Data Efficient Domain Adaptation

Presentation for IKDD Uplink Research Internship Program, 2022

Anindya Mondal
anindyam.jan@gmail.com

Jadavpur University
*Work done while interning at the **Indian Institute of Science**

Aug 27, 2022

Train a classifier on the training data and directly apply it to the test data (**Same Domain**)

Domain Adaptation





A classifier trained on one domain may perform poorly on another domain (**Different Domain**)

- Domain Adaptation solves the issue of disparity across domains.

- Domain Adaptation solves the issue of disparity across domains.
- Achieved in various ways.

Domain Adaptation

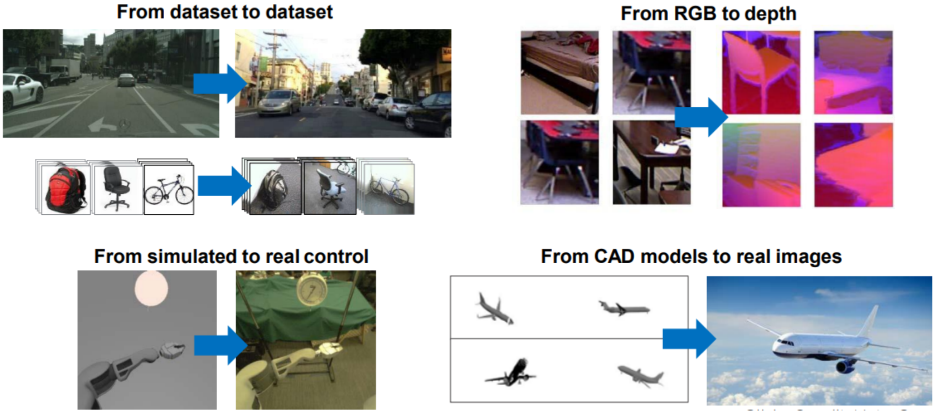


Figure: Different DA applications

Depending upon the nature of supervision, DA can be broadly divided into 3 types:

- **Supervised DA**: Having labelled data in both source and target.

Depending upon the nature of supervision, DA can be broadly divided into 3 types:

- **Supervised DA**: Having labelled data in both source and target.
- **Unsupervised DA**: Having labelled data in the source only.

Depending upon the nature of supervision, DA can be broadly divided into 3 types:

- **Supervised DA**: Having labelled data in both source and target.
- **Unsupervised DA**: Having labelled data in the source only.
- **Semi-supervised DA**: Having some labelled target data, but not enough to train from scratch.

Various DA Approaches

Source Data

Target Data

Fully Labelled	Supervised DA	Fully Labeled
Fully Labelled	Unsupervised DA	No Labels
Fully Labelled	Semi-supervised DA	Few Labels

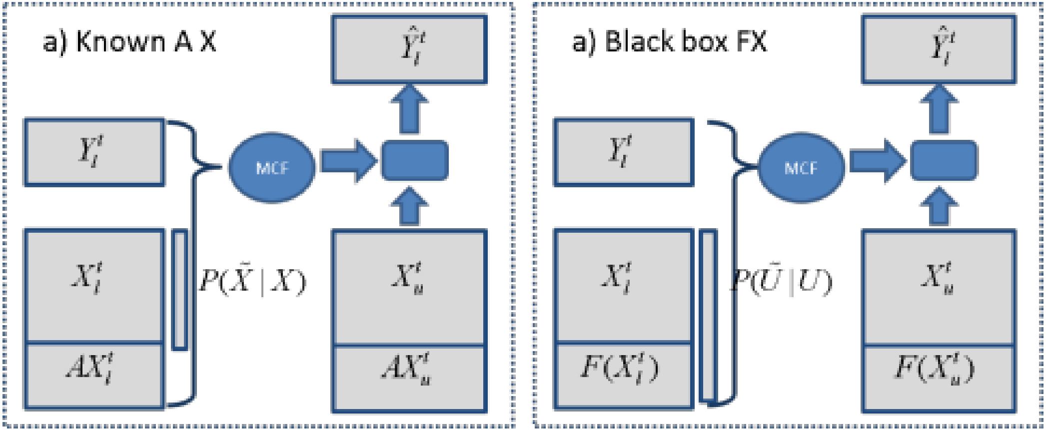
- Unsupervised DA assumes that data distribution from the labelled source data and the unlabeled target data is related but different.
- Assumes that the samples from both domains are freely available during the training process.

- This consideration often raises privacy concerns and is subject to legal constraints.

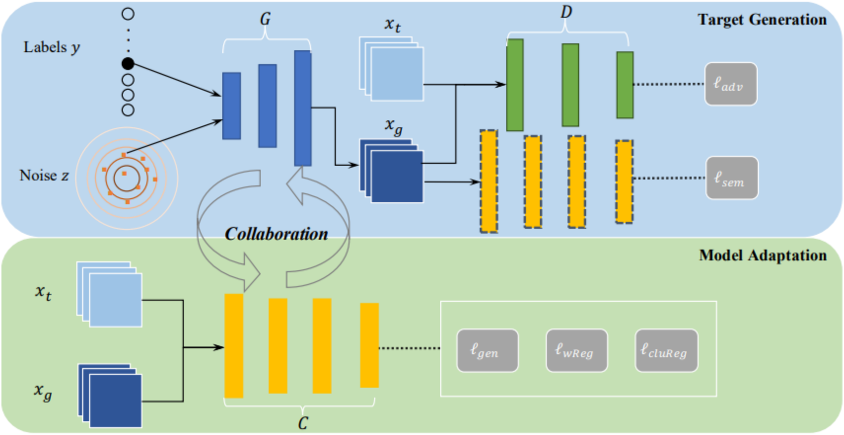
- This consideration often raises privacy concerns and is subject to legal constraints.
- Training on entire source data is not a feasible solution, especially if the labelled source data is large scale.

- Source Free DA is a newly introduced approach that tackles the requirement of access to the source data.

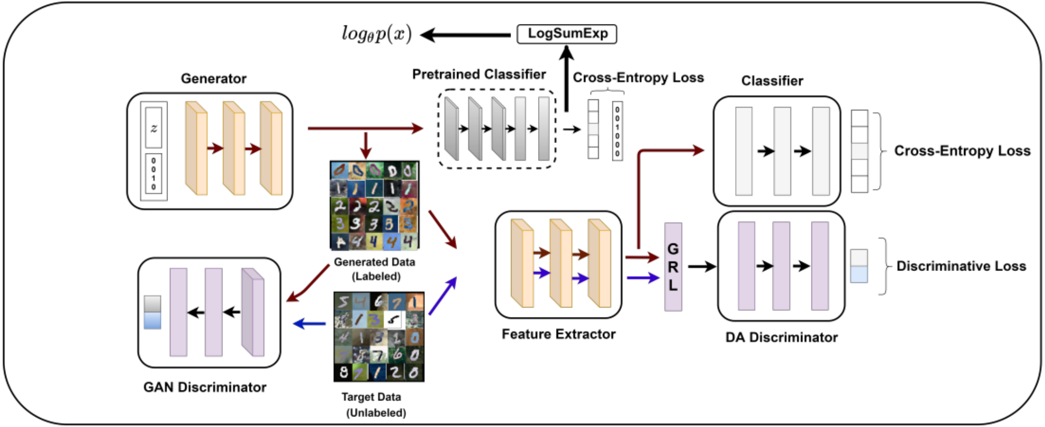
- Source Free DA is a newly introduced approach that tackles the requirement of access to the source data.
- The source pre-trained model is adapted to the target domain without source data.



Chidlovskii *et al.* [KDD'16] first introduced Source Free DA in both supervised and unsupervised settings using Marginalized Corrupted Features (MCF) and Marginalized Denoising Autoencoder (MDA) framework.

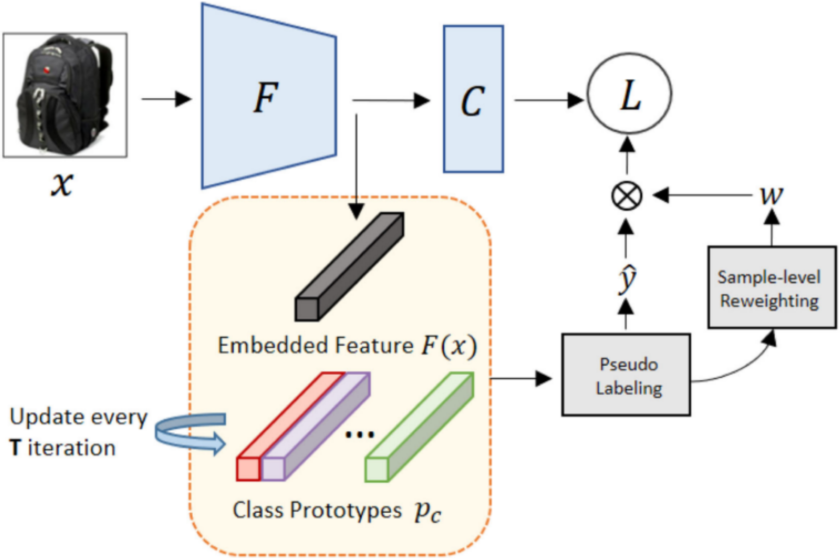


Li *et al.* [CVPR '20] proposed 3CGAN for producing target-style training samples, where the generator and the prediction model can be collaboratively enhanced during adaptation.

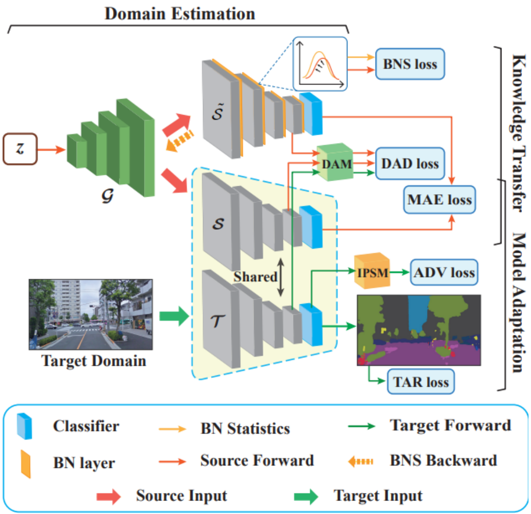


Kurmi *et al.* [ICCV '21] developed a generative framework where the source-data trained classifier generates samples from the source classes without directly accessing the source data.

Previous Source Free DA approaches (Contd..)



Kim *et al.* [IEEE SPL '20] proposed a pseudo-labelling approach from the source pre-trained model.



Liu *et al.* [CVPR '21] first proposed an SFDA method for semantic segmentation by combining knowledge transfer and model adaptation without requiring any source data and target labels.

- Study two latest Source Free DA approaches:

- Study two latest Source Free DA approaches:
- Yang *et. al.* [NeurIPS '21] exploits the intrinsic neighbourhood structure of the target data.

Studied Source Free DA approaches

- Study two latest Source Free DA approaches:
- Yang *et. al.* [NeurIPS '21] exploits the intrinsic neighbourhood structure of the target data.
- Huang *et. al.* [NeurIPS '21] aims to adapt source-trained models to fit target data distribution without accessing the source-domain data by remembering the source hypothesis.

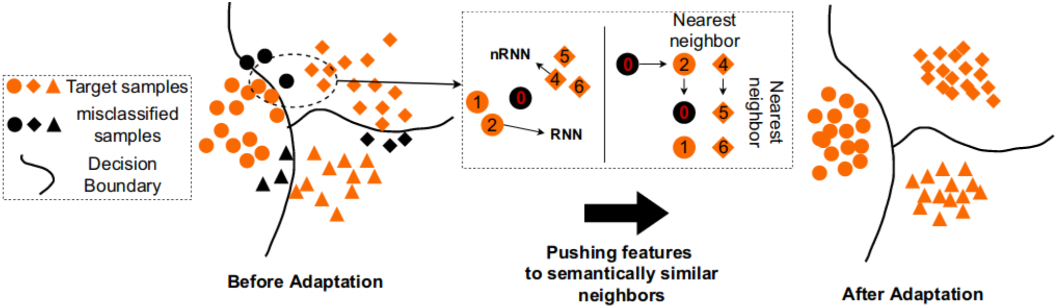


Figure: Schematic diagram of NRC SFDA

- Method exploits the intrinsic neighbourhood structure of the target data.

- Method exploits the intrinsic neighbourhood structure of the target data.
- Even though there may be a shift of the target data in the feature space due to covariance shift, target data belonging to the same class will form a cluster in the embedding space.

- Method exploits the intrinsic neighbourhood structure of the target data.
- Even though there may be a shift of the target data in the feature space due to covariance shift, target data belonging to the same class will form a cluster in the embedding space.
- All nearest neighbours may belong to the same class.

- Method exploits the intrinsic neighbourhood structure of the target data.
- Even though there may be a shift of the target data in the feature space due to covariance shift, target data belonging to the same class will form a cluster in the embedding space.
- All nearest neighbours may belong to the same class.
- Classify the neighbours into two classes: Reciprocal nearest neighbours (RNN) and Non-reciprocal nearest neighbours (nRNN).

NRC SFDA (Yang *et. al.*)

- Method exploits the intrinsic neighbourhood structure of the target data.
- Even though there may be a shift of the target data in the feature space due to covariance shift, target data belonging to the same class will form a cluster in the embedding space.
- All nearest neighbours may belong to the same class.
- Classify the neighbours into two classes: Reciprocal nearest neighbours (RNN) and Non-reciprocal nearest neighbours (nRNN).
- Assigned differential weights to the supervision of nearest neighbours, developing Neighbourhood Reciprocity Clustering (NRC).

- Method exploits the intrinsic neighbourhood structure of the target data.
- Even though there may be a shift of the target data in the feature space due to covariance shift, target data belonging to the same class will form a cluster in the embedding space.
- All nearest neighbours may belong to the same class.
- Classify the neighbours into two classes: Reciprocal nearest neighbours (RNN) and Non-reciprocal nearest neighbours (nRNN).
- Assigned differential weights to the supervision of nearest neighbours, developing Neighbourhood Reciprocity Clustering (NRC).
- nRNNs assigned weaker connections, RNNs assigned stronger connections with higher affinity scores.

- Adaptation objective achieved by:

$$\mathcal{L} = -\frac{1}{n_t} \sum_{x_i \in \mathcal{D}_t} \sum_{x_j \in \mathrm{Neigh}(x_i)} \frac{D_{sim}(p_i, p_j)}{D_{dis}(x_i, x_j)}$$

  Push the data towards their semantically close neighbours by encouraging similar predictions.

- Adaptation objective achieved by:

$$\mathcal{L} = -\frac{1}{n_t} \sum_{x_i \in \mathcal{D}_t} \sum_{x_j \in \mathrm{Neigh}(x_i)} \frac{D_{sim}(p_i, p_j)}{D_{dis}(x_i, x_j)}$$

  Push the data towards their semantically close neighbours by encouraging similar predictions.

- The final adaptation is obtained by:

$$\mathcal{L} = \mathcal{L}_{div} + \mathcal{L}_{\mathcal{N}} + \mathcal{L}_{E} + \mathcal{L}_{self}.$$

- Calculate adaptation accuracy for **object recognition**.

- Calculate adaptation accuracy for **object recognition**.
- Use the **Office-Home** [CVPR '17] dataset

- Calculate adaptation accuracy for **object recognition**.
- Use the **Office-Home** [CVPR '17] dataset
- Dataset contains images from 4 domains: Clipart, Art, Product and Real World.

- Calculate adaptation accuracy for **object recognition**.
- Use the **Office-Home** [CVPR '17] dataset
- Dataset contains images from 4 domains: Clipart, Art, Product and Real World.
- Contain images of objects commonly used in office and home settings.

- Calculate adaptation accuracy for **object recognition**.
- Use the **Office-Home** [CVPR '17] dataset
- Dataset contains images from 4 domains: Clipart, Art, Product and Real World.
- Contain images of objects commonly used in office and home settings.
- Perform inter-domain adaptation and calculate adaptation accuracy.

- Perform ablation study of the different neighbours and affinity values.

- Perform ablation study of the different neighbours and affinity values.
- Vary the neighbourhood values.

Table 1: Ablation study on the different components of NRC SFDA algorithm. L_{div}**: prediction diversity loss,** L_N**: nearest-neighbours prediction loss,** L_E**: expanded neighbourhood prediction loss,** L_{E^-}**: expanded neighbourhood prediction loss (removing duplication),** A**: affinity value. + denotes the presence of that component.**

L_{div}	L_N	L_E	L_{E^-}	A	Average Accuracy												
					Ar-Cl	Ar-Rw	Ar-Pdt	Cl-Ar	Cl-Pdt	Cl-Rw	Ptd-Ar	Pdt-Cl	Pdt-Rw	Rw-Ar	Rw-Cl	Rw-Pdt	Avg
					44.39	**69.28**	**64.72**	**56.61**	**65.56**	**55.37**	**53.63**	**41.58**	**71.35**	**60.94**	**43.85**	**75.65**	**58.60**
+					**49.32**	**70.58**	**66.65**	**58.75**	**66.78**	**59.41**	**56.30**	**47.28**	**75.63**	**59.97**	**47.12**	**80.21**	**61.50**
+	+				**50.00**	**80.65**	**78.36**	**61.58**	**75.24**	**74.19**	**58.62**	**49.10**	**74.11**	**68.21**	**53.10**	**80.60**	**66.98**
+	+			+	**54.79**	**85.27**	**80.36**	**69.38**	**76.51**	**79.58**	**66.87**	**54.48**	**64.97**	**70.65**	**63.90**	**85.00**	**70.98**
+	+	+			**46.68**	**78.54**	**73.72**	**60.01**	**72.84**	**72.69**	**56.42**	**44.58**	**69.25**	**65.89**	**54.69**	**79.89**	**64.60**
+	+	+		+	**57.60**	**81.78**	**80.17**	**66.98**	**80.71**	**77.98**	**64.97**	**55.79**	**81.69**	**71.53**	**57.68**	**85.44**	**71.86**
+	+		+	+	**51.58**	**78.36**	**75.37**	**59.48**	**72.58**	**72.00**	**56.59**	**49.69**	**71.38**	**65.51**	**54.24**	**78.28**	**65.43**

Table 2: Scores under different neighbourhood values (NRC SFDA).

K = 3, M = 2	**71.86**
Without E (K = 9)	**64.65**

- Reduce the number of samples per class while performing the adaptation.

Table: Adaptation accuracy under the different samples per category.

% samples per category	Ar-Clip	Ar-Rw	Ar-Pdt	Clip-Ar	Clip-Pdt	Clip-Rw	Pdt-Ar	Pdt-Clip	Pdt-Rw	Rw-Ar	Rw-Clip	Rw-Pdt	Average
100	56.98	81.97	80.19	67.97	79.56	78.50	64.86	55.49	82.75	70.86	57.97	85.45	**71.88**
75	47.78	73.95	72.73	52.86	63.88	65.22	52.41	43.77	69.11	58.32	45.33	69.66	**59.59**
50	36.22	59.41	53.57	40.51	43.77	49.62	44.53	29.93	51.92	48.99	34.90	51.85	**45.44**
25	22.34	48.29	34.52	27.89	38.78	31.34	26.82	19.69	41.81	35.11	24.77	45.23	**33.04**

- Performs well when all neighbours are present.

- Performs well when all neighbours are present.
- Presence of noisy expanded neighbours can be detrimental to the model's efficiency

- Performs well when all neighbours are present.
- Presence of noisy expanded neighbours can be detrimental to the model's efficiency
- Absence of E (expanded neighbours, neighbours of K nearest neighbours of a feature vector) degrade the model's performance, signifying its importance.

- Performs well when all neighbours are present.
- Presence of noisy expanded neighbours can be detrimental to the model's efficiency
- Absence of E (expanded neighbours, neighbours of K nearest neighbours of a feature vector) degrade the model's performance, signifying its importance.
- Decreasing samples show a significant loss in accuracy.

- Performs well when all neighbours are present.
- Presence of noisy expanded neighbours can be detrimental to the model's efficiency
- Absence of E (expanded neighbours, neighbours of K nearest neighbours of a feature vector) degrade the model's performance, signifying its importance.
- Decreasing samples show a significant loss in accuracy.
- Less data means fewer representative features from each class.

- Redefining the different neighbours.

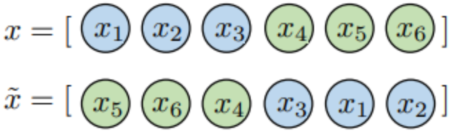
NRC SFDA (Scope of improvement)

- Redefining the different neighbours.
- Modifying the weights assigned to different neighbours.

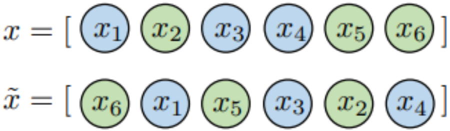
- Redefining the different neighbours.
- Modifying the weights assigned to different neighbours.
- Augmenting the available samples to compensate for the unavailability of sufficient samples using MixStyle.

NRC SFDA (Scope of improvement)

- Redefining the different neighbours.
- Modifying the weights assigned to different neighbours.
- Augmenting the available samples to compensate for the unavailability of sufficient samples using MixStyle.
- Testing the method on other datasets.



(a) Shuffling batch w/ domain label



(b) Shuffling batch w/ random shuffle

Reference batch generation by MixStyle

- Boris Chidlovskii, Stephane Clinchant, and Gabriela Csurka. 2016. Domain adaptation in the absence of source domain data. In Proceedings of the 22nd ACM KDD.
- Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. 2017. Deep hashing network for unsupervised domain adaptation. In Proceedings of the IEEE/CVF CVPR.
- Shiqi Yang, Joost van de Weijer, Luis Herranz, Shangling Jui, et al. 2021. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. Advances in NeurIPS.
- Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. 2021. Model adaptation: historical contrastive learning for unsupervised domain adaptation without source data. Advances in NeurIPS.
- Zhou, Kaiyang, Yongxin Yang, Yu Qiao, and Tao Xiang. "Domain Generalization with MixStyle.", ICLR '21

Thanks to IACV Lab, IISc for the opportunity and the computational resources.

THANK YOU