DiscoNet: Towards Mitigating Shortcut Learning with Cross-Domain Regularization

Kyle Otstot
School of Computing & Augmented Intelligence
Arizona State University
Tempe, US
kotstot@asu.edu

John Kevin Cava
School of Computing & Augmented Intelligence
Arizona State University
Tempe, US
jcava@asu.edu

I. Problem Definition

In the task of image classification, deep learning (DL) methods have achieved impressive results. Specifically, these methods rely on learning a DL model that best estimates the true posterior distribution $P(Y|X)$, where (X,Y) is a feature-label pair of random variables drawn from an underlying joint distribution $P_{X,Y}$. This can be accomplished by finding a posterior that maximizes the expected probability of the true class, as shown below:

$$\hat{P} := \text{argmax}_P \mathbb{E}_{x,y\sim X,Y} P(y|x). \tag{1}$$

Using a train set sampled from $P_{X,Y}$, DL methods learn a posterior that not only classifies the features accurately, but also generalizes its classification ability to unforeseen examples drawn from the train-set generating distribution. This objective is encapsulated in the i.i.d. assumptions, where the train and test sets are expected to be independent and identically distributed. With a restricted DL architecture, the model is forced to learn the inter-class semantics instead of merely memorizing the labeling of the train data. However, when the test data is drawn from a different distribution $Q_{X,Y} \neq P_{X,Y}$, problems begin to arise in our task formulation. The "domain shift" from P to Q may be subtle– including artificial corruptions or adversarial nudges– or a more complex one leveraging a degree of domain expertise. For example, a classifier trained on handwritten digits may fail to generalize to a test set of digits found on traffic signs. Since the problem of domain shift is especially prevalent in real world systems, it is important for deployed classifiers to be robust in the face of any reasonable shift in the image data. In the following sections, we will elaborate on domain shifts, as well as discuss several specific "domain adaptation" problems.

II. Challenges & Main Contributions

Advances in domain adaptation require both a modeling of the problem and development of a solution; at times, the former proves to be more difficult than the latter. Historically, practitioners have evaluated their robust methods on synthetically-corrupted test sets: for example, the benchmark CIFAR-10 and CIFAR-100 datasets have prompted the creation of CIFAR-10-C and CIFAR-100-C, the same image sets altered by 15 different "real-world" corruptions, including blur, pixelation, and weather categories. Using slight corruptions to the original image set has proven to be effective at subverting classifiers trained on the original data; in [2], the classification error of a model trained on ImageNet jumped from 24% to 76% after testing on ImageNet-C. Similarly, adversarial attack algorithms, such as fast-sign gradient descent (FSGD) and projected gradient descent (PGD), can slightly alter images in a way that clearly preserves the semantics, yet successfully flips the model's classification. On the other hand, recent work has attempted to model real-world domain shifts, including the WILDS dataset [6] – a collection of dataset pairs that reflect a particular real-world shift in the data. Although the collection of real-world data shift is expensive, it is still valuable because synthetically-corrupted datasets are not suitable in capturing all kinds of potential domain shifts.

Another notable example of a real-world domain shift is called *shortcut learning*, the phenomenon where a model "cheats" on the train set by learning spurious cues instead of properly embedding the comprehensive, human-esque set of class representations. When these cues are present in both the train and test data, the practitioner may not even notice: after all, the model appears to be performing up to its best potential. However, if the spurious cues are unique to the train data, the model– to everyone's befuddlement– may experience a sharp drop in validation accuracy: all because the seemingly-meaningless features in the train data are suddenly not present in out-of-distribution samples. An example of this resides in the medical industry, where a model studying chest X-rays may disproportionately diagnose pacemaker patients with congestive heart failure [4].

To put this notion more formally, we claim that given the opportunity, a model will learn a classifier f_S that effectively minimizes the expected loss according to the source distribution P_S, but this classifier may differ vastly in structure from the loss minimizer of the target distribution $f_T(\neq f_S)$, or even more broadly, the loss minimizer of the source-target mixture distribution $f_{S\cup T}$. When attempting to learn the target domain's posterior distribution from prior knowledge of the source domain, we are generally required to establish a correspondence between the two domain's respective class semantics; for instance, when domain adaptation methods choose to leverage the classifier f_S as a compact representation of the source distribution, it is of great importance that f_S properly identifies and embeds the image semantics mutually present in each domain. However, with spurious cues in the source data, the learned classifier may behave fundamentally different than expected on the target data, thus subverting the promise of a common ground between distributions. Moving in this direction, we will show in our work that even state-of-the-art (SoTA) methods for domain adaptation (DA) can be misled by the simplest examples of shortcut learning. In doing so, we make the following contributions: (1) we introduce *Striped-MNIST*, a novel benchmark dataset for evaluating DA methods against shortcut learning; (2) we propose our own solution to Striped-MNIST (and potentially other DA settings) called *DiscoNet*, a method that leverages the DiscoGAN [5] architecture to establish a cross-domain mapping between the two datasets, which serves good use for regularizing the classifier trained on the labeled source data; (3) we provide a comparative analysis of DiscoNet with two SoTA DA methods, which are shown to perform poorly on one (or many) variations of the Striped-MNIST task.

III. Related Work & Shortcomings

In light of model vulnerability to domain shifts, recent work has argued for a variety of techniques to help combat this threat, including data augmentation, adversarial learning, and entropy minimization. On one hand, data augmentation techniques have been shown to increase the diversity of the train set, thus subjecting the model to an artificially-broader distribution. For example, *Mixup* creates new train examples by synthesizing pairs of original examples via convex combination [10], and *AugMix* enhances the training set with mixtures of three augmented images, each generated by its own chain of stochastically-sampled operations (e.g., translation, rotation) [3]. However, one main critique of data augmentation is that it relies on domain knowledge to identify and exploit transformations in the pixel space that supposedly preserve the semantics; consequently, the techniques are inherently restricted to a subset of domains where these assumptions hold true. On the other hand, adversarial approaches have been crafted to robustify classifiers against "worst-case-scenario" corruptions– for example, DL models can learn to accurately classify PGD-attacked train images alongside the original ones [7]. Moving beyond the simplistic nature of PGD attacks, *unsupervised domain adaptation* (UDA) trains with an adversarial component to learn a feature-extracting CNN that projects the source and target images into a shared feature distribution [1]; ideally, the feature extractor learns to identify and encode the semantic representations present in both domains, which are then applied to downstream classification tasks. Lastly, taking a different perspective, the *test entropy minimization* (TENT) method allows for the model to train unregularized on the source data, then tweaks the batch normalization parameters of the model to minimize the entropy with the target data [9].

Out of these methods, we are particularly interested in the latter two (UDA + TENT) as they have achieved SoTA results on benchmark DA tasks, and are fueled by strategies independent of domain knowledge, thus encouraging a rather "universal" framework for domain adaptation. Overall, we observe that the two methods adapt well to conventional source-target domain pairs (e.g., SVHN to MNIST [9]) because the model either learns the domain-invariant feature representations (UDA), or uses its prior embedding of the source data to confidently adjust the target classifications (TENT). However, the UDA algorithm provides no guarantee (entropy, accuracy, etc.) on the classification output of the target features, and the TENT algorithm is particularly reliant on the initial state of the trained source classifier; *because of these reasons, we hypothesize that such volatile strategies may be exploited by shortcut learning*. As a result, we move to the formulation of our shortcut learning benchmark, Striped-MNIST, as well as the proposal of our solution, DiscoNet.

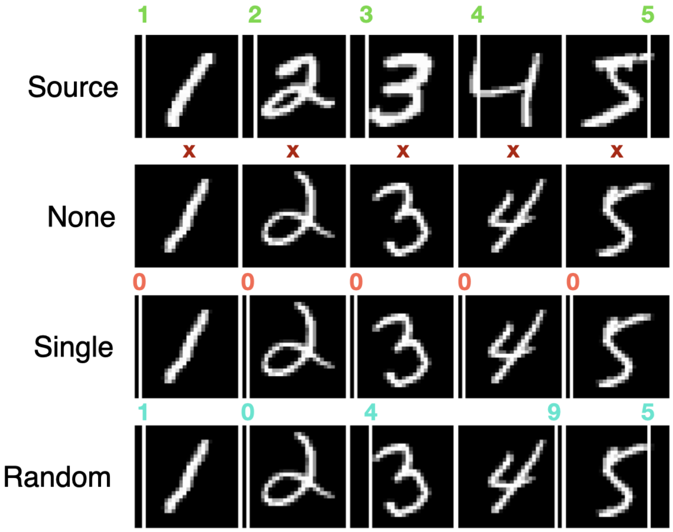

Fig. 1. Examples of Striped-MNIST. The first row is fixed as the source domain, while the next three are each considered as the target domain.

IV. Methodology

In this section, we provide an overview of our two main contributions– the shortcut learning dataset and solution.

A. *Striped-MNIST*

Training on the conventional MNIST hand-written digit dataset, it is nowadays common to receive a 99%+ test accuracy with a CNN; however, achieving such an accuracy may still take a considerable amount of iterations, which leaves the model potentially vulnerable to even simpler "shortcuts" toward class separation. In our *Striped-MNIST* dataset, we simply add a vertical stripe to each image, with the stripe's horizontal position being a function of the image's label. With the exception of some noise, the dataset classes (originally digits 0-9) can alternatively be described by the position of the stripe; generally, digits 0 through 4 have a vertical stripe spanning the 2nd through 6th leftmost pixel columns, respectively, while digits 5 through 9 have their stripe spanning

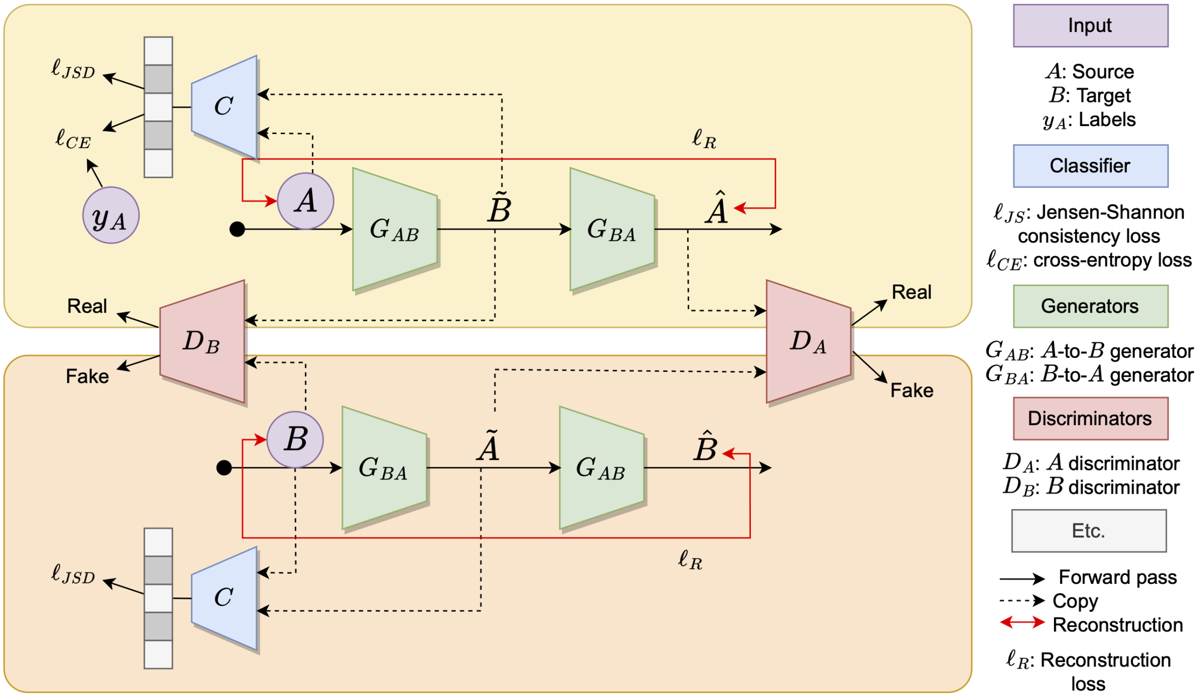

Fig. 2. System block diagram of *DiscoNet*, our proposed solution to shortcut learning.

the 6th through 2nd rightmost pixel columns, respectively. As the stripe pattern is clearly simpler and more accurate than digit recognition, we expect the model to greedily fit on the stripe positions in place of learning digit semantics. Having done so, we posit the following questions: would the model generalize to traditional MNIST? What if the test stripes were all in the same leftmost position? What if the test stripes were randomly distributed? Specifically, Figure 1 illustrates each of these possibilities, beginning with the source domain of ordered stripes and ending with the potential target domains. Ideally, the model would recognize the inconsistencies in stripe patterns across the source and target domains, then focus on what they have in common– the handwritten digits. However, it is possible that a performance-hungry classifier will narrow its vision to the stripes and disregard the cross-domain stripe relationships if not properly regularized. In the next section, we break down a strategy that places a proper check on the classifier to help adapt to these particular distribution shifts.

B. DiscoNet

The key to solving Striped-MNIST (or any similar DA task) is to best understand the differences between the source and target data distributions. In particular, a clever way is to model a one-to-one correspondence between the two domains; having done so, we may consider a target image x_t and find its "source equivalent" $x_s(\leftrightarrow \ x_t)$ by appealing to the cross-domain mapping. An unsupervised example of modeling cross-domain pairs is DiscoGAN [5], the establishment of an image-to-image translation with adversarial learning and reconstruction loss. Motivated by their architecture, we propose *DiscoNet*, a method for strategically defining cross-domain relationships in order to regularize the jointly-trained classifier. Specifically, the algorithm iterates as follows: first, we sample from the source domain (batch A and labels y_A) and the target domain (batch B). Without loss of generality, the generator G_{AB} is trained to receive batch A and produce $\tilde{B}$ in a way that the discriminator D_B has a difficult time distinguishing between the fake $\tilde{B}$ and real B; this objective is reflected in the generator loss ℓ_G. Moreover, G_{AB} is encouraged to map A to a batch in the target domain that is similarly embedded in the classifier C: this is quantified by the Jensen-Shannon consistency loss

$$\ell_{JS}(C_A, C_{\tilde{B}}) = \frac{1}{2}D_{KL}(C_A \| C_{\text{mix}}) + \frac{1}{2}D_{KL}(C_{\tilde{B}} \| C_{\text{mix}}), \quad (2)$$

where C_A and $C_{\tilde{B}}$ are the classifier outputs for A and $\tilde{B}$, respectively, and C_{mix} is the mixture of the two outputs. Assuming a fixed classifier, the Jensen-Shannon loss reinforces both generators to map images in one domain to semantically-similar images in the other domain. At the same time, the classifier C is trained to fit on the source labels with cross-entropy loss ℓ_{CE}, as well as similarly embed the image pairs with ℓ_{JS}. Then, generator G_{BA} maps $\tilde{B}$ to $\hat{A}$, which is trained to be a reconstructed version of A; the reconstruction loss ℓ_R is commonly MSE and works to reinforce the bijective nature of the mapping. As a result, the total loss of the system is

$$L := \ell_{CE} + \lambda_{JS}\ell_{JS} + \lambda_G\ell_G + \lambda_R\ell_R \quad (3)$$

where $\lambda_{JS}, \lambda_G, \lambda_R$ are tunable nonnegative scalars. For a visual overview of the algorithm, Figure 2 is provided.

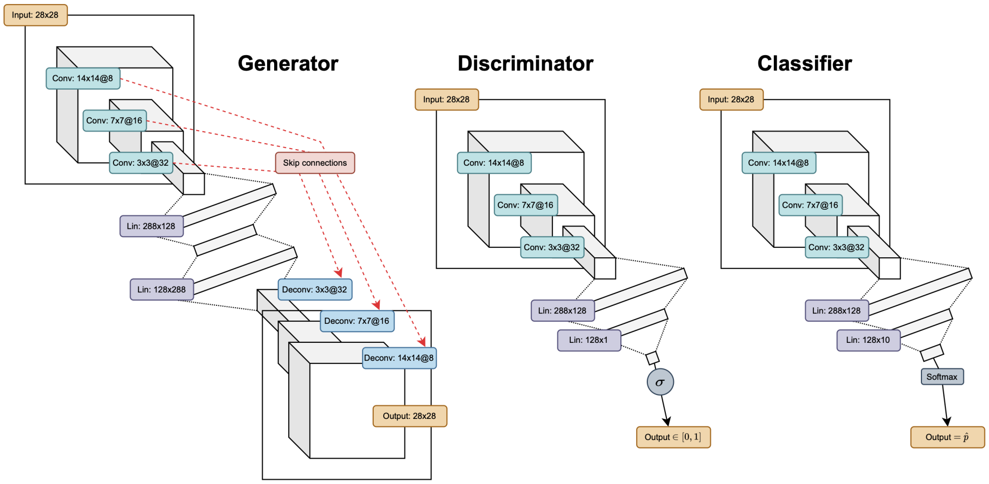

Fig. 3. Architectures for each model type– generator (×2), discriminator (×2), classifier– in DiscoNet.

V. Experimental Setup & Results

In this section, we compare the performances of two state-of-the-art DA methods, namely *unsupervised domain adaptation* (UDA) and *test entropy minimization* (TENT), with our method DiscoNet on our shortcut learning Striped-MNIST dataset. The source domain is fixed to the stripes ordered by digit (top row of Figure 1), and the next three rows are each considered as the target domain. The architecture for the DiscoNet models is given in Figure 3: the two generators follow a standard encoder-decoder scheme with skip connections. The two discriminators and classifier follow a similar CNN structure, with the discriminators outputting a single sigmoid unit and the classifier outputting a softmax vector of size 10. For each model, the convolution and deconvolution layers are succeeded with batch normalization, and the linear layers are preceded with dropout. Furthermore, each model is optimized with Adam, and the loss scalars $\lambda_{JS}, \lambda_G, \lambda_R$ are tuned to maximize the target data accuracy. Lastly, the generators and discriminators are updated using the least-squares GAN objective functions [8]. More details can be found in the repository[1], and results are given in Table I.

		Target Stripes		
Method	Domain	None	Single	Random
TENT	S	99.7 ± 0.0	59.7 ± 31.3	99.9 ± 0.0
	T	72.7 ± 0.1	10.4 ± 0.0	10.1 ± 0.3
UDA	S	98.1 ± 0.2	98.0 ± 0.2	**99.8 ± 0.1**
	T	99.3 ± 0.0	99.2 ± 0.1	10.0 ± 0.2
DiscoNet	S	**99.3 ± 0.4**	**99.37 ± 0.3**	99.6 ± 0.7
	T	**99.5 ± 0.1**	**99.40 ± 0.2**	**27.9 ± 40.0**

TABLE I
Classification accuracy (mean% ± std. deviation) reported across 5 seeds for three methods and three target domains. Best results for each target domain are **emboldened**.

When the target domain does not contain any stripes, we see that each method performs generally well; UDA and DiscoNet appear to completely adapt, while TENT achieves some gain with 72.7% target accuracy. In general, TENT has a lot of trouble with this dataset because the model learns the shortcut when training on the source data, and since the stripes are not included in the target data, the model fails to embed class representations present in both domains. On the other hand, when there exists a single fixed stripe in the target data, TENT performs even poorly because the source classifier confidently learns that the leftmost stripe means "0", so the entropy is already very small; in fact, minimizing the entropy further causes the model to output too many zeros on the source data, which brings down the accuracy from an initial 99% to less than 60%. For the single stripe, both UDA and DiscoNet identify the difference between the source and target, which allows them to focus on the digit features present in both domains. Figure 4 gives an example of DiscoNet's cross-domain mapping: clearly, the model learns to shift the stripe to position 0, back to the appropriate spot based on the digit. Every method has some trouble with the random stripes, most likely because there is not a shift in the label distribution (uniform for both), so cross-domain mappings can occur without forcing the classifier to focus on the digits.

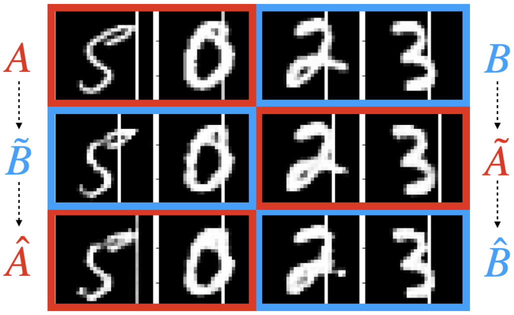

Fig. 4. Cross-domain mappings from DiscoNet trained on Stripe-GAN.

[1] GitHub repository: https://github.com/kotstot6/DiscoNet

References

[1] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation, 2014.

[2] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations, 2019.

[3] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty, 2019.

[4] Sarah Jabbour, David Fouhey, Ella Kazerooni, Michael W. Sjoding, and Jenna Wiens. Deep learning applied to chest x-rays: Exploiting and preventing shortcuts. *CoRR*, abs/2009.10132, 2020.

[5] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *CoRR*, abs/1703.05192, 2017.

[6] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, Tony Lee, Etienne David, Ian Stavness, Wei Guo, Berton A. Earnshaw, Imran S. Haque, Sara Beery, Jure Leskovec, Anshul Kundaje, Emma Pierson, Sergey Levine, Chelsea Finn, and Percy Liang. Wilds: A benchmark of in-the-wild distribution shifts, 2020.

[7] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2017.

[8] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, and Zhen Wang. Multi-class generative adversarial networks with the L2 loss function. *CoRR*, abs/1611.04076, 2016.

[9] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Fully test-time adaptation by entropy minimization. *CoRR*, abs/2006.10726, 2020.

[10] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization, 2017.