# Multi-step domain adaptation by adversarial attack to $\mathcal{H}\Delta\mathcal{H}$-divergence

**Arip Asadulaev** [1 2] **Alexander Panfilov** [3] **Andrey Filchenkov** [1]

## Abstract

Adversarial examples are transferable between different models. In our paper, we propose to use this property for multi-step domain adaptation. In unsupervised domain adaptation settings, we demonstrate that replacing the source domain with adversarial examples to $\mathcal{H}\Delta\mathcal{H}$-divergence can improve source classifier accuracy on the target domain. Our method can be connected to most domain adaptation techniques. We conducted a range of experiments and achieved improvement in accuracy on Digits and Office-Home datasets.

## 1. Introduction

In domain adaptation, we train a model to make correct predictions on the different domains named source and target (Ben-David et al., 2010a). Domain adaptation techniques aim to bring the target domain "closer'' to the source domain or find a common representation for these domains. (Ganin & Lempitsky, 2015; Long et al., 2018).

Theoretically, the accuracy of domain adaptation is bounded by the source domain error and the divergence $\mathcal{H}\Delta\mathcal{H}$ between source and target domains (Ben-David et al., 2010a;b; Germain et al., 2013). The divergence $\mathcal{H}\Delta\mathcal{H}$ can be approximated by the binary classifier between source and target domains. We find that source class samples turned into the target class sample (by an adversarial attack to the $\mathcal{H}\Delta\mathcal{H}$ neural approximation) are actually "closer" to the real target samples. We show that training the source classifier on these samples can improve its accuracy in the target domain.

We call our method $\mathcal{H}\Delta\mathcal{H}$-divergence Domain Adaptation (HDA). We tested a range of domain adaptation algorithms coupled with the proposed technique on datasets "Digits" (LeCun & Cortes, 2010) and "Office-Home" (Venkateswara et al., 2017a).

[1]ITMO University, Saint-Petersburg, Russia [2]Artificial Intelligence Research Institute, Moscow, Russia [3]Tubingen Universitat, Germany. Correspondence to: Arip Asadulaev <aripasadulaev@itmo.ru>.

*Proceedings of the $39^{th}$ International Conference on Machine Learning*, Baltimore, Maryland, USA, PMLR 162, 2022. 

## 2. Background

**Adversarial Attacks:** Having sample $x$, target label $y$, model parameters $\theta$, loss function $L$, we can apply adversarial attack (Szegedy et al., 2014; Goodfellow et al., 2015; Moosavi-Dezfooli et al., 2016; Madry et al., 2018; Luo et al., 2018). For example, Fast Sign Gradient Descent (FSGD) (Goodfellow et al., 2015) can be presented as: $x_{i+1} = x_i + \varepsilon \operatorname{sign}\left(-\nabla_x L\left(\theta, x_i, y_i\right)\right)$, with perturbation size $\varepsilon$ around the original image $x_i$. It was shown that adversarial examples are transferable between different models (Petrov & Hospedales, 2019). Recent studies have revealed that the adversarial examples are largely invariant to the models trained on different sampled datasets (Papernot et al., 2016).

**Domain Adaptation:** For the given two domains, source $S_X$ and target $T_X$ over the input space $X$, the goal is to bring the target domain or its representation "closer'' to the source domain. Following the learning theory (Kifer et al., 2004; Ben-David et al., 2010a;b), the domain adaptation error is bounded by the source domain classifier error and the $\mathcal{H}\Delta\mathcal{H}$ discrepancy between source and target domains. It is impossible to estimate exactly the value of $\mathcal{H}\Delta\mathcal{H}$, but it can be approximated by a binary classifier that discriminates the source and the target samples (Ben-David et al., 2010a;b).

## 3. Algorithm

First of all, we train a binary classifier $\mathcal{H}_\omega$ between domains $S_X$ and $T_X$ that approximates $\mathcal{H}\Delta\mathcal{H}$. Then, we apply perturbations in the input space of $\mathcal{H}_\omega$ to make source samples $S_X$ look like the target domain samples $T_X$ for $\mathcal{H}_\omega$ and get a new domain $A_X$. In the third step, we pretrain the source classifier $\mathcal{F}_\theta$ on the resulted domain $A_X$, instead of the real source $S_X$. Due to the adversarial attacks' transferability, $A_X$ adversarial examples are transferable to the $\mathcal{F}_\theta$ classifier too, and actually, increase $\mathcal{F}_\theta$ accuracy on the target domain. Finally, we apply domain adaptation for the source classifier $\mathcal{F}_\theta$, using the arbitrary distance-based domain adaptation techniques (Ganin & Lempitsky, 2015; Long et al., 2018; 2015; 2017), but also replacing the real source data $S_X$ with the $A_X$ in the training pipeline. As a result, we obtain a multi-step domain adaptation pipeline (Wang & Deng, 2018), where at the first step, the algorithm creates a new dataset that is used by the second step method.

| MODEL | MNIST→MNIST-M | MNIST-M→MNIST | SVHN→MNIST | MNIST→SVHN | USPS→MNIST |
|---|---|---|---|---|---|
| DANN | 97.2 +- 1.37 | **73.4 +- 1.71** | 21.3 +- 2.38 | 67.3 +- 3.20 | 97.2 +- 1.37 |
| CDAN | 97.8 +- 0.22 | 61.0 +- 2.76 | 15.7 +- 4.55 | 61.5 +- 5.14 | 97.8 +- 0.22 |
| CDAN-E | 97.8 +- 0.41 | 67.0 +- 4.32 | 13.9 +- 1.47 | 60.5 +- 4.44 | 97.8 +- 0.41 |
| DAN | 97.6 +- 1.17 | 43.6 +- 1.93 | 19.2 +- 2.97 | 61.2 +- 2.27 | 97.6 +- 1.17 |
| JAN | 97.6 +- 0.39 | 44.7 +- 4.48 | 10.7 +- 1.70 | 59.9 +- 3.14 | 97.6 +- 0.39 |
| SHOT | 98.1 +- 1.22 | 77.4 +- 1.44 | 24.4 +- 1.50 | 98.9 +- 2.11 | 98.0 +- 0.51 |
| DANN + HDA | **98.1 +- 0.25** | 70.7 +- 2.86 | **24.5 +- 3.47** | **78.6 +- 7.99** | **98.1 +- 0.25** |
| CDAN + HDA | **98.0 +- 0.41** | **65.2 +- 3.69** | **17.2 +- 2.24** | **70.8 +- 8.29** | **98.0 +- 0.41** |
| CDAN-E + HDA | **98.2 +- 0.03** | **69.9 +- 2.28** | **16.0 +- 1.46** | **72.8 +- 3.95** | **98.2 +- 0.03** |
| DAN + HDA | **97.6 +- 0.15** | **44.6 +- 1.24** | **23.2 +- 2.91** | 60.7 +- 3.34 | 97.6 +- 0.15 |
| JAN + HDA | 97.1 +- 0.16 | **49.8 +- 6.47** | **13.6 +- 2.18** | 59.3 +- 10.56 | 97.1 +- 0.16 |
| SHOT + HDA | **98.2 +- 0.45** | **77.6 +- 1.91** | **42.6 +- 3.21** | 98.9 +- 1.72 | 98.0 +- 0.40 |

Table 1: Results of domain adaptation with HDA on Digits datasets.

| MODEL | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| DAN | 43.6 | 57.0 | 67.9 | 45.8 | 56.5 | 60.4 | 44.0 | 43.6 | 67.7 | 63.1 | 51.5 | 74.3 |
| CDAN-E | 50.7 | 70.6 | 76.0 | 57.6 | 70.0 | 70.0 | 57.4 | 50.9 | 77.3 | 70.9 | 56.7 | 81.6 |
| SHOT | 57.1 | 78.1 | 81.5 | 68.0 | 78.2 | 78.1 | 67.4 | 54.9 | 82.2 | 73.3 | 58.8 | 84.3 |
| SHOT+HDA | 57.0 | **78.5** | 80.6 | **69.1** | **79.5** | **78.4** | **68.3** | **55.5** | 81.5 | **73.7** | **60.5** | 83.8 |

Table 2: Results on Office-Home datasets. Domains are: Art (A), Clipart (C), Product (P), Real-World (R)

## 4. Experiments

**Datasets:** All experiments were conducted in unsupervised settings (only source labels are known) on the range of digits domains (MNIST (LeCun & Cortes, 2010), USPS (Hull, 1994), MNIST-M, SVHN) and Office-Home (Venkateswara et al., 2017b) domains.

**Settings:** In our experiments, we used $A_X$ instead of the real source $S_X$ for prominent adversarial-based approach's DANN (Ganin & Lempitsky, 2015), CDAN, CDAN-E (Long et al., 2018). Also, we tested our method on Maximum Mean Discrepancy (MMD) (Gretton et al., 2012) based on domain adaptation techniques like DAN (Long et al., 2015) and JAN (Long et al., 2017). In addition, we tested our method in connection to the SHOT (Liang et al., 2020) method on the Office-Home domains. We used ADA (Tousch & Renaudin, 2020) library for training and testing, and all training settings were set equal to the default parameters proposed by ADA.

The $\mathcal{H}_\omega$ domains classifier was trained 5 epochs using Adam optimizer (Kingma & Ba, 2015) with a $0.01$ learning rate. For $A_X$ dataset generation $l_inf$ FSGD attack with perturbation size $\varepsilon$ equal to $0.01$ and 7 steps was used.

Before applying domain adaptation, we pretrained the $\mathcal{F}_\theta$ classifier by 10 epochs on the $A_X$ dataset. Then, each domain adaptation method used 20 epochs of training using the target domain samples. For parameters updating during the domain adaptation, we used the Adam optimizer with a learning rate equal to $0.01$. All hyperparameters were equal for each tested domain adaptation method and each pair of source and target domains.

**Results:** Averaged results over three random seeds for the best-selected hyperparameters for each adaptation task are presented in Table 1,2. Across all benchmarks, the training with a $A_x$-adversarial domain makes reasonable gains over the basic settings. We find that different domains require different sizes of perturbations to generate transferable examples. Searching adversarial attack parameters more carefully individually for each task can increase adaptation accuracy.

## 5. Conclusion and future works

Our paper is still a work in progress, but we propose a simple remedy to improve the accuracy of domain adaptation methods. The proposed method allows minimizing the $\mathcal{H}\Delta\mathcal{H}$-distance between the source and target domains more successfully. By combining our method with domain adaptation techniques, we hope it may probably result in the development of less complicated and more efficient domain adaptation techniques. In the future, we plan to test our method on other models and other datasets, and also we are going to increase the $\mathcal{H}\Delta\mathcal{H}$-divergence attack features transferability via linearization of backpropagation (Guo et al., 2020) and adversarial robustness training for robust features transfer. In addition, we are planning to test our method with more advanced techniques for adversarial attacks.

## References


Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Mach. Learn.*, 79(1-2):151–175, 2010a. doi: 10.1007/s10994-009-5152-4. URL https://doi.org/10.1007/s10994-009-5152-4.

Ben-David, S., Lu, T., Luu, T., and Pál, D. Impossibility theorems for domain adaptation. In Teh, Y. W. and Titterington, D. M. (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pp. 129–136. JMLR.org, 2010b. URL http://proceedings.mlr.press/v9/david10a.html.

Ganin, Y. and Lempitsky, V. S. Unsupervised domain adaptation by backpropagation. In Bach, F. R. and Blei, D. M. (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 1180–1189. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/ganin15.html.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A pac-bayesian approach for domain adaptation with specialization to linear classifiers. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 738–746. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/germain13.html.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. J. A kernel two-sample test. *J. Mach. Learn. Res.*, 13:723–773, 2012. URL http://dl.acm.org/citation.cfm?id=2188410.

Guo, Y., Li, Q., and Chen, H. Backpropagating linearly improves transferability of adversarial examples. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/00e26af6ac3b1c1c49d7c3d79c60d000-Abstract.html.

Hull, J. J. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. doi: 10.1109/34.291440.

Kifer, D., Ben-David, S., and Gehrke, J. Detecting change in data streams. In Nascimento, M. A., Özsu, M. T., Kossmann, D., Miller, R. J., Blakeley, J. A., and Schiefer, K. B. (eds.), *(e)Proceedings of the Thirtieth International Conference on Very Large Data Bases, VLDB 2004, Toronto, Canada, August 31 - September 3 2004*, pp. 180–191. Morgan Kaufmann, 2004. doi: 10.1016/B978-012088469-8.50019-X. URL http://www.vldb.org/conf/2004/RS5P1.PDF.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Liang, J., Hu, D., and Feng, J. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6028–6039. PMLR, 2020. URL http://proceedings.mlr.press/v119/liang20a.html.

Long, M., Cao, Y., Wang, J., and Jordan, M. I. Learning transferable features with deep adaptation networks. In Bach, F. R. and Blei, D. M. (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 97–105. JMLR.org, 2015. URL http://proceedings.mlr.press/v37/long15.html.

Long, M., Zhu, H., Wang, J., and Jordan, M. I. Deep transfer learning with joint adaptation networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2208–2217. PMLR, 2017. URL http://proceedings.mlr.press/v70/long17a.html.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. Conditional adversarial domain adaptation. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 1647–1657, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/ab88b15733f543179858600245108dd8-Abstract.html.

Luo, B., Liu, Y., Wei, L., and Xu, Q. Towards imperceptible and robust adversarial example attacks against neural networks. In McIlraith, S. A. and Weinberger, K. Q. (eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 1652–1659. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16217.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Moosavi-Dezfooli, S., Fawzi, A., and Frossard, P. Deepfool: A simple and accurate method to fool deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2574–2582. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.282. URL https://doi.org/10.1109/CVPR.2016.282.

Papernot, N., McDaniel, P. D., and Goodfellow, I. J. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *CoRR*, abs/1605.07277, 2016. URL http://arxiv.org/abs/1605.07277.

Petrov, D. and Hospedales, T. M. Measuring the transferability of adversarial examples. *CoRR*, abs/1907.06291, 2019. URL http://arxiv.org/abs/1907.06291.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. Intriguing properties of neural networks. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

Tousch, A.-M. and Renaudin, C. (yet) another domain adaptation library, 2020. URL https://github.com/criteo-research/pytorch-ada.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5018–5027, 2017a.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5018–5027, 2017b.

Wang, M. and Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018. doi: 10.1016/j.neucom.2018.05.083. URL https://doi.org/10.1016/j.neucom.2018.05.083.