2017 IEEE International Conference on Data Mining

Balanced Distribution Adaptation for Transfer Learning

Jindong Wang[*†‡], Yiqiang Chen[*†‡], Shuji Hao[§], Wenjie Feng[*‡††], Zhiqi Shen[‖]
[*]*Beijing Key Laboratory of Mobile Computing and Pervasive Device*
[††]*CAS Key Laboratory of Network Data Science & Technology*
[†]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*
[‡]*University of Chinese Academy of Sciences,*[§]*Institute of High Performance Computing, A*STAR*
[‖]*School of Computer Science and Engineering, Nanyang Technological University, Singapore*
Email:{wangjindong,yqchen}@ict.ac.cn, haosj@ihpc.a-star.edu.sg, fengwenjie@software.ict.ac.cn, zqshen@ntu.edu.sg

***Abstract*—Transfer learning has achieved promising results by leveraging knowledge from the source domain to annotate the target domain which has few or none labels. Existing methods often seek to minimize the distribution divergence between domains, such as the marginal distribution, the conditional distribution or both. However, these two distances are often treated equally in existing algorithms, which will result in poor performance in real applications. Moreover, existing methods usually assume that the dataset is balanced, which also limits their performances on imbalanced tasks that are quite common in real problems. To tackle the distribution adaptation problem, in this paper, we propose a novel transfer learning approach, named as Balanced Distribution Adaptation (BDA), which can adaptively leverage the importance of the marginal and conditional distribution discrepancies, and several existing methods can be treated as special cases of BDA. Based on BDA, we also propose a novel Weighted Balanced Distribution Adaptation (W-BDA) algorithm to tackle the class imbalance issue in transfer learning. W-BDA not only considers the distribution adaptation between domains but also adaptively changes the weight of each class. To evaluate the proposed methods, we conduct extensive experiments on several transfer learning tasks, which demonstrate the effectiveness of our proposed algorithms over several state-of-the-art methods.**

***Keywords*-Transfer learning, domain adaptation, distribution adaptation, class imbalance**

I. Introduction

Preparing labeled data is crucial for training machine learning algorithms. However, it is often expensive and time-consuming to obtain sufficient labeled data in real applications. In this case, transfer learning [1] has been a promising approach by transferring knowledge from a labeled source domain to the target domain. Transfer learning often assumes the training and testing data are from similar but different distributions [1]. For instance, the images of an object taken in different angles, backgrounds and illuminations could lead to different marginal or conditional distributions. By observing this, existing transfer learning methods are mainly focusing on *distribution adaptation* to minimize the distribution divergence between domains [2], [3], [4].

Most of the existing distribution adaptation methods adapt either marginal distribution [5], conditional distribution [6] or both [2], [4]. It is shown in [2] that adapting both could achieve better performance. The work of [2], [4], [7] also proposed several approaches to adapt the joint distribution. However, those two distributions are often treated equally in existing methods, while the importance of each other is not leveraged. When the datasets are much more dissimilar, it means the marginal distributions are more dominant; when the datasets are similar, it means the conditional distributions needs more attention. Hence, it will deteriorate the performance of algorithms by only adding them together with equal weight. Therefore, how to adaptively leverage the importance of each distribution is a critical problem.

Moreover, class imbalance often exists in many transfer learning scenarios. When the class proportion of domains is highly imbalanced, it needs to be considered carefully for distribution adaptation. Existing methods [2], [4] often ignore this issue by treating the classes as balanced across domains, or they only handle the bias on one domain [3], and this may hinder the effectiveness of transfer learning. Therefore, how to handle the class imbalance situation in transfer learning is another important challenge.

In this paper, we propose two novel methods to tackle the above two issues. For distribution adaptation, we propose Balanced Distribution Adaptation (BDA). BDA can not only adapt both the marginal and conditional distributions between domains, but also leverage the importance of those two distributions, thus it can be effectively adjusted to specific transfer learning tasks. Several existing methods can be regarded as special cases of BDA. Based on BDA, we also propose a novel Weighted Balanced Distribution Adaptation (W-BDA) algorithm to tackle the class imbalance issue in transfer learning. The proposed W-BDA can adaptively change the weight of each class when performing distribution adaptation. To evaluate BDA and W-BDA, we conduct extensive experiments on five image datasets.

To sum up, our contributions are mainly three-fold:

1) We propose a novel transfer learning method, which is named as BDA to balance the marginal and conditional distribution adaptation. BDA can adaptively adjust the importance of those two distances and can achieve a better performance. Several transfer learning algorithms can be regarded as special cases of BDA.

2374-8486/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDM.2017.150

2) We also propose another novel method W-BDA by extending BDA to handle the class imbalance problem which is common in transfer learning. The proposed W-BDA not only considers the distribution adaptation of domains but also adaptively changes the weight of each class, thus it can handle the class imbalance problem for transfer learning.

3) We conduct extensive experiments on five image datasets to evaluate the BDA and W-BDA methods, indicating their superiority against other state-of-the-art methods.

II. Related Work

Transfer learning has been widely applied to activity recognition [8], incremental learning [9], and online learning [10], [11]. Our proposed BDA and W-BDA are mainly related to the feature-based transfer learning methods. Thus, in this section, we present a detailed discussion on this category, specifically on two aspects.

Joint distribution adaptation. [12] proposed to jointly select feature and preserve structural properties. Long *et al.* [2] proposed joint distribution adaptation method (J-DA) to match both marginal and conditional distribution between domains. Others extended JDA by adding structural consistency [4], domain invariant clustering [7], and target selection [13]. Those methods tend to ignore the importance between two distinct distributions by just adding them together. However, when there is a large discrepancy between both distributions, those methods cannot evaluate the importance of each distribution, and may not generalize well in most cases. Our work is capable of investigating the importance of each distribution. Thus it can be more generalized to transfer learning scenarios with complex data distributions.

Class imbalance problem. Previous sample re-weighting methods [14] only learned weights of specific samples, but ignore the class weights balance for different classes. [15] developed a Closest Common Space Learning (CCSL) method to adapt the cross-domain weights. CCSL is an instance selection method, while ours is a feature based approach. Multiset feature learning was proposed in [16] to learn discriminant features. [3] proposed weighted maximum mean discrepancy to construct a source reference collection on the target domain but it only adapted the prior of source domain, while our method could adapt the priors from both source and target domains. [17] tackled the imbalance issue when target domain has some labels, while in our method, target domain has no labels. [18] adjusted the weights of different samples according to their predictions, while our work focuses on adjusting the weight of each class.

III. Balanced Distribution Adaptation

This section elaborates our proposed algorithms. First, we introduce the problem definition. Then, we present the Balanced Distribution Adaptation (BDA) approach. Finally, the Weighted BDA (W-BDA) method is introduced.

A. Problem Definition

Given a labeled source domain $\{\mathbf{x}_{s_i}, y_{s_i}\}_{i=1}^{n}$, an unlabeled target domain $\{\mathbf{x}_{t_j}\}_{j=1}^{m}$, and assume feature space $\mathcal{X}_s = \mathcal{X}_t$, label space $\mathcal{Y}_s = \mathcal{Y}_t$ but marginal distributions $P_s(\mathbf{x}_s) \neq P_t(\mathbf{x}_t)$ with conditional distributions $P_s(y_s|\mathbf{x}_s) \neq P_s(y_t|\mathbf{x}_t)$. Transfer learning aims to learn the labels $\mathbf{y}_t$ of $\mathcal{D}_t$ using the source domain $\mathcal{D}_s$.

Balanced distribution adaptation solves the transfer learning problem by *adaptively* minimizing the marginal and conditional distribution discrepancy between domains, and handle the class imbalance problem, i.e. to minimize the discrepancies between: 1) $P(\mathbf{x}_s)$ and $P(\mathbf{x}_t)$, 2) $P(y_s|\mathbf{x}_s)$ and $P(y_t|\mathbf{x}_t)$.

B. Balanced Distribution Adaptation

Transfer learning methods often seek to adapt both the marginal and conditional distributions between domains [2], [7]. Specifically, this refers to minimizing the distance

$$\begin{aligned} D(\mathcal{D}_s, \mathcal{D}_t) \approx\ & D(P(\mathbf{x}_s), P(\mathbf{x}_t)) \\ & + D(P(y_s|\mathbf{x}_s), P(y_t|\mathbf{x}_t)) \end{aligned} \tag{1}$$

However, simply matching both distributions is **not** enough. Existing methods usually assume they are equally important, and that implicit assumption does not hold. In this section, we propose Balanced Distribution Adaptation (B-DA) to adaptively adjust the importance of both the marginal and conditional distributions based on each specific tasks. Concretely speaking, BDA exploits a *balance factor* μ to leverage the different importance of distributions:

$$\begin{aligned} D(\mathcal{D}_s, \mathcal{D}_t) \approx\ & (1-\mu) D(P(\mathbf{x}_s), P(\mathbf{x}_t)) \\ & + \mu D(P(y_s|\mathbf{x}_s), P(y_t|\mathbf{x}_t)) \end{aligned} \tag{2}$$

where $\mu \in [0, 1]$. When $\mu \to 0$, it means the datasets are more dissimilar, so the marginal distribution is more dominant; when $\mu \to 1$, it reveals the datasets are similar, so the conditional distribution is more important to adapt. Therefore, the balance factor μ can adaptively leverage the importance of each distribution and lead to good results.

It is worth noting that, since the target domain $\mathcal{D}_t$ has no labels, it is not feasible to evaluate the conditional distribution $P(y_t|\mathbf{x}_t)$. Instead, we use the class conditional distribution $P(\mathbf{x}_t|y_t)$ to approximate $P(y_t|\mathbf{x}_t)$. Because $P(\mathbf{x}_t|y_t)$ and $P(y_t|\mathbf{x}_t)$ can be quite involved according to the sufficient statistics when sample sizes are large [2]. In order to compute $P(\mathbf{x}_t|y_t)$, we apply prediction on $\mathcal{D}_t$ using some base classifier trained on $\mathcal{D}_s$ to get the soft labels for $\mathcal{D}_t$. The soft labels may be less reliable, so we iteratively refine the them.

In order to compute the marginal and conditional distribution divergences in Eq. (2), we adopt maximum mean discrepancy (MMD) [5] to empirically estimate both distribution discrepancies. As a nonparametric measurement, MMD has been widely applied to many existing transfer learning approaches [5], [2]. Formally speaking, Eq. (2) can be represented as

$$
\begin{aligned}
D(\mathcal{D}_s, \mathcal{D}_t) \approx & (1-\mu) \left\| \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_{s_i} - \frac{1}{m} \sum_{j=1}^{m} \mathbf{x}_{t_j} \right\|_{\mathcal{H}}^{2} \\
& + \mu \sum_{c=1}^{C} \left\| \frac{1}{n_c} \sum_{\mathbf{x}_{s_i} \in \mathcal{D}_s^{(c)}} \mathbf{x}_{s_i} - \frac{1}{m_c} \sum_{\mathbf{x}_{t_j} \in \mathcal{D}_t^{(c)}} \mathbf{x}_{t_j} \right\|_{\mathcal{H}}^{2}
\end{aligned}
\tag{3}
$$

where $\mathcal{H}$ denotes the reproducing kernel Hilbert space (RKHS), $c \in \{1, 2, \cdots, C\}$ is the distinct class label, n, m denote the number of samples in the source / target domain, and $\mathcal{D}_s^{(c)}$ and $\mathcal{D}_t^{(c)}$ denote the samples belonging to class c in source and target domain, respectively. $n_c = |\mathcal{D}_s^{(c)}|, m_c = |\mathcal{D}_t^{(c)}|$, denoting the number of samples belonging to $\mathcal{D}_s^{(c)}$ and $\mathcal{D}_t^{(c)}$, respectively. The first term denotes the marginal distribution distance between domains, while the second term is the conditional distribution distance.

By further taking advantage of matrix tricks and regularization, Eq. (2) can be formalized as:

$$
\begin{aligned}
& \min \ \operatorname{tr} \left(\mathbf{A}^\top \mathbf{X} \left((1-\mu)\mathbf{M}_0 + \mu \sum_{c=1}^{C} \mathbf{M}_c \right) \mathbf{X}^\top \mathbf{A} \right) + \lambda \|\mathbf{A}\|_F^2 \\
& \text{s.t. } \mathbf{A}^\top \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{A} = \mathbf{I}, \quad 0 \leq \mu \leq 1
\end{aligned}
\tag{4}
$$

There are two terms in Eq. (4): the adaptation of marginal and conditional distribution with balance factor (term 1), and the regularization term (term 2). λ is the regularization parameter with $\|\cdot\|_F^2$ the Frobenius norm. Two constraints are involved in Eq. (4): the first constraint ensures that the transformed data ($\mathbf{A}^\top \mathbf{X}$) should preserve the inner properties of the original data. The second constraint denotes the range of the balance factor μ.

More specifically, in Eq. (4), $\mathbf{X}$ denotes the input data matrix composed of $\mathbf{x}_s$ and $\mathbf{x}_t$, $\mathbf{A}$ denotes the transformation matrix, $\mathbf{I} \in \mathbb{R}^{(n+m) \times (n+m)}$ is the identity matrix, and $\mathbf{H}$ is the centering matrix i.e. $\mathbf{H} = \mathbf{I} - (1/n)\mathbf{1}$. Similar as in work [2], $\mathbf{M}_0$ and $\mathbf{M}_c$ are MMD matrices and can be constructed in the following ways:

$$
(\mathbf{M}_0)_{ij} = \begin{cases} \frac{1}{n^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s \\ \frac{1}{m^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t \\ -\frac{1}{mn}, & \text{otherwise} \end{cases}
\tag{5}
$$

$$
(\mathbf{M}_c)_{ij} = \begin{cases} \frac{1}{n_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \\ \frac{1}{m_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ -\frac{1}{m_c n_c}, & \begin{cases} \mathbf{x}_i \in \mathcal{D}_s^{(c)}, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ \mathbf{x}_i \in \mathcal{D}_t^{(c)}, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \end{cases} \\ 0, & \text{otherwise} \end{cases}
\tag{6}
$$

Learning algorithm: Denote $\Phi = (\phi_1, \phi_2, \cdots, \phi_d)$ as Lagrange multipliers, then Lagrange function for Eq. (4) is

$$
\begin{aligned}
L = & \operatorname{tr} \left(\mathbf{A}^\top \mathbf{X} \left((1-\mu)\mathbf{M}_0 + \mu \sum_{c=1}^{C} \mathbf{M}_c \right) \mathbf{X}^\top \mathbf{A} \right) \\
& + \lambda \|\mathbf{A}\|_F^2 + \operatorname{tr} \left((\mathbf{I} - \mathbf{A}^\top \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{A}) \Phi \right)
\end{aligned}
\tag{7}
$$

Set derivative $\partial L / \partial \mathbf{A} = 0$, the optimization can be derived as a generalized eigendecomposition problem

$$
\left(\mathbf{X} \left((1-\mu)\mathbf{M}_0 + \mu \sum_{c=1}^{C} \mathbf{M}_c \right) \mathbf{X}^\top + \lambda \mathbf{I} \right) \mathbf{A} = \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{A} \Phi
\tag{8}
$$

Finally, the optimal transformation matrix $\mathbf{A}$ can be obtained by solving Eq. (8) and finding its d smallest eigenvectors.

Estimation of μ: Note that μ is technically not a free parameter like λ and it has to be estimated according to data distributions. However, there is no effective solution for its estimation. For now, we evaluate the performance of μ by searching its values in experiments. For real application, we recommend getting the optimal μ through cross-validation.

C. Weighted Balanced Distribution Adaptation

BDA is able to adaptively leverage the importance of marginal and conditional distributions between domains. BDA indicates when the marginal distributions are relatively close, the performance of transfer learning is highly dependent on the conditional distribution distance. When computing the conditional distributions, BDA utilizes class conditional distributions instead, i.e. $P(\mathbf{x}|y)$ is used to approximate $P(y|\mathbf{x})$. This implicitly assumes that the probability of this class in each domain is similar, which is usually not the case in real world. In this section, we propose a more robust approximation of the conditional distribution for class imbalance problem:

$$
\begin{aligned}
& \|P(y_s|\mathbf{x}_s) - P(y_t|\mathbf{x}_t)\|_{\mathcal{H}}^2 \\
= & \left\| \frac{P(y_s)}{P(\mathbf{x}_s)} P(\mathbf{x}_s|y_s) - \frac{P(y_t)}{P(\mathbf{x}_t)} P(\mathbf{x}_t|y_t) \right\|_{\mathcal{H}}^2 \\
= & \|\alpha_s P(\mathbf{x}_s|y_s) - \alpha_t P(\mathbf{x}_t|y_t)\|_{\mathcal{H}}^2
\end{aligned}
\tag{9}
$$

Technically, we approximate α_s and α_t by the class prior of both domains. To this end, weighted balanced distribution adaptation (W-BDA) is proposed to balance the class proportion of each domain. Evaluating the conditional distribution divergence in Eq. (9) requires to estimate the marginal distributions $P(\mathbf{x}_s)$ and $P(\mathbf{x}_t)$. However, it is non-trivial. Since BDA is fully capable of adapting $P(\mathbf{x}_s)$ and $P(\mathbf{x}_t)$, we do not estimate them in this step and assume they are unchanged. Then, we construct a weight matrix $\mathbf{W}_c$ for each class:

$$(\mathbf{W}_c)_{ij} = \begin{cases} \frac{P\left(y_s^{(c)}\right)}{n_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \\ \frac{P\left(y_t^{(c)}\right)}{m_c^2}, & \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ -\frac{\sqrt{P\left(y_s^{(c)}\right)P\left(y_t^{(c)}\right)}}{m_c n_c}, & \begin{cases} \mathbf{x}_i \in \mathcal{D}_s^{(c)}, \mathbf{x}_j \in \mathcal{D}_t^{(c)} \\ \mathbf{x}_i \in \mathcal{D}_t^{(c)}, \mathbf{x}_j \in \mathcal{D}_s^{(c)} \end{cases} \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where $P\left(y_s^{(c)}\right)$ and $P\left(y_t^{(c)}\right)$ denote the class prior on class c in the source and target domain, respectively.

Embedding Eq. (10) into BDA, we get the trace optimization problem of W-BDA:

$$\begin{aligned} &\min \ \operatorname{tr}\left(\mathbf{A}^\top \mathbf{X}\left((1-\mu)\mathbf{M}_0 + \mu \sum_{c=1}^{C} \mathbf{W}_c\right)\mathbf{X}^\top \mathbf{A}\right) + \lambda \|\mathbf{A}\|_F^2 \\ &\text{s.t. } \mathbf{A}^\top \mathbf{X}\mathbf{H}\mathbf{X}^\top \mathbf{A} = \mathbf{I}, \quad 0 \le \mu \le 1 \end{aligned} \tag{11}$$

Remark: Eq. (5) of BDA and Eq. (10) of W-BDA are much similar in spirit. Their differences are: 1) Eq. (5) of BDA only considers the number of samples in each class, while Eq. (10) also considers the class prior. 2) Eq. (10) provides more accurate approximation to the conditional distributions than Eq. (5) when handling the class imbalance.

Kernelization: When applied to nonlinear problems, we can use a kernel map ψ: $\mathbf{x} \mapsto \psi(\mathbf{x})$, and a kernel matrix $\mathbf{K} = \psi(\mathbf{X})^\top \psi(\mathbf{X})$. The kernel matrix $\mathbf{K} \in \mathbb{R}^{(n+m)\times(n+m)}$ can be constructed using linear or RBF kernel.

In summary, Algorithm 1 presents the detail of BDA and W-BDA methods.

Algorithm 1 BDA: Balanced Distribution Adaptation

```
Input: Source and target feature matrix $\mathbf{X}_s$ and $\mathbf{X}_t$, source label vector $\mathbf{y}_s$, #dimension $d$, balance factor $\mu$, regularization parameter $\lambda$
Output: Transformation matrix $\mathbf{A}$ and classifier $f$
Train a base classifier on $\mathbf{X}_s$ and apply prediction on $\mathbf{X}_t$ to get its soft labels $\hat{\mathbf{y}}_t$. Construct $\mathbf{X} = [\mathbf{X}_s, \mathbf{X}_\mathbf{t}]$, initialize $\mathbf{M}_0$ and $\mathbf{M}_c$ by Eq. (5) and (6) (or $\mathbf{W}_c$ using Eq. (10) for W-BDA)
repeat
    Solve the eigendecomposition problem in Eq. (8) (or Eq. (11) for W-BDA) and use $d$ smallest eigenvectors to build $\mathbf{A}$
    Train a classifier $f$ on $\{\mathbf{A}^\top \mathbf{X}_s, \mathbf{y}_s\}$
    Update the soft labels of $\mathcal{D}_t$: $\hat{\mathbf{y}}_t = f(\mathbf{A}^\top \mathbf{X}_t)$
    Update matrix $\mathbf{M}_c$ using Eq. (6) (or update $\mathbf{W}_c$ using Eq. (10) for W-BDA)
until Convergence
return Classifier $f$
```

IV. Experiments

In this section, we evaluate the performance of the proposed methods through extensive experiments.

A. Datasets

We adopt five widely-used datasets: USPS + MNIST, COIL20 and Office + Caltech. Table I shows the details of the datasets. **USPS** (U) and **MNIST** (M) are standard digit recognition datasets containing handwritten digits from 0-9. USPS consists of 7,291 training images and 2,007 test images. MNIST contains 60,000 training images and 10,000 test images. **COIL20** (CO) includes 1,440 images belonging to 20 objects. **Office** is composed of three real-world object domains: **Amazon**, **Webcam** and **DSLR**. It has 4,652 images with 31 object categories. **Caltech-256** (C) contains 30,607 images and 256 categories. Detailed descriptions about those datasets can be found in [2]. For all the datasets, we follow [2] to construct 16 different tasks.

Table I
Introduction of the five digit/object datasets.

Dataset	Type	#Sample	#Feature	#Class	Domain
USPS	Digit	1,800	256	10	U
MNIST	Digit	2,000	256	10	M
COIL20	Object	1,440	1,024	20	CO1, CO2
Office	Object	1,410	800	10	A, W, D
Caltech	Object	1,123	800	10	C

B. Comparison Methods

We choose six state-of-the-art comparison methods:

- 1 Nearest Neighbor classifier (1NN)
- Principal Component Analysis (PCA) + 1NN
- Geodesic Flow Kernel (GFK) [19] + 1NN
- Transfer Component Analysis (TCA) [5] + 1NN
- Joint Distribution Adaptation (JDA) [2] + 1NN
- Transfer Subspace Learning (TSL) [20] + 1NN

Among those methods, 1NN and PCA are traditional learning methods, while GFK, TCA, JDA, and TSL are state-of-the-art transfer learning approaches.

C. Implementation Details

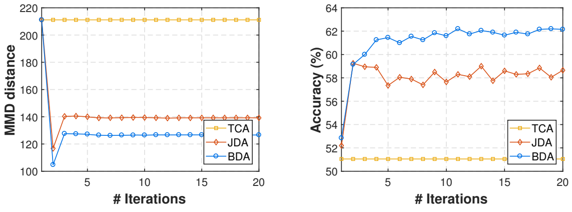
PCA, TCA, JDA, TSL, and BDA are acting as dimensionality reduction process, then 1NN is applied. For GFK, 1NN is applied after we get the geodesic flow kernel. For BDA and W-BDA, μ is searched in $\{0, 0.1, \cdots, 0.9, 1.0\}$. Since BDA can achieve a stable performance under a wide range of parameter values, for the comparison study, we set $d = 100$; $\lambda = 0.1$ for MNIST + USPS / Office + Caltech datasets and $\lambda = 0.01$ for COIL20 dataset. For the kernel-based methods, we use linear kernel. The iteration number for JDA and TCA is set to be $T = 10$. The codes of BDA and W-BDA are available online[1]. Classification *accuracy* on target domain is adopted as the evaluation metric which is widely used in literatures [2], [3].

[1] Code available at http://tinyurl.com/yd3ol4om

Table II

Accuracy (%) of BDA and other methods on 16 tasks.

Task	1NN	PCA	GFK	TCA	JDA	TSL	BDA
U → M	44.70	44.95	46.45	52.20	57.45	53.75	**59.35**
M → U	65.94	66.22	67.22	54.28	62.89	66.06	**69.78**
CO1 → CO2	83.61	84.72	72.50	88.61	97.22	88.06	**97.22**
CO2 → CO1	82.78	84.03	74.17	96.25	86.39	87.92	**96.81**
C → A	23.70	36.95	41.02	44.89	42.90	44.47	**44.89**
C → W	25.76	32.54	40.68	36.61	38.64	34.24	**38.64**
C → D	25.48	38.22	38.85	45.86	47.13	43.31	**47.77**
A → C	26.00	34.73	40.25	40.78	38.82	37.58	**40.78**
A → W	29.83	35.59	38.98	37.63	37.29	33.90	**39.32**
A → D	25.48	27.39	36.31	31.85	40.13	26.11	**43.31**
W → C	19.86	26.36	30.72	27.16	25.29	**29.83**	28.94
W → A	22.96	31.00	29.75	30.69	31.84	30.27	**32.99**
W → D	59.24	77.07	80.89	90.45	90.45	87.26	**91.72**
D → C	26.27	29.65	30.28	32.50	30.99	28.50	**32.50**
D → A	28.50	32.05	32.05	31.52	32.25	27.56	**33.09**
D → W	63.39	75.93	75.59	87.12	91.19	85.42	**91.86**
Average	40.84	47.34	48.48	51.78	53.18	50.27	**55.56**



(a) MMD distance

(b) Accuracy

Figure 1. MMD distance and classification accuracy comparison of TCA, JDA and BDA on U → M. It can be noted that BDA achieves better accuracy with relatively small MMD distance.

D. Performance Evaluation of BDA

1) Classification accuracy: We test the performance of BDA and the other comparison methods on 16 cross-domain learning tasks. The results are shown in Table II, based on which, we can draw the following observations.

First, BDA outperforms most of the existing methods (15 out of 16 tasks). Specifically, the average classification accuracy of BDA is **55.56%**, which shows an average improvement of 2.38% compared to the best comparison method JDA. JDA is only capable of adapting the marginal and conditional distribution with the equal weight ($\mu = 0.5$). Thus JDA can be considered as a special case of BDA. However, BDA can dramatically improve the accuracy by adjusting the balance parameter μ to adapt various scenarios.

Second, TCA is also a special case of BDA ($\mu = 0$) since it only adapts the marginal distribution. Therefore, the performance of TCA is worse than JDA and BDA.

Third, TSL only adapts the marginal distributions which highly relies on the distribution density. The performance of GFK is better on object recognition tasks. The reason is that GFK learns a global geodesic flow kernel on the low-dimension representation, which may be enough to transit smoothly for the object datasets. But as for the digit tasks, it may not enough to construct smooth transit when the marginal distribution distance is large.

Last, all transfer learning methods perform better than traditional learning approaches due to the large distribution gap between different domains. This indicates the effectiveness of transfer learning methods, among which BDA could achieve the best performance.

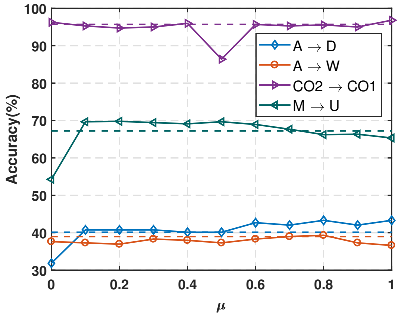


Figure 2. Classification accuracy w.r.t. μ on different tasks. Dashed lines are the best comparison methods.

2) Effectiveness of distribution adaptation: We further verify the effectiveness of BDA by comparing its distribution adaptation with other two distribution adaptation methods: TCA and JDA. Specifically, we investigate their performance with MMD distances calculated using Eq. (4).

Fig. 1(a) and Fig. 1(b) show the MMD distance and accuracy of TCA, JDA, and BDA with increasing iteration, respectively. Based on the results, we can observe: a) MMD distances of all methods can be reduced. This indicates the effectiveness of TCA, JDA and BDA; b) MMD distance of TCA is not reduced largely as it only adapts the marginal distribution distance and requires no iteration; c) MMD distance of JDA is obviously larger than BDA, since BDA could balance the importance of marginal and conditional distribution via μ; d) BDA achieves the best performance.

E. Effectiveness of Balance Factor

In this section, we evaluate the effectiveness of the balance factor μ. We run BDA with $\mu \in \{0, 0.1, \cdots, 1.0\}$ on some tasks and compare the performances with the best baseline method. Fig. 2 shows the results. It is obvious that the optimal μ varies on different tasks, indicating the importance to balance the marginal and conditional distributions between domains. In comparison, the best baseline JDA (dash lines) is only the special case of BDA ($\mu = 0.5$), which means to treat those distributions equally. However, this assumption does not hold. In task A → W with optimal $\mu = 0.8$, it means the marginal distributions are almost the same so the performance of transfer learning mostly depends on conditional distributions. In task M → U with optimal $\mu = 0.1$, it means the marginal distributions contribute most to the discrepancy, so μ is relatively small. In other 13 tasks, the observations are similar. It indicates in cross-domain learning problems, μ is extremely important to balance both the marginal and conditional distributions. Therefore, BDA is more capable of achieving good performance.

To be noticed, there may be more than one optimal μ for some tasks (A → D), and the tendency of μ is not always stable (CO2 → CO1). The problems behind those facts still need to be addressed in future research.

F. Effectiveness of Weighted BDA

We extensively verify the effectiveness of proposed W-BDA. We choose some tasks with highly imbalanced class distributions and compare the performance of W-BDA with BDA and JDA. TABLE. III demonstrates the classification accuracy of 6 tasks. Note that for comparison, classes on tasks $1 \sim 4$ are rather imbalanced, while classes are rather balanced on tasks $5 \sim 6$.

Table III
ACCURACY OF JDA, BDA AND W-BDA ON SOME TASKS.

Index	Task	JDA	BDA	WBDA
1	C → D	47.13	47.77	**48.41**
2	W → C	25.29	28.94	**31.08**
3	U → M	57.45	59.15	**59.35**
4	A → W	37.29	39.32	**40.68**
5	C → A	42.90	44.89	**45.20**
6	CO1 → CO2	97.22	**97.22**	96.69

From the results, we can observe: 1) JDA achieves the worst results since it does not consider the gap between marginal and conditional distributions. BDA is able to handle the distribution discrepancy and outperforms JDA in all situations. 2) For the first four tasks which are under imbalanced class distributions, W-BDA could improve the performance by adaptively weighting each class. In the other two tasks where class distributions are rather balanced, W-BDA still achieves comparable results. On the other 10 tasks, the results follow the same tendency. To sum up, the results demonstrate that in transfer learning, W-BDA remains an effective method to balance the different class distribution between domains.

In addition, BDA and W-BDA have other two parameters: feature dimension d and regularization parameter λ. Their sensitivity evaluations are omitted due to page limit. In our actual experiments, BDA and W-BDA are relatively robust to those two parameters.

V. CONCLUSION

Balancing the probability distributions and class distributions between domains are both two important problems in transfer learning. In this paper, we propose Balanced Distribution Adaptation (BDA) to adaptively weight the importance of both marginal and conditional distribution adaptations. Thus, it could significantly improve the transfer learning performance. Moreover, we consider handling the class imbalance problem for transfer learning by proposing Weighted BDA (W-BDA). Extensive experiments on five image datasets demonstrate the superiority of our methods over several state-of-the-art methods. In the future, we will continue the exploration in these two aspects: by developing more strategies to leverage the distributions and handle the class imbalance problem.

ACKNOWLEDGMENT

This work is supported in part by National Key R & D Plan of China (No.2016YFB1001200), NSFC (No.61572471), and Beijing Municipal Science & Technology Commission (No.Z161100000216140 & Z171100000117013).

REFERENCES

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *TKDE*, vol. 22, no. 10, pp. 1345–1359, 2010.

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

[3] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo, "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation," in *CVPR*, 2017.

[4] C.-A. Hou, Y.-H. H. Tsai, Y.-R. Yeh, and Y.-C. F. Wang, "Unsupervised domain adaptation with label and structural consistency," *TIP*, vol. 25, no. 12, pp. 5552–5562, 2016.

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

[6] S. Satpal and S. Sarawagi, "Domain adaptation of conditional probability models via feature subsetting," in *PKDD*. Springer, 2007, pp. 224–235.

[7] J. Tahmoresnezhad and S. Hashemi, "Visual domain adaptation via transfer feature learning," *Knowl. Inf. Syst.*, 2016.

[8] J. Wang, Y. Chen, S. Hao, X. Peng, and L. Hu, "Deep learning for sensor-based activity recognition: A survey," *arXiv preprint arXiv:1707.03502*, 2017.

[9] L. Hu, Y. Chen, S. Wang, J. Wang, J. Shen, X. Jiang, and Z. Shen, "Less annotation on personalized activity recognition using context data," in *UIC*, July 2016, pp. 327–332.

[10] S. Hao, P. Zhao, Y. Liu, S. C. H. Hoi, and C. Miao, "Online multi-task relative similarity learning," in *IJCAI*, 2017.

[11] Y. Chen, Y. Gu, X. Jiang, and J. Wang, "Ocean: A new opportunistic computing model for wearable activity recognition," in *UbiComp*. ACM, 2016, pp. 33–36.

[12] J. Li, J. Zhao, and K. Lu, "Joint feature selection and structure preservation for domain adaptation," in *IJCAI*, 2016.

[13] C.-A. Hou, Y.-R. Yeh, and Y.-C. F. Wang, "An unsupervised domain adaptation approach for cross-domain visual classification," in *AVSS*. IEEE, 2015, pp. 1–6.

[14] S. Ando and C. Y. Huang, "Deep over-sampling framework for classifying imbalanced data," *arXiv preprint arXiv:1704.07515*, 2017.

[15] T. Ming Harry Hsu, W. Yu Chen, C.-A. Hou, and H. T. et al., "Unsupervised domain adaptation with imbalanced cross-domain data," in *ICCV*, 2015, pp. 4121–4129.

[16] F. Wu, X.-Y. Jing, S. Shan, W. Zuo, and J.-Y. Yang, "Multiset feature learning for highly imbalanced data classification," in *AAAI*, 2017.

[17] P.-H. Hsiao, F.-J. Chang, and Y.-Y. Lin, "Learning discriminatively reconstructed source data for object recognition with few examples," *TIP*, vol. 25, no. 8, pp. 3518–3532, 2016.

[18] S. Li, S. Song, and G. Huang, "Prediction reweighting for domain adaptation," *TNNLS*, no. 99, pp. 1–14, 2016.

[19] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *CVPR*, 2012.

[20] S. Si, D. Tao, and B. Geng, "Bregman divergence-based regularization for transfer subspace learning," *TKDE*, vol. 22, no. 7, pp. 929–942, 2010.