XDC: Adversarial Adaptive Cross Domain Face Clustering (Student Abstract)

Saed Rezayi[1], **Handong Zhao**[2], **Sheng Li**[1]

[1]University of Georgia
[2]Adobe Research
{saedr, sheng.li}@uga.edu, hazhao@adobe.com

Abstract

In this work we propose a scheme, called XDC, that uses adversarial learning to train an adaptive cross domain clustering model. XDC trains a classifier on a labeled dataset and assigns labels to an unlabeled dataset. We benefit from adversarial learning such that the target dataset takes part in the training. We also use an existing image classifiers in a plug-and-play fashion (*i.e.*, it can be replaced with any other image classifier). Unlike existing works we update the parameters of the encoder and expose the target dataset to the model during training. We apply our model on two face dataset and one non-face dataset and obtain comparable results with state-of-the-art face clustering models.

Introduction

Clustering, in general, is challenging as it is an unsupervised tasks and labeling information is not available during training. To tackle this challenge, proposed clustering schemes make prior assumptions about type or shape of the data distribution (Lloyd 1982), number of clusters (Van Gansbeke et al. 2020), or number of samples in each clusters (Shi and Malik 2000). Domain Adaptation (DA) is another promising approach to clustering where a model is trained on a labeled dataset (source dataset) and the trained model is used to find clusters of the unlabeled dataset (target dataset) (Wang et al. 2019). The shortcoming of this approach is that they only rely on source dataset to train a clustering model and this is problematic when there is a large shift between source and target datasets.

The motivation behind our model is simple yet effective; we claim that the target dataset contains useful information about its underlying distribution, and its exposure to the model during training can be used to develop more effective features which are both discriminative and invariant to the change of domains. This has been shown to be advantageous in many application and we demonstrate its effectiveness in cross domain image clustering. Our contributions are threefold: we set up a problem setting, cross domain clustering, in which we integrate domain adaptation and link prediction for the task of clustering. We propose XDC which modifies face clustering methods by incorporating adversarial domain adaptation into the learning process to reduce the domain shift between source and target datasets. We apply XDC on multiple benchmark datasets and show its generalizability by considering both face and non-face image datasets.

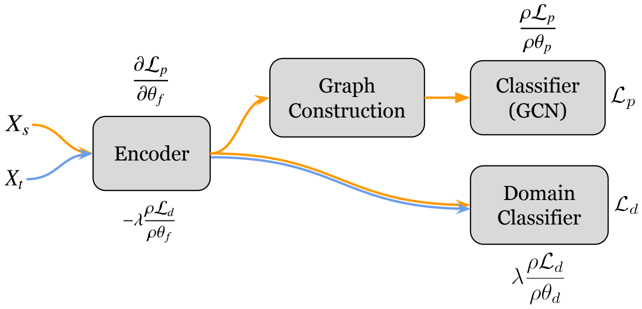
$\frac{\rho\mathcal{L}_p}{\rho\theta_p}$
$\frac{\partial\mathcal{L}_p}{\partial\theta_f}$
X_s
X_t
Encoder
Graph Construction
Classifier (GCN)
$\mathcal{L}_p$
Domain Classifier
$\mathcal{L}_d$
$-\lambda\frac{\rho\mathcal{L}_d}{\rho\theta_f}$
$\lambda\frac{\rho\mathcal{L}_d}{\rho\theta_d}$

Figure 1: The training module of our proposed model. We used resnet-50 as the encoder and graph convolutional networks (Kipf and Welling 2016) as image classifier. Once the training is complete, the parameters of the encoder and the GCN classifier are saved/fixed and used in the test phase to obtain clustering assignment on the target dataset.

Methodology

In this section we first set up the problem definition and introduce the notations used in the paper, and then discuss the details of different components of our model.

Notation and Problem Definition – We define cross domain clustering as follows: Let $\mathcal{D}^s$ be the source domain consisting of a feature space $\mathcal{X}^s$, a label space $\mathcal{Y}^s$, and a conditional distribution $P(Y^s|X^s)$. Every sample drawn from this distribution can be shown as $\{x_i^s, y_i^s\}$ where $x_i^s \in \mathcal{X}^s$ and $y_i^s \in \mathcal{Y}^s$. Similarly, we assume $\mathcal{D}^t$ is the target domain consisting of feature space $\mathcal{X}^t$ and the data distribution $P(X^t)$ which indicates label space is not available for target dataset. We also define domain label, d_i, as a binary variable for i-th sample, which indicates whether x_i is drawn from $\mathcal{X}^s$ or $\mathcal{X}^t$. Cross domain clustering, in general, aims to learn $P(Y|X)$ on labeled source dataset and assign a pseudo label $\hat{y}_i^t$ to each unlabeled target data point, x_i^t.

Proposed Model – We approach this problem in an adversarial fashion. Our idea is to expose the target dataset to the model during training and use the adversarial loss introduced in (Ganin et al. 2016) to update the features. This helps to learn features that are more discriminative between the source and target datasets. We also propose to use a Graph Convolutional Networks (GCN) (Kipf and Welling 2016) to train a classifier on labeled source dataset. The graph generation process is similar to what proposed in (Wang et al. 2019). Our proposed model is different from (Wang et al. 2019) as we participate the feature extraction module in the training to learn more effective features, and additionally we expose the model to the target dataset in an adversarial fashion to learn more discriminative features. The proposed method is illustrated in Figure 1.

As Figure 1 shows, both source and target datasets go through feature extractor module and the feature vectors of source dataset are used to construct graphs, as described in (Wang et al. 2019). The constructed graphs are then fed into a GCN module and parameters θ_p and θ_f are updated using the supervised loss function $\mathcal{L}_p$. The bottom path is the domain classifier where feature vectors of source and target datasets are inputted into a fully connected neural network and parameters θ_d and θ_f are updated using the supervised loss function $\mathcal{L}_d$.

More formally, adversarial domain adaptation aims at minimizing the following loss function:

$$E(\theta_f, \theta_p, \theta_d) = \sum_{\substack{i=1..N \\ d_i=0}} L_y^i(\theta_f, \theta_p) - \lambda \sum_{i=1..N} L_d^i(\theta_f, \theta_d) \tag{1}$$

where $L_y(.,.)$ is the loss for label classifier and $L_d(.,.)$ is the loss for domain classifier. We can expand the mathematical term inside the first summation, $L_y^i(\theta_f, \theta_p)$, to the following equation:

$$L_y^i(G_p(G_f(x_i^s; \theta_f); \theta_p), y_i^s) \tag{2}$$

where G_p is the label classifier and G_f is the encoder. In our proposed model we replace G_p with a GCN model which is formulated as:

$$G_p = \text{GCN}(\mathbf{A}, \mathbf{F}) = \sigma\left([\mathbf{F}||\text{agg}(A, \mathbf{F})]\theta_p\right) \tag{3}$$

where σ is the sigmoid function, $||$ is the concatenation operation, and agg is an aggregation function such as mean aggregation or attention aggregation (Veličković et al. 2018). Additionally, $\mathbf{A}$ is the adjacency matrix of the constructed graph generated by feature vectors f after each batch and $\mathbf{F}$ is the feature matrix obtained from the following equation:

$$\mathbf{F} = G_f(x_i^s; \theta_f) \tag{4}$$

Once the GCN model is trained we can save its parameters and use them in the test phase. Since we require to obtain features for the target dataset as well, we fix the weights of the encoder and use them in the test phase. In fact, we use the trained G_p and G_f to learn new features from target dataset and perform linkage clustering assignment as explained in (Wang et al. 2019).

Method	source	IJB-B-512	VGG-50	CIFAR10
	target	LFW	IJB-B-512	STL10
k-means		0.68	0.61	0.19
ARO (Otto, Wang, and Jain 2017)		0.87	0.76	-
ConPac (Shi, Otto, and Jain 2018)		0.92	0.65	-
IPS (Wang et al. 2019)		0.90	**0.83**	0.25
XDC [our model]		**0.95**	**0.83**	**0.42**

Table 1: F-1 score for clustering task across five different baselines and three different settings

Experiment

In this section we use benchmark datasets to find perform unsupervised clustering on face datasets. We consider Several datasets, including LFW, IJB-B-512, and STL10, and compare our model (XDC) with state-of-the-art baseline in face clustering.

Results – Table 1 presents the results for the four baselines and our model (last row) for three different settings. In the first setting we use IJB-B-512 dataset as the source dataset and LFW as the target dataset. In this case we outperform state-of-the-art with 5 percent, and in the second scenario where VGG-50 is the source dataset and IJB-B-512 is the target ddataset we obtain comparable results with IPS.

In the last column of the Table 1 we present the clustering performance for non-image datasets, *i.e.*, CIFAR-10 and STL-10. The goal of this exercise is to generalize our model to other domains. As this column shows our model outperforms all existing models by a very large margin ($\sim 17\%$).

References

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *UMLR*, 17(1): 2096–2030.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *ICLR*.

Lloyd, S. 1982. Least squares quantization in PCM. *IEEE transactions on information theory*, 28(2): 129–137.

Otto, C.; Wang, D.; and Jain, A. K. 2017. Clustering millions of faces by identity. *IEEE transactions on pattern analysis and machine intelligence*, 40(2): 289–303.

Shi, J.; and Malik, J. 2000. Normalized cuts and image segmentation. *TPAMI*, 22(8): 888–905.

Shi, Y.; Otto, C.; and Jain, A. K. 2018. Face clustering: representation and pairwise constraints. *IEEE Transactions on Information Forensics and Security*, 13(7): 1626–1640.

Van Gansbeke, W.; Vandenhende, S.; Georgoulis, S.; Proesmans, M.; and Van Gool, L. 2020. Scan: Learning to classify images without labels. In *ECCV*, 268–285. Springer.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Wang, Z.; Zheng, L.; Li, Y.; and Wang, S. 2019. Linkage based face clustering via graph convolution network. In *CVPR*, 1117–1125.