Unsupervised Domain Adaptation with Regularized Domain Instance Denoising

Gabriela Csurka, Boris Chidlowskii, Stéphane Clinchant and Sophia Michel

Xerox Research Center Europe
6 chemin de Maupertuis, 38240 Meylan, France
Firstname.Lastname@xrce.xerox.com

Abstract. We propose to extend the marginalized denoising autoencoder (MDA) framework with a domain regularization whose aim is to denoise both the source and target data in such a way that the features become domain invariant and the adaptation gets easier. The domain regularization, based either on the maximum mean discrepancy (MMD) measure or on the domain prediction, aims to reduce the distance between the source and the target data. We also exploit the source class labels as another way to regularize the loss, by using a domain classifier regularizer. We show that in these cases, the noise marginalization gets reduced to solving either the linear matrix system $\mathbf{AX} = \mathbf{B}$, for which there exists a closed-form solution, or to a Sylvester linear matrix equation $\mathbf{AX} + \mathbf{XB} = \mathbf{C}$ that can be solved efficiently using the Bartels-Stewart algorithm. We did an extensive study on how these regularization terms improve the baseline performance and we present experiments on three image benchmark datasets, conventionally used for domain adaptation methods. We report our findings and comparisons with state-of-the-art methods.

Keywords: Unsupervised Domain Adaptation, Marginalized Denoising Autoencoder, Sylvester equation, Domain regularization

1 Introduction

Domain Adaptation problems arise each time we need to leverage labeled data in one or more related *source* domains, to learn a classifier for unseen or unlabeled data in a *target* domain. The domains are assumed to be related, but not identical. The underlying *domain shift* occurs in multiple real-world applications. Numerous approaches have been proposed in the last years to address textual and visual domain adaptation (we refer the reader to [32, 23, 36] for recent surveys on transfer learning and domain adaptation methods). For text data, the domain shift is frequent in named entity recognition, statistical machine translation, opinion mining, speech tagging and document ranking [11, 33, 3, 41]. Domain adaptation has equally received a lot of attention in computer vision [14, 34, 20, 13, 22, 21, 17, 29, 15, 1, 35] where domain shift is a consequence of changing conditions, such as background, location or pose, or considering different image types, such as photos, paintings, sketches [25, 9, 4].

In this paper, we build on an approach to domain adaptation based on noise marginalization [5]. In deep learning, a denoising autoencoder (DA) learns a robust feature rep-

resentation from training examples. In the case of domain adaptation, it takes the unlabeled instances of both source and target data and learns a new feature representation by reconstructing the original features from their noised counterparts. A *marginalized denoising autoencoder* (MDA) is a technique to marginalize the noise at training time; it avoids the explicit data corruption and does not require an optimization procedure for learning the model parameters but computes the model in a closed form. This makes MDAs scalable and computationally faster than the regular denoising autoencoders. The principle of noise marginalization has been successfully extended to learning with corrupted features [30], link prediction and multi-label learning [6], relational learning [7], collaborative filtering [26] and heterogeneous cross-domain learning [40, 27].

The *marginalized domain adaptation* refers to such a denoising of source and target instances that explicitly makes their features *domain invariant*. To achieve this goal, we extend the MDA with a domain regularization term. We explore three ways of such a regularization. The first way uses the *maximum mean discrepancy* (MMD) measure [24]. The second way is inspired by the adversarial learning of deep neural networks [19]. The third regularization term is based on preserving accurate classification of the denoised source instances. In all cases, the regularization term belongs to the class of squared loss functions. This guarantees the noise marginalization and the computational efficiency, either as a closed form solution or as a solution of Sylvester linear matrix equation $\mathbf{AX} + \mathbf{XB} = \mathbf{C}$.

2 Feature Denoising for Domain Adaptation

Let $\mathbf{X}^s = [\mathbf{X}_1, \dots, \mathbf{X}_{n_S}]$ denote a set of n_S source domains, with the corresponding labels $\mathbf{Y}^s = [\mathbf{Y}_1, \dots \mathbf{Y}_{n_S}]$, and let $\mathbf{X}^t$ denote the unlabeled target domain data. The *Marginalized Denoising Autoencoder* (MDA) approach [5] is to reconstruct the input data from partial random corruption [39] with a marginalization that yields optimal reconstruction weights $\mathbf{W}$ in a closed form. The MDA minimizes the loss written as:

$$\mathcal{L}(\mathbf{W}, \mathbf{X}) = \frac{1}{K} \sum_{k=1}^{K} \|\mathbf{X} - \tilde{\mathbf{X}}_k \mathbf{W}\|^2 + \omega \|\mathbf{W}\|^2, \tag{1}$$

where $\tilde{\mathbf{X}}_k \in \mathbb{R}^{N \times d}$ is the k-th corrupted version of $\mathbf{X} = [\mathbf{X}^s, \mathbf{X}^t]$ by random feature dropout with a probability p, $\mathbf{W} \in \mathbb{R}^{d \times d}$, and $\omega \|\mathbf{W}\|^{\mathbf{2}}$ is a regularization term. To avoid the explicit feature corruption and an iterative optimization, Chen *et al.* [5] has shown that in the limiting case $K \to \infty$, the weak law of large numbers allows to rewrite $\mathcal{L}(\mathbf{W}, \mathbf{X})$ as its expectation. The optimal solution $\mathbf{W}$ can be written as $\mathbf{W} = (\mathbf{Q} + \omega \mathbf{I}_d)^{-1} \mathbf{P}$, where $\mathbf{P} = \mathbb{E}[\mathbf{X}^\top \tilde{\mathbf{X}}]$ and $\mathbf{Q} = \mathbb{E}[\tilde{\mathbf{X}}^\top \tilde{\mathbf{X}}]$ depend only on the covariance matrix $\mathbf{S}$ of the uncorrupted data, $\mathbf{S} = \mathbf{X}^\top \mathbf{X}$, and the noise level p:

$$\mathbf{P} = (1-p)\mathbf{S} \quad \text{and} \quad \mathbf{Q}_{ij} = \begin{cases} \mathbf{S}_{ij}(1-p)^2, & \text{if} \quad i \neq j, \\ \mathbf{S}_{ij}(1-p), & \text{if} \quad i = j. \end{cases} \tag{2}$$

2.1 Domain regularization

To better address the domain adaptation, we extend the feature denoising with a *domain regularization* in order to favor the learning of domain invariant features. We explore

three versions of the domain regularization. We combine them with the loss (1) and show how to marginalize the noise for each version and to keep $\mathbf{W}$ as a solution of a linear matrix equation. The three versions of the domain regularization are as follows:

Regularization $\mathcal{R}_m$ based on the maximum mean discrepancy (MMD) with the linear kernel; it aims at reducing the gap between the denoised domain means. The MMD was already used for domain adaptation with feature transformation learning [31, 2] and as a regularizer for the cross-domain classifier learning [13, 38, 28]. In this paper, in contrast to these papers where the distributions are approximated with MMD using multiple nonlinear kernels we use MMD with the linear kernel[1], the only one allowing us to keep the solution for $\mathbf{W}$ closed form.

The regularization term for K corrupted versions of $\mathbf{X}$ is given by:

$$\mathcal{R}_m = \frac{1}{K}\sum_{k=1}^{K} Tr(\mathbf{W}^\top \tilde{\mathbf{X}}_k^\top \mathbf{N} \tilde{\mathbf{X}}_k \mathbf{W}), \quad \text{where} \quad \mathbf{N} = \begin{bmatrix} \frac{1}{N_s^2}\mathbf{1}^{s,s} & \frac{1}{N_s N_t}\mathbf{1}^{s,t} \\ \frac{1}{N_s N_t}\mathbf{1}^{s,t} & \frac{1}{N_t^2}\mathbf{1}^{t,t} \end{bmatrix},$$

$\mathbf{1}^{a,b}$ is a constant matrix of size $N_a \times N_b$ with all elements being equal to 1 and N_s, N_t are the number of source and target examples. After the noise marginalization, we obtain $\mathbb{E}[\mathcal{R}_m] = Tr(\mathbf{W}^\top \mathbf{M} \mathbf{W})$, where $\mathbf{M} = \mathbb{E}[\tilde{\mathbf{X}}^\top \mathbf{N} \tilde{\mathbf{X}}]$ is computed similarly to $\mathbf{Q}$ in (2), by using $\mathbf{S}_m = \mathbf{X}^\top \mathbf{N} \mathbf{X}$ instead of the correlation matrix $\mathbf{S}$.

Regularization $\mathcal{R}_d$ based on domain prediction; it explicitly pushes the denoised source examples toward target instances. The domain regularizer $\mathcal{R}_d$, proposed in [8], is inspired by [18] where intermediate layers in a deep learning model are regularized using a domain prediction task. The main idea is to learn the denoising while pushing the source towards the target (or *vice versa*) and hence allowing the source classifier to perform better on the target. The regularization term $\mathcal{R}_d$ can be written as follows:

$$\mathcal{R}_d = \frac{1}{K}\sum_{k=1}^{K} \|\mathbf{Y}_\mathcal{T} - \tilde{\mathbf{X}}_k \mathbf{W} \mathbf{Z}_\mathcal{D}\|^2, \tag{3}$$

where $\mathbf{Z}_\mathcal{D} \in \mathbb{R}^d$ is a domain classifier trained on the uncorrupted data to distinguish the target from the source and $\mathbf{Y}_\mathcal{T} = \mathbf{1}^N$ is a vector containing only ones, as all denoised instances should look like the target[2]. After the noise marginalization, the partial derivatives on $\mathbf{W}$ of this term expectation are the following:

$$\frac{\partial\, \mathbb{E}[\mathcal{R}_d]}{\partial \mathbf{W}} = -2(1-p)\mathbf{X}^\top \mathbf{Y}_\mathcal{T} \mathbf{Z}_\mathcal{D} + 2\mathbf{Q}\mathbf{W}\mathbf{Z}_\mathcal{D}\mathbf{Z}_\mathcal{D}^\top.$$

Classification regularization $\mathcal{R}_l$; it encourages the denoised source data to remain well classified by the classifier pre-trained on source data. The regularizer $\mathcal{R}_l$ is similar to $\mathcal{R}_d$, except that $\mathbf{Z}_l$ is trained on the uncorrupted source $\mathbf{X}^s$ and acts only on the labeled

[1] Minimizing the distance between the corresponding domain centroids.

[2] In the multi source case, $\mathbf{Z}_\mathcal{D} \in \mathbb{R}^{d\times(n_S+1)}$, with the columns corresponding n_S sources and 1 target domain classifiers, and $\mathbf{Y}_\mathcal{T} \in \mathbb{R}^{N\times(n_S+1)}$, with $y_{ns} = 1$ if $s = n_S + 1$ and -1 otherwise. N is the total number of instances (source and target).

Table 1. A summary of our models and corresponding notations.

Method	Loss	$\mathbf{W}$ closed form solution
M1	$\mathcal{L}$	$(\mathbf{Q}+\omega\mathbf{I}_d)^{-1}\mathbf{P}$
MRm	$\mathcal{L}+\gamma_m\mathcal{R}_m$	$(\mathbf{Q}+\omega\mathbf{I}_d+\gamma_m\mathbf{M})^{-1}\mathbf{P}$
	Loss	$\mathbf{W}$ solution of $\mathbf{AW}+\mathbf{WB}=\mathbf{C}$
MRd	$\mathcal{L}+\gamma_d\mathcal{R}_d$	$\mathbf{A}=\omega\mathbf{Q}^{-1}$, $\mathbf{B}=(\mathbf{I}_d+\gamma_d\mathbf{Z}_\mathcal{D}\mathbf{Z}_\mathcal{D}^\top)$ $\mathbf{C}=\mathbf{Q}^{-1}(\mathbf{P}+\gamma_d(1-p)\mathbf{X}^\top\mathbf{Y}_\mathcal{T}\mathbf{Z}_\mathcal{D}^\top)$
MRl	$\mathcal{L}+\gamma_l\mathcal{R}_l$	$\mathbf{A}=\mathbf{Q}_l^{-1}(\mathbf{Q}+\omega\mathbf{I}_d)$, $\mathbf{B}=\gamma_l\mathbf{Z}_l\mathbf{Z}_l^\top$ $\mathbf{C}=\mathbf{Q}_l^{-1}(\mathbf{P}+\gamma_l(1-p)\mathbf{X}_l^\top\mathbf{Y}_l\mathbf{Z}_l^\top)$

source data. Also, instead of $\mathbf{Y}_\mathcal{T}$, the groundtruth source labels $\mathbf{Y}_l=\mathbf{Y}^s$ are used[3]. In the marginalized version of $\mathcal{R}_l$, The partial derivatives on $\mathbf{W}$ can be written as

$$\frac{\partial\,\mathbb{E}[\mathcal{R}_l]}{\partial\mathbf{W}}=-2(1-p)\mathbf{X}_l^\top\mathbf{Y}_l\mathbf{Z}_l+2\mathbf{Q}_l\mathbf{W}\mathbf{Z}_l\mathbf{Z}_l^\top,$$

where $\mathbf{X}_l=\mathbf{X}^s$ and $\mathbf{Q}_l$ is computed similarly to $\mathbf{Q}$, with $\mathbf{S}_l=\mathbf{X}_l^\top\mathbf{X}_l$.

2.2 Minimizing the regularized loss

We extend the noise marginalization framework for optimizing the data reconstruction loss (1) and minimize the expected loss $\mathbb{E}[\mathcal{L}+\gamma_\phi\mathcal{R}_\phi]$, denoted $\mathbb{E}[\mathcal{L}_\phi]$, where in the regularization term $\mathcal{R}_\phi$, ϕ refers to m, d or l version. From the marginalized terms presented in the previous sections, it is easy to show that when minimizing these regularized losses, the optimal solution for $\mathbf{W}$ given by $\partial\,\mathbb{E}[\mathcal{L}_\phi]/\partial\mathbf{W}=\mathbf{0}$ can be reduced to solving the linear matrix system $\mathbf{AW}=\mathbf{B}$, for which there exists a closed-form solution, or to a Sylvester linear matrix equation $\mathbf{AW}+\mathbf{WB}=\mathbf{C}$ that can be solved efficiently using the Bartels-Stewart algorithm. Due to the limited space, we report all the details in the full version and summarize the baseline, three extensions and the corresponding solutions in Table 1.

Similarly to the stacked MDAs, we can stack several layers together with only forward learning, where the denoised features of the previous layer serve as the input to the next layer and nonlinear functions such as tangent hyperbolic or rectified linear units can be applied between the layers.

3 Experimental Results

Datasets. We run experiments on the popular **OFF31** [34] and **OC10** [22] datasets, both with the *full training* protocol [21] where all source data is used for training and with the *sampling* protocol [34, 22]. We evaluated our models both with the provided SURFBOV and the DECAF6 [12] features. In addition we run experiments with the

[3] $\mathbf{Y}_l\in\mathbb{R}^{N_s\times C}$, where $y_{nc}=1$ if $\mathbf{x}_n$ belongs to the class c and -1 otherwise. In the multi source case, we concatenate n_S multi-class $\mathbf{Z}_l^a$ linear classifiers and the corresponding $\mathbf{Y}_l^a$ label matrices, where $\mathbf{Z}_l^a$ was trained on the source $\mathcal{D}^{s_a}$.

Table 2. Single source domain adaptation with a single ($r = 1$) and 3 stacked layers ($r = 3$). Bold indicates the best result per column, underline refers to best single layer results.

DECAF	**OC10**			**OFF31**			**TB**		
	NN	DSCM	Ridge	NN	DSCM	Ridge	NN	DSCM	Ridge
BL (full)	84.5	78.7	82.6	65.2	61.6	62.8	39.8	42.6	37.2
BL (PCA)	84.1	81.8	82.5	65.4	63.7	62.4	40.9	42.7	39.7
M1 ($r = 1$)	84.1	82.0	83.6	65.3	63.6	64.4	41.0	42.8	40.6
MRm ($r = 1$)	84.1	82.1	83.6	65.4	63.6	64.4	41.0	42.8	40.6
MRd ($r = 1$)	84.4	<u>82.9</u>	<u>83.7</u>	65.7	64.0	64.7	41.1	43.2	40.6
MRl ($r = 1$)	<u>84.5</u>	82.2	82.2	<u>66.9</u>	<u>65.6</u>	<u>65.6</u>	<u>41.3</u>	<u>43.3</u>	<u>40.9</u>
M1 ($r = 3$)	84.3	82.4	84.0	64.7	63.6	65.6	41.2	42.6	41.3
MRm ($r = 3$)	84.3	82.4	84.0	64.7	63.6	65.6	41.2	42.6	41.3
MRd ($r = 3$)	**84.8**	**83.9**	**84.9**	66.0	64.7	66.0	41.4	43.8	41.2
MRl ($r = 3$)	84.1	82.2	81.8	**67.7**	**65.9**	**66.5**	**41.8**	**43.9**	**41.5**

full training protocol on the Testbed Cross-Dataset [37] (**TB**) using both the provided SIFTBOV and the DECAF7 features.

Parameter setting. To compare different models we run all experiments with the same preprocessing and parameter values[4] . Features are L2 normalized and the feature dimensionality is PCA reduced to 200 (BOV features are in addition power normalized). Parameter values are $\omega = 0.01$, $\gamma_\phi = 1$ and $p = 0.1$. Between layers we apply tangent hyperbolic nonlinearities and we concatenate the outputs of all layers with the original features (as in [5]).

We evaluate how the optimal denoising matrix $\mathbf{W}$ influences three different classification methods, a regularized multi-class ridge classifier trained on the source ($\mathbf{Z} = (\mathbf{X}_l^\top \mathbf{X}_l + \delta \mathbf{I}_d)^{-1} \mathbf{X}_l^\top \mathbf{Y}_l$), the nearest neighbor classifier (NN) and the Domain Specific Class Means (DSCM) classifier [10] where a target test example is assigned to a class based on a soft-max distance to the domain specific class means. Two last classifiers are selected for their non-linearity. Also the NN is related to retrieval and DSCM to clustering, so the impact of $\mathbf{W}$ on these two extra tasks is indirectly assessed.

Table 2 shows the domain adaptation results with a single source and Table 3 shows multi source results, both under the full training protocol. For each dataset, we consider all possible source-target pairs for the domain adaptation tasks. Hence we average over 9 tasks on **OFF31** (with 3 domains A,D,W), and over 12 tasks on **OC10** (4 domains (A,C,D,W) and **TB** (4 domains B,C,I,S).

Table 2 shows the results on L2 normalized DECAF features. It compares the domain regularization extensions to the baselines (BL) obtained with the L2 normalized features (full) and with the PCA reduced features as well as with MDA. As the table shows, the best results are often obtained with **MRl**, except in the OC10 case where **MRd** performs better. On the other hand, the $\mathcal{R}_m$ regularizer (**MRm**) does not improve the **M1** performance. Stacking several layers can further improve the results. When comparing these results to the literature we can see that on **OC10** we perform comparably to DAM [14] (84%) and DDC [38] (84.6%) but worse than more complex methods

[4] Cross validation on the source was only helpful for some of the configurations, for others it yielded performance decrease.

Table 3. Multi-source adaptation results without stacking. Bold indicates best result per column.

BOV	**OC10**			**OFF31**			**TB**		
	NN	DSCM	Ridge	NN	DSCM	Ridge	NN	DSCN	Ridge
BL	50.4	54.6	51.9	39.7	33.3	25.4	16.5	17.6	21
M1	50.8	**54.7**	52.1	39.9	33.8	25.6	16.6	17.7	21
MRd	50.8	54.1	51.5	**40.1**	33.5	26.9	16.6	17.7	**21.1**
MRl	**53.8**	53.3	**52.5**	39	**36.5**	**28.5**	**17.1**	**19.6**	20.9

such as JDA [29] (87.5%), TTM [16] (87.5%) or DAN [28] (87.3%). On **OFF31**, the deep adaptation method DAN [28] (72.9%) significantly outperforms our results. On the **TD** dataset, in order to compare our results on DECAF6 to CORAL+SVM [35] (40.2%) we average six source-task pairs (without the domain B) and obtain 43.6% with **MRd**+DSCM and 43.1% with **MRl**+DSCM. We also outperform[5] CORAL+SVM [35] (64%) with our **MRd**+Ridge (65.2%) when using the sampling protocol on **OFF31** .

Concerning the BOV features, the best results (using 3 layers) with the full training protocol on **OFF31** are with **MRl**+NN (29.7%) and on **OC10** with **MRd**+Ridge(48.2%). The latter is comparable to CORAL+SVM [35] (48.8%), but is below LSSA [1] (52.3%) that first selects landmarks before learning the transformation. The landmark selection is complementary to our approach and can boost our results as well.

In Table 3, we report the averaged results for the multi-source cases, obtained with BOV features, under the full training protocol. For each dataset, all the configurations with at least 2 source domains are considered. It yields 6 such configurations for **OFF31** and 16 configurations for **OC10** and **TB**. The results indicate clearly that taking into account the domain regularization improves the performance.

4 Conclusion

In this paper we extended the marginalized denoising autoencoder (MDA) framework with a domain regularization to enforce domain invariance. We studied three versions of regularization, based on the maximum mean discrepancy measure, the domain prediction and the class predictions on source. We showed that in all these cases, the noise marginalization is reduced to closed form solution or to a Sylvester linear matrix system, for which there exist efficient and scalable solutions. This allows furthermore to easily stack several layers with low cost. We studied the effect of these domain regularizations and run single source and multi-source experiments on three benchmark datasets showing that adding the new regularization terms allow to outperform the baselines. Compared to the state-of-the-art, our method performs better than classical feature transformation methods but it is outperformed by more complex deep domain adaptation methods. Compared to the latter methods, the main advantage of the proposed approach, beyond its low computational cost, is that as we learn an unsupervised feature transformation, we can boost the performance of other tasks such as retrieval or clustering in the target space.

[5] Their best results (68.5% and 69.4%) obtained with fine-tuned features are not directly comparable as our results can also be boosted when using these fine-tuned features.

References

1. Aljundi, R., Emonet, R., Muselet, D., Sebban, M.: Landmarks-based kernelized subspace alignment for unsupervised domain adaptation. In: Proc. of CVPR, (IEEE). pp. 56–63 (2015)
2. Baktashmotlagh, M., Harandi, M., Lovell, B., Salzmann, M.: Unsupervised domain adaptation by domain invariant projection. In: Proc. of ICCV, (IEEE). pp. 769–776 (2013)
3. Blitzer, J., Kakade, S., Foster, D.P.: Domain adaptation with coupled subspaces. In: Proc. of AISTATS. pp. 173–181 (2011)
4. Castrejón, L., Aytar, Y., Vondrick, C., Pirsiavash, H., Torralba, A.: Learning aligned cross-modal representations from weakly aligned data. In: Proc. of CVPR, (IEEE) (2016)
5. Chen, M., Xu, Z., Weinberger, K.Q., Sha, F.: Marginalized denoising autoencoders for domain adaptation. In: Proc. of ICML. pp. 767–774 (2012)
6. Chen, Z., Chen, M., Weinberger, K.Q., Zhang, W.: Marginalized denoising for link prediction and multi-label learning. In: Proc. of AAAI (2015)
7. Chen, Z., Zhang, W.: A marginalized denoising method for link prediction in relational data. In: Proc. of ICDM (2014)
8. Clinchant, S., Csurka, G., Chidlovskii, B.: A domain adaptation regularization for denoising autoencoders. In: Proc. of ACL (2016)
9. Crowley, E.J., Zisserman, A.: In search of art. In: Computer Vision for Art analysis, ECCV workshop (2014)
10. Csurka, G., Chidlovskii, B., Perronnin, F.: Domain adaptation with a domain specific class means classifier. In: TASK-CV, ECCV workshop (2014)
11. Daume III, H., Marcu, D.: Domain adaptation for statistical classifiers. Journal of Artificial Intelligence Research 26(1), 101–126 (2006)
12. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. CoRR arXiv:1310.1531 (2013)
13. Duan, L., Tsang, I.W., Xu, D.: Domain transfer multiple kernel learning. Transactions of Pattern Recognition and Machine Analyses (PAMI) 34(3), 465–479 (2012)
14. Duan, L., Tsang, I.W., Xu, D., Chua, T.S.: Domain adaptation from multiple sources via auxiliary classifiers. In: Proc. of ICML. pp. 289–296 (2009)
15. Farajidavar, N., deCampos, T., Kittler, J.: Adaptive transductive transfer machines. In: Proc. of BMVC (2014)
16. Farajidavar, N., deCampos, T., Kittler, J.: Transductive transfer machines. In: Proc. of ACCV. pp. 623–639 (2014)
17. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: Proc. of ICCV, (IEEE). pp. 2960–2967 (2013)
18. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. CoRR arXiv:1409.7495 (2014)
19. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: Proc. of ICML. pp. 1180–1189 (2015)
20. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: A deep learning approach. In: Proc. of ICML. pp. 513–520 (2011)
21. Gong, B., Grauman, K., Sha, F.: Connecting the dots with landmarks: Discriminatively learning domain invariant features for unsupervised domain adaptation. In: Proc. of ICML. pp. 222–230 (2013)
22. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: Proc. of CVPR, (IEEE). pp. 2066–2073 (2012)
23. Gopalan, R., Li, R., Patel, V.M., Chellappa, R.: Domain adaptation for visual recognition. Foundations and Trends in Computer Graphics and Vision 8(4) (2015)

24. Huang, J., Smola, A., Gretton, A., Borgwardt, K., Schölkopf, B.: Correcting sample selection bias by unlabeled data. In: Proc. of NIPS, (Curran Associates) (2007)
25. Klare, B.F., Bucak, S.S., Jain, A.K., Akgul, T.: Towards automated caricature recognition. In: Proc. of ICB (2012)
26. Li, S., Kawale, J., Fu, Y.: Deep collaborative filtering via marginalized denoising auto-encode. In: Proc. of CIKM, (ACM). pp. 811–820 (2015)
27. Li, Y., Yang, M., Xu, Z., Zhang, Z.: Learning with marginalized corrupted features and labels together. In: Proc. of AAAI. vol. arXiv:1602:07332 (2016)
28. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. In: Proc. of ICML (2015)
29. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer feature learning with joint distribution adaptation. In: Proc. of ICCV, (IEEE). pp. 2200–2207 (2013)
30. Maaten, L.v.d., Chen, M., Tyree, S., Weinberger, K.: Learning with marginalized corrupted features. In: Proc. of ICML (2013)
31. Pan, S.J., Tsang, Ivor W.and Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. Transactions on Neural Networks 22(2), 199 – 210 (2011)
32. Pan, S.J., Yang, Q.: A survey on transfer learning. Transactions on Knowledge and Data Engineering 22(10), 1345–1359 (2010)
33. Pan, S.J., Ni, X., Sun, J.T., Yang, Q., Chen, Z.: Cross-domain sentiment classification via spectral feature alignment. In: Proc. of WWW (2010)
34. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: Proc. of ECCV, (Springer). pp. 213–226 (2010)
35. Sun, B., Feng, J., Saenko, K.: Return of frustratingly easy domain adaptation. In: Proc. of AAAI (2016)
36. Sun, S.S., Shi, H., Wu, Y.: A survey of multi-source domain adaptation. Information Fusion 24, 84–92 (2015)
37. Tommasi, T., Tuytelaars, T.: A testbed for cross-dataset analysis. In: TASK-CV, ECCV workshop (2014)
38. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: Maximizing for domain invariance. CoRR arXiv:1412.3474 (2014)
39. Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A.: Extracting and composing robust features with denoising autoencoders. In: Proc. of ICML (2008)
40. Zhou, J.T., Pan, S.J., Tsang, I.W., Yan, Y.: Hybrid heterogeneous transfer learning through deep learning. In: Proc. of AAAI (2014)
41. Zhou, M., Chang, K.C.: Unifying learning to rank and domain adaptation: Enabling cross-task document scoring. In: Proc. of SIGKDD (ACM). pp. 781–790 (2014)