Partial Transfer Learning with Selective Adversarial Networks

Zhangjie Cao[†], Mingsheng Long[†], Jianmin Wang[†], and Michael I. Jordan[♯]
[†]KLiss, MOE; NEL-BDS; TNList; School of Software, Tsinghua University, China
[♯]University of California, Berkeley, Berkeley, USA
caozhangjie14@gmail.com {mingsheng,jimwang}@tsinghua.edu.cn jordan@cs.berkeley.edu

Abstract

Adversarial learning has been successfully embedded into deep networks to learn transferable features, which reduce distribution discrepancy between the source and target domains. Existing domain adversarial networks assume fully shared label space across domains. In the presence of big data, there is strong motivation of transferring both classification and representation models from existing large-scale domains to unknown small-scale domains. This paper introduces partial transfer learning, which relaxes the shared label space assumption to that the target label space is only a subspace of the source label space. Previous methods typically match the whole source domain to the target domain, which are prone to negative transfer for the partial transfer problem. We present Selective Adversarial Network (SAN), which simultaneously circumvents negative transfer by selecting out the outlier source classes and promotes positive transfer by maximally matching the data distributions in the shared label space. Experiments demonstrate that our models exceed state-of-the-art results for partial transfer learning tasks on several benchmark datasets.

1. Introduction

Deep networks have significantly improved the state of the art for a wide variety of machine learning problems and applications. At the moment, these impressive gains in performance come only when massive amounts of labeled data are available. Since manual labeling of sufficient training data for diverse application domains on-the-fly is often prohibitive, for problems short of labeled data, there is strong motivation to establishing effective algorithms to reduce the labeling consumption, typically by leveraging off-the-shelf labeled data from a different but related source domain. This promising transfer learning paradigm, however, suffers from the shift in data distributions across different domains, which poses a major obstacle in adapting classification models to target tasks [22].

Existing transfer learning methods assume shared label space and different feature distributions across the source and target domains. These methods bridge different domains by learning domain-invariant feature representations without using target labels, and the classifier learned from source domain can be directly applied to target domain. Recent studies have revealed that deep networks can learn more transferable features for transfer learning [4, 31], by disentangling explanatory factors of variations behind domains. The latest advances have been achieved by embedding transfer learning in the pipeline of deep feature learning to extract domain-invariant deep representations [28, 15, 6, 29, 17].

In the presence of big data, we can readily access large-scale labeled datasets such as ImageNet-1K. Thus, a natural ambition is to directly transfer both the representation and classification models from large-scale dataset to our target dataset, such as Caltech-256, which are usually small-scale and with unknown categories at training and testing time. From big data viewpoint, we can assume that the large-scale dataset is big enough to subsume all categories of the small-scale dataset. Thus, we introduce a novel *partial* transfer learning problem, which assumes that the target label space is a subspace of the source label space. As shown in Figure 1, this new problem is more general and challenging than standard transfer learning, since outlier source classes ("sofa") will result in negative transfer when discriminating the target classes ("soccer-ball" and "binoculars"). Negative transfer is the phenomenon that a transfer learner performs even worse than a supervised classifier trained solely on the source domain, which is the key challenge of transfer learning [22]. Thus, matching the whole source and target domains as previous methods is not an effective solution to this new problem.

This paper presents Selective Adversarial Networks (SAN), which largely extends the ability of deep adversarial adaptation [6] to address partial transfer learning from big domains to small domains. SAN aligns the distributions of source and target data in the shared label space and more importantly, selects out the source data in the outlier source classes. A key improvement over previous methods is the capability to simultaneously promote positive transfer of rel-

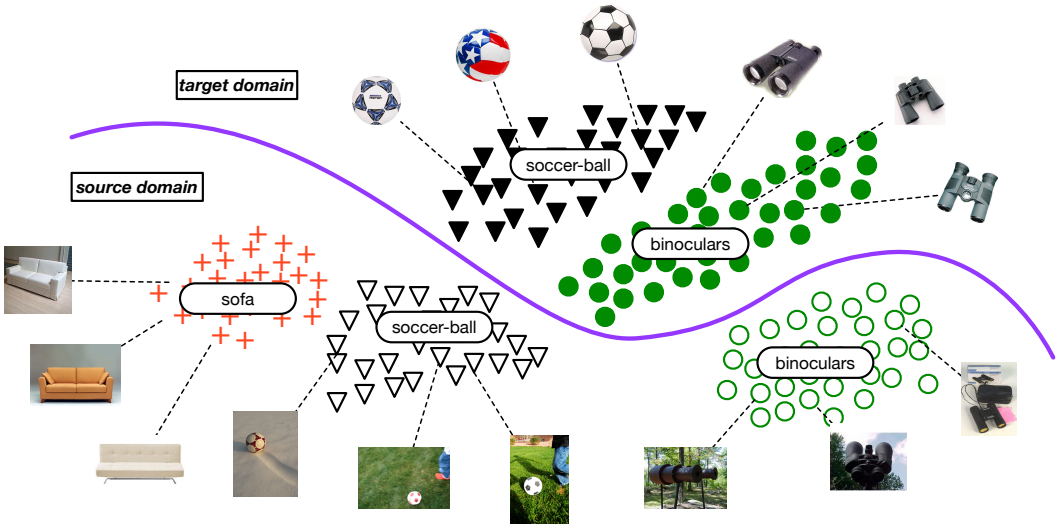


Figure 1. The partial transfer learning problem, where source label space subsumes target label space (should be prone to negative transfer).

evant data and alleviate negative transfer of irrelevant data, which can be trained in an end-to-end framework. Experiments show that our models exceed state-of-the-art results for deep transfer learning on public benchmark datasets.

2. Related Work

Transfer learning [22] bridges different domains or tasks to mitigate the burden of manual labeling for machine learning [21, 5, 32, 30], computer vision [24, 8, 13] and natural language processing [3]. The main technical difficulty of transfer learning is to formally reduce the distribution discrepancy across different domains. Deep networks can learn abstract representations that disentangle different explanatory factors of variations behind data [2] and manifest invariant factors underlying different populations that transfer well from original tasks to similar novel tasks [31]. Thus deep networks have been explored for transfer learning [7, 20, 13], multimodal and multi-task learning [3, 19], where significant performance gains have been witnessed relative to prior shallow transfer learning methods.

However, recent advances show that deep networks can learn abstract feature representations that can only reduce, but not remove, the cross-domain discrepancy [7, 28], resulting in unbounded risk for target tasks [18, 1]. Some recent work bridges deep learning and domain adaptation [28, 15, 6, 29, 17], which extends deep convolutional networks (CNNs) to domain adaptation by adding adaptation layers through which the mean embeddings of distributions are matched [28, 15, 17], or by adding a subnetwork as domain discriminator while the deep features are learned to confuse the discriminator in a domain-adversarial training paradigm [6, 29]. While performance was significantly improved, these state of the art methods may be restricted by the assumption that the source and target domains share the same label space. This assumption is violated in partial transfer learning, which transfers both representation and classification models from existing big domains to unknown small domains. To our knowledge, this is the first work that addresses partial transfer learning in adversarial networks.

3. Partial Transfer Learning

In this paper, we propose *partial transfer learning*, a novel transfer learning paradigm where the target domain label space $\mathcal{C}_t$ is a subspace of the source domain label space $\mathcal{C}_s$ i.e. $\mathcal{C}_t \subset \mathcal{C}_s$. This new paradigm finds wide applications in practice, as we usually need to transfer a model from a large-scale dataset (e.g. ImageNet) to a small-scale dataset (e.g. Caltech-256). Similar to standard transfer learning, in partial transfer learning we are also provided with a *source* domain $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ of n_s labeled examples associated with $|\mathcal{C}_s|$ classes and a *target* domain $\mathcal{D}_t = \{\mathbf{x}_i^t\}_{i=1}^{n_t}$ of n_t unlabeled examples associated with $|\mathcal{C}_t|$ classes, but differently, we have $|\mathcal{C}_s| > |\mathcal{C}_t|$ in partial transfer learning. The source domain and target domain are sampled from probability distributions p and q respectively. In standard transfer learning, we have $p \neq q$; and in partial transfer learning, we further have $p_{\mathcal{C}_t} \neq q$, where $p_{\mathcal{C}_t}$ denotes the distribution of the source domain labeled data belonging to label space $\mathcal{C}_t$. The goal of this paper is to design a deep neural network that enables learning of transfer features $\mathbf{f} = G_f(\mathbf{x})$ and adaptive classifier $y = G_y(\mathbf{f})$ to bridge the cross-domain discrepancy, such that the target risk $\Pr_{(\mathbf{x},y)\sim q}[G_y(G_f(\mathbf{x})) \neq y]$ is minimized by leveraging the source domain supervision.

In standard transfer learning, one of the main challenges is that the target domain has no labeled data and thus the source classifier G_y trained on source domain $\mathcal{D}_s$ cannot be directly applied to target domain $\mathcal{D}_t$ due to the distribution discrepancy of $p \neq q$. In partial transfer learning, another

more difficult challenge is that we even do not know which part of the source domain label space $\mathcal{C}_s$ is shared with the target domain label space $\mathcal{C}_t$ because $\mathcal{C}_t$ is not accessible during training, which results in two technical difficulties. On one hand, the source domain labeled data belonging to *outlier* label space $\mathcal{C}_s \backslash \mathcal{C}_t$ will cause negative transfer effect to the overall transfer performance. Existing deep transfer learning methods [15, 6, 29, 17] generally assume source domain and target domain have the same label space and match the whole distributions p and q, which are prone to negative transfer since the source and target label spaces are different and thus cannot be matched in principle. Thus, how to eliminate or at least decrease the influence of the source labeled data in outlier label space $\mathcal{C}_s \backslash \mathcal{C}_t$ is the key to alleviating negative transfer. On the other hand, reducing the distribution discrepancy between $p_{\mathcal{C}_t}$ and q is crucial to enabling knowledge transfer in the shared label space $\mathcal{C}_t$. These challenges should be tackled by filtering out the negative influence of unrelated part of source domain and at the same time enabling effective transfer learning between related part of source domain and target domain.

In summary, there are two essential challenges to enabling partial transfer learning. **(1)** Circumvent negative transfer by filtering out the unrelated source labeled data belonging to the outlier label space $\mathcal{C}_s \backslash \mathcal{C}_t$. **(2)** Promote positive transfer by maximally matching the data distributions $p_{\mathcal{C}_t}$ and q in the shared label space $\mathcal{C}_t$. We propose a novel selective adversarial network to address both challenges.

3.1. Domain Adversarial Network

Domain adversarial networks have been successfully applied to transfer learning [6, 29] by extracting transferable features that can reduce the distribution shift between the source domain and the target domain. The adversarial learning procedure is a two-player game, where the first player is the domain discriminator G_d trained to distinguish the source domain from the target domain, and the second player is the feature extractor G_f fine-tuned simultaneously to confuse the domain discriminator.

To extract domain-invariant features $\mathbf{f}$, the parameters θ_f of feature extractor G_f are learned by maximizing the loss of domain discriminator G_d, while the parameters θ_d of domain discriminator G_d are learned by minimizing the loss of the domain discriminator. In addition, the loss of label predictor G_y is also minimized. The objective of domain adversarial network [6] is the functional:

$$C_0\left(\theta_f, \theta_y, \theta_d\right) = \frac{1}{n_s} \sum_{\mathbf{x}_i \in \mathcal{D}_s} L_y\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right), y_i\right) - \frac{\lambda}{n_s + n_t} \sum_{\mathbf{x}_i \in \left(\mathcal{D}_s \cup \mathcal{D}_t\right)} L_d\left(G_d\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right) \tag{1}$$

where λ is a trade-off parameter between the two objectives that shape the features during learning. After training convergence, the parameters $\hat{\theta}_f$, $\hat{\theta}_y$, $\hat{\theta}_d$ will deliver a saddle point of the functional (1):

$$\begin{aligned} \left(\hat{\theta}_f, \hat{\theta}_y\right) &= \arg \min_{\theta_f, \theta_y} C_0\left(\theta_f, \theta_y, \theta_d\right), \\ \left(\hat{\theta}_d\right) &= \arg \max_{\theta_d} C_0\left(\theta_f, \theta_y, \theta_d\right). \end{aligned} \tag{2}$$

Domain adversarial networks are among the top-performing architectures for standard transfer learning where the source domain label space and target domain label space are the same, $\mathcal{C}_s = \mathcal{C}_t$.

3.2. Selective Adversarial Network

In partial transfer learning, however, the target domain label space is a subset of the source domain label space, $\mathcal{C}_t \subset \mathcal{C}_s$. Thus, matching the whole source domain distribution p and target domain distribution q will result in negative transfer caused by the *outlier* label space $\mathcal{C}_s \backslash \mathcal{C}_t$. The larger the outlier label space $\mathcal{C}_s \backslash \mathcal{C}_t$ compared to the target label space $\mathcal{C}_t$, the severer the negative transfer effect will be. To combat *negative transfer*, we should find a way to select out the outlier source classes as well as the associated source labeled data in $\mathcal{C}_s \backslash \mathcal{C}_t$ when performing domain adversarial adaptation.

To match the source and target domains of different label spaces $\mathcal{C}_s \neq \mathcal{C}_t$, we need to split the domain discriminator G_d in Equation (1) into $|\mathcal{C}_s|$ class-wise domain discriminators $G_d^k, k = 1, \ldots, |\mathcal{C}_s|$, each is responsible for matching the source and target domain data associated with label k, as shown in Figure 2. Since the target label space $\mathcal{C}_t$ is inaccessible during training while the target domain data are fully unlabeled, it is not easy to decide which domain discriminator G_d^k is responsible for each target data point. Fortunately, we observe that the output of the label predictor $\hat{\mathbf{y}}_i = G_y(\mathbf{x}_i)$ to each data point $\mathbf{x}_i$ is a probability distribution over the source label space $\mathcal{C}_s$. This distribution well characterizes the probability of assigning $\mathbf{x}_i$ to each of the $|\mathcal{C}_s|$ classes. Therefore, it is natural to use $\hat{\mathbf{y}}_i$ as the probability to assign each data point $\mathbf{x}_i$ to the $|\mathcal{C}_s|$ domain discriminators $G_d^k, k = 1, \ldots, |\mathcal{C}_s|$. The assignment of each point $\mathbf{x}_i$ to different discriminators can be implemented by a *probability-weighted* domain discriminator loss for all $|\mathcal{C}_s|$ domain discriminators $G_d^k, k = 1, \ldots, |\mathcal{C}_s|$ as follows,

$$L_d' = \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \sum_{\mathbf{x}_i \in \mathcal{D}_s \cup \mathcal{D}_t} \hat{y}_i^k L_d^k\left(G_d^k\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right), \tag{3}$$

where G_d^k is the k-th domain discriminator while L_d^k is its cross-entropy loss, and d_i is the domain label of point $\mathbf{x}_i$. Compared with the single-discriminator domain adversarial network in Equation (1), the proposed multi-discriminator

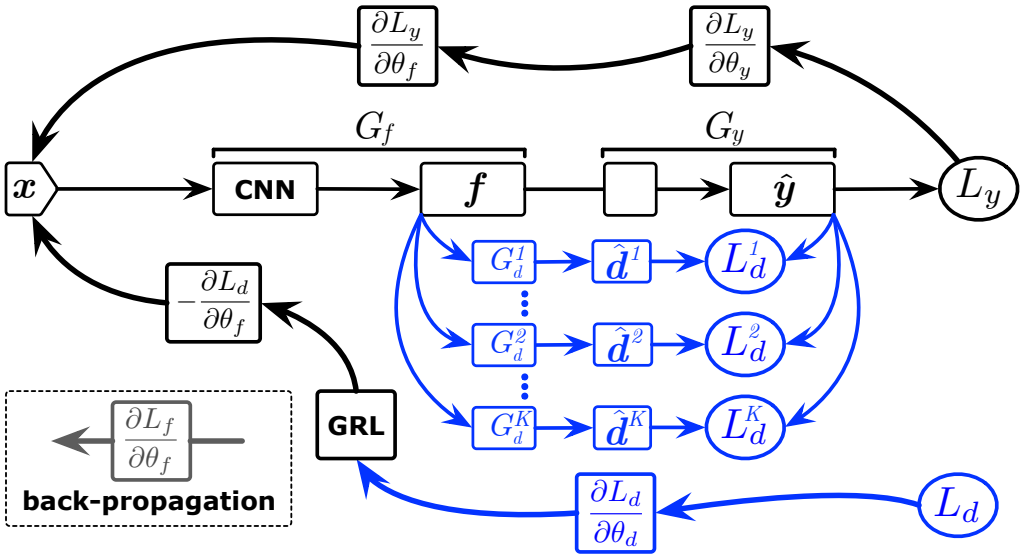


Figure 2. The architecture of the proposed Selective Adversarial Networks (SAN) for partial transfer learning, where $\mathbf{f}$ is the extracted deep features, $\hat{\mathbf{y}}$ is the predicted data label, and $\hat{\mathbf{d}}$ is the predicted domain label; G_f is the feature extractor, G_y and L_y are the label predictor and its loss, G_d^k and L_d^k are the domain discriminator and its loss; GRL stands for Gradient Reversal Layer. The blue part shows the class-wise adversarial networks ($|\mathcal{C}_s|$ in total) designed in this paper. *Best viewed in color.*

domain adversarial network enables fine-grained adaptation where each data point $\mathbf{x}_i$ is matched only by those relevant domain discriminators according to its probability $\hat{\mathbf{y}}_i$. This fine-grained adaptation may introduce three benefits. **(1)** It avoids the hard assignment of each point to only one domain discriminator, which tends to be inaccurate for target domain data. **(2)** It circumvents negative transfer since each point is only aligned to one or several most relevant classes, while the irrelevant classes are filtered out by the probability-weighted domain discriminator loss. **(3)** The probability-weighted domain discriminator loss puts different losses to different domain discriminators, which naturally learns multiple domain discriminators with different parameters θ_d^k; these domain discriminators with different parameters can promote *positive transfer* for each instance.

Besides the *instance-level* weighting mechanism described above, we introduce another *class-level* weighting method to further remove the negative influence of outlier source classes $\mathcal{C}_s \backslash \mathcal{C}_t$ and the associated source data. We observe that only the domain discriminators responsible for the target classes $\mathcal{C}_t$ are effective for promoting *positive transfer*, while the other discriminators responsible for the outlier source classes $\mathcal{C}_s \backslash \mathcal{C}_t$ only introduce noises and deteriorate the positive transfer between the source domain and the target domain in the shared label space $\mathcal{C}_t$. Therefore, we need to down-weight the domain discriminators responsible for the outlier source classes, which can be implemented by class-level weighting of these domain discriminators. Since target data are not likely to belong to the outlier source classes, their probabilities $y_i^k, k \in \mathcal{C}_s \backslash \mathcal{C}_t$ are also sufficiently small. Thus, we can down-weight the domain discriminators responsible for the outlier source classes as follows,

$$L_d = \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \left\{ \left(\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k \right) \times \left(\sum_{\mathbf{x}_i \in (\mathcal{D}_s \cup \mathcal{D}_t)} \hat{y}_i^k L_d^k \left(G_d^k \left(G_f \left(\mathbf{x}_i \right) \right), d_i \right) \right) \right\}, \tag{4}$$

where $\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k$ is the class-level weight for class k, which is small for the outlier source classes.

Although the multiple domain discriminators introduced in Equation (4) can selectively transfer relevant knowledge to target domain by decreasing the negative influence of outlier source classes $\mathcal{C}_s \backslash \mathcal{C}_t$ and by effectively transferring knowledge of shared label space $\mathcal{C}_t$, it highly depends on the probability $\hat{\mathbf{y}}_i = G_y(\mathbf{x}_i)$. Thus, we further refine the label predictor G_y by exploiting the entropy minimization principle [9] which encourages low-density separation between classes. This criterion is implemented by minimizing the conditional-entropy E of probability $\hat{y}_i^k$ on target domain $\mathcal{D}_t$ as

$$E = \frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} H \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right) \right) \tag{5}$$

where $H(\cdot)$ is the conditional-entropy loss functional $H \left(G_y \left(G_f \left(\mathbf{x}_i \right) \right) \right) = - \sum_{k=1}^{|\mathcal{C}_s|} \hat{y}_i^k \log \hat{y}_i^k$. By minimizing entropy (5), the label predictor $G_y(\mathbf{x}_i)$ can directly access

target unlabeled data and will amend itself to pass through the target low-density regions to give more accurate probability $\hat{\mathbf{y}}_i$.

Integrating all things together, the objective of the proposed Selective Adversarial Network (SAN) is

$$\begin{aligned} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right) = & \frac{1}{n_s} \sum_{\mathbf{x}_i \in \mathcal{D}_s} L_y\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right), y_i\right) \\ & + \frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} H\left(G_y\left(G_f\left(\mathbf{x}_i\right)\right)\right) \\ & - \frac{1}{n_s + n_t} \sum_{k=1}^{|\mathcal{C}_s|} \left\{ \left(\frac{1}{n_t} \sum_{\mathbf{x}_i \in \mathcal{D}_t} \hat{y}_i^k \right) \right. \\ & \left. \times \left(\sum_{\mathbf{x}_i \in (\mathcal{D}_s \cup \mathcal{D}_t)} \hat{y}_i^k L_d^k\left(G_d^k\left(G_f\left(\mathbf{x}_i\right)\right), d_i\right) \right) \right\} \end{aligned} \tag{6}$$

where λ is a hyper-parameter that trade-offs the two objectives in the unified optimization problem. The optimization problem is to find the parameters $\hat{\theta}_f$, $\hat{\theta}_y$ and $\hat{\theta}_d^k (k = 1, 2, ..., |\mathcal{C}_s|)$ that satisfy

$$\begin{aligned} (\hat{\theta}_f, \hat{\theta}_y) &= \arg \min_{\theta_f, \theta_y} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right), \\ (\hat{\theta}_d^1, ..., \hat{\theta}_d^{|\mathcal{C}_s|}) &= \arg \max_{\theta_d^1, ..., \theta_d^{|\mathcal{C}_s|}} C\left(\theta_f, \theta_y, \theta_d^k|_{k=1}^{|\mathcal{C}_s|}\right). \end{aligned} \tag{7}$$

The selective adversarial network (SAN) successfully enables partial transfer learning, which simultaneously circumvents negative transfer by filtering out outlier source classes $\mathcal{C}_s \backslash \mathcal{C}_t$, and promotes positive transfer by maximally matching the data distributions $p_{\mathcal{C}_t}$ and q in the shared label space $\mathcal{C}_t$.

4. Experiments

We conduct experiments on three benchmark datasets to evaluate the efficacy of our approach against several state-of-the-art deep transfer learning methods. Codes and datasets will be available online.

4.1. Setup

The evaluation is conducted on three public datasets: Office-31, Caltech-Office and ImageNet-Caltech.

Office-31 [24] is a standard benchmark for domain adaptation in computer vision, consisting of 4,652 images and 31 categories collected from three distinct domains: *Amazon* (**A**), which contains images downloaded from amazon.com, *Webcam* (**W**) and *DSLR* (**D**), which contain images taken by web camera and digital SLR camera with different settings, respectively. We denote the three domains with 31 categories as **A 31**, **W 31** and **D 31**. Then we use the ten categories shared by *Office-31* and *Caltech-256* and select images of these ten categories in each domain of *Office-31* as target domains, denoted as **A 10**, **W 10** and **D 10**. We evaluate all methods across six transfer tasks **A 31** $\rightarrow$ **W 10**, **D 31** $\rightarrow$ **W 10**, **W 31** $\rightarrow$ **D 10**, **A 31** $\rightarrow$ **D 10**, **D 31** $\rightarrow$ **A 10** and **W 31** $\rightarrow$ **A 10**. These tasks represent the performance on the setting where both source and target domains have small number of classes.

Caltech-Office [8] is built by using *Caltech-256* (**C 256**) [11] as source domain and the three domains in *Office 31* as target domains. We use the ten categories shared by *Caltech-256* and *Office-31* and select images of these ten categories in each domain of *Office-31* as target domains [8, 16, 26]. Denoting source domains as **C 256**, we can build 3 transfer tasks: **C 256** $\rightarrow$ **W 10**, **C 256** $\rightarrow$ **A 10** and **C 256** $\rightarrow$ **D 10**. This setting aims to test the performance of different methods on the task setting where source domain has much more classes than the target domain.

ImageNet-Caltech is constructed with *ImageNet-1K* [23] dataset containing 1000 classes and *Caltech-256* containing 256 classes. They share 84 common classes, thus we form two transfer learning tasks: **ImageNet 1000** $\rightarrow$ **Caltech 84** and **Caltech 256** $\rightarrow$ **ImageNet 84**. To prevent the effect of the pre-trained model on ImageNet, we use ImageNet validation set when ImageNet is used as target domain and ImageNet training set when ImageNet is used as source domain. This setting represents the performance on tasks with large number of classes in both source and target domains.

We compare the performance of **SAN** with state of the art transfer learning and deep learning methods: Convolutional Neural Network (**AlexNet** [14]), Deep Adaptation Network (**DAN**) [15], Reverse Gradient (**RevGrad**) [6], Residual Transfer Networks (**RTN**) [17], and Adversarial Discriminative Domain Adaptation (**ADDA**) [27]. DAN learns transferable features by embedding deep features of multiple task-specific layers to reproducing kernel Hilbert spaces (RKHSs) and matching different distributions optimally using multi-kernel MMD. RevGrad improves domain adaptation by making the source and target domains indistinguishable for a discriminative domain classifier via an adversarial training paradigm. RTN jointly learns transferable features and adapts different source and target classifiers via deep residual learning [12]. ADDA combines discriminative modeling, untied weight sharing, and a GAN loss to yield much better results than RevGrad. All prior methods do not address partial transfer learning where the target label space is a subspace of the source label space. To test SAN on different base-networks, we also compare different methods on **VGG-16** [25]. To go deeper with the efficacy of selective mechanism and entropy minimization, we perform ablation study by evaluating two variants of **SAN**: (1) **SAN-selective** is the variant without selective mechanism, which has the same model complexity as AlexNet; (2) **SAN-entropy** is the variant without entropy minimization, which has the same

Table 1. Accuracy (%) of partial transfer learning tasks on *Office-31*

Method	Office-31						
	A 31 → W 10	D 31 → W 10	W 31 → D 10	A 31 → D 10	D 31 → A 10	W 31 → A 10	Avg
AlexNet [14]	58.51	95.05	98.08	71.23	70.6	67.74	76.87
DAN [15]	56.52	71.86	86.78	51.86	50.42	52.29	61.62
RevGrad [6]	49.49	93.55	90.44	49.68	46.72	48.81	63.11
RTN [17]	66.78	86.77	99.36	70.06	73.52	76.41	78.82
ADDA [27]	70.68	96.44	98.65	72.90	74.26	75.56	81.42
SAN-selective	71.51	98.31	100.00	78.34	77.87	76.32	83.73
SAN-entropy	74.61	98.31	100.00	80.29	78.39	82.25	85.64
SAN	**80.02**	**98.64**	**100.00**	**81.28**	**80.58**	**83.09**	**87.27**

Table 2. Accuracy (%) of partial transfer learning tasks on *Caltech-Office* and *ImageNet-Caltech*

Method	Caltech-Office				ImageNet-Caltech		
	C 256 → W 10	C 256 → A 10	C 256 → D 10	Avg	I 1000 → C 84	C 256 → I 84	Avg
AlexNet [14]	58.44	76.64	65.86	66.98	52.37	47.35	49.86
DAN [15]	42.37	70.75	47.04	53.39	54.21	52.03	53.12
RevGrad [6]	54.57	72.86	57.96	61.80	51.34	47.02	49.18
RTN [17]	71.02	81.32	62.35	71.56	63.69	50.45	57.07
ADDA [27]	73.66	78.35	74.80	75.60	64.20	51.55	57.88
SAN-selective	76.44	81.63	80.25	79.44	66.78	51.25	59.02
SAN-entropy	72.54	78.95	76.43	75.97	55.27	52.31	53.79
SAN	**88.33**	**83.82**	**85.35**	**85.83**	**68.45**	**55.61**	**62.03**

model complexity as SAN.

We follow standard protocols and use all labeled source data and all unlabeled target data for unsupervised transfer learning [24, 15]. We compare average classification accuracy of each transfer task using three random experiments. For MMD-based methods (DAN and RTN), we use Gaussian kernel with bandwidth b set to median pairwise squared distances on training data, i.e. median heuristic [10]. For all methods, we perform transfer cross-validation [33] on labeled source data and unlabeled target data to select their hyper-parameters.

We implement all deep methods based on the Caffe deep-learning framework, and fine-tune from Caffe-provided models of AlexNet [14] pre-trained on ImageNet. We add a bottleneck layer between the $fc7$ and $fc8$ layers as RevGrad [6] except for the task **ImageNet 1000 → Caltech 84** since the pre-trained model is trained on ImageNet dataset and it can fully exploit the advantage of pre-trained model with the original $fc7$ and $fc8$ layer. For SAN, we fine-tune all the feature layers and train the bottleneck layer, the classifier layer and the adversarial networks. Since these new layers and networks are trained from scratch, we set their learning rate to be 10 times that of the other layers. We use mini-batch stochastic gradient descent (SGD) with momentum of 0.9 and the learning rate annealing strategy implemented in RevGrad [6]: the learning rate is adjusted during SGD using the following formula: $\eta_p = \frac{\eta_0}{(1+\alpha p)^\beta}$, where p is the training progress linearly changing from 0 to 1, $\eta_0 = 0.001$, $\alpha = 10$ and $\beta = 0.75$, which is optimized for low error on the source domain. As **SAN** can work stably across different transfer tasks, the penalty of adversarial networks is increased from 0 to 1 gradually as RevGrad [6]. All the hyper-parameters of the learning rate and penalty strategies are selected through transfer cross-validation [33] on the labeled source data and unlabeled target data.

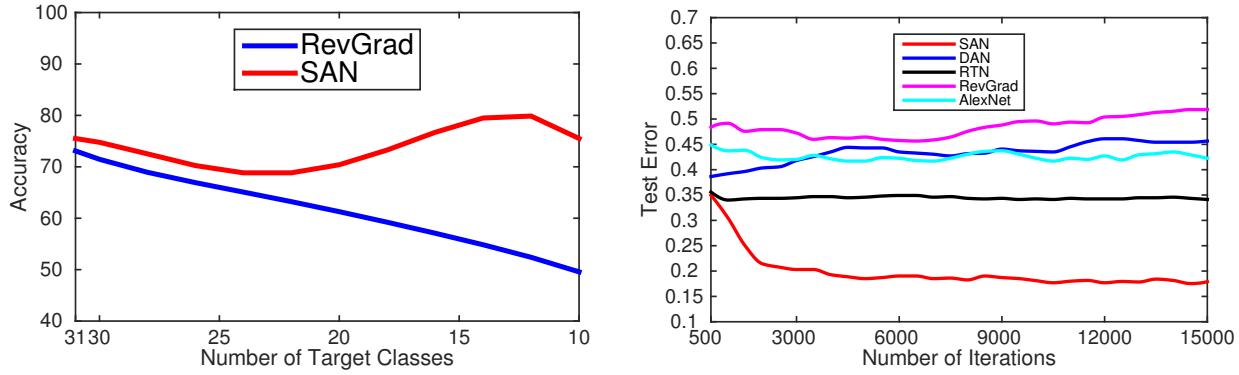
4.2. Results

The classification results on the six tasks of *Office-31*, the three tasks of *Caltech-Office* and the two tasks of *ImageNet-Caltech* are shown in Table 1 and 2. The SAN model outperforms all comparison methods on all the tasks. In particular, SAN substantially improves the accuracy by huge margins on tasks with small source domain and small target domain, e.g. **A 31 → W 10** , **A 31 → D 10**, and tasks with large source domain and small target domain, e.g. **C 31 → W 10**. And it achieves considerable accuracy gains on tasks with large-scale source domain and target domain, e.g. **I 1000 → C 84**. These results suggest that SAN can learn transferable features for partial transfer learning in all the tasks under the setting where the target label space is a subspace of the source label space.

The results reveal several interesting observations. **(1)** Previous deep transfer learning methods including those based on adversarial-network like RevGrad and those based on MMD like DAN perform worse than standard AlexNet, which demonstrates the influence of negative transfer effect. These methods try to transfer knowledge from all classes of source domain to target domain but there are classes in

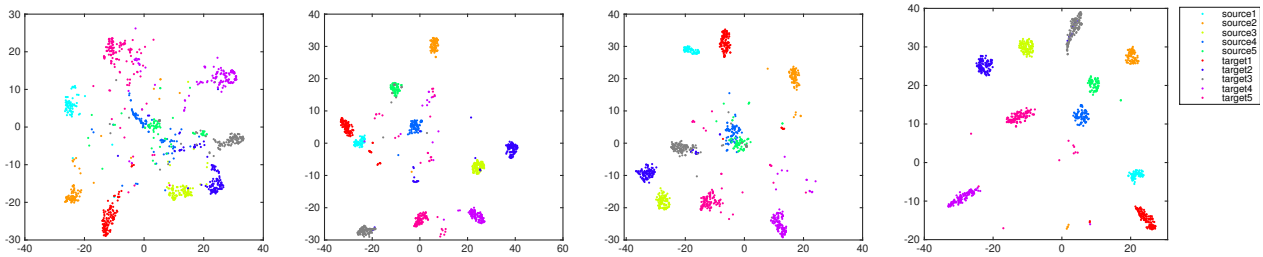
Table 3. Accuracy (%) based on VGG-16 on *Office-31*

Method	A 31→W 10	D 31→W 10	W 31→D 10	A 31→D 10	D 31→A 10	W 31→A 10	Avg
VGG [25]	60.34	97.97	99.36	76.43	72.96	79.12	81.03
DAN [15]	58.78	85.86	92.78	54.76	55.42	67.29	69.15
RevGrad [6]	50.85	95.23	94.27	57.96	51.77	62.32	68.73
RTN [17]	69.35	98.42	99.59	75.43	81.45	82.98	84.54
ADDA [27]	72.85	98.42	99.59	77.96	84.77	85.32	86.49
SAN	83.39	99.32	100	90.70	87.16	91.85	92.07



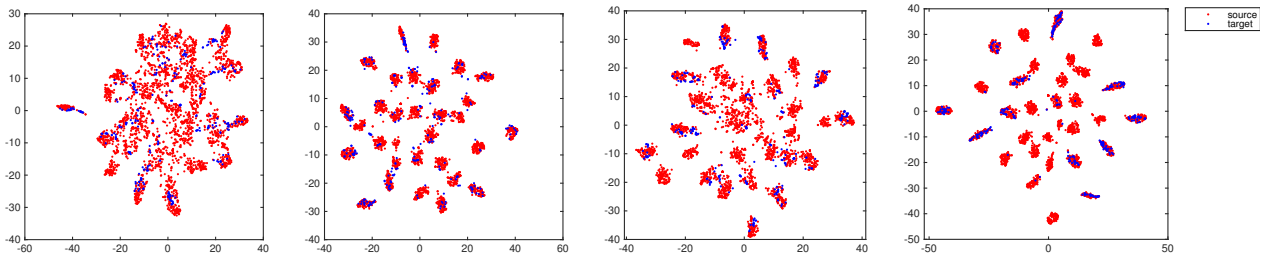
(a) Accuracy w.r.t #Target Classes (b) Test Error

Figure 3. Empirical analysis: (a) Accuracy by varying #target domain classes; (b) Target test error.



(a) DAN (b) RevGrad (c) RTN (d) SAN

Figure 4. The t-SNE visualization of DAN, RevGrad, RTN, and SAN with class information.



(a) DAN (b) RevGrad (c) RTN (d) SAN

Figure 5. The t-SNE visualization of DAN, RevGrad, RTN, and SAN with domain information.

source domain that do not exist in the target domain, a.k.a. outlier source data. Fooling the adversarial network to match the distribution of outlier source data and target data will make the classifier more likely to classify target data in these outlier classes, which is prone to negative transfer. Thus these previous methods perform even worse than standard

AlexNet. However, SAN outperforms them by large margins, indicating that SAN can effectively avoid negative transfer by eliminating the outlier source classes irrelevant to target domain. **(2)** RTN performs better than AlexNet because it executes entropy minimization criterion which can avoid the impact of outlier source data to some degree. But comparing RTN with SAN-selective which only has entropy minimization loss, we observe that SAN-selective outperforms RTN in most tasks, demonstrating that RTN also suffers from negative transfer effect and even the residual branch of RTN cannot learn the large discrepancy between source and target domain. **(3)** ADDA first learns a discriminative representation using the labels in the source domain and then a separate encoding that maps the target data to the same space using an asymmetric mapping learned through a domain-adversarial loss. By combining discriminative modeling, untied weight sharing, and a GAN loss, ADDA yields much better results than RevGrad and RTN. SAN outperforms ADDA in all the tasks, proving that our selective adversarial mechanism can jointly promote positive transfer from relevant source domain data to target domain and circumvent negative transfer from outlier source domain data to target domain. From Table 3, we can observe that SAN outperforms all the other methods on VGG-16 network, which demonstrates that SAN can generalize to different base networks. We go deeper into different modules of SAN by comparing the results of SAN variants in Tables 1 and 2. **(1)** SAN outperforms SAN-selective, proving that using selective adversarial mechanism can selectively transfer knowledge from source data to target data. It can successfully select the source data belonging to the classes shared with target classes by the corresponding domain discriminators. **(2)** SAN outperforms SAN-entropy especially in tasks where source and target domains have very large distribution gap in terms of the different numbers of classes, e.g. **I 1000** $\rightarrow$ **C 84**. Entropy minimization can effectively decrease the probability of predicting each point to irrelevant classes especially when there are a large number of irrelevant classes, which can in turn boost the performance of the selective adversarial mechanism. This explains the improvement from SAN-entropy to SAN.

4.3. Analysis

Accuracy for Different Numbers of Target Classes: We investigate a wider spectrum of partial transfer learning by varying the number of target classes. Figure 3(a) shows that when the number of target classes decreases, the performance of RevGrad degrades quickly, meaning that negative transfer becomes severer when the domain gap is enlarged. The performance of SAN degenerates when the number of target classes decreases from 31 to 20, where negative transfer problem arises but the transfer problem itself is still hard; the performance of SAN increases when the number of target classes decreases from 20 to 10, where the transfer problem itself becomes easier. The margin that SAN outperforms RevGrad becomes larger when the number of target classes decreases. SAN also outperforms RevGrad in standard transfer learning setting when the number of target classes is 31.

Convergence Performance: We examine the convergence of SAN by studying the test error through training process. As shown in Figure 3(b), the test errors of DAN and RevGrad are increasing due to negative transfer. RTN converges very fast depending on the entropy minimization, but converges to a higher test error than SAN. SAN converges fast and stably to a lowest test error, meaning it can be trained efficiently and stably to enable positive transfer and alleviate negative transfer simultaneously.

Feature Visualization: We visualize the t-SNE embeddings [4] of the bottleneck representations by DAN, RevGrad, RTN and SAN on transfer task **A 31** $\rightarrow$ **W 10** in Figures 4(a)–4(d) (with class information) and Figures 5(a)–5(d) (with domain information). We randomly select five classes in the source domain not shared with target domain and five classes shared with target domain. We can make intuitive observations. **(1)** Figure 4(a) shows that the bottleneck features are mixed together, meaning that DAN cannot discriminate both source and target data very well; Figure 5(a) shows that the target data are aligned to all source classes including those outlier ones, which embodies the negative transfer issue. **(2)** Figures 4(b)– 4(c) show that both RevGrad and RTN discriminate the source domain well but the features of most target data are very close to source data even to the wrong source classes; Figures 5(b)– 5(c) further indicate that both RevGrad and RTN tend to draw target data close to all source classes even to those not existing in target domain. Thus, their performance on target data degenerates due to negative transfer. **(3)** Figures 4(d) and 5(d) demonstrate that SAN can discriminate different classes in both source and target while the target data are close to the right source classes, while the outlier source classes cannot influence the target classes. These promising results demonstrate the efficacy of both selective adversarial adaptation and entropy minimization.

5. Conclusion

This paper presented a novel selective adversarial network approach to partial transfer learning. Unlike previous adversarial adaptation methods that match the whole source and target domains based on the shared label space assumption, the proposed approach simultaneously circumvents negative transfer by selecting out the outlier source classes and promotes positive transfer by maximally matching the data distributions in the shared label space. Our approach successfully tackles partial transfer learning where source label space subsumes target label space, which is testified by extensive experiments.

References

[1] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *MLJ*, 79(1-2):151–175, 2010. 2

[2] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *TPAMI*, 35(8):1798–1828, 2013. 2

[3] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *JMLR*, 12:2493–2537, 2011. 2

[4] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014. 1, 8

[5] L. Duan, I. W. Tsang, and D. Xu. Domain transfer multiple kernel learning. *TPAMI*, 34(3):465–479, 2012. 2

[6] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17:59:1–59:35, 2016. 1, 2, 3, 5, 6, 7

[7] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*, 2011. 2

[8] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, 2012. 2, 5

[9] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. In *NIPS*, pages 529–536, 2004. 4

[10] A. Gretton, B. Sriperumbudur, D. Sejdinovic, H. Strathmann, S. Balakrishnan, M. Pontil, and K. Fukumizu. Optimal kernel choice for large-scale two-sample tests. In *NIPS*, 2012. 6

[11] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical report, California Institute of Technology, 2007. 5

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[13] J. Hoffman, S. Guadarrama, E. Tzeng, R. Hu, J. Donahue, R. Girshick, T. Darrell, and K. Saenko. LSDA: Large scale detection through adaptation. In *NIPS*, 2014. 2

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 5, 6

[15] M. Long, Y. Cao, J. Wang, and M. I. Jordan. Learning transferable features with deep adaptation networks. In *ICML*, 2015. 1, 2, 3, 5, 6, 7

[16] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu. Transfer feature learning with joint distribution adaptation. In *ICCV*, 2013. 5

[17] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016. 1, 2, 3, 5, 6, 7

[18] Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *COLT*, 2009. 2

[19] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In *ICML*, 2011. 2

[20] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, June 2013. 2

[21] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *TNNLS*, 22(2):199–210, 2011. 2

[22] S. J. Pan and Q. Yang. A survey on transfer learning. *TKDE*, 22(10):1345–1359, 2010. 1, 2

[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 5

[24] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010. 2, 5, 6

[25] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR, 2015 (arXiv:1409.1556v6)*, 2015. 5, 7

[26] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. In *AAAI*, 2016. 5

[27] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017. 5, 6, 7

[28] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. 2014. 1, 2

[29] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Simultaneous deep transfer across domains and tasks. In *ICCV*, 2015. 1, 2, 3

[30] X. Wang and J. Schneider. Flexible transfer learning under support and model shift. In *NIPS*, 2014. 2

[31] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *NIPS*, 2014. 1, 2

[32] K. Zhang, B. Schölkopf, K. Muandet, and Z. Wang. Domain adaptation under target and conditional shift. In *ICML*, 2013. 2

[33] E. Zhong, W. Fan, Q. Yang, O. Verscheure, and J. Ren. Cross validation framework to choose amongst models and datasets for transfer learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD)*, pages 547–562. Springer, 2010. 6