arXiv:2101.08237v1 [cs.CV] 19 Jan 2021

On The Consistency Training for Open-Set Semi-Supervised Learning

Huixiang Luo, Hao Cheng, Yuting Gao, Ke Li, Mengdan Zhang, Fanxu Meng,

Xiaowei Guo, Feiyue Huang, Xing Sun

Abstract

Conventional semi-supervised learning (SSL) methods, e.g., MixMatch, achieve great performance when both labeled and unlabeled dataset are drawn from the same distribution. However, these methods often suffer severe performance degradation in a more realistic setting, where unlabeled dataset contains out-of-distribution (OOD) samples. Recent approaches mitigate the negative influence of OOD samples by filtering them out from the unlabeled data. Our studies show that it is not necessary to get rid of OOD samples during training. On the contrary, the network can benefit from them if OOD samples are properly utilized. We thoroughly study how OOD samples affect DNN training in both low- and high-dimensional spaces, where two fundamental SSL methods are considered: Pseudo Labeling (PL) and Data Augmentation based Consistency Training (DACT). Conclusion is twofold: (1) unlike PL that suffers performance degradation, DACT brings improvement to model performance; (2) the improvement is closely related to class-wise distribution gap between the labeled and the unlabeled dataset. Motivated by this observation, we further improve the model performance by bridging the gap between the labeled and the unlabeled datasets (containing OOD samples). Compared to previous algorithms paying much attention to distinguishing between ID and OOD samples, our method makes better use of OOD samples and achieves state-of-the-art results.

1. Introduction

The majority of SSL algorithms are designed assuming that both the labeled and the unlabeled dataset are drawn from the same distribution, which means they share the same classes and no outlier exists in the unlabeled dataset. When it comes to a more realistic setting where the unlabeled dataset contains out-of-distribution (OOD) samples, performance of many popular SSL algorithms are severely damaged [34, 16, 49, 9]. This setting is firstly introduced by [49], and is named as "Open-Set Semi-Supervised Learning" (open-set SSL, illustrated in Figure 1). To mitigate the negative influence of OOD samples, previous studies [9, 49] directly roll back from the realistic open-set SSL setting to the conventional SSL setting by detecting OOD samples and reducing their weights in loss functions.

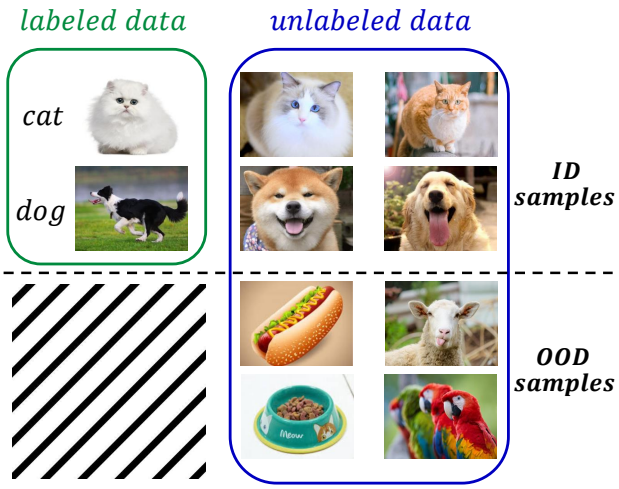

Figure 1: In open-set semi-supervised learning, unlabeled dataset contains OOD samples that do not belong to any labeled classes.

However, it is not clear whether removing the OOD samples is the best choice. In this paper, we find that UDA [45], a recent SOTA method that would keep OOD samples during training, shows great robustness in open-set SSL. For example, when CIFAR-10 [26] and Tiny ImageNet [11] are provided as ID and OOD dataset respectively [49], the network trained by UDA could benefit from OOD samples and improve classification performance without explicitly detecting OOD samples. To better understand this phenomenon, we compare Pseudo Labeling (PL) and Data Augmentation based Consistency Regularization (DACT), two mainstream methods in SSL methods, in terms of open-set SSL on both low-dimensional and high-dimensional spaces. We observe that OOD samples trained by DACT

could bring feature invariant property to networks which leads to better robustness, while PL fails on such setting. Further, we find the performace gain brought by OOD samples is closely related to metrics widely used in transfer learning for distribution-gap measurement: Mean Maximum Discrepancy (MMD) [15] & class-wise MMD [32]. With the metirc, we observe that DACT-based SSL methods perform better when the distance between the labeled and the unlabeled data (containing OOD samples) distributions is smaller.

The observation above suggests that we could further improve DNN by decreasing the distance between labeled data and unlabeled data containing OOD samples. Motivated by [22], we train Neural Style Transfer (NST) module on labeled ID samples and OOD samples to convert OOD samples into ID-like samples. When created samples are added into unlabeled dataset, the distance between labeled and unlabeled data distributions is decreased, and the network trained on the new data achieves better performance. We also find that in the case where the number of ID samples is small, we could better use OOD samples via self-supervised learning to get a better pre-trained model. However, this process does not contradict with NST module, since they improve the network in variant ways (sufficient ablation studies are performed in Section 5.3). We evaluate our method on diverse open-set SSL settings and show our method achieves great advantages over other existing methods. The contributions of this paper can be summarized as follows:

- We analyze two fundamental SSL methods, Pseudo Labeling (PL) and Data Augmentation based Consistency Regularization (DACT) on open-set SSL and find that DACT can benefit from OOD samples while PL suffers a severe performance degradation. The performance is related to the distance between the labeled and the unlabeled data distributions.
- Instead of removing OOD samples, we use Neural Style Transfer (NST) to decrease the distance between labeled and unlabeled data distributions. Trained on new data created by NST, the network gets better performance.

2. Related Work

2.1. Conventional semi-supervised learning.

The goal of semi-supervised learning (SSL) methods is to leverage unlabeled data for performance improvement on labeled data. We focus on current SOTA methods for image classification tasks, so SSL techniques including graph-based methods [25, 23, 31, 47] and generative modeling [24, 13, 33] are not discussed here. In conventional semi-supervised learning, pseudo labeling (PL) and consistency regularization are two popular and fundamental methods of many recent SOTA algorithms [3, 45, 2, 40, 27, 35, 46].

Pseudo Labeling takes the idea that model trained by labeled data can obtain artificial labels of unlabeled dataset by itself [29, 35, 30]. In a narrow sense, PL refers to using 'hard' pseudo-labels of samples that satisfy constraints of threshold η (e.g. the largest class probability $\geq \eta$ [29]). The hard PL method is closely related to entropy minimization [29, 14], a common method forcing the model's prediction to be in high confidence on unlabeled samples. In a broad sense, methods using 'soft' pseudo-label for supervision also belong to PL-based algorithms. MPL [35] combines meta-learning [42] theory with knowledge distillation [19], and generates dynamic target distributions of training examples by teacher network to improve the learning of student network.

Consistency regularization, another fundamental component in SSL algorithms, is usually implemented by enforcing the model output stable with perturbations during training. Consistency constraints are required in various ways, such as image-level [3, 2, 40, 45, 44], model-level [46, 52], feature-level [27], distribution-level [2] and temporal-level [41, 28, 54] consistency. The most widely used consistency regularization is Data-Augmentation based Consistency Training (DACT). MixMatch [3] applies random horizontal flips and crops several times on a single image, and the average prediction is used for consistency training. ReMixMatch [2] requires the model's prediction of weakly and strongly augmented images to be consistent with each other. UDA [45] leverages CutOut [12] and RandAugment [10] to unlabeled samples. FixMatch [40] follows UDA and ReMixMatch to adopt similar strategies as strong augmetation with samples filtered by threshold.

2.2. Open-set semi-supervised learning

Open-set SSL is a more realistic setting mentioned in [34, 49], where only 'dirty' unlabeled dataset (*i.e.,* dataset has both In-Distribution (ID) and Out-Of-Distribution (OOD) samples) is available. Algorithms [45, 35, 28] for conventional SSL settings tend to filter out OOD samples with threshold in advance before using dirty unlabeled dataset. These offline 'hard-weight' methods (*i.e.,* weight of OOD and ID samples is set to zeros and ones respectively) are later improved as 'soft-weight' and online filtering methods. UASD [9] ensembles model predictions temporally, and maximal probability of unlabeled samples are compared with a dynamic threshold to filter out OOD samples online. MTCF [49] takes the idea of Positive and Unlabeled (PU) learning to detect OOD samples online, and train the model by SSL methods simultaneously. DS3L [16] designs an online 'soft-weight' framework (i.e. weight of samples $\in [0, 1]$) formulated as a bi-level optimization problem. Guided by meta-learning [38], weight of OOD samples is

reduced to zero softly. Unlike previous methods handling OOD samples by OOD detection and reducing their weight, our method tries to make better use of OOD samples by neural style transfer and self-supervised pretraining.

3. Pseudo Labeling vs. Data-Augmentation based Consistency Training

3.1. Illustration in low-dimensional space

To better understand the differences between pseudo labeling (PL) and data-augmentation-based consistency training (DACT), we visualize the decision boundary of model on a two-dimensional synthetic dataset.

3.1.1 Experiment settings and results

The In-Distribution (ID) dataset consists of two classes, class $A = \{(r_a = r, \theta_a = \theta)|0 < r < 1, \frac{\pi}{2} < \theta < \frac{3\pi}{2}\}$, and class $B = \{(r_b = r, \theta_b = \theta)|1 < r < 2, \frac{\pi}{2} < \theta < \frac{3\pi}{2}\}$, where r and θ are the radius and the angle in polar coordinates. In the following experiments, labeled dataset D_L is drawn from class A and B ($|D_L|$ = 1000). Since data augmentation would not change the labels of samples from common assumption, we simulate the data augmentation process in 2-dimensional space by adding subtle disturbance on polar angle: $DA_{real} = \{F_i((r, \theta)) = (r, \theta + \Delta\theta_i)|\Delta\theta_i \in [-\Theta, \Theta], i = 1, ..., N_d\}$. The classification model here is an NN classifier with a 100-unit hidden layer.

Experiments with unlabeled ID dataset. In the conventional setting of SSL where labeled and unlabeled data are assumed to have the same distribution, both PL and DACT perform similarly well. In this experiment, unlabeled dataset D_U is also sampled from class A and B ($|D_U|$ = 10000). The model is trained over 1,000 epochs with batch size 256. Hinge loss is used for labeled dataset D_L. For DACT, we use MSE loss to keep consistency of model predictions before and after data augmentation. For PL, the model is trained with labeled data in the first 500 epochs; in the last 500 epochs, all unlabeled samples whose prediction confidences are larger than threshold η (0.7 or 0.9) are pseudo-labeled and trained with hinge loss. Loss function for semi-supervised setting is written as:

$$\mathcal{L} = \mathcal{L}_{sup} + \lambda\mathcal{L}_{unsup} \tag{1}$$

where $\lambda = 0.1$.

Figure 2 (a) shows that the two methods have similar positive influences on the decision boundary with unlabeled ID samples. Compared with the decision boundary of model trained by D_L only, both methods push the boundary closer to the optimal one.

Experiments with unlabeled OOD dataset. When D_U is made up of OOD samples, however, PL and DACT methods would have dramatically different influences.

When D_U are sampled from $OOD_{distant} = \{(r_d = r, \theta_d = \theta)|2 < r < 3, -\frac{\pi}{2} < \theta < 0\}$, whose distribution is far from the distribution of D_L ($D_L \& D_U$ lie in different quadrants of the plane respectively), boundaries influenced by either method are also far from the ID distribution. Consequently, PL performs slight worse than DACT, as is depicted in Figure 2 (b).

When D_U are sampled from $OOD_{close} = \{(r_c = r, \theta_c = \theta)|2 < r < 3, \frac{\pi}{2} < \theta < \frac{3\pi}{2}\}$, whose distribution is nearer to the distribution of D_L ($D_L \& D_U$ lie in the same quadrants of the plane), DACT optimizes the model by pushing its boundary closer to the optimal one, while PL still has no gain from D_U, as is depicted in Figure 2 (c).

Experiments above show that performance of both method varies with the distribution of D_U, and DACT performs more robustly than PL.

3.1.2 Analysis of PL on OOD samples

Recent studies [29, 30, 4] explain how PL works under the common low-density separation assumption. The assumption states that the decision boundary should not cross high-density regions, but instead lies in low-density regions. For example, Entropy Minimization (EntMin) [29] achieves the assumption by making model predictions more **confident**. Similar to EntMin, PL can push the decision boundary into low-density regions by selecting high **confident** unlabeled samples [29, 30]. However, these unlabeled samples must lie in the same distribution as labeled samples. Without such assumption, the decision boundary would be less optimal since the data distribution varies. From Figure 2 (b)(c), the decision boundary of PL is less robust to OOD samples when we take into account less confident samples (with lower threshold), and the performance gets worse. In Section 3.2, experiments of removing PL module from SSL methods in high-dimensional space further verify our observation. We provide more justifications in the supplementary materials.

3.1.3 Analysis of DACT on OOD samples

Data augmentation is widely used to improve the generalization ability of models [37]. Training dataset is enlarged by label-preserving augmentation methods so that models can learn more robust representations and avoid overfitting. In the synthetic two-class dataset, the key knowledge (or invariance) for classification is: Label of samples is decided by the only factor radius, and is invariant to polar angle. The invariance can be expressed equivalently to an ideal data augmentation method: $DA_{ideal} = \{F_i((r, \theta)) = (r, \theta + \Delta\theta_i)|\Delta\theta_i \in [-\pi, \pi], i = 1, 2, ...\}$. In the real setting, however, only a proper subset of all data augmentation methods can be enumerated and used. Our chosen

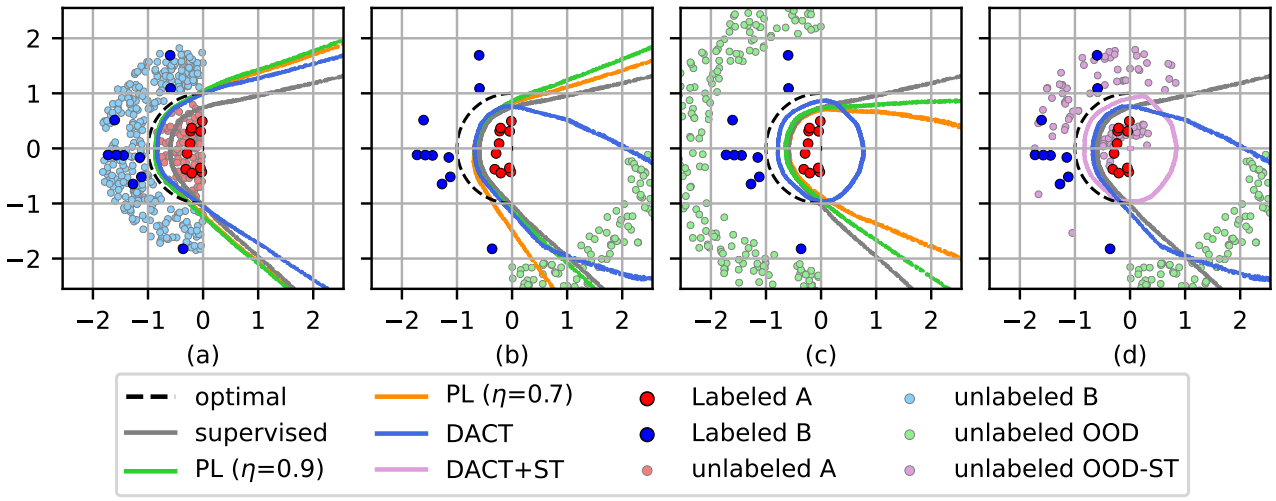

Figure 2: Analysis of PL & DACT in low-dimensional space. (a): With unlabeled ID dataset, both methods perform similarly well. (b): With unlabeled OOD dataset whose distribution gap to labeled ID dataset is large, PL with lower threshold performs worse while DACT makes nearly no influence on model performance. (c): When the distribution gap from unlabeled OOD dataset to labeled ID dataset is smaller, DACT has positive influence on model performance while PL keeps it still. (d): With enlarged unlabeled dataset containing both OOD samples (the same samples as (b)) and style-transferred samples, model performance is improved as the gap is getting smaller.

Method	OOD dataset					Mean acc change
	Clean	LSUN	TIN	Gaussian	Uniform	
MixMatch	90.67 ± 0.29	87.03 ± 0.41	88.03 ± 0.22	84.49 ± 1.06	85.71 ± 1.14	↓ 4.36
MixMatch(w/o PL)	90.67 ± 0.29	87.93 ± 0.17	88.64 ± 0.20	86.09 ± 1.27	86.90 ± 0.12	↓ 3.28
UDA	88.29 ± 0.25	88.60 ± 0.22	88.86 ± 0.37	89.15 ± 0.19	89.22 ± 0.25	↑ 0.67
mmd_{gap}	/	1.71 ± 0.26	1.47 ± 0.26	1.20 ± 0.20	1.19 ± 0.19	/

(a) CIFAR-10 with 1000 labeled and 54000 unlabeled samples

Method	OOD dataset					Mean acc change
	Clean	LSUN	TIN	Gaussian	Uniform	
MixMatch	93.30 ± 0.10	91.18 ± 0.33	91.25 ± 0.13	90.47 ± 0.38	91.51 ± 0.35	↓ 2.20
MixMatch(w/o PL)	93.30 ± 0.10	91.56 ± 0.02	91.98 ± 0.09	92.09 ± 0.11	91.79 ± 0.24	↓ 1.45
UDA	93.36 ± 0.40	93.56 ± 0.04	93.65 ± 0.14	93.80 ± 0.08	93.84 ± 0.47	↑ 0.35
mmd_{gap}	/	1.85 ± 0.35	1.54 ± 0.37	1.28 ± 0.32	1.26 ± 0.31	/

(b) CIFAR-10 with 4000 labeled and 51000 unlabeled samples

Table 1: Accuracy(%) for CIFAR-10 and OOD dataset pairs. Following the setup in [49], we report the averages and the standard deviations of the scores obtained from three trials. "Clean" means unlabeled dataset doesn't contain OOD samples.

DA method DA_{real} is exactly a proper subset of the ideal DA_{ideal}. As a result, DA_{real} could help model learn classification invariance partially as well.

By comparing blue lines in Figure 2 (b)(c)(a), we can observe that: DACT helps model learn the invariance by warping decision boundaries that lie in/near the distribution of D_U. In other words, when OOD samples is sampled from distributions nearer to that of D_L, more boundaries lying in the ID distribution would be warped, and model performance would be further improved. Due to better use of OOD samples, DACT outperforms PL by warping decision boundaries. Experiments in Section 3.2 further ver-

ify our observation by comparing the performance of both methods on high-dimensional space.

3.1.4 Bridging the distribution gap by style transfer

Since DACT performs better when the distribution of D_U is closer to ID distribution, we infer that bridging the gap between them could further improve model's performance. A simple way to bridge the gap is adding to D_U new samples that are closer to the ID distribution. Inspired by both the label-preserving principle of data augmentation and neural style transfer methods, we generate images by transfering their styles from OOD to ID samples.

Each sample x could be split into two parts: $x = (x_{content}, x_{style})$, where $x_{content}$ preserves the information of class label and x_{style} brings variance to the dataset. Take the synthetic dataset above to validate our inference, and sample x is rewritten as (r_x, θ_x). Then the style-transferred sample x^{st} can be defined as:

$$\begin{aligned} x^{st} &= (x^{ID}_{content}, (1-\omega)x^{ID}_{style} + \omega x^{OOD}_{style}) \\ &= (r^{ID}, (1-\omega)\theta^{ID} + \omega\theta^{OOD}) \end{aligned} \tag{2}$$

where $\omega \in [0, 1]$ is the weighting coefficient.

To better illustrate the effect of style transfer together with DACT, we choose the same experiment setting as Figure 2 (b), where neither PL nor DACT could improve model performance remarkably. Each transferred sample x^{st} is generated according to 2 with a pair of samples (x^{ID}, x^{OOD}) chosen in random, ω is randomly selected from $[0, 0.5]$. The transferred dataset D^{ST} is denoted by pink points in Figure 2 (d). Because many transferred samples lie in the same quadrants as D_L, the gap between new unlabeled dataset $D_U \cup D^{ST}$ and D_L is getting smaller. Provided with the new unlabeled dataset, DACT would help push the boundary closer to the optimal one, as is depicted in Figure 2 (d). Next we turn to high-dimensional space for further experiment.

3.2. Validation in high-dimensional space

To validate our observation in high-dimensional space, MixMatch [3] and UDA [45], two excellent SSL methods on image classification tasks, are chosen as representatives for PL and DACT respectively. MixMatch is regarded as a PL-related method here, because unlabeled dataset are pseudo-labeled and used for supervision together with labeled dataset by Mixup [51]. UDA is a typical DACT-based method, and we only adopt its consistency training module in this paper without any other regularization terms.

As is reported in [49], the performance of MixMatch is damaged sharply once OOD samples appear in unlabeled dataset. We follow the same experiment setting on CIFAR-10 [26] to verify our observation. Table 1 shows that UDA could surprisingly use unlabeled OOD samples together with ID samples to improve model performance, while MixMatch brings severe damage to model performance. The result proves the robustness of DACT on OOD samples.

To verify our observation further, we remove the PL-related module in MixMatch by disabling the Mixup procedure. As is depicted in Table 1, MixMatch (w/o PL) surpasses the original algorithm on OOD dataset by 1.08% and 0.75% respectively. The ablation study of PL-related module again verifies our observation that PL is not robust to OOD samples.

4. Bridging the gap between labeled and unlabeled datasets

4.1. Distribution-gap measurement

Section 3.1 shows that influence of OOD samples on the model varies with distribution, and DACT performs better when the distribution of labeled dataset D_L and unlabeled dataset D_U is closer to each other. To measure the distribution gap in high-dimensional space quantitatively, we choose empirical criteria of Maximum Mean Discrepancy (MMD) [15], which is widely used in transfer learning. Following studies [32, 43, 20] to measure the gap better by balancing both label and structural information, MMD and class-wise MMD [32] are adopted together to evaluate the marginal and conditional distribution gap between labeled and unlabeled dataset. The metric mmd_{gap} is written as:

$$\begin{aligned} mmd_{gap} = \| &\frac{1}{|D_L|}\sum_{i=1}^{|D_L|} p_\phi(y|x_i) - \frac{1}{|D_U|}\sum_{j=1}^{|D_U|} p_\phi(y|x_j)\|^2_{\mathcal{H}} \\ + \sum_{c=1}^{K} \| &\frac{1}{|D^c_L|}\sum_{x_i \in D^c_L} p_\phi(y|x_i) - \frac{1}{|D^c_U|}\sum_{x_j \in D^c_U} p_\phi(y|x_j)\|^2_{\mathcal{H}} \end{aligned} \tag{3}$$

where K is the number of class in D_L, D^c_L is the labeled c-th class dataset, D^c_U is the pseudo-labeled c-th dataset, ϕ is the model, x is the input image and $\phi(x)$ is a 1xK vector representing the probability of x to each class.

We use mmd_{gap} on the high-dimensional experiment (Section 3.2) for validation. Model ϕ is trained by D_L only. The last columns of Table 1 show how the distribution gap varies with the OOD samples of unlabeled dataset. We notice that the top-1 accuracy of model trained by UDA rises when mmd_{gap} drops, indicating that mmd_{gap} is appropriate for distribution-gap measurement. Another interesting phenomenon is observed in Table 1: noisy samples generated by pixels chosen in random (e.g., from Gaussian or Uniform distribution), could help improve more performance than samples from real scenarios (e.g., from LSUN [48] or Tiny ImageNet [11]).

Algorithm 1 **B**ridging Distribution **G**ap for **D**ata-**A**ugmentation-based **C**onsistency **T**raining (BG-DACT) algorithm

Input: Labeled minibatch $\mathcal{X} = \{(x_l, y_l)\}, |\mathcal{X}| = B_l$, and unlabeled minibatch $\mathcal{U} = \{x_u\}, |\mathcal{U}| = B_u$; Classification model ϕ, style-transfering model $AdaIN$, memory bank $Bank$, split OOD set S_{OOD}, data augmentation policies Aug; ID class number K, momentum parameter α, mixing parameter β, consistency loss weight λ_u, total epoch number E, the epoch to split OOD samples E_{sp}.

for $x_u \in D_u$ **do** ▷ Memory bank initialization
  $Bank[x_u] = [0, ..., 0, 1]_{K+1}$
for epoch = 1 to E **do**
  if epoch $\%E_{sp}$ == 0 **then** ▷ Splitting OOD samples
    $S_{OOD} = \{x_u | \arg\max Bank[x_u] == K + 1\}$
  for $(x_l, y_l) \in \mathcal{X}$ **do**
    $\mathcal{L}_{sup} = \sum CE(y_l, p_\phi(y|x_l))$ ▷ Calculate the Cross-Entropy loss L_{sup} for $\mathcal{X}$
  for $x_u \in \mathcal{U}$ **do**
    $Bank[x_u] = p_\phi(y|x_u) = \alpha Bank[x_u] + (1 - \alpha)p_\phi(y|x_u)$ ▷ Update memory bank
    $\mathcal{L}_{unsup} = \lambda_u \sum KL(p_\phi(y|x_u), p_\phi(y|Aug(x_u)))$ ▷ Calculate the consistency loss L_{unsup} for $\mathcal{U}$
    if $x_u \in S_{OOD}$ **then**
      $x^{st} = \beta AdaIN(x_u, x_l) + (1 - \beta)x_l$ ▷ Bridge the distribution gap by style transfer
      $\mathcal{L}_{ST} = \sum KL(p_\phi(y|x^{st}), p_\phi(y|Aug(x^{st})))$ ▷ Calculate the consistency loss L_{ST} for transferred samples
  $\mathcal{L}_{total} = \mathcal{L}_{sup} + \mathcal{L}_{unsup} + \mathcal{L}_{ST}$ ▷ Update ϕ to minimize total loss

Return: classification model ϕ

4.2. Gap reduction by neural style transfer

Since model performance boosts when the gap between D_L and D_U is getting smaller, methods such as Neural Style Transfer (NST) can be taken for gap reduction, as is demonstrated in Section 3.1.4. Before style transfer, OOD samples whose distribution gap is far from ID distribution need to be sorted out. For convenience, previous method [49] is simplified and integrated in our pipeline: the two projection heads designed for K-class image classification and OOD sample detection are merged into one (K+1)-class head, and the (K+1)-th class denotes the probability of samples to be OOD. All unlabeled samples are regarded as OOD samples at the beginning of training. To prevent unlabeled samples from being split into ID samples too early, the prediction of original image $p_\phi(y|x)$ in consistency training loss is replaced with $p^t_\phi(y|x)$, the weighted sum of model's previous predictions $p^{t-1}_\phi(y|x)$ and current prediction $p_\phi(y|x)$:

$$p^t_\phi(y|x) = \alpha p^{t-1}_\phi(y|x) + (1 - \alpha)p_\phi(y|x) \tag{4}$$

where $\alpha \in [0, 1]$. Motivated by momentum update in MoCo [17], previous predictions of D_U are stored in a memory bank. Unlabeled samples are regarded as OOD ones if the $(K + 1)$-th probability of output is the largest after proper epochs.

Next, we choose AdaIN [21], a real-time arbitrary style transfer method for distribution-gap reduction. The transferred image x^{st}_i is generated with content of labeled ID image x^{ID}_i and style of unlabeled OOD image x^{OOD}_i. To avoid negative effect of artifacts caused by style transfer, x^{st}_i is further linearly interpolated with x^{ID}_i:

$$x^{st}_i = \beta AdaIN(x^{OOD}_i, x^{ID}_i) + (1 - \beta)x^{ID}_i \tag{5}$$

where $\beta \in [0, 1]$ controls the similarity of x^{st}_i to x^{ID}_i, $AdaIN$ is the style-transfer network. The style-transferred dataset D^{ST} would be used in the same way as D_U to keep consistency of model predictions with KL-divergence, and a standard cross-entropy loss is used for labeled data.

In brief, our method takes the DACT module of UDA as the basic method; AdaIN is used to generate OOD-style images with content of ID images, and these images are used together with original unlabeled dataset for DACT. The overall loss function is composed of the UDA loss $\mathcal{L}_{UDA}$ and loss of style-transferred samples $\mathcal{L}_{ST}$. It can be written as:

$$\mathcal{L}_{total} = \mathcal{L}_{UDA} + \mathcal{L}_{ST} = \mathcal{L}_{sup} + \mathcal{L}_{unsup} + \mathcal{L}_{ST} \tag{6}$$

$$\mathcal{L}_{sup} = \sum_{x_l, y_l \in D_L} CE(y_l, p_\phi(y|x_l)) \tag{7}$$

$$\mathcal{L}_{unsup} = \lambda_u \sum_{x_u \in D_U} KL(p^t_\phi(y|x_u), p_\phi(y|Aug(x_u))) \tag{8}$$

$$\mathcal{L}_{ST} = \sum_{x^{st}_i \in D^{ST}} KL(p_\phi(y|x^{st}_i), p_\phi(y|Aug(x^{st}_i))) \tag{9}$$

where λ_u is the weighting coefficient of consistency loss in UDA, and Aug is the default data augmentation policy in UDA. The framework of our method is illustrated in Algorithm 1.

5. Experiments

In this section, we evaluate the efficacy of our method on a diverse set of ID and OOD dataset pairs for open-set SSL. The result shows that our method outperforms current SOTA methods on various datasets and network architectures.

5.1. Datasets

Following conventional settings on OpenSet SSL [49, 34, 45, 3], we use the following two datasets to perform the experiments:

CIFAR-10: a dataset containing 10 classes of small-size (width * height $\leq$ 1,000) images. We use the same setting as [49]: 5,000 samples are split from the original training data as validation dataset; the remaining 45,000 samples are further split into labeled and unlabeled dataset; number of labeled dataset is chosen from $\{1000, 4000\}$. 10,000 OOD samples are added into unlabeled dataset from the following four datasets for each setting: Tiny ImageNet (TIN) [11], LSUN [48], Gaussian noise and Uniform Noise.

Tiny ImageNet: a dataset containing 200 different classes with larger-size (width $\times$ height $\geq$ 200 $\times$ 200) images. Here we set a more extreme ratio of ID : OOD samples as 1:9, only the last 20 classes are selected as ID classes, while the remaining 180 classes are used as OOD samples. 10,000 ID samples are then split into 1,000 labeled and 9,000 unlabeled samples. All images are resized to 224 x 224 during training.

5.2. Implementation Details

Backbone. Following [49, 34, 45, 3], we employ Wide-ResNet-28-2 [50] on CIFAR-10 dataset. For Tiny ImageNet, ResNet-50 [18], a network architecture widely used for ImageNet classification is adopted here.

DACT module. This module is a submodule of UDA [45]. The model is trained for 1,600 epochs by default. For CIFAR-10, batch sizes for labeled & unlabeled dataset are 36 and 960, softmax temperature is 0.8, λ_u is 5, learning rate is 1e-4, learning rate schedule is cosine learning rate decay schedule with learning rate warming up for 120 epochs. We use an SGD optimizer with nesterov momentum, momentum hyper-parameter is set to 0.9. For Tiny ImageNet, we subtly modify the hyper-parameters above: unlabeled batch size is reduced to 64, softmax temperature is 1.

Style transfer module. This module is based on AdaIN [21]. The style transfer network is an encoder-decoder framework, the encoder is a pretrained VGG-19 [39] network, and the decoder in a reversed architecture is trained with style dataset D_U and content dataset D_L. We train the model with default hyper-parameters. All images are resized to 224 x 224 in both model training and image generation procedures. For CIFAR-10, generated images are further resized to 32 x 32. The interpolation hyper-parameter β is randomly chosen from [0, 0.5]. Unlabeled OOD samples and labeled samples are treated as style and content dataset respectively. The whole D_U is regarded as OOD dataset in the beginning; after half of the total epochs (800 epochs), D_U is split into ID & OOD parts and style dataset is updated. The momentum hyper-parameter α is set to 0.8.

5.3. Results

Comparison with other methods. The results for the CIFAR-10 dataset are summarized in Table 2. Compared to MTCF [49] and DS3L [16], our method outperforms the others not only on classification performance but also on robustness: previous methods avoid disadvantage of OOD samples by reducing their weight softly (DS3L) or hardly (MTCF), and degradation of model performance is mitigated notably; however, our method tries to take advantages of OOD samples, and improves model performance by 3.35% and 0.92% respectively after using them.

More experiments and ablation studies. To verify the generalization of methods, we turn to Tiny ImageNet and change the backbone of models (ResNet-50). Using the official implementation of DS3L & MW-Net [38] (backbone of DS3L) and MTCF, our experiments show that both methods have unsatisfactory performance. Besides, hardly can I tune hyper-parameters because both methods are very time-consuming and memory-unfriendly, as is shown in Table 4. Also we could not find any references help guide the hyper-parameter tuning procedure for either method on ImageNet or Tiny ImageNet. Consequently, we only make several trials for each method, and report the best result of them. In contrast to the above two methods, ours is much faster and far more robust on OOD samples. As is shown in the 5-th row of Table 3, our method enhances model performance by 2.63%.

The quickly advancing field of Self-Supervised Learning [17, 8, 6, 7, 5] also motivates us to make better use of OOD samples for better pretrained models. We simply choose MoCo [17], a GPU-friendly method to pretrain the model with both D_L and D_U. The pretrained model is then used to initialize the network for subsequent tasks.

We perform an ablation study on Tiny ImageNet to better understand how each module works. We analyze the effect of components in our method and find that each module has an orthogonal contribution to the overall improvements, as is summarized in Table 3. We observe that: (1) Adding 90,000 OOD samples to D_U directly could bring about 1% improvement, and it again verifies the robustness of DACT; (2) Module of style transfer boosts the performance more than 1.5%; Apart from splitting module, it still contributes about 1% to improvement; Since the splitting module performs in a similar way as another widely-used SSL method Temporal Ensembling [28], it brings roughly 0.5% improvement as well; (3) The pretraining module also

Method	OOD dataset					Mean acc change
	Clean	LSUN	TIN	Gaussian	Uniform	
DS3L [16]	67.79 ± 0.27	69.74 ± 0.08	70.10 ± 0.47	62.86 ± 0.67	62.89 ± 1.65	↓ 1.39
MTCF [49]	90.67 ± 0.29	90.19 ± 0.47	89.85 ± 0.11	89.87 ± 0.08	89.80 ± 0.26	↓ 0.74
Ours	88.29 ± 0.25	**91.30 ± 0.36**	**91.10 ± 0.65**	**92.33 ± 0.59**	**91.82 ± 0.04**	↑ 3.35

(a) CIFAR-10 with 1000 labeled and 54000 unlabeled samples

Method	OOD dataset					Mean acc change
	Clean	LSUN	TIN	Gaussian	Uniform	
DS3L [16]	83.23 ± 0.07	82.89 ± 0.69	82.58 ± 0.14	80.44 ± 0.01	80.59 ± 0.03	↓ 1.61
MTCF [49]	93.30 ± 0.10	92.91 ± 0.03	93.03 ± 0.05	92.83 ± 0.04	92.53 ± 0.08	↓ 0.48
Ours	93.36 ± 0.40	**94.27 ± 0.21**	**93.84 ± 0.10**	**94.52 ± 0.07**	**94.50 ± 0.13**	↑ 0.92

(b) CIFAR-10 with 4000 labeled and 51000 unlabeled samples

Table 2: Accuracy(%) for CIFAR-10 and OOD dataset pairs using different methods.

w/ OOD	split	style trans	pretrain	top-1 acc
-	-	-	-	59.90±0.54
✓	-	-	-	60.95±0.55
✓	✓	-	-	61.50±0.12
✓	✓	✓	-	62.53±0.24
✓	-	-	✓	62.58±0.79
✓	✓	✓	✓	**65.70±0.16**

Table 3: Ablation studies on Tiny ImageNet.

Method	Time/trial	Device num	Top-1 acc
DS3L [16]	>1 week	8	4.50
MTCF [49]	>2 weeks	8	29.05
Ours	<40 hours	2	**65.70±0.16**

Table 4: Comparison of different methods on Tiny ImageNet. The device we used is NVIDIA Tesla V100.

shows the value of OOD dataset by enhancing performance for about 1.6% (the detailed setting of pretraining is left to the supplementary material); (4) The combination of all components improves totally 5.8% and shows great advantage on other open-set SSL methods. We measure the distribution gap between D_L&D_U quantitatively by mmd_{gap} to understand how style-transfer module works. As is shown in Table 5, distribution of unlabeled OOD samples is getting closer to D_L after style transfer, thus model performance is improved.

6. Conclusion and Discussion

In this paper we analyze the robustness of two fundamental SSL methods, PL and DACT, to the more realistic open-set SSL setting. Our study shows that the latter is more robust than the former by illustration in low-dimensional space and experiments in high-dimensional space. Besides, the proposed metric mmd_{gap} quantitatively measures the distribution gap between labeled & unlabeled dataset to better understand the influence of unlabeled OOD samples on model performance. Our method combines DACT with neural style transfer methods, to make better use of OOD samples for further performance improvement. Experiments on several open-set SSL benchmarks prove that our method achieve better performance than previous SOTA methods.

dataset	mmd_{gap}
split OOD samples	61.21 ± 0.57
transferred OOD samples	20.55 ± 0.78
total unlabeled samples	41.96 ± 0.49
total unlabeled & transferred OOD samples	38.79 ± 0.54

Table 5: Understand how module of style-transfer works on OOD samples by measuring mmd_{gap} from unlabeled dataset to labeled dataset D_L.

It is worth noticing that our style transfer module is only one way to decrease the distribution gap between labeled and unlabeled data. There may exists other efficient methods to be considered which is worth further studying. Also our experimental result in Table 1 reveals that in some extreme settings, synthetic noisy images could help improve model performance better than images from real scenarios. This interesting phenomenon is worth investigating further.

References

[1] Yuki M Asano, Christian Rupprecht, and Andrea Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. *arXiv preprint arXiv:1904.13132*, 2019.

[2] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*, 2019. 2

[3] 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, 5, 7

[4] Avleen S Bijral, Nathan Ratliff, and Nathan Srebro. Semi-supervised learning with density based distances. *arXiv preprint arXiv:1202.3702*, 2012. 3

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in Neural Information Processing Systems*, 33, 2020. 7

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 7

[7] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33, 2020. 7

[8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 7

[9] Yanbei Chen, Xiatian Zhu, Wei Li, and Shaogang Gong. Semi-supervised learning under class distribution mismatch. In *AAAI*, pages 3569–3576, 2020. 1, 2

[10] 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. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 5, 7

[12] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 2

[13] M Ehsan Abbasnejad, Anthony Dick, and Anton van den Hengel. Infinite variational autoencoder for semi-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5888–5897, 2017. 2

[14] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005. 2

[15] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex J Smola. A kernel method for the two-sample-problem. In *Advances in neural information processing systems*, pages 513–520, 2007. 2, 5

[16] Lan-Zhe Guo, Zhen-Yu Zhang, Yuan Jiang, Yu-Feng Li, and Zhi-Hua Zhou. Safe deep semi-supervised learning for unseen-class unlabeled data. In *Proceedings of the 37th International Conference on Machine learning (ICML)*, 2020. 1, 2, 7, 8

[17] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 6, 7

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 7

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2

[20] Cheng-An Hou, Yao-Hung Hubert Tsai, Yi-Ren Yeh, and Yu-Chiang Frank Wang. Unsupervised domain adaptation with label and structural consistency. *IEEE Transactions on Image Processing*, 25(12):5552–5562, 2016. 5

[21] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization.

In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 6, 7

[22] Philip TG Jackson, Amir Atapour Abarghouei, Stephen Bonner, Toby P Breckon, and Boguslaw Obara. Style augmentation: data augmentation via style randomization. In *CVPR Workshops*, pages 83–92, 2019. 2

[23] Bo Jiang, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. Semi-supervised learning with graph learning-convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11313–11320, 2019. 2

[24] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. *Advances in neural information processing systems*, 27:3581–3589, 2014. 2

[25] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 2

[26] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1, 5

[27] Chia-Wen Kuo, Chih-Yao Ma, Jia-Bin Huang, and Zsolt Kira. Featmatch: Feature-based augmentation for semi-supervised learning. *arXiv preprint arXiv:2007.08505*, 2020. 2

[28] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016. 2, 7

[29] 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, 2013. 2, 3

[30] Suichan Li, Bin Liu, Dongdong Chen, Qi Chu, Lu Yuan, and Nenghai Yu. Density-aware graph for deep semi-supervised visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13400–13409, 2020. 2, 3

[31] Wanyu Lin, Zhaolin Gao, and Baochun Li. Shoestring: Graph-based semi-supervised classification with severely limited labeled data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4174–4182, 2020. 2

[32] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 2200–2207, 2013. 2, 5

[33] Augustus Odena. Semi-supervised learning with generative adversarial networks. *arXiv preprint arXiv:1606.01583*, 2016. 2

[34] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in neural information processing systems*, pages 3235–3246, 2018. 1, 2, 7

[35] Hieu Pham, Qizhe Xie, Zihang Dai, and Quoc V Le. Meta pseudo labels. *arXiv preprint arXiv:2003.10580*, 2020. 2

[36] Yichun Shi, Xiang Yu, Kihyuk Sohn, Manmohan Chandraker, and Anil K Jain. Towards universal representation learning for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6817–6826, 2020.

[37] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019. 3

[38] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Advances in Neural Information Processing Systems*, pages 1919–1930, 2019. 2, 7

[39] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[40] 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. 2

[41] 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. 2

[42] Ricardo Vilalta and Youssef Drissi. A perspective view and survey of meta-learning. *Artificial intelligence review*, 18(2):77–95, 2002. 2

[43] Wei Wang, Haojie Li, Zhihui Wang, Jing Sun, Zhengming Ding, and Fuming Sun. Importance filtered cross-domain adaptation. 5

[44] Xiao Wang, Daisuke Kihara, Jiebo Luo, and Guo-Jun Qi. Enaet: Self-trained ensemble autoencoding transformations for semi-supervised learning. *arXiv preprint arXiv:1911.09265*, 2019. 2

[45] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2, 5, 7

[46] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020. 2

[47] Chunyan Xu, Zhen Cui, Xiaobin Hong, Tong Zhang, Jian Yang, and Wei Liu. Graph inference learning for semi-supervised classification. *arXiv preprint arXiv:2001.06137*, 2020. 2

[48] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 5, 7

[49] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *European Conference on Computer Vision*, pages 438–454. Springer, 2020. 1, 2, 4, 5, 6, 7, 8

[50] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 7

[51] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 5

[52] Liheng Zhang and Guo-Jun Qi. Wcp: Worst-case perturbations for semi-supervised deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3912–3921, 2020. 2

[53] Nanxuan Zhao, Zhirong Wu, Rynson WH Lau, and Stephen Lin. What makes instance discrimination good for transfer learning? *arXiv preprint arXiv:2006.06606*, 2020.

[54] Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Time-consistent self-supervision for semi-supervised learning. In *International Conference on Machine Learning (ICML)*, 2020. 2