.8	54.9	60.0	60.1	60.4	59.5
SCP	55.1	55.6	55.4	55.7	60.1	59.8	59.9	60.2	65.6	66.0	65.4	65.3	68.5	69.4	69.0	68.7
SHFA	62.0	61.5	62.2	61.4	67.0	66.4	66.9	67.5	**71.1**	69.4	70.3	70.1	**70.9**	70.5	**70.7**	70.1
CDLS	60.6	61.0	60.6	60.1	66.5	67.0	66.7	66.7	68.5	68.9	69.0	68.1	70.0	69.9	70.2	70.5
TNT	39.5	39.6	38.4	37.3	42.4	43.2	42.3	44.8	45.7	46.7	46.0	48.0	49.6	50.9	49.0	49.1
MKMF	56.6	56.4	57.3	57.0	61.0	61.4	61.3	60.6	65.3	66.0	65.8	65.1	67.2	67.4	67.9	66.9
Ours (Linear)	58.6	59.1	58.5	58.3	64.5	65.2	65.1	64.5	66.8	66.9	67.1	67.3	69.5	70.4	69.8	69.9
Ours (MKL)	**64.4**	**65.0**	**64.0**	**64.9**	**68.7**	**68.0**	**68.3**	**67.8**	70.2	**70.4**	**70.6**	**70.5**	**70.9**	**71.2**	**70.7**	**70.9**

TABLE V

CLASSIFICATION RESULTS (%) BY CONSIDERING ITALIAN TEXT AS TARGET DOMAIN FOR CROSS-LANGUAGE TEXT CLASSIFICATION

	#Target label=5				#Target label=10				#Target label=15				#Target label=20			
Source	English	French	German	Spanish	English	French	German	Spanish	English	French	German	Spanish	English	French	German	Spanish
Target	Italian				Italian				Italian				Italian			
SVM_t	42.9				55.2				62.7				66.6			
DMMC	35.0	36.4	38.7	36.8	50.7	49.2	50.8	49.9	54.8	53.9	55.0	54.3	60.8	60.3	59.5	61.2
SCP	50.4	50.4	51.0	50.8	60.1	58.9	60.4	59.5	66.7	68.0	67.2	66.2	68.8	70.1	69.2	68.5
SHFA	55.1	54.2	54.9	55.3	63.8	64.2	64.1	63.0	68.2	68.5	67.4	68.6	71.9	**72.3**	72.5	72.0
CDLS	53.3	53.9	54.1	52.9	63.5	62.8	63.0	63.1	68.2	68.1	67.0	68.0	71.8	71.9	72.0	**72.1**
TNT	41.0	41.4	40.4	40.8	45.2	45.2	46.3	46.0	48.1	47.5	48.2	47.8	51.6	52.6	52.0	50.8
MKMF	50.5	50.4	51.3	51.0	61.0	60.8	59.9	60.6	65.9	66.0	65.5	65.6	67.2	67.8	67.0	67.5
Ours (Linear)	52.9	54.1	53.5	53.8	61.1	62.0	61.4	61.2	66.9	67.0	67.0	66.5	70.1	70.6	70.8	69.9
Ours (MKL)	**57.8**	**58.0**	**57.6**	**58.2**	**66.8**	**65.8**	**65.9**	**66.5**	**68.9**	**69.0**	**70.0**	**70.5**	**72.2**	72.1	**73.0**	71.9

the classifier regularization term. Regarding μ which is to control distribution matching, the proposed method achieves good and stable performance when μ is reasonable large, implying the importance of distribution matching. However, we also notice that if μ is set to be a very large value, i.e., $\mu \geq 10^2$, the performance drops. A similar observation is found regarding the parameter γ which is to control the impact of the manifold regularization term. We also find that the term of encoding labeled information is quite important for our proposed method, i.e., η cannot be set to a very small value, which is reasonable since the classifier acts as a bridge between domains through label information. Finally, we find that if λ is set to be smaller than 0.2, the performance drops, which indicates that low-rank regularization is important for HDA problem.

G. Computational Time Analysis

We conduct computational time analysis on the object recognition task with the Webcam data set as the source and

TABLE VI

CLASSIFICATION RESULTS (%) BY CONSIDERING SPANISH TEXT AS TARGET DOMAIN FOR CROSS-LANGUAGE TEXT CLASSIFICATION

	#Target label=5				#Target label=10				#Target label=15				#Target label=20			
Source	English	French	Italian	German	English	French	Italian	German	English	French	Italian	German	English	French	Italian	German
Target	Spanish				Spanish				Spanish				Spanish			
SVM_t	52.5				64.4				67.2				71.3			
DMMC	49.7	50.7	48.8	48.8	57.7	58.2	58.8	59.9	61.4	60.3	61.1	59.9	62.8	63.3	65.5	65.2
SCP	61.9	60.4	61.0	61.5	70.1	69.2	70.3	67.4	73.9	72.2	73.0	71.3	75.1	75.9	76.0	74.6
SHFA	64.9	64.5	64.2	65.6	72.8	72.4	72.5	72.9	74.0	73.5	74.0	73.6	76.9	76.9	76.6	**77.4**
CDLS	61.7	61.9	62.6	63.1	69.5	70.0	68.7	69.7	73.3	72.7	73.2	72.7	76.8	**77.7**	76.2	76.5
TNT	42.1	41.3	40.3	41.9	46.2	44.4	47.5	47.1	49.2	48.5	48.7	49.0	52.9	54.0	55.0	53.7
MKMF	58.6	58.4	57.3	58.0	65.0	66.8	63.9	65.6	69.3	70.0	69.5	68.1	73.0	74.7	74.2	72.9
Ours (Linear)	60.4	61.2	61.6	61.0	69.5	69.8	68.7	68.2	72.9	73.0	73.0	72.5	75.1	75.5	75.9	75.7
Ours (MKL)	**65.6**	**65.5**	**65.3**	**65.9**	**73.8**	**73.3**	**73.6**	**73.8**	**75.7**	**75.4**	**75.6**	**75.7**	**77.5**	77.4	**77.7**	77.2

TABLE VII

AVERAGE TRAINING TIME (IN SECONDS) COMPARISONS OF ALL METHODS ON THE OBJECT RECOGNITION TASK WITH WEBCAM DATA SET

Method	DMMC	SCP	SHFA	CDLS	TNT	MKMF	Ours (Linear)	Ours (Kernel)
Time (second)	6.64	10.03	7.79	5.85	40.34	9.41	8.78	36.26

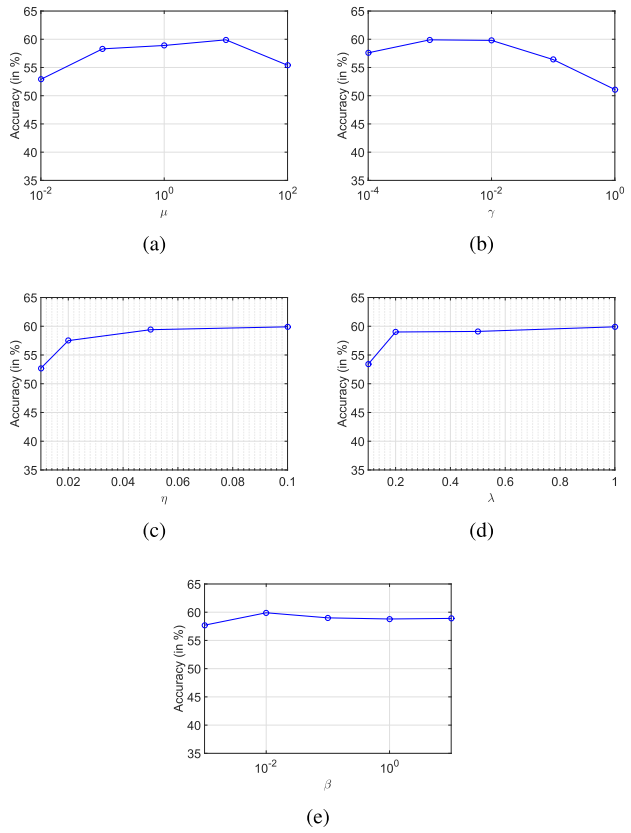


Fig. 5. Parameter sensitivity analysis (Webcam as source domain, DSLR as target domain). (a) Accuracy under varying μ. (b) Accuracy under varying γ. (c) Accuracy under varying η. (d) Accuracy under varying λ. (e) Accuracy under varying β.

the target domains (which is consistent with our parameter analysis study). As our baseline methods consist of both deep-learning-based and nondeep-learning-based methods, we conduct all experiments using K40 GPU for fair comparison. The results are shown in Table VII. We observe that our proposed method with kernelization took more time compared with the other nondeep-learning-based methods, which is reasonable as we adopt random kitchen sinks which significantly enlarges the dimension of feature. However, one can always use kernel acceleration method [62] to accelerate our proposed method, which will be discussed in the future.

VII. CONCLUSION

In this paper, we propose a novel framework for HDA. In contrast with the previous works based on matrix completion, the proposed scheme leverages the advantage of matrix factorization as well as random kitchen sinks, which can be effectively applied to RKHS space. Within the framework, the latent feature embedding, classifier with a distribution matching, and geometric manifold regularizer can be jointly learned. A joint optimization algorithm is further proposed to solve the problem. Extensive experiments on object recognition tasks and text classification task demonstrate the effectiveness of our proposed algorithm over a number of state-of-the-art HDA methods.

Deep learning has also been proved to be effective for domain adaptation task. In the future, we will investigate how to incorporate deep learning techniques into HDA task. Another direction is that, different from traditional domain adaptation task, the theoretical study of generalization bound of HDA is still lack, it is fruitful to investigate the prediction error bound under the setting for heterogeneous domain adaptation.

ACKNOWLEDGMENT

This research was carried out at the Rapid-Rich Object Search (ROSE) Laboratory, Nanyang Technological University, Singapore. The authors also would like to thank the Sino-Singapore International Joint Research Institute for funding the projects titled as "Research on Key Technologies of Biometric Identity Authentication for Financial Services" and "Face Spoofing Detection based on Self-Service System."

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, Oct. 2010.

[2] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[3] J. Blitzer, M. Dredze, and F. Pereira, "Biographies, bollywood, boomboxes and blenders: Domain adaptation for sentiment classification," in *Proc. 45th Annu. Meeting Assoc. Comput. Linguistics*, 2007, pp. 440–447.

[4] S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 751–760.

[5] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2066–2073.

[6] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. ECCV*, 2010, pp. 213–226.

[7] P. Carey, *Data Protection: A Practical Guide to U.K. and E.U. Law*. New York, NY, USA: Oxford Univ. Press, 2018.

[8] A. Salvador *et al.*, "Learning cross-modal embeddings for cooking recipes and food images," in *Proc. CVPR*, Jul. 2017, pp. 3068–3076.

[9] W. Dai, Y. Chen, G.-R. Xue, Q. Yang, and Y. Yu, "Translated learning: Transfer learning across different feature spaces," in *Proc. NIPS*, 2008, pp. 353–360.

[10] M. Xiao and Y. Guo, "A novel two-step method for cross language representation learning," in *Proc. NIPS*, 2013, pp. 1259–1267.

[11] J. T. Zhou, S. J. Pan, I. W. Tsang, and S.-S. Ho, "Transfer learning for cross-language text categorization through active correspondences construction," in *Proc. AAAI*, 2016, pp. 2400–2406.

[12] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.

[13] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *Proc. CVPR*, Jun. 2011, pp. 1785–1792.

[14] J. T. Zhou, I. W. Tsang, S. J. Pan, and M. Tan, "Heterogeneous domain adaptation for multiple classes," in *Proc. AISTATS*, 2014, pp. 1095–1103.

[15] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A Kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. 3, pp. 723–773, 2012.

[16] B. K. Sriperumbudur, K. Fukumizu, A. Gretton, G. R. G. Lanckriet, and B. Schölkopf, "Kernel choice and classifiability for RKHS embeddings of probability distributions," in *Proc. NIPS*, 2009, pp. 1750–1758.

[17] B. Schlkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA, MIT, Press, 2001.

[18] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. NIPS*, 2007, pp. 1177–1184.

[19] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Scholkopf, "Correcting sample selection bias by unlabeled data," in *Proc. NIPS*, 2006, pp. 601–608.

[20] S. Li, S. Song, and G. Huang, "Prediction reweighting for domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 7, pp. 1682–1695, Jul. 2017.

[21] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer joint matching for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2014, pp. 1410–1417.

[22] S. Mehrkanoon and J. A. K. Suykens, "Regularized semipaired kernel CCA for domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 7, pp. 3199–3213, Jul. 2018.

[23] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, 2015, pp. 97–105.

[24] M. Long, Y. Cao, Z. Cao, J. Wang, and M. I. Jordan, "Transferable representation learning with deep adaptation networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2017, pp. 84–90.

[26] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. NIPS*, 2016, pp. 136–144.

[27] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 2962–2971.

[28] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 59, pp. 1–35, 2016.

[29] J. Hoffman *et al.*, "CyCADA: Cycle-consistent adversarial domain adaptation," in *Proc. ICML*, 2018, pp. 1989–1998.

[30] H. Li, S. J. Pan, S. Wang, and A. C. Kot, "Domain generalization with adversarial feature learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5400–5409.

[31] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 2672–2680.

[32] C. Wang and S. Mahadevan, "Heterogeneous domain adaptation using manifold alignment," in *Proc. IJCAI*, 2011, pp. 1541–1546.

[33] M. Harel and S. Mannor, "Learning from multiple outlooks," in *Proc. ICML*, 2011, pp. 401–408.

[34] X. Shi, Q. Liu, W. Fan, P. S. Yu, and R. Zhu, "Transfer learning on heterogenous feature spaces via spectral transformation," in *Proc. ICDM*, Dec. 2010, pp. 1049–1054.

[35] J. Choo, S. Bohn, G. C. Nakamura, A. M. White, and H. Park, "Heterogeneous data fusion via space alignment using nonmetric multidimensional scaling," in *Proc. SIAM*, 2012, pp. 177–188.

[36] L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for heterogeneous domain adaptation," in *Proc. ICML*, 2012, pp. 667–674.

[37] X. Shi, Q. Liu, W. Fan, and P. S. Yu, "Transfer across completely different feature spaces via spectral embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 4, pp. 906–918, Apr. 2013.

[38] M. Xiao and Y. Guo, "Semi-supervised subspace co-projection for multi-class heterogeneous domain adaptation," in *Proc. ECML/PKDD*, 2015, pp. 525–540.

[39] W.-Y. Chen, T.-M. H. Hsu, Y.-H. H. Tsai, Y.-C. F. Wang, and M.-S. Chen, "Transfer neural trees for heterogeneous domain adaptation," in *Proc. ECCV*, 2016, pp. 399–414.

[40] Y.-H. H. Tsai, Y.-R. Yeh, and Y.-C. F. Wang, "Learning cross-domain landmarks for heterogeneous domain adaptation," in *Proc. CVPR*, Jun. 2016, pp. 5081–5090.

[41] Y. Yan *et al.*, "Learning discriminative correlation subspace for heterogeneous domain adaptation," in *Proc. IJCAI*, 2017, pp. 3252–3258.

[42] H. Li, S. J. Pan, R. Wan, and A. C. Kot, "Heterogeneous transfer learning via deep matrix completion with adversarial kernel embedding," in *Proc. 33rd AAAI Conf. Artif. Intell. (AAAI)*, 2019.

[43] P. Prettenhofer and B. Stein, "Cross-language text classification using structural correspondence learning," in *Proc. ACL*, 2010, pp. 1118–1127.

[44] F. Zhuang *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, Nov. 2012.

[45] L. Yang, L. Jing, J. Yu, and M. K. Ng, "Learning transferred weights from co-occurrence data for heterogeneous transfer learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2187–2200, Nov. 2016.

[46] Y. Luo, Y. Wen, and D. Tao, "Heterogeneous multitask metric learning across multiple domains," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4051–4064, Sep. 2017.

[47] H. Daumé, III, "Frustratingly easy domain adaptation," in *Proc. ACL*, 2007, pp. 256–263.

[48] W. Li, L. Duan, D. Xu, and I. W. Tsang, "Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 6, pp. 1134–1148, Jun. 2013.

[49] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A Kernel method for the two-sample-problem," in *Proc. NIPS*, 2006, pp. 513–520.

[50] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. ICCV*, Dec. 2013, pp. 2960–2967.

[51] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. ICCV*, Dec. 2013, pp. 2200–2207.

[52] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *Proc. ICCV*, Nov. 2011, pp. 999–1006.

[53] J. Ni, Q. Qiu, and R. Chellappa, "Subspace interpolation via dictionary learning for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2013, pp. 692–699.

[54] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proc. ICML*, 2005, pp. 713–719.

[55] J. Ham, D. D. Lee, and L. K. Saul, "Semisupervised alignment of manifolds," in *Proc. AISTATS*, 2005, pp. 120–127.

[56] F. Chung, *Spectral Graph Theory*, no. 92. Providence, RI, USA: American Mathematical Society, 1997.

[57] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan, "Multiple kernel learning, conic duality, and the SMO algorithm," in *Proc. ICML*, 2004, p. 6.

[58] S. Gao, I. W.-H. Tsang, and L.-T. Chia, "Sparse representation with kernels," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 423–434, Feb. 2013.

[59] X. Liu, C. Aggarwal, Y.-F. Li, X. Kong, X. Sun, and S. Sathe, "Kernelized matrix factorization for collaborative filtering," in *Proc. SIAM*, 2016, pp. 378–386.
[60] Y. Jia *et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 675–678.
[61] M. Amini, N. Usunier, and C. Goutte, "Learning from multiple partially observed views—An application to multilingual text categorization," in *Proc. NIPS*, 2009, pp. 28–36.
[62] Z. Lu *et al.* (2014). "How to scale up Kernel methods to be as good as deep neural nets." [Online]. Available: https://arxiv.org/abs/1411.4000
[63] J. Ramos *et al.*, "Using TF-IDF to determine word relevance in document queries," in *Proc. 1st Instruct. Conf. Mach. Learn.*, Piscataway, NJ, USA, vol. 242, 2003, pp. 133–142.

Haoliang Li received the B.S. degree in communication engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2013, and the Ph.D. degree from Nanyang Technological University (NTU), Singapore, in 2018.

He is currently a Research Fellow with the Rapid-Rich Object Search Laboratory, NTU, Singapore. His current research interests include information forensics and security and transfer learning.

Sinno Jialin Pan received the Ph.D. degree in computer science from The Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2011.

He was with Nanyang Technological University (NTU), Singapore. From 2010 to 2014, he was a Scientist and the Lab Head of text analytics with the Data Analytics Department, Institute for Infocomm Research, Singapore. In 2014, he joined NTU as a Nanyang Assistant Professor (University-named Assistant Professor). He is currently a Provost's Chair Associate Professor with the School of Computer Science and Engineering, NTU. His current research interests include transfer learning, and its applications to wireless-sensor-based data mining, text mining, sentiment analysis, and software engineering.

Dr. Pan was named to AI 10 to Watch by *IEEE Intelligent Systems* magazine in 2018.

Shiqi Wang (M'15) received the B.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2008, and the Ph.D. degree in computer application technology from Peking University, Beijing, China, in 2014.

From 2014 to 2016, he was a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. From 2016 to 2017, he was a Research Fellow with the Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore. He is currently an Assistant Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong. He has proposed more than 40 technical proposals to ISO/MPEG, ITU-T, and AVS standards. He has authored more than 150 refereed journal/conference papers. His current research interests include video compression, image/video quality assessment, and image/video search and analysis.

Dr. Wang was a recipient of the Best Paper Award of IEEE Multimedia 2018, the Best Paper Award at the 2017 Pacific-Rim Conference on Multimedia (PCM). He has coauthored a paper that received the Best Student Paper Award in IEEE International Conference on Image Processing 2018.

Alex C. Kot (S'85–M'89–SM'98–F'06) was the Head of the Division of Information Engineering, School of Electrical and Electronic Engineering. Since 1991, he has been with Nanyang Technological University (NTU), Singapore, where he was an Associate Chair (Research) and the Vice Dean (Research) of the School of Electrical and Electronic Engineering, and an Associate Dean of the College of Engineering. He is currently a Professor and the Director of the Rapid-Rich Object Search (ROSE) Laboratory, NTU, Singapore, and the NTU-PKU Joint Research Institute, Singapore. His current research interests include signal processing for communication, biometrics, image forensics, information security and computer vision, and machine learning.

Mr. Kot is a fellow of IES and Academy of Engineering, Singapore. He was a recipient of the Best Teacher of the Year Award. His coauthored papers received several best paper awards, including ICPR, the IEEE International Workshop on Information Forensics and Security (WIFS), and the International Workshop on Digital Forensics and Watermarking (IWDW). He was elected as the IEEE Distinguished Lecturer for the Signal Processing Society and the Circuits and Systems Society. He has served for the IEEE SP Society in various capacities, such as the General Co-Chair for the 2004 IEEE International Conference on Image Processing and the Vice-President for the IEEE Signal Processing Society. He served as an Associate Editor for more than ten journals, mostly for the IEEE transactions.