Deep Reinforcement Learning Boosted Partial Domain Adaptation

Keyu Wu[1] , **Min Wu**[1] , **Jianfei Yang**[2] , **Zhenghua Chen**[1*] , **Zhengguo Li**[1] and **Xiaoli Li**[1,2]
[1]Institute for Infocomm Research , A*STAR, Singapore
[2]Nanyang Technological University
{wu_keyu, wumin}@i2r.a-star.edu.sg, {yang0478, chen0832}@e.ntu.edu.sg, {ezgli, xlli}@i2r.a-star.edu.sg

Abstract

Domain adaptation is critical for learning transferable features that effectively reduce the distribution difference among domains. In the era of big data, the availability of large-scale labeled datasets motivates partial domain adaptation (PDA) which deals with adaptation from large source domains to small target domains with less number of classes. In the PDA setting, it is crucial to transfer relevant source samples and eliminate irrelevant ones to mitigate negative transfer. In this paper, we propose a deep reinforcement learning based source data selector for PDA, which is capable of eliminating less relevant source samples automatically to boost existing adaptation methods. It determines to either keep or discard the source instances based on their feature representations so that more effective knowledge transfer across domains can be achieved via filtering out irrelevant samples. As a general module, the proposed DRL-based data selector can be integrated into any existing domain adaptation or partial domain adaptation models. Extensive experiments on several benchmark datasets demonstrate the superiority of the proposed DRL-based data selector which leads to state-of-the-art performance for various PDA tasks.

1 Introduction

Deep neural networks have achieved remarkable performance on a variety of machine learning tasks. Generally, in learning theories, it is assumed that training and testing data are from the same distribution. However, data annotation for every new task can be impractical due to the prohibitive cost. Instead, it is preferable to train a model by leveraging massively available labeled data from related yet distinct domains. Therefore, domain adaptation (DA) techniques have been introduced to bridge domains of different distributions and enable knowledge transfer from label-rich source domain to label-scarce target domain.

Deep domain adaptation methods attempt to improve the learning of domain-invariant feature representations by embedding distribution matching modules into the network architectures. So far, a variety of distribution similarity measures have been incorporated into the network architecture to learn transferable representations [Long *et al.*, 2017]. Meanwhile, a collection of adversarial learning based domain adaptation methods have been proposed, which aligns domain distributions via minimizing an approximate discrepancy in an adversarial training setting [Ganin and Lempitsky, 2014].

Currently, most deep domain adaptation methods assume that the source and target domains share an identical label space. Nevertheless, in real-world applications, it is often nontrivial to find source domains with the same label spaces as target domains of interest. Instead, due to the availability of large-scale labelled datasets, such as ImageNet [Russakovsky *et al.*, 2015], a more practical yet more challenging scenario is referred to as partial domain adaptation (PDA), which relaxes the constraint of shared label spaces. It enables knowledge transfer from source domains with more classes to target domains with fewer classes without any knowledge on the size and categories of the target classes.

The PDA problems cannot be addressed by simply aligning the two domains because the knowledge migrated from irrelevant source classes can result in negative transfer. To mitigate negative interference caused by irrelevant source samples, several pioneering PDA methods have been proposed to upweight relevant source instances while down-weighting outlier source samples in domain adversarial networks [Cao *et al.*, 2018a; Cao *et al.*, 2018b]. In recent years, deep reinforcement learning (DRL) methods have also been implemented to learn source data selection policies. The Reinforced Transfer Network (RTNet) proposed in [Chen *et al.*, 2020b] adopts an actor-critic algorithm [Konda and Tsitsiklis, 2000] to optimize the source data selector, while the Domain Adversarial Reinforcement Learning (DARL) framework proposed in [Chen *et al.*, 2020a] utilizes Deep Q-Network (DQN) [Mnih *et al.*, 2015] to learn source data selection policies.

In this paper, we propose a novel deep reinforcement learning based source data selector (DRL-DS) to boost knowledge transfer in partial domain adaptation. The DRL-based data selector is developed to automatically filter out irrelevant source samples based on their transferability. Since forcefully aligning target classes to outlier source classes can lead to negative transfer [Cao *et al.*, 2018a], only the selected source samples will be used to train an adaptive classifier. During

*Corresponding Author

training, the DRL-DS model determines whether to select the source samples based on their feature representations. In the meantime, a (partial) domain adaptation model is trained to reduce the domain shift between the target and selected source samples while it is also employed to calculate the reward signals for the DRL-DS module. In this way, the DRL and DA models are jointly trained to leverage each other's capability for better knowledge transfer across domains.

Compared to conventional PDA algorithms, the integration of the DRL-DS module can lead to improved knowledge transfer from one domain to another through combining RL and DA techniques. Moreover, our DRL-based source data selector also addresses the limitations of existing DRL-based PDA algorithms. Since RTNet utilizes an on-policy DRL algorithm, it has to re-collect new samples every time the policy is updated. Besides, it also requires additional generators to calculate reconstruction error based rewards. In contrary, we employs the Dueling Double Deep Q-Network [Wang *et al.*, 2016], which is an off-policy DRL algorithm, to learn the source data selection policies in a more efficient way. Moreover, we also design the reward functions meticulously so that better performance can be achieved without any additional network block. Although DARL also employs an off-policy DRL algorithm, it relies on domain adversarial learning to calculate rewards. In contrast, the proposed DRL-DS module is a general paradigm that can be integrated into any (partial) domain adaptation method.

The main contributions of our work are as follows:

- We have proposed a novel deep reinforcement learning based source data selector to boost knowledge transfer in partial domain adaptation. The proposed DRL-based source data selector is capable of mitigating negative transfer through eliminating irrelevant samples automatically. Moreover, it is a general paradigm that can be integrated into any (partial) domain adaptation method.
- A novel reward function is meticulously designed to better guide the learning without introducing any additional network block or relying on any network architecture.
- Extensive experiments on benchmark datasets have demonstrated the remarkable superiority of the proposed DRL-based source data selector. By adopting it, the PDA methods can outperform the state-of-the-art approaches by a large margin.

2 Related Work

2.1 Domain Adaptation

Domain adaptation is a branch of transfer learning which bridges domains of different distributions. The key challenge of DA is to reduce the distribution shift across different domains. Recently, DA has been combined with deep neural networks to minimize the statistical discrepancies between the deep embeddings of source and target domains. To begin with, a variety of distribution similarity measures have been incorporated into the network architecture and minimized along with the standard source classification loss for end-to-end learning of transferable representations [Long *et al.*, 2015; Long *et al.*, 2016; Long *et al.*, 2017]. Distinctively, in [Xu *et al.*, 2019], it is revealed that features with larger norm are more transferable so that the Stepwise Adaptive Feature Norm (SAFN) method has been proposed to progressively enlarge feature norms of the two domains.

Another category of DA methods introduce domain classifiers to learn transferable features in adversarial settings. Generally, the domain discriminator is to differentiate features of different domains while the feature extractor is to deceive the domain discriminator [Ganin and Lempitsky, 2014; Tzeng *et al.*, 2017].

2.2 Partial Domain Adaptation

To address PDA problems, a number of pioneering methods have been proposed to up-weight relevant source instances while down-weighting outlier source samples [Cao *et al.*, 2018a; Cao *et al.*, 2018b; Zhang *et al.*, 2018; Li *et al.*, 2020b; Li *et al.*, 2020a; Jing *et al.*, 2020; Kim and Hong, 2021]. For instance, Selective Adversarial Network (SAN) [Cao *et al.*, 2018a] introduced multiple discriminators to realize fine-grained adaptation. It weighted the source instances according to their class probabilities predicted by the source classifier. In [Cao *et al.*, 2018b], Partial Adversarial Domain Adaptation (PADA) [Cao *et al.*, 2018b] used only one domain adversarial network and added the predicted target label probabilities to the source classifier as the class-level weights. Rather than using class-level weight, Importance Weighted Adversarial Nets (IWAN) [Zhang *et al.*, 2018] introduced two domain classifiers so that the outputs of a second domain classifier was adopted to weight source examples. Particularly, Example Transfer Network (ETN) [Cao *et al.*, 2019] calculated the weights of source samples according to their similarities based on a discriminative domain discriminator, and in the meantime down-weighted irrelevant source samples when updating the source classifier.

Recently, reinforcement learning algorithms have also been applied to learn source data selection policies for partial domain adaptation tasks. In [Chen *et al.*, 2020a], Chen et al. proposed the Domain Adversarial Reinforcement Learning (DARL) method to learn data selection policies automatically through DQN. The calculation of their reward signals relied on the domain adversarial learning framework. In [Chen *et al.*, 2020b], Chen et al. introduced the Reinforced transfer network (RTNet) to eliminate outlier source classes through a reinforced data selector while combining both high-level and pixel-level information. These two pioneering DRL-based PDA approaches have validated the feasibility of applying RL in the context of PDA. In this work, we propose a novel DRL-based source data selector to boost knowledge transfer in partial domain adaptation and adopt a Dueling Double DQN model to verify its feasibility.

3 Method

3.1 Problem Statement

The partial domain adaptation scenario constitutes source domain $\mathcal{D}_s = \{(\mathbf{x}_i^s, \mathbf{y}_i^s)\}_{i=1}^{n_s}$ which has n_s labeled examples associated with $|\mathcal{C}_s|$ classes, and the target domain $\mathcal{D}_t = \{\mathbf{x}_j^t\}_{j=1}^{n_t}$ which has n_t unlabelled examples associated with

$|\mathcal{C}_t|$ classes. The source domain subsumes the target domain so that $\mathcal{C}_t \subset \mathcal{C}_s$. Since the two domains are drawn from different probability distributions p and q, $p \neq q$ and $p_{\mathcal{C}_t} \neq q$, where $p_{\mathcal{C}_t}$ represents the distribution of source data in the target label space. Direct alignment of these two distributions can result in negative transfer because $p_{\mathcal{C}_s \setminus \mathcal{C}_t}$ and q are not overlapped, where $p_{\mathcal{C}_s \setminus \mathcal{C}_t}$ indicates the distribution of source instances in the outlier categories. Therefore, it is important to eliminate irrelevant source samples in PDA tasks while reducing the distribution shift between $p_{\mathcal{C}_t}$ and q.

The objective of this paper is to develop a DRL-based source data selector (DRL-DS) to boost knowledge transfer in PDA. As depicted in Fig. 1, the proposed DRL-DS module is a general paradigm that can be integrated into any (partial) domain adaptation method. In this paper, it is integrated into a domain adaptation method (SAFN) [Xu *et al.*, 2019] and a partial domain adaptation method (ETN) [Cao *et al.*, 2019] respectively to fully prove its effectiveness and generalization capability. The objective of the DRL-DS module is to learn data selection policies automatically so that negative transfer can be mitigated through eliminating outlier source samples. In the meantime, the (partial) domain adaptation module aims to learn domain-invariant feature representations.

3.2 DRL-DS Module

In this paper, the source data selection task is modeled as a Markov Decision Process (MDP) defined by a tuple $M = (S, A, R, P, \gamma)$, where S, A, R, P, γ denote the state space, action space, immediate reward, state transition function, and discount factor, respectively. At time step t, an action $a_t \in A$ is executed based on the current state $s_t \in S$. The DRL agent then receives a reward $R(s_t, a_t)$, and transits to a new state s_{t+1}. A policy $\pi(a|s)$ which specifies the mapping from a state s to an action a is assessed by the Q-value function defined as:

$$Q^{\pi}(s,a) = \mathbb{E}^{\pi}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) | s_0 = s, a_0 = a\right]. \quad (1)$$

The goal is then to maximize the expectation of cumulative reward, which can be solved by the Q-learning algorithm shown in the following:

$$Q^*(s_t, a_t) = R(s_t, a_t) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}). \quad (2)$$

Thereby, the optimal policy can be derived as $\pi^*(s) = \operatorname{argmax}_a Q^*(s,a)$.

In this paper, we employ the Dueling Double Deep Q-Network to figure out the optimal data selection policies and verify the feasibility of the proposed DRL-DS based PDA framework. Given a batch of source data $\{\mathbf{x}_i^s\}_{i=1}^{n_b} = \mathbf{X}_b^s$, where n_b and b denote the batch size and ID respectively, the feature representations of these source samples $F(\mathbf{X}_b^s)$ can be obtained, where F indicates the feature extractor in the (partial) domain adaptation module. The DRL-DS module then maps $F(\mathbf{X}_b^s)$ to a series of actions $\{a_i^s\}_{i=1}^{n_b} = A_b^s$, which in turn weights the source instances while updating the parameters of the (partial) domain adaptation network. Meanwhile, the rewards of the actions $\{R_i^s\}_{i=1}^{n_b}$ are determined according to the transferability of the source samples. In the following, the state, action, reward, and training of our DRL-DS module are introduced in detail.

State. Feed a batch of n_b source samples to the feature extractor of the (partial) domain adaptation module, a number of n_b features $F_t(\mathbf{X}_b^s) = [F_t(\mathbf{x}_1^s), \cdots, F_t(\mathbf{x}_{n_b}^s)]$ can be generated, where $F_t(\mathbf{x}_i^s)$ represents the feature vector extracted from source instance x_i^s at time step t. In the DRL-DS module, a state is defined as one source feature vector $F_t(\mathbf{x}_i^s)$ and at each state, an action is taken to determine whether to keep or discard the corresponding source sample. In this way, the weights of the n_b source instances can be derived via passing the n_b source feature vectors to the DRL-DS module successively. The (partial) domain adaptation network is then optimized based on the selected source samples so that a number of n_b new features $F_{t+1}(\mathbf{X}_b^s)$ can be generated by the updated feature extractor. For a specific source sample x_i^s, if it is selected at time step t while the episode is not completed, its next state will be $F_{t+1}(\mathbf{x}_i^s)$. Otherwise, a terminate state will be triggered. Therefore, each batch of source samples are trained T times where T represent the episode length and at each training step, it can generate n_b experiences.

Action. In our DRL-DS module, the action space is binary and each action $a_i^s \in \{0, 1\}$ denotes whether a source sample is retained or discarded. Specifically, $a_i^s = 1$ means to keep source sample x_i^s while $a_i^s = 0$ means to eliminate it. The output of the DRL Network is a two-dimensional Q-value vector mapped from the input state through fully connected layers. The optimal action at state $F_t(\mathbf{x}_i^s)$ is then determined as:

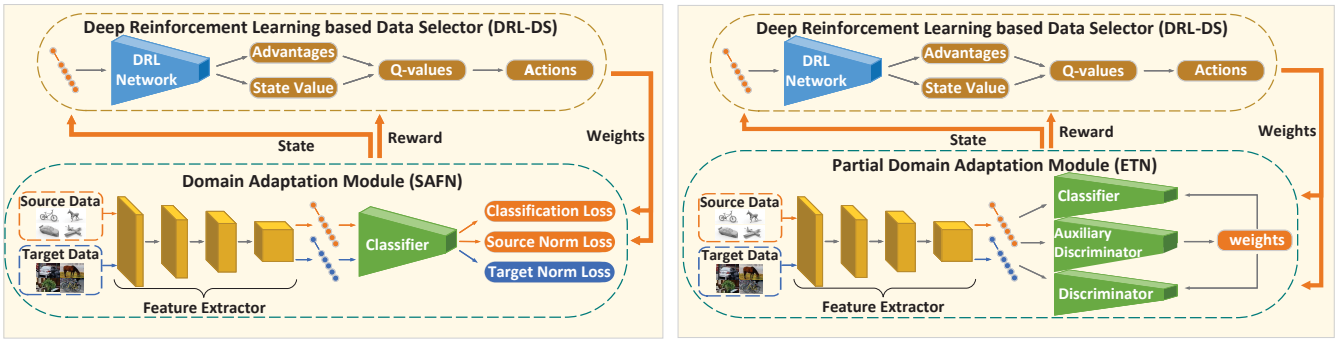
$$a_i^{s*} = \operatorname*{argmax}_a Q(F_t(\mathbf{x}_i^s), a). \quad (3)$$

Reward. The reward function is shaped to provide feedback signals and thereby guide the learning of source data selection policies. As the objective is to select source samples that are more relevant to the target domain, we evaluate the transferability of source instances by measuring their similarity to the target instances and design the reward function based on this transferability. At each training step, in addition to source data, a batch of n_b target samples $\{\mathbf{x}_j^t\}_{j=1}^{n_b} = \mathbf{X}_b^t$ are also fed into the (partial) domain adaptation module. Similarly, n_b feature vectors of these target data $F_t(\mathbf{X}_b^t) = [F_t(\mathbf{x}_1^t), \cdots, F_t(\mathbf{x}_{n_b}^t)]$ can be obtained through the feature extractor. The relevance of each source instance to the target domain can then be estimated by measuring the similarity between its feature vector and those of the target batch. To begin with, the transferability of each source sample to the target domain is evaluated through calculating two functions, i.e.,

$$D_i^1 = \min_{F_t(\mathbf{x}_j^t) \in F_t(\mathbf{X}_b^t)} 1 - \frac{F_t(\mathbf{x}_i^s) \cdot F_t(\mathbf{x}_j^t)}{\|F_t(\mathbf{x}_i^s)\|_2 \left\|F_t(\mathbf{x}_j^t)\right\|_2}, \quad (4)$$

and

$$D_i^2 = \min_{F_t(\mathbf{x}_j^t) \in F_t(\mathbf{X}_b^t)} \sum_m \frac{|F_t(\mathbf{x}_i^s)_m - F_t(\mathbf{x}_j^t)_m|}{|F_t(\mathbf{x}_i^s)_m| + |F_t(\mathbf{x}_j^t)_m|}. \quad (5)$$



(a) DRL-DS + SAFN (b) DRL-DS + ETN

Figure 1: DRL-based data selector for partial domain adaptation. The DRL-DS module eliminates irrelevant source samples based on the estimated Q-values while the (partial) domain adaptation module learns domain-invariant feature representations using the target and selected source data. The reward signals are determined based on the relevance of the source samples to the target domain.

The first function is based on the cosine distance. It is a measure of dissimilarity between two vectors of an inner product space and is invariant to scaling. The second function is based on the Canberra distance. It is a sensitive metric which measures fraction differences between point pairs in a vector space. The smaller the distance values, the greater the match between the feature vectors. Hence, the relevance of each source sample is determined considering both angle and distance between its feature vector and those of the target batch. Based on these two measures, the reward of taking an action a_i^s is designed as:

$$R(F_t(\mathbf{x}_i^s), a_i^s) = \lambda_1(A \oplus B - \lambda_2) + \lambda_3(A \oplus C - \lambda_4), \quad (6)$$

where $\oplus$ is the exclusive-or operation, $A = (a_i^s == 1)$, $B = (D_i^1 > \frac{1}{n_b}\sum_{k=1}^{n_b} D_k^1)$, $C = (D_i^2 > \frac{1}{n_b}\sum_{k=1}^{n_b} D_k^2)$ are three boolean functions, and λ_1, λ_2, λ_3, λ_4 denote four constants to determine the range of the reward. For each measure, a positive value will be obtained if a source is selected while it exhibits higher relevance to the target domain. Similarly, a positive value will also be obtained if a source is eliminated while it exhibits lower relevance to the target domain. Otherwise, a negative value will be triggered. In our DRL-DS module, the reward signals are bounded between -1 and 1 to provide explicit guidance to the agent so that it can learn to distinguish good actions from bad ones efficiently.

Training. Our DRL-DS module maintains two deep neworks, i.e., an online network with parameters θ and a separate target network with parameters θ^-. During training, the online network is updated constantly while the target network is softly updated via polyak averaging to stabilize the training. Instead of using the ϵ-greedy strategy for exploration, our model implements NoisyNets [Fortunato *et al.*, 2017] rather than conventional fully connected layers to achieve more efficient exploration in a consistent way. A linear layer can be expressed as $y = \omega x + b$, where x, y, ω and b denote the input, output, weight matrix and bias, respectively. To achieve exploration, the uncertainty can be added by factorized Gaussian noises so that the weight matrix can be re-formatted as:

$$\omega_{i,j} = \mu_{i,j}^{\omega} + \sigma_{i,j}^{\omega} f(\epsilon_i) f(\epsilon_j), \quad (7)$$

and the corresponding bias can be re-written as:

$$b_j = \mu_j^b + \sigma_j^b f(\epsilon_j), \quad (8)$$

where $f(\epsilon) = \text{sgn}(\epsilon)\sqrt{|\epsilon|}$, $\mu_{i,j}^{\omega}$, $\sigma_{i,j}^{\omega}$, μ_j^b, σ_j^b are network parameters, and ϵ_i, ϵ_j are random noises. In this way, the state value $V(s_t)$ and the advantage $A(s_t, a_t)$ are estimated through noisy fully connected layers and thereafter combined to produce the Q-value using:

$$Q(s_t, a_t; \theta_C, \theta_V, \theta_A) = V(s_t; \theta_C, \theta_V) + A(s_t, a_t; \theta_C, \theta_A) - \frac{1}{2}\sum_a A(s_t, a; \theta_C, \theta_A), \quad (9)$$

where θ_C, θ_V, θ_A represent the parameters of the common network, the state value stream and the advantage stream, respectively. At each training step, n_b new transitions are stored in an experience replay buffer, and a random mini-batch of the stored transitions is sampled from the buffer. The Q-value is updated by minimizing the Huber loss between the estimated Q-value $Q(s_t, a_t; \theta)$ and the target Q-value which is defined as:

$$y_t = R(s_t, a_t) + \beta\gamma Q(s_{t+1}, \underset{a_{t+1}}{\text{argmax}} Q(s_{t+1}, a_{t+1}; \theta); \theta^-), \quad (10)$$

where γ is the discount factor and β is a binary value which equals zero if the episode terminates at step t+1 or one otherwise. It is worth mentioning that the proposed DRL-DS model can be directly integrated with any DA method without modifying either the DA or the DRL-DS model. This is because the DRL-DS model regards source features as its inputs and calculates rewards based on source and target features, while feature extractor is general in all the DA models.

3.3 Domain Adaptation Module

In this paper, the proposed DRL-DS module is integrated with both a domain adaptation method (SAFN) and a partial domain adaptation method (ETN) respectively to evaluate its generalization capability.

SAFN. Stepwise Adaptive Feature Norm (SAFN) learns a feature extractor F with parameters θ_f and a classifier C with parameters θ_y. The parameters of C are updated by minimizing the softmax cross entropy loss on the selected source samples. Meanwhile, in addition to the cross entropy loss, the parameters of F are also updated by enlarging the feature norms of both the target and the selected source samples. Hence, SAFN is dedicated to solve the following optimization problem:

$$\begin{aligned} E(\theta_f, \theta_y) = & \frac{1}{n_s} \sum_{i=1}^{n_s} w(\mathbf{x}_i^s) L_y(C(F(\mathbf{x}_i^s)), \mathbf{y}_i^s) \\ & + \frac{\lambda}{n_s} \sum_{i=1}^{n_s} w(\mathbf{x}_i^s) L_d(h(\mathbf{x}_i^s; \theta_0) + \Delta r, h(\mathbf{x}_i^s)) \\ & + \frac{\lambda}{n_t} \sum_{j=1}^{n_t} L_d(h(\mathbf{x}_j^t; \theta_0) + \Delta r, h(\mathbf{x}_j^t)), \end{aligned} \tag{11}$$

where $\mathbf{y}_i^s$ is the one-hot label for $\mathbf{x}_i^s$, $w(\mathbf{x}_i^s) \in \{0, 1\}$ is the binary weight of $\mathbf{x}_i^s$ determined by the output action a_i^s of the DRL-DS module, $h(x) = (\|\cdot\|_2 \circ F)(x)$, θ_0 is the current parameters of F, Δr represents a positive residual scalar to control the norm enlargement, $L_d(\cdot, \cdot)$ denotes the L_2-distance, and λ is a hyperparameter to trade off the objectives.

ETN. Example Transfer Network (ETN) jointly learns a domain-invariant classifier and a progressive weighting scheme which quantifies the transferability of source samples. It learns a feature extractor F with parameters θ_f, a classifier C with parameters θ_y and a discriminator D with parameters θ_d. In addition, it also learns an auxiliary domain discriminator $\widetilde{D}$ to quantify source samples' transferability. Combined with the DRL-DS module, the domain adaptation network is updated through $\min_{\theta_f, \theta_y} E_C - E_D$ and $\min_{\theta_d} E_D$, where

$$\begin{aligned} E_C = & \frac{1}{n_s} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_s} w(\mathbf{x}_i^s) W(\mathbf{x}_i^s) L_y(C(F(\mathbf{x}_i^s)), \mathbf{y}_i^s) \\ & + \frac{\lambda}{n_t} \sum_{(\mathbf{x}_j, \mathbf{y}_j) \in \mathcal{D}_t} H(C(F(\mathbf{x}_j^t))), \\ E_D = & -\frac{1}{n_s} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_s} w(\mathbf{x}_i^s) W(\mathbf{x}_i^s) \log(D(F(\mathbf{x}_i^s))) \\ & - \frac{1}{n_t} \sum_{(\mathbf{x}_j, \mathbf{y}_j) \in \mathcal{D}_t} \log(1 - D(F(\mathbf{x}_j^t))). \end{aligned} \tag{12}$$

In Eq. 12, λ is a trade-off parameter, $w(\mathbf{x}_i^s)$ denotes the binary weight generated by the DRL-DS module and $H(\cdot)$ is the entropy loss. In addition, $W(\mathbf{x}_i^s)$ is the weight generated using the auxiliary discriminator of the ETN model so that:

$$W(\mathbf{x}_i^s) = 1 - \widetilde{D}(F(\mathbf{x}_i^t)). \tag{13}$$

Generally, the value of $\widetilde{D}(F(\mathbf{x}_i^t))$ will be smaller if a source example is more probable to be in the shared label space. Since $\widetilde{D}(F(\mathbf{x}_i^t))$ is closer to one, the weights $\widetilde{D}(F(\mathbf{x}_i^t))$ in each batch size are normalized as $W(\mathbf{x}_i^s) = \frac{W(\mathbf{x}_i^s)}{\frac{1}{n_b} \sum_{i=1}^{n_b} W(\mathbf{x}_i^s)}$.

4 Experiments

4.1 Setup

The proposed method is evaluated on three datasets.

Office-31 [Saenko *et al.*, 2010] is a widely used domain adaptation dataset containing 4,652 images in 31 categories from three domains, i.e., Amazon (**A**), Webcam (**W**) and DSLR (**D**). Following [Cao *et al.*, 2018a], the same ten categories are selected as target domains to create six PDA tasks.

Office-Home [Venkateswara *et al.*, 2017] is a more challenging dataset that contains about 15,500 images in 65 categories from four different domains, i.e., Artistic (**Ar**), Clipart (**Cl**), Product (**Pr**), Real-World (**Rw**). Following the settings in [Cao *et al.*, 2018b], the first 25 categories in alphabetic order are selected as target domains to create twelve PDA tasks.

VisDA2017 [Peng *et al.*, 2017] is a large-scale dataset which includes over 280,000 images across 12 categories and aims to bridge the significant gap between synthetic and real domains. Following the settings in [Cao *et al.*, 2018b], we select the first six categories in alphabetic order as target categories to create the Synthetic-12 $\rightarrow$ Real-6 task.

The DRL-DS module consists of one common fully connected layer, two noisy fully connected layers for the state value stream and two noisy fully connected layers for the advantage stream. During training, the backbone network, ResNet-50, is pre-trained on ImageNet while the other layers are trained from scratch. The DA module is trained via SGD with a batch size of 32 and a learning rate of 1e-3. The DRL-DS module is trained using Adam with a batch size of 32 and a learning rate of 1e-4. The episode length is set to five. The discount factor in Eq. 10 is set to 0.9 and λ_1, λ_2, λ_3 and λ_4, in Eq. 6 are set to 0.2, 0.5, 1.8 and 0.5, respectively. We compare the proposed method with a variety of state-of-the-art methods. For all the baseline methods, we either refer to the reported results in [Cao *et al.*, 2019; Li *et al.*, 2020b; Chen *et al.*, 2020b; Chen *et al.*, 2020a; Xu *et al.*, 2019; Li *et al.*, 2020a; Jing *et al.*, 2020; Kim and Hong, 2021] or calculate the average values of three runs using the original code. The proposed methods are also trained three times to calculate the average values.

4.2 Results and Discussions

The classification results on the six tasks of *Office-31* are shown in Table 1. It is observed that the two proposed methods significantly outperform most of the baseline methods with average accuracies 97.21% and 98.41%, respectively. Compared to SAFN and ETN, DRL-DS + SAFN and DRL-DS + ETN achieve a 4.56% and a 4.18% improvement in average accuracy, respectively. It is worth mentioning that the DRL-DS + ETN method achieves state-of-the-art accuracy. By integrating the proposed DRL-DS module, the performance of the (partial) domain adaptation methods are consistently improved on all tasks by a large margin with similar convergence speed, which convincingly demonstrates the significance of the proposed DRL based source data selector.

In addition, it is noticed that DAN, DANN, ADDA and RTN all lead to performance degradation compared to ResNet. This is because direct aligning domains with different label spaces can result in negative transfer. In contrast, the

Method	Office-31						
	A → W	D → W	W → D	A → D	D → A	W → A	Avg
ResNet [He *et al.*, 2016]	75.59	96.27	98.09	83.44	83.92	84.97	87.05
DAN [Long *et al.*, 2015]	59.32	73.90	90.45	61.78	74.95	67.64	71.34
DANN [Ganin *et al.*, 2016]	73.56	96.27	98.73	81.53	82.78	86.12	86.50
ADDA [Tzeng *et al.*, 2017]	75.67	95.38	99.85	83.41	83.62	84.25	87.03
RTN [Long *et al.*, 2016]	78.98	93.22	85.35	77.07	89.25	89.46	85.56
SAN [Cao *et al.*, 2018a]	93.90	99.32	99.36	94.27	94.15	88.73	94.96
IWAN [Zhang *et al.*, 2018]	89.15	99.32	99.36	90.45	95.62	94.26	94.69
PADA [Cao *et al.*, 2018b]	86.54	99.32	**100.00**	82.17	92.69	95.41	92.69
DRCN [Li *et al.*, 2020b]	90.80	**100.00**	**100.00**	94.30	95.20	94.80	95.90
DAPDA [Li *et al.*, 2020a]	95.06	**100.00**	**100.00**	92.15	95.13	**97.40**	96.62
DCDF [Jing *et al.*, 2020]	95.93	99.66	**100.00**	98.09	95.09	95.51	97.38
AGAN [Kim and Hong, 2021]	97.28	**100.00**	**100.00**	94.26	95.72	95.72	97.16
RTNet [Chen *et al.*, 2020b]	96.20	**100.00**	**100.00**	97.60	92.30	95.40	96.90
DARL [Chen *et al.*, 2020a]	94.58	99.66	**100.00**	**98.73**	94.57	94.26	96.97
SAFN [Xu *et al.*, 2019]	87.12	96.72	99.36	88.11	93.04	93.46	92.97
ETN [Cao *et al.*, 2019]	88.59	99.89	99.36	89.17	94.68	95.06	94.46
DRL-DS + SAFN	96.61±1.18	**100.00±0.0**	**100.00±0.0**	95.97±0.74	95.37±0.06	95.30±0.11	97.21
DRL-DS + ETN	**99.55±0.20**	**100.00±0.0**	**100.00±0.0**	98.52±0.37	**96.24±0.18**	96.17±0.16	**98.41**

Table 1: Classification Accuracy (%) for Partial Domain Adaptation on *Office-31* Dataset (ResNet-50)

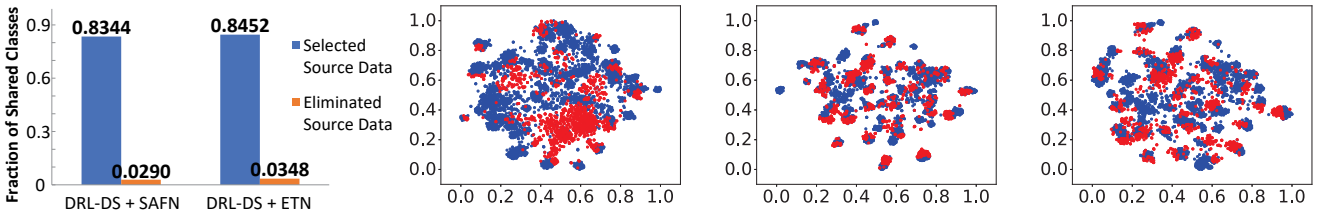
Method	Office-Home												
	Ar → Cl	Ar → Pr	Ar → Rw	Cl → Ar	Cl → Pr	Cl → Rw	Pr → Ar	Pr → Cl	Pr → Rw	Rw → Ar	Rw → Cl	Rw → Pr	Avg
ResNet	46.33	67.51	75.87	59.14	59.94	62.73	58.22	41.79	74.88	67.40	48.18	74.17	61.35
DANN	43.76	67.90	77.47	63.73	58.99	67.59	56.84	37.07	76.37	69.15	44.30	77.48	61.72
ADDA	45.23	68.79	79.21	64.56	60.01	68.29	57.56	38.89	77.45	70.28	45.23	78.32	62.82
RTN	49.31	57.70	80.07	63.54	63.47	73.38	65.11	41.73	75.32	63.18	43.57	80.50	63.07
SAN	44.42	68.68	74.60	67.49	64.99	77.80	59.78	44.72	80.07	72.18	50.21	78.66	65.30
IWAN	53.94	54.45	78.12	61.31	47.95	63.32	54.17	52.02	81.28	76.46	56.75	82.90	63.56
PADA	51.95	67.00	78.74	52.16	53.78	59.03	52.61	43.22	78.79	73.73	56.60	77.09	62.06
DRCN	54.00	76.40	83.00	62.10	64.50	71.00	70.80	49.80	80.50	77.50	59.10	79.90	69.00
DAPDA	56.49	77.56	80.29	65.73	71.52	77.28	66.53	55.96	85.65	77.02	60.82	84.82	71.64
DCDF	60.30	80.17	81.23	67.49	68.24	76.04	68.31	55.05	83.77	75.39	58.93	83.14	71.51
AGAN	56.36	77.25	85.09	74.20	73.84	**81.12**	70.80	51.52	84.54	78.97	56.78	83.42	72.82
RTNet	**63.20**	80.10	80.70	66.70	69.30	77.20	71.60	53.90	84.60	77.40	57.90	85.50	72.30
DARL	55.31	80.73	**86.36**	67.93	66.16	78.52	68.74	50.93	**87.74**	**79.45**	57.19	**85.60**	72.06
SAFN	60.82	79.37	83.49	74.14	74.92	79.37	75.42	58.19	82.92	78.33	63.08	82.29	74.36
ETN	52.95	73.09	83.78	70.00	68.48	77.49	68.66	49.93	81.98	76.61	54.67	81.34	69.92
DRL-DS + SAFN	60.96 ±0.15	**80.75** ±0.96	84.47 ±0.34	**75.51** ±1.01	**75.82** ±0.78	80.05 ±0.51	**75.97** ±0.46	**60.06** ±1.14	83.42 ±0.35	79.00 ±0.69	**64.34** ±0.54	83.21 ±0.32	**75.30**
DRL-DS + ETN	55.10 ±1.79	75.82 ±2.12	86.4 ±0.40	73.31 ±0.27	72.08 ±0.55	80.54 ±1.58	70.656 ±0.96	52.30 ±1.56	83.93 ±0.69	78.97 ±0.88	57.55 ±0.59	83.60 ±0.57	72.52

Table 2: Classification Accuracy (%) for Partial Domain Adaptation on *Office-Home* Dataset (ResNet-50)

PDA methods achieve better performance on most tasks since their weighting schemes mitigate negative influence caused by outlier source data. Therefore, it is crucial to eliminate irrelevant source instances while aligning the two domains.

Different from *Office-31*, *Office-Home* and *VisDa2017* are larger and more challenging. The classification results on these two datasets are shown in Table 2 and 3, respectively. Similarly, DRL-DS + SAFN and DRL-DS + ETN outperform SAFN and ETN consistently on all tasks, respectively. By incorporating the proposed DRL based source data selector, DRL-DS + SAFN and DRL-DS + ETN gain 1.26% and 3.72% improvement on the *Office-Home* tasks, respectively, and gain 3.95% and 17.17% on the *VisDa2017* task, respectively. Therefore, it proves that the proposed DRL-DS module is superior to filter out irrelevant source data automatically and thereby boost knowledge transfer in PDA tasks. Compared to existing RL-based PDA methods, our method is capable of achieving better performance in an off-policy manner without relying on any network architecture or requiring any additional network block for reward calculation.

In addition, the fraction of shared classes in the selected and eliminated source samples are depicted in Fig. 2(a). Theoretically, source data in shared classes should be selected while those in unshared classes should be eliminated



(a) Fraction of shared classes (b) Before training on task Rw → Cl (c) DRL-DS + SAFN on task Rw → Cl (d) DRL-DS + ETN on task Rw → Cl

Figure 2: (a) Fraction of shared classes in the selected and eliminated source samples. (b)-(d) The t-SNE visualization on task Rw → Cl with domain information, where the blue dots represent the source data and the red dots represent the target data.

Method	Synthetic-12 → Real-6
ResNet [He *et al.*, 2016]	45.26
DAN [Long *et al.*, 2015]	47.60
DANN [Ganin *et al.*, 2016]	51.01
RTN [Long *et al.*, 2016]	50.04
SAN [Cao *et al.*, 2018a]	49.90
IWAN [Zhang *et al.*, 2018]	48.60
PADA [Cao *et al.*, 2018b]	53.53
DRCN [Li *et al.*, 2020b]	58.20
AGAN [Kim and Hong, 2021]	67.71
DARL [Chen *et al.*, 2020a]	67.77
SAFN [Xu *et al.*, 2019]	65.27
ETN [Cao *et al.*, 2019]	69.20
DRL-DS + SAFN	67.85±0.66
DRL-DS + ETN	**81.08±0.96**

Table 3: Classification Accuracy (%) for Partial Domain Adaptation on *VisDa2017* Dataset (ResNet-50)

during training. It is observed that the source samples selected by the DRL-DS module mostly belong to the categories shared between the source and target domains, while the instances eliminated by the DRL-DS module rarely belong to the shared classes. This demonstrates the capability of the proposed DRL-based data selector to boost knowledge transfer through filtering out irrelevant source instances. Lastly, the t-SNE embeddings with domain information are demonstrated in Fig. 2. The source and target representations are in blue and red respectively. Before training, the source and target correlated features are not well aligned and feature representations collide into a mess. After training, DRL-DS + SAFN and DRL-DS +ETN are capable of aligning the target samples to corresponding source domain clusters and discriminating different classes in both domains, which manifests the effectiveness of DRL-DS to boost PDA performance.

Despite the superiority of DRL-DS, the key limitation of the proposed method is its sensitivity to hyper-parameters. Specifically, the episode length of RL and the batch size of DA are two important parameters. If the episode length is too small, the RL task cannot be well formulated, and if it is too large, the process can be redundant. For instance, DRL-DS + SAFN achieves an average accuracy of 96.61, 93.33 and 94.58 on the a→w task with the episode length being set to 5, 2 and 10, respectively. Similarly, since the transferability of source samples is only calculated based on a batch of data, batch size can be a sensitive hyper-parameter, especially on larger dataset. If the batch size is too small, the similarity cannot be approximated accurately and stably, while it also cannot be too large due to limited computational resources and increased computational time. For instance, DRL-DS + SAFN achieves an average accuracy of 96.61, 95.25 and 86.67 on the a→w task with the batch size being set to 32, 8 and 4, respectively. Though these hyper-parameters can be obtained empirically, we still aim to deal with this limitation by redesigning the reward or the framework in the future.

5 Conclusion

In this paper, we propose a DRL-based source data selector to boost cross-domain knowledge transfer in PDA tasks. The developed DRL-based data selector is a general paradigm that enables automatic and efficient elimination of irrelevant source instances to circumvent negative transfer and in turn, boost positive transfer across domains. Experimental results on different benchmark datasets have demonstrated the significance of the proposed DRL-based data selector.

Acknowledgements

This research is supported by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funds (Grant No. A20H6b0151) and Career Development Award (Grant No. C210112046).

References

[Cao *et al.*, 2018a] Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Partial transfer learning with selective adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2724–2732, 2018.

[Cao *et al.*, 2018b] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 135–150, 2018.

[Cao *et al.*, 2019] Zhangjie Cao, Kaichao You, Mingsheng Long, Jianmin Wang, and Qiang Yang. Learning to transfer examples for partial domain adaptation. In *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2985–2994, 2019.

[Chen *et al.*, 2020a] J. Chen, X. Wu, L. Duan, and S. Gao. Domain adversarial reinforcement learning for partial domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2020.

[Chen *et al.*, 2020b] Zhihong Chen, Chao Chen, Zhaowei Cheng, Boyuan Jiang, Ke Fang, and Xinyu Jin. Selective transfer with reinforced transfer network for partial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12706–12714, 2020.

[Fortunato *et al.*, 2017] Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, et al. Noisy networks for exploration. *arXiv preprint arXiv:1706.10295*, 2017.

[Ganin and Lempitsky, 2014] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv:1409.7495*, 2014.

[Ganin *et al.*, 2016] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[He *et al.*, 2016] 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.

[Jing *et al.*, 2020] Taotao Jing, Ming Shao, and Zhengming Ding. Discriminative cross-domain feature learning for partial domain adaptation. *arXiv preprint arXiv:2008.11360*, 2020.

[Kim and Hong, 2021] Youngeun Kim and Sungeun Hong. Adaptive graph adversarial networks for partial domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[Konda and Tsitsiklis, 2000] Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pages 1008–1014, 2000.

[Li *et al.*, 2020a] Lusi Li, Zhiqiang Wan, and Haibo He. Dual alignment for partial domain adaptation. *IEEE transactions on cybernetics*, 2020.

[Li *et al.*, 2020b] Shuang Li, Chi Harold Liu, Qiuxia Lin, Qi Wen, Limin Su, Gao Huang, and Zhengming Ding. Deep residual correction network for partial domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[Long *et al.*, 2015] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I Jordan. Learning transferable features with deep adaptation networks. *arXiv preprint arXiv:1502.02791*, 2015.

[Long *et al.*, 2016] 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.

[Long *et al.*, 2017] 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-Volume 70*, pages 2208–2217. JMLR. org, 2017.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[Peng *et al.*, 2017] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.

[Russakovsky *et al.*, 2015] 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.

[Saenko *et al.*, 2010] 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.

[Tzeng *et al.*, 2017] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[Venkateswara *et al.*, 2017] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.

[Wang *et al.*, 2016] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International conference on machine learning*, pages 1995–2003. PMLR, 2016.

[Xu *et al.*, 2019] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1426–1435, 2019.

[Zhang *et al.*, 2018] 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.