Compact Feature Learning for Multi-domain Image Classification

Yajing Liu[1], Xinmei Tian[1(✉)], Ya Li[2], Zhiwei Xiong[1], Feng Wu[1]
[1]University of Science and Technology of China, Hefei, China
[2]iFLYTEK Research, Hefei, China
lyj123@mail.ustc.edu.cn, xinmei@ustc.edu.cn,
yali8@iflytek.com, zwxiong@ustc.edu.cn, fengwu@ustc.edu.cn

Abstract

The goal of multi-domain learning is to improve the performance over multiple domains by making full use of all training data from them. However, variations of feature distributions across different domains result in a non-trivial solution of multi-domain learning. The state-of-the-art work regarding multi-domain classification aims to extract domain-invariant features and domain-specific features independently. However, they view the distributions of features from different classes as a general distribution and try to match these distributions across domains, which lead to the mixture of features from different classes across domains and degrade the performance of classification. Additionally, existing works only force the shared features among domains to be orthogonal to the features in the domain-specific network. However, redundant features between the domain-specific networks still remain, which may shrink the discriminative ability of domain-specific features. Therefore, we propose an end-to-end network to obtain the more optimal features, which we call compact features. We propose to extract the domain-invariant features by matching the joint distributions of different domains, which have distinct boundaries between different classes. Moreover, we add an orthogonal constraint between the private features across domains to ensure the discriminative ability of the domain-specific space. The proposed method is validated on three landmark datasets, and the results demonstrate the effectiveness of our method.

1. Introduction

Image classification is one of the fundemental branches of computer vision tasks and has achieved impressive success with the advances of deep learning [11, 13]. However, due to the various exteral factors, such as viewpoint changes and background noise, in the real world, a classifier trained on one domain is likely to perform poorly on another domain. What's more, labeling enough samples in each domain is time consuming. To address these drawbacks, multi-domain learning aims to make full use of the training data to simulataneously improve the general classification performance over all domains [15]. Previous works regarding multi-domain learning trains classifiers in a collabrative way based on multi-task learning [23]. These approaches decompose the classifier in each domain into a shared part and a domain-specific part. The sharing knowledge used in the network significantly improves the classification performance over all domains. Unfortunately, these networks do not consider the matching of distributions across domains. Consequently, the shared knowledge cannot realize its full potential to improve the general classification performance in related domains.

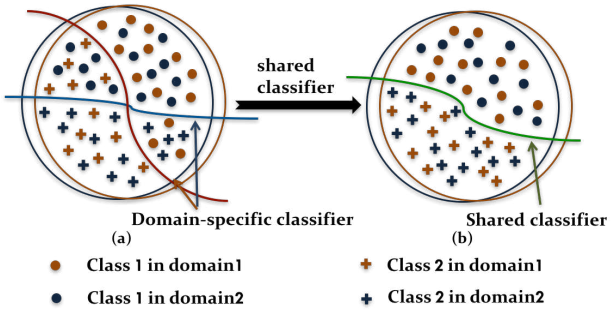


Figure 1. The two curves in (a) represent two classifiers in two domains. Features in each domain can be well-separated by its own classifier. However, when features from different domains are put together, the features from different classes across domains are mixed up. Therefore, we apply a shared classifier to match the conditional distributions $P(Y|\mathcal{F}_s(X))$ across domains. As shown in (b), features from different classes are well separated.

Recent works [4, 17, 3] use a network with the adversarial loss to overcome the aforementioned drawbacks. Let X, $\mathcal{F}_s(X)$ and Y denote the images, shared features after transformation and category labels, respectively. Adversarial training is applied to match the marginal distributions of shared features $\mathcal{F}_s(X)$. Under the assumption that the correlations $P(Y|\mathcal{F}_s(X))$ between the marginal distributions

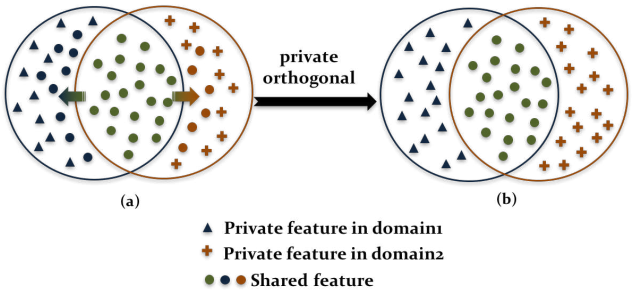


Figure 2. Redundant common features probably arise in the private space in previous works as shown in (a). The redundancy exists in the private network, then the private network does not make full use of the parameters and learning spaces. This can decrease the discriminability of the learned features, and thus increases the difficulty of accurate classification. In (b), we apply the orthogonal restriction between private features across domains in order to prevent the redundant features between domain-specific spaces.

$P(\mathcal{F}_s(X))$ and $P(Y)$ keep stable across domains, early approaches deem the distributions of features in the same class are matched. Unfortunately, the assumption does not always hold in practice. As shown in Figure 1(a), although the features in each domain can be well-separated by its own classifier, when they are put together only with the matching of marginal distributions $\mathcal{F}_s(X)$, features from different classes across domains can be mixed up. To separate samples from different classes, we apply a shared classifier that matches the conditional distributions $P(Y|\mathcal{F}_s(X))$ of the features. In this way, joint distributions $P(\mathcal{F}_s(X), Y)$ of different domains are matched, and samples from different classes are well separated as shown in Figure 1(b).

Another problem that arises in existing works [4, 17, 3] is that they make assumptions that domain-specific features can be learned automatically just utilizing the orthogonal constraint between the domain-specific space and the shared space. However, the orthogonal property between domain-specific spaces cannot be guaranteed, which results in the redundant features between domain-specific spaces. As shown in Figure 2(a), if the redundancy exists in the private network, the private network does not make full use of the parameters and learning spaces. This can decrease the discriminability of the learned features, and thus increases the difficulty of accurate classification. Furthermore, the shared space is not well learned for the reason that the shared features occur in the private space instead. Considering these drawbacks, as in Figure 2(b),we apply an orthogonal regularization between private features across domains to prevent the redundant features between domain-specific spaces and then ensure the uniqueness of each private space.

In this paper, we propose a compact feature learning method that guarantees the uniqueness of each domain-specific space and prevents the mixture of samples from different classes across domains during domain-invariant feature learning. The contributions of our work can be summarized as follows:

- The proposed network realizes compact feature learning, which extracts independent domain-specific features and the domain-invariant features with distinct class boundaries.
- We train an adversarial network with a shared classifier across domains, in which the joint distributions $P(\mathcal{F}_s(X), Y)$ can be automatically matched. Then the features in different classes across domains can be well separated.
- We propose an orthogonal regularization between private spaces to ensure the uniqueness of the private space and eradicate the redundant information in the network.

We conduct extensive experiments on several multi-domain datasets and the results demonstrate the effectiveness of our approach.

2. Related Work

Multi-domain learning aims to improve the classification performance in general domains by making full use of the information in each domain. Previous works have proposed to improve the performance of multi-domain learning using multi-task learning.

Existing approaches in multi-task learning based on neural networks can be broadly categorized into two methods: the parameter sharing methods [19], which entail the parameter restriction of hidden layers; the feature sharing methods, in which task-invariant representations can automatically be learned through the architecture of the network. In parameter restriction studies, Caruna et al. [2] realizes complete parameter sharing across tasks in lower layers, while task-specific output layers are maintained at the end of each task. Additionally, Duong et al. [6] and Yang et al. [24] respectively apply l_2 regularizations and trace norm regularizations between parameters to encourage the learning of common knowledge across tasks. However, the location of the shared layers in parameter restriction approaches is determined before training, which constrains the flexibility of the network [8]. For feature sharing approaches, the cross-stitch network [18] aims to linearly combine the outputs from the identical layers in each task-specific network, while the cross-stitch units determine the influence degree of the shared knowledge from each task. Moreover, cross-connect network [8] uses 1×1 convolution layers to connect the identical layers in each network for different tasks, which increases the flexibility of the network. However, all of these networks have no consider matching of distributions of the task-invariant features across tasks, subsequently degrading the performance of these works.

Unlike above methods, recent works aim to extract the shared knowledge through Bregman Divergence-Based Regularization [21] and the adversarial training network, which performs well in transfer learning and multi-domain learning. Transfer learning generally aims to learn invariant representations or parameters in different domains. However, multi-domain classification aims to learn invariant representations while preserving the private representations. Liu et al. [17] obtains the domain-invariant information in multiple related domains through the adversarial training strategy while eliminating redundant features between the private and shared spaces via orthogonal regularization. Chen et al. [3] proposes to use the negative log-likelihood loss and the l_2 loss instead of the adversarial loss alone. The combination of domain-invariant features and domain-specific features significantly improves the performance in text classification for multi-domain learning. In this paper, we focus on the multi-domain learning for image classification problems and realize the compact feature learning, consequently obtaining a remarkable result.

3. Approach

We first introduce multi-domain learning and define the notation used in this paper. Then we present details of the proposed multi-domain learning approach.

3.1. Multi-domain learning

Suppose the feature and the label spaces are represented by $\mathcal{X}$ and $\mathcal{Y}$ respectively. A domain defined on $\mathcal{X} \times \mathcal{Y}$ can be represented by a joint probability distribution $P(X, Y)$. For simplicity, $P_m(X, Y)$ denotes the joint distribution of the datasets in the m-th domain and $P_m(X)$ denotes the marginal distribution of the datasets. Each dataset is associated with a sample $D_m = \{x_i, y_i\}_{i=1}^{N_m}$, where N_m is the sample size of the m-th domain. Given C related domains $P_1(X, Y), P_2(X, Y), \ldots, P_C(X, Y)$ and their corresponding datasets $D_m = \{x_i, y_i\}_{i=1}^{N_m}$, the goal of multi-domain learning discussed in this paper is to learn a multi-branch model $f : \mathcal{X}_m \rightarrow \mathcal{Y}_m$, $m = \{1, 2, \ldots, C\}$ to classify all datasets correctly in parallel. Multi-domain learning considers the distribution bias between different domains, which might be caused by the camera's viewpoint, background noise or image style, etc. And then it aims to learn a general multi-branch model to improve the performance over all domains simultaneously.

3.2. Proposed multi-domain classification model

As shown in Figure 3, our multi-domain classification model is composed of three components: the shared feature learning network, the private feature learning network and the classification network for each domain.

To extract domain-invariant features through the matching of joint distributions of features $P(\mathcal{F}_s(X), Y)$ across domains, the joint adversarial shared network is applied. It is in the middle of the architecture with blue colors in Figure 3. Image features are extracted with several convolution neural networks. Besides, the matching of feature distributions is guaranteed by one image classification network and one discriminator simultaneously. The multi-player adversarial discriminator is applied to ensure the match of marginal distributions $P(\mathcal{F}_s(X))$ of features [16]. Additionally, the shared classifier matches the conditional distributions $P(Y|\mathcal{F}_s(X))$ across domains. Consequently, the joint distributions of shared features $P(Y, \mathcal{F}_s(X))$ in different domains are matched.

The private network aims to learn the domain-specific information which preserves the characteristics of each domain. It is shown at two sides of the shared network in Figure 3. To obtain the domain-specific knowledge and further promoting domain-invariant feature learning, we introduce two types of orthogonal regularizations: the regularization between private features and shared features in each domain; the regularization between private features across domains. Consequently, the independence of each feature space can be guaranteed. At training and testing process, the learned shared features $\mathcal{F}_s(x)$ and private features $\mathcal{F}_p(x)$ are concatenated and fed into the classification network for each domain. We call the shared features and private features as compact features $\mathcal{F}_{compact}$:

$$\mathcal{F}_{compact} = [\mathcal{F}_s(x), \mathcal{F}_p(x)] \tag{1}$$

The main learning goal in our multi-domain learning architecture can be formulated as:

$$L_{domain} = -\sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\mathcal{C}_m(y_i^m|\mathcal{F}_{compact})] \tag{2}$$

we denote the classifier in each domain by $\mathcal{C}_m$. The proposed compact feature learning improves the classification performance of each domain.

3.3. Joint adversarial shared network

Previous methods are proposed to obtain domain-invariant features through matching the marginal distributions of features $P(\mathcal{F}_s(X))$. Each domain has its own classifier, which has no restriction of the matching of conditional distributions $P(Y|\mathcal{F}_s(X))$ across domains. Without considering their class labels in the adversarial training, domain-invariant features in different classes are probably mixed up. To address this drawback, our approach separates shared features from different classes through applying an adversarial training algorithm with a shared classifier.

This shared classifier and the domain adversarial network cope with all domain-invariant features in the same manner. The domain adversarial network is connected with the feature exacting network through a gradient reversal layer (GRL), which layer [9] forwards the input to the following layers but reverses the gradient during the backward

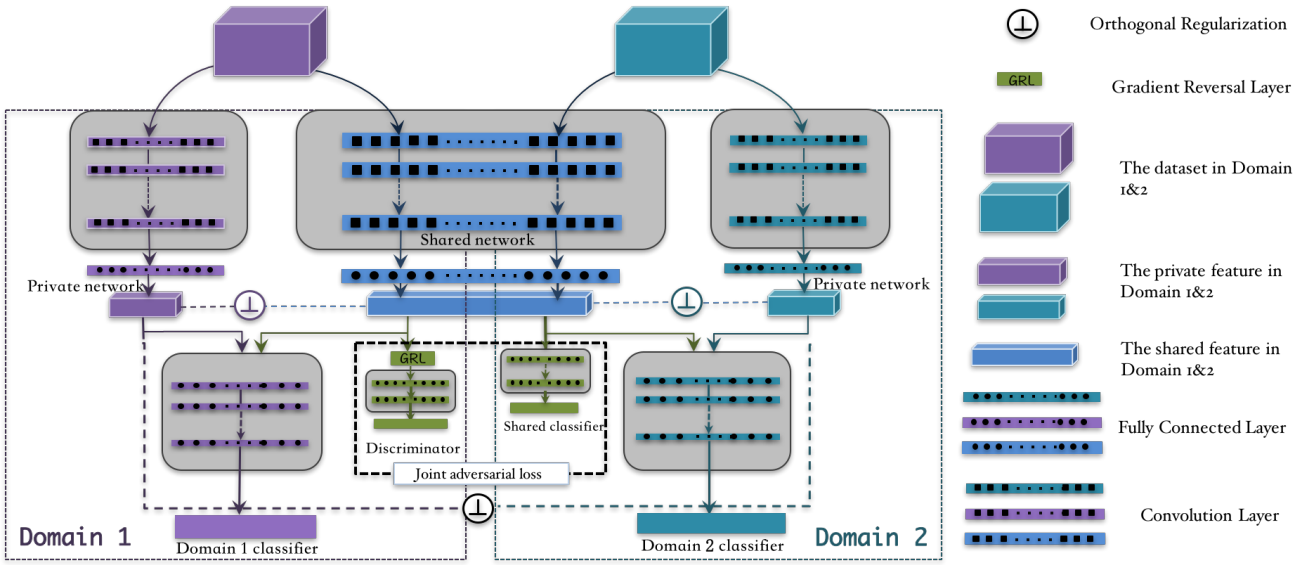


Figure 3. The proposed multi-domain classification model. The joint adversarial shared network is in the middle of the architecture with blue colors. A domain discriminator and a shared classifier are applied at the end of the shared feature extracting network to guarantee the joint distribution matching of features. The private network is shown at two sides of the shared network. Feature orthogonal regularization is applied between private features across domains, in addition to private features and shared features in each domain. Compared with [3], we add the joint adversarial loss and the orthogonal regularizations between private features across domains.

propagation. We define the minimax game of joint adversarial learning(jal) as follow:

$$L_{jal} = \min_{\mathcal{F}_s, \mathcal{C}_s} \max_{d_1,\ldots,d_c} (-\sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\mathcal{C}_s(y_i|(\mathcal{F}_s(x_i))] + \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[d_m(\mathcal{F}_s(x_i))]) \tag{3}$$

where d_m is the m-th domain discriminator and $\mathcal{F}_s, \mathcal{C}_s$ represent the shared feature extracting network and the shared classification network, respectively. Since we constrain $\mathcal{C}_s$ to a simple linear transformation or shallow network, $\mathcal{C}_s(Y|\mathcal{F}_s(X))$ can be simplified as $P^i_{\mathcal{F}_s}(Y|X)$. Under the assumption that the shared classifier predicts the samples from the same class in different domains as the same accurate one-hot label, the network can perfectly match the conditional distributions across different domains. It can be formulated as $P^1_{\mathcal{F}_s}(Y|X) = \ldots = P^C_{\mathcal{F}_s}(Y|X) = P_{\mathcal{F}_s}(Y|X)$. At the same time, under the restriction of the negative gradients, the shared feature extracting network expects that the exacted features can mislead the domain classification results, but the discriminator tries its best to correctly classify the domain category of features [10]. Consequently, the domain discriminator results in the minimax game in the network, which leads to the domain-invariant property of the shared features. At the end of the training phase, the network will reach a point that each network cannot improve its performance and the domain-invariant features in joint distribution matching are obtained.

- **Proof of joint distribution matching**

The cost function of the minimax game in the network can be formulated as:

$$\phi = \min_{\mathcal{F}_s} \max_{d_1,\ldots,d_c} \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[d_m(\mathcal{F}_s(x_i))], \quad s.t. \sum_{m=1}^{C} d_m((\mathcal{F}_s(x))) = 1 \tag{4}$$

The marginal distribution and the joint distribution of shared features after transformation in the i-th domain are denoted by $P^i_{\mathcal{F}_s}(X)$ and $P^i_{\mathcal{F}_s}(X)$, respectively.

The optimal procedure in the network is consist of two stages. In Stage 1, the discriminator is unrelated to the distribution of Y, then, we provide the optimal discriminator under a fixed transformation $\mathcal{F}_s$ from [10, 16] directly. In Stage 2, we obtain the optimal domain-invariant feature transformation under the shared classifier and the fixed discriminator in Stage 1.

Stage 1. *Let $x', y = \mathcal{F}_s(x)$. For a fixed transformation $\mathcal{F}_s$, the optimal prediction probabilities $d^1_{\mathcal{F}_s}, \ldots, d^C_{\mathcal{F}_s}$ of discriminator d are:*

$$d^{m*}_{\mathcal{F}_s}(x') = \frac{P^{m*}_{\mathcal{F}_s}(x')}{\sum_{m=1}^{C} P^{m*}_{\mathcal{F}_s}(x')} \tag{5}$$

Stage 2.*When we obtain the optimal point of d, let*

$$\begin{aligned}\phi' &= \min_{\mathcal{F}_s} \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[d_m(\mathcal{F}_s(x_i))], \\ &= \min_{\mathcal{F}_s} \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\frac{P^m_{\mathcal{F}_s}(x')}{\sum_{m=1}^{C} P^m_{\mathcal{F}_s}(x')}]\end{aligned} \tag{6}$$

ϕ' achieves the value $-ClogC$ in the global minimum, and the balanced point is achieved if and only if $P^1_{\mathcal{F}_s}(X,Y) = \cdots = P^C_{\mathcal{F}_s}(X,Y)$

Proof. Under the assumption that we can obtain the optimal shared classifier, we can arrive the optimal point $P^1_{\mathcal{F}_s}(Y|X) = \cdots = P^C_{\mathcal{F}_s}(Y|X) = P_{\mathcal{F}_s}(Y|X)$. Then we can obtain:

$$\begin{aligned}\phi' &= \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} [log[\frac{P^m_{\mathcal{F}_s}(x')}{\sum_{m=1}^{C} P^m_{\mathcal{F}_s}(x')}] + logC] - ClogC \\ &= \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\frac{P^m_{\mathcal{F}_s}(x') P_{\mathcal{F}_s}(y|x')}{\frac{1}{C}\sum_{m=1}^{C} P^m_{\mathcal{F}_s}(x') P_{\mathcal{F}_s}(y|x')}] - ClogC \\ &= \sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\frac{P^m_{\mathcal{F}_s}(x', y)}{\frac{1}{C}\sum_{m=1}^{C} P^m_{\mathcal{F}_s}(x', y)}] - ClogC \\ &= C \cdot JSD(P^1_{\mathcal{F}_s}(X,Y), \ldots, P^C_{\mathcal{F}_s}(X,Y)) - ClogC\end{aligned} \tag{7}$$

since the Jensen-Shannon divergence(JSD) loss is non-negative, the only zero solution arrives at $P^1_{\mathcal{F}_s}(X,Y) = P^2_{\mathcal{F}_s}(X,Y) = \cdots = P^C_{\mathcal{F}_s}(X,Y)$, which implies that the joint distributions of learned feature representations across domains are perfectly matched. Moreover, ϕ' achieves the global minimum $-ClogC$.

3.4. Orthogonal regularization

Private neural networks are shown on two sides of the shared network in Figure 3. They aim to learn domain-specific information that can not be learned in the shared network, which mostly preserve the discriminative ability and provide strong support for the domain-invariant features. To obtain the domain-specific knowledge, we introduce two types of orthogonal regularizations: the orthogonal regularization between domain-invariant features and domain-specific features in each domain; the orthogonal regularization between domain-specific features across domains. We call the first kind of regularization as intra-domain orthogonal regularization and the second as extra-domain orthogonal regularization.

In previous works, they only apply a soft subspace orthogonal regularization [1] between the private and the shared space in each domain to ensure their independence, which can be formulated as follows:

$$R_{intra\_orth} = \sum\nolimits_{m=1}^{C} \|\mathbf{S}_m^\top \mathbf{H}_m\|_F^2 \tag{8}$$

where $\mathbf{S}_m$ is the matrix whose rows are the domain-invariant features in the m-th domain, and $\mathbf{H}_m$ is the matrix whose rows are the domain-specific features in the m-th domain.

There exists a potential drawback in this model: the domain-invariant features can simultaneously occur in multiple private networks among different domains. Let $\mathcal{H}^1, \mathcal{H}^1_p$ represents the entire feature space and private feature space in domain 1, $\mathcal{H}^2, \mathcal{H}^2_p$ represents the entire feature space and the private feature space in domain 2, and $\mathcal{S}$ represents the shared space between domains; then, we can formulate the loss in previous work as:

$$\begin{cases}\mathcal{H}^1 = \mathcal{H}^1_p + \mathcal{S} \\ \mathcal{H}^2 = \mathcal{H}^2_p + \mathcal{S} \\ \mathcal{H}^1_p \cap \mathcal{S} = \{0\} \\ \mathcal{H}^2_p \cap \mathcal{S} = \{0\}\end{cases} \tag{9}$$

The solution is not unique and can be represented as:

$$\begin{cases}\mathcal{H}^1_p = \mathcal{H}^{1*}_p + \mathcal{S}' \\ \mathcal{H}^2_p = \mathcal{H}^{2*}_p + \mathcal{S}' \\ \mathcal{S} = \mathcal{S}^* - \mathcal{S}'\end{cases} \tag{10}$$

where $\mathcal{H}^{1*}_p, \mathcal{H}^{2*}_p, \mathcal{S}^*$ represent the optimal feature spaces; $\mathcal{S}'$ is the complementary subspace of $\mathcal{S}$ in $\mathcal{S}^*$. The duplicate subspace $\mathcal{S}'$ probably arises in the private feature spaces, then the individual network loses its own characteristics. Therefore, we apply the subspace orthogonal regularization between private features across domains. We express the extra-domain regularization as:

$$R_{extra\_orth} = \sum_{m_1=1}^{C} \sum_{\substack{m_2=1, \\ m_2 \neq m_1}}^{C} \|\mathbf{H}_{m_1}^\top \mathbf{H}_{m_2}\|_F^2 \tag{11}$$

where we denote the domain-specific features in the m_1-th and m_2-th domains as $\mathbf{H}_{m_1}$ and $\mathbf{H}_{m_2}$. Then, by forcing $\mathcal{S}' = \{0\}$, we obtain the optimal solutions $\mathcal{H}^{1*}_p, \mathcal{H}^{2*}_p, \mathcal{S}^*$. Furthermore, the disappearance of domain-invariant features in the domain-specific feature space promotes the learning of the shared network.

To reduce the correlation between private features and shared features in each domain, we add the procedure of training domain-specific features individually. We set all domain-invariant features to zero and only train the domain-specific features in this procedure:

$$\mathcal{F}_{zeros\_private} = [zeros, \mathcal{F}_p(x)] \tag{12}$$

The private feature learning can be formulated as:

$$L_p = \min_{\mathcal{F}_p}(-\sum_{m=1}^{C} \frac{1}{N_m} \sum_{i=1}^{N_m} log[\mathcal{C}_m(y_i^m | \mathcal{F}_{zeros\_private})]) \tag{13}$$

where $\mathcal{F}_p$ represents parameters in the private feature extracting network. The entire network realizes the optimal independent feature learning with the well-designed regu-

larizations and losses.

3.5. Training procedure

The aforementioned losses and regularizations can be linearly combined as follows:

$$L_{all} = L_{domain} + \lambda_1 L_{jal} + \lambda_2 R_{intra\_orth} + \lambda_3 R_{extra\_orth} \tag{14}$$

where λ_i denotes the weight of each restriction. The stochastic gradient descent is applied to update the parameters in the network and the gradient reversal layer reverses the gradient from the domain discriminative network to update the parameters in the feature extracting network. The entire training procedure is shown in Algorithm 1, the optimization process stops until finding the saddle point in neural networks.

Algorithm 1 The training procedure for proposed compact feature learning

Input:

C labeled datasets $\{X_m, Y_m\}_{m=1}^{C}$; Initiated shared feature extractor $\mathcal{F}_s$; Initiated private feature extractor $\mathcal{F}_p$; Shared category classifier $\mathcal{C}_s$; Domain discriminator D; Domain category classifier $\{\mathcal{C}_m\}_{m=1}^{C}$;

Output:

Well-trained shared feature extractor $\mathcal{F}_s^*$; Initiated private feature extractor $\mathcal{F}_p^*$; Domain category classifier $\{\mathcal{C}_m\}_{m=1}^{C}$;

while not converged **do**
  Sample mini-batch from $\{X_m, Y_m\}_{m=1}^{C}$
  for m = 1 : C **do**
    Update $\mathcal{F}_p$, $\mathcal{F}_s$, $\mathcal{C}_m$ by Eq.2, Eq.8, Eq.11;
    Update $\mathcal{F}_p$ by Eq.13, fix the classifiers in each domain and set the shared features to zeros, then update the parameters in $\mathcal{F}_p$;
    Update $\mathcal{F}_s$, $\mathcal{C}_s$, D by Eq.3, reverse the gradient from the discriminator during the backward propagation to update the parameters in the $\mathcal{F}_s$;
  end for
end while
return $\mathcal{F}_p^* = \mathcal{F}_p; \mathcal{F}_s^* = \mathcal{F}_s; \mathcal{C}_m^* = \mathcal{C}_m;$

4. Experiments

We evaluate our proposed method on three image classification datasets: the MNIST dataset, the VLCS dataset [22, 12] and the PACS [14] dataset. We compare our proposed methods with these following works:

- **Indiv**: Different networks are applied to deal with different domains. Each network is trained individually without any connection with the networks for related domains.

Table 1. Performance comparison between different methods for multi-domain learning with respect to accuracy(%) on MNIST dataset.

Method	Domain_1	Domain_2
	Mnist	Mnist_m
Indiv	96.01	85.76
Indiv_l2	96.32	85.93
Cross_stitch	96.38	86.34
Cross_connect	96.28	87.09
Share	96.48	86.16
MAN	96.48	86.62
JARN	96.56	88.54
JOARN	**97.10**	**89.34**

Figure 4. Visualization of examples in MNIST-M and MNIST dataset.

- **Indiv_l2 [6]**: Each domain has an individual network, and the l_2 distance regularization among parameters between the networks for different domains is applied.

- **Cross_stitch [18]**: Different networks are applied to address different domains. The outputs in identical shallow layers of each domain are linearly combined for feature sharing.

- **Cross_connect [8]**: Different networks are applied to address different domains. 1×1 convolution layers between identical layers in each domain are applied to learn the influence degree of each feature map in the related domains.

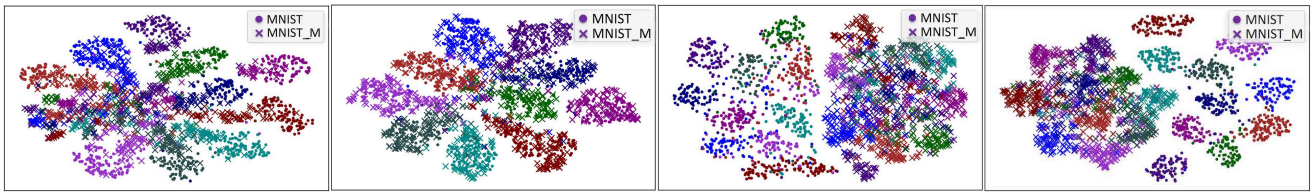
- **Share**: Instead of the individual network for each domain, a single network is applied to simultaneously address all related domains.

- **MAN [3]**: The adversarial training strategy is applied to obtain domain-invariant information and orthogonal regularizations are applied to eliminate redundant features between the private and shared feature spaces.

- **JARN** (joint adversarial restriction network): The joint adversarial loss is applied to obtain the joint distribution matching and orthogonal regularizations are applied to eliminate redundant features between the private and shared feature spaces.

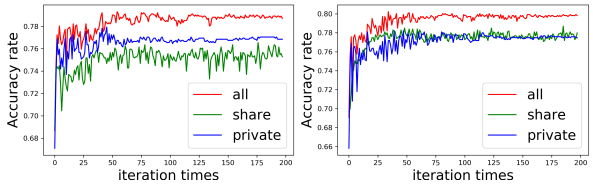
- **JOARN** (joint orthogonal and adversarial restriction network): Losses and regularizations in Eq.14 are applied in JOARN.

To ensure the fairness of the experiments, the same architecture is applied in these aforementioned works.



(a) Initial shared features (b) Shared features with restriction (c) Initial private features (d) Private features with restriction

Figure 5. The visualization of extracted features in MNIST and MNIST-M. Different colors refer to different classes. From Figure 5(a) and Figure 5(b), we can observe that with the help of the discriminator and the shared classifier, the distribution of shared features becomes nondiscriminative across domains and has a clear and distinct boundaries between classes. From Figure 5(c) and Figure 5(d), the domain-specific features obtain uniqueness and discriminability in each domain with the orthogonal restriction between them.



(a) Accuracy curve of voc2007 (b) Accuracy curve of sun09

Figure 6. Accuracy curves in VLCS. The combination of domain-invariant features and domain-specific features can significantly improve the classification performance in each domain.

4.1. MNIST and MNIST-M Dataset

4.1.1 Settings

Examples from MNIST-M and MNIST datasets are shown in Figure 4. The MNIST-M dataset is composed of MNIST dataset and patches randomly extracted from BSD500. The variation of backgrounds means the change of image domains. Obviously, the classification in MNIST-M is more difficult than that in MNIST because of the more complicated background in the pictures. We randomly select 1000 training samples in each domain. All 10000 test examples in each dataset are used. The architecture that we used is identical to that in [1]. It has two convolution layers and three fully connected layers. We connect the GRL layer to the first fully connected layer in the main network. The domain-invariant features and domain-specific features are extracted at the same position.

4.1.2 Visualization and analysis

As indicated in Table 1, we can see that the improvement in different domains is unbalanced since the shared features account for different importance in the classification problems across datasets. Previous works such as indiv_l2, cross_stitch, and cross_connect only obtain limited improvement in multi-domain learning. This is because they do not consider the matching of domain-invariant feature distributions across domains. Then, the domain-invariant features cannot be well learned. MAN achieves better performance since it obtains domain-invariant features through adversarial training, and the domain-specific features and domain-invariant features are extracted individually to prevent the interfering between them. Our JARN outperforms MAN. The shared classifier is applied in JARN to get the joint distribution matching of domain-invariant features. JOARN obtains a better result and outperforms all other methods.

To obtain an intuitive observation of the influence of our proposed regularizations in the network, we use the t-SNE projection to visualize the domain-invariant features and domain-specific features in different situations. From Figure 5(a), we can observe two drawbacks of the shared features with no restriction. One drawback is the network can only learn the partial matching between the distributions of shared features across domains. The other is the cross-domain features in different classes are mismatched at the interfacial boundaries, which produces the difficulty in classification. As shown in Figure 5(b), with the adversarial loss and the shared classifier, the distribution of shared features becomes nondiscriminative across domains and has a clear and distinguishable boundary for the main learning goal. On the other side, the network prefers to learn domain-specific features of different properties in different domains in Figure 5(c). Along with the increasing complexity of the dataset, there is probable intersection between domain-specific feature spaces. In Figure 5(d) the private features in each domain obtain the uniqueness and discriminability with the orthogonal restriction between them.

4.2. VLCS Dataset

4.2.1 Setting

VLCS is a real world image classification dataset. We select 3 different types of subdatasets in it for three domains: PASCAL VOC2007 (V) [7], LabelMe (L) [20] and SUN09 (S) [5]. Each subdataset contains five common classes: “bird”, “car”, “chair”, “dog” and “person”. Each subdataset is randomly split into two parts: 70% for training and 30% for testing. To be consistent with earlier researches and facilitate comparison experiments, we only

Table 2. Performance comparison between different methods for multi-domain learning with respect to accuracy(%) on VLCS dataset.

		Indiv	Indiv_l2	Cross_stitch	Cross_connect	Share	MAN	**JARN**	**JOARN**
Dataset_pair_1	Voc2007	78.13	78.13	77.73	78.52	78.42	78.72	79.31	**80.00**
	Sun09	78.40	78.60	78.80	78.90	78.90	78.80	81.03	**81.54**
Dataset_pair_2	Voc2007	78.13	78.62	77.73	78.42	77.93	77.93	78.62	**79.41**
	Labelme	73.13	73.5	74.13	73.25	74.13	73.88	74.5	**76.13**
Dataset_pair_3	Sun09	78.40	78.80	78.90	78.60	78.49	78.30	78.70	**80.02**
	Labelme	73.13	72.75	72.63	72.75	72.88	73.5	74.13	**75.75**

Table 3. Performance comparison between different methods for multi-domain learning with respect to accuracy(%) on PACS dataset.

		Indiv	Indiv_l2	Cross_stitch	Cross_connect	Share	MAN	**JARN**	**JOARN**
Dataset_pair_1	Cartoon	91.65	91.94	92.26	91.80	91.65	91.80	92.65	**93.35**
	Art_painting	87.38	87.54	87.06	87.54	87.86	88.19	89.00	**89.97**
Dataset_pair_2	Cartoon	91.65	91.94	92.08	91.94	91.51	91.51	92.93	**93.21**
	Sketch	90.65	90.74	90.91	90.65	91.08	91.59	92.44	**93.12**
Dataset_pair_3	Art_painting	87.38	87.70	87.70	87.86	86.73	87.54	88.51	**89.42**
	Sketch	90.65	90.91	91.16	91.08	91.33	91.93	92.10	**93.54**

use two subdatasets each time. However, new domains can be easily extended by appending the affiliated private network. Following previous works, the structures of shared network and the private network in our multi-domain model are the same as AlexNet[13], and the convolution layers are initial with the pretrained Alexnet. Moreover, we extract the FC6 features as the domain-invariant features and domain-specific features[14], and we use three fully connected layers $(1024-1024-2)$ in our domain discriminative network then connect it with the FC6 layer in the main network via the GRL.

4.2.2 Analysis

As shown in Figure 6, the domain-invariant features and domain-specific features can efficiently classify the images, and the combination of shared features and private features significantly improves the classification performance. Additionally, along with the training, the performance of each network gradually improves and arrives at a stable stage. The entire network realizes the optimal independent feature learning. The experimental results are summarized in Table 2. Similar conclusions can be obtained as in the experiments of MNIST and MNIST-M. Note that the result of MAN is worse than previous works sometimes. This outcome is because the shared features in MAN only match the marginal distributions, and the accuracy varies significantly in the training process since the changes of the conditional distributions $P(Y|\mathcal{F}_s(X))$ across domains. Moreover, the redundant features in the domain-specific network lead to the discriminablity of private features. Our network extracts the shared features through matching joint distributions $P(\mathcal{F}_s(X), Y)$ across domains and obtains independent private features, thus performs better than other ones.

4.3. PACS Dataset

We select three different image styles in PACS: art-painting (A), cartoon (C) and sketch (S). Each image style can be viewed as one domain. The image styles across domains in PACS are of marked difference. We also split each subdataset into two parts randomly: 70% for training and 30% for testing. There are 7 common categories in each dataset: "dog", "elephant", "giraffe", "guitar", "horse", "house", "person". Additionally, the training architecture of the PACS is identical to that of VLCS, except the features are extracted from the FC7 layer [14] and the GRL layer is connected to the FC7 layer. Note that PACS has a bigger domain bias across subdatasets than VLCS. Consequently, the conditional distributions $P(Y|\mathcal{F}_s(X))$ across domains vary significantly and the matching of joint distributions of features is of great important. From the results presented in Table 3, we can observe that the proposed JARN and JOARN algorithm achieve the better performance than others, which demonstrate the effectiveness of our method.

5. Conclusion

In this paper, we propose compact feature learning to individually extract more optimal domain-invariant features and domain-specific features. We train the adversarial network with a shared classifier across domains, where the joint distribution of each domain can be matched. Moreover, the orthogonal loss is applied to ensure the uniqueness of each private space. Compact feature learning significantly improves the general classification performance over related domains, as the results demonstrate the effectiveness of our method.

Acknowledgements

We acknowledge funding from National Key R&D Program of China under Grants 2017YFA0700800 and 2017YFB1002203, and Natural Science Foundation of China (NSFC) under Grant 61872329.

References

[1] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems*, pages 343–351, 2016.

[2] R. Caruna. Multitask learning: A knowledge-based source of inductive bias. In *Machine Learning: Proceedings of the Tenth International Conference*, pages 41–48, 1993.

[3] X. Chen and C. Cardie. Multinomial adversarial networks for multi-domain text classification. *arXiv preprint arXiv:1802.05694*, 2018.

[4] X. Chen, Z. Shi, X. Qiu, and X. Huang. Adversarial multi-criteria learning for chinese word segmentation. *arXiv preprint arXiv:1704.07556*, 2017.

[5] M. J. Choi, J. J. Lim, A. Torralba, and A. S. Willsky. Exploiting hierarchical context on a large database of object categories. 2010.

[6] L. Duong, T. Cohn, S. Bird, and P. Cook. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pages 845–850, 2015.

[7] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[8] S. Fukuda, R. Yoshihashi, R. Kawakami, S. You, M. Iida, and T. Naemura. Cross-connected networks for multi-task learning of detection and segmentation. *arXiv preprint arXiv:1805.05569*, 2018.

[9] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *European Conference on Computer Vision*, pages 158–171. Springer, 2012.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[14] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales. Deeper, broader and artier domain generalization. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 5543–5551. IEEE, 2017.

[15] S. Li and C. Zong. Multi-domain sentiment classification. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pages 257–260. Association for Computational Linguistics, 2008.

[16] Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018.

[17] P. Liu, X. Qiu, and X. Huang. Adversarial multi-task learning for text classification. *arXiv preprint arXiv:1704.05742*, 2017.

[18] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003, 2016.

[19] S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[20] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1-3):157–173, 2008.

[21] S. Si, D. Tao, and B. Geng. Bregman divergence-based regularization for transfer subspace learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(7):929–942, 2010.

[22] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1521–1528. IEEE, 2011.

[23] F. Wu and Y. Huang. Collaborative multi-domain sentiment classification. In *2015 IEEE International Conference on Data Mining (ICDM)*, pages 459–468. IEEE, 2015.

[24] Y. Yang and T. M. Hospedales. Trace norm regularised deep multi-task learning. *arXiv preprint arXiv:1606.04038*, 2016.