Multi-Source Open-Set Deep Adversarial Domain Adaptation

Sayan Rakshit[1][0000−0002−0189−7257], Dipesh Tamboli[1], Pragati Shuddhodhan Meshram[1], Biplab Banerjee[1][0000−0001−8371−8138], Gemma Roig[2][0000−0002−6439−8076], and Subhasis Chaudhuri[1]

[1] Indian Institute of Technology Bombay, Mumbai, India
{sayan1by2, dipeshtamboli1, pragatimeshram30, getbiplab}@gmail.com, sc@iitb.ac.in
[2] Goethe University Frankfurt, Frankfurt, Germany
roig@cs.uni-frankfurt.de

Abstract. We introduce a novel learning paradigm of multi-source open-set unsupervised domain adaptation (MS-OSDA). Recently, the notion of single-source open-set domain adaptation (SS-OSDA) which considers the presence of previously unseen open-set (unknown) classes in the target-domain in addition to the source-domain closed-set (known) classes has drawn attention. In the SS-OSDA setting, the labeled samples are assumed to be drawn from the same source. Yet, it is more plausible to assume that the labeled samples are distributed over multiple source-domains, but the existing SS-OSDA techniques cannot directly handle this more realistic scenario considering the diversities among multiple source-domains. As a remedy, we propose a novel adversarial learning-driven approach to deal with MS-OSDA. Precisely, we model a shared feature space for all the domains which explicitly mitigates the domain-gap among the source-domains. The adversarial learning strategy is introduced to align the known-class samples from the target-domain with the source data while making the unknown-classes more separable. We validate our method on the Office-31, Office-Home, Office-CalTech, and Digits datasets and find that the proposed model consistently outperforms the baseline and benchmark SS-OSDA approaches.

Keywords: Domain Adaptation, Multi-Source, Open-Set.

1 Introduction

Deep learning techniques are attested to be highly successful over a wide variety of visual inference tasks, thanks to their data-driven feature learning capabilities [13, 16]. However, their performance is heavily dependent on the availability of voluminous labeled training samples to achieve a reliable level of generalization. Ideally, a supervised learning algorithm trained on a certain distribution of labeled samples (source-domain) often fails to generalize convincingly when deployed on a new environment (target-domain) in the presence of distributions-shift. In this regard, unsupervised domain adaptation (DA) [22] algorithms seek

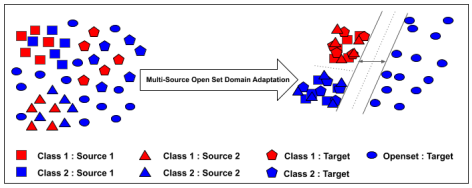


Fig. 1: Given an MS-OSDA setup, our goal is to obtain a discriminative feature space for the known-class samples from all the source and target domains while pushing the unknown-class samples from the target domain far from the known-class support.

to combat the domain-shift problem by aligning the data distributions of the source and target domains by learning a domain-invariant feature space using statistical or adversarial learning approaches, preferably in the absence of label information in the target-domain [29, 3]. In this paper, we tackle the completely novel paradigm of multi-source open-set domain adaptation (MS-OSDA), illustrated in Figure 1.

In general, the notion of multi-source DA (MSDA) [28] is regarded more practical as well as challenging than the single-source DA (SSDA) setup considering that labeled samples may come from diverse sources. In MSDA, we note that the source-domains have different data distributions among themselves in addition to the usual domain-gap between the source and the target domains. One of the straight-forward solutions to MSDA is based on the idea of combining all the source-domains into a single auxiliary source-domain and subsequently deploying any SSDA method. Clearly, such a naive approach may lead to sub-optimal classification results if proper care is not taken in mitigating the gaps among all the domains exclusively.

The paradigm of closed-set DA has mostly been practiced in the literature for both SSDA and MSDA where the same set of classes is shared across the domains [22]. In contrast, the recently introduced single-source open-set DA (SS-OSDA) [21] setting allows the presence of domain-specific classes in addition to the classes shared by the domains. There exists two possibilities in this regard. While the SS-OSDA setup by [21] considers that both source and target specific open-set classes may be available, the setup followed in [25, 15] permits the presence of target specific open-set samples only. Such an SS-OSDA arrangement of [25, 15] is extremely challenging given the unavailability of any prior information regarding the open-set distribution. The closed-set DA techniques cannot be directly applied in this case since these target specific open-set samples, in turn, may jeopardize the domain alignment process. In order to tackle such a situation, accurate discrimination between the known and unknown target-domain classes is advocated during adaptation so that only the shared classes can be aligned.

The existing MSDA techniques contemplate the presence of the same set of classes in all the source and the target domains [34, 35]. This is a strict scenario as far as the unsupervised DA setup is considered where the target-domain is

assumed to be completely unlabeled. Hence, it is highly likely that the target-domain may contain samples from novel classes different from those already in the source-domains. Inspired by these arguments, we propose a novel learning scenario in this paper for multi-source open-set unsupervised DA (MS-OSDA) where there exists multiple labeled source-domains each containing samples from the same set of semantic classes, and the unlabeled target-domain contains two types of data items: either from the source-domain known-classes or from novel unknown-classes. Under this setup, the task is to classify the target-domain samples either in one of the known categories or they are assigned a common unknown-class label. Such an MS-OSDA setup invariably holds huge applications in fields relating to on-the-fly real-world visual perception like medical imaging, remote sensing, where acquisition of multi-domain data is perennial and novel categories may turn up abruptly. Nonetheless, we note that the MS-OSDA problem cannot be effectively solved by directly utilizing the SS-OSDA paradigm of [25, 15] mainly because of the following factors: i) the presence of multiple diverse source-domains hinders the effectiveness of a traditional SS-OSDA technique, and ii) design of the known/unknown class discriminator for target samples may become non-trivial since the target-domain may have varied degrees of relatedness with different source-domains.

To solve these problems, we propose a new framework which aims at learning a shared feature space for all the source and target domains where i) the source-domains are purposefully aligned among themselves, and ii) a target-domain pseudo-classifier is designed to accomplish two tasks: a) to align the target-domain known-class samples with those of the source-domains, and b) to maximize the gap between the known and unknown target-domain samples (Figure 1). To this end, our proposed model: Multi-source Open-Set DA NETwork (MOSDANET) consists of a shared feature encoder for the source and target domains and separate multi-class classifiers for the source-domains, respectively. The classifiers are augmented with an extra unknown-class label for all the open-set samples in addition to the known-class labels. The pseudo-classifier for the target-domain is subsequently designed using an ensemble of these source classifiers.

Recently, [23] argued that reducing the domain-gap among the source-domains explicitly leads to a more robust and effective MSDA model. We find this idea to be particularly relevant to MS-OSDA since aligning the source-domains among themselves inherently helps in better discrimination of the target-domain samples into known and unknown categories. Otherwise, the domain-shift among the source-domains may mislead the pseudo-classifier to wrongly identify an unknown-class sample to be originated from a known-class or vice-versa. In the same line, we propose to perform fine-grained alignment among the source-domains in the shared space to induce the notion of discriminativeness among the known-class data. On the other hand, a novel adversarial loss function is proposed to train the large-margin pseudo-classifier for the target-domain samples. We consider to use adversarial strategy in this case given their recent success in implicit distributions matching for cross-domain inference tasks [32]. Imposing

the large-margin constraint helps in dealing with different openness factors (fraction of classes present in the open-set) in an efficient manner since the margin idea offers more separability for the unknown-class data. Our major contributions can be summarized as follows:

- We introduce the problem setting for multi-source open-set DA and propose an adversarial learning based framework termed as MOSDANET.
- We highlight some of the important aspects of MOSDANET as: a) aligning the source-domains explicitly at class-level, b) design of an intuitive large-margin discriminator for the target-domain known/unknown classes through a newly developed adversarial training strategy, and c) consideration of target-domain samples with pseudo-labels corresponding to the known-classes to explicitly aid in the fine-grained domain alignment process.
- We establish the efficacy of MOSDANET through extensive experiments on four benchmark datasets where we perform thorough robustness analysis.

2 Related Works

Closed-set and open-set single-source DA: The existing literature is rich in methods relating both to closed-set and open-set DA involving a single source-domain. For SSDA, several ad-hoc techniques existed prior to the deep learning era where the goal was to either project both the domains onto a shared latent space or to align the data distribution of a given domain to match the properties of the other [20, 5, 2]. These techniques were subsequently replaced by more accurate deep CNN based approaches which reduce the domain-gap in the learned CNN representations through an end-to-end training [17, 27, 1]. Nowadays, there exist a plethora of models influenced by the adversarial training strategy which have showcased superlative performance [29, 31]. Typically, these approaches pose the DA problem as learning a domain-confused feature space through an adversarial training between two players: a feature generator and a domain discriminator, respectively. For example, domain adversarial neural network (DANN) [4] introduces the *gradient-reversal* layer to accomplish the task. A few methods in this respect resort to the notion of ensemble learning and exploit the outcomes of the committee of source-domain classifiers to define the adversary [25, 18].

The SS-OSDA problem, on the other hand, was first coined in assign and transform iterative (ATI-λ) [21] which considers the distance between the target samples and the source clusters to decide on the potential known/unknown class labels for the target data. Note that ATI-λ utilizes some of the open-set classes from the source-domain during training. The open-set DA by back-propagation (OSDA-BP) [25] trains the feature generator within a typical generator and discriminator based adversarial learning framework to lead the discriminator to predict the class-label of a target sample to be unknown if the likelihood exceeds a predefined threshold. The improved OSDA-BP [3] replaces the cross-entropy based adversarial loss of [25] by a symmetric version of Kullback-Leibler

(KL) divergence and showcases improved SS-OSDA performance. While these approaches are based on aligning the source and the target domains at one go, an alternate approach proposed in [15] progressively builds the alignment given the domains.

Closed-set multi-source DA: MSDA techniques assume that the labeled training samples are distributed over multiple source-domains. One of the first MSDA approaches (A-SVM) [33] in this respect distills the capacity of all the source classifiers to better model a target classifier. Following [33], there have been several endeavors towards MSDA for different application areas like language processing, sentiment analysis etc. [11]. As far as the adversarial approaches are concerned, [23] proposes a moment-matching network for MSDA which is based on aligning higher-order moments between the domain-specific features. On the other hand, [35] proposes a multi-source domain adversarial network (MDAN) which models separate mapping functions for each of the source-target pairs. [6] introduces the idea of deploying a mixture of source experts for MSDA for cross-domain sentiment analysis.

MS-OSDA is a completely new paradigm that we introduce with an adversarial training strategy followed for the target-domain pseudo-classifier, which is loosely inspired by [25]. Since the adversarial training of [25] is based on the standard cross-entropy based classification loss, it is susceptible to severe misclassification if the known and unknown-class samples have some similarities. Furthermore, this restricts the ability of the model to deal with different openness. In MOSDANET, we solve these bottlenecks of [25] by introducing a margin-based loss-term along with the classification loss. In addition, our target-domain pseudo-classifier is essentially a committee of classifiers since we are dealing with multiple source-domains. Finally, as opposed to the existing MSDA methods [6, 35, 23], we are interested in diminishing any domain-shift among the source-domains in the shared space.

3 Proposed Methodology

3.1 Problem definition & notation

Let us consider the availability of L different source-domains $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_L\}$ each equipped with the domain-specific training set $\mathbb{X}_l = \{x_l^i, y_l^i\}_{i=1}^{n_l}$, ($l \in \{1, 2, \cdots, L\}$ and n_l defines the number of training samples of $\mathcal{S}_l$). Further we note that $(x_l, y_l) \in \mathcal{X}_l \otimes \mathcal{Y}_s$ given the domain-specific feature space $\mathcal{X}_l$ while the label space $\mathcal{Y}_s = \{1, 2, 3, \cdots, K\}$ is shared among all the source-domains. On the other hand, there exists a target-domain $\mathcal{T}$ consisting of n_t unlabeled test samples $\mathbb{X}_t = \{x_t^j\}_{j=1}^{n_t}$ arising from $\mathcal{Y}_t$ categories where $x_t \in \mathcal{X}_t$ given the target-domain feature space $\mathcal{X}_t$. According to our setup, $\mathcal{Y}_s \subset \mathcal{Y}_t$ and $\mathcal{Y}_{t/s}$ denotes the open-set classes of $\mathcal{T}$ which are not part of $\mathcal{S}$. In a typical closed-set MSDA setup, it is assumed that the marginal data distributions of all the source and the target domains are mutually different: $P_l(\mathcal{X}_l) \neq P_m(\mathcal{X}_m)$ and $P_l(\mathcal{X}_l) \neq P_t(\mathcal{X}_t)$, $\mathcal{S}_l, \mathcal{S}_m \in \mathcal{S}$. For MS-OSDA, the distribution of the known classes from $\mathcal{T}$ differs

from that of a given $\mathcal{S}_l$: $P_l(\mathcal{X}_l) \neq P_t(\mathcal{X}_t^{1:K})$ where $\mathcal{X}_t^{1:K}$ represents the target-domain samples with known class-labels. Also let $\mathcal{X}_t^{K+1}$ be the samples from unknown classes in $\mathcal{T}$.

Under this setup, the task is to classify the data from $\mathbb{X}_t$ into $K+1$ categories where the first K indices correspond to classes in $\mathcal{Y}_s$ and the $(K+1)^{th}$ index denotes a common label for all the classes in $\mathcal{Y}_{t/s}$. In order to accomplish the task, we propose a deep neural network with a shared feature encoder $\mathcal{E}(;,\theta_\mathcal{E})$ having parameters $\theta_\mathcal{E}$, L source-domains specific $(K+1)$-class classifiers $\{\mathcal{F}_l(;,\theta_\mathcal{F}^l)\}_{l=1}^L$ each with its own set of parameters $\theta_\mathcal{F}^l$ (an illustration is provided in Figure 2). We use $\theta_\mathcal{F}$ to define all the classifier's parameters $\{\theta_\mathcal{F}^l\}_{l=1}^L$ together. Finally, the classifier model for $\mathcal{T}$: $\mathcal{F}_t(;,\theta_\mathcal{F})$ is essentially an ensemble-classifiers system which is defined by average-pooling the responses of the $\{\mathcal{F}_l\}_{l=1}^L$ for each sample from $\mathcal{X}_t$. In particular, we average-pool the unnormalized logit-scores and apply the softmax function on the pooled responses for obtaining the posterior class-distributions. For a given x_t, let us denote $\mathbf{q}_t = [q_t^1, q_t^2, \cdots, q_t^{K+1}]$ to be the final logit vector obtained in this way. The posterior probability for the k^{th} class ($k \in \{1, 2, \cdots, K+1\}$) is mentioned as: $p(y_t = k|x_t) = \frac{\exp(q_t^k)}{\sum_{c=1}^{K+1}\exp(q_t^c)}$.

3.2 Training & Inference

Overview of the training process: Following the aforementioned setup, we focus on three objectives for training MOSDANET in order to obtain the domain-independent and discriminative shared feature encoder $\mathcal{E}$: i) align the L source-domains in $\mathcal{S}$ at a fine-scale, ii) align known-class target-domain samples in $\mathcal{X}_t^{1:K}$ with $\mathcal{S}$, and iii) widen the margin between $\mathcal{X}_t^{1:K}$ and the unknown-class samples in $\mathcal{X}_t^{K+1}$. Apparently, Objective-(i) is easy to achieve given the availability of labeled training samples for the source-domains. On the other hand, Objectives (ii) and (iii) are non-trivial to attain since $\mathcal{T}$ is unlabeled. We follow an adversarial game between $\mathcal{E}$ and $\mathcal{F}_t$ for approximating the labels for the target-domain samples and thus realizing Objectives (ii) and (iii) simultaneously. To ensure a fine-grained alignment between $\mathcal{S}$ and $\mathcal{T}$, we furthermore propose to re-use some of the potential samples from $\mathcal{X}_t$ with pseudo-labels corresponding to one of the K known-classes in Objective-(i) pretending that they belong to $\mathcal{S}$. The loss functions are detailed in the following.

i) Alignment of the source-domains: We propose to maximize the pairwise similarities among the samples from different source-domains but sharing identical class-labels in the shared space, thus reducing the domain-shift among the source-domains. It leads us to obtain a unified feature space for the known-class samples from different source-domains, which subsequently helps in the better alignment of $\mathcal{X}_t^{1:K}$ with samples from $\mathcal{S}$. Precisely, for each of the known-class labels $k \in \mathcal{Y}_s$, we separately select samples from all of the L source-domains. Let $\mathcal{X}_l^k \subset \mathcal{X}_l$ define the set of samples with class label k obtained from the l^{th} source-domain. Given that, we define the source alignment loss $\mathcal{L}_{SA}$ as:

$$\mathcal{L}_{SA} = \frac{1}{K}\sum_{k\in\mathcal{Y}_s}\frac{1}{L}\sum_{l,m=1,l\neq m}^{L}\mathbb{E}||\mathcal{E}(\mathcal{X}_l^k) - \mathcal{E}(\mathcal{X}_m^k)||_2^2 + \frac{1}{L}\sum_{l=1}^{L}\mathcal{L}_{CE}(\mathcal{F}_l(\mathcal{E}(\mathcal{X}_l)), \mathcal{Y}_s) \quad (1)$$

The first term of Equation 1 accomplishes two goals: a) maximizing the pairwise cosine similarities among the encoded features of the samples originating from different source-domains but sharing identical class-labels: min $|\mathcal{E}(\vec{x}_i) - \mathcal{E}(\vec{x}_j)|_2 \approx \max \mathcal{E}(\vec{x}_i)\mathcal{E}(\vec{x}_j)^T$ for a pair of unit-norm vectors $\vec{x}_i$ $\vec{x}_j$ (note that the $\mathcal{E}$ is designed to ensure the norm constraint), and b) bringing the centroids of the feature samples for each of the class-labels from different source-domains closer. Together, the L source classifiers $\{\mathcal{F}_l\}_{l=1}^{L}$ are trained using separate instances of cross-entropy loss $\mathcal{L}_{CE}$. Hence, the shared space becomes class-wise discriminative taking all the source samples into account.

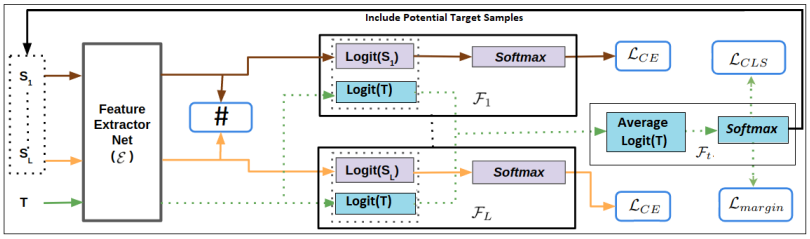


Fig. 2: A depiction of MOSDANET. It majorly consists of a shared feature encoder $\mathcal{E}$ and separate $K+1$-class classifiers $\mathcal{F}_l$s for each of the source domains. $\mathcal{F}_t$ denotes the classifier for $\mathcal{T}$. The figure also depicts the loss terms to be evaluated. # in the figure refers to the first term of Equation 1. Logit(S) means the unnormalized logit vectors.

ii) Alignment between $\mathcal{S}$ and $\mathcal{T}$: For samples in $\mathcal{T}$, we intend to i) correctly classify the known-class samples in one of the K categories by ensuring proper alignment with the source data in the shared space, and ii) classify any unknown-class data with label $K+1$. In addition, we also constrain that the known/unknown separation should be carried out with a high confidence. By confidence, we aim at imposing a large-margin between the known and unknown-class supports as produced by $\mathcal{F}_t$.

Ideally, we need to construct a decision boundary for the open-set, but we are devoid of any prior information in this respect. Alternately, it is intuitive to initially construct a pseudo decision boundary for the unknown-classes in $\mathcal{T}$ using $\mathcal{F}_t$ and to subsequently train $\mathcal{E}$ to deceive the classifier. This refers to the adversarial game between $\mathcal{E}$ and $\mathcal{F}_t$. We deduce a binary cross-entropy based classification loss ($\mathcal{L}_{CLS}$) for defining the pseudo- unknown-class boundary for $\mathcal{T}$ which considers the probability of a given $x_t \in \mathcal{X}_t$ to belong to the $(K+1)^{th}$-class

or the cumulative probability to belong to the K known categories as yielded by $\mathcal{F}_t$ (Equation 2):

$$\mathcal{L}_{CLS} = \mathbb{E}[-0.5\log(p(y_t = K+1|\mathcal{E}(x_t)) - 0.5\log(1 - p(y_t = K+1|\mathcal{E}(x_t))] \quad (2)$$

Following [25], the ground-truth probability for $\mathcal{L}_{CLS}$ is set to 0.5 in order to avoid any trivial solution where all the samples from $\mathcal{T}$ may be wrongly labeled with only known or unknown class labels. We find that the adversarial training using $\mathcal{L}_{CLS}$ produces good alignment between $\mathcal{S}$ and $\mathcal{T}$ if the known and unknown classes are quite distinct but fails if they are find-grained in nature. This is due to the fact that $\mathcal{F}_t$ becomes uncertain in estimating the class-labels if the classes are overlapping in the feature space. In order to tackle the situation, we propose to maximize the margin between the supports of the known and the unknown class boundaries. Ideally for a given x_t, if $|p(y_t = K+1|\mathcal{E}(x_t) - (1 - p(y_t = K+1|\mathcal{E}(x_t))| \geq \tau$ for some predefined threshold τ ($\tau \in [0,1]$, $\tau \approx 1$), then we can claim that the classification is confident. We introduce a margin loss $\mathcal{L}_{margin}$ given the softmax predictions of $\mathcal{F}_t$ as a solution that penalizes samples for which the known and unknown class predictions are closer than τ as follows,

$$\mathcal{L}_{margin} = \mathbb{E}[\min\,(0, |\sum_{k=1}^{K} p(y_t = k|\mathcal{E}(x_t)) - p(y_t = K+1|\mathcal{E}(x_t))|_1 - \tau)] \quad (3)$$

Ideally, the encoder $\mathcal{E}$ seeks to maximize $\mathcal{L}_{margin}$ in order to ensure a large-margin classification by $\mathcal{F}_t$. However, we note that the maximum value of $\mathcal{L}_{margin}$ at optimality is bounded at 0. As a result, the unknown-class data are pushed further away from the known-classes, making MOSDANET robust to different openness.

iii) Inclusion of potential target samples in $\mathcal{L}_{SA}$ during training: In order to further encourage fine-grained alignment between $\mathcal{S}$ and $\mathcal{T}$, we propose to incorporate potential samples from $\mathcal{T}$ with pseudo-labels corresponding to one of the K known classes in the evaluation of $\mathcal{L}_{SA}$ professing that these samples are part of one of the source-domains. We initiate the process of identifying such samples at least after one training epoch is over to ensure reliability.

However, we cannot blindly rely on such pseudo-labels considering the implicit domain differences. A possible solution could be to threshold the predicted class probabilities to decide on the reliability of the pseudo-labels obtained. If the predicted probability for a certain class is extremely high(≈ 1), then the chance of the sample to actually belong to that particular class automatically increases. Given a threshold hyper-parameter α, we use the following rule to decide whether a given x_t qualifies for consideration in this regard:

- If only

$$p(y_t = k|\mathcal{E}(x_t)) \geq \alpha, \alpha \in [0,1], k \in \mathcal{Y}_s \quad (4)$$

(x_t, k) can be included in the augmented training set.

Network optimization: We follow an alternate optimization strategy for training MOSDANET end-to-end. The following three stages are iterated until convergence:

Stage-1: For a fixed encoder, $\mathcal{E}$, the source-domain classifiers are trained to minimize the following cost:

$$\min_{\theta_{\mathcal{F}}} \frac{1}{L}\sum_{l=1}^{L}\mathcal{L}_{CE}(\mathcal{F}_l(\mathcal{E}(\mathcal{X}_l)),\mathcal{Y}_s) + \mathcal{L}_{CLS} \tag{5}$$

Equation 5 signifies that the parameters of the classifiers are simultaneously optimized for correctly classifying the respective source-domain samples in one of the K classes (second term of Equation 1) while preserving the pseudo unknown-class boundary given samples from $\mathcal{T}$ (Equation 2). Please note that the first term of Equation 1 deals with optimizing the encoder parameters, which in this case are fixed.

Stage-2: On the second step, $\mathcal{E}$ is assigned the job of minimizing $\mathcal{L}_{SA}$ (Equation 1), thus classwise aligning the source domains together with optimizing the source classifiers and, ii) maximizing $\mathcal{L}_{CLS} + \mathcal{L}_{margin}$ (Equation 2 and 3, respectively) in order to classify the target samples in known or unknown classes with a high confidence. In particular, $\mathcal{E}$ is updated by optimizing the following cost given a fixed $\theta_{\mathcal{F}}$:

$$\min_{\theta_{\mathcal{E}}} \mathcal{L}_{SA} - (\mathcal{L}_{CLS} + \mathcal{L}_{margin}) \tag{6}$$

Stage-3: We investigate the occurrence of potential target-domain samples with pseudo known-class labels (Equation 4). Once we obtain such samples, they are used in evaluating Equation 2 along with samples from $\mathcal{S}$ from subsequent iteration.

Method	AD - W		AW - D		WD - A		AVG	
	OS*	OS	OS*	OS	OS*	OS	OS*	OS
OSVM[26](†)	73.3	70.2	95.1	94.4	40.2	39.1	69.5±0.3	67.9±0.4
OSVM[26](‡)	71.2	51.2	84.9	56.2	58.2	61.4	71.4±0.5	56.3±0.7
OSVM+DANN[4](‡)	65.0	83.3	68.0	91.9	51.2	37.5	61.4±0.3	70.9±0.4
OSVM+[23]	88.3	58.8	95.5	59.5	**82.3**	52.7	88.7±0.6	57.0±0.4
OSDA-BP [25] (†)	98.1	93.0	99.0	94.1	77.1	75.0	91.4±0.5	87.4±0.4
OSDA-BP[25] (‡)	94.0	90.0	93.0	89.0	79.0	75.0	88.7±0.7	84.7±0.4
IOSDA-BP[3] (†)	98.7	67.0	98.1	62.1	74.7	74.1	90.5±0.5	67.7±0.3
IOSDA-BP[3] (‡)	91.1	88.0	87.8	87.1	75.0	74.5	84.6±0.6	83.2±0.5
MOSDANET(∗)	95.2	91.4	94.4	90.3	77.5	73.1	89.0±0.3	84.9±0.2
MOSDANET	**99.0**	**98.2**	**99.4**	**98.3**	81.0	**79.3**	**93.1±0.4**	**91.9±0.2**

Table 1: The performance comparison for MOSDANET with 20 shared and 11 unknown-classes for Office-31 dataset (in %) . † = single best, ‡ = source combine and * = best member classifier.

Method	Office-Caltech					Office-Home				
	ADW-C OS	ADC-W OS	AWC-D OS	DCW-A OS	AVG OS	ACP - R OS	APR - C OS	PCR - A OS	ACR - P OS	AVG OS
OSVM[26] (†)	43.1	36.5	42.6	44.6	41.7	67.1	59.7	59.3	75.1	65.3
OSVM[26] (‡)	45.3	35.3	34.4	45.5	40.1	60.2	46.3	48.6	57.0	53.0
OSVM+DANN [4](‡)	46.2	42.5	42.3	47.1	44.5	54.5	31.6	40.9	53.8	45.2
OSVM+[23]	18.6	39.5	40.3	21.9	30.1	60.2	51.5	**69.9**	59.8	60.3
OSDA-BP[25](†)	86.4	91.2	92.4	88.4	89.6	73.0	57.0	58.1	70.4	64.6
OSDA-BP[25](‡)	80.4	87.4	91.7	90.4	87.4	53.6	38.0	46.9	54.9	48.3
IOSDA-BP[3](†)	78.6	91.5	93.0	87.0	87.5	58.6	31.4	46.2	64.0	50.0
IOSDA-BP[3](‡)	58.6	57.9	61.6	62.7	60.2	64.5	46.2	54.9	66.4	58.1
MOSDANET(∗)	86.1	96.8	96.7	94.1	92.9	78.0	66.0	62.0	76.3	70.5
MOSDANET	**90.6**	**99.2**	**98.9**	**94.8**	**95.8**	**80.3**	**67.5**	60.6	**80.0**	**72.1**

Table 2: The performance comparison for MOSDANET for 5 shared and 5 unknown classes for Office-caltech dataset and 45 shared 20 unknown classes for the Office-Home dataset (in %). † = single best, ‡ = source combine and * = best member classifier.

Inference: During inference, the target samples in $\mathbb{X}_t$ are propagated through the encoder $\mathcal{E}$ followed by the classifiers-ensemble $\mathcal{F}_t$ and the class with maximum softmax score is selected.

4 Experimental Evaluations

Datasets: We evaluate the MOSDANET on four benchmark datasets: Office-31 [24], Office-Home [30], and Office-CalTech [24], and Digits, respectively. The three domains of Office-31 are: Amazon (**A**), Web (**W**), and DSLR (**D**) each consisting of images from 31 categories. A total of 4652 images are present in this dataset. We consider all possible combinations with two source and one target domains. Besides, 20 shared classes and 11 open-set classes are considered based on the alphabetic order. Office-CalTech is an extension of the Office-31 and consists of the 10 shared classes of Amazon (**A**), CalTech (**C**), Webcam (**W**), and DSLR (**D**), respectively. We consider all the setups having three source-domains and one target-domain with 5 known and 5 open-set classes. The Office-Home dataset contains four domains: Art (**A**), Clipart (**C**), Real-world (**R**), and Product (**P**) where each of the domains is equipped with 65 object categories. In total, there are $15,500$ images present in this dataset. We consider all four possible combinations with three source-domains and one target-domain with 45 shared and 20 open-set classes. Finally, the Digits dataset consists of three domains of hand-written digits: MNIST (**M**)[14], USPS (**U**)[8], and SVHN (**S**)[19] and we consider the combined available training and test samples per domain for this dataset with a total of $1,78,589$ images together for all the domains. 5 known (digits $0-4$) and 5 unknown-classes (digits $5-9$) are considered for comparative analysis. For completeness, we report results on the subset of the DomainNet challenge [23] in the supplementary along with some more analysis.

Model architecture and experimental protocols: For Office-31, Office-CalTech, and Office-Home, our feature encoder $\mathcal{E}$ is constructed from the Imagenet pre-trained Resnet-50 model [7]. However, we replace all the layers after the final 2048-dimensional fully-connected (fc) layer by three new fc-layers with 1000, 512, and 128 nodes, respectively. Batch-norm [9] and Leaky-ReLU non-linearity are considered after each of the new layers for stable training. The parameters of the original network layers remain fixed during training and only the newly considered layers are trained. For Digits, we consider the LeNet [14] architecture as the feature encoder $\mathcal{E}$ and the network is trained from the scratch in contrast to the previous case. The classifiers are further constructed in terms of a small neural network that project the features onto the $K+1$-dimensional class scores.

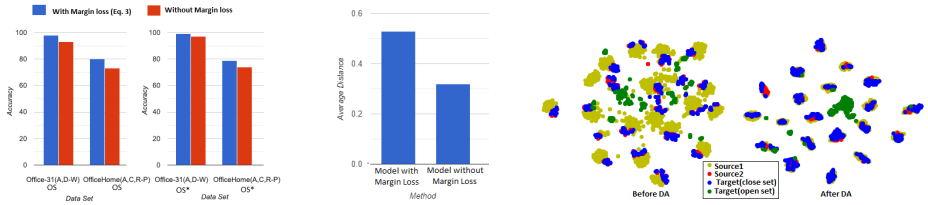
Method	M+U - S			S+M - U			S+U - M			AVG OS
	OS	OS*	UNK	OS	OS*	UNK	OS	OS*	UNK	
OSVM[26](†)	65.8	62.6	71.3	74.8	76.9	87.0	50.3	61.1	11.5	63.6±0.4
OSVM[26](‡)	66.9	65.4	69.4	72.8	74.9	85.6	60.2	68.0	17.1	66.6±0.3
OSVM+[23]	58.4	87.9	10.9	59.1	96.9	6.5	10.7	20.7	0.3	42.7±0.4
IOSDA-BP[3](†)	76.5	74.6	89.9	85.6	88.6	90.5	82.9	83.4	80.4	81.6±0.8
IOSDA-BP[3](‡)	78.6	75.8	92.7	82.9	79.7	91.1	82.1	82.3	90.5	81.2±0.5
MOSDANET	**79.1**	**88.2**	**93.8**	**86.6**	**98.8**	**98.9**	**95.2**	**98.2**	**99.1**	**87.1±0.3**

Table 3: The performance analysis of MOSDANET for the Digits dataset with 5 known and 5 open-set classes (in %). † = single best, ‡ = source combine.

The network is trained using the Adam optimizer [12] with an initial learning rate of 0.001 and a batch size of 64 (for Office datasets) and of size 100 for Digits. Regarding fixing the hyper-parameters, we are convinced that the a higher value for the margin parameter τ (Equation 3) indeed helps in attaining better separation between the known and the unknown classes in $\mathcal{T}$ and we set $\tau = 0.6$ for all the experiments in Table 1-3 as the performance mostly saturates for $\tau \geq 0.6$. The α parameter (Equation 4) which is entitled to decide on the pseudo-labels for the target-domain samples is set to 0.9 since a high α helps in producing more confident pseudo-labels.

4.1 Comparison to the literature and baselines

We compare our method with three different experimental settings in Table 1-3 considering the absence of any prior MS-OSDA literature: i) **source-combine**: here the source-domains are combined to construct an auxiliary source-domain and the single-source and single-target DA setup is followed. In the source-domain, only the known-class samples are considered whereas both known and unknown-class samples are used in the target-domain during training. In this regard, we use three situations: a) Baseline case where the open-set multi-class support vector machines (OSVM) [10] is trained on the auxiliary source-domain



(a) (b) (c)

Fig. 3: (a) Effect of $\mathcal{L}_{margin}$ in terms of OS value for two combinations of Office-31 (A,D-W) and Office-Home(ACR-P), (b) Effect of $\mathcal{L}_{margin}$ in maximizing the difference between the known and unknown-class probabilities for the correctly classified target samples for A,D-W (Office-31), (c) t-SNE plot for the case A,D-W(Office-31) before and after adaptation.

and is then directly evaluated on the target-domain samples. b) We perform domain alignment between the auxiliary source and the target domains using the benchmark closed-set DA method of DANN [4]. Once the training is over, we consider the generated features and use the OSVM for open-set classification. c) We consider two existing SS-OSDA techniques: OSDA-BP [25] and improved OSDA-BP (IOSDA-BP) [3] to be trained and evaluated on the auxiliary source and the target domains, respectively. ii) **single-best**: For baseline OSVM, OSDA-BP, and IOSDA-BP, we also report the best results obtained for the single-source and single-target setup, e.g., for **A,D** $\mapsto$ **W**, the best result between **A** $\mapsto$ **W** and **D** $\mapsto$ **W** is reported, and iii) **multi-source**: In this regard, we consider the very recent benchmark MSDA method of [23] and train the model using known-classes in the source-domains and known + unknown classes in the target and then use OSVM for classification. For the single-best and source-combine cases, we follow similar architecture for the feature encoder as of MOSDANET (Section 4). For baseline OSVM, we train the feature encoder on the (auxiliary) source domain and subsequently utilize the same as the feature generator for the target samples. We report the OS and OS^* [21] scores to signify the average classwise accuracy for known + unknown classes and only for known-classes, respectively. For Digits, we also report the performance on the unknown-classes (UNK).

We note that all the considered DA techniques except OSDA-BP and IOSDA-BP are basically designed for closed-set DA. Hence, their performances on the known-classes are good while they fail to detect the unknown-classes properly (Table 1-3). Similar trends can be observed when the benchmark multi-source DA method [23] is used for the domain alignment. On the other hand, the source combine versions of OSDA-BP and IOSDA-BP produce better average OS values than DANN + OSVM. Finally, our method produces the best average OS for all the datasets. In particular, Digits is extremely large-scale in terms of the number of samples and Office-Home has complex class-distributions with several fine-grained categories. Still, the performance of MOSDANET on these datasets are comparatively high. We also report the OS values for the best-performing

member-classifier within the ensemble. It can be observed that the decision fusion used in $\mathcal{F}_l$ sharply enhances the performance over each member.

4.2 Critical analysis

The effect of the margin-loss ($\mathcal{L}_{margin}$) and the margin parameter τ: We showcase the effectiveness of $\mathcal{L}_{margin}$ on two cases: **A,D** $\mapsto$ **W** (Office-31) and **A,C,R** $\mapsto$ **P** (Office-Home) (Figure 3(a)) in terms of the OS and OS^* values for our full model and the model without $\mathcal{L}_{margin}$. It can be observed that the inclusion of $\mathcal{L}_{margin}$ in Equation 6 causes an enhancement in the OS values at least by 4%. Furthermore, $\mathcal{L}_{margin}$ depends on the choice of the τ hyper-parameter for controlling the confidence of the classifier. In Table 4(a), we showcase an ablation analysis on the τ parameter. As deserved, a large τ is preferred as it maximizes the margin between the known and unknown class samples. Increasing trends for both the OS and OS^* can be seen as τ is increased, however, τ is found to get saturated after 0.6. In Figure 3(b), we show the difference between the known and unknown class probabilities for the full model and the model trained without $\mathcal{L}_{margin}$ in Equation 6 for $\tau = 0.6$. In this regard, the full model has average difference score of ≈ 0.55 which is superior to the model without $\mathcal{L}_{margin}$ (≈ 0.3).

Sensitivity to different openness: Openness is defined as $\mathcal{O} = 1 - \frac{|\mathcal{Y}_s|}{|\mathcal{Y}_t|}$ [15] where $|\mathcal{Y}_s|$ denotes the number of classes in $\mathcal{S}$ whereas $|\mathcal{Y}_t|$ is the number of known and unknown classes present in $\mathcal{T}$. A large $\mathcal{O}$ signifies that the number of unknown classes is much higher than the number of shared classes. The source-combine or single-best versions of [25] and [3] show poor performance when $\mathcal{O} \approx 1$ whereas they show high accuracy for $\mathcal{O} \approx 0.5$. This is due to the fact that methods like [25, 3] are prone to confound the known with unknown classes. From Table 4(b), we observe that MOSDANET produces promising results even when $\mathcal{O} \to 1$. This guarantees the invariance of MOSDANET to varied openness which is majorly attributed to the efficacy of $\mathcal{L}_{margin}$ in separating the unknown classes. To establish this, we mention a comparison of the OS scores for different $\mathcal{O}$ values between the models with and without the margin-loss in Figure 4(c)

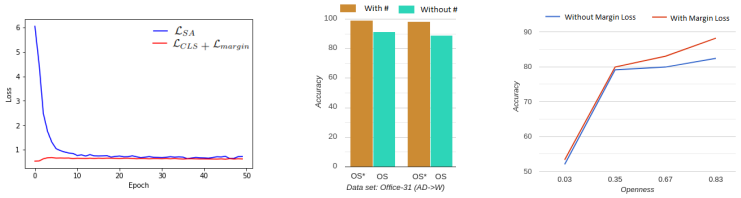
	AD - W		ACR - P	
	OS*	OS	OS*	OS
$\tau = 0.2$	98.1	95.8	66.5	66.4
$\tau = 0.4$	98.2	96.8	76.1	75.3
$\tau = 0.6$	99.0	98.2	79.0	80.0

(a)

	AD - W		AW - D		WD - A	
	OS*	OS	OS*	OS	OS*	OS
$\mathcal{O} = 0.83$	99.2	97.6	100.0	97.3	89.3	88.2
$\mathcal{O} = 0.67$	100.0	97.4	99.6	97.1	81.3	83.0
$\mathcal{O} = 0.35$	99.0	98.2	99.4	98.3	81.0	79.3
$\mathcal{O} = 0.03$	89.7	89.4	91.9	91.1	56.5	53.3

(b)

Table 4: (a)Sensitivity to the margin term τ for two cases of Office-31 and Office-Home, (b) Accuracy assessment for different openness values for different combinations of Office-31.



(a) (b) (c)

Fig. 4: (a)The training graphs showcasing the evolution of $\mathcal{L}_{SA}$ and $\mathcal{L}_{CLS} + \mathcal{L}_{margin}$ for **A,D** $\mapsto$ **W** (Office-31) for 50 epochs, (b) Effects of explicit source alignment on OS and OS^* for **A,D** $\mapsto$ **W**(Office-31). # in the figure refers to the first term of Equation 1. (c)Openness analysis of full model and model without $\mathcal{L}_{margin}$ for **D,W** $\mapsto$ **A** (Office-31),

for **D,W** $\mapsto$ **A** (a challenging scenario of Office-31). The full model in this case consistently produces superior OS scores.

Effects of aligning the source-domains: In order to assess the importance of the source alignment term (first term in Equation 1), we train two separate models with and without the first term of Equation 1 for **A,D** $\mapsto$ **W** (Office-31). As can be observed in Figure 4(b), we observe sharp improvements of more than 10% both in OS and OS^* when source-domains are explicitly aligned.

Visualization: In Figure 3(c), we depict the t-SNE plot to highlight the discriminating nature of the shared feature space as provided by our full MOSDANET model for **A,D** $\mapsto$ **W** (Office-31). It can be seen that the open-set target-domain classes are mostly clustered around the center while the known-class samples of all the domains are properly overlapped with clear discrimination among the different categories. Besides, the evolution of different loss terms for the full model during training can be found in Figure 4(a) which shows early convergence.

5 Conclusions

We formally introduce the learning paradigm of multi-source open-set domain adaptation in this paper and propose a novel framework which seeks to learn a shared feature space for all the source and target domains under consideration. In the process, we explicitly align the source-domains using class information while an improved adversarial learning paradigm is introduced to map the known-class samples from the target-domain with the source-domains. We judiciously incorporate target-domain samples with pseudo known-class labels during training to encourage fine-grained domain alignment. We believe that the proposed problem paradigm opens a new set of possibilities that can be expanded. For instance, in the future, we would be interested to explore the inclusion of open-set categories in the different source domains.

Acknowledgment: B. Banerjee was partially supported by grant ECR-2017-000365 from SERB, DST.

References

1. Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D.: Domain separation networks. In: Advances in neural information processing systems. pp. 343–351 (2016)
2. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: Proceedings of the IEEE international conference on computer vision. pp. 2960–2967 (2013)
3. Fu, J., Wu, X., Zhang, S., Yan, J.: Improved open set domain adaptation with backpropagation. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 2506–2510. IEEE (2019)
4. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. arXiv preprint arXiv:1409.7495 (2014)
5. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2066–2073. IEEE (2012)
6. Guo, J., Shah, D.J., Barzilay, R.: Multi-source domain adaptation with mixture of experts. arXiv preprint arXiv:1809.02256 (2018)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
8. Hull, J.J.: A database for handwritten text recognition research. IEEE Transactions on pattern analysis and machine intelligence **16**(5), 550–554 (1994)
9. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
10. Jain, L.P., Scheirer, W.J., Boult, T.E.: Multi-class open set recognition using probability of inclusion. In: European Conference on Computer Vision. pp. 393–409. Springer (2014)
11. Jhuo, I.H., Liu, D., Lee, D., Chang, S.F.: Robust visual domain adaptation with low-rank reconstruction. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2168–2175. IEEE (2012)
12. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
14. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
15. Liu, H., Cao, Z., Long, M., Wang, J., Yang, Q.: Separate to adapt: Open set domain adaptation via progressive separation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2927–2936 (2019)
16. Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F.E.: A survey of deep neural network architectures and their applications. Neurocomputing **234**, 11–26 (2017)
17. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks. arXiv preprint arXiv:1502.02791 (2015)
18. Luo, Y., Zheng, L., Guan, T., Yu, J., Yang, Y.: Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2507–2516 (2019)

19. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NIPS Workshop on Deep Learning and Unsupervised Feature Learning (2011)
20. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. IEEE Transactions on Neural Networks **22**(2), 199–210 (2011)
21. Panareda Busto, P., Gall, J.: Open set domain adaptation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 754–763 (2017)
22. Patel, V.M., Gopalan, R., Li, R., Chellappa, R.: Visual domain adaptation: A survey of recent advances. IEEE signal processing magazine **32**(3), 53–69 (2015)
23. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. arXiv preprint arXiv:1812.01754 (2018)
24. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European conference on computer vision. pp. 213–226. Springer (2010)
25. Saito, K., Yamamoto, S., Ushiku, Y., Harada, T.: Open set domain adaptation by backpropagation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 153–168 (2018)
26. Scheirer, W.J., Jain, L.P., Boult, T.E.: Probability models for open set recognition. IEEE transactions on pattern analysis and machine intelligence **36**(11), 2317–2324 (2014)
27. Sun, B., Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation. In: European Conference on Computer Vision. pp. 443–450. Springer (2016)
28. Sun, S., Shi, H., Wu, Y.: A survey of multi-source domain adaptation. Information Fusion **24**, 84–92 (2015)
29. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7167–7176 (2017)
30. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5018–5027 (2017)
31. Wang, M., Deng, W.: Deep visual domain adaptation: A survey. Neurocomputing **312**, 135–153 (2018)
32. Wang, Z., She, Q., Ward, T.E.: Generative adversarial networks: A survey and taxonomy. arXiv preprint arXiv:1906.01529 (2019)
33. Yang, J., Yan, R., Hauptmann, A.G.: Cross-domain video concept detection using adaptive svms. In: Proceedings of the 15th ACM international conference on Multimedia. pp. 188–197. ACM (2007)
34. Zhao, H., Zhang, S., Wu, G., Moura, J.M., Costeira, J.P., Gordon, G.J.: Adversarial multiple source domain adaptation. In: Advances in Neural Information Processing Systems. pp. 8559–8570 (2018)
35. Zhao, H., Zhang, S., Wu, G., Moura, J.M., Costeira, J.P., Gordon, G.J.: Adversarial multiple source domain adaptation. In: Advances in Neural Information Processing Systems. pp. 8559–8570 (2018)