Domain-Symmetric Networks for Adversarial Domain Adaptation

Yabin Zhang[1,2], Hui Tang[1], Kui Jia [*1], and Mingkui Tan[1]

[1]South China University of Technology
[2]DAMO Academy, Alibaba Group
{zhang.yabin,eehuitang}@mail.scut.edu.cn, {kuijia,mingkuitan}@scut.edu.cn

Abstract

Unsupervised domain adaptation aims to learn a model of classifier for unlabeled samples on the target domain, given training data of labeled samples on the source domain. Impressive progress is made recently by learning invariant features via domain-adversarial training of deep networks. In spite of the recent progress, domain adaptation is still limited in achieving the invariance of feature distributions at a finer category level. To this end, we propose in this paper a new domain adaptation method called Domain-Symmetric Networks (SymNets). The proposed SymNet is based on a symmetric design of source and target task classifiers, based on which we also construct an additional classifier that shares with them its layer neurons. To train the SymNet, we propose a novel adversarial learning objective whose key design is based on a two-level domain confusion scheme, where the category-level confusion loss improves over the domain-level one by driving the learning of intermediate network features to be invariant at the corresponding categories of the two domains. Both domain discrimination and domain confusion are implemented based on the constructed additional classifier. Since target samples are unlabeled, we also propose a scheme of cross-domain training to help learn the target classifier. Careful ablation studies show the efficacy of our proposed method. In particular, based on commonly used base networks, our SymNets achieve the new state of the art on three benchmark domain adaptation datasets.

1. Introduction

Deep learning methods have achieved great success in various machine learning tasks. A common pre-requisite for such success is the availability of massive amounts of annotated training data. For many other tasks, however,

these training data are either difficult to collect, or annotating them costs prohibitively. Therefore, to address the scarcity of annotated data on some *target* tasks/domains, there is a strong motivation to leverage the massively available annotated data on related *source* ones via a manner of transfer learning or domain adaptation [17]. Unfortunately, this attractive learning paradigm suffers from the problem of domain shift [8], which stands as a major obstacle for adapting the learned models on source domains to be useful for target ones.

Domain adaptation aims to obtain models that have smaller risks on target data. Theoretical analysis [2] suggests that such a target risk can be minimized by bounding the risk of a model on the source data and the discrepancy between distributions of the two domains, which inspires many of existing methods [28, 31, 11, 32, 25, 3, 4, 26, 27, 10, 19]. Among existing methods, those based on domain-adversarial training of deep networks [3, 4] achieve the current state of the art on many benchmark domain adaptation datasets [21, 1, 29]. Inspired by generative adversarial networks [6], domain-adversarial training typically plays a minimax game to learn a domain discriminator, which aims to distinguish features of source samples from those of target samples, and a feature extractor, which aims to learn domain-invariant feature representations in order to confuse the domain discriminator. Domain alignment is expected when the minimax optimization reaches an equilibrium.

In spite of the remarkable empirical results achieved by domain-adversarial training methods, they still suffer from a major limitation: even though the feature extractor is well trained to give domain-invariant features of both the source and target samples, the corresponding model/classifier is trained on the source samples and cannot perfectly generalize to the target ones, i.e., the joint distributions of feature and category are not well aligned across data domains. Some of existing methods have paid attention to this issue. For example, in [22, 34, 30], pseudo labels are assigned to target samples, on which the category-level alignment

---

*Corresponding author

is promoted. In [12, 18], multiplicative interactions between feature representations and category predictions are exploited as high-order features to help adversarial training.

These existing methods have to some extent alleviated the above issue. To push further along this line, we propose in this paper a novel design of Domain-Symmetric Networks (SymNets) to facilitate, via adversarial training, the alignment of joint distributions of feature and category across data domains. Similar to [13], our proposed SymNet contains an explicit task classifier for the target domain. Different from [13], we also construct an additional classifier that shares its neurons with those of the source and target classifiers (cf. Section 3.1 for how the three classifiers are constructed). In this work, we propose a novel adversarial learning method to train the thus constructed SymNet, which includes category-level and domain-level confusion losses and can thus enhance domain-invariant feature learning towards the category level. To make the target classifier more symmetric with the source one in terms of predicting task categories, we also propose a cross-domain training scheme to help training of the target classifier. Careful ablation studies show the efficacy of key designs of our proposed SymNet.

We summarize our main contributions as follows.

- We propose in this paper a novel method termed Sym-Net for adversarial domain adaptation. Our proposed SymNet is based on a symmetric design of source and target task classifiers, based on which we also construct an additional classifier that shares with them its layer neurons. Both domain discrimination and domain confusion are implemented based on the constructed additional classifier.

- To train the SymNet, we propose a novel adversarial learning method based on two-level domain confusion losses, where the category-level confusion loss improves over the domain-level one by driving the learning of intermediate network features to be invariant at the corresponding categories of the two domains. Since target samples are unlabeled, we also propose a scheme of cross-domain training to help learn the target classifier.

- We conduct careful ablation studies to investigate the efficacy of key designs of our proposed SymNet. These studies empirically corroborate our designs. In particular, based on commonly used base networks, our proposed SymNets achieve the new state of the art on benchmark domain adaptation datasets of Office-31 [21], ImageCLEF-DA [1], and Office-Home [29].

2. Related Works

In this section, we briefly review recent domain adaptation methods, in particular those aiming to align the joint distributions of feature and category across two data domains [22, 34, 30, 13, 23, 12, 18].

Existing domain adaptation methods [28, 11, 32, 25, 3, 4, 26, 27, 13, 22, 34, 30, 12, 18] typically learn domain invariant features to minimize the domain discrepancy. Some of the existing methods [28, 11, 32, 25, 3, 4, 27] neglect the alignment between the corresponding categories of the two domains. In contrast, to align the joint distributions of feature and category across two data domains, Saito *et al.* [22] proposes to asymmetrically use three task classifiers, where two task classifiers are utilized to label the unlabeled target samples according to their prediction consistency and the confidence, and another task classifier is trained by these target samples with pseudo labels. However, the trueness of pseudo labels is doubtful and false labels have a profoundly negative impact on the performance. To improve the reliability of pseudo labels for target samples, Zhang *et al.* [34] reweights the target samples by the degree of confusion between domains, specifically those target samples which well confuse the domain discriminator in domain labels, are thus assigned by higher weights. Xie *et al.* [30] aligns the centroid of each category between the two domains, instead of treating the pseudo labels as true ones directly. Long *et al.* [13] uses a residual function to model the shift between the learned task classifiers of the two domains, which can be useful in the adaptation tasks of small domain discrepancy but inadequate to tackle the large domain discrepancy. In [12, 18], multiplicative interactions between feature representations and category predictions are exploited as high-order features to help adversarial training. By taking the category decision boundaries into account, Saito *et al.* [23] proposes to detect the target samples near the category decision boundaries by maximizing the discrepancy between the outputs of two separate task classifiers and learn a feature extractor to generate features near the source support for these target samples to minimize the discrepancy.

To further promote the alignment of joint distributions of feature and category across data domains, our SymNets contain an explicit task classifier for the target domain and an additional classifier to enable domain discrimination and domain confusion, and have two-level domain confusion losses, where the category-level confusion loss improves over the domain-level one by driving the learning of intermediate network features to be invariant at the corresponding categories of the two domains.

3. The Proposed Domain-Symmetric Networks

In unsupervised domain adaptation, we are given a source domain $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ of n_s labeled samples
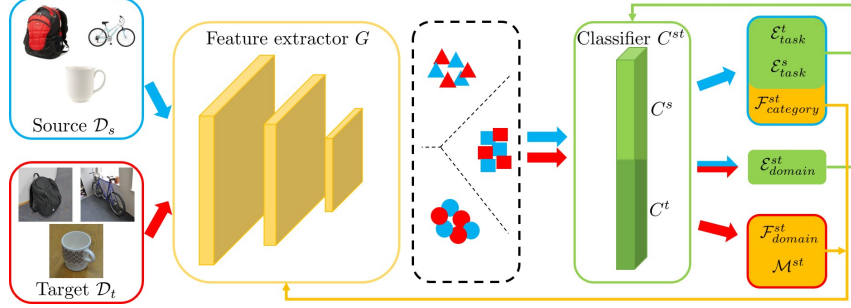
Figure 1. The architecture of our proposed SymNet, which includes a feature extractor G and three classifiers of C^s, C^t and C^{st}. Note that the classifier C^{st} shares its layer neurons with C^s and C^t. The red and blue colors indicate the target data and source data, and the losses generated by them, respectively. The yellow and green colors represent the feature extractor and classifiers, and the losses applied to them, respectively. The middle dashed rectangle presents a toy example of the features of the SymNet that are invariant at the corresponding categories of two domains. Please refer to the main text for how the two-level domain confusion training objectives are defined.

and a target domain $\mathcal{D}_t = \{(\mathbf{x}_j^t)\}_{j=1}^{n_t}$ of n_t unlabeled samples. The i.i.d. assumption is violated as the source domain $\mathcal{D}_s$ and target domain $\mathcal{D}_t$ are assumed to be different. The goal of unsupervised domain adaptation is to learn a feature extractor G and a classifier C such that the expected target risk $\mathbb{E}_{(\mathbf{x}^t,y^t)\sim\mathcal{D}_t}[\mathcal{L}(C(G(\mathbf{x}^t)), y^t)]$ can be minimized for a certain loss $\mathcal{L}$.

Theoretical analysis [2] suggests that the target risk can be minimized by bounding the source risk and the discrepancy between two domains. Inspired by GANs [6], domain-adversarial training [3, 4] is explored to achieve the later objective. As summarized in [27], there are three ways to implement the domain-adversarial training losses: minimax [3, 4], confusion [26], and GAN [27]. We introduce the domain confusion loss [26] that is most related to our method.

Given a deep neural network that is composed of convolutional and fully-connected (FC) layers, the domain confusion method uses the lower convolutional layers as the feature extractor G and upper FC layers as the task classifier C. The domain discriminator D, which is in parallel with C, is added on top of G to distinguish features of samples from the two domains. Source risk minimization is achieved based on a standard supervised classification objective:

$$\min_{C,G} \mathcal{E}_{task} = \frac{1}{n_s} \sum_{i=1}^{n_s} \mathcal{L}^s\left(C(G(\mathbf{x}_i^s)), y_i^s\right), \qquad (1)$$

where $\mathcal{L}^s$ is typically a cross-entropy loss. Due to the existence of domain discrepancy, there is a large drop in performance when directly applying the model trained by (1) to the target data. Given feature representations of different domains extracted by G, we can learn a domain discrimina-

tor D using the following objective:

$$\min_{D} \mathcal{E}_{domain} = -\frac{1}{n_s} \sum_{i=1}^{n_s} \log(1 - D(G(\mathbf{x}_i^s)))$$
$$-\frac{1}{n_t} \sum_{j=1}^{n_t} \log(D(G(\mathbf{x}_j^t))). \qquad (2)$$

Given a D, the domain confusion loss aims to learn G to maximally "confuse" the two domains, by computing the cross entropy between the domain predictions and a uniform distribution over domain labels:

$$\min_{G} \mathcal{F}_{domain} = \frac{1}{2}\mathcal{E}_{domain} - \frac{1}{2n_s} \sum_{i=1}^{n_s} \log(D(G(\mathbf{x}_i^s)))$$
$$-\frac{1}{2n_t} \sum_{j=1}^{n_t} \log(1 - D(G(\mathbf{x}_j^t))). \qquad (3)$$

Domain alignment is achieved by learning a domain-invariant G based on the following adversarial objective of domain confusion:

$$\min_{G,C} \mathcal{E}_{task}(G,C) + \lambda \mathcal{F}_{domain}(G,D)$$
$$\min_{D} \mathcal{E}_{domain}(G,D), \qquad (4)$$

where λ is a trade-off parameter.

3.1. A Symmetric Design of Source and Target Task Classifiers

As discussed in Section 1, although impressive results are obtained by existing methods of domain-adversarial training, they still suffer from the fundamental challenge of unsupervised domain adaptation, i.e., the joint distributions of feature and category cannot be well aligned across data domains. To address this challenge, we propose in this

paper a novel domain-symmetric network (SymNet), with the corresponding domain-adversarial training method. We first present architectural design of our proposed SymNet as follows (cf. Figure 1 for an illustration).

The design of a SymNet starts with two parallel task classifiers C^s and C^t. Assume each of the two classifiers is based on a single FC layer (with a subsequent softmax operation). C^s and C^t respectively contain K^s and K^t neurons corresponding to the numbers of categories on the source and target domains. In unsupervised domain adaptation, we have $K^s = K^t = K$. For an input $\mathbf{x}$ of the SymNet, we respectively denote as $\mathbf{v}^s(\mathbf{x}) \in \mathbb{R}^K$ and $\mathbf{v}^t(\mathbf{x}) \in \mathbb{R}^K$ the output vectors of C^s and C^t before the softmax operation, and $\mathbf{p}^s(\mathbf{x}) \in [0,1]^K$ and $\mathbf{p}^t(\mathbf{x}) \in [0,1]^K$ after the softmax operation. Except for C^s and C^t, our Sym-Net also has a classifier C^{st} whose design is as follows. Given $\mathbf{v}^s(\mathbf{x})$ and $\mathbf{v}^t(\mathbf{x})$ for an input $\mathbf{x}$, we first concatenate them to form $[\mathbf{v}^s(\mathbf{x}); \mathbf{v}^t(\mathbf{x})] \in \mathbb{R}^{2K}$, and we then apply the softmax operation to the concatenated vector to have a probability vector $\mathbf{p}^{st}(\mathbf{x}) \in [0,1]^{2K}$. We thus have $C^{st}(G(\mathbf{x})) = \mathbf{p}^{st}(\mathbf{x})$. For ease of subsequent notations, we also write $p_k^s(\mathbf{x})$ (resp. $p_k^t(\mathbf{x})$ or $p_k^{st}(\mathbf{x})$), $k \in \{1, \ldots, K\}$, for the kth element of the category probability vector $\mathbf{p}^s(\mathbf{x})$ (resp. $\mathbf{p}^t(\mathbf{x})$ or $\mathbf{p}^{st}(\mathbf{x})$) predicted by $C^s(G(\mathbf{x}))$ (resp. $C^t(G(\mathbf{x}))$ or $C^{st}(G(\mathbf{x}))$).

Note that there exists no an explicit domain discriminator in our design of SymNet. Both the domain discrimination and domain confusion is achieved by applying appropriate losses to the classifier C^{st}, which we will present shortly. We first present in the following how to train C^s and C^t.

Learning of Source Task Classifier

The task classifier C^s is trained using the following cross-entropy loss over the labeled source samples:

$$\min_{C^s} \mathcal{E}_{task}^s(G, C^s) = -\frac{1}{n_s} \sum_{i=1}^{n_s} \log(p_{y_i^s}^s(\mathbf{x}_i^s)). \quad (5)$$

Cross-Domain Learning of Target Task Classifier

Since target samples are unlabeled, there exist no direct supervision signals to learn a task classifier C^t. Our idea is to leverage the labeled source samples, and use the following cross-entropy loss to train C^t:

$$\min_{C^t} \mathcal{E}_{task}^t(G, C^t) = -\frac{1}{n_s} \sum_{i=1}^{n_s} \log(p_{y_i^s}^t(\mathbf{x}_i^s)). \quad (6)$$

At a first glance, it seems that (6) learns C^t that is a duplicate of C^s. However, a domain discrimination training via C^{st} will make them distinguishable. In fact, the use of (6) is essential to establish a neuron-wise correspondence between C^s and C^t, which provides the basis to achieve category-level domain confusion presented in Section 3.2. The use of labeled source samples in (6) also makes the learned C^t more discriminative among task categories. We
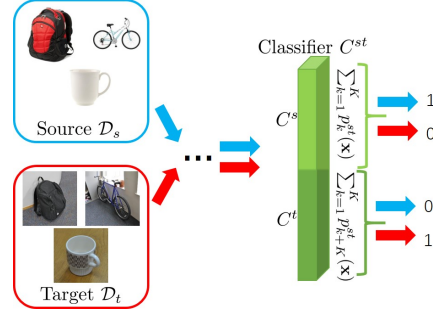

Figure 2. An intuitive presentation of how the loss (7) differentiates the target classifier C^t from the source classifier C^s.

present ablation studies in Section 4.3 that confirm the efficacy of our way of learning the target task classifier C^t.

Domain Discrimination

Both C^s and C^t are trained using the labeled source samples. To differentiate between them, we leverage the constructed classifier C^{st} in the SymNet. We train C^{st} using the following two-way cross-entropy loss:

$$\min_{C^{st}} \mathcal{E}_{domain}^{st}(G, C^{st}) = -\frac{1}{n_t} \sum_{j=1}^{n_t} \log(\sum_{k=1}^{K} p_{k+K}^{st}(\mathbf{x}_j^t))$$
$$-\frac{1}{n_s} \sum_{i=1}^{n_s} \log(\sum_{k=1}^{K} p_k^{st}(\mathbf{x}_i^s)), \quad (7)$$

where $\sum_{k=1}^{K} p_k^{st}(\mathbf{x})$ and $\sum_{k=1}^{K} p_{k+K}^{st}(\mathbf{x})$ can be viewed as the probabilities of classifying an input sample $\mathbf{x}$ as the source and target domains respectively. The objective of the loss (7) is intuitively illustrated in Figure 2.

Ideally, for the total $2K$ neurons of C^{st}, the combined effect of imposing losses (5), (6), and (7) would be to make the set of first K neurons discriminative among task categories, the set of last K neurons discriminative among task categories, and to make the two sets distinguishable from each other. For example, for a source sample $\mathbf{x}^s$ of the category k, both C^s and C^t tend to make accurate predictions, and for C^{st}, the probability of p_k^{st} would be larger than p_{k+K}^{st}, due to the use of loss (7). Similarly, for a target sample $\mathbf{x}^t$ of the category k, both C^s and C^t tend to make accurate predictions, and for C^{st}, the probability of p_{k+K}^{st} would be larger than p_k^{st}.

3.2. A Two-level Domain Confusion Training of Domain-Symmetric Networks

Similar to existing methods, we adopt the general strategy of adversarial training to learn an invariant feature extractor G for the SymNet. More specifically, we propose a novel two-level domain confusion method that is based on a domain-level confusion loss and a category-level confusion loss. The proposed two-level losses aim to maximally

" confuse" the two domains in order to align the joint distributions of feature and category across them.

To have a category-level confusion loss, we again rely on labeled source samples. For a source sample of category k, we identify its corresponding pair of the kth and $(k + K)^{th}$ neurons in C^{st}, and use a cross-entropy between predictions on this neuron pair and uniform distribution, which gives the following objective to learn the feature extractor G:

$$\min_{G} \mathcal{F}_{category}^{st}(G, C^{st}) = -\frac{1}{2n_s} \sum_{i=1}^{n_s} \log(p_{y_i^s + K}^{st}(\mathbf{x}_i^s))$$
$$-\frac{1}{2n_s} \sum_{i=1}^{n_s} \log(p_{y_i^s}^{st}(\mathbf{x}_i^s)). \quad (8)$$

To have a domain-level confusion loss, we use the unlabeled target samples, since label information is unnecessary for confusion at the domain level. For a target sample, we simply use a cross-entropy between aggregated predictions from the two half sets of neurons in C^{st}, and uniform distribution, which gives the following objective to learn the feature extractor G:

$$\min_{G} \mathcal{F}_{domain}^{st}(G, C^{st}) = -\frac{1}{2n_t} \sum_{j=1}^{n_t} \log(\sum_{k=1}^{K} p_{k+K}^{st}(\mathbf{x}_j^t))$$
$$-\frac{1}{2n_t} \sum_{j=1}^{n_t} \log(\sum_{k=1}^{K} p_{k}^{st}(\mathbf{x}_j^t)). \quad (9)$$

Note that one may opt for another domain-level confusion loss by using labeled source samples. We note that effect of such an additional loss may have been subsumed by the category-level confusion loss (8), which uses labeled source samples.

3.2.1 Entropy Minimization Principle

Entropy minimization principle [7] is adopted by some domain adaptation methods [13, 33, 24] to enhance discrimination of learned models for target data. In this work, we adapt this principle to the symmetric structure of our proposed SymNet. We propose the following entropy minimization objective that enhances discrimination among task categories by summing over the probabilities at each pair of category-corresponding neurons in C^{st}:

$$\min_{G} \mathcal{M}^{st}(G, C^{st}) = -\frac{1}{n_t} \sum_{j=1}^{n_t} \sum_{k=1}^{K} q_k^{st}(\mathbf{x}_j^t) \log(q_k^{st}(\mathbf{x}_j^t)), \quad (10)$$

where $q_k^{st}(\mathbf{x}_j^t) = p_k^{st}(\mathbf{x}_j^t) + p_{k+K}^{st}(\mathbf{x}_j^t)$, $k \in \{1, \ldots, K\}$. As suggested by [33], instead of using (10) for updating both the feature extractor G and the classifier C^{st}, the entropy

minimization loss is only used here to update G, in order to reduce the side effect that due to large domain shift, target samples may be stuck into wrong category predictions in the early stage of training, and are difficult to be corrected later on.

3.3. The Overall Training Objective of Domain-Symmetric Networks

Combining the losses (5), (6), and (7) for updating classifiers, (8) and (9) of category- and domain-level confusion for updating the feature extractor G, and also the regularizer (10), we have the following training objective for a SymNet:

$$\min_{C^s, C^t, C^{st}} \mathcal{E}_{task}^s(G, C^s) + \mathcal{E}_{task}^t(G, C^t) + \mathcal{E}_{domain}^{st}(G, C^{st})$$
$$\min_{G} \mathcal{F}_{category}^{st}(G, C^{st}) + \lambda(\mathcal{F}_{domain}^{st}(G, C^{st}) + \mathcal{M}^{st}(G, C^{st})), \quad (11)$$

where $\lambda \in [0, 1]$ is a trade-off parameter to suppress noisy signals of $\mathcal{F}_{domain}^{st}(G, C^{st})$ and $\mathcal{M}^{st}(G, C^{st})$ at early stages of training. The $\mathcal{F}_{category}^{st}(G, C^{st})$ is noise-free since it is based on the labeled source samples.

4. Experiments

We evaluate our SymNets on unsupervised domain adaptation tasks of three benchmark datasets and investigate the effects of the components in detail. The codes are available at http://sites.scut.edu.cn/GPI/main.psp

4.1. Setup

Office-31 The office-31 dataset [21] is a standard benchmark dataset for domain adaptation, which contains $4,110$ images of 31 categories shared by three distinct domains: *Amazon* (**A**), *Webcam* (**W**) and *DSLR* (**D**). We follow the common evaluation protocol on all six adaptation tasks.

ImageCLEF-DA The ImageCLEF-DA dataset [1] is a benchmark dataset for ImageCLEF 2014 domain adaptation challenge, which contains three domains: *Caltech-256* (**C**), *ImageNet ILSVRC 2012* (**I**) and *Pascal VOC 2012* (**P**). For each domain, there are 12 categories and 50 images in each category. The three domains in this dataset are of the same size, which is a good complementation of the Office-31 dataset where different domains are of different sizes. We evaluate all methods on all six adaptation tasks.

Office-Home The Office-Home dataset [29] is a very challenging dataset for domain adaptation, which contains $15,500$ images from 65 categories of everyday objects in the office and home scenes, shared by four significantly different domains: Artistic images (**Ar**), Clip Art (**Cl**), Product images (**Pr**) and Real-World images (**Rw**). We evaluate all methods on all 12 adaptation tasks.

We compare our SymNets with shallow domain adaptation methods [16, 5] and the state-of-the-art deep domain

Table 1. Accuracy (%) on the Office-31 dataset [21]. All methods are based on models adapted from a 50-layer ResNet.

Methods	A → W	D → W	W → D	A → D	D → A	W → A	Avg
ResNet-50 [9]	68.4±0.2	96.7±0.1	99.3±0.1	68.9±0.2	62.5±0.3	60.7±0.3	76.1
GFK [5]	72.8±0.0	95.0±0.0	98.2±0.0	74.5±0.0	63.4±0.0	61.0±0.0	77.5
TCA [16]	72.7±0.0	96.7±0.0	99.6±0.0	74.1±0.0	61.7±0.0	60.9±0.0	77.6
DAN [11]	80.5±0.4	97.1±0.2	99.6±0.1	78.6±0.2	63.6±0.3	62.8±0.2	80.4
RTN [13]	84.5±0.2	96.8±0.1	99.4±0.1	77.5±0.3	66.2±0.2	64.8±0.3	81.6
RevGrad [3]	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
ADDA [27]	86.2±0.5	96.2±0.3	98.4±0.3	77.8±0.3	69.5±0.4	68.9±0.5	82.9
JAN-A[14]	86.0±0.4	96.7±0.3	99.7±0.1	85.1±0.4	69.2±0.3	70.7±0.5	84.6
MADA [18]	90.0±0.1	97.4±0.1	99.6±0.1	87.8±0.2	70.3±0.3	66.4±0.3	85.2
iCAN [34]	92.5	98.8	100.0	90.1	72.1	69.9	87.2
Kang *et al.* [10]	86.8±0.2	**99.3**±0.1	**100.0**±.0	88.8±0.4	74.3±0.2	**73.9**±0.2	87.2
CDAN+E [12]	**94.1**±0.1	98.6±0.1	**100.0**±.0	92.9±0.2	71.0±0.3	69.3±0.3	87.7
SymNets	90.8±0.1	98.8±0.3	**100.0**±.0	**93.9**±0.5	**74.6**±0.6	72.5±0.5	**88.4**

Table 2. Accuracy (%) on the ImageCLEF-DA dataset [1]. All methods are based on models adapted from a 50-layer ResNet.

Methods	I → P	P → I	I → C	C → I	C → P	P → C	Avg
ResNet-50 [9]	74.8±0.3	83.9±0.1	91.5±0.3	78.0±0.2	65.5±0.3	91.2±0.3	80.7
DAN [11]	74.5±0.4	82.2±0.2	92.8±0.2	86.3±0.4	69.2±0.4	89.8±0.4	82.5
RevGrad [3]	75.0±0.6	86.0±0.3	96.2±0.4	87.0±0.5	74.3±0.5	91.5±0.6	85.0
MADA [18]	75.0±0.3	87.9±0.2	96.0±0.3	88.8±0.3	75.2±0.2	92.2±0.3	85.8
iCAN [34]	79.5	89.7	94.7	89.9	78.5	92.0	87.4
CDAN+E [12]	77.7±0.3	90.7±0.2	**97.7**±0.3	91.3±0.3	74.2±0.2	94.3±0.3	87.7
SymNets	**80.2**±0.3	**93.6**±0.2	97.0±0.3	**93.4**±0.3	**78.7**±0.3	**96.4**±0.1	**89.9**

adaptation methods [11, 13, 3, 27, 14, 18, 19, 10, 12]. We follow standard evaluation protocols for unsupervised domain adaptation [3, 11]: all labeled source samples and all unlabeled target samples are used for training. The average classification accuracy and the standard error of each adaptation task are reported on three random experiments. Our SymNets and all comparative methods are based on models adapted from a 50-layer ResNet [9]. Especially, the deep representations output by the layer $pool5$ of ResNet are used as features for shallow methods.

We implement our SymNets based on PyTorch. A 50-layer ResNet pre-trained on the ImageNet dataset [20], which excludes the last FC layer, is adopted as the feature extractor G. We fine-tune the feature extractor G and train a classifier C^{st} from scratch through back propagation. The learning rate of the classifier C^{st} is 10 times that of the feature extractor G. All parameters are updated by stochastic gradient descent (SGD) with momentum of 0.9. The batch size is set to 128. We follow [3] to employ the annealing strategy of learning rate and the progressive strategy of λ: the learning rate is adjusted by $\eta_p = \frac{\eta_0}{(1+\alpha p)^\beta}$, where p is the progress of training epochs linearly changing from 0 to 1, $\eta_0 = 0.01$, $\alpha = 10$ and $\beta = 0.75$, which are optimized to promote convergence and low error on source samples; λ is gradually changed from 0 to 1 by $\lambda_p = \frac{2}{1+\exp(-\gamma \cdot p)} - 1$, where γ is set to 10 in all experiments. Our classification

results are obtained from the target task classifier C_t unless otherwise specified, and the comparison between the performance of the source and target task classifiers is illustrated in Figure 4.

4.2. Results

The classification results on the Office-31 [21], ImageCLEF-DA [1] and Office-Home [29] datasets are reported in Table 1, Table 2 and Table 3, respectively. For fair comparison, results of other methods are either directly reported from their original papers if available or quoted from [12]. Our SymNets outperform all state-of-the-art methods on three benchmark datasets, highly affirming the effectiveness of our SymNets in aligning the joint distributions of feature and category across domains. It is compelling that our SymNets substantially enhance the classification accuracies on difficult adaptation tasks (e.g. **A → D** and **D → A**) and the challenging dataset (e.g. Office-Home). The Office-Home dataset is a very challenging dataset for domain adaptation due to following reasons as described in its original paper [29]: (1) the number of categories is large in each domain; (2) different domains are visually very dissimilar; (3) the in-domain classification accuracy is low. Especially, the presence of large number of categories prejudices the domain alignment methods [3, 11, 27] for their ignorance of the alignment between corresponding categories of the two domains. It is desirable that our SymNets dramat-

Table 3. Accuracy (%) on the Office-Home dataset [29]. All methods are based on models adapted from a 50-layer ResNet.

Methods	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
ResNet-50 [9]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DAN [11]	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3	56.3
RevGrad [3]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN+E [12]	**50.7**	70.6	76.0	57.6	70.0	70.0	57.4	**50.9**	77.3	70.9	**56.7**	81.6	65.8
SymNets	47.7	**72.9**	**78.5**	**64.2**	**71.3**	**74.2**	**64.2**	48.8	**79.5**	**74.5**	52.6	**82.7**	**67.6**

Table 4. Ablation experiments on the Office-31 dataset [21]. All methods are based on models adapted from a 50-layer ResNet. Please refer to the main text for the detail definitions of these methods.

Methods	A → W	D → W	W → D	A → D	D → A	W → A	Avg
ResNet-50 [9]	79.9±0.3	96.8±0.4	99.5±0.1	84.1±0.4	64.5±0.3	66.4±0.4	81.9
ResNet-50 (Adding Em) [9]	89.3±0.1	**99.0**±0.1	**100.0**±.0	89.2±0.7	73.4±0.1	69.0±0.2	86.6
Domain Confusion [26]	83.0±0.1	98.5±0.3	99.8±0.0	83.9±0.0	66.9±0.4	66.4±0.1	83.1
Domain Confusion (Adding Em) [26]	89.8±0.7	**99.0**±0.2	**100.0**±.0	90.1±0.3	73.9±0.7	69.0±0.8	87.0
SymNets (w/o $\mathcal{E}_{task}^{t}$)	75.3±0.9	95.9±0.2	99.6±0.2	75.1±0.9	60.2±0.3	62.7±0.7	78.1
SymNets (w/o $\mathcal{M}^{st}$)	87.9±0.1	98.4±0.2	99.9±0.1	90.8±0.5	67.4±0.6	69.7±0.7	85.7
SymNets (w/o confusion)	89.2±0.6	**99.0**±0.3	**100.0**±.0	93.8±0.3	73.7±0.2	65.9±0.6	86.9
SymNets (w/o category confusion)	89.9±0.6	98.1±0.1	99.8±0.0	93.7±0.5	71.9±0.2	**73.5**±0.1	87.8
SymNets	**90.8**±0.1	98.8±0.3	**100.0**±.0	**93.9**±0.5	**74.6**±0.6	72.5±0.5	**88.4**

ically improve the performance on most adaptation tasks, demonstrating the efficiency of our proposed two-level domain confusion training of SymNets in aligning the joint distributions of feature and category across domains.

4.3. Analysis

Ablation Study In this section, we conduct ablation experiments on the Office-31 dataset [21] to investigate the effects of different components in our SymNets, which are based on models adapted from a 50-layer ResNet. We begin with the simplest baseline that fine-tunes on source samples the ResNet-50 model that is pre-trained on the ImageNet dataset [20], which is denoted as "ResNet-50". To find out how the existing domain confusion method introduced in Section 3 performs, we conduct the experiment using the adversarial objective of domain confusion (4), which is denoted as "Domain Confusion". To make it clear how our adopted entropy minimization loss presented in Section 3.2.1 can help the above two baselines, we additionally optimize the entropy minimization loss of target samples over their feature extractors and denote them as "ResNet-50 (Adding Em)" and "Domain Confusion (Adding Em)" respectively. To investigate how different components in our SymNets benefit the adaptation performance, we remove the cross-domain category supervised loss $\mathcal{E}_{task}^{t}(G, C^t)$ (6) and the entropy minimization loss $\mathcal{M}^{st}(G, C^{st})$ (10) from the overall adversarial training objective (11), the training settings of which are denoted as "SymNets (w/o $\mathcal{E}_{task}^{t}$)" and "SymNets (w/o $\mathcal{M}^{st}$)", respectively. Note that classification accuracies for SymNets (w/o $\mathcal{E}_{task}^{t}$) are obtained from

the source task classifier C^s due to the inexistence of the direct supervision signals in target task classifier C^t. Besides, to explore the effects of our proposed two-level domain confusion losses, we degenerate the category-level confusion loss $\mathcal{F}_{category}^{st}$ (8) for source samples to a domain-level one:

$$\min_{G} -\frac{1}{2n_s}\sum_{i=1}^{n_s}\log(\sum_{k=1}^{K}p_k^{st}(\mathbf{x}_i^s)) - \frac{1}{2n_s}\sum_{i=1}^{n_s}\log(\sum_{k=1}^{K}p_{k+K}^{st}(\mathbf{x}_i^s)),$$
(12)

the training setting of which is denoted as "SymNets (w/o category confusion)". We remove the domain-level confusion loss $\mathcal{F}_{domain}^{st}$ (9) for target samples from the overall adversarial training objective (11) and degenerate the category-level confusion loss $\mathcal{F}_{category}^{st}$ (8) for source samples to a general category classification loss:

$$\min_{G} -\frac{1}{2n_s}\sum_{i=1}^{n_s}\log(p_{y_i^s}^{s}(\mathbf{x}_i^s)) - \frac{1}{2n_s}\sum_{i=1}^{n_s}\log(p_{y_i^s}^{t}(\mathbf{x}_i^s)),$$
(13)

the training setting of which is denoted as "SymNets (w/o confusion)".

The results are reported in Table 4. "Domain Confusion" performs much better than "ResNet-50", and "SymNets (w/o category confusion)" improves over "SymNets (w/o confusion)", testifying the effectiveness of the domain-level confusion in the feature alignment. Observed that the performance of "SymNets (w/o $\mathcal{E}_{task}^{t}$)" suffers a slump of 10.3%, manifesting the importance of the cross-domain

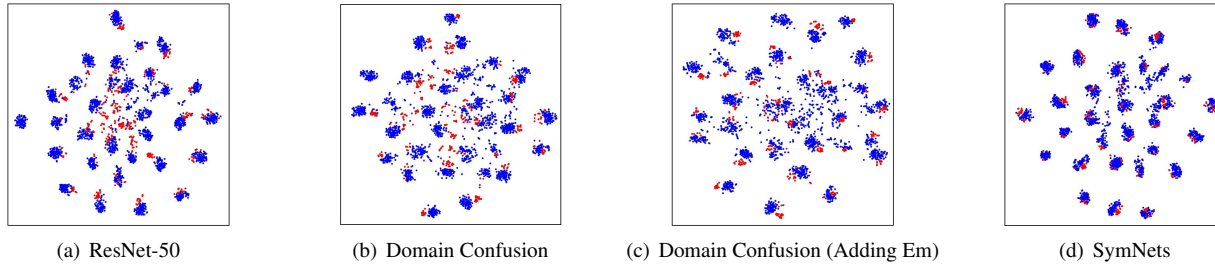| (a) ResNet-50 | (b) Domain Confusion | (c) Domain Confusion (Adding Em) | (d) SymNets |

Figure 3. The t-SNE visualization of feature representations learned by (a) ResNet-50, (b) Domain Confusion, (c) Domain Confusion (Adding Em) and (d) SymNets. Note that the blue and red points are samples from the source domain **A** and target domain **W** respectively.

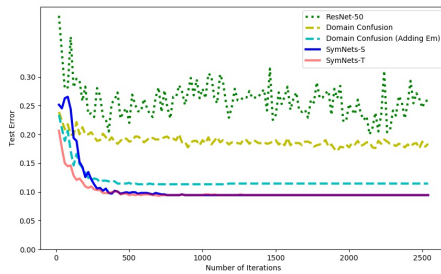Figure 4. Convergence performance on the adaptation task of $\mathbf{A} \rightarrow \mathbf{W}$ by ResNet-50, Domain Confusion, Domain Confusion (Adding Em), and the source and target task classifiers of our SymNets, which are denoted as SymNets-S and SymNets-T respectively.

category supervised loss $\mathcal{E}_{task}^{t}(G, C^t)$ (6) to learn a well-performed target task classifier in the two-level confusion training of SymNets. SymNets enhances the adaptation performance over "SymNets (w/o category confusion)", certifying the usefulness of our proposed category-level confusion in the alignment between corresponding categories of the two domains. The entropy minimization loss $\mathcal{M}^{st}(G, C^{st})$ (10) consistently improves both the two baselines of "ResNet-50" and "Domain Confusion" and our SymNets by a large margin in performance, demonstrating its efficacy. By fair comparison, our SymNets achieve the best result among these ablation experiments, confirming their excellent effect in aligning the joint distributions of feature and category across domains.

Convergence Performance   We compare the convergence performances of task classifiers of C^s and C^t in our SymNets with "ResNet-50", "Domain Confusion" and "Domain Confusion (Adding Em)" in Figure 4. The test errors of different methods on the adaptation task $\mathbf{A} \rightarrow \mathbf{W}$ are reported. We observe that our SymNets converge much smoother. The convergence performances of the source task classifier C^s and target task classifier C^t are expected. At the beginning of adversarial training, the performance of the target task classifier on target samples is better, since

the source task classifier and target task classifier are specified to corresponding domains. As the training proceeds, the joint distributions of feature and category are gradually aligned across domains, thus the performance of two task classifiers almost converge to the same level.

Feature Visualization   We visualize the network activations from feature extractors of "ResNet-50", "Domain Confusion", "Domain Confusion (Adding Em)" and our SymNets on the adaptation task $\mathbf{A} \rightarrow \mathbf{W}$ by t-SNE [15] in Figure 3. The source and target domains are not well aligned for features of "ResNet-50". For features of "Domain Confusion", the two domains are better aligned, however, the data structure of target samples is scattered and the shared categories across domains are not well aligned. For features of "Domain Confusion (Adding Em)", the data structure of target samples is well preserved, but the shared categories across domains are not well aligned. For features of our SymNets, the shared categories across domains are perfectly aligned while different categories are well distinguished. The effectiveness of the two-level domain confusion training of SymNets in aligning joint distributions of feature and category across domains is verified intuitively.

5. Conclusion

We propose a novel adversarial learning method termed domain-symmetric networks (SymNets) to overcome the limitation in aligning the joint distributions of feature and category across domains via two-level domain confusion losses. The category-level confusion loss improves over the domain-level one by driving the learning of intermediate network features to be invariant at the corresponding categories of the two domains. As a component of the SymNets, an explicit target task classifier is learned through a cross-domain training scheme. Experiments on three benchmark datasets verify the efficacy of our proposed SymNets.

References

[1] Imageclef-da dataset. http://imageclef.org/2014/adaptation/.

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

[3] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1180–1189, 2015.

[4] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2017.

[5] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2066–2073. IEEE, 2012.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[7] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005.

[8] A. Gretton, AJ. Smola, J. Huang, M. Schmittfull, KM. Borgwardt, and B. Schölkopf. *Covariate shift and local learning by distribution matching*, pages 131–160. MIT Press, Cambridge, MA, USA, 2009.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[10] Guoliang Kang, Liang Zheng, Yan Yan, and Yi Yang. Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In *The European Conference on Computer Vision (ECCV)*, September 2018.

[11] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. Learning transferable features with deep adaptation networks. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 97–105. JMLR.org, 2015.

[12] Mingsheng Long, ZHANGJIE CAO, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 1640–1650. Curran Associates, Inc., 2018.

[13] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

[14] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I. Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 2208–2217, 2017.

[15] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[16] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[17] Sinno Jialin Pan, Qiang Yang, et al. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[18] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

[19] Pedro O Pinheiro and AI Element. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8004–8013, 2018.

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[21] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[22] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. Asymmetric tri-training for unsupervised domain adaptation. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2988–2997, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[23] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018.

[24] Rui Shu, Hung H Bui, Hirokazu Narui, and Stefano Ermon. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*, 2018.

[25] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

[26] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

[27] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

[28] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[29] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proc. CVPR*, pages 5018–5027, 2017.

[30] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 5419–5428, 2018.

[31] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 3, 2017.

[32] Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. Central moment discrepancy (cmd) for domain-invariant representation learning. *arXiv preprint arXiv:1702.08811*, 2017.

[33] Jing Zhang, Zewei Ding, Wanqing Li, and Philip Ogunbona. Importance weighted adversarial nets for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8156–8164, 2018.

[34] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018.