

A PROOF OF LEMMA 1

Proof. The two-player game in (3) can be written as:

$$\min_{\hat{P}(Y|X)} \max_{Q(Y|X) \ni} - \sum_x P_t(x) \sum_y Q(y|x) \log \hat{P}(y|x) - r \sum_x P_t(x) \sum_y Q(y|x) \mathbf{I}(\arg \max_{y'} Q(y'|x) = y) \log \hat{P}(y|x).$$

According to strong Lagrangian duality, we can switch the order of the two players and it is equivalent to:

$$\max_{Q(Y|X) \ni} \min_{\hat{P}(Y|X)} - \sum_x P_t(x) \sum_y Q(y|x) \log \hat{P}(y|x) - r \sum_x P_t(x) \sum_y Q(y|x) \mathbf{I}(\arg \max_{y'} Q(y'|x) = y) \log \hat{P}(y|x).$$

Solving the minimizing problem first assuming that we know $Q(Y|X)$, we get the result that $\hat{P}(Y|X) = Q(Y|X)$. Plugging it into the maximizing problem, the whole problem reduces to (4). \square

B PROOF OF THEOREM 1

Proof. The generalized constrained optimization problem in (4) can be written as:

$$\begin{aligned} & \max_{\hat{P}(Y|X)} - \sum_{x \in \mathcal{X}} P_t(x) \sum_{y \in \mathcal{Y}} \hat{P}(y|x) (\mathbf{1} + r \mathbf{I}(\arg \max_{y'} \hat{P}(y'|x) = y)) \log \hat{P}(y|x) \\ \text{such that: } & \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_s^k(x) \hat{P}(y|x) [f_k(X, Y)] = \check{c}'_k, \forall k \in \{1, \dots, K\} \\ & \forall x \in \mathcal{X}: \sum_{y \in \mathcal{Y}} \hat{P}(y|x) = 1 \\ & \forall x \in \mathcal{X}, y \in \mathcal{Y}: \hat{P}(y|x) \geq 0. \end{aligned}$$

Note that the final constraint is superfluous since the domain of the objective function is the positive real numbers. The Lagrangian associated with this problem is:

$$\begin{aligned} \mathcal{L}(\theta, \lambda) = & - \sum_{x \in \mathcal{X}} P_t(x) \sum_{y \in \mathcal{Y}} \hat{P}(y|x) (\mathbf{1} + r \mathbf{I}(\arg \max_{y'} \hat{P}(y'|x) = y)) \log \hat{P}(y|x) + \quad (11) \\ & \sum_k \theta_k \left[\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P_s^k(x) \hat{P}(y|x) f_k(x, y) - \check{c}'_k \right] + \sum_{x \in \mathcal{X}} \lambda(x) \left[\sum_{y \in \mathcal{Y}} \hat{P}(y|x) - 1 \right], \end{aligned}$$

where θ and $\lambda(x)$ are the Lagrangian multipliers. Taking the derivative with respect to $\hat{P}(y|x)$,

$$\frac{\partial}{\partial \hat{P}(y|x)} \mathcal{L}(\theta, \lambda) = -P_t(x) (\mathbf{1} + r \mathbf{I}(\arg \max_{y'} \hat{P}(y'|x) = y)) \log \hat{P}(y|x) - P_t(x) + \sum_k P_s^k(x) \theta_k f_k(x, y) + \lambda(x),$$

setting equal to zero, $\frac{\partial}{\partial \hat{P}(y|x)} \mathcal{L}(\theta, \lambda) = 0$, and solving, we get:

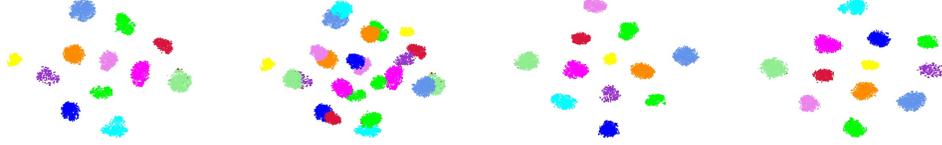
$$(\mathbf{1} + r \mathbf{I}(\arg \max_{y'} \hat{P}(y'|x) = y)) \log \hat{P}(y|x) = -1 + \sum_k \frac{P_s^k(x)}{P_t(x)} \theta_k f_k(x, y) + \frac{\lambda(x)}{P_t(x)} + \frac{(\mathbf{1} + r \mathbf{I}(\arg \max_{y'} \hat{P}(y'|x) = y))}{P_t(x)}$$

Therefore, if we instead set y as a one-hot encoding of the prediction, we conclude:

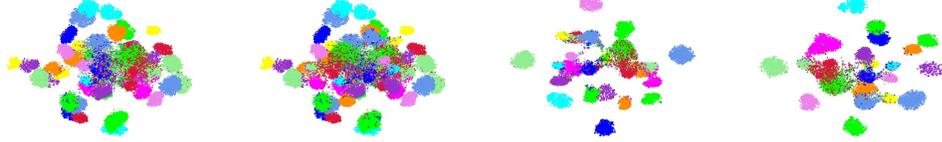
$$\hat{P}(y|x) \propto e^{\frac{\sum_k \frac{P_s^k(x)}{P_t(x)} \theta_k f_k(x, y) + r y}{1 + r y}},$$

The derivation of gradient then resembles the derivation in Theorem 1 in (Liu & Ziebart, 2014). \square

Source:



Target:



ASG

ASG+CBST

ASG+CRST

ASG+DRST

Figure 8: TSNE visualization of the learned classifier using different methods. Using DRST, the classes are well-separated.

C DETAILED DERIVATION OF GRADIENTS

Gradient of the expected target loss over source densities d_s , where we use R for the density ratio $\frac{P_s(x)}{P_t(x)}$ or $\frac{d_s}{d_t}$:

$$\begin{aligned} \frac{\partial L(\theta)}{\partial d_s} &= \frac{\partial}{\partial d_s} (-\theta \mathbb{E}_{P_s(x)P(y|x)}[\Phi(X, Y)] + \mathbb{E}_{P_t(x)}[\log Z_\theta(X)]) = \mathbb{E}_{P_t(x)\hat{P}(y|x)}[\theta\Phi(X, Y)/d_t] \\ &= \frac{1}{d_t} \frac{\partial L(\theta)}{\partial R}, \end{aligned} \quad (12)$$

and gradient of loss over target densities d_t :

$$\frac{\partial L(\theta)}{\partial d_t} = \frac{\partial}{\partial d_t} \mathbb{E}_{P_t(x)}[\log(Z_\theta(X))] = -\frac{d_s}{d_t^2} \mathbb{E}_{P_t(x)\hat{P}(y|x)}[\theta\Phi(X, Y)] = -\frac{d_s}{d_t^2} \frac{\partial L(\theta)}{\partial R}, \quad (13)$$

where:

$$\frac{\partial L(\theta)}{\partial R} = \mathbb{E}_{P_t(x)} \left[\sum_{y \in Y} \frac{\exp(R, \theta, \Phi)}{Z_\theta(X)} \theta\Phi(X, Y) \right] = \mathbb{E}_{P_t(x)\hat{P}(y|x)}[\theta\Phi(X, Y)]. \quad (14)$$

D ADDITIONAL DRST EXPERIMENTAL RESULTS

We demonstrate the TSNE plot of the learned decision boundaries for CBST, CRST and DRST in figure 8. Figure 9 shows the misclassification entropy comparison between source model and DRL model. Misclassification entropy is calculated as $S_i = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m p_{ij} \log p_{ij}$, where n is the number of samples and m is the number of categories in the dataset, and p_{ij} indicates the prediction probability of the i th sample on the j th category. The larger misclassification entropy is, the more uncertain the model prediction result is on the wrong predictions. This means the model would fail more gently. Figure 10 demonstrates additional model attention visualization. Figure 11 shows additional target examples with high and low density ratios. Our model is able to find noisy and less-represented image from the target data by estimating the density ratios. DRL would be more uncertain on those data.

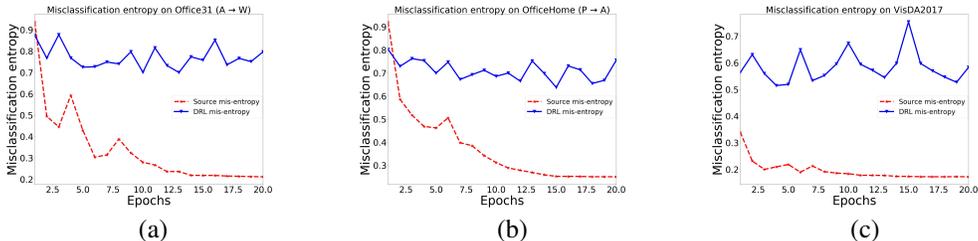


Figure 9: Comparison of DRL and DRST of misclassification entropy on different datasets.

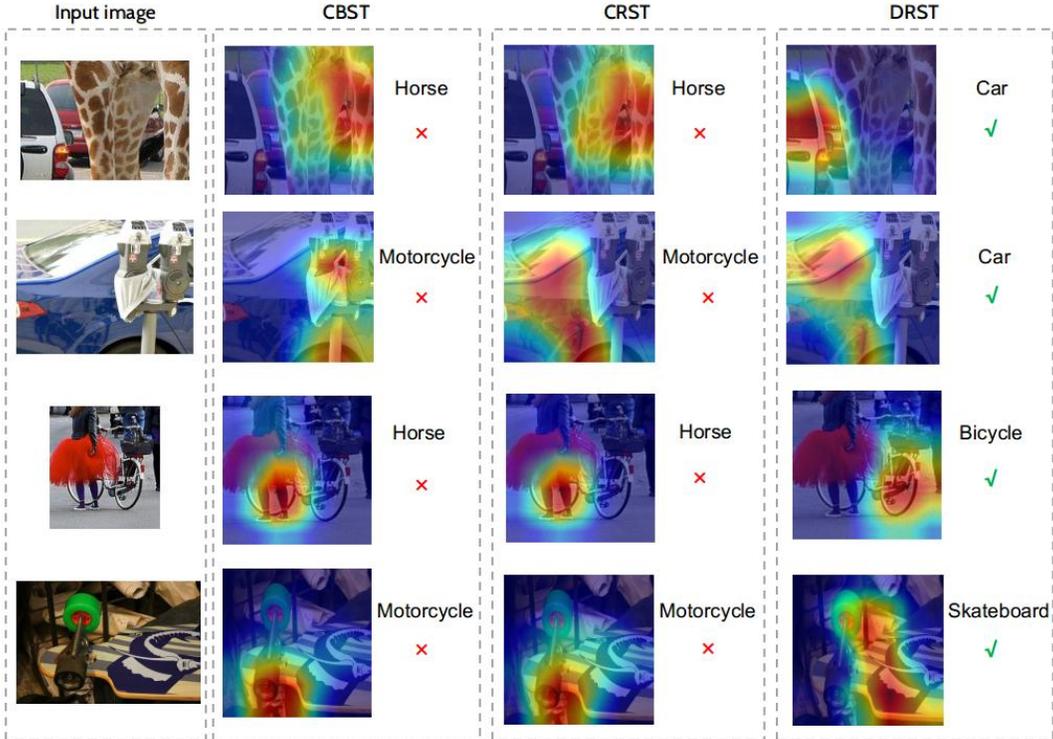


Figure 10: Model attention visualized using Grad-Cam (Selvaraju et al., 2017). Our method can also capture the domain knowledge well. For example, the first row input image contains a giraffe and a car. However, giraffe is not a existing label for VisDA2017. While CBST and CRST captures the wrong information, DRST is able to correctly capture the domain information.

E SIMULATION ON PLUG-IN ESTIMATOR

DRL for domain shift requires $P_s(x)/P_t(x)$ to adjust the representation-level conservativeness. Like many other machine learning methods using importance weights, like in transfer learning (Pan & Yang, 2009), or off-policy evaluation (Dudík et al., 2011), we can use a plug-in estimator for the density ratio $P_s(x)/P_t(x)$. However, density ratio estimation (Sugiyama et al., 2012), especially in the high-dimensional data, is rather different. Here, we ask the question: **whether a more accurate density ratio estimation leads to greater predictive performance in the downstream tasks?**

We show a two-dimensional binary classification example in figure 12 to demonstrate relation between the performance of the density (ratio) estimation and the performance of the ultimate target learning tasks. We use the RBA method (Liu & Ziebart, 2014) as an example. We conduct kernel density estimation (KDE) and evaluate the average log-likelihood of the source and target domain. We take the ratio of the density from KDE and plug in the RBA method. We can see that the case with higher log-likelihood actually fail to give informative predictions. One of the reason is the

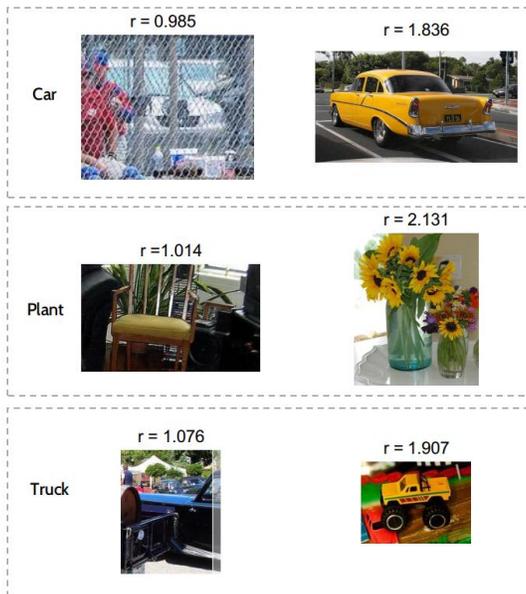


Figure 11: Additional examples for density ratio estimation for different categories. We can observe that data less well-represented in the source data has much lower density ratio. This shows that our learned density ratio is a good measure of the level of representation of data in source and target domains.

density (ratio) estimation task, as an independent learning task, does not share information with the downstream prediction tasks that use the ratios.

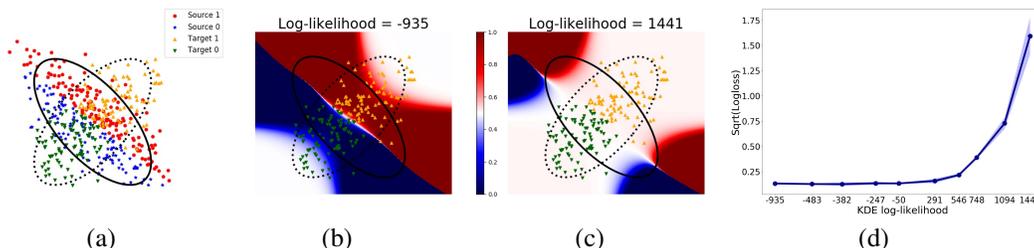


Figure 12: (a) Source and target data points are drawn from two Gaussian distributions. Solid line: source and dashed line: target. The underlying true decision boundary for the binary classes is the same between the two domains. (b)-(c) Prediction with density ratios from low and high density estimation likelihoods. With more accurate density estimation, the RBA predictor gives overly conservative predictions on the target domain. The colormap is the confidence $P("1" | x)$. (d) With larger likelihood in density ratio estimation, the target log loss becomes worse.

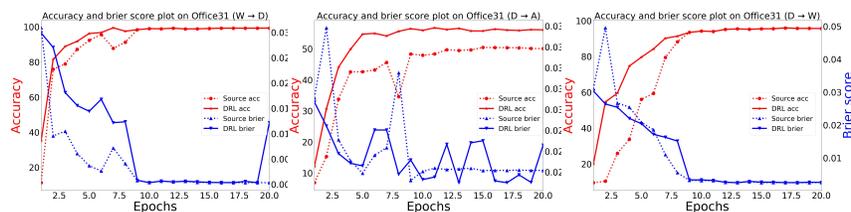


Figure 13: Additional Office31 results on DRL compared with source-only, which is a complement for Figure 6.

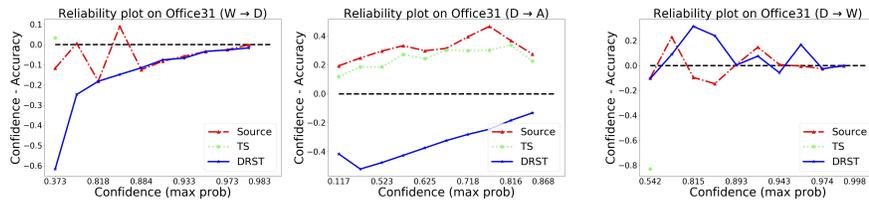


Figure 14: Additional Office31 results on reliability plots, which is a complement for Figure 7. DRL is compared with source-only and temperature scaling.

F ADDITIONAL RESULTS FOR DRL ON OFFICE-31

We include additional result for distributionally robust learning on office-31 in Figure 13 and Figure 14.

G CODE REPOSITORY

Code can be found at: <https://anonymous.4open.science/r/2ed8a9ce-f404-4489-9ef7-a3a83e02a44c/>