Semi-supervised Models are Strong Unsupervised Domain Adaptation Learners –Supplemental Material–

Anonymous Author(s)
Affiliation
Address
email

A Appendix

A.1 Datasets

Office31 [14] is the most popular but small-scale benchmark for UDA, which contains about 4.1K images shared by 31 classes and three domains, i.e., Amazon (**A**), Webcam (**W**), and DSLR (**D**).

OfficeHome [18] is a UDA dataset larger than the Office31, which contains $15.5K$ images shared by 65 classes and four domains, i.e., Artistic image (**Ar**), Clip Art (**Cl**), Product image (**Pr**), and Real-World images (**Rw**).

VisDA-2017 [13] is a challenging dataset, where the source domain includes synthetic samples and target images are real-world ones. There are 280.2K images shared by 12 classes and three sets, where the source training set contains 152.4K labeled data, the target training set contains 55.4K unlabeled data, and the target test set contains 72.4K unlabeled data.

DomainNet [12] is the largest UDA dataset to the best of our knowledge. It contains about 586.6K images shared by 345 classes and 6 domains, i.e., Clipart (clp), Infograph (inf), Painting (pnt), Quickdraw (qdr), Real (real), and Sketch (skt).

All datasets are licensed under the Custom License (non-commercial research and educational purposes). The data, which are more about artifacts and animals, contain no personally identifiable information or offensive content.

We adopt the same data pre-processing in all compared methods, including UDA ones in [7]. Specifically, for datasets of Office31, OfficeHome, and DomainNet, we pre-process each sample by resizing its shorter size to 256 while keeping the aspect ratio unchanged; then we randomly crop a region of 224×224 for training and the central 224×224 region for testing. For VisDA-2017 dataset, we resize each sample to the size of 256×256 and crop the central 224×224 region for training and testing following [7]. In all datasets, we use the horizontal flip version of the cropped 224×224 region for data augmentation in training; we normalize all samples with the mean and standard deviation calculated from the ImageNet dataset. Advanced data augmentation methods are adopted as the components in some methods (e.g,. Self-ensembling [3], Xie *et al.* [19] , and FixMatch [16]), which are also adopted in our implementations of these methods.

A.2 SSL methods

We focus on SSL methods that can be trained end-to-end, as introduced in the following.

Entropy minimization [5] promotes the low-entropy predictions of unlabeled data.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Table A1: Transductive results on the large-scale DomainNet dataset (ResNet50).

	Methods	clp→inf	inf→pnt	pnt→qdr	qdr→real	real→skt	skt→clp	Avg.
	Source Only	17.5±0.1	32.6±0.3	3.6±0.3	6.0±0.1	34.8±0.1	47.2±0.1	23.6
UDA	DANN [4]	20.3±0.1	27.5±0.1	6.1±0.4	15.6±0.3	39.8±0.2	48.5±0.1	26.3
	CDAN [10]	20.4±0.1	29.1±0.1	3.3±0.1	16.1±0.1	41.9±0.1	50.8±0.1	26.9
	AFN [20]	18.8±0.1	35.9±0.1	**6.3**±0.3	16.3±0.1	38.2±0.2	52.3±0.2	27.9
	MCD [15]	20.0±0.1	34.5±0.1	3.4±0.1	16.5±0.2	41.6±0.1	53.1±0.1	28.2
	MDD [22]	21.9±0.1	35.3±0.1	3.6±0.1	**19.8**±0.2	44.7±0.1	55.0±0.1	30.0
	Self-ensembling [3]	17.9±0.2	33.0±0.1	5.1±0.1	15.5±0.5	41.5±0.1	50.3±0.2	27.2
	MCC [6]	19.6±0.1	38.1±0.2	4.8±0.1	14.6±0.1	37.9±0.1	55.9±0.1	28.5
SSL	π-Model [8]	17.7±0.1	34.8±0.1	5.1±0.1	11.3±0.1	35.1±0.1	49.4±0.1	25.6
	VAT [11]	17.3±0.1	34.5±0.1	4.1±0.1	14.2±0.1	35.4±0.1	49.7±0.1	25.9
	Mean Teacher [17]	17.3±0.1	34.5±0.1	5.5±0.1	13.4±0.1	35.3±0.1	48.8±0.1	25.8
	Entropy mini. [5]	17.1±0.1	36.5±0.1	4.5±0.2	14.1±0.3	37.7±0.1	53.9±0.1	27.3
	MixMatch [1]	18.1±0.1	37.4±0.1	1.0±0.2	17.0±0.1	37.4±0.2	55.2±0.2	27.7
	Self-training [9]	18.3±0.1	38.0±0.1	3.9±0.2	16.9±0.2	40.8±0.1	55.0±0.1	28.8
	Xie *et al.* [19]	21.2±0.1	39.7±0.1	6.1±0.2	17.6±0.3	43.4±0.2	57.6±0.1	30.9
	FixMatch [16]	21.7±0.1	41.2±0.4	4.2±0.3	18.3±0.4	46.1±0.1	60.6±0.1	32.0
	MCC + Consistency	19.6±0.1	38.1±0.2	4.9±0.1	14.6±0.1	37.9±0.1	55.8±0.1	28.5
	MDD + Consistency	**23.4**±0.5	**42.1**±0.1	2.4±0.4	14.9±0.7	**50.9**±0.1	**63.5**±0.1	**32.8**

Self-training [9] trains models with labeled data and unlabeled data with pseudo labels, which are based on model predictions on unlabeled data. The self-training also minimizes the prediction entropy, presenting a similar objective to entropy minimization [5].

π-Model [8] penalizes the differences of two predictions for the same input using mean square loss, where random input perturbations and random model perturbations are adopted in different predictions. We drop the dropout perturbation since the vanilla ResNet model does not contain the dropout components.

Virtual Adversarial Training (VAT) [11] penalizes the differences of two predictions for the same input using KL divergence, where one prediction is achieved with the proposed adversarial input perturbation strategy.

Mean Teacher [17] penalizes the differences of two predictions for the same input using mean square loss, where one prediction is from the student model while the other prediction is from the teacher model; the weights of teacher model is updated as an exponential moving average of the student weights. We drop the dropout perturbation since the vanilla ResNet model does not contain the dropout components.

MixMatch [1] unifies objectives of entropy minimization [5], consistency regularization [8], and the generic regularization techniques (e.g., MixUp [21]) into one framework.

Unsupervised data augmentation (i.e., Xie *et al.* in Tables) [19] penalizes the differences of two predictions for the same input using KL divergence, where the two predictions are achieved with images of normal data augmentation and advanced data augmentation [2] respectively.

FixMatch [16] empowers the self-training [9] with an advanced data augmentation strategy [2], which simultaneously promotes low-entropy predictions and prediction consistency against input perturbations.

A.3 Results and analyses

Transductive results on the DomainNet dataset. Transductive results on the DomainNet dataset are shown in Table A1, which are quite close to the results in the inductive setting.

Combining UDA and SSL methods for UDA tasks. We enhance UDA methods of MCC [6] and MDD [22] with the popular consistency regularization [19, 16] in SSL. Specifically, MCC instantiates the regularization term $\mathcal{L}_{reg}(f, \mathcal{D}_u, \mathcal{D}_l)$ with unlabeled target data $\mathcal{D}_u$ (i.e., $\mathcal{D}_t$ in UDA) as:

$$\mathcal{L}_{reg}^{mcc}(f, \mathcal{D}_u, \mathcal{D}_l) = \mathbb{E}_{\boldsymbol{X}_u \in \mathcal{D}_u} \frac{1}{K} \sum_{i=1}^{K} \sum_{j \neq i}^{K} |\widetilde{\boldsymbol{C}}_{i,j}(\boldsymbol{X}_u)|, \tag{A.1}$$

where K is the total number of classes, $\boldsymbol{X}_u$ is a sample batch of size B, and $\widetilde{\boldsymbol{C}}_{ij}(\boldsymbol{X}_u)$ measures the prediction confusion between class i and class j. Specifically, with a slight abuse of notation, we introduce the target predictions $f(\boldsymbol{X}_u) = \boldsymbol{Y} \in \mathbb{R}^{B\times K}$; then the prediction probability with temperature scaling is introduced as:

$$\boldsymbol{P}_{ij} = \frac{\exp(\boldsymbol{Y}_{ij}/T)}{\sum_{j'=1}^{K}\exp(\boldsymbol{Y}_{ij'}/T)}, \tag{A.2}$$

where T is the temperature hyper-parameter. The preliminary definition of the class confusion is defined as:

$$\boldsymbol{C}_{i,j} = \boldsymbol{P}_{\cdot i}^{T}\boldsymbol{W}\boldsymbol{P}_{\cdot j}, \tag{A.3}$$

where $\boldsymbol{W}$ is a diagonal matrix with $\boldsymbol{W}_{ii} = \frac{B(1+\exp(-H(\boldsymbol{P}_{i\cdot})))}{\sum_{i'=1}^{B}(1+\exp(-H(\boldsymbol{P}_{i'\cdot})))}$, which quantifies the importance of the i-th example, and $H(\boldsymbol{P}_{i\cdot}) = -\sum_{k=1}^{K}\boldsymbol{P}_{ij}\log\boldsymbol{P}_{ij}$ is the entropy function. Finally, the class confusion is achieved with a category normalization technique:

$$\widetilde{\boldsymbol{C}}_{ij} = \frac{\boldsymbol{C}_{ij}}{\sum_{j'=1}^{K}\boldsymbol{C}_{ij'}}. \tag{A.4}$$

Based on the confusion matrix $\widetilde{\boldsymbol{C}}$ (A.4), we introduce the consistency regularization for MCC as:

$$\mathcal{L}_{consis}^{mcc}(f,\mathcal{D}_u) = \mathbb{E}_{\boldsymbol{X}_u\in\mathcal{D}_u}\frac{1}{K}\sum_{i=1}^{K}\sum_{j=1}^{K}\left(\widetilde{\boldsymbol{C}}_{i,j}(\boldsymbol{X}_u') - \widetilde{\boldsymbol{C}}_{i,j}(\widehat{\boldsymbol{X}_u'})\right)^2\frac{\sum_{i=1}^{B}\mathbb{I}(\max(\boldsymbol{P}_{i\cdot}) > thr)}{B}, \tag{A.5}$$

where $\boldsymbol{X}_u'$, a subset of $\boldsymbol{X}_u$, is composed of instances satisfying $\max(\boldsymbol{P}_{i\cdot}) > thr$, $\widehat{\boldsymbol{X}_u'}$ contains the same samples in $\boldsymbol{X}_u'$ with image perturbations [2] following [19, 16], $\mathbb{I}$ is the indicator function, and thr is a hyper-parameter. By promoting the confusion matrix consistency against input perturbations (A.5), we introduce the MCC + Consistency as:

$$\min_{f=h\circ g}\mathcal{L}_{sup}(f,\mathcal{D}_l) + \omega\mathcal{L}_{reg}^{mcc}(f,\mathcal{D}_u,\mathcal{D}_l) + \omega_1\mathcal{L}_{consis}^{mcc}(f,\mathcal{D}_u), \tag{A.6}$$

where $\omega_1 = 1$ works well for all datasets; we set thr as 0.7 and 0.95 for datasets of DomainNet and others, respectively. Other hyper-parameters are set following [6].

MDD [22] introduces the regularization term $\mathcal{L}_{reg}(f,\mathcal{D}_u,\mathcal{D}_l)$ with labeled source data, unlabeled target data, and an auxiliary task classifier h':

$$\mathcal{L}_{reg}^{mdd}(f = h\circ g,\mathcal{D}_u,\mathcal{D}_l) = \mathbb{E}_{\boldsymbol{x}^t\in\mathcal{D}_u}\mathcal{L}_{div}(h'(g(\boldsymbol{x}^t)),h(g(\boldsymbol{x}^t))) - \gamma\mathbb{E}_{\boldsymbol{x}^s\in\mathcal{D}_l}\mathcal{L}'_{div}(h'(g(\boldsymbol{x}^s)),h(g(\boldsymbol{x}^s))), \tag{A.7}$$

where $\mathcal{L}_{div}(h'(g(\boldsymbol{x}^t)),h(g(\boldsymbol{x}^t)))$ measures the margin disparity between $h'(g(\boldsymbol{x}^t))$ and $h(g(\boldsymbol{x}^t))$; $\mathcal{L}'_{div}$ is defined similarly. The auxiliary classifier h' is introduced by maximizing the $\mathcal{L}_{reg}^{mdd}(f,\mathcal{D}_u,\mathcal{D}_l)$ (A.7). We introduce the consistency regularization as the KL divergence between predictions of f following [11, 19]:

$$\mathcal{L}_{consis}^{mdd}(f,\mathcal{D}_u) = \mathbb{E}_{\boldsymbol{x}^t\in\mathcal{D}_u}KL(\sigma(f(\boldsymbol{x}^t)),\sigma(f(\mathcal{A}(\boldsymbol{x}^t))))\mathbb{I}(\max(\sigma(f(\boldsymbol{x}^t) > thr)), \tag{A.8}$$

where σ is the softmax function, KL is the KL divergence, and $\mathcal{A}(\boldsymbol{x}^t)$ is an augmentation of $\boldsymbol{x}^t$ following [2, 19]. Then we introduce the MDD + Consistency as:

$$\min_{f=h\circ g}\mathcal{L}_{sup}(f,\mathcal{D}_l) + \omega\mathcal{L}_{reg}^{mdd}(f,\mathcal{D}_u,\mathcal{D}_l) + \omega_1\mathcal{L}_{consis}^{mdd}(f,\mathcal{D}_u). \tag{A.9}$$

SSL loss on g only. The 'SSL loss on g only' in Section 4.2 is defined as:

$$\begin{aligned}&\min_{h}\mathcal{L}_{sup}(f = h\circ g,\mathcal{D}_l)\\&\min_{g}\mathcal{L}_{sup}(f = h\circ g,\mathcal{D}_l) + \omega\mathcal{L}_{reg}(f = h\circ g,\mathcal{D}_u,\mathcal{D}_l).\end{aligned} \tag{A.10}$$

Advanced data augmentation. We follow [16] to implement the advanced data augmentation [2] in methods of FixMatch [16], Xie *et al.* [19], MDD + Consistency, and MCC + Consistency.

Data pre-processing. Our data pre-processing strategies for datasets of Office31, OfficeHome, and DomainNet (cf. Section A.1) are slightly different from these in [7]. Each sample is firstly resized to the size of 256×256 in [7] whereas we resize the shorter size of each image to 256 and keep the aspect ratio unchanged. Then, we randomly crop a region of 224×224 for training while Long *et al.* [7] randomly crop regions of various sizes and resize the cropped regions to the size of 224×224 for training. Other pre-processing strategies, such as horizontal image flip and normalization, are shared. Note that we adopt the same data pre-processing strategies across all methods, including these in [7], for a fair comparison.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019.

[2] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[3] Geoff French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018.

[4] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[5] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems 17 [Neural Information Processing Systems, NIPS 2004, December 13-18, 2004, Vancouver, British Columbia, Canada]*, pages 529–536, 2004.

[6] Ying Jin, Ximei Wang, Mingsheng Long, and Jianmin Wang. Minimum class confusion for versatile domain adaptation. In *European Conference on Computer Vision*, pages 464–480. Springer, 2020.

[7] Mingsheng Long Junguang Jiang, Bo Fu. Transfer-learning-library. `https://github.com/thuml/Transfer-Learning-Library`, 2020.

[8] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[9] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 2, 2013.

[10] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650, 2018.

[11] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.

[12] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019.

[13] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv preprint arXiv:1710.06924*, 2017.

[14] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[15] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018.

[16] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[17] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017.

[18] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5018–5027, 2017.

[19] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6256–6268. Curran Associates, Inc., 2020.

[20] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1426–1435, 2019.

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

[22] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413, 2019.