Hochschule
Bonn-Rhein-Sieg
University of Applied Sciences

Bonn-Aachen
International Center for
Information Technology

Evaluation of Deep Neural Network Domain Adaptation Techniques for Image Recognition

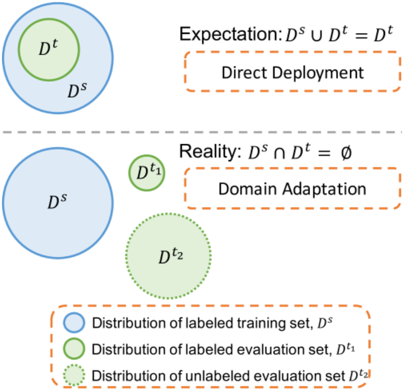
Advanced Machine Learning

April 20, 2020

Venkata Santosh Sai Ramireddy Muthireddy
Alan Preciado Grijalva

Introduction

- Deep learning algorithms assume that training and testing data is drawn from independent and identical distributions (i.i.d.).
- This assumption rarely holds true.
- Reducing the domain shift between distribution of datasets is called **domain adaptation**



Types of domain adaptation

Different types of domain adaptation are:

- Unsupervised domain adaptation
- Semi supervised domain adaptation
- Weakly supervised domain adaptation
- Few shot learning

Further Unsupervised domain adaptation based on technique used:

- Adversarial Methods
- Distance-based Methods
- Incremental Methods
- Optimal Transport
- Other Methods

Selected methods

We have chosen four different unsupervised domain adaptation techniques for image classification tasks

- Deep coral:Correlation alignment for deep domain adaptation. (DeepCORAL)[2]
- Deep domain confusion: Max-imizing for domain invariance. (DDC)[3]
- Conditional adversarial domain adaptation. (CDAN)[4]
- CDAN with Entropy conditioning. (CDAN+E) [4]

Dataset: Office-31

- Office-31 dataset contains a total of 4652 images collected from three different sources.
- Images are collected are online web (amazon), digital SLR (dslr) camera and webcam. Office-31 dataset have 31 classes [1] in total.
- **Images from the web:** These are collected from amazon website. We call this dataset as amazon dataset (A). It has around 2800 images and with an average of 90 images per each class. The resolution of the images are each 300x300 pixels.
- **Images from a dslr camera:** These are captured from dslr camera in real environments. We call this dataset as dslr dataset (D). It has total of 498 images and with an average of 16 images per each class. The resolution of the images are 1000x1000 pixels.
- **Images from a webcam:** These are collected from a webcam with lower resolution and will have more noise. From now this is called as webcam dataset (W). It has around 795 images and with an average of 25 images per each class. The resolution of the images is HxW where H and W are varying between 400 and 600.

[1] The 31 classes are: backpack, bike, bike helmet, bookcase, bottle, calculator, desk chair, desk lamp, computer, file cabinet, headphones, keyboard, laptop, letter tray, mobile phone, monitor, mouse, mug, notebook, pen, phone, printer, projector, puncher, ring binder, ruler, scissors, speaker, stapler, tape, and trash can.

Dataset: Office-31

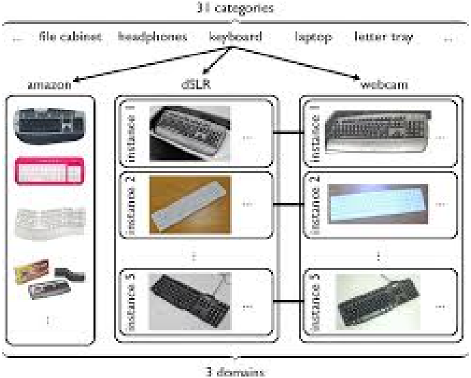


Figure 2: Sample of keyboard category images for all the three domains[5]

DeepCORAL

- Domain adaptation method based on Correlation Alignment (CORAL) [2]
- CORAL attempts to increase performance in domain shift by aligning the second order statistics between both data distributions.
- DeepCORAL extends the capacities of CORAL by learning a non-linear mapping using convolutional neural networks and a differentiable loss function

$$l_{CORAL} = \frac{1}{4d^2} \|C_S - C_T\|_F^2 \tag{1}$$

$$C_S = \frac{1}{n_S - 1}\left(D_S^T D_S - \frac{1}{n_S}(1^T D_S)^T (1^T D_S)\right) \tag{2}$$

$$C_T = \frac{1}{n_T - 1}\left(D_T^T D_T - \frac{1}{n_T}(1^T D_T)^T (1^T D_T)\right) \tag{3}$$

$$l = l_{CLASS} + \sum_{i=1}^{t} \lambda_i l_{CORAL} \tag{4}$$

DeepCORAL

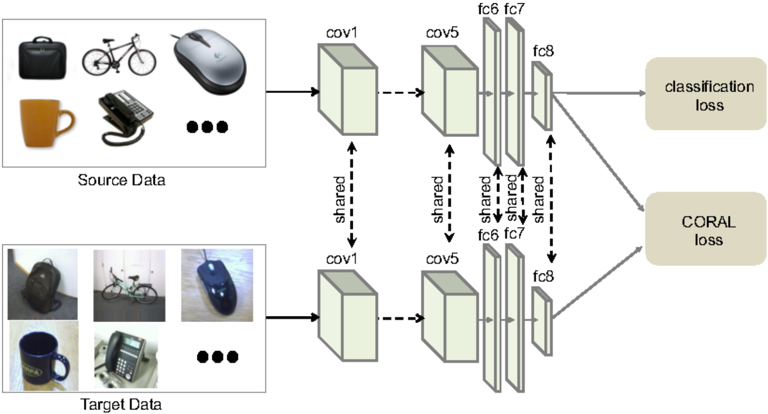


Figure 3: DeepCORAL neural network architecture.

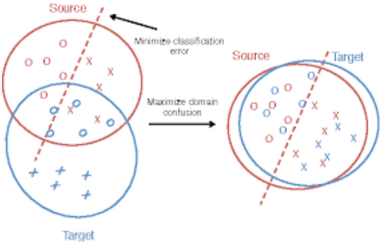
DDC

- Deep domain confusion is another domain adaptation method that uses a deep neural network to learn a non-linear transformation [3]
- Distance between data distributions is minimized using a standard metric called Maximum Mean Discrepancy (MMD)
- DDC intializes its weights with a pre-trained AlexNet and adds bottleneck layer for regularization

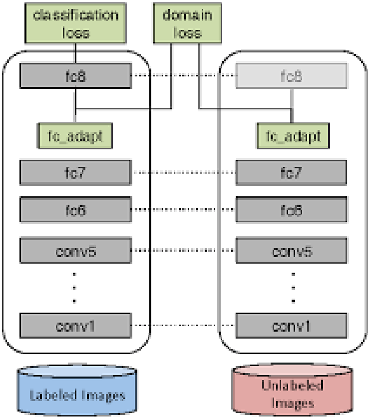
$$MMD(X_S,X_T)=\left| \frac{1}{\|X_S\|}\sum \phi(x_s)-\frac{1}{\|X_T\|}\sum \phi(x_t)\right| \tag{5}$$

$$l=l_C(X_L,y)+\lambda MMD^2(X_S,X_T) \tag{6}$$

DDC



(a) DDC approach: Maximizing domain confusion can lead to good performance in target domains.



(b) DDC neural network architecture.

CDAN

- Adversarial domain adaptation method.
- Solves two issues
  - can handle datasets which have complex multi modal distributions.
  - can handle uncertain information difference between source and target data.
-

$$E(G) = \mathbb{E}_{(x_i^s, y_i^s) \sim D_s} L(G(x_i^s), y_i^s) \tag{7}$$

$$E(D, G) = -\mathbb{E}_{x_i^s \sim D_s} log[D(f_i^s, g_i^s)] - \mathbb{E}_{x_i^t \sim D_t} log[D(f_j^t, g_j^t)] \tag{8}$$

where L(.) is the cross entropy loss. Now CDAN is described in minmax optimization problem as

$$minG \; E(G) - \lambda E(D, G) \; and \; minG \; E(D, G) \tag{9}$$

where λ is a hyper-parameter that balances the domain adversary and source risk.

CDAN

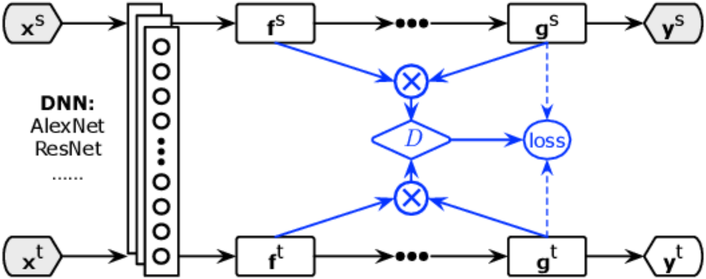


Figure 5: Architecture of CDAN for domain adaptation.[4]

CDAN+E

- CDAN+E is an variant of CDAN.
- Additional involvement of **Entropy Conditioning**.
- So uncertainty corresponding to classifier predictions are quantified using entropy criteria explained in Eq. 10.

$$H(g) = -\sum_{c=1}^{C} g_c log\ g_c \tag{10}$$

where C is number of classes and g_c is probability of predictions.
Now training examples are prioritized by an entropy-aware weight given by

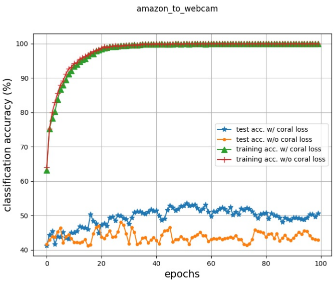
$$w(H(g)) = 1 + e^{-H(g)} \tag{11}$$

Results

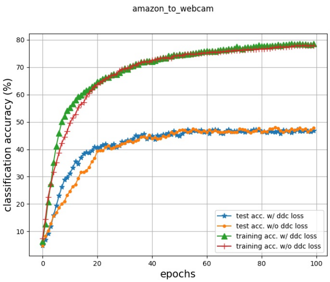
Testing accuracy for different domain adaptation techniques across different datasets.

	A → W	A → D	W → A	W → D	D → A	D → W
No Adaptation	43.1±2.5	49.2±3.7	35.6±0.6	94.2±3.1	35.4±0.7	90.9±2.4
DeepCORAL	49.5±2.7	40.0±3.3	38.3±0.4	74.4±4.3	38.5±1.5	89.1±4.4
DDC	41.7±9.1	—	—	—	—	—
CDAN	44.9±3.3	49.5±4.6	34.8±2.4	93.3±3.4	32.9±2.4	88.3±3.8
CDAN+E	48.7±7.5	53.7±4.7	35.3±2.7	93.6±3.4	33.9±2.2	87.7±4.0

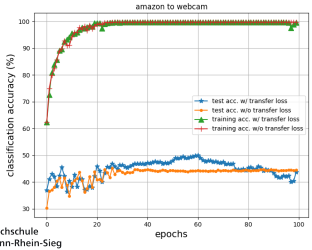
Results



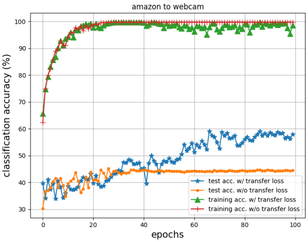
(a) DeepCORAL



(b) DDC

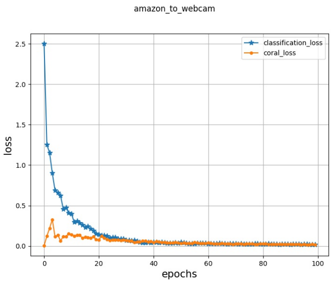


(c) CDAN

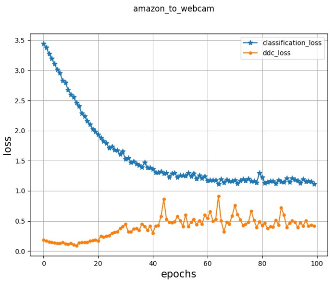


(d) CDAN+E

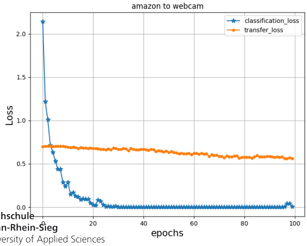
Results

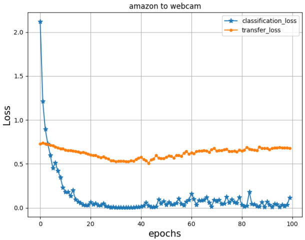


(a) DeepCORAL



(b) DDC





Conclusions

- Proof of concept of domain adaptation: improved accuracy performance using domain adaptation losses
- Accuracy can actually decrease or remain the same for other shifts (room for improvement)
- Hyperparameter tunning: carry out trainings for 200 epochs (not 100), understand better the behavior of our losses
- We gained hands-on experience building end-to-end pipelines using deep neural networks for domain adaptation tasks

References

1. X. Xu, X. Zhou, R. Venkatesan, G. Swaminathan, and O. Majumder,“d-sne: Domain adaptation using stochastic neighborhood embedding,”in2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019, pp. 2492–2501.
2. B. Sun and K. Saenko, “Deep coral: Correlation alignment for deepdomain adaptation,” in Computer Vision – ECCV 2016 Workshops,G. Hua and H. J egou, Eds.Cham: Springer International Publishing,2016, pp. 443–45.
3. E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, “Deepdomain confusion: Maximizing for domain invariance,”arXiv preprint arXiv:1412.3474, 2014.
4. M. Long, Z. CAO, J. Wang, and M. I. Jordan, “Conditional adversarialdomain adaptation,” inAdvances in Neural Information Processing Systems 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman,N. CesaBianchi, and R. Garnett, Eds.Curran Associates, Inc.,2018, pp. 1640–1650.
5. K. Saenko, B. Kulis, M. Fritz, and T. Darrell, “Adapting visual categorymodels to new domains,” inComputer Vision – ECCV 2010, K. Dani-ilidis, P. Maragos, and N. Paragios, Eds. Berlin, Heidelberg: SpringerBerlin Heidelberg, 2010, pp. 213–226.

Thank You