RESEARCH ARTICLE

MULTI-EPL: Accurate multi-source domain adaptation

Seongmin Lee, Hyunsik Jeon, U. Kang *

Seoul National University, Seoul, Republic of Korea

* ukang@snu.ac.kr

Abstract

Given multiple source datasets with labels, how can we train a target model with no labeled data? Multi-source domain adaptation (MSDA) aims to train a model using multiple source datasets different from a target dataset in the absence of target data labels. MSDA is a crucial problem applicable to many practical cases where labels for the target data are unavailable due to privacy issues. Existing MSDA frameworks are limited since they align data without considering labels of the features of each domain. They also do not fully utilize the target data without labels and rely on limited feature extraction with a single extractor. In this paper, we propose MULTI-EPL, a novel method for MSDA. MULTI-EPL exploits label-wise moment matching to align the conditional distributions of the features for the labels, uses pseudolabels for the unavailable target labels, and introduces an ensemble of multiple feature extractors for accurate domain adaptation. Extensive experiments show that MULTI-EPL provides the state-of-the-art performance for MSDA tasks in both image domains and text domains, improving the accuracy by up to 13.20%.

Introduction

Given multiple source datasets with labels, how can we train a target model with no labeled data? Large training data are essential for training deep neural networks. Collecting abundant data is, unfortunately, an obstacle in practice; even if enough data are obtained, manually labeling those data is prohibitively expensive. Using other available or much cheaper datasets would be a solution for these limitations; however, indiscriminate usage of other datasets often brings severe generalization error due to the presence of dataset shifts [1]. Unsupervised domain adaptation (UDA) tackles these problems where no labeled data from the target domain are available, but labeled data from other source domains are provided. Finding out domain-invariant features has been the focus of UDA since it allows knowledge transfer from the labeled source dataset to the unlabeled target dataset. There have been many efforts to transfer the knowledge from a single source domain to a target one. Most recent frameworks minimize the distance between two domains by deep neural networks and distance-based techniques such as discrepancy regularizers [2–4], adversarial networks [5, 6], and generative networks [7–9].

While the above-mentioned approaches consider a single source, we address multi-source domain adaptation (MSDA), which is very crucial and more practical in real-world applications as well as more challenging. MSDA is able to bring significant performance enhancement by virtue of accessibility to multiple datasets as long as multiple domain shift problems are resolved. Previous works have extensively presented both theoretical analysis [10–15] and models [14, 16–20] for MSDA. MDAN [14], DCTN [16], and MDDA [18] build adversarial networks for each source domain to generate features domain-invariant enough to confound domain classifiers. However, these approaches do not encompass the interactions among source domains, counting only shifts between source and target domain. M^{3}SDA [17] adopts a moment matching strategy but makes the unrealistic assumption that matching the marginal probability $p(\mathbf{x})$ would guarantee the alignment of the conditional probability $p(\mathbf{x}|y)$. Most of these methods also do not fully exploit the knowledge of the target domain, imputing to the inaccessibility of the labels. Furthermore, these methods require individual deep neural networks for each source domain as described in Fig 1, which have great redundancy and significantly increase the overall model complexity. LtC-MSDA configures prototypes of the features from each domain and learns the interaction between multiple domains deploying GCN. However, summarizing each domain into only one prototype cannot fully represent the feature distributions of the domain and therefore deteriorates the performance.

In this paper, we propose MULTI-EPL (Multi-source domain adaptation with Ensemble of feature extractors, Pseudolabels, and Label-wise moment matching), a novel MSDA framework that mitigates the limitations of these methods of not explicitly considering conditional probability $p(\mathbf{x}|y)$, and having great redundancy in their models. MULTI-EPL is illustrated in Fig 2. MULTI-EPL aligns the conditional probability $p(\mathbf{x}|y)$ by utilizing label-wise moment matching. We employ pseudolabels for the inaccessible target labels to maximize the usage of the target data. Moreover, we generate an ensemble of features from multiple feature extractors to capture rich information about labels. Extensive experiments show the superiority of MULTI-EPL (see Fig 3).

Our contributions are summarized as follows:

- **Method**. We propose MULTI-EPL, a novel approach for MSDA that effectively and efficiently obtains domain-invariant features from multiple domains by matching conditional probability $p(\mathbf{x}|y)$, utilizing pseudolabels for inaccessible target labels to fully exploit target data,
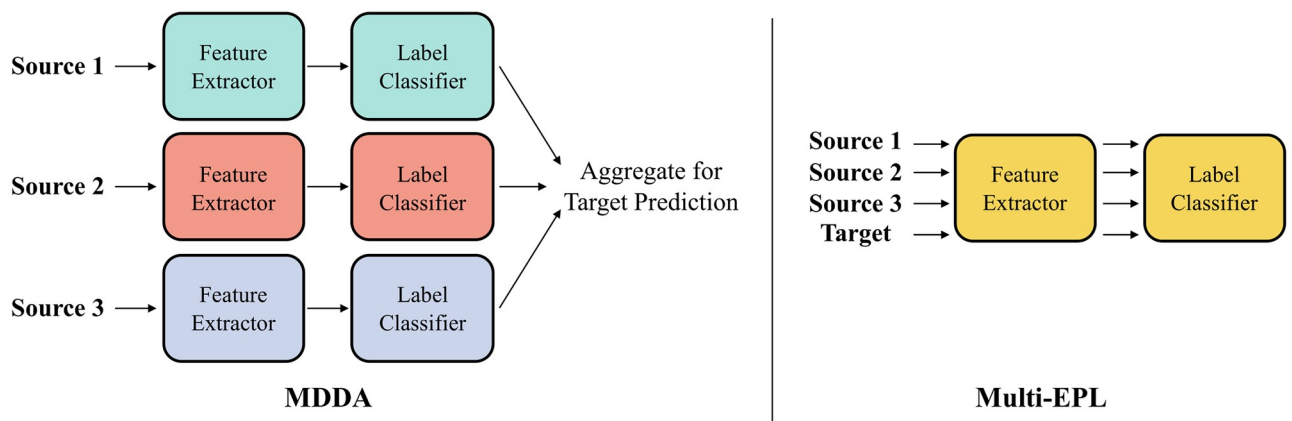

Fig 1. Overall model structure of MDDA and MULTI-EPL. To handle 3 source domains, most existing methods deploy 3 different sets of deep neural networks, while one single set is enough for MULTI-EPL. This allows MULTI-EPL to use ensemble learning without an excessive cost of model complexity.
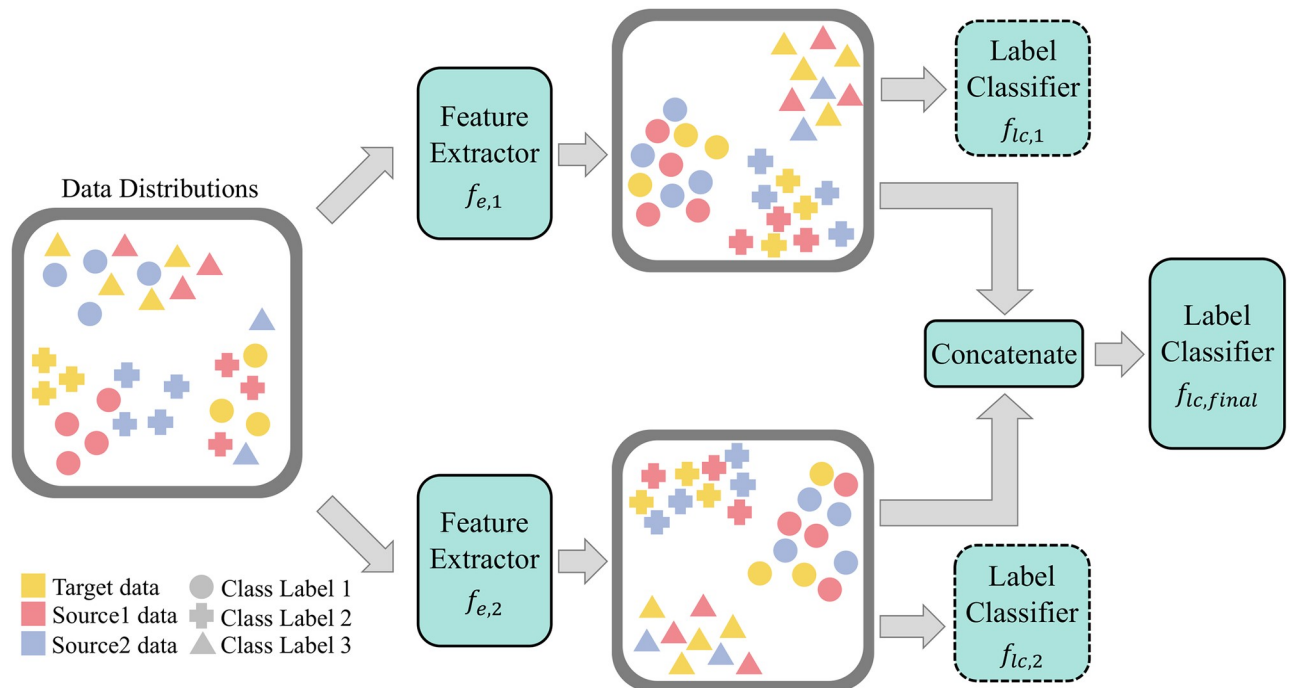
https://doi.org/10.1371/journal.pone.0255754.g001

Fig 2. Illustration of MULTI-EPL. MULTI-EPL consists of 2 pairs of feature extractors and label classifiers, and one final label classifier. Colors and symbols of the markers indicate domains and class labels of the data, respectively. The networks with the solid line are used for inference while the ones with the dashed line are used only for training.

https://doi.org/10.1371/journal.pone.0255754.g002

handling all the source domains with one single neural network, and using an ensemble of multiple feature extractors for further enhancement. It allows domain-invariant features to be extracted, capturing the intrinsic differences of labels.

- **Experiments**. We conduct extensive experiments on image and text datasets. We show that 1) MULTI-EPL provides the state-of-the-art performance, and 2) each of our main ideas significantly contributes to the superior performance.

In the rest of this paper, we first introduce the related works and describe our proposed method. Then, we experimentally evaluate the performance of MULTI-EPL and its competitors. The code for MULTI-EPL can be found in https://github.com/snudatalab/MultiEPL. Frequently used symbols are summarized in Table 1.

Related works

Single-source domain adaptation

Given a labeled source dataset and an unlabeled target dataset, single-source domain adaptation aims to train a model that performs well on the target domain. The challenge of single-source domain adaptation is to reduce the discrepancy between the two domains and to obtain appropriate domain-invariant features. Various discrepancy measures such as Maximum Mean Discrepancy (MMD) [2–4, 21, 22] and KL divergence [23] have been used as regularizers. Inspired by the insight that the domain-invariant features should exclude the clues about its domain, constructing adversarial networks against domain classifiers has shown superior performance. [7] and [9] deploy GAN to transform data across the source and target domains,
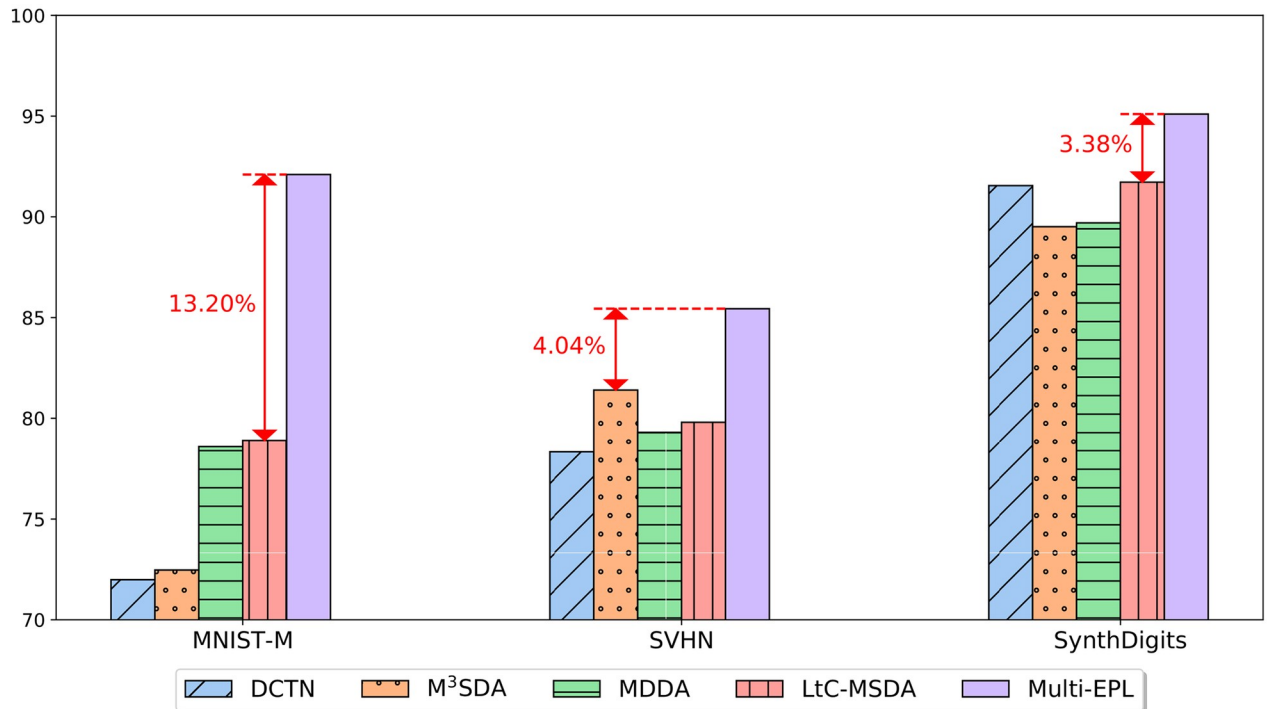
Fig 3. Accuracy of MULTI-EPL and its competitors on 3 cases with Digits-Five datasets.

while [5] and [6] leverage the adversarial networks to extract common features of the two domains. Unlike these works, we focus on *multiple* source domains.

Multi-source domain adaptation

Single-source domain adaptation should not be naively employed for multiple source domains due to domain shifts. Many previous works have tackled Multi-source Domain Adaptation (MSDA) problems theoretically. [11] establishes distribution weighted combining rule that the weighted combination of source hypotheses is a good approximation for the target hypothesis.

Table 1. Table of symbols.

Symbols	Definition
N	Number of source domains.
$\mathcal{X}$	Data space.
$\mathcal{C}$	Label set.
$\mu_{\mathcal{D}}$	Data distribution of the domain $\mathcal{D}$.
$l_{\mathcal{D}}$	Labeling function of the domain $\mathcal{D}$.
$\mathcal{S}_i = (\mu_{\mathcal{S}_i}, l_{\mathcal{S}_i})$	i-th source domain.
$\mathcal{T} = (\mu_{\mathcal{T}}, l_{\mathcal{T}})$	Target domain.
$n_{\mathcal{S}_i}$	Number of instance in i-th source dataset.
$n_{\mathcal{T}}$	Number of instance in the target dataset.
$\mathbf{X}_{\mathcal{S}_i} = \{(\mathbf{x}_j^{\mathcal{S}_i}, y_j^{\mathcal{S}_i})\}_{j=1}^{n_{\mathcal{S}_i}}$	i-th source dataset. $\mathbf{x}_j^{\mathcal{S}_i}$: j-th instance in $\mathbf{X}_{\mathcal{S}_i}$, $y_j^{\mathcal{S}_i}$: label of $\mathbf{x}_j^{\mathcal{S}_i}$.
$\mathbf{X}_{\mathcal{T}} = \{\mathbf{x}_j^{\mathcal{T}}\}_{j=1}^{n_{\mathcal{T}}}$	Target dataset. $\mathbf{x}_j^{\mathcal{T}}$: j-th instance in $\mathbf{X}_{\mathcal{T}}$.

The rule is further extended to a stochastic case with joint distribution over the input and the output space in [13]. [12] proposes a general theory of how to sift appropriate samples out of multi-source data using expected loss. Efforts to find out transferable knowledge from multiple sources from the causal viewpoint are made in [24]. There have been salient studies on the learning bounds for MSDA. [10] finds the generalization bounds based on $\mathcal{H}\Delta\mathcal{H}$-divergence, which are further tightened by [14].

Frameworks for MSDA have been presented as well. [14] proposes learning algorithms based on the generalization bounds for MSDA. DCTN [16] resolves domain and category shifts between source and target domains via adversarial networks. TMDA [25] aligns multiple domains utilizing clustering and adversarial training. M³SDA [17] associates all the domains into a common distribution by matching the moments of the feature distributions of multiple domains. In [26], attempts to find out the common latent space of source and target domains are made, focusing on the visual sentiment classification tasks. MDDA [18] employs Wasserstein distance to figure out which data from which source domains are closely related to the target data. In LtC-MSDA [19], the interactions among multiple domains are learned by constructing a knowledge graph. However, most of these methods do not consider multimode structures [27] that differently labeled data follow distinct distributions, even if they are drawn from the same domain. Also, the domain-invariant features in these methods contain the label information for only one label classifier which leads these methods to miss a large amount of label information. Differently from these methods, our framework fully considers the multimode structures, handles the data distributions in a label-wise manner, and minimizes the label information loss considering multiple label classifiers.

Moment matching

Moment matching strategy has been used to minimize the discrepancy between source and target domains in domain adaptation. MMD regularizer [2–4, 21, 22] can be interpreted as the first-order moment matching while [28] addresses second-order moment maching of source and target distributions. [29] investigates the effect of higher-order moment matching. M³SDA [17] demonstrates that moment matching yields remarkable performance also with multiple sources. While previous works have focused on matching the moments of marginal distributions for single-source adaptation, we handle conditional distributions in multi-source scenarios.

Methods

In this section, we describe our proposed method, MULTI-EPL. We first formulate the problem definition and describe our main ideas. Then, we elaborate on how to match label-wise moment with pseudolabels and extend the approach by adding the concept of ensemble learning. Fig 2 shows the overview of MULTI-EPL.

Problem definition

Given a set of labeled datasets from N source domains $\mathcal{S}_1, \ldots, \mathcal{S}_N$ and an unlabeled dataset from a target domain $\mathcal{T}$, we aim to construct a model that minimizes the test error on $\mathcal{T}$. We formulate source domain $\mathcal{S}_i$ as a tuple of the data distribution $\mu_{\mathcal{S}_i}$ on data space $\mathcal{X}$ and the labeling function $l_{\mathcal{S}_i}: \mathcal{S}_i = (\mu_{\mathcal{S}_i}, l_{\mathcal{S}_i})$. Source dataset drawn with the distribution $\mu_{\mathcal{S}_i}$ is denoted as $\mathbf{X}_{\mathcal{S}_i} = \{(\mathbf{x}_j^{\mathcal{S}_i}, y_j^{\mathcal{S}_i})\}_{j=1}^{n_{\mathcal{S}_i}}$, where $n_{\mathcal{S}_i}$ is the number of instance in $\mathbf{X}_{\mathcal{S}_i}$. Likewise, the target domain and the target dataset are denoted as $\mathcal{T} = (\mu_{\mathcal{T}}, l_{\mathcal{T}})$ and $\mathbf{X}_{\mathcal{T}} = \{\mathbf{x}_j^{\mathcal{T}}\}_{j=1}^{n_{\mathcal{T}}}$, respectively, where $n_{\mathcal{T}}$ is

the number of instance in $\mathbf{X}_T$. We narrow our focus down to homogeneous settings in classification tasks: all domains share the same data space $\mathcal{X}$ and label set $\mathcal{C}$.

Overview

We propose MULTI-EPL based on the following observations: 1) existing methods focus on aligning the marginal distributions $p(\mathbf{x})$ not the conditional ones $p(\mathbf{x}|y)$, 2) knowledge of the target data is not fully employed as no target label is given, 3) existing methods that require separate neural networks for each source domain have considerable inefficiency in model size, and 4) there is a large amount of loss in label information since domain-invariant features are extracted for only one label classifier. Designing a method to solve these limitations entails the following challenges:

1. **Matching conditional distributions**. How can we align the conditional distribution, $p(\mathbf{x}|y)$, of multiple domains, not the marginal one, $p(\mathbf{x})$?

2. **Exploitation of the target data**. How can we fully exploit the knowledge of the target data despite the absence of the target labels?

3. **Maximization of the model efficiency**. How can we maximize the model efficiency and performance?

We propose the following main ideas to address the challenges:

1. **Label-wise moment matching**. We match the *label-wise* moments of the domain-invariant features so that the features with the same labels have similar distributions regardless of their original domains. This improves not only adaptation but also classification performance compared to the previous methods, which align features not considering labels and therefore cannot clearly separate differently labeled instances.

2. **Pseudolabels**. We use pseudolabels as alternatives to the target labels. While the existing MSDA methods have made only limited use of target data, this allows the intrinsic properties related to the label prediction of each target instance to be better reflected.

3. **Ensemble of feature representations**. We integrate multiple neural networks, each of which handles each source domain, into one neural network. For further improvement, we propose a variant of ensemble learning to concatenate features from multiple feature extractors. This enhances model performance without an extreme increase in model size, whereas the existing methods have significantly increased model size for better performance.

Our model MULTI-EPL consists of two pairs of feature extractor and label classifier, $(f_{e,1}, f_{lc,1})$ and $(f_{e,2}, f_{lc,2})$, and one final label classifier, $f_{lc,final}$ as shown in Fig 2. The feature extractors distill the domain-invariant features, which are aligned to have similar distributions regardless of their domains. Then, the label classifiers take the features from the corresponding feature extractor as inputs and predict their labels. Meanwhile, the features from $f_{e,1}$ and $f_{e,2}$ are concatenated and fed into the final label classifier $f_{lc,final}$. The label prediction of $f_{lc,final}$ is used for the final inference.

Label-wise moment matching with pseudolabels

We describe how MULTI-EPL matches conditional distributions $p(\mathbf{x}|y)$ of the features from multiple distinct domains. In MULTI-EPL, a feature extractor f_e and a label classifier f_{lc} lead the features to be domain-invariant and label-informative at the same time. The feature extractor f_e extracts features from data, and the label classifier f_{lc} receives the features and predicts the

labels for the data. We train f_e and f_{lc}, according to the losses for *label-wise moment matching* and *label classification*, which make the features domain-invariant and label-informative, respectively.

Label-wise moment matching.   To achieve the alignment of domain-invariant features, we define a label-wise moment matching loss as follows:

$$\mathcal{L}_{lmm,K} = \frac{1}{|\mathcal{C}|} \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{c \in \mathcal{C}} \left\| \frac{1}{n_{\mathcal{S}_i,c}} \sum_{j:y_j^{\mathcal{S}_i}=c} f_e(\mathbf{x}_j^{\mathcal{S}_i})^k - \frac{1}{n_{\mathcal{T},c}} \sum_{j:y_j^{\mathcal{T}}=c} f_e(\mathbf{x}_j^{\mathcal{T}})^k \right\|_2, \quad (1)$$

where K is a hyperparameter indicating the maximum order of moments considered by the loss, and $n_{\mathcal{D},c}$ is the number of data labeled as c in $\mathbf{X}_{\mathcal{D}}$. We introduce *pseudolabels* to determine the label c for the target data, which are determined by the outputs of the model currently being trained, to manage the absence of the ground truths for the target data. In other words, we compute $f_{lc}(f_e(\mathbf{x}^{\mathcal{T}}))$ using f_{lc} and f_e trained up to the previous iteration step to give the pseudolabels to the target data $\mathbf{x}^{\mathcal{T}}$.

The L2 norm term in Eq 1 measures how much k-th order moments of the features labeled as c are different when it comes to the source domain $\mathcal{S}_i$ and the target domain $\mathcal{T}$. The sum of the term for every possible c, i, and k gives the discrepancy of the feature distributions between the source domains and the target domain. By minimizing $\mathcal{L}_{lmm,K}$, the feature extractor f_e aligns data from multiple domains by bringing consistency in distributions of the features with the same labels. The data with distinct labels are aligned independently, taking account of the multimode structures that differently labeled data follow different distributions.

Label classification.   The label classifier f_{lc} gets the features projected by f_e as inputs and makes the label predictions. The *label classification loss* is defined as follows:

$$\mathcal{L}_{lc} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_{\mathcal{S}_i}} \sum_{j=1}^{n_{\mathcal{S}_i}} \mathcal{L}_{ce}(f_{lc}(f_e(\mathbf{x}_j^{\mathcal{S}_i})), y_j^{\mathcal{S}_i}), \quad (2)$$

where $\mathcal{L}_{ce}$ is the softmax cross-entropy loss. Minimizing $\mathcal{L}_{lc}$ separates the features with different labels so that each of them becomes label-distinguishable.

Ensemble of feature representations

In this section, we introduce ensemble learning for further enhancement. Features extracted with the method described in the previous section contain the label information for a single label classifier. However, each label classifier leverages only limited label characteristics, and thus the conventional scheme to adopt only one pair of feature extractor and label classifier captures only a small part of the label information. Our idea is to leverage an ensemble of multiple pairs of feature extractors and label classifiers in order to make the features to be more label-informative.

We train two pairs of feature extractor and label classifier in parallel following the label-wise moment matching approach explained in the previous section. We denote the two (feature extractor, label classifier) pairs as $(f_{e,1}, f_{lc,1})$ and $(f_{e,2}, f_{lc,2})$, and the resultant features from each feature extractor as $feat_1$ and $feat_2$ respectively. After obtaining two different feature mappings, we concatenate the two into one vector $feat_{final} = concat(feat_1, feat_2)$. The final label classifier $f_{lc,final}$ takes the concatenated feature as input and predicts the label of the feature.

MULTI-EPL: Accurate multi-source domain adaptation

Our final model MULTI-EPL consists of two pairs of feature extractor and label classifier, $(f_{e,1}, f_{lc,1})$ and $(f_{e,2}, f_{lc,2})$, and one final label classifier, $f_{lc,final}$. We train the model in an iterative manner where each iteration is composed of two steps. We first train the entire model except for the final label classifier with the loss $\mathcal{L}$:

$$\mathcal{L} = \sum_{n=1}^{2} (\mathcal{L}_{lc,n} + \alpha \mathcal{L}_{lmm,K,n}), \tag{3}$$

where $\mathcal{L}_{lc,n}$ is the label classification loss of the classifier $f_{lc,n}$, $\mathcal{L}_{lmm,K,n}$ is the label-wise moment matching loss of the feature extractor $f_{e,n}$, α is a hyperparameter that weights each of the loss term, and K is the hyperparameter for the maximum order of moments in $\mathcal{L}_{lmm,K,n}$. Then, the final label classifier is trained with respect to the label classification loss $\mathcal{L}_{lc,final}$ using the concatenated features from the multiple feature extractors. We repeat these two steps over and over until the number of iterations reaches the predetermined number of epochs.

Experimental results

We conduct experiments to answer the following questions.

Q1 Accuracy. How well does MULTI-EPL perform in classification tasks?

Q2 Ablation Study. How much does each component of MULTI-EPL contribute to performance improvement?

Q3 Effects of Degree of Ensemble. How does the performance change as the number of the pairs of the feature extractor and the label classifier increases?

Q4 Parameter Efficiency. What is the parameter efficiency of MULTI-EPL compared to the other methods?

Experimental settings

Datasets. We use three collections of datasets, Digits-Five, Office-Caltech10 [30], and Amazon Reviews [31], listed in Table 2. Digits-Five consists of five datasets for digit recognition: MNIST [32], MNIST-M [33], SVHN [34], SynthDigits [33], and USPS [35]. We set one of them as a target domain and the rest as source domains. Following the conventions in prior works [16, 17], we randomly sample 25000 instances from the source training set and 9000 instances from the target training set to train the model except for USPS for which the whole training set is used. The entire test set is exploited to evaluate the performance. Office-Caltech10 is for image classification with 10 categories that Office31 dataset and Caltech dataset have in common. It involves four different domains: Amazon, Caltech, DSLR, and Webcam. We double the number of instances by data augmentation and exploit all the original instances and augmented instances as training and test sets, respectively. Amazon Reviews contains customers' reviews on 4 product categories: Books, DVDs, Electronics, and Kitchen appliances. The instances are encoded into 5000-dimensional vectors and are labeled as being either positive or negative depending on their sentiments. We set each of the four categories as a target and the rest as sources. For all the domains, 2000 instances are sampled for training, and the rest of the data are used for the test.

Table 2. Summary of datasets.

	Datasets	Features	Class#	Training	Test
Digits-Five	MNIST	1x28x28	10	60000	10000
	MNIST-M	3x32x32	10	59001	9001
	SVHN	3x32x32	10	73257	26032
	SynthDigits	3x32x32	10	479400	9553
	USPS	1x16x16	10	7291	2007
Office-Caltech10	Amazon	3x300x300	10	958	958
	Caltech	Variable	10	1123	1123
	DSLR	3x1000x1000	10	157	157
	Webcam	Variable	10	295	295
Amazon Reviews	Books	5000	2	2000	4465
	DVDs	5000	2	2000	3586
	Electronics	5000	2	2000	5681
	Kitchen appliances	5000	2	2000	5945

Competitors. We use 5 MSDA algorithms, DCTN [16], M³SDA, M³SDA-β [17], MDDA [18], and LtC-MSDA [19] with state-of-the-art performances as baselines. All the frameworks share the same architecture for the feature extractor and the label classifier for consistency. For Digits-Five, we use convolutional neural networks based on LeNet5 [32]. For Office-Caltech10, ResNet50 [36] pretrained on ImageNet is used as the backbone architecture. For Amazon Reviews, the feature extractor is composed of three fully-connected layers each with 1000, 500, and 100 output units, and a single fully-connected layer with 100 input units and 2 output units is adopted for the label classifier. With Digits-Five, LeNet5 [32] and ResNet14 [36] without any adaptation are additionally investigated in two different manners: *Source Combined* and *Single Best*. In *Source Combined*, multiple source datasets are simply combined and fed into a model. In *Single Best*, we train the model with each source dataset independently and report the result of the best performing one. Likewise, ResNet50 and MLP consisting of 4 fully-connected layers with 1000, 500, 100, and 2 units are investigated without adaptation for Office-Caltech10 and Amazon Reviews, respectively.

Training details. We train our models for Digits-Five with Adam optimizer [37] with β_1 = 0.9, β_2 = 0.999, and the learning rate of 0.0004 for 100 epochs. All images are scaled to 32×32 and the mini-batch size is set to 128. We set the hyperparameters α = 0.01, and K = 1. For the experiments with Office-Caltech10, all the modules comprising our model are trained with the SGD-momentum optimizer with the weight decay of 0.001 and the momentum factor of 0.9. The learning rate for the feature extractors and the label classifiers are 0.0001 and 0.001, respectively. We scale all the images to 224×224 and set the mini-batch size to 48. All the other hyperparameters are kept the same as in the experiments with Digits-Five. For Amazon Reviews, we train the models for 50 epochs using Adam optimizer with β_1 = 0.9, β_2 = 0.999, and the learning rate of 0.0001. We set α = 0.1, K = 2, and the mini-batch size to 100.

Performance evaluation

We evaluate the performance of MULTI-EPL against the competitors. We repeat experiments for each setting five times and report the mean and the standard deviation. The results are summarized in Tables 3–5. In the tables, SC and SB indicate *Source Combined* and *Single Best*, respectively. Note that MULTI-EPL provides the best accuracy in all the datasets, showing its superiority in both image datasets (Digits-Five and Office-Caltech10) and text datasets

Table 3. Classification accuracy on Digits-Five with and without domain adaptation.

Method	→T	→M	→S	→D	→U
LeNet5 (SC)	97.6±0.2	61.7±1.4	75.2±0.8	80.3±0.7	81.6±1.5
ResNet14 (SC)	98.2±0.3	63.5±0.8	79.1±1.6	92.9±0.5	94.5±0.3
LeNet5 (SB)	97.1±0.1	51.1±1.9	76.8±0.6	79.9±0.5	83.3±0.9
ResNet14 (SB)	97.1±1.0	49.5±1.3	81.4±0.7	91.8±0.5	91.5±2.7
DCTN	99.3±0.1	72.0±1.6	78.3±1.1	91.6±0.7	**98.4±0.2**
M³SDA	98.8±0.1	67.8±0.7	81.8±0.6	88.5±0.3	97.2±0.2
M³SDA-β	99.0±0.1	72.5±0.2	81.4±0.3	89.5±0.4	97.4±0.2
MDDA	98.8±0.4	78.6±0.6	79.3±0.8	89.7±0.7	93.9±0.5
LtC-MSDA	99.1±0.1	78.9±1.8	79.8±2.2	91.7±0.3	98.3±0.1
MULTI-EPL	**99.4±0.1**	**92.1±0.2**	**85.4±0.3**	**95.1±0.1**	98.2±0.1

The letters T, M, S, D, and U stand for MNIST, MNIST-M, SVHN, SynthDigits, and USPS, respectively. For MDDA, we report the performances in [18, 19].

Table 4. Classification accuracy on Office-Caltech10 with and without domain adaptation.

Method	→A	→C	→D	→W
ResNet50 (SC)	95.47±0.25	91.59±0.51	99.36±0.78	99.26±0.37
ResNet50 (SB)	95.03±0.48	89.05±0.88	99.87±0.28	98.24±0.61
DCTN	95.05±0.24	90.60±0.71	**100.0±0.00**	99.46±0.62
M³SDA	95.14±0.31	93.59±0.40	99.49±0.53	99.86±0.19
M³SDA-β	94.36±0.26	91.70±0.71	99.75±0.35	99.39±0.15
LtC-MSDA	95.68±0.84	92.34±0.61	**100.0±0.00**	99.86±0.19
MULTI-EPL	**96.23±0.13**	**93.52±0.49**	**100.0±0.00**	**99.93±0.16**

The letters A, C, D, and W stand for Amazon, Caltech, DSLR, and Webcam, respectively.

(Amazon Reviews). The enhancement is remarkable especially when MNIST-M is the target domain in Digits-Five, improving the accuracy by 13.20% compared to the state-of-the-art methods. It is also notable that MULTI-EPL consistently achieves successful adaptation of multiple domains, while other state-of-the-art methods sometimes fail to adapt and even deteriorate the performance. The failure appears to be attributable to negative transfer [38], but we leave this issue as a future work.

Table 5. Classification accuracy on Amazon Reviews with and without domain adaptation.

Method	→B	→D	→E	→K
MLP (SC)	79.76±0.70	82.18±0.59	84.42±0.27	87.23±0.51
MLP (SB)	79.00±0.92	80.38±0.61	84.76±0.45	87.46±0.36
DCTN	78.92±0.56	81.22±1.01	83.56±1.52	86.47±0.71
M³SDA	78.97±0.79	80.51±0.99	83.63±0.68	85.99±0.85
M³SDA-β	80.26±0.43	81.80±0.72	85.02±0.34	86.99±0.56
LtC-MSDA	76.73±0.79	78.03±1.92	79.51±2.09	81.49±2.38
MULTI-EPL	**81.00±0.53**	**83.42±0.31**	**86.53±0.44**	**88.64±0.53**

The letters B, D, E, and K stand for Books, DVDs, Electronics, and Kitchen appliances, respectively.
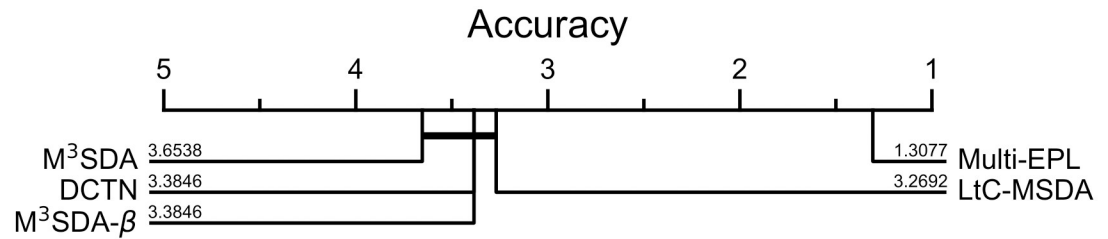
Fig 4. CD Diagram with various adaptation methods: DCTN, M³SDA, M³SDA-β, LtC-MSDA, and Multi-EPL.

We also illustrate the summary of the results in Fig 4 using CD (critical difference) diagram [39]. We tackled every single source and target scenario, and the five adaptation methods DCTN, M³SDA, M³SDA-β, LtC-MSDA, and MULTI-EPL. It demonstrates that MULTI-EPL gives significant performance enhancement compared to the existing methods.

Ablation study

We perform an ablation study on Digits-Five to identify what exactly enhances the performance of MULTI-EPL. We compare MULTI-EPL with three of its variants: MULTI-0, MULTI-PL, and MULTI-PL-DED. MULTI-0 aligns moments regardless of the labels of the data. MULTI-PL trains the model without ensemble learning. MULTI-PL-DED consists of four feature generators and four label classifiers, each of which is dedicated to each source domain.

The results are shown in Table 6. By comparing MULTI-0 with MULTI-PL, we observe that considering labels in moment matching plays a significant role in extracting domain-invariant features. The remarkable performance gap between MULTI-PL and MULTI-EPL verifies the effectiveness of ensemble learning. The overall accuracy of MULTI-PL-DED is much lower than that of MULTI-PL or MULTI-EPL; it demonstrates that the existing methods that assign individual networks for each source domain deteriorate not only the performance but also the model efficiency.

Effects of ensemble

We evaluate the performance on Digits-Five while varying the number n of pairs of feature extractor and label classifier. The results are summarized in Table 6. While an ensemble of two pairs gives much better performance than the model with a single pair, using more than two pairs does not bring remarkable improvement, except for the case of SVHN being the target dataset. We presume that the overfitting due to the excessive number of parameters has

Table 6. Experiments with MULTI-EPL and its variants.

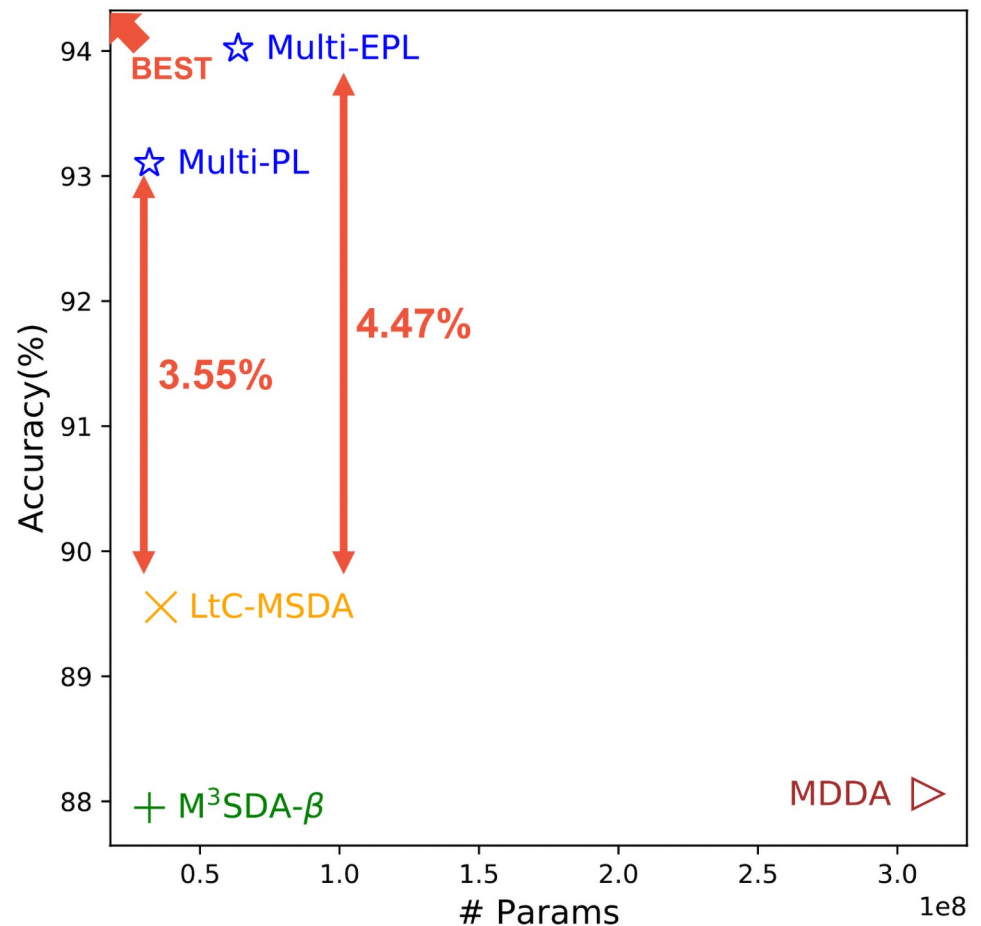| Method	→T	→M	→S	→D	→U
MULTI-0	98.8±0.1	67.8±0.7	81.8±0.6	88.5±0.3	97.2±0.2
MULTI-PL	99.3±0.1	90.2±0.5	83.7±0.4	94.4±0.2	98.0±0.2
MULTI-PL-DED	**99.4±0.1**	65.5±3.8	29.4±0.7	41.0±1.3	**98.6±0.2**
MULTI-EPL (n = 2)	**99.4±0.1**	**92.0±0.2**	85.4±0.3	95.1±0.1	98.2±0.1
MULTI-EPL (n = 3)	99.3±0.1	91.6±0.9	85.9±0.7	**95.2±0.2**	98.5±0.1
MULTI-EPL (n = 4)	99.3±0.1	91.5±1.2	**86.9±0.9**	95.1±0.1	98.5±0.1

Fig 5. The number of parameters and the model accuracy of the MSDA methods.

hindered the further improvement. We leave the task of figuring out proper regularization methods for the ensembles as a future work.

Parameter efficiency

We compare the number of parameters and performance of Multi-EPL with other state-of-the-art methods to demonstrate Multi-EPL's efficient usage of the model complexity. Fig 5 illustrates the number of model parameters and the average accuracy of each method that are evaluated with the Digits-Five dataset. Multi-PL is the variation of Multi-EPL that does not exploit the ensemble technique. Comparing Multi-PL and LtC-MSDA, the superiority of the proposed method is proved under the fair model complexity. On the other hand, the significant performance enhancement that the ensemble learning technique has made in Multi-EPL demonstrates that Multi-EPL greatly benefits from the additional model parameters, while MDDA has made little performance improvement even though it requires much more model parameters.

Visualization

We visualize the features from distinct adaptation methods using T-SNE [40] to verify the effect of label-wise moment matching. Fig 6 shows the feature distributions when no
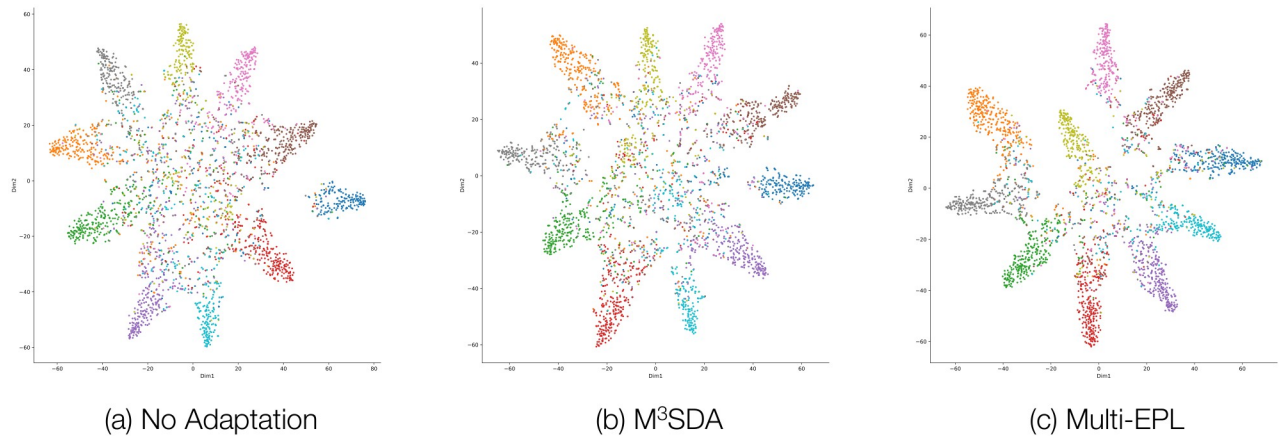
(a) No Adaptation   (b) M³SDA   (c) Multi-EPL

Fig 6. T-SNE visualization of the features from different adaptation methods.

adaptation method, M³SDA, and MULTI-EPL are applied, respectively. All the experiments are conducted with Digits-Five with MNIST-M as the target dataset. Each color in Fig 6 stands for a label.

Note that MULTI-EPL clearly separates features with different labels, while other do not; this explains the outstanding performance of MULTI-EPL.

Conclusion

We propose MULTI-EPL, a novel framework for the multi-source domain adaptation problem. MULTI-EPL overcomes the problems in existing methods of not directly addressing conditional distributions of data $p(\mathbf{x}|y)$, not fully exploiting the knowledge of target data, and having redundancy in model networks. MULTI-EPL aligns data from multiple source domains and the target domain considering the data labels, and exploits pseudolabels for unlabeled target data. MULTI-EPL further enhances the performance by generating an ensemble of multiple feature extractors. Our framework exhibits superior performance on both image and text classification tasks. Considering labels in moment matching and adding ensemble learning are shown to bring remarkable performance enhancement through ablation study. Future works include extending our approach to other tasks such as regression, which may require modification in the pseudolabeling method.

Author Contributions

Conceptualization: Seongmin Lee.

Data curation: Seongmin Lee.

Formal analysis: Seongmin Lee, U. Kang.

Funding acquisition: U. Kang.

Investigation: Seongmin Lee, Hyunsik Jeon.

Methodology: Seongmin Lee.

Project administration: U. Kang.

Resources: Seongmin Lee, U. Kang.

Software: Seongmin Lee.

Supervision: U. Kang.

Validation: Seongmin Lee.

Visualization: Seongmin Lee.

Writing – original draft: Seongmin Lee.

Writing – review & editing: Seongmin Lee, Hyunsik Jeon, U. Kang.

References

1. Torralba A, Efros AA. Unbiased look at dataset bias. In: CVPR; 2011.

2. Long M, Cao Y, Wang J, Jordan MI. Learning Transferable Features with Deep Adaptation Networks. In: ICML; 2015.

3. Long M, Zhu H, Wang J, Jordan MI. Unsupervised Domain Adaptation with Residual Transfer Networks. In: NIPS; 2016.

4. Long M, Zhu H, Wang J, Jordan MI. Deep Transfer Learning with Joint Adaptation Networks. In: ICML; 2017.

5. Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, et al. Domain-Adversarial Training of Neural Networks. JMLR. 2016;.

6. Tzeng E, Hoffman J, Saenko K, Darrell T. Adversarial Discriminative Domain Adaptation. In: CVPR; 2017.

7. Liu M, Breuel T, Kautz J. Unsupervised Image-to-Image Translation Networks. In: NIPS; 2017.

8. Zhu J, Park T, Isola P, Efros AA. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In: ICCV; 2017.

9. Hoffman J, Tzeng E, Park T, Zhu J, Isola P, Saenko K, et al. CyCADA: Cycle-Consistent Adversarial Domain Adaptation. In: ICML; 2018.

10. Ben-David S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW. A theory of learning from different domains. Mach Learn. 2010; 79(1-2):151–175. https://doi.org/10.1007/s10994-009-5152-4

11. Mansour Y, Mohri M, Rostamizadeh A. Domain Adaptation with Multiple Sources. In: NIPS; 2008.

12. Crammer K, Kearns MJ, Wortman J. Learning from Multiple Sources. JMLR. 2008;.

13. Hoffman J, Mohri M, Zhang N. Algorithms and Theory for Multiple-Source Adaptation. In: NIPS; 2018.

14. Zhao H, Zhang S, Wu G, Moura JMF, Costeira JP, Gordon GJ. Adversarial Multiple Source Domain Adaptation. In: NIPS; 2018.

15. Zellinger W, Moser BA, Saminger-Platz S. Learning Bounds for Moment-Based Domain Adaptation. CoRR. 2020;abs/2002.08260.

16. Xu R, Chen Z, Zuo W, Yan J, Lin L. Deep Cocktail Network: Multi-Source Unsupervised Domain Adaptation With Category Shift. In: CVPR; 2018.

17. Peng X, Bai Q, Xia X, Huang Z, Saenko K, Wang B. Moment Matching for Multi-Source Domain Adaptation. In: ICCV; 2019.

18. Zhao S, Wang G, Zhang S, Gu Y, Li Y, Song Z, et al. Multi-Source Distilling Domain Adaptation. In: AAAI; 2020.

19. Wang H, Xu M, Ni B, Zhang W. Learning to Combine: Knowledge Aggregation for Multi-source Domain Adaptation. In: ECCV; 2020.

20. Jeon H, Lee S, Kang U. Unsupervised Multi-Source Domain Adaptation with No Observable Source Data. PLOS ONE. 2021;. https://doi.org/10.1371/journal.pone.0253415 PMID: 34242258

21. Tzeng E, Hoffman J, Zhang N, Saenko K, Darrell T. Deep Domain Confusion: Maximizing for Domain Invariance. CoRR. 2014;abs/1412.3474.

22. Ghifary M, Kleijn WB, Zhang M, Balduzzi D, Li W. Deep Reconstruction-Classification Networks for Unsupervised Domain Adaptation. In: ECCV; 2016.

23. Zhuang F, Cheng X, Luo P, Pan SJ, He Q. Supervised Representation Learning: Transfer Learning with Deep Autoencoders. In: IJCAI; 2015.

24. Zhang K, Gong M, Schölkopf B. Multi-Source Domain Adaptation: A Causal View. In: AAAI; 2015.

25. Wang H, Yang W, Lin Z, Yu Y. TMDA: Task-Specific Multi-source Domain Adaptation via Clustering Embedded Adversarial Training. In: ICDM; 2019.

26. Lin C, Zhao S, Meng L, Chua T. Multi-Source Domain Adaptation for Visual Sentiment Classification. In: AAAI; 2020.

27. Pei Z, Cao Z, Long M, Wang J. Multi-Adversarial Domain Adaptation. In: AAAI; 2018.

28. Sun B, Feng J, Saenko K. Return of Frustratingly Easy Domain Adaptation. In: AAAI; 2016.

29. Zellinger W, Grubinger T, Lughofer E, Natschläger T, Saminger-Platz S. Central Moment Discrepancy (CMD) for Domain-Invariant Representation Learning. CoRR. 2017;abs/1702.08811.

30. Hoffman J, Rodner E, Donahue J, Saenko K, Darrell T. Efficient Learning of Domain-invariant Image Representations. In: ICLR; 2013.

31. Chen M, Xu Z, Weinberger K, Sha F. Marginalized Denoising Autoencoders for Domain Adaptation; 2012.

32. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proceedings of the IEEE. 1998; 86(11):2278–2324. https://doi.org/10.1109/5.726791

33. Ganin Y, Lempitsky VS. Unsupervised Domain Adaptation by Backpropagation. In: ICML; 2015.

34. Netzer Y, Wang T, Coates A, Bissacco A, Wu B, Ng AY. Reading digits in natural images with unsupervised feature learning. 2011;.

35. Hastie T, Friedman JH, Tibshirani R. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics. Springer; 2001.

36. He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In: CVPR; 2016.

37. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. In: ICLR; 2015.

38. Pan SJ, Yang Q. A Survey on Transfer Learning. TKDE. 2010;.

39. Demsar J. Statistical Comparisons of Classifiers over Multiple Data Sets. JMLR. 2006; p. 1–30.

40. van der Maaten L, Hinton G. Visualizing Data using t-SNE. JMLR. 2008;.