A. Supplementary Material

A.1. Comparison to ALDA [5]

Although joint training with importance sampling is one way to extend the ALDA [5] method (as compared in Section 4.3 in the main paper), here we consider the original algorithm of online ALDA (O-ALDA) on digit classification. We first extract the features from our domain adversarial model and train a perceptron classifier \( w_{\text{src}} \), a source classifier \( w_{\phi} \), and a domain separator \( \psi \) separately. There are two main differences: 1) this algorithm is performed in an online version, i.e., selecting one sample at a time then updating the classifier, and 2) if the selected image is similar to the source domain (determined by \( w_{\phi} \)), we use the pseudo-label from \( w_{\text{src}} \) without cost, and hence the number of selected images may be larger than the actual budget. The results are shown in Table A1, and our method outperforms O-ALDA by 10-15%.

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of Labeled Target</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
</tr>
<tr>
<td>AADA (Ours)</td>
<td>76.5</td>
</tr>
<tr>
<td>O-ALDA [5]</td>
<td>76.5</td>
</tr>
</tbody>
</table>


A.2. More Object Detection Results

Here we show the results on object detection after more sample selection rounds, which is an extension of Table 1 in the main paper. We perform 9 rounds in total with \( b = \{10, 10, 10, 20, 50, 100, 100, 200, 500\} \) for each round. We plot the \( x \)-axis in log scale for a better illustration in Figure A1. Our AADA improves over other baselines, including other sampling strategies with adversarial training and random sampling with different training schemes when up to 1000 labeled targets are available.

A.3. Comparison of Training Schemes on Office

In this section, we compare the results of adversarial training with different training schemes on the Office dataset [4] in Figure A2, as an extension of Section 5.1 in the main paper. With a random selection, adversarial training is better than other baselines including fine-tuning, joint training, and train on target data only. When using importance weight for sampling, adversarial training outperforms fine-tuning baseline. In addition, sampling with the proposed importance weight improves the performance over random selection when either adversarial training or fine-tuning is used. Overall, our adversarial training with importance weight (AADA) performs the best compared to other combinations of training schemes and sampling strategies.

A.4. Comparison of Training Schemes on VisDA

As described in Section 5.3 in the main paper, VisDA [2, 3] is a special case where the target domain is closer to images from ImageNet which is used for pre-training, and thus we utilize the fine-tuning strategy. In Figure A3, we further provide results of using adversarial training when few labeled targets \( L_t \) are available. To show more fine-grained results, we sample 10 images per round, i.e., \( b = 10 \), and perform 10 rounds of selection. In an unsupervised domain adaptation setting, i.e., no labeled target is available \( L_t = \emptyset \), using adversarial training on \( (L_u, U_t) \) improves the test accuracy on the target domain from 57.0% to 62.5%, compared to the model trained only on labeled source \( L_u \).
without adaptation. However, after adding labeled targets, the accuracy of the model using adversarial training decreases, as shown in blue and red curves in Figure A3, regardless of which sampling strategy is used. On the other hand, the accuracy of the model using fine-tuning increases when the number of labeled target increases, showing that VisDA is more suitable for fine-tuning due to its dataset property. Nevertheless, fine-tuning with our proposed importance weight still performs better than random sampling.

A.5. Effect of Source Data Number

We investigate the effectiveness of our method when the labeled source data is also limited. We use a subset \( \{5, 20, 50\}_\% \) of the source data, and compare results using adversarial training. We select 50 labeled targets per round and perform 10 rounds in total. As shown in Figure A4, using importance weight improves over random sampling in all the cases, especially on a smaller subset.

A.6. Results on Digits Classification with CDAN [1]

Our AADA framework can be applied to any domain adaptation model with a domain classifier and adversarial training. Here we integrate CDAN [1] model in our AADA framework and experiment on digit classification (SVHN→MNIST). We use the implementation provided by the authors and follow their training procedure, which yields 87.8% accuracy when there is no labeled target. We select 10 labeled targets in each round, and the performance saturates after 50 rounds. We compare different sampling methods in Table A2. Our proposed importance weight performs favorably against other methods.

Table A2. Results on SVHN → MNIST. We use CDAN [1] for training and compare different sampling approaches. In each round, we select 10 target samples to label.

References


