Neural Computing and Applications
https://doi.org/10.1007/s00521-018-3846-x

ORIGINAL ARTICLE

Unsupervised domain adaptation with target reconstruction and label confusion in the common subspace

Boyuan Jiang[1] · Chao Chen[1] · Xinyu Jin[1]

Received: 4 June 2018 / Accepted: 26 October 2018

Abstract
Deep neural networks can learn powerful and discriminative representations from a large number of labeled samples. However, it is typically costly to collect and annotate large-scale datasets, which limits the applications of deep learning in many real-world scenarios. Domain adaptation, as an option to compensate for the lack of labeled data, has attracted much attention in the community of machine learning. Although a mass of methods for domain adaptation has been presented, many of them simply focus on matching the distribution of the source and target feature representations, which may fail to encode useful information about the target domain. In order to learn invariant and discriminative representations for both domains, we propose a Cross-Domain Minimization with Deep Autoencoder method for unsupervised domain adaptation, which simultaneously learns label prediction on the source domain and input reconstruction on the target domain using shared feature representations aligned with correlation alignment in a unified framework. Furthermore, inspired by adversarial training and cluster assumption, a task-specific class label discriminator is incorporated to confuse the predicted target class labels with samples draw from categorical distribution, which can be regarded as entropy minimization regularization. Extensive empirical results demonstrate the superiority of our approach over the state-of-the-art unsupervised adaptation methods on both visual and non-visual cross-domain adaptation tasks.

Keywords Domain adaptation · Autoencoder · Adversarial training · Cluster assumption

1 Introduction

The development of deep neural networks has already achieved significant success in many machine learning tasks including object recognition [24, 31], semantic segmentation [3, 37], object detection [19, 36], image style transfer [17] and video advertising [62, 63]. However, one major problem of deep neural networks is that although they perform well on the testing data sampled from the same distribution as the training data, they may find it difficult to generalize to data sampled from different distributions and make correct predictions. This phenomenon is known as *dataset bias* or *domain shift* [23]: the model trained on one large dataset cannot generalize well to a novel dataset or task. In reality, labeled data may be rare or costly to obtain in some cases, e.g., annotated biomedical dataset in the area of medicine [28, 47]. Therefore, a straightforward but practical question is: Can the knowledge from a different but related source domain with plenty of labeled data be leveraged to help improve the model performance of the target domain where scarce labeled data exist? To address this issue, *domain adaptation* (DA), which aims to mitigate the harmful effect of domain shift with sufficient source domain data and limited target domain data, has emerged as a new framework to solve this problem and received great interest in the machine learning community. In this work, we consider a more extreme setting where the target domain is totally unlabeled. In the literature, this setting is seen as *unsupervised domain adaptation* [45], which can be regarded as an extension of the semi-supervised learning [64] where

✉ Xinyu Jin
jinxy@zju.edu.cn

Boyuan Jiang
byjiang@zju.edu.cn

Chao Chen
chench@zju.edu.cn

[1] Institution of Information Science and Electrical Engineering, Zhejiang University, Hangzhou 310037, Zhejiang, China

Published online: 15 November 2018

labeled samples and unlabeled samples are drawn from the same distribution.

To solve domain adaptation tasks, a series of approaches have been put forward and can be roughly classified into three categories: feature matching-based domain adaptation [14], instance reweighting-based domain adaptation [13] and parameter-based domain adaptation [27]. Among them, feature matching-based approaches are the most widely used domain adaptation approaches. The aim behind these methods is to learn domain-invariant and discriminative representations that can simultaneously reduce domain difference and retain as much information of the original domain as possible. Under the new feature space, a classifier is then trained in a supervised manner via labeled source domain data [57].

Recently, deep learning-based domain adaptation methods have attracted much more attention to the highly scalable and nonlinear mapping ability. Most of the existing methods consider convolutional neural networks (for images) or Recurrent Neural Network (for natural language or time series) as a feature extractor together with discrepancy metric, such as maximum mean discrepancy (MMD) [22], Kullback Leibler divergence (KL) [50] or Hellinger Distance [4], as a regularization part of the joint loss function [40] or an auxiliary domain discriminator [54] to reduce the divergence between the distribution of source and target features. However, these methods only focus on reducing the distance between the source and target features but do not make use of the target data effectively and thus may fail to extract discriminative representations of the target domain for the subsequent classification task.

As a typical and effective unsupervised learning algorithm, autoencoder can extract discriminative representations by copying its input to its output [34]. In order to address the aforesaid limitation, several algorithms which are based on autoencoder and feature discrepancy minimization to learn invariant and discriminative feature representations have been presented [7, 18, 29, 58, 65]. Despite achieving appealing results, these methods still suffer from one limitation. They only focus on transferring the source and target domain samples into a new domain-invariant space meanwhile maintain as much of the remaining information of the target domain as possible, followed by a traditional supervised learning algorithm applying on the transferred source domain samples. That is to say, no target domain information is considered during the label encoder training.

To address the aforementioned limitations in unsupervised domain adaptation tasks, we introduce a Cross-Domain Minimization with Deep Autoencoder (CDMDA) framework in this paper. An overall insight of our proposal is shown in Fig. 1. In CDMDA, there are two encoding layers: a feature encoding layer and a task-specific class label encoding layer, and a decoding layer. More specifically, in the feature encoding layer, CORrelation ALignment (CORAL) [52] is incorporated to minimize distribution discrepancy of the source and target features. In the label encoding layer, we introduce a label discriminator to force the distribution of predicted target labels to be indistinguishable from the categorical distribution following the line of cluster assumption [11], i.e., decision boundaries of a classifier should not cross high-density data regions. In the decoding layer, target inputs are reconstructed via latent feature representations, which encourages the feature encoding layer to capture more information about the structure of target data. See Fig. 2 for high level summary.

In summary, the main contributions of this paper are described below.

1. We propose a new autoencoder-based unsupervised domain adaptation algorithm CDMDA. Different from most existing methods that only focus on minimizing the distance of feature representations between two different domains, we further incorporate a target domain data reconstruction process based on the

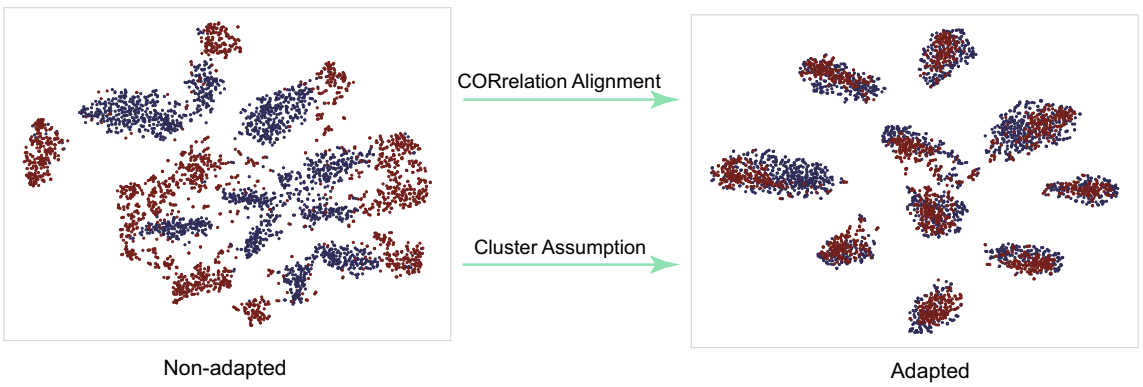

Fig. 1 Illustration of our proposal. In order to maximize the classification accuracy of the unlabeled target domain, we minimize the discrepancy between the source (blue) and target (red) domains and maximize the distance between the classifier hyperplane and target data points (color figure online)

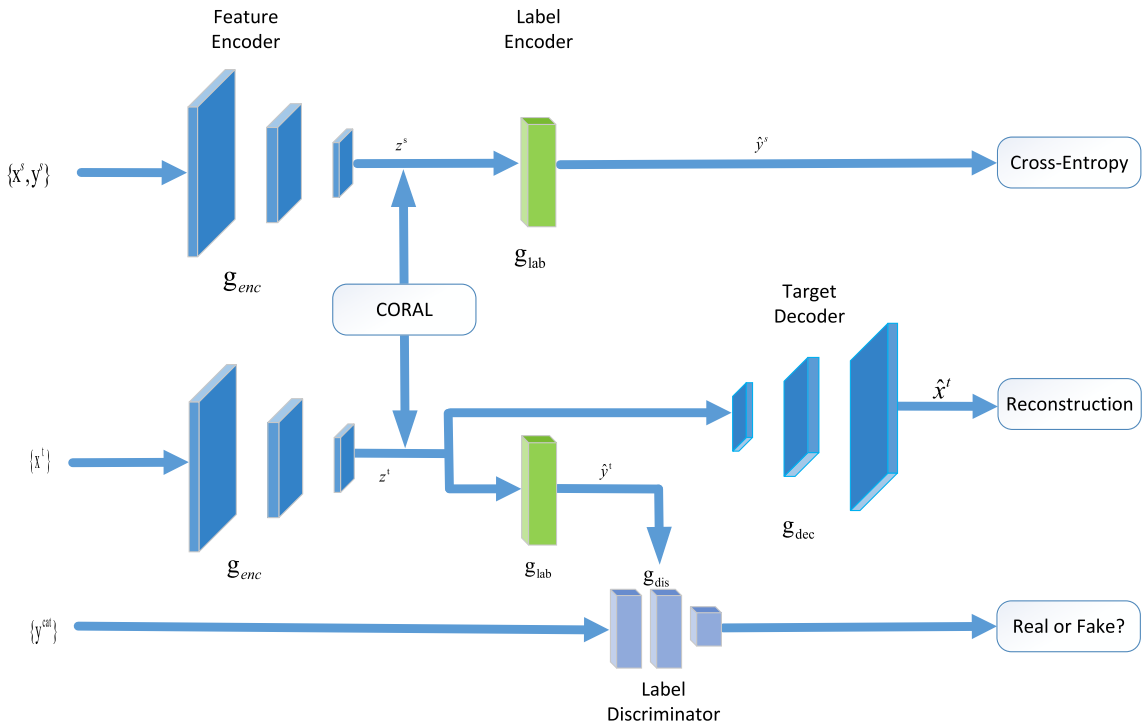

Fig. 2 An overall architecture of the CDMDA model, where the source domain feature encoder and label encoder share same weights with target domain's. There are three novelties compared to previous methods: (1) we use CORAL as a regularizer to reduce the distribution discrepancy between the source feature representations z^s and the target feature representations z^t, (2) a label discriminator is introduced to force the distribution of the label encoder output $\hat{y}_t$ to be indistinguishable from the categorical distribution y^{cat}, and (3) we reconstruct the target input via z^t to capture the structure of the target domain. See text for more information

autoencoder to extract the structure of the target domain for the subsequent classification task.

2. We are among the first to introduce a label discriminator to the target label encoding layer to force the distribution of predicted target labels to become practically indistinguishable from the categorical distribution in the field of unsupervised domain adaptation, which is deeply intertwined with the entropy minimization algorithm [21].

The rest of the paper is organized as follows. In Sect. 2, we review some most related work. In Sect. 3, some preliminary knowledge related to our method is introduced. In Sect. 4, we present the CDMDA method from four training phase and its learning algorithm. In Sect. 5, we report a broad range of empirical results compared with other competing methods and demonstrate the advantage of incorporating the label discriminator empirically. Finally, in Sect. 6 we draw conclusions and discuss the future work.

2 Related work

There has already been extensive prior work on transfer learning [13, 14, 27, 32, 57, 59]. However, most of these methods are designed for small datasets such as *Office* dataset [49] and work on low dimensions surf features [5] rather than high dimensions raw image pixels. With the aforementioned limitations, these shallow methods are not practical enough for real-world applications. Recent work has paid more attention to deep learning-based methods [7, 9, 15, 16, 18, 29, 39, 40, 46, 51, 53, 54, 58, 60, 65] due to their empirical superiority on this problem and scalability to large datasets.

In the case of unsupervised deep domain adaptation (the focus of this paper), one major line of work follows parallel CNN architectures such as Siamese Networks [8] whose weights of corresponding layers are shared between source domain and target domain, and then trains with a combination loss of a cross-entropy (applied to source domain) and a discrepancy loss or an adversarial loss (applied to source and target domains). One of the first such work is deep domain confusion (DDC) [53] where a linear MMD metric is applied to the last fully connected layer (before the output layer) as an additional loss function. Long et al. proposed deep adaptation network (DAN) [39], which considers the sum of MMDs over several corresponding layers and explores multiple kernels for adapting deep representations. Apart from minimizing the discrepancy loss, adversarial learning is another class of methods to encourage domain confusion. Ganin et al. [15, 16]

proposed domain adversarial neural networks (DANN) which contain two classifiers: the first one is trained to predict the task-specific class labels correctly, and the second one is trained to predict the domain of the input. The training strategy has the analogy of a minimax game: the feature extractor tries to obtain domain-invariant features to maximize the loss of the domain classifier, while the domain classifier tries to discriminate which domain the input is to minimize the loss of the domain classifier. The minimax optimization becomes possible by integrating a gradient reversal layer (GRL), which is left unchanged during the forward propagation and reverses its gradient during backpropagation. Similarly, Tzeng et al. [54] proposed the adversarial discriminative domain adaptation (ADDA), which considers inverted label GAN loss [20] instead of GRL loss, to learn discriminative and domain-invariant features.

Another line of work considers an autoencoder-based architecture to better model both the label and the structure of the data [7, 12, 18, 29, 58, 65]. These approaches are in the same spirit as multitask learning [10] that the main task can be beneficial from learning an auxiliary task. Marginalized stacked denoising autoencoder (mSDA) [12] is among the first to adopt the greedy layer-by-layer training of stacked denoising autoencoder (SDAs) to learn new representations for domain adaptation. Ghifary et al. [18] proposed deep reconstruction classification networks (DRCN) which combine standard convolutional neural networks for source data prediction with deconvolutional neural networks for target data reconstruction. Bousmalis et al. [7] proposed domain separation networks (DSN) which integrate data reconstruction and discrepancy minimization together to simultaneously learn a private feature space and a common feature space. Our work is in a similar spirit to autoencoder-based methods but adopts the CORAL to measure the similarity of the source and target representations, which is different from the previous work based on MMD or domain adversarial loss.

Cluster assumption is another critical component of our work, which states that the decision boundary should not cross high-density regions, but instead lie in low-density regions [11]. This assumption is the key to semi-supervised learning, leading to many successful learning algorithms such as entropy regularization [21] and Pseudo-Label [35]. Recently, some unsupervised domain adaptation approaches [9, 40, 46] have borrowed the idea from the cluster assumption, which leverages entropy regularization as a proxy. In this work, from another viewpoint of cluster assumption, we consider incorporate a task-specific class label discriminator to force the distribution of predicted target labels to be indistinguishable from the categorical distribution. Apparently, label confusion and entropy minimization may seem to be two unrelated approaches for domain adaptation. However, this is not the case, and indeed, these two types of approaches are deeply intertwined. When the distribution of predicted target labels is indistinguishable from the categorical distribution, the entropy of the target domain reaches the minimum.

Table 1 The notation and denotation

Notation	Denotation
$\mathcal{D}_S, \mathcal{D}_T$	The source and target domains
x^s, x^t	Samples of the source and target domains
y^s	The label of the sample x^s
y^{cat}	The sample drawn from a categorical distribution
n_s, n_t	The batch size of the source and target domain samples
Z_S, Z_T	The image features of the source and target domains
g_{enc}	The feature encoding function
g_{dec}	The target reconstruction function
g_{lab}	The label encoding function
g_{dis}	The discriminator function of the label regularization phase
g_{gen}	The generator function of the label regularization phase
f	The prediction logit output function

3 Preliminary knowledge

In this section, we will first review some most related preliminary knowledge that is used in our proposed approach and then briefly discuss the unsupervised domain adaptation problem. A theoretical analysis of the expected target error bound for domain adaptation will be introduced at last. Note that all frequently used notations are listed in Table 1.

3.1 Autoencoder

An autoencoder is a neural network used to learn efficient latent codings in an unsupervised manner [48]. Internally, it has an encoder g_{enc} and a decoder g_{dec} which are both multilayer neural networks. Given an input $\boldsymbol{x}$, the encoder first maps it to latent feature codings $\boldsymbol{z}$, and then the decoder attempts to reconstruct its input from $\boldsymbol{z}$. The process of encoding and decoding of the basic autoencoder can be summarized as:

$$z = g_{enc}(x; \theta) \tag{1}$$

$$\hat{x} = g_{dec}(z; \theta') \tag{2}$$

where g_{enc} and g_{dec} are multilayer neural networks, θ and θ' are corresponding parameters that can be optimized by minimizing the following mean square reconstruction error:

$$\min_{\theta,\theta'} \|x - \hat{x}\|^2 = \min_{\theta,\theta'} \|x - g_{dec}(g_{enc}(x))\|^2 \tag{3}$$

Along the similar line, denoising autoencoder [56] reconstructs original input data from partial corruption input to make the representations more robust.

3.2 Correlation alignment

CORrelation ALignment (CORAL) is a relatively easy unsupervised domain adaptation method that minimizes domain shift by aligning the second-order statistics of source and target distributions [52]. A linear transformation A can be applied to the source feature space and use the Frobenius norm as the matrix distance metric:

$$\min_A \|C_{\hat{S}} - C_T\|_F^2 = \min_A \|A^T C_S A - C_T\|_F^2 \tag{4}$$

where C_S and C_T are the covariance matrices of the source and target features and $\|\cdot\|_F^2$ denotes the matrix Frobenius norm. The optimal transformation A^* can be obtained as follows:

$$A^* = \sqrt{\frac{C_T + \alpha \mathbf{I}}{C_S + \alpha \mathbf{I}}} \tag{5}$$

where $\alpha\mathbf{I}$ is the regularization term which guarantees the full rank of the covariance matrix, α is typically set to 1 as recommended in Sun et al. [52], and $\mathbf{I}$ is the identity matrix.

3.3 Definitions and problem statement

Definition 1 *Domain.* A *domain* $\mathcal{D}$ consists of two components: a feature space $\mathcal{X} \in \mathbb{R}^d$ and a marginal distribution $\mathcal{P}(X)$, i.e., $\mathcal{D} = \{\mathcal{X}, \mathcal{P}(X)\}$, where $X \in \mathcal{X}$.

Definition 2 *Task.* Given a specific domain $\mathcal{D}$, a task $\mathcal{T}$ is composed of a label space $\mathcal{Y}$ and an objective predictive function $f(\cdot)$, i.e., $\mathcal{T} = \{\mathcal{Y}, f(X)\}$, where $f(X)$ can be interpreted as the conditional probability distribution $P(Y|X)$ from a probabilistic viewpoint.

Definition 3 *Unsupervised domain adaptation.* Given a source domain $\mathcal{D}_S$ and learning task $\mathcal{T}_S$ with all samples labeled, a target domain $\mathcal{D}_T$ and learning task $\mathcal{T}_T$ with all samples unlabeled, *unsupervised domain adaptation* aims to improve the performance of the target predictive function $f_T(\cdot)$ in D_T with the help of knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$. Typically, the distribution of the source domain $\mathcal{D}_S$ is different from the target domain $\mathcal{D}_T$ but is related.

3.4 Theoretical analysis of the expected target error bound

Theoretical studies have been developed for the bound of the target domain generalization performance of a classifier trained in the source domain. As analyzed in Ben-David et al. [6], the bound of the expected target error can be estimated as:

$$\epsilon_t(h) \leq \epsilon_s(h) + \frac{1}{2} d_{\mathcal{L}\Delta\mathcal{L}(X_s, X_t)} + C \tag{6}$$

h is the learned hypothesis which can be interpreted as the predictor function. $\epsilon_t(h)$ and $\epsilon_s(h)$ are the prediction error of the target domain and source domain, respectively. $d_{\mathcal{L}\Delta\mathcal{L}}$ denotes the $\mathcal{L}$-divergence between the source and target domains, which can be minimized by reducing the distribution discrepancy. $C = \arg\min_{h' \in \mathcal{L}} \epsilon_s(h') + \epsilon_t(h')$ is the generalization error of the ideal joint hypothesis h', which can be disregarded because it is considered to be negligibly small [39]. Therefore, a practical domain adaptation algorithm should simultaneously minimize the first two right-hand terms of Eq. (6), i.e., $\epsilon_s(h)$ and $\frac{1}{2} d_{\mathcal{L}\Delta\mathcal{L}(X_s, X_t)}$. We will show how our model jointly minimizes these two terms as follows.

4 Methodology

4.1 Main idea

To address the unsupervised domain adaptation scenario where there is no label information about the target domain data, we propose CDMDA which jointly learns a feature extractor and a label predictor through four training phases. The architecture of CDMDA is illustrated in Fig. 2, which consists of a feature encoder, a label encoder, a target decoder and a label discriminator. More specifically, let g_{enc} denote the feature encoder that maps input x to feature representations z which should be aligned between domains, g_{lab} denote the label encoder that maps z to task-specific predictions $\hat{y}$, g_{dec} denote the target decoder that takes z as input to reconstruct the origin input $\hat{x}$ of the target domain, and g_{dis} denote the label discriminator that discriminates whether its input is the task-specific predictions $\hat{y}$ or sampled from a categorical distribution. Note that all four components are deep neural networks and can be optimized through backpropagation with stochastic gradient descent (SGD) in a uniform framework.

4.2 Connections to existing work

Compared with the previous work [7, 18, 65], which are also based on autoencoders to learn a common subspace via capturing representations shared by both domains, a major difference between ours and previous works is that, inspired by Adversarial Autoencoders [42], we incorporate a task-specific class label discriminator to regularize the process of the target label prediction, which can be regarded as minimizing the target domain entropy and pushing the decision boundaries of classifier away from data-dense regions [11, 21]. In the rest of this section, we will describe CDMDA from four training phases together with their

corresponding loss functions and give the learning algorithm in detail.

4.3 Model learning

4.3.1 Feature divergence regularization phase

One of the key points of CDMDA as well as many other DA methods is to learn a shared feature space that the source features are aligned with the target features. Unlike many existing methods that incorporate maximum mean discrepancy (MMD) [22] as the feature divergence regularizer, we employ correlation alignment to calculate the distance of the second-order statistical information between feature representations z^s and z^t of two domains. Since we adopt deep neural networks instead of linear transformation A to get feature space, we can rewrite Eq. (4) and minimize the following $\mathcal{L}_{coral}$ term, which is similar to [51]:

$$\mathcal{L}_{coral} = \frac{1}{4d^2}\|C_S - C_T\|_F^2 \tag{7}$$

$$C_S = \frac{1}{n_s - 1}\left(Z_S^\top Z_S - \frac{1}{n_s}\left(\boldsymbol{I}^\top Z_S\right)^\top \left(\boldsymbol{I}^\top Z_S\right)\right) \tag{8}$$

$$C_T = \frac{1}{n_t - 1}\left(Z_T^\top Z_T - \frac{1}{n_t}\left(\boldsymbol{I}^\top Z_T\right)^\top \left(\boldsymbol{I}^\top Z_T\right)\right) \tag{9}$$

where $\boldsymbol{I}$ is a column vector with all elements equal to 1, d is the dimension of the feature representations, $\|\cdot\|_F^2$ denotes the squared matrix Frobenius norm, and $Z_S = [z_1^s, \ldots, z_{n_s}^s]$ and $Z_T = [z_1^t, \ldots, z_{n_t}^t]$ denote batches of the feature representations from the last layer of feature encoder. C_S and C_T are the second-order statistics of the source and target feature distributions.

According to chain rule and incorporating Eqs. (8) and (9) into Eq. (7), the gradient of Eq. (7) w.r.t. its input can be derived as follows:

$$\frac{\partial \mathcal{L}_{coral}}{\partial Z_S} = \frac{\partial \mathcal{L}_{coral}}{\partial C_S}\frac{\partial C_S}{\partial Z_S} = \frac{1}{d^2(n_s - 1)}\left(Z_S - \frac{1}{n_s}\mathbf{1}(\mathbf{1}^T Z_S)\right)(C_S - C_T) \tag{10}$$

$$\frac{\partial \mathcal{L}_{coral}}{\partial Z_T} = \frac{\partial \mathcal{L}_{coral}}{\partial C_T}\frac{\partial C_T}{\partial Z_T} = -\frac{1}{d^2(n_t - 1)}\left(Z_T - \frac{1}{n_t}\mathbf{1}(\mathbf{1}^T Z_T)\right)(C_S - C_T) \tag{11}$$

4.3.2 Supervised learning phase

In this phase, we build a classifier $f(x) = (g_{enc} \circ g_{lab})(x)$ via minimizing the following cross-entropy on the labeled source domain, where g_{enc} is a series of deep neural networks that extracts features from original inputs and g_{lab} is a fully connected layer that maps features to the task-specific class labels in our work:

$$\mathcal{L}_{class} = -\frac{1}{n_s}\sum_{i=1}^{n_s}\sum_{k=1}^{c} 1\{y_i^s = k\}\log f_k(x_i^s) \tag{12}$$

where c is the number of classes, $f_k(x_i^s)$ denotes the probability of ith source sample belonging to kth class. $1\{y_i^s = k\}$ equals to 1 when the ith source sample belongs to the kth class and otherwise 0.

4.3.3 Reconstruction learning phase

In this phase, we simultaneously learn the feature encoder g_{enc} and decoder g_{dec} by minimizing reconstruction error of the target domain, which can be regarded as learning discriminative feature representations that approximate the structure of the target data. We use the following mean square loss to measure the reconstruction error:

$$\mathcal{L}_{recon} = \frac{1}{2n_t}\sum_{i=1}^{n_t}\|x_i^t - \hat{x}_i^t\|^2 = \frac{1}{2n_t}\sum_{i=1}^{n_t}\|x_i^t - g_{dec}\left(g_{enc}\left(x_i^t\right)\right)\|^2 \tag{13}$$

where $\hat{\mathbf{x}}^t$ is the reconstruction of the $\mathbf{x}^\mathbf{t}$.

4.3.4 Label regularization phase

As we have mentioned in Sect. 2, cluster assumption is a critical component of our model. Since there is no label information of the target domain, a label discriminator is incorporated to regularize the target label encoder, which is inspired by the Adversarial Autoencoders [42]. In this phase, the classifier $f(x)$ and label discriminator g_{dis} work in a way that mimics the generative adversarial networks [20], where the label discriminator tries to tell apart true samples (draw from a categorical distribution) and fake samples (output of the label encoder), while the classifier tries to confuse it. Similar to the GANs, we can define the discriminator loss and generator loss as follows:

$$\mathcal{L}_{dis} = -\frac{1}{n_t}\sum_{i=1}^{n_t}\left[\log g_{dis}\left(y_i^{cat}\right) + \log\left(1 - g_{dis}\left(f(x_i^t)\right)\right)\right] \tag{14}$$

$$\mathcal{L}_{gen} = -\frac{1}{n_t}\sum_{i=1}^{n_t}\log g_{dis}\left(f\left(x_i^t\right)\right) \tag{15}$$

$\mathbf{y}^{cat} \sim Cat(\mathbf{y})$ denotes samples that are drawn from the categorical distribution, which can be viewed as a "1-of-K" vector (a vector with one element containing a 1 and all other elements containing a 0).

4.3.5 Learning algorithm

Since weights of the feature encoder and label encoder are shared between the source and target domains, we can

combine Eqs. (7), (12), (13) and (15) into one joint loss function and minimize it through SGD at one time:

$$\mathcal{L}_{joint} = \mathcal{L}_{class} + \lambda_1 \mathcal{L}_{coral} + \lambda_2 \mathcal{L}_{recon} + \lambda_3 \mathcal{L}_{gen} \tag{16}$$

where λ_1, λ_2 and λ_3 are three trade-off parameters used to balance the contributions of the four terms to the joint objective function. Note that our proposed CDMDA method well agrees with the theory of domain adaptation provided by Ben-David et al. [6]. The minimization of $\mathcal{L}_{class}$ and $\mathcal{L}_{coral}$ is related to the first two terms of Eq. (6), respectively. The detailed learning algorithm of the CDMDA is summarized in Algorithm 1, and the stopping criterion is determined by monitoring the accuracy of the validation set.

Algorithm 1: Learning Algorithm of the CDMDA Method

Input : labeled source domain $\mathcal{D}_s = (\mathbf{X}_s, \mathbf{Y}_s)$, unlabeled target domain $\mathcal{D}_t = (\mathbf{X}_t)$, trade-off parameters $\lambda_1, \lambda_2, \lambda_3$, learning rate η

1 Initialize parameters of the feature encoder, the target decoder, the label encoder and the label discriminator;
2 **while** *not stop* **do**
3 Sample a batch of source domain data with size n_s and a batch of target domain data with size n_t;
4 Do a forward pass according to Eq. 16 to calculate the joint loss;
5 Do a backward pass to calculate the gradient of Eq. 16 and update the parameters of the feature encoder, the target decoder, the label encoder and the label discriminator;
6 Do a forward pass according to Eq. 14 to calculate the discriminator loss;
7 Do a backward pass to calculate the gradient of Eq. 14 and update the parameters of the label discriminator
8 **end**

Output: optimal parameters of the feature encoder, the target decoder, the label encoder and the label discriminator

4.4 Relation to $\mathcal{L}\Delta\mathcal{L}$-divergence

As we have mentioned in Sect. 3.4, a practical domain adaptation algorithm should minimize both the source domain prediction error and the $\mathcal{L}\Delta\mathcal{L}$-discrepancy distance between two distributions $\mathcal{S}$ and $\mathcal{T}$. In this section, we give a brief analysis of our method regarding $\mathcal{L}\Delta\mathcal{L}$-divergence [6]. Given a hypothesis $h \in \mathcal{L}$, $d_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T})$ can be bounded by the empirical estimate

$$\begin{aligned} d_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T}) &= 2 \sup_{\eta \in \mathcal{L}} |Pr[\eta(X_S) = 1] - Pr[\eta(X_T) = 1]| \\ &\leq \hat{d}_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T}) + C_1 \\ &\leq 2\left(1 - \inf_{\eta \in \mathcal{L}} \left[\frac{1}{n_s} \sum_{i=1}^{n_s} \mathcal{L}\left[\eta(g_{enc}(x_i^s)) = 1\right] \right.\right. \\ &\quad \left.\left. + \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}\left[\eta(g_{enc}(x_i^t)) = -1\right]\right]\right) + C_1 \\ &= 2\left(1 - \inf_{\eta \in \mathcal{L}} err(\eta)\right) + C_1 \end{aligned} \tag{17}$$

$d_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T})$ shows that the empirical $\mathcal{L}$-divergence between two samples from distribution $\mathcal{S}$ and $\mathcal{T}$ converges uniformly to the true $\mathcal{L}$-divergence for hypothesis classes $\mathcal{L}$ of finite VC dimension. $\eta \in \mathcal{L}$ is a classifier which achieves minimum error on the binary classification problem of distinguishing between features extracted from the source domain and the target domain. $\mathcal{L}[\cdot]$ is the linear loss function of the Parzen window classifier where $\mathcal{L}[\eta = 1] = -1$ and $\mathcal{L}[\eta = -1] = 1$. C_1 is a constant related to the complexity of hypothesis space. The optimal domain discriminator η gives the upper bound for $d_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T})$. By iteratively decreasing the feature discrepancy between the source and target domains with the feature regularization phase, the classification error of η would be maximized so that $d_{\mathcal{L}\Delta\mathcal{L}}(\mathcal{S}, \mathcal{T})$ would be reduced.

5 Experiments

In this section, we evaluate the performance of CDMDA by comparing with several state-of-the-art unsupervised domain adaptation methods related to ours on four very different visual digits datasets: MNIST [33], USPS [26], SVHN [44] and SYN DIGITS dataset [15], which consist of 10 common classes of digits. Example images can be seen in Fig. 3. For non-visual domain adaptation, we evaluate on multilingual text categorization dataset [2, 55].

5.1 Baselines

The following baselines are evaluated in the experiments of this section. The *source only* model is trained without any target domain data. The *target only* model is trained with sufficient labeled target domain data in a supervised

Fig. 3 Examples of the four visual datasets used in our experiments. From left to right: MNIST, USPS, SVHN and SYN DIGITS

manner, which can be regarded as an upper bound of all domain adaptation methods. In addition, we further consider some recently proposed unsupervised domain adaptation methods related to ours as a comparison:

- *ADDA* The adversarial discriminative domain adaptation [54] combines discriminative modeling, untied weight sharing, and a GAN loss to adversarially learn an indistinguishable feature space.
- *DANN* The domain adversarial neural networks [15, 16] employ the gradient reversal layer to learn discriminative and invariant domain features with adversarial training process.
- *DRCN* The Deep Reconstruction Classification Network [18] which jointly learns a shared encoding representation for supervised classification and unsupervised reconstruction.
- **Deep CORAL** The Deep CORAL [51] extends CORAL to learn a nonlinear transformation that aligns correlations of layer activations in deep neural networks.

5.2 Implement details

For the visual digits domain adaptation experiments, we employ two convolutional layers with 64 and 128 filters following two fully connected layers with 1024 and 128 units as the feature encoder. For the target decoder, two transposed convolutional (deconvolution) [61] layers are used to upsample and reconstruct the original input image. For the label discriminator, there are three fully connected layers with 100, 100 and 1 units. All layers are followed by a ReLU [43] activation function apart from output layers. The details can be seen in Fig. 4. For non-visual domain adaptation experiments, we employ multilayer perceptron (MLP) with two hidden layers (512 and 128 units) as the encoder and its symmetrical architecture as the decoder. Other parts of the model are the same as the previous one.

In all experiments, we adopt the ADAM optimization algorithm [30] with a learning rate of 10^{-4} and conduct mini-batch training with batch size 256 (128 labeled source samples and 128 unlabeled target samples) to train the model. For each task, we conduct hyper-parameters tuning by randomly choosing 1000 labeled target samples from test set as a validation set. Specifically, we restrict the hyper-parameters search for each task to $\lambda_1 = [0, 5]$, $\lambda_2 = [0, 5]$ and $\lambda_3 = [0, 2]$.

We implement our framework with TensorFlow [1]. To speed up the networks training process, all experiments are carried on one NVIDIA Tesla P4 GPU with 8GB on-board memory. The code has been released online.[1]

5.3 Results

We show the main results of all experiments in Tables 2 and 3. As can be seen, in most cases, when training with labeled source samples together with unlabeled target samples, unsupervised domain adaptation methods perform better than *source only* model. Specifically, our proposed method outperforms all competing methods on average in both visual and non-visual domain adaptation tasks.

5.3.1 USPS ↔ MNIST

We first evaluate the adaptation scenario between the handwritten digits dataset MNIST and USPS. To fairly compare with previous works, we follow the protocol proposed by Long et al. [38], which samples 1800 images from the USPS dataset and 2000 images from the MNIST dataset as the source or target domain. From the target testing dataset, we randomly choose 1000 labeled samples as a validation split to tune the hyper-parameters λ_1, λ_2, λ_3 and monitor the stopping criterion.

Since both MNIST and USPS are handwritten digits, the domain discrepancy is not as large as other scenarios. As can be seen from the first two columns of Table 2, the *source only* model achieves a relatively high accuracy when testing in the target domain. However, with DA, our method can further gain considerable performance improvement and outperform the competing methods by about 8%.

5.3.2 SVHN ↔ MNIST

In this scenario, we increase the gap between the distributions of two domains. The SVHN dataset is a real-world

[1] https://github.com/BoyuanJiang/CDMDA.

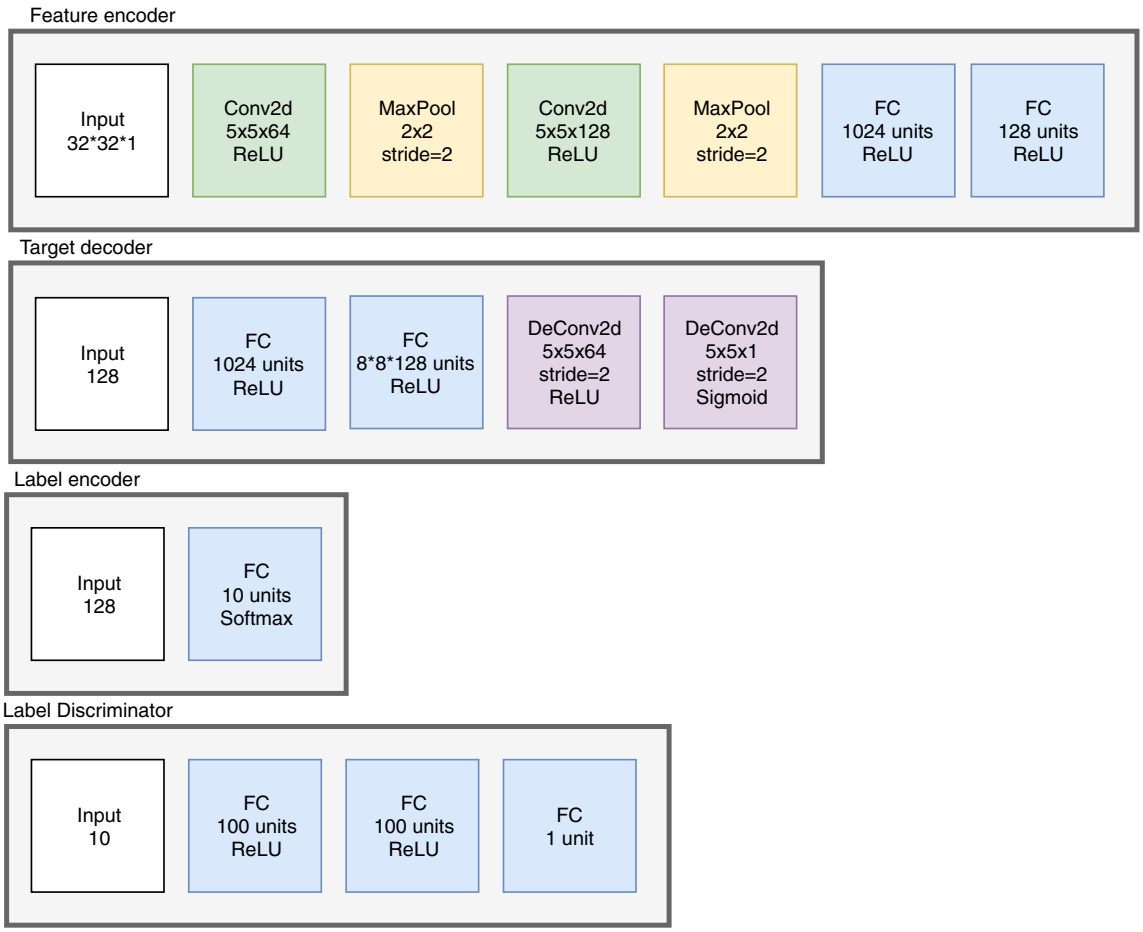

Fig. 4 The network topology for the visual domain adaptation experiments

dataset that is obtained from house number in Google Street View images. It contains significant variations, e.g., in scale, background, color, rotation, shape and some images contain more than one digit, while the images in MNIST are grayscale and most images contain a centered digit but variations in thickness (see Fig. 3 for detail). Therefore, this is a rather difficult domain adaptation scenario compared with the previous scenario. As above, we use 1000 target samples for validation.

From the third column of Table 2, it can be seen that, for the SVHN → MNIST scenario, the *source only* model achieves 66.2% accuracy on the target domain for the large discrepancy between domains. With DA, our proposed method can achieve 96.9% accuracy, which covers almost 90% of the gap between model trained on source samples only and model trained on the sufficient target samples with known target labels. On the contrary, the ADDA, DANN and DRCN methods result in a slight accuracy drop, which also indicates the task is more difficult than the case of USPS ↔ MNIST. To verify the effectiveness of our model, we utilize t-SNE [41] to visualize the source and the target features from the last layer of the feature encoder in Fig. 1. In the left part of the figure, though the source domain representations (blue) are discriminative and well clustered, the target representations (red) are separated and not aligned with the source representations well. However, in right part of the figure, both the source and the target representations are well clustered and aligned with each other, which makes the classifier trained on the source domain generalizable to the target domain.

The inverse direction MNIST → SVHN gives a failure example for our approach (approximately 35% target accuracy and *source only* model is 30%). To our best knowledge, there is no unsupervised DA method so far which can cover the large discrepancy from MNIST to SVHN and achieve satisfactory target accuracy.

5.3.3 SYN DIGITS → SVHN

In this experiment, we aim to address a practical domain adaptation scenario from synthetic images to real-world images, which is of great interest for research in computer vision. Generally, generating labeled synthetic data requires less effort than obtaining a large number of labeled

Table 2 Recognition accuracies of visual domain adaptation experiments on cross-domain digits datasets

Method	MNIST → USPS	USPS → MNIST	SVHN → MNIST	SYN → SVHN	Mean
SOURCE ONLY	83.0	73.7	66.2	83.5	76.6
ADDA	89.4	90.1	76.0	**91.2**	86.7
DANN	88.7	76.8	73.9	90.5	82.5
DRCN	91.8	73.7	82.0	84.2	82.9
DEEP CORAL	93.9	93.4	91.8	85.1	91.1
CDMDA (ours)	**96.2**	**96.0**	**96.9**	89.5	**94.7**
TRAIN ON TARGET	98.6	99.2	99.2	92.9	97.5

Bold values indicate the best result for each domain split

Table 3 Recognition accuracies of non-visual domain adaptation experiments on the Text dataset with Spanish as the target domain

Source articles	SOURCE ONLY	TRAIN ON TARGET	DANN	DEEP CORAL	**CDMDA (ours)**
English	44.1	95.7	43.6	35.8	**46.7**
French	61.9		63.9	66.2	**73.8**
German	60.6		62.3	63.5	**74.8**
Italian	64.3		65.7	68.3	**73.4**
Mean	57.8	95.7	58.9	58.5	**67.2**

Bold values indicate the best result for each domain split

real-world images. The SYN DIGITS dataset [15] contains about 500,000 images from Windows fonts by varying the text, positioning, orientation, background, stroke colors, and the amount of blur. Similar as above, we also randomly sample 1000 target domain images for validation.

In this scenario, ADDA achieves the highest accuracy 91.2% of all methods. Our result 89.5% is close to the highest one and gains considerable improvement (8%) compared to *source only* model.

5.3.4 Text categorization

To evaluate the performance of our model on the non-visual domain adaptation task, we apply our model to the Multilingual Reuters Collection dataset [2, 55]. This dataset, which is collected by sampling from the Reuters RCV1 and RCV2 collections, contains feature characteristics of 111,740 documents originally written in five different languages and their translations (i.e., English, French, German, Italian and Spanish), over a common set of six categories (i.e., C15, CCAT, E21, ECAT, GCAT and M11). Documents belonging to more than one of the 6 categories are assigned the label of their smallest category. Therefore, there are 12–30K documents per language and 11–34K documents per category. All documents are represented as a bag of words, and then 11,547 dimensions of TF-IDF features are extracted.

Similar to [25], we also select {English, French, German or Italian} as the source domain and select Spanish as the target domain. In each experiment, we randomly select 10,000 labeled samples from the source domain and 10,000 unlabeled samples from the target domain. Similar as above, 1000 target samples are selected for validation. In order to learn the model efficiently, we first perform principal components analysis (PCA) for dimension reduction and the dimensions after PCA are 1000 (approximately 60% energy is preserved). Table 3 shows that our model can significantly improve classification accuracy compared with *source only* model and other competing methods on three out of four tasks. However, it is worth noticing that all methods fail on the English → Spanish scenario. The reason we believe is that the domain discrepancy between these two domains is more significant than the other three scenarios.

5.4 Ablation study

In this subsection, we evaluate the effectiveness of four phases in our framework. Due to the space limitation, we only conduct the experiments on the SVHN → MNIST scenario. Results are shown in Table 4 and Fig. 5. In all experiments, supervised learning (**SL**) phase is selected, while other three phases (i.e., feature regularization (**FR**), reconstruction target (**RT**) and label regularization (**LR**)) are varied in each experiment. Therefore, there are eight different combinations altogether, as can be seen from Table 4. Index 1 means the model is trained without any target domain information, and Index 8 means all four phases of our framework are included when training. With all four phases together, the highest target accuracy can be achieved.

In Fig. 5, we make a comparison between model behavior with the label regularization phase and without that on four digits adaptation scenarios. It is obvious that with **LR** (the blue line), the validation error on the target domain declines faster than without **LR** (the green line).

Table 4 Ablation study of four training phases of CDMDA on the SVHN→MNIST task

Index	Supervised learning	Feature regularization	Reconstruction target	Label regularization	Accuracy
1	√	*o*	*o*	*o*	66.2
2	√	*o*	*o*	√	67.6
3	√	*o*	√	*o*	72.1
4	√	*o*	√	√	73.0
5	√	√	*o*	*o*	86.7
6	√	√	*o*	√	94.9
7	√	√	√	*o*	85.0
8	√	√	√	√	96.9

√ denotes this phase is considered in the experiment, and *o* denotes this phase is not considered

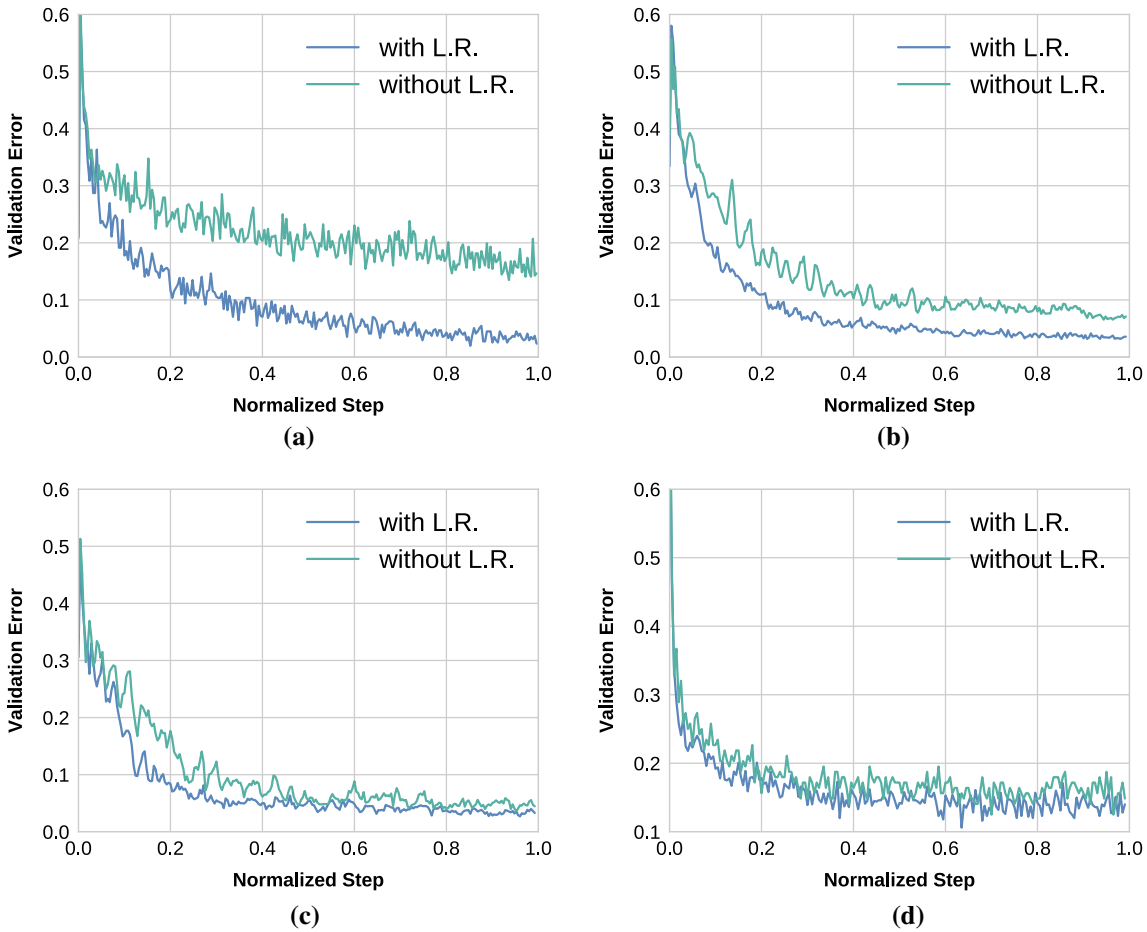

Fig. 5 Ablation study: with and without the label regularization (LR) phase on four digital domain adaptation scenarios. Each experiment adopts the same settings and iteration steps apart from with or without label regularization phase. **a** SVHN → MNIST. **b** USPS → MNIST. **c** MNIST → USPS. **d** SYN → SVHN

5.5 Entropy minimization versus label regularization

As we have mentioned in Sect. 4, a critical component to our paper is cluster assumption. We achieve this by label regularization, which confuses the distribution of predicted target labels with the categorical distribution via the adversarial training process. Since our aim is entropy minimization, one may ask whether directly minimizing the entropy of the target domain works. In this subsection, we empirically compare these two ways. From Fig. 6, it can be seen that both minimizing entropy directly and label regularization via adversarial training can minimize the entropy and improve the target validation accuracy

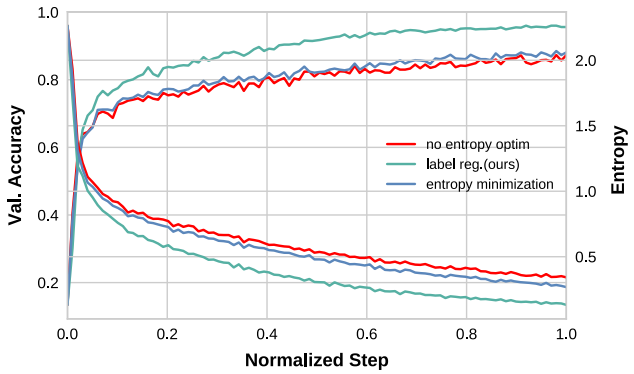

Fig. 6 Entropy (bottom half) and accuracy (upper half) with no entropy minimization, label regularization and minimizing entropy directly on SVHN → MNIST scenario. All experiment settings are same except the entropy optimization technique

compared with no entropy optimization method, which also demonstrates the effectiveness of cluster assumption. Also, when comparing the two entropy optimization methods, label regularization via adversarial training provides a lower entropy and a higher accuracy than minimizing entropy directly, meaning that our approach is outperforming than minimizing entropy directly.

5.6 Parameter sensitivity

There are three hyper-parameters λ_1, λ_2 and λ_3 involved in our model. Although these hyper-parameters can be determined via cross-valuation, insensitive parameter performance is desirable in real-world scenarios. Therefore, we conduct empirical parameter analysis for the three hyper-parameters. Due to space limitation, we only consider four visual domain adaptation tasks, i.e., SVHN → MNIST, SYN → SVHN, USPS → MNIST and MNIST → USPS. The initial parameters are chosen as $\lambda_1 = 2.0$, $\lambda_2 = 0.1$ and $\lambda_3 = 0.15$. Each time, only one parameter is allowed to change with other parameters fixed. The detailed results are shown in Fig. 7, and we give a brief analysis here. For λ_1, it is a regularization term to guarantee the small cross-domain distribution divergence. As shown in Fig. 7a, when λ_1 is close to 0, the target domain features fail to align with the source domain features, which leads to the poor performance on the target domain. Therefore, a reasonable value of λ_1 should be larger than 1. The hyper-parameter λ_2 is to balance the contribution of target domain reconstruction. From Fig. 7b can be seen, a large λ_2 may cause the model to pay more attention to finding representations to reconstruct target data rather than a common subspace for both source and target domains. The reasonable choice can be $\lambda_2 \in (0, 1]$. λ_3 controls the contribution of generator loss of the label discriminator. When λ_3 is extremely close to 0, the model suffers from the significant performance degradation in all scenarios. Therefore, a relatively large λ_3 (e.g., $\lambda_3 > 0.1$) can be a reasonable choice.

6 Conclusions and future work

In this paper, we propose Cross-Domain Minimization with Deep Autoencoder (CDMDA) for unsupervised domain adaptation, which performs a multitask learning strategy, i.e., simultaneously learning label prediction on the source domain and input reconstruction on the target domain via the shared feature representations aligned with CORAL in a unified framework. What is more, in order to correspond with the cluster assumption, we further incorporate a label discriminator to confuse the distribution of predicted target labels with the categorical distribution via the adversarial training process. Several domain adaptation experiments on both visual and non-visual datasets show that our model outperforms the competing unsupervised domain adaptation methods in most cases. Also, we empirically demonstrate the superiority of the label discriminator based on the

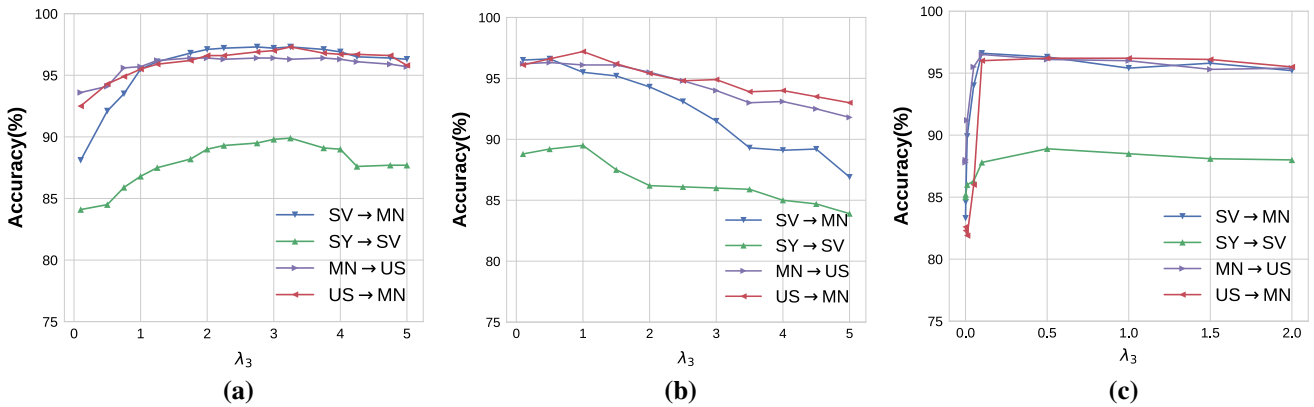

Fig. 7 Parameters sensitivity analysis of the proposed method: **a** accuracy w.r.t. λ_1; **b** accuracy w.r.t. λ_2; **c** accuracy w.r.t. λ_3 on four visual domain adaptation tasks. This figure is best viewed in color

cluster assumption in the field of unsupervised domain adaptation.

In the future, we plan to extend our proposal in the following two aspects. (1) Extending the CDMDA to multiple source domain adaptation method. (2) Extending our method to time series data by incorporating Recurrent Neural Network architecture.

Acknowledgements This research is supported by the National Science and Technology Major Projects (NO. 2013ZX03005013) and the Opening Foundation of the State Key Laboratory for Diagnosis and Treatment of Infectious Diseases (NO. 2014KF06).

References

1. Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, Corrado GS, Davis A, Dean J, Devin M et al (2016) Tensorflow: large-scale machine learning on heterogeneous distributed systems. ArXiv: Distributed, Parallel, and Cluster Computing
2. Amini M, Usunier N, Goutte C (2009) Learning from multiple partially observed views-an application to multilingual text categorization. In: Advances in neural information processing systems, pp 28–36
3. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell 39(12):2481–2495
4. Baktashmotlagh M, Harandi M, Salzmann M (2016) Distribution-matching embedding for visual domain adaptation. J Mach Learn Res 17(1):3760–3789
5. Bay H, Tuytelaars T, Van Gool L (2006) Surf: speeded up robust features. In: European conference on computer vision. Springer, pp 404–417
6. Ben-David S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW (2010) A theory of learning from different domains. Mach Learn 79(1–2):151–175
7. Bousmalis K, Trigeorgis G, Silberman N, Krishnan D, Erhan D (2016) Domain separation networks. In: Neural information processing systems, pp 343–351
8. Bromley J, Guyon I, LeCun Y, Säckinger E, Shah R (1994) Signature verification using a "siamese" time delay neural network. In: Advances in neural information processing systems, pp 737–744
9. Carlucci FM, Porzi L, Caputo B, Ricci E, Bulò SR (2017) Autodial: automatic domain alignment layers. In: International conference on computer vision
10. Caruana R (1998) Multitask learning. In: Learning to learn. Springer, pp 95–133
11. Chapelle O, Zien A, Ghahramani RCZ (2005) Semi-supervised classification by low density separation, pp 57–64
12. Chen M, Xu Z, Sha F, Weinberger KQ (2012) Marginalized denoising autoencoders for domain adaptation. In: International conference on machine learning, pp 1627–1634
13. Dai W, Yang Q, Xue GR, Yu Y (2007) Boosting for transfer learning. In: Proceedings of the 24th international conference on machine learning. ACM, pp 193–200
14. Fernando B, Habrard A, Sebban M, Tuytelaars T (2013) Unsupervised visual domain adaptation using subspace alignment. In: 2013 IEEE international conference on computer vision, pp 2960–2967. https://doi.org/10.1109/ICCV.2013.368
15. Ganin Y, Lempitsky V (2015) Unsupervised domain adaptation by backpropagation. In: International conference on machine learning, pp 1180–1189
16. Ganin Y, Ustinova E, Ajakan H, Germain P, Larochelle H, Laviolette F, Marchand M, Lempitsky V (2016) Domain-adversarial training of neural networks. J Mach Learn Res 17(1):2030–2096
17. Gatys LA, Ecker AS, Bethge M (2016) Image style transfer using convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2414–2423
18. Ghifary M, Kleijn WB, Zhang M, Balduzzi D, Li W (2016) Deep reconstruction-classification networks for unsupervised domain adaptation. In: European conference on computer vision, pp 597–613
19. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 580–587
20. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
21. Grandvalet Y, Bengio Y (2005) Semi-supervised learning by entropy minimization. In: Advances in neural information processing systems, pp 529–536
22. Gretton A, Borgwardt KM, Rasch M, Schölkopf B, Smola AJ (2007) A kernel method for the two-sample-problem. In: Advances in neural information processing systems, pp 513–520
23. Gretton A, Smola A, Huang J, Schmittfull M, Borgwardt K, Schölkopf B (2009) Covariate shift and local learning by distribution matching. MIT Press, Cambridge, pp 131–160
24. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
25. Hubert Tsai YH, Yeh YR, Frank Wang YC (2016) Learning cross-domain landmarks for heterogeneous domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5081–5090
26. Hull JJ (1994) A database for handwritten text recognition research. IEEE Trans Pattern Anal Mach Intell 16(5):550–554
27. Joachims T (1999) Transductive inference for text classification using support vector machines. ICML 99:200–209
28. Kamnitsas K, Baumgartner C, Ledig C, Newcombe V, Simpson J, Kane A, Menon D, Nori A, Criminisi A, Rueckert D et al (2017) Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In: International conference on information processing in medical imaging. Springer, pp 597–609
29. Kan M, Shan S, Chen X (2015) Bi-shifting auto-encoder for unsupervised domain adaptation. In: Proceedings of the IEEE international conference on computer vision, pp 3846–3854
30. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: International conference on learning representations
31. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp 1097–1105
32. La L, Guo Q, Cao Q, Wang Y (2014) Transfer learning with reasonable boosting strategy. Neural Comput Appl 24(3–4):807–816
33. Lecun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
34. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436
35. Lee DH (2013) Pseudo-label: the simple and efficient semi-supervised learning method for deep neural networks. In: Workshop on challenges in representation learning, vol 3. ICML, p 2

36. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC (2016) Ssd: single shot multibox detector. In: European conference on computer vision. Springer, pp 21–37
37. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3431–3440
38. Long M, Wang J, Ding G, Sun J, Philip SY (2013) Transfer feature learning with joint distribution adaptation. In: 2013 IEEE international conference on computer vision (ICCV). IEEE, pp 2200–2207
39. Long M, Cao Y, Wang J, Jordan MI (2015) Learning transferable features with deep adaptation networks. In: International conference on machine learning, pp 97–105
40. Long M, Zhu H, Wang J, Jordan MI (2016) Unsupervised domain adaptation with residual transfer networks. In: Advances in neural information processing systems, pp 136–144
41. Lvd M, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(Nov):2579–2605
42. Makhzani A, Shlens J, Jaitly N, Goodfellow I, Frey B (2015) Adversarial autoencoders. ArXiv preprint arXiv:151105644
43. Nair V, Hinton GE (2010) Rectified linear units improve restricted Boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp 807–814
44. Netzer Y, Wang T, Coates A, Bissacco A, Wu B, Ng AY (2011) Reading digits in natural images with unsupervised feature learning. In: NIPS workshop on deep learning and unsupervised feature learning, vol 2011, p 5
45. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359
46. Pietro Morerio VM Jacopo Cavazza (2018) Minimal-entropy correlation alignment for unsupervised deep domain adaptation. In: International conference on learning representations
47. Purushotham S, Carvalho W, Nilanon T, Liu Y (2017) Variational recurrent adversarial deep domain adaptation
48. Rumelhart DE, Hinton GE, Williams RJ (1985) Learning internal representations by error propagation. Tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science
49. Saenko K, Kulis B, Fritz M, Darrell T (2010) Adapting visual category models to new domains. In: European conference on computer vision. Springer, pp 213–226
50. Si S, Tao D, Geng B (2010) Bregman divergence-based regularization for transfer subspace learning. IEEE Trans Knowl Data Eng 22(7):929–942
51. Sun B, Saenko K (2016) Deep coral: correlation alignment for deep domain adaptation. In: ECCV 2016 workshops
52. Sun B, Feng J, Saenko K (2016) Return of frustratingly easy domain adaptation. In: AAAI
53. Tzeng E, Hoffman J, Zhang N, Saenko K, Darrell T (2014) Deep domain confusion: maximizing for domain invariance. ArXiv preprint arXiv:14123474
54. Tzeng E, Hoffman J, Saenko K, Darrell T (2017) Adversarial discriminative domain adaptation. In: 2017 IEEE conference on computer vision and pattern recognition (CVPR), pp 2962–2971. https://doi.org/10.1109/CVPR.2017.316
55. Ueffing N, Simard M, Larkin S, Johnson H (2007) NRCs portage system for WMT 2007. ACL 2007:185–188
56. Vincent P, Larochelle H, Bengio Y, Manzagol PA (2008) Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on machine learning. ACM, pp 1096–1103
57. Wang H, Xu A, Wang S, Chughtai S (2018) Cross domain adaptation by learning partially shared classifiers and weighting source data points in the shared subspaces. Neural Comput Appl 29(6):237–248
58. Wei P, Ke Y, Goh CK (2016) Deep nonlinear feature coding for unsupervised domain adaptation. In: IJCAI, pp 2189–2195
59. Yang S, Lin M, Hou C, Zhang C, Wu Y (2012) A general framework for transfer sparse subspace learning. Neural Comput Appl 21(7):1801–1817
60. Yang Z, Yu W, Liang P, Guo H, Xia L, Zhang F, Ma Y, Ma J (2018) Deep transfer learning for military object recognition under small training set condition. Neural Comput Appl 1–10
61. Zeiler MD, Krishnan D, Taylor GW, Fergus R (2010) Deconvolutional networks. In: 2010 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 2528–2535
62. Zhang H, Cao X, Ho JK, Chow TW (2017) Object-level video advertising: an optimization framework. IEEE Trans Ind Inform 13(2):520–531
63. Zhang H, Ji Y, Huang W, Liu L (2018) Sitcom-star-based clothing retrieval for video advertising: a deep learning framework. Neural Comput Appl. https://doi.org/10.1007/s00521-018-3579-x
64. Zhu X (2006) Semi-supervised learning literature survey. Computer Science, University of Wisconsin-Madison, vol 2, no. 3
65. Zhuang F, Cheng X, Luo P, Pan SJ, He Q (2015) Supervised representation learning: transfer learning with deep autoencoders. In: IJCAI, pp 4119–4125