Cleaning Noisy Labels by Negative Ensemble Learning for Source-Free Unsupervised Domain Adaptation

Waqar Ahmed[†‡], Pietro Morerio[†] and Vittorio Murino[†*]
[†]Pattern Analysis & Computer Vision, Istituto Italiano di Tecnologia, Genova, Italy
[‡]Dipartimento di Ingegneria Navale, Elettrica, Elettronica e delle Telecomunicazioni, University of Genova, Italy
[*] Dipartimento di Informatica, University of Verona, Italy
{waqar.ahmed, pietro.morerio, vittorio.murino}@iit.it

Abstract

Conventional Unsupervised Domain Adaptation (UDA) methods presume source and target domain data to be simultaneously available during training. Such an assumption may not hold in practice, as source data is often inaccessible (e.g., due to privacy reasons). On the contrary, a pre-trained source model is usually available, which performs poorly on target due to the well-known domain shift problem. This translates into a significant amount of misclassifications, which can be interpreted as structured noise affecting the inferred target pseudo-labels. In this work, we cast UDA as a pseudo-label refinery problem in the challenging source-free scenario. We propose Negative Ensemble Learning (NEL) technique, a unified method for adaptive noise filtering and progressive pseudo-label refinement. NEL is devised to tackle noisy pseudo-labels by enhancing diversity in ensemble members with different stochastic (i) input augmentation and (ii) feedback. The latter is achieved by leveraging the novel concept of Disjoint Residual Labels, which allow propagating diverse information to the different members. Eventually, a single model is trained with the refined pseudo-labels, which leads to a robust performance on the target domain. Extensive experiments show that the proposed method achieves state-of-the-art performance on major UDA benchmarks, such as Digit5, PACS, Visda-C, and DomainNet, without using source data samples at all.

1. Introduction

Deep Convolutional Neural Networks (CNNs) have shown remarkable achievements in a variety of tasks [9]. However, to perform well, training and testing data are assumed to be drawn from the same distribution. This is unrealistic when the system needs to be deployed in real-world scenarios. Consequently, a model trained on some source domain often fails to generalize well on a related but different target domain, due to the well-known problem called *domain shift* [39, 29]. Since annotating data from every possible domain is expensive and sometimes even impossible, Unsupervised Domain Adaptation (UDA) methods seek to address such a problem by minimizing discrepancy across the domains or trying to learn domain-invariant feature embeddings, without accessing target label information.

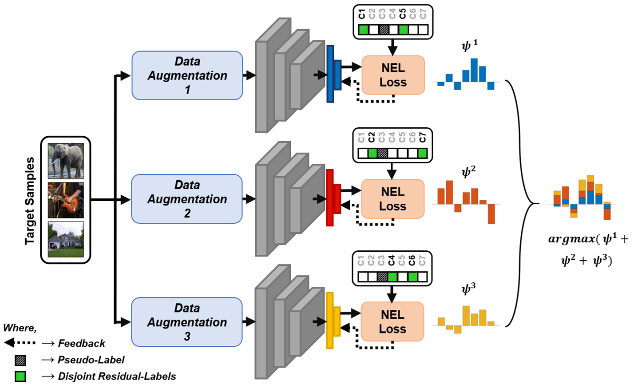


Figure 1: Illustration of the proposed method. In each iteration, a batch of target samples with different augmentation is fed to each ensemble member. Next, considering the inferred pseudo-label, different feedback is backpropagated by leveraging *Disjoint Residual Labels* with *Negative Ensemble Learning* (NEL) loss. This allows each member to learn diverse characteristics from data, possibly complementary, leading to a superior noise resilience and a stronger consensus leaning towards the actual class label.

Several research efforts have been devoted to developing UDA methods by either enforcing class-level feature distribution alignment [6, 8], matching moments [32, 5], applying domain-specific batch normalization [4], or adopting domain adversarial learning [2, 38]. However, these methods require joint access to both (labeled) source and (unlabeled) target data during training, making them unsuitable for scenarios where source data is inaccessible during the adaptation stage, or when source and target data are not available at the same time. Further, such solutions are also not viable when target data is provided incrementally at different times or if the source/target datasets are very large. Moreover, most UDA methods either focus on the single-source or single-target scenario with specific framework, regardless of the fact that data may belong to multiple source or target distributions, *e.g.*, images taken in different environments or obtained from the web (*e.g.,* sketches, photos).

To get rid of such restrictive assumptions, we propose to cast UDA as a pseudo-label refinement problem in a source data-free scenario. Consequently, unlike most of the former works, our method does not require any (target-style) data to be generated and can cope with single-source, multi-source, and multi-target UDA indifferently, making it easy to use and generalizable to the dataset of diverse complexity and challenges. Our approach only assumes the availability of a model pre-trained on the source domain to infer *pseudo-labels* of unlabeled target samples. Obviously, this results in a significant amount of incorrect acquired pseudo-labels, which is a consequence of the *domain shift* [39, 29]. Hence, it appears natural to adapt to the target domain by cleaning the pseudo-labels via a (re-)assignment process, so that a new model can be trained from scratch (or a pre-trained model can be fine-tuned) using the cleaned target labels.

To clean noisy pseudo-labels, we propose Negative Ensemble Learning (NEL) technique, a unified method for adaptive noise filtering and pseudo-label refinement. Our method takes advantage of several expert ensemble members, each trained using a batch of target samples with different stochastic data augmentation and a novel concept of *disjoint feedback*. The design of disjoint feedback requires two essential components (i) more than one trainable models, and (ii) *different labels* for each member and a supporting *loss function*. The latter is achieved by employing an indirect learning scheme, i.e., instead of using the inferred pseudo-label (out of total C classes) corresponding to target sample x, the remaining $(C-1)$ residual labels (RL) are equally distributed over all ensemble members. Next, separately for each member, the proposed NEL loss function attempts to minimize the confidence of corresponding disjoint residual labels (DRL) as in "x does not belong to either of them". Consequently, the collective form of feedback pushes confidence of inferred pseudo-label to rise.

The intuition behind is that, in case of incorrect pseudo-label, at least $N_e - 1$ members out of N_e (the total number of expert members in the ensemble network) should receive the correct information: stochastically sampling disjoint subsets of residual labels forces each ensemble member to learn different concepts, which is known to be beneficial in ensemble learning to reach a strong, hence more robust, consensus. Thus, such a consensus contributes to reaching higher confidence on clean pseudo-labels, allowing us to introduce a novel *fully adaptive* noise filtering technique to refine labels of the samples with low confidence (*i.e.*, noisy pseudo-labels) via reassignment. Finally, a standard supervised learning procedure is used to train a *single* model on the target domain data using the refined pseudo-labels featuring high confidence only.

The proposed pipeline obtains a significant noise reduction from the inferred pseudo-labels. With extensive experiments on various benchmarks yielding a wide range of shift-noise (PACS: 4% - DomainNet: 82.4%), we show that training on the target domain with refined pseudo-labels outperforms state-of-the-art UDA methods by a considerable margin. To summarise, the contributions of our work can be stated as follows:

- We propose a new, fully-adaptive method that dynamically filters out label noise and assigns cleaner pseudo-labels to noisy target samples. To do so, we introduce Negative Ensemble Learning, a new strategy that enhances diversity among members by different data augmentation and disjoint feedback, leading to improved noise resilience and a stronger consensus.
- Our method can naturally cope with the absence of source data during adaptation. It does not require new (target-style) data to be generated, avoiding the use of GAN-based models that are often difficult to train with stability. Also, it can deal with single/multi-source and multi-target UDA scenarios indifferently.
- We validate our method through detailed ablation analyses and extensive experiments on four well-known benchmarks, demonstrating its superiority over state-of-the-art UDA methods with a significant margin, *e.g.*, up to 21.8% better accuracy for PACS benchmark.

The remainder of the paper is organized as follows. In Section 2, we discuss related works in the literature. Section 3 describes the proposed method. Section 4 illustrates the experimental setup and reports the obtained results. Finally, conclusions and future work are drawn in Section 5.

2. Related Work

Unsupervised Domain Adaptation. Most of the existing UDA methods focused on cross-domain feature *alignment* either by employing discriminative class-conditional alignment [8], features and prototype alignment using reliable samples [6], or customized CNN models with domain alignment layers and feature whitening [34]. Other works proposed feature distribution *matching* by approximating joint distributions [40], matching graph [10], or matching moments [32]. However, such methods assume the co-existence of source and target data during training, making them unsuitable for more realistic scenarios where source data is inaccessible, *e.g.*, due to data-privacy issues.

Source-free UDA. A few recent works showed interest in source-free UDA. For example, [7] proposed a feature corruption and marginalization technique using few labeled source samples and [30] adapted the outputs from an off-the-shelf model to minimize distribution shift using some labeled target samples. An instance-level weighting method using negative classes is proposed in [22], which is highly dependent on a procurement stage requiring source data. Another approach leveraged a pre-trained source model to update the target model progressively by generating target-style samples through conditional generative adversarial networks [29], also combined with clustering-based regularization [26]. Similarly, to improve UDA performance in person re-identification task, [16] proposes a pseudo-label cleaning process with on-line refined soft pseudo-labels.

Our proposed approach lies in this category and takes partial inspiration from the methods developed for source-free UDA in [26, 29]. Here, a pre-trained source model is used to infer pseudo-labels of target data, and then a target-style sample generator is employed for adaptation. Like in many previous works, while we start by inferring pseudo-labels using a pre-trained model but, subsequently, we progressively refine such labels exploiting the consensus of an ensemble network, without generating any (target-style), thus data avoiding the use of GAN-based models that require careful hyperparameter balance to reach stability.

Ensemble Learning. Such methods exploit features extracted from multiple models through a diversity of data projections and bring forward the mutual consensus to achieve better performances than those obtained by any individual model [47]. A comprehensive review about ensemble methods is well illustrated in [13]. The importance of learning diverse contributions from data for classifier selection and parameters update is proposed in many works, for instance [45]. Also, multiple choice learning is employed in [15] to improve the accuracy of an ensemble of models.

We also drew inspiration from the general idea proposed in these works, which agree in stressing that *diversity* among members is beneficial for ensemble robustness. We differentiate from them by introducing a new way of inducing diversity in the members, *i.e.* we back-propagate different feedback to each member by leveraging the novel concept of Disjoint Residual Labels. This allows each member to learn diverse characteristics from data, possibly complementary, leading to a superior noise resilience and a stronger consensus leaning towards the actual class label.

Learning with Noisy Labels. Deep CNNs are capable of memorizing the entire data even when labels are noisy [21]. To overcome such overfitting, existing methods try to select a subset of possibly clean labels for training, *e.g.*, using two networks under a co-teaching framework [18], adopting meta-learning for exemplar weight estimation [46], applying an one-out filtering approach based on the local and global consistency [11], or investigating Negative Learning (NL) as an indirect learning method [21].

In these works, the type of label noise is an important factor to be considered. The above methods typically only consider random noise from selective or uniform distribution, which has a completely different structure from the label noise injected by the domain shift affecting the inferred pseudo-labels (see Section 3). Thus, [21] fails when the noise is not uniform, and the performance results actually affected by threshold sensitivity, which limits the generalization capability of the method across benchmarks. In contrast to a fixed threshold, our Negative Ensemble Learning method features a fully *adaptive* procedure to progressively filter out the structured noise affecting target pseudo-labels.

3. Proposed Method

In the context of UDA for a C-class classification task, we use a model pre-trained on source data to infer pseudo-labels of the entire target set $\mathcal{D}_t$ — such set of labels will be noisy due to domain shift [39, 29]. The standard training procedure *i.e.,* training with cross-entropy loss tries to *maximize* the probability of $\boldsymbol{x}$ belonging to the corresponding inferred pseudo-label $\tilde{y}$. But, in case of noisy pseudo-label $\tilde{y} \neq y_t$ (where y_t is the inaccessible actual target label), the model would undeniably be provided the wrong information which results in poor performance.

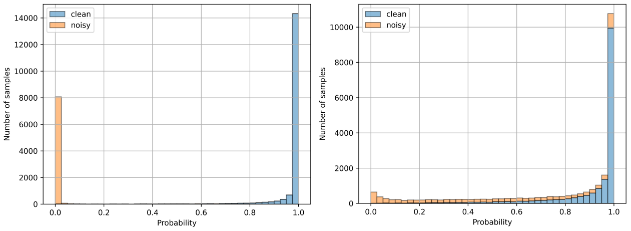
Instead, Negative Learning (NL) [21], can reduce such probability to $\frac{1}{(C-1)}$. NL refers in fact to an indirect learning method, which instead of using a given label — $\tilde{y}$ in our case — attempts to train the classifier using a complementary label $\bar{y}$ (randomly selected from $\{1, ..., C\}\backslash\{\tilde{y}\}$) as in "data sample $\boldsymbol{x}$ does not belong to $\bar{y}$". Since the chances of selecting a true label as a complementary label are low, NL decreases the risk of providing incorrect information.

Nevertheless, the existing NL method with a single network can not tackle *shift-noise* associated with inferred pseudo-labels. To understand better, let our CNN architecture be composed of a feature extractor $\nu_{\boldsymbol{\phi}}(.)$, a classifier $\psi_{\boldsymbol{\theta}}(.)$, and a *softmax* $\sigma(.)$, being $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ the related network parameters. The function $f : \mathcal{X} \to \mathbb{R}^C$, defined as $f(\boldsymbol{x}) = \sigma(\psi_{\boldsymbol{\phi}}(\nu_{\boldsymbol{\theta}}(\boldsymbol{x})))^1$, maps the input $\boldsymbol{x} \in \mathcal{X}$ to the C-dimensional vector of probabilities $\boldsymbol{p} \in \mathbb{R}^C$. Within a standard training procedure, namely Positive Learning (PL), the cross entropy loss function can be defined as:

$$\mathcal{L}_{PL}(\mathcal{D}_t) = -\mathbb{E}_{x_t \sim \mathcal{D}_t} \sum_{c=1}^{C} \mathbb{1}_{[c=\tilde{y}]} \log(\boldsymbol{p}) \qquad (1)$$

where $\mathbb{1}$ is an indicator function, $\mathcal{D}_t$ represents the unlabeled target domain and $\boldsymbol{p} = f(\boldsymbol{x}_t)$. Clearly, Eq. (1) pushes the probability p for the given pseudo-label $\tilde{y}$ towards $\boldsymbol{p}_{\tilde{y}} = 1$. On the contrary, NL aims at encouraging the probabilities of complementary labels $\bar{y}$ to move away

[1] Networks' parameters θ and ϕ will be omitted for brevity from now on.



(a) Symmetric-noise

(b) Shift-noise

Figure 2: Histogram showing the noise-filtering performance of [21] on MNIST. In both cases, the amount of noise equals 32.9% (cf. SVHN→MNIST shift-noise in Table 1).

from 1, actually pushing them towards $\boldsymbol{p}_{\tilde{y}} = 0$. The NL loss function would be so defined as:

$$\mathcal{L}_{NL}(\mathcal{D}_t) = -\mathbb{E}_{x_t \sim \mathcal{D}_t} \sum_{c=1}^{C} \mathbb{1}_{[c=\tilde{y}]} \log(1 - \boldsymbol{p}) \qquad (2)$$

While Eq. (2) optimizes the output probability of the complementary label to be close to zero, the probability values of other classes are increased. In such a contest, the samples carrying clean $\tilde{y}$ get higher confidence, whereas noisy ones struggle with scarcely confident scores, meeting the purpose of NL. Nevertheless, the fundamental limitation of naive NL [21] is that it is suitable for the noise showing a uniform distribution (*symmetric-noise*) only. As reported in Figure 2a, data samples with labels are assigned low confidence, resulting in effective noise separation.

Yet, a significant amount of noise is overfitted with high confidence by the network when we consider *shift-noise* — namely noise turned up during inferring pseudo-labels on the target using a model trained with the source (MNIST and SVHN, respectively, in the example of Figure 2b). Note that overfitting is not to be ascribed to the *amount* of noise, which was carefully balanced in the experiment, but rather to its *distribution* (see supplementary material for more details). So, NL works nicely when data is affected by symmetric noise but, when complying with shift-noise as in our case, important modifications are required to be able to clean pseudo-labels effectively.

To mitigate such limitations, we propose to employ Ensemble Learning which refers to concurrently training multiple networks of similar configuration. The idea is to create a set of experts trained in different ways, in order to produce predictions with low bias and high variance. Generally, an ensemble produces the final output as a weighted sum of all the experts' logits, *i.e.*, for a data sample $\boldsymbol{x}$, the final prediction can be obtained as:

$$\boldsymbol{p}_e = \sigma(\sum_{k=1}^{N_e} \beta^k \psi^k(\nu^k(\boldsymbol{x})) \qquad (3)$$

where N_e corresponds to the number of experts in the ensemble network, and β^k is a set of weights modulating the contribution of each expert member.

Precisely, we set $\beta^k = 1, \forall k \in [1, N_e]$ and propose Negative Ensemble Learning loss for learning with noisy labels that, amplifies the diversity of the ensemble members by different stochastic (i) input augmentation, and (ii) feedback using Disjoint Residual Labels. The resulting strong consensus gives rise to the cleaner pseudo-labels, better than those obtained by any stand-alone network. The following sections discuss specific details of the proposed approach.

3.1. Adaptive Pseudo-Label Refinement

Problem Setup. The goal of UDA methods is to adapt a model pre-trained on a labelled source domain $\mathcal{D}_s = \{(\boldsymbol{x}_s^i, y_s^i)\}_{i=1}^{N_s}$ in order to generalize well on a different, yet related, unlabeled target domain $\mathcal{D}_t = \{\boldsymbol{x}_t^j\}_{j=1}^{N_t}$, where N_s and N_t denote the number of samples in the source and the target domain, respectively, and the label set $\mathcal{Y}$ is the same for the 2 domains, *i.e.* $\mathcal{Y}_s = \mathcal{Y}_t$. For the sake of generality, we assume to work with $M_d + 1$ domains: M_d source domains $\mathcal{D}_s$, where $s = \{1, ..., M_d\}$, and a target domain $\mathcal{D}_t$. Differently from many standard UDA methods, our proposed approach does not use any source data for adaptation, nor generate target-style data at any stage. Instead, we simply use only a pre-trained source model to infer pseudo-labels $\mathcal{P} = \{\tilde{y}^j\}_{j=1}^{N_t}$ on the target domain. Being aware that a severe noise (due to domain shift [39]) is affecting such labels resulting in a significant amount of incorrect labels, we propose a way to progressively filter out noisy target samples from the clean ones, and carry out pseudo-label refinement to obtain a cleaner set $\mathcal{P}$.

Pseudo-Label Refinement. The first step for our proposed method refers to inferring pseudo-labels $\mathcal{P}$ of unlabeled data samples of $\mathcal{D}_t$ using f_s, a model pre-trained on the labeled source samples from $\mathcal{D}_s$. In this context, for *single-source* and *multi-target* UDA, the model pre-trained on the chosen source data is used to infer pseudo-labels of target domain(s) being considered. For *multi-source* UDA, we often have a single model pre-trained on aggregated data from all source domains, f_{agg}, which can be used to infer target pseudo-labels $\mathcal{P}$ as:

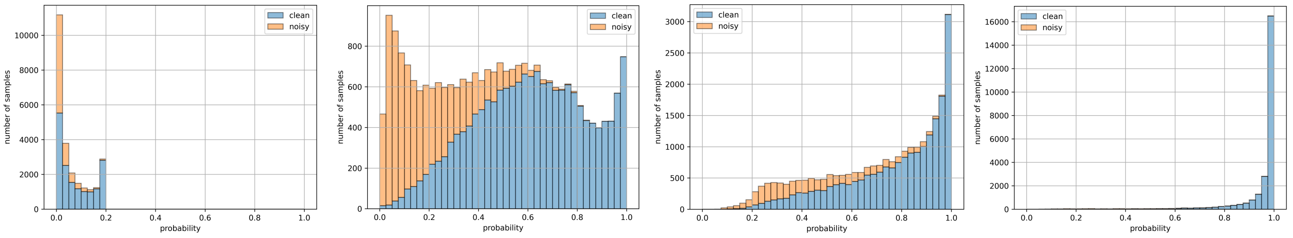
$$\tilde{y}^j = \operatorname{argmax}(\psi_{agg}(\nu_{agg}(\boldsymbol{x}_t^j))) \qquad \forall j \in \{1, ..., N_t\}, \quad (4)$$

Subsequently, the proposed label refinement procedure is carried out. In particular, to obtain robust and cleaner pseudo-labels by employing ensemble network, we use moving average of N_a previous ensemble output predictions. For a certain sample x, this results to:

$$\boldsymbol{p} = \sigma(\frac{1}{N_a \cdot N_e} \sum_{l=1}^{N_a} \sum_{k=1}^{N_e} \psi_e^{k,l}(\nu_e^{k,l}(\boldsymbol{x})) \qquad (5)$$

where we set $N_a = 10$ for all the experiments in this study.[2]

[2] We anticipate that this is a non sensitive parameter, it has not a relevant



(a) epochs = 1

(b) epochs = 10

(c) epochs = 25

(d) epochs = 50

Figure 3: Prediction-confidence trend during training and pseudo-label refinement by our proposed NEL method in case of SVHN→MNIST source-free UDA. Almost all samples are predicted with very low confidence at the beginning (a). As the network starts learning, noisy samples are segregated in a low confidence interval. Only if confidence is lower than γ (eq. 6), pseudo-labels are reassigned. Noise is thus progressively reduced (c-d). (Best viewed in color)

Now, let us define $\tilde{p}$ as the confidence of the pseudo-label $\tilde{y}$, namely the entry of the obtained probability vector $\boldsymbol{p}$ corresponding to $\tilde{y}$. We categorize the target samples with $\tilde{p} > \alpha$ as High Confidence Samples (HCS) which enable us to define the ratio of HCS over the total number of samples as:

$$\gamma = \frac{\#\ of\ HCS}{N_t} \tag{6}$$

Thus, the γ threshold derived by parameter α and instantaneous generalization capability of the ensemble network is the key behind adaptive nature of the noise filtering ability of the proposed method. So, the pseudo-label of each sample $\boldsymbol{x}_t^j$ is updated/retained according to the following condition:

$$\tilde{y}^j(n) = \begin{cases} \text{argmax}(\boldsymbol{p}^j), & \text{if } \tilde{p}^j < \gamma \\ \tilde{y}^j(n-1), & \text{otherwise} \end{cases} \quad \forall j \tag{7}$$

where n denotes the epoch number. The intuition behind such reassignment rule can be drawn from the trend of the prediction confidence along training. As shown in Figure 3a and 3b, with the growing number of epochs, noisy samples remain towards low confidence regime and clean samples obtain high confidence progressively. In Figure 3b, the ratio of HCS (with $\alpha = 0.9$) gives $\gamma \approx 0.15$ that corresponds to a confidence region $[0, \gamma] = [0, 0.15]$ in which the noisy samples are prevalent, hence they are the best candidates to be subjected to label reassignment. Consequently, pseudo-label refinement is achieved progressively during training in an adaptive manner (see Figure 3c and 3d, where total noise is progressively reduced). We ablate to find optimal value of α in Section 4.1.

3.2. Negative Ensemble Learning

The adaptive pseudo-label refinement procedure discussed in Section 3.1 heavily depends on the diversity exists among ensemble members. Our approach induces such a diversity by different stochastic data augmentation and feedback. The latter is achieved by employing *Residual Labels (RL)* — randomly chosen complementary labels other than the inferred pseudo-label — a key attribute of our proposed Negative Ensemble Learning (NEL) loss that we define as:

$$\mathcal{L}_{ENL}(\mathcal{D}_t) = -\mathbb{E}_{x_t \sim \mathcal{D}_t} \frac{1}{N_{RL}} \sum_{c=1}^{C} \mathbb{1}_{[c \in RL]} log(1 - \boldsymbol{p}_c) \tag{8}$$

The NEL loss in Eq. 8 is used to train each member independently, where N_{RL} refers to the number residual labels. Thus, for any value $N_{RL} > 1$, the proposed approach can influence the training process in three ways: (a) The likelihood of the actual-label y_t being randomly picked as one of the residual labels increases by factor of $\frac{N_{RL}}{C-1}$, which is bad. (b) In case $y_t \cap RL = \varnothing$, the training is accelerated with the stronger feedback provided by the multiple contributions of RLs, which is good. (c) In case $y_t \cap RL \neq \varnothing$, instead of providing entirely wrong feedback using Eq. 2, the impact of wrong feedback is mitigated by a factor of $\frac{N_{RL}-1}{N_{RL}}$, and this is again good. In fact, gradients will follow a mean direction according to Eq. 8. Although cases (b) & (c) are essential advantages of using multiple RL, (a) is a downside. To understand the balance among these aspects, we ablate on the different values of N_{RL} in Section 4.1.

However, we found that best results are obtained using a completely *disjoint* random subset of residual labels (DRL). Not only this allows each member to receive a different feedback (thus enhancing the ensemble's diversity), but also restricts the possibility of receiving wrong feedback to one member only. Thus, in this paper, we don't use specific number of RL, rather, we use equally distributed DRL over all ensemble members.

Further, to induce additional diversity in ensemble expert members, we consider several standard stochastic data augmentation strategies including the composition of (i) spatial/geometric transformation via random cropping (with uniform area = 0.08 to 1.0 and aspect-ratio = $\frac{3}{4}$ to $\frac{4}{3}$) followed by resizing to the original size, (ii) affine transformation followed by Gaussian blur, and (iii) color distortion. We found that composition of different stochastic data augmentation is crucial to avoid noise overfitting and extend diversity in the ensemble network (See Section 4.1).

effect on the performance as long as $N_a \geq 5$.

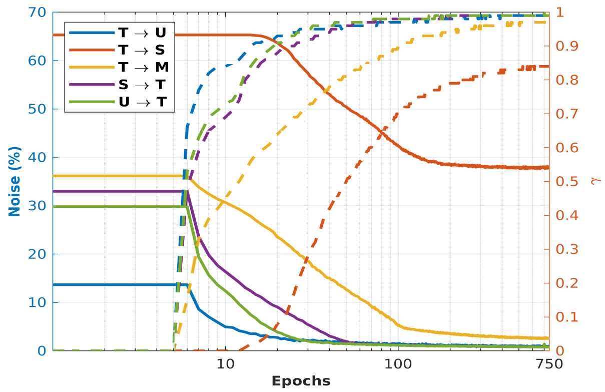


Figure 4: Correlation between adaptiveness of γ threshold (right y-axis, dashed lines) and progressive noise reduction (left y-axis, solid lines) achieved by NEL during training for various amount of noise. Legend: ***T***: *MNIST,* ***S***: *SVHN,* ***U***: *USPS, and* ***M***: *MNIST-M.*

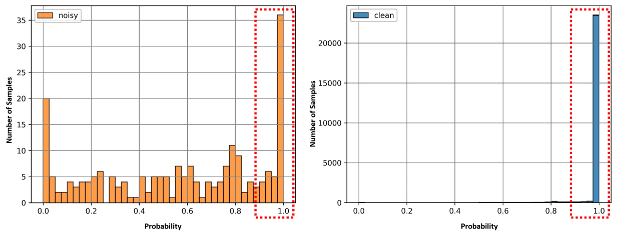


Figure 5: Distribution of remaining noise in refined pseudo-labels after SVHN→MNIST UDA. The highlighted bars (in a rectangle) represent the set of samples with confidence greater than $\alpha = 0.9$.

All these ingredients concur in making proposed method capable of filtering out different amounts of noise in an adaptive manner while progressively refining pseudo-labels. In Figure 4, the profiles related to the right y-axis (dashed lines) show the capability of the threshold γ to adapt to different single-source UDA cases, while the profiles related to the left y-axis show the corresponding noise reduction. In particular, it is worth noting that no pseudo-label refinement is applied during the first few epochs until the ensemble gets a mature state of generalization capability. As the ensemble members get more and more confident in the clean examples, label reassignment becomes more rigorous preventing the network to overfit to noisy samples.

After a certain number of training epochs, the pseudo-label refinement process stalls down to an insignificant noise reduction rate (see Figure 4). Therefore, instead of pushing refinement process more for several epochs to achieve further small reduction, we apply standard supervised learning only using high confidence samples determined by α. This is done using only one model (the final target model) for a fair comparison with state-of-the-art methods. As shown in Figure 5, the remaining noise is distributed over the entire probability spectrum, whereas the majority of the clean samples are predicted with high confidence (red box in the right plot). Therefore, only a small fraction of noisy samples affects training.

4. Experiments

We consider the image classification task to comprehensively evaluate the proposed method on major UDA benchmarks including Digit5 (MNIST [23], SVHN [31], USPS [12], MNIST-M [14], and Synthetic-Digits [14]), PACS [25], VisDA-C [33], and DomainNet [32]. For all the benchmarks, we use batch-size $= 32$, $\alpha = 0.9$ and Adam as the optimizer with a weight decay of $5e^{-4}$. The base learning rate is set to $1e^{-4}$ and the feature extractors are optimized with a learning rate of $1e^{-5}$. The feature extractor of ensemble members are initialized with a pre-trained source model. For single-source and multi-target UDA, we consider one pre-trained source model to adapt on every target domain. For multi-source UDA, each domain is selected as the target domain while the rest of the domains are treated as aggregated source (according to Eq. 4).

Digit5 refers to a set of digit benchmarks. In this paper, following [32], we sample a subset of 25000 images from the training and 9000 images from the testing set for MNIST, MINST-M, SVHN, and Synthetic-Digits. Since USPS contain a total of 9298 images, we use standard train-test splits. To keep comparable image resolution, we resize all images to 32×32 and a naive 3-layer CNN is used as ensemble members. For single-source and multi-source UDA, label refinement takes 750 and 300 epochs, respectively, whereas the final target model is trained for 200 epochs.

PACS contains 4 domains, namely *(Art-Painting, Cartoon, Photo, and Sketch)*. There are only 9991 images of 227x227 resolution from 7 object categories that accommodates a large domain shift due to the different image style depictions. We use ResNet-18 as ensemble members. For single-source, multi-target and multi-source UDA, label refinement takes 200, 200 and 100 epochs, respectively, whereas the final target model is trained for 200 epochs.

Visda-C is a challenging large-scale benchmark attempting to bridge the significant synthetic-to-real domain gap across 12 object categories. We follow standard protocol in which the source domain (training split) contains 152K synthetic images and the target domain (testing split) contains 72K real images. We resize all images to 256×256 resolution and use ResNet-101 as ensemble members. Label refinement takes 150 epochs and just 25 epochs were found to be enough for training the final target model.

DomainNet is by far the largest UDA benchmark with 6 domains, 600K images and 345 categories. We resize all images to 256×256 and use ResNet-101 as ensemble members. It was mainly developed for multi-source UDA task for which our proposed label refinement takes 100 epochs for *Infogragh* and *Quickdraw* domain, while 40 epoch were enough for the rest. For training the final target model, 100

	Single-Source UDA							Multi-Source UDA					
Source	***T***	***T***	***T***	***S***	***U***	Avg.		***M,S, D,U***	***T,S, D,U***	***T,M, D,U***	***T,M, S,U***	***T,M, S,D***	Avg.
Target	***U***	***S***	***M***	***T***	***T***			***T***	***M***	***S***	***D***	***U***	
ATT [36]	–	52.8	94.0	85.8	–	–	DCTN [42]	–	70.9	77.5	–	–	–
SBA [35]	97.1	50.9	**98.4**	74.2	87.5	81.6	MM [32]	98.4	72.8	81.3	89.5	96.1	87.6
MALT [28]	97.0	**78.7**	71.4	98.7	20.7	73.3	OML [24]	98.7	71.7	84.8	91.1	97.8	88.8
MTDA [17]	94.2	52.0	85.5	84.6	91.5	81.5	CMSS [44]	99.0	75.3	88.4	**93.7**	97.7	90.8
GPLR [29]	89.3	63.4	94.3	97.3	91.8	87.5							
NEL	**97.4**	61.6	95.4	**99.2**	**99.2**	**90.6**		**99.1**	**95.5**	**89.6**	90.0	**97.8**	**94.4**

Table 1: Classification accuracy on Digit5 with a naive 3-layer CNN. Legend: ***T***: *MNIST,* ***S***: *SVHN,* ***U***: *USPS,* ***M***: *MNIST-M, and* ***D***: *Synthetic-Digits.*

	Multi-Target UDA								Multi-Source UDA				
Source	***P***			***A***			Avg.		***C,P,S***	***A,P,S***	***A,C,S***	***A,C,P***	Avg.
Target	***A***	***C***	***S***	***P***	***C***	***S***			***A***	***C***	***P***	***S***	
1-NN*	15.2	18.1	25.6	22.7	19.7	22.7	20.7	DD [27]	87.5	87.0	96.6	71.6	85.7
ADDA*	24.3	20.1	22.4	32.5	17.6	18.9	22.6	SIB [19]	88.9	89.0	98.3	82.2	89.6
DSN*	28.4	21.1	25.6	29.5	25.8	24.6	25.8	OML [24]	87.4	86.1	97.1	78.2	87.2
ITA*	31.4	23.0	28.2	35.7	27.0	28.9	29.0	RABN [41]	86.8	86.5	98.0	71.5	85.7
KD [1]	24.6	32.2	**33.8**	35.6	46.6	**57.5**	46.6	JiGen [3]	84.8	81.0	97.9	79.0	85.7
								CMSS [44]	88.6	**90.4**	96.9	82.0	89.5
NEL	**80.1**	**76.1**	25.9	**96.0**	**82.8**	49.8	**68.4**		**90.8**	89.5	**98.8**	**85.2**	**91.1**

Table 2: Classification accuracy on PACS with ResNet18. * results are taken from [17]. Legend: ***A***: *Art-Painting,* ***C***: *Cartoon,* ***P***: *Photo, and* ***S***: *Sketch.*

	Single-Source UDA						
Source	***P***	***P***	***P***	***A***	***A***	***A***	Avg.
Target	***A***	***C***	***S***	***P***	***C***	***S***	
NEL	**82.6**	**80.5**	**32.3**	**98.4**	**84.3**	**56.1**	**72.4**

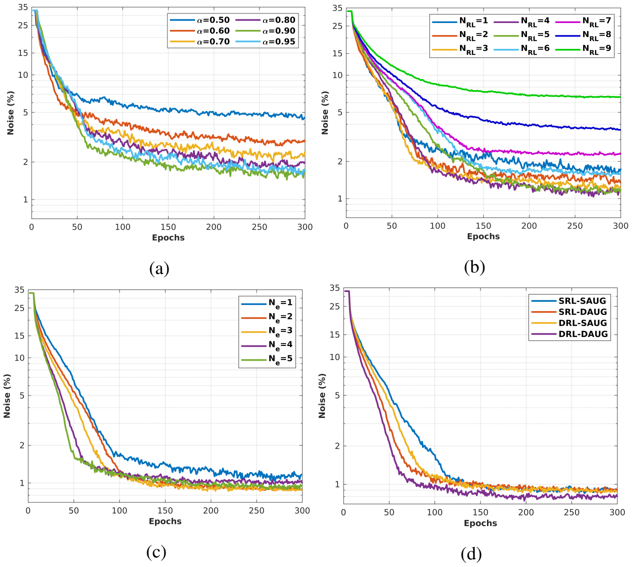
Table 3: Classification accuracy on PACS with ResNet18.

epochs were sufficient in all the cases.

4.1. Ablation study

We ablate the design choices described in Section 3 on the SVHN→MNIST adaptation task. It is important to note that the parameters estimated here are then used in all subsequent experiments, demonstrating the little sensitivity of the proposed method to such coefficients.

We consider $N_e = 1$ (*i.e.*, no ensemble) to ablate the different values of α. Figure 6a shows that $\alpha = 0.90$ results in the highest noise reduction, whereas $\alpha = 0.50$ is the least effective. These findings make sense, since with $\alpha = 0.50$, the reassignment would start too early (*i.e.*, as soon as some example has confidence greater than 0.5, since γ is always zero beforehand), when the network is not yet "ready" for it. Instead, with $\alpha = 0.90$, reassignment will start only when some samples have very high confidence. From that moment on, γ starts adapting, so that the more the samples with high confidence, the more permissive the threshold γ will be. The reason $\alpha = 0.95$ results slightly less effective is because a relatively higher number of noisy samples overfits before they are reassigned. Keeping $N_e = 1$ and $\alpha = 0.9$, we ablate on the different values of parameter N_{RL} (Figure 6b). Results show that $N_{RL} = 3, 4$, or 5 can be considered as the legitimate choice. Further, the results shown in Figure 6c exhibit the improvement achieved by the RL approach in comparison to $N_e = 1$ (with $N_{RL} = 4$, *i.e.*, the best choice found in Figure 6b). Though faster noise reduction is achieved with higher number of ensemble members, $N_e = 3$ can be regarded as the optimal choice considering performance *vs.* computational cost trade-off.



(a)

(b)

(c)

(d)

Figure 6: Ablation study considering SVHN→MNIST UDA task to determine optimal parameters of our proposed NEL method. (a): Single model is trained with 1 residual-label (RL) to choose the best α required to compute adaptive noise filtering threshold γ. (b): Searching for the right number of RL *i.e.,* N_{RL}. (c): Searching for the optimal number of members in the ensemble network ($N_{RL} = 4$ is used for $N_e = 1$). (d): Investigating the effect of same/disjoint RL (SRL and/or DRL) and same/different data augmentation (SAUG and/or DAUG) in all four possible scenarios.

Methods	***plane***	***bcycl***	***bus***	***car***	***horse***	***knife***	***mcycl***	***person***	***plant***	***skate***	***train***	***truck***	Avg.
MCD [37]	87.0	60.9	83.7	64.0	88.9	79.6	84.7	76.9	88.6	40.3	83.0	25.8	71.9
GPDA [20]	83.0	74.3	80.4	66.0	87.6	75.3	83.8	73.1	90.1	57.3	80.2	37.9	73.3
SAFN [43]	93.6	61.3	84.1	70.6	94.1	79.0	**91.8**	79.6	89.9	55.6	89.0	24.4	76.1
DSBN [4]	**94.7**	**86.7**	76.0	72.0	**95.2**	75.1	87.9	81.3	**91.1**	68.9	88.3	45.5	80.2
DADA [38]	92.9	74.2	82.5	65.0	90.9	**93.8**	87.2	74.2	89.9	71.5	86.5	48.7	79.8
NEL	94.5	60.8	**92.3**	**87.3**	87.3	93.2	87.6	**91.1**	56.9	**83.4**	**93.7**	**86.6**	**84.2**

Table 4: Classification accuracy on Visda-C with ResNet101.

Target	***C***	***I***	***P***	***Q***	***R***	***S***	Avg.
MM [32]	58.6	26.0	52.3	6.3	62.7	49.5	42.6
OML [24]	62.8	21.3	50.5	15.4	64.5	50.4	44.1
CMSS [44]	64.2	**28.0**	53.6	16.0	63.4	53.8	46.5
NEL	**68.3**	22.1	**54.7**	**22.8**	**67.3**	**57.1**	**48.7**

Table 5: Classification accuracy on DomainNet with ResNet101. For each target, the rest of the domains are considered as source (multi-source UDA). Legend: ***C***: *Clipart,* ***I***: *Infograph,* ***P***: *Painting,* ***Q***: *Quickdraw,* ***R***: *Real, and* ***S***: *Sketch.*

Also, as shown in Figure 6d, just one type of augmentation for all the ensemble members (ref. SAUG in the figure) is not enough. Also, the results validate the ineffectiveness of using same residual labels (SRL in the figure) for all. The best noise reduction is achieved using Disjoint Residual Labels along with different stochastic augmentations, DRL and DAUG, respectively.

4.2. Performances

The reported results in Table 1-5 present the average accuracy of 3 runs[3]. In Table 1 *(left)*, we compare our method (NEL) with the existing methods which address the challenging *MNIST→SVHN* and *MNIST→MNIST-M* tasks in a multi-target UDA framework. In both cases, the source contains gray-scale images, and the target holds colored (RGB) images, carrying a massive distribution gap across domains for which our method achieves third and second-best performance, respectively. Nevertheless, by outperforming in 3 out of 5 cases, our method achieves state-of-the-art average accuracy. For multi-source UDA task in Table 1 *(right)*, the performance of proposed method is slightly affected while adapting to *Synthetic-Digits* benchmark. In the remaining 4 out of 5 cases, our method outperforms existing methods and achieves state-of-the-art average accuracy. Especially, the difference is substantial in the case of *MNIST-M*.

In Table 2 *(left)*, we compare NEL with the existing methods addressing multi-target UDA on PACS. As can be noticed, despite the sub-optimal performance in 2 cases, our method achieves superior average accuracy. For multi-source UDA, we compare recent works in Table 2 *(right)*. Also in this framework, our method consistently outperforms existing methods, with only in one case getting lower, yet comparable, accuracy. To the best of our knowledge, we are the first to report single-source UDA results on PACS. So, in Table 3, we consider similar pairs as of Table 2 *(left)* to evaluate the performance difference. As expected, single-source UDA brings comparatively better performance because of the pairwise UDA. In Table 4, along with 2 comparable results for Visda-C, the proposed method achieves superior performance in 6 out of 12 categories that give rise to state-of-the-art average accuracy on such a challenging benchmark. Also In Table 5, except one case, NEL consistently outperforms existing methods despite the large number of classes and discrepancy across domains.

Discussion. In the pseudo-label refinery framework, the single-source and multi-target UDA scenarios can be considered as the most challenging tasks since the pre-trained source model is optimized for one particular data distribution only. Consequently, inferred pseudo-labels are affected by a relatively higher amount of shift-noise with respect to the multi-source UDA scenario. Thus, in such cases, NEL requires a bit larger amount of epochs for filtering noise and, thus, refining pseudo-labels. On the other hand, starting from a better pre-trained source model in a multi-source UDA scenario, NEL performs better and faster. Moreover, there is no existing method in the literature that targets all three frameworks (*i.e.*, single-source, multi-target, and multi-source UDA) at a time. To sum up, NEL outperforms existing methods in all scenarios, even without using source data, which highlights the general applicability of the proposed method to cope with challenging tasks of different levels of complexity.

5. Conclusions

In this work, we cast UDA as a pseudo-label refinery problem in the challenging source-free scenario. We propose Negative Ensemble Learning technique, which takes advantage of different data augmentation and feedback using Disjoint Residual Labels to diversify the learning of the ensemble members. Thanks to this new training procedure, we were able to obtain an extraordinary cleaning of the target data labels. It requires a minimal tuning of parameters (estimated once and fixed for all the experiments), and can work in single-source, multi-target, and multi-source scenarios indifferently, unlike the existing methods in the literature. Results demonstrate the actual goodness of the proposed approach, outperforming the state-of-the-art average performances in all the challenging public benchmarks.

[3] Additional details, such as standard deviations and remaining noise in refined pseudo-labels are provided in the supplementary material

References

[1] Atif Belal, Madhu Kiran, Jose Dolz, Louis-Antoine Blais-Morin, Eric Granger, et al. Knowledge distillation methods for efficient unsupervised adaptation across multiple domains. *Image and Vision Computing*, page 104096, 2021.

[2] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. Learning disentangled semantic representation for domain adaptation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 2060–2066. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

[3] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019.

[4] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7354–7362, 2019.

[5] Chao Chen, Zhihang Fu, Zhihong Chen, Sheng Jin, Zhaowei Cheng, Xinyu Jin, and Xian-Sheng Hua. Homm: Higher-order moment matching for unsupervised domain adaptation. In *AAAI*, 2020.

[6] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 627–636, 2019.

[7] Boris Chidlovskii, Stephane Clinchant, and Gabriela Csurka. Domain adaptation in the absence of source domain data. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 451–460, 2016.

[8] Safa Cicek and Stefano Soatto. Unsupervised domain adaptation via regularized conditional alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1416–1425, 2019.

[9] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, and Gulshan Kumar. A survey of deep learning and its applications: A new paradigm to machine learning. *Archives of Computational Methods in Engineering*, pages 1–22, 2019.

[10] Debasmit Das and C. S. George Lee. Graph matching and pseudo-label guided deep unsupervised domain adaptation. In Věra Kůrková, Yannis Manolopoulos, Barbara Hammer, Lazaros Iliadis, and Ilias Maglogiannis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2018*, pages 342–352, Cham, 2018. Springer International Publishing.

[11] Bruno Klaus de Aquino Afonso and Lilian Berton. Identifying noisy labels with a transductive semi-supervised leave-one-out filter. *Pattern Recognition Letters*, 2020.

[12] John S Denker, WR Gardner, Hans Peter Graf, Donnie Henderson, Richard E Howard, W Hubbard, Lawrence D Jackel, Henry S Baird, and Isabelle Guyon. Neural network recognizer for hand-written zip code digits. In *Advances in neural information processing systems*, pages 323–331, 1989.

[13] Xibin Dong, Zhiwen Yu, Wenming Cao, Yifan Shi, and Qianli Ma. A survey on ensemble learning. *Frontiers of Computer Science*, pages 1–18, 2020.

[14] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[15] Nuno C. Garcia, Sarah Adel Bargal, Vitaly Ablavsky, Pietro Morerio, Vittorio Murino, and Stan Sclaroff. DMCL: distillation multiple choice learning for multimodal action recognition. *CoRR*, abs/1912.10982, 2019.

[16] Yixiao Ge, Dapeng Chen, and Hongsheng Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In *International Conference on Learning Representations*, 2020.

[17] Behnam Gholami, Pritish Sahu, Ognjen Rudovic, Konstantinos Bousmalis, and Vladimir Pavlovic. Unsupervised multi-target domain adaptation: An information theoretic approach. *IEEE Transactions on Image Processing*, 29:3993–4002, 2020.

[18] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pages 8527–8537, 2018.

[19] Shell Xu Hu, Pablo Garcia Moreno, Yang Xiao, Xi Shen, Guillaume Obozinski, Neil Lawrence, and Andreas Damianou. Empirical bayes transductive meta-learning with synthetic gradients. In *International Conference on Learning Representations*, 2020.

[20] Minyoung Kim, Pritish Sahu, Behnam Gholami, and Vladimir Pavlovic. Unsupervised visual domain adaptation: A deep max-margin gaussian process approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4380–4390, 2019.

[21] Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 101–110, 2019.

[22] Jogendra Nath Kundu, Naveen Venkat, Rahul M V, and R. Venkatesh Babu. Universal source-free domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[23] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[24] Da Li and Timothy Hospedales. Online meta-learning for multi-source and semi-supervised domain adaptation. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 382–403, Cham, 2020. Springer International Publishing.

[25] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[26] Rui Li, Qianfen Jiao, Wenming Cao, Hau-San Wong, and Si Wu. Model adaptation: Unsupervised domain adaptation without source data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9641–9650, 2020.

[27] Massimiliano Mancini, Lorenzo Porzi, Samuel Rota Bulò, Barbara Caputo, and Elisa Ricci. Boosting domain adaptation by discovering latent domains. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3771–3780, 2018.

[28] Breton Minnehan and Andreas Savakis. Deep domain adaptation with manifold aligned label transfer. *Machine Vision and Applications*, 30(3):473–485, 2019.

[29] Pietro Morerio, Riccardo Volpi, Ruggero Ragonesi, and Vittorio Murino. Generative pseudo-label refinement for unsupervised domain adaptation. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 3130–3139, 2020.

[30] Arun Reddy Nelakurthi, Ross Maciejewski, and Jingrui He. Source free domain adaptation using an off-the-shelf classifier. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 140–145. IEEE, 2018.

[31] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[32] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019.

[33] Xingchao Peng, Ben Usman, Neela Kaushik, Dequan Wang, Judy Hoffman, and Kate Saenko. Visda: A synthetic-to-real benchmark for visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2021–2026, 2018.

[34] Subhankar Roy, Aliaksandr Siarohin, Enver Sangineto, Samuel Rota Bulo, Nicu Sebe, and Elisa Ricci. Unsupervised domain adaptation using feature-whitening and consensus loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9471–9480, 2019.

[35] Paolo Russo, Fabio M Carlucci, Tatiana Tommasi, and Barbara Caputo. From source to target and back: symmetric bi-directional adaptive gan. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8099–8108, 2018.

[36] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. Asymmetric tri-training for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 2988–2997. PMLR, 2017.

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

[38] Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *AAAI*, pages 5940–5947, 2020.

[39] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1521–1528, 2011.

[40] Jun Wen, Nenggan Zheng, Junsong Yuan, Zhefeng Gong, and Changyou Chen. Bayesian uncertainty matching for unsupervised domain adaptation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 3849–3855. International Joint Conferences on Artificial Intelligence Organization, 7 2019.

[41] Jiaolong Xu, Liang Xiao, and Antonio M López. Self-supervised domain adaptation for computer vision tasks. *IEEE Access*, 7:156694–156706, 2019.

[42] Ruijia Xu, Ziliang Chen, Wangmeng Zuo, Junjie Yan, and Liang Lin. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3964–3973, 2018.

[43] 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.

[44] Luyu Yang, Yogesh Balaji, Ser-Nam Lim, and Abhinav Shrivastava. Curriculum manager for source selection in multi-source domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[45] Xu-Cheng Yin, Kaizhu Huang, Hong-Wei Hao, Khalid Iqbal, and Zhi-Bin Wang. A novel classifier ensemble method with sparsity and diversity. *Neurocomputing*, 134:214–221, 2014.

[46] Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9294–9303, 2020.

[47] Zhi-Hua Zhou. *Ensemble methods: foundations and algorithms*. CRC press, 2012.