Maintaining Structural Information by Pairwise Similarity for Unsupervised Domain Adaptation

Jian Liu[1,2] and Yahong Han[1,2](✉)

[1] College of Intelligence and Computing, Tianjin University, Tianjin, China
{liujian_cic,yahong}@tju.edu.cn

[2] Tianjin Key Lab of Machine Learning, Tianjin University, Tianjin, China

Abstract. Unsupervised domain adaptation (UDA) aims to transfer the knowledge learned from the labeled source domain to the unlabeled target domain. Among them, the source domain and the target domain have the same label space, but the representation distributions of their input space are different. Mainstream approaches resort to domain adversarial training to align input distributions of two domains in the feature space. Although these methods have made remarkable progress, they have the risk of destroying discriminative structural information between different classes in the target domain. To alleviate this risk, we are inspired by the *problem reduction* method in ensemble methods and binarization techniques, and propose a novel approach Maintaining Structural Information of the target domain based on Pairwise semantic Similarity (whether two instances belong to the same class or not) (MSIPS). Specifically, We introduce Contrastive Learning to obtain feature prototypes for each category on the source domain, and then use these prototypes to predict the similarity of paired target domain samples. Finally, we restrict the target domain to maintain discriminative structural information through such weak information (i.e., pairwise similarity). Extensive experiments of various domain shift scenarios show that our method obtains competitive performance with SOTA, and qualitative visualization can demonstrate the effectiveness of our method.

Keywords: Unsupervised domain adaptation · Problem reduction · Pairwise similarity

1 Introduction

The breakthrough of convolutional neural network (CNN) [12] in the field of computer vision is inseparable from the support of massive labeled data [6]. Nevertheless, the collection and annotation of numerous data is an extremely expensive and time-consuming process. Meanwhile, In many practical scenarios, the models performed well on testing data will have serious performance degradation when predicting the data with distribution discrepancy (i.e., domain

L. Fang et al. (Eds.): CICAI 2022, LNAI 13605, pp. 268–279, 2022.
https://doi.org/10.1007/978-3-031-20500-2_22

shift). To alleviate these problems, unsupervised domain adaptation (UDA) has attracted a lot of attention, which transfers knowledge from a label-rich source domain to a fully-unlabeled target domain.

A variety of unsupervised domain adaptation methods have been proposed and have achieved significant progress [3,9,17,21,29]. The mainstream methods resort to domain adversarial training to align input distributions of two domains in the feature space, so that the model has good generalization ability in inferencing the data from different domains with domain shift. Although this method effectively reduces the distribution discrepancy in the feature space of the source domain and the target domain, it also leads to the mixing of the representation of different classes in the target domain to a certain extent. Recently, some studies [27,29,33] have begun to consider compensating or maintaining the structural information of the target domain damaged by domain adversarial training. Specifically, some of them propose fine-tuning the model using pseudo-labelled target domain data to compensate for structural information between different classes of the target domain. However, the quality of pseudo-labels is hard to control, especially when the domain shift is serious, the false pseudo-labels will cause error accumulation, and eventually lead to negative transfer [30] of the model. Another method is to learn the structural information between the classes of the source domain in the process of model training, and use it as a regularization term to guide the representations distribution of the target domain with similar structural information. However, these works are based on the assumption that the structural information of different classes of representations of the target domain and the source domain is consistent, which is difficult to fully guarantee in complex real scenes.

To effectively address the above problem, we are inspired by the *problem reduction* method, and propose a novel approach Maintaining Structural Information of the target domain based on Pairwise semantic Similarity (whether two instances belong to the same class or not) (MSIPS). The *problem reduction* method has had a long history in the literature, especially in binarization techniques and ensemble methods [7]. Its core strategy is to transform a complex task into a different and simpler task. Specifically, in domain adaptation, the ideal way to maintain structural information between classes within the target domain is to have labeled target domain data for different classes as Fig. 1(a), which is naturally not achievable in unsupervised settings. We consider transforming the above multi-classification task that requires all class labels into a simpler binary classification task. As shown in the Fig. 1(b), given a pair of target domain instances, we indirectly maintain the structural information between all classes of the target domain by predicting whether the two instances belong to the same class. That is, if two instances are predicted to be the same class, their representation is as close as possible in the feature space, and vice versa.

A direct idea is to predict whether the two samples belong to the same category by calculating the similarity of their representations, but we cannot scientifically set a reasonable threshold for the similarity. Moreover, domain adversarial training has the risk of destroying the structural information between the original classes of the target domain, which will further reduce the accuracy of our prediction of pairwise similarity. Therefore, we introduce Contrastive Learning to obtain feature prototypes for each category of the source domain, which will explicitly model the intra-class discrepancy and the inter-class discrepancy, and predict the pairwise similarity of the target domain instances by these prototype features that are far away from each other.

In summary, the main contributions of this paper are as follows:

(1) We are the first to use pairwise similarity of the target domain to maintain structural information in unsupervised domain adaptation, which is motivated by the *problem reduction* method.
(2) Contrastive Learning is introduced to obtain the category prototypes of the source domain. We use them as Category-Prototypes Bank to provide more accurate pairwise similarity for our method.
(3) We conduct careful ablation studies on benchmark UDA datasets, which demonstrate the effectiveness of our method and show competitive performance with several state-of-the-art methods

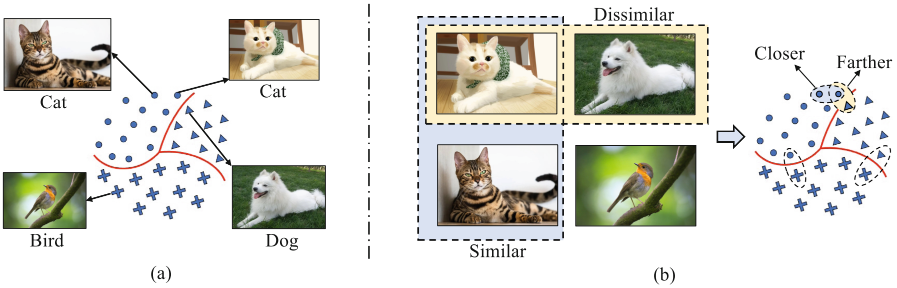


Fig. 1. Methods of maintaining structural information.

2 Related Works

2.1 Alignment Based Unsupervised Domain Adaptation

Unsupervised Domain Adaptation (UDA) generally assumes that the two domains have the same conditional distribution, but different marginal distributions (i.e., domain shift). In order to alleviate the influence of domain shift and improve the generalization ability of the trained source domain model on the target domain data, many work [19,27–29,31,33] proposed to align the representation distribution of the source domain and the target domain in the model training process. Some of them measure the degree of domain shift quantitatively by various metrics, e.g., maximum mean discrepancy (MMD) [17,18], and achieve domain-level alignment by minimizing the metrics. Another part of

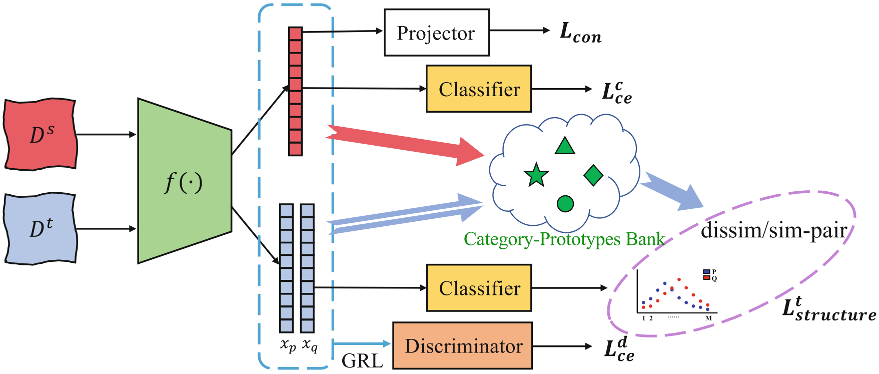


Fig. 2. Proposed method of maintaining structural information by pairwise similarity (Best viewed in color). Our method consists of a backbone, a classifier, a domain discriminator and a category prototype bank. Red and blue refer to data or representation from source domain and target domain respectively. The modules with same color have the same parameters, and GRL refers to the gradient reversal layer for domain-adversarial training. (Color figure online)

them [9,24,31,33] is to learn discriminative domain invariant features by introducing a domain discriminator for domain adversarial training. Although these methods effectively align the representation distributions of source domain and target domain, they all have the risk of damaging the structural information within the target domain, that is, the discrimination of the feature of the target domain is reduced [4]. In order to achieve class-level domain alignment, [23] introduces the class label into the domain discriminator, so that it can be aware of the classification boundary. [10] uses the pseudo-labeled target domain data to fine-tune the model to compensate for the damaged structural information between classes. [3] attempts to obtain the category centroid of different domains, forcing them to maintain consistency. Our method maintains structural information through pairwise similarity between classes inherent in the target domain, so we do not rely on the assumption that two domains have standard structural similarity.

2.2 Problem Reduction Method

Problem Reduction Method has had a long history in the literature, especially in binarization techniques and ensemble methods [1]. The most well-known strategies are “one-vs-all” [25] and “one-vs-one” [11]. The core idea of these work is to transform a more complex task into one or more simpler tasks. We are motivated by this method, and propose to maintain structural information by pairwise similarity for unsupervised domain adaptation.

3 Our Approach

Consider labeled data $\mathcal{D}^s = \{(x_i^s, y_i^s)\}_{i=1}^{N_s}$ from the source domain and unlabeled data $\mathcal{D}^t = \{(x_i^t)\}_{i=1}^{N_t}$ from the target domain, where $y^s = \{1, 2, \dots, M\}$ is M different classes, N_s and N_t is the number of source and target samples, respectively. Unsupervised domain adaptation (UDA) assumes that source domain and target domain have shared label space $\mathcal{Y}$ but large distribution gap between $P(\mathcal{D}^s)$ and $P(\mathcal{D}^t)$. Our goal is to transfer the knowledge learned from the source domain to the target domain. In this section, we introduce our MSIPS method, as shown in Fig. (2).

3.1 Category-Prototypes Guided Pairwise Similarity

Given the representation of a pair of target domain samples x_p^t and x_q^t, the traditional method to predict whether they are similar is to measure the distance, e.g., Cosine Similarity, Euclidean Distance, between the two representations through some metrics, and compare it with the threshold. But it is hard to find such a suitable threshold, let alone the risk of domain adversarial training mixing different classes of representations in the target domain. In order to solve this problem, considering that the source domain and the target domain have the same label space, we explore to find a reference point for each category in the feature space of the source domain and the target domain, which can not only be clearly distinguished from each other, but also alleviate the influence of domain shift. By comparing pairwise target domain representations with these benchmark representations, we can accurately determine whether the two samples belong to the same class. Specifically, motivated by InfoNCE [22] in Contrastive Learning, in addition to cross entropy (CE) loss, we impose a contrastive loss to make different classes of source domain representations more representative:

$$\mathcal{L}_{\text{con}} = -\log \frac{\exp\left(d\left(x^s, x^{s+}\right)/\tau\right)}{\exp\left(d\left(x^s, x^{s+}\right)/\tau\right) + \sum_{j=1}^{K-1} \exp\left(d\left(x^s, x_j^{s-}\right)/\tau\right)}, \tag{1}$$

where x^s denote any source representation output by Projector, x^{s+} is the randomly sampled positive representation with the same class label from the same batch and x_j^{s-} is the negative representation with different class labels. Moreover, $d(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature factor. That is, the overall training procedure of MSIPS for labeled source domain data can be summarized as follows:

$$\mathcal{L}_{total}^s = \mathcal{L}_{ce}^c + \lambda \mathcal{L}_{\text{con}}, \tag{2}$$

where λ is trade-off parameter to balance losses. For simplicity, we set the λ to 1 based on our preliminary experiment.

Further, we calculate the mean values of the source domain features of each category as Eq. (3), and store them as the prototype of the category into the Category Prototype Bank.

$$p_m = \frac{1}{n} \sum_{i=1}^{n} (x_i^s | y_i^s = m) \tag{3}$$

For a pair of target domain representations x_p^t and x_q^t, we measure the distance between them and M prototypes, and further predict their pseudo labels y_p^t, y_q^t by softmax operation. So far, we can get their pairwise similarity label $Y_{ps} \in \{0, 1\}$. As shown in Fig. (2), if they belong to the same category, we represent it as *sim-pair* ($Y_{ps} = 1$), and vice versa as *dissim-pair* ($Y_{ps} = 0$).

3.2 Maintaining Structural Information via Pairwise Similarity

For source domain data, class labels can be used as supervised information to guide the model to learn the classification boundaries of different classes of data. We call this discriminative information between multiple classes structural information. But in unsupervised domain adaptation, there is no labeled data in the target domain, so it is difficult to learn this structural information directly. Different from the methods [3,29] maintaining structural information consistency between two domains based on the assumption of domain closeness [2], which cannot always be fully guaranteed in real scenarios, we propose to maintain structural information via pairwise similarity. In the previous subsection, for a given pair of target domain samples x_p^t and x_q^t, we have predicted their pairwise similarity, that is, whether they belong to the same class. Note that the correct probability of such two-class classification task prediction is higher than that of multi-class classification task prediction. Therefore, we are inspired by the *problem reduction* method [14] and use pairwise similarity to maintain the structural information of the target domain. If x_p^t and x_q^t are *sim-pair*, we believe that the corresponding distributions output by the source domain classifier $\mathcal{P}$ and $\mathcal{Q}$ are similar, and KL-divergence is used to measure the distance between the two distributions. We optimize the loss function as follows:

$$\mathcal{L}_{sim\_pair} = \mathcal{D}_{KL}\left(\mathcal{P}^{\star} \| \mathcal{Q}\right) + \mathcal{D}_{KL}\left(\mathcal{Q}^{\star} \| \mathcal{P}\right), \tag{4}$$

where $\mathcal{P}^{\star}$ and $\mathcal{Q}^{\star}$ are alternatively assumed to be constant, because the function for calculating KL-divergence is asymmetric. If x_p^t and x_q^t are *dissim-pair*, we optimize a hinge-loss as Eq. (5) (where the value of margin σ refers to [13]), and expect their distributions to be different.

$$\mathcal{L}_{dissim\_pair} = L_h\left(\mathcal{D}_{KL}\left(\mathcal{P}^{\star} || \mathcal{Q}\right), \sigma\right) + L_h\left(\mathcal{D}_{KL}\left(\mathcal{Q}^{\star} \| \mathcal{P}\right), \sigma\right) \tag{5}$$

$$L_h(e, \sigma) = \max(0, \sigma - e) \tag{6}$$

Combining $\mathcal{L}_{sim\_pair}$ and $\mathcal{L}_{dissim\_pair}$, we get our structure loss. In addition, in order to further suppress the influence of pairwise similarity error caused by pseudo-label noise, we set a weight η for the structure loss based on the confidence of pseudo-label prediction as Eq. (7):

$$\eta = \sum_{m=1}^{M} P(y_p^t = m) P(y_q^t = m), \tag{7}$$

where $P(\cdot)$ is the output of softmax operation. Intuitively, if both y_p^t and y_q^t are predicted as the same class with greater confidence, the structure loss between them will be more involved in model optimization. Therefore, the total loss for maintaining structural information can be defined as follows: s

$$\mathcal{L}_{structure}^t = \eta Y_{ps} \mathcal{L}_{sim\_pair} + (1 - \eta)(1 - Y_{ps}) \mathcal{L}_{dissim\_pair}. \tag{8}$$

3.3 Combining with Domain Adversarial Training

Domain adversarial training can achieve domain-level alignment, and make the category prototype of the source domain we learn tend to be domain-invariant, thereby improving the accuracy of our prediction of pairwise similarity for the target samples. Therefore, we combine it with the proposed method. Finally, the total loss of MSIPS is as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{total}^s + \mathcal{L}_{structure}^t + \mathcal{L}_{ce}^d, \tag{9}$$

where $\mathcal{L}_{ce}^d$ is the loss of domain discriminator (a classifier that distinguishes samples from source or target domains) in domain adversarial training as [9], usually Cross Entropy loss with domain labels.

4 Experiments

4.1 Datasets

We evaluated our method in the following two standard benchmarks for UDA.

Office-31 [26] is the most popular real-world benchmark dataset for visual domain adaptation, which is made up of three distinct domains, i.e., Amazon (A), Webcam (W) and DSLR (D). It contains 4,110 images of 31 categories in three domains. We evaluated our method on six domain adaptation tasks.

Office-Home [32] is a more challenging recent dataset for UDA, which consists of 15500 images from 65 categories. There are 4 different domains in it: Art (Ar), Clip Art (Cl), Product (Pr), and Real-World (Rw). We evaluate our method in all the 12 one-source to one-target adaptation cases.

4.2 Implementation Details

Our implementation is based on [15]. Following the standard protocol for UDA, we use all labeled source data and all unlabeled target data. We use the ImageNet [6] pre-trained ResNet-50 [12] as the base network for fair comparison, where the last FC layer is replaced with the task-specific FC layer(s) to parameterize the classifier. We use mini-batch stochastic gradient descent (SGD) with a momentum of 0.9, an initial learning rate of 0.001, and a weight decay of 0.005. All reported results of mean(±std) are obtained from the average of three runs.

Table 1. Accuracy (%) on Office-31 for unsupervised domain adaptation (ResNet-50). The best accuracy is indicated in bold.

Method	A→W	D→W	W→D	A→D	D→A	W→A	Avg
DANN	82.0±0.4	96.9±0.2	99.1±0.1	79.7±0.4	68.2±0.4	67.4±0.5	82.2
CCN	78.2	97.4	98.6	73.5	62.8	60.6	78.5
CCN⋆	88.5±0.4	97.4±0.5	98.6±0.1	84.5±0.7	73.5±0.2	73.4±0.3	86.0
PFAN	83.0±0.3	99.0±0.2	99.9±0.1	76.3±0.3	63.3±0.3	60.8±0.5	80.4
MDD	94.5±0.3	98.4±0.1	**100.0±0.0**	93.5±0.2	74.6±0.3	72.2±0.1	88.9
SymNets	90.8±0.1	98.8±0.3	**100.0±0.0**	93.9±0.5	74.6±0.6	72.5±0.5	88.4
CAN	94.5±0.3	99.1±0.2	99.8±0.2	95.0±0.3	**78.0±0.3**	77.0±0.3	90.6
GVB	92.0±0.3	98.7±0.0	**100.0±0.0**	91.4±0.5	74.9±0.5	73.4±0.1	88.3
SRDC	95.7±0.2	**99.2±0.1**	**100.0±0.0**	**95.8±0.2**	76.7±0.3	77.1±0.1	90.8
MSIPS	**95.9±0.3**	98.7±0.2	**100.0±0.0**	**95.8±0.1**	76.8±0.2	**78.6±0.5**	**91.0**

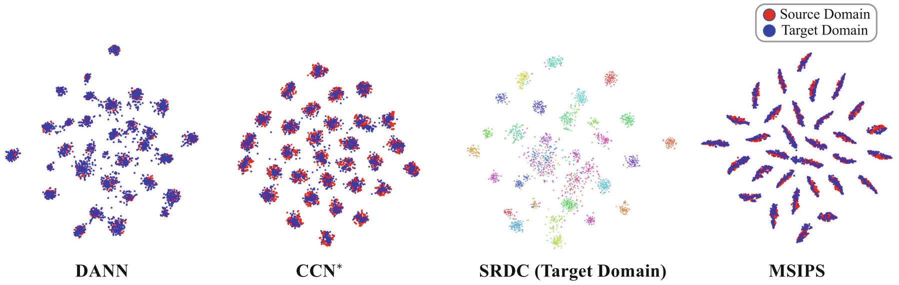
4.3 Comparison with Baselines

We select a series of popular methods as baselines to compare with MSIPS, including DANN [8] as a baseline for further analysis of our contributions and some state-of-the-art methods for performance comparisons. In particular, we compare with CCN [13] which uses pairwise similarity to perform conditional clustering on target domain data, PFAN [3] aligns the category prototypes of the target domain and source domain directly, SRDC [29] uses the structural information of source domain contained in source domain labels to guide target domain data clustering. Note that most of them are domain adaptation methods based on alignment. In addition, bridging theory based MDD [35], alignment based CAN [16], GVB [5] and SymNets [34] are also selected as baselines.

Results on **Office-31** are reported in Table 1, where results of existing methods are quoted from their respective papers or the works. CCN uses imageNet as an auxiliary dataset to learn a model for predicting pairwise similarity. On this basis, we fine-tune the model using all data outside the target data to further improve its accuracy in predicting pairwise similarity. The results are recorded as CCN⋆. We can see that MSIPS outperforms all compared methods on most of the transfer tasks, and the average performance reaches SOTA. We utilize t-SNE [20] to visualize embedded features on the source domain and the target domain by Source Model from DANN, CCN⋆, SRDC (only target domain representations) and our method. As shown in Fig. (3), we can find that the source domain features and target domain features learned by MSIPS have the characteristics of close distance within the class and long distance between classes. At the same time, the same classes of features from different domains are clustered, which fully proves that our method effectively maintains the structural information of the target domain. The results of **Office-Home** are reported in Table 2. Although the average performance of our method is slightly lower than SRDC (−0.1%), it reaches SOTA in multiple domain adaptation tasks. Especially in Ar→Cl, our method exceeds SRDC by 7.7 %, which indicates that our method is promising.

Table 2. Accuracy (%) on office-home for unsupervised domain adaptation (ResNet-50). The best accuracy is indicated in bold.

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
DANN	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
SymNets	47.7	72.9	78.5	64.2	71.3	74.2	64.2	48.8	79.5	74.5	52.6	82.7	67.6
MDD	54.9	73.7	77.8	60.0	71.4	71.8	61.2	53.6	78.1	72.5	60.2	82.3	68.1
GVB	57.0	74.7	79.8	64.6	74.1	74.6	65.2	55.1	81.0	74.6	59.7	84.3	70.4
SRDC	52.3	**76.3**	81.0	**69.5**	76.2	**78.0**	**68.7**	53.8	**81.7**	**76.3**	57.1	**85.0**	**71.3**
MSIPS	**60.0**	75.4	**81.1**	65.7	**76.4**	75.1	65.1	**57.7**	**81.7**	72.4	**61.1**	82.7	71.2



DANN
CCN*
SRDC (Target Domain)
MSIPS

Fig. 3. The t-SNE visualization of embedded features on the task W→A (Best viewed in color). Note that for SRDC we refer to the visualization results in the original paper, in which different classes are denoted by different colors.

4.4 Ablation Studies and Discussions

For a more detailed analysis of our proposed method, we conducted ablation studies on the Office-31 dataset, and all experiments are aimed at adaptation task W→A.

Comparison of Accuracy of Predicting Pairwise Similarity. Using pairwise similarity to constrain the structure between classes has a certain history in Conditional Clustering. CCN [13] tried to use this constraint to cluster the target domain samples under cross-domain settings, but showed poor performance. The reason we analyze the failure is that although pairwise similarity has a good effect on maintaining structural information, the accuracy of predicting pairwise similarity will be severely reduced by domain shift. Instead of using the auxiliary data set to train the pairwise similarity prediction network directly, we use the data of the non-target domain from the current dataset to fine-tune the network, and further improve its accuracy in predicting the pairwise similarity (i.e., CCN*). Even so, we can observe from Fig. (4) that our method is more accurate in predicting pairwise similarity, which fully demonstrates the effectiveness of our method for predicting pairwise similarity based on category prototypes.

Effect and Error Analysis of Maintaining Structural Information. As shown in Fig. (5), our method effectively maintains the structural information of the source domain and the target domain, different classes of features are distributed in different clusters, and features with the same classes are closer

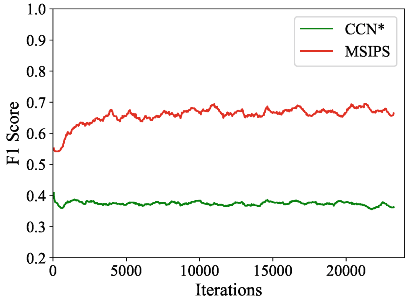


Fig. 4. F1 Score of pairwise similarity prediction (Best viewed in color).

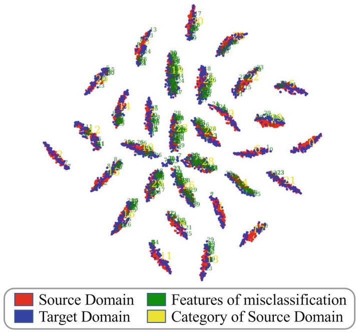


Fig. 5. The visualization of features of misclassification (Best viewed in color).

in the feature space. However, we find that although our method has higher accuracy in predicting pairwise similarity and has adopted some strategies to suppress the impact of similarity prediction errors, there will still be some sample classification errors. We claim that further improving the accuracy of pairwise similarity prediction helps to improve the effect of domain adaptation.

5 Conclusion

In this paper, we propose a method based on pairwise similarity to maintain the structural information of the target domain for unsupervised domain adaptation. Different from the existing method of directly aligning the category structure of source domain and target domain based on domain closeness assumption, our method aims to make full use of the inherent information of whether different samples in the target domain belong to the same category, and effectively maintain the structural information of the target domain through the idea of problem reduction. The experimental results show that our method has achieved comparable performance with SOTA, and a large number of visualization results fully demonstrate the effectiveness of our method. Finally, we analyze the errors in the ablation studies and propose further research directions for improvement.

References

1. Allwein, E.L., Schapire, R.E., Singer, Y.: Reducing multiclass to binary: a unifying approach for margin classifiers. J. Mach. Learn. Res. **1**, 113–141 (2000)
2. Ben-David, S., Blitzer, J., et al.: A theory of learning from different domains. Mach. Learn. **79**(1), 151–175 (2010). https://doi.org/10.1007/s10994-009-5152-4
3. Chen, C., et al.: Progressive feature alignment for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 627–636 (2019)

4. Chen, X., Wang, S., Long, M., Wang, J.: Transferability vs. discriminability: batch spectral penalization for adversarial domain adaptation. In: International Conference on Machine Learning, pp. 1081–1090. PMLR (2019)
5. Cui, S., Wang, S., Zhuo, J., Su, C., Huang, Q., Tian, Q.: Gradually vanishing bridge for adversarial domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12455–12464 (2020)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
7. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: An overview of ensemble methods for binary classifiers in multi-class problems: experimental study on one-vs-one and one-vs-all schemes. Pattern Recogn. **44**(8), 1761–1776 (2011)
8. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. In: International Conference on Machine Learning, pp. 1180–1189. PMLR (2015)
9. Ganin, Y., et al.: Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**(1), 2030–2096 (2016)
10. Gu, X., Sun, J., Xu, Z.: Spherical space domain adaptation with robust pseudo-label loss. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9101–9110 (2020)
11. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. In: Advances in Neural Information Processing Systems, vol. 10 (1997)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
13. Hsu, Y.C., Lv, Z., Kira, Z.: Learning to cluster in order to transfer across domains and tasks. In: International Conference on Learning Representations (2018)
14. Hsu, Y.C., Lv, Z., Schlosser, J., Odom, P., Kira, Z.: Multi-class classification without multi-class labels. In: International Conference on Learning Representations (2018)
15. Jiang, J., Chen, B., Bo, F., Long, M.: Transfer-learning-library. https://github.com/thuml/Transfer-Learning-Library (2020)
16. Kang, G., Jiang, L., Yang, Y., Hauptmann, A.G.: Contrastive adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4893–4902 (2019)
17. Long, M., Cao, Y., Wang, J., Jordan, M.: Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning, pp. 97–105. PMLR (2015)
18. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Unsupervised domain adaptation with residual transfer networks. In: Advances in Neural Information Processing Systems, vol. 29 (2016)
19. Long, M., Zhu, H., Wang, J., Jordan, M.I.: Deep transfer learning with joint adaptation networks. In: International Conference on Machine Learning, pp. 2208–2217. PMLR (2017)
20. Van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. J. Mach. Learn. Res. **9**(11), 2579–2605 (2008)
21. Na, J., Jung, H., Chang, H.J., Hwang, W.: Fixbi: bridging domain spaces for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1094–1103 (2021)
22. Oord, A.V.D., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint. arXiv:1807.03748 (2018)

23. Pei, Z., Cao, Z., Long, M., Wang, J.: Multi-adversarial domain adaptation. In: 32nd AAAI Conference on Artificial Intelligence (2018)
24. Pinheiro, P.O.: Unsupervised domain adaptation with similarity learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8004–8013 (2018)
25. Rifkin, R., Klautau, A.: In defense of one-vs-all classification. J. Mach. Learn. Res. **5**, 101–141 (2004)
26. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010. LNCS, vol. 6314, pp. 213–226. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_16
27. Sener, O., Song, H.O., Saxena, A., Savarese, S.: Learning transferrable representations for unsupervised domain adaptation. In: Advances in Neural Information Processing Systems, vol. 29 (2016)
28. Sun, B., Feng, J., Saenko, K.: Correlation alignment for unsupervised domain adaptation. In: Csurka, G. (ed.) Domain Adaptation in Computer Vision Applications. ACVPR, pp. 153–171. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58347-1_8
29. Tang, H., Chen, K., Jia, K.: Unsupervised domain adaptation via structurally regularized deep clustering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8725–8735 (2020)
30. Torrey, L., Shavlik, J.: Transfer learning. In: Handbook of Research on Machine Learning Applications and Trends: Algorithms Methods, and Techniques, pp. 242–264. IGI global, Hershey (2010)
31. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7167–7176 (2017)
32. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5018–5027 (2017)
33. Zhang, W., Ouyang, W., Li, W., Xu, D.: Collaborative and adversarial network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3801–3809 (2018)
34. Zhang, Y., Tang, H., Jia, K., Tan, M.: Domain-symmetric networks for adversarial domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5031–5040 (2019)
35. Zhang, Y., Liu, T., Long, M., Jordan, M.: Bridging theory and algorithm for domain adaptation. In: International Conference on Machine Learning, pp. 7404–7413. PMLR (2019)