A Unified Generative Adversarial Network Training via Self-Labeling and Self-Attention

Tomoki Watanabe [1] **Paolo Favaro** [2]

Abstract

We propose a novel GAN training scheme that can handle any level of labeling in a unified manner. Our scheme introduces a form of artificial labeling that can incorporate manually defined labels, when available, and induce an alignment between them. To define the artificial labels, we exploit the assumption that neural network generators can be trained more easily to map nearby latent vectors to data with semantic similarities, than across separate categories. We use generated data samples and their corresponding artificial conditioning labels to train a classifier. The classifier is then used to self-label real data. To boost the accuracy of the self-labeling, we also use the exponential moving average of the classifier. However, because the classifier might still make mistakes, especially at the beginning of the training, we also refine the labels through self-attention, by using the labeling of real data samples only when the classifier outputs a high classification probability score. We evaluate our approach on CIFAR-10, STL-10 and SVHN, and show that both self-labeling and self-attention consistently improve the quality of generated data. More surprisingly, we find that the proposed scheme can even outperform class-conditional GANs.

1. Introduction

Generative Adversarial Networks (GAN) (Goodfellow et al., 2014; Brock et al., 2019; Karras et al., 2019) provide an attractive approach to constructing generative models that output samples of a target distribution. In their most basic form, these models consist of two neural networks, a generator and a discriminator. The first network is trained to generate samples from some latent representation (typically a sample from a Gaussian distribution), while the second network is trained to distinguish real samples x_r from the generated samples x_f (see network D in Figure 1 (a)). The most effective GANs seem to benefit greatly from class conditioning. The class information is provided as input to the generator and either injected into the discriminator as an input (Mirza & Osindero, 2014) or through intermediate layers (Reed et al., 2016) or via a projection (Miyato & Koyama, 2018) or an auxiliary loss (Odena et al., 2017; Kavalerov et al., 2019) (see *e.g.*, Figure 1 (b)). The family of these generators is generically called *conditional GANs* (cGAN).

In particular, in AC-GAN (Odena et al., 2017) (see Figure 1 (b)) one uses a discriminator with two outputs, one for the classification of input images into real or fake and the other for classification into multiple categories (see the label y in the green box in Figure 1 (b)). We hypothesize that providing the class information helps the training of the generator, because the neural network architecture of the generator tends to map similar latent vectors to data samples that are semantically related. Thus, we expect a gradient-based training to converge more easily to a good set of network parameters with class conditioning than without it. This suggests that one might be able to train a generator in a conditional manner even when manually defined labels are not available. Based on this assumption, we train a generator conditioned on an artificial set of labels and simultaneously learn its inverse mapping through a classifier, which we call *teacher*. The teacher is trained only on synthetic data (networks G-inference and C-training in Figure 1 (d)), and then it is applied to real data samples to obtain their corresponding artificial labels (network C-inference in Figure 1 (d)), a process that we call *self-labeling*. Because the accuracy of a trained classifier is limited, using all the artificially labeled real data would not help the generator especially at the beginning of the training, when all predicted labels may be highly inaccurate. Hence, we introduce a way to select data, where the label consistency is high. To do so we introduce a *self-attention* mechanism that is based on selecting samples x_r whose estimated label probability is above a given threshold (blue box in Figure 1 (d)). The selected samples and their corresponding synthetic labels

This work has been done while the first author was visiting the University of Bern. [1]Toshiba Corporation, Kawasaki, Japan [2]University of Bern, Bern, Switzerland. Correspondence to: Tomoki Watanabe <tomoki8.watanabe@toshiba.co.jp>, Paolo Favaro <paolo.favaro@inf.unibe.ch>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

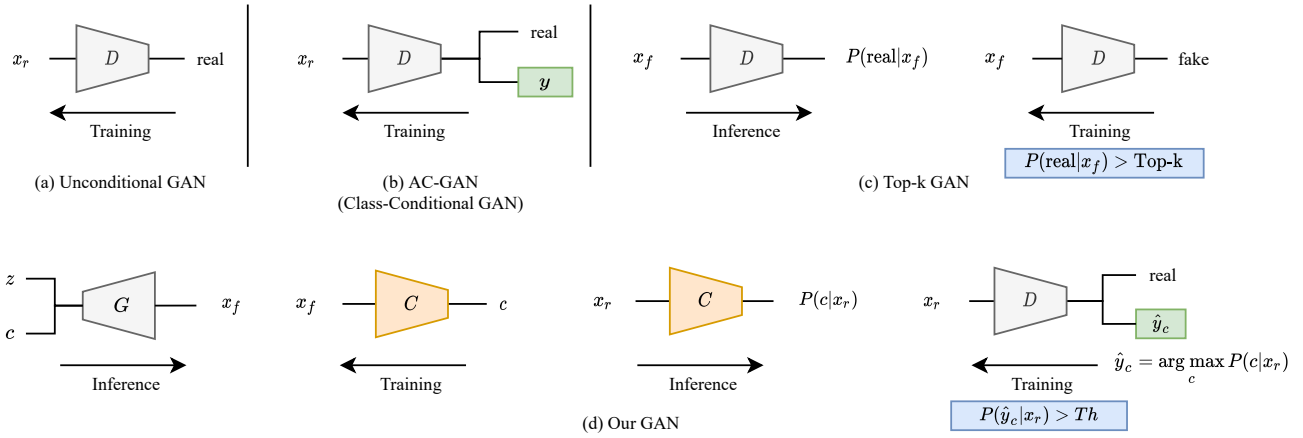


Figure 1. **Comparison of GAN variants and our GAN training scheme**: (a) Unconditional GAN discriminates real samples x_r and fake samples x_f; (b) AC-GAN (Odena et al., 2017) learns to predict *supervised* labels y; (C) Top-k GAN (Sinha et al., 2020) updates the discriminator D with fake samples where the discriminator D outputs high confidence, *i.e.*, $P(\text{real}|x_f) >$ Top-k; (d) Our proposed GAN scheme learns to predict *artificial* labels $\hat{y}_c$ defined by the teacher classifier C and updates the discriminator D on *real* samples, where the teacher classifier C outputs high confidence, *i.e.*, $P(c|x_r) \geq Th$. We train the teacher classifier only on fake samples x_f and conditional labels c of the generator G.

$\hat{y}_c$ are then used to train the discriminator (network D and green box in Figure 1 (d)). This idea is similar to the one exploited by Top-k GAN (Sinha et al., 2020), which uses the output of the discriminator as the confidence for the labeling of fake images (see Figure 1 (c)), while we apply it instead to real images. Moreover, we use the EMA (exponential moving average) of the teacher during inference to further improve its labeling accuracy. This technique has been used in semi-supervised and unsupervised learning methods to improve the classification/clustering accuracy (Tarvainen & Valpola, 2017; Sohn et al., 2020; He et al., 2020).

So far, we have described a GAN training method that exploits the same benefits that conditional GANs enjoy, but without using manually labeled data. When data is partially or fully labeled, it is desirable to take advantage of the available information. Our scheme can seamlessly integrate such available labels and also indirectly transfer their categorical information to the artificial labels. This is possible because our artificial labels are defined relative to the generator and the generator can adapt to a new reference during training.

We evaluate our method on CIFAR-10 (Krizhevsky & Hinton, 2009), STL-10 (Coates et al., 2011), and SVHN (Netzer et al., 2011) datasets using the BigGAN model (Brock et al., 2019) and show that our method improves the quality of the generated images in terms of the FID (Fréchet Inception Distance) score (Heusel et al., 2017). Our method achieves better FID scores than the state-of-the-art GAN and even that of fully supervised cGAN methods on the CIFAR-10 dataset. Our contributions can be summarized as follows:
1) A unified GAN training that can handle any level of labeling in a unified manner by using: **Self-labeling**: a method to automatically assign labels to real data samples, and **Self-attention**: a method to select real data samples with highly consistent synthetic labels;
2) Consistent improvement in the FID scores across several datasets (evaluation on CIFAR-10, STL-10, and SVHN);
3) The ability to outperform class-conditional GANs (fully labeled dataset).

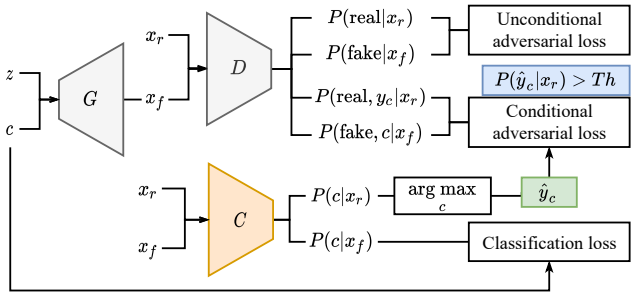


Figure 2. **Overall architecture of our proposed unified GAN training.** The important components of our scheme are: 1) The conditional training of the generator G, which is based on artificial labels c; 2) A classifier C, which is trained only on synthetic data samples x_f and their corresponding artificial labels c; 3) A discriminator D that is trained on artificial labels c for fake data x_f, real labels y for real data x_r when available, and generated labels $\hat{y}_c$ for real data x_r that do not have manually defined labels.

2. Prior Work

Generative Adversarial Networks. The unconditional GAN (Brock et al., 2019) consists of two networks, a generator G and a discriminator D. The generator outputs a fake image from a (vector) noise instance and the discriminator distinguishes input images as real or fake. The discriminator

and the generator are trained by minimizing the following adversarial loss terms

$$L_D^U = -E_{x_r}[\log P(\text{real}|x_r)] - E_z[\log P(\text{fake}|G(z))], \quad (1)$$

$$L_G^U = -E_z[\log P(\text{real}|G(z))], \quad (2)$$

where x_r is a random variable representing real images (and is used also to indicate samples from the distribution of real images) and z is a random variable with a fixed distribution, typically the Normal distribution $\mathcal{N}(0, I_d)$ (and is also used to indicate instances from that distribution). $P(\text{class}|\text{input})$ indicates the probability that `input` belongs to the class `class`. Class-conditional GANs (*e.g.*, AC-GAN (Odena et al., 2017)) are instead trained by minimizing the following adversarial loss terms

$$L_D^Y = -E_{x_r,y}[\log P(\text{real}, y|x_r)] - E_{z,c}[\log P(\text{fake}, c|G(z,c))], \quad (3)$$

$$L_G^C = -E_{z,c}[\log P(\text{real}, c|G(z,c))], \quad (4)$$

where y and c are supervised labels and artificial conditional labels respectively. The discriminator of cGAN does not only classify images into real/false, but also estimates the labels of the input images. The training requires supervised labels for all real images. An alternative approach to using real labels is proposed in self-conditional GANs, which are cGANs trained with unlabeled samples, where artificial labels are defined through some heuristic tasks. An example of this approach is to use the orientation of rotated images as a label for conditioning (Chen et al., 2019b; Tran et al., 2019). In comparison to these methods, we obtain our labels implicitly from the generator.

Semi-Supervised Learning. Semi-supervised learning (SSL) is a branch of machine learning where the training data is a mix of labeled samples (typically, a small amount) and of unlabeled samples (typically, in larger number compared to the labeled samples) (van Engelen & Hoos, 2020). Recent methods that work in the SSL regime are RemixMatch (Berthelot et al., 2019) and FixMatch (Sohn et al., 2020).

In particular, of interest to us is FixMatch (Sohn et al., 2020), which trains a classifier C by minimizing two cross-entropy loss terms: a supervised loss on labeled samples x^L and an unsupervised loss on unlabeled samples x^U as

$$L_{label} = H[y, C(A(x^L))] + H[\hat{y}, C(A(x^U))], \quad (5)$$

where y and $\hat{y}$ are supervised labels and artificial labels respectively, and A is an image augmentation function. As in our approach, the artificial labels are assigned by a classifier with the parameters $\bar{\theta}_C$ of the running average model (Tarvainen & Valpola, 2017) via

$$\hat{y} = \arg\max_i C_i(\alpha(x^U); \bar{\theta}_C). \quad (6)$$

where α is a *weak* image augmentation function, *i.e.*, with image transformations close to the identity.

Other important recent SSL methods for GAN training are the work of (Lucic et al., 2019; Noroozi, 2020). As in our approach, they train a network for classification/clustering, which is then used to provide labels for conditioning. However, while they train the classifier on real samples, we train it only on fake samples, and, to the best of our knowledge, are the first ones to do so.

3. A Unified GAN Training

Our unified GAN training uses a cGAN as backbone, where the discriminator classifies the input into real/fake and image categories. cGANs require semantic labels for training. While the labels of generated data are implicitly defined, the labels of real data are either provided through manual labeling or through our unsupervised self-labeling and self-attention procedures.

3.1. Self-Labeling and Self-Attention

Our objective is to assign artificial labels to unlabeled real images that are used in the conditional adversarial loss. To this purpose, we train a classifier C, which we call *teacher*, on fake images $x_f = G(z, c)$, where the class-correspondence is known. We train the teacher with the cross-entropy loss

$$L_C = H[c, C(A(x_f))]. \quad (7)$$

where H denotes the entropy, the fake image is obtained via $x_f = G(z, c)$ with $z \sim \mathcal{N}(0, I_d)$, and c is a random variable (with a discrete Uniform distribution) and also denotes its instance. Since the teacher may not be a perfect inverse of G with respect to the conditional label c, we introduce two methods to ensure a high classification accuracy.

First, we use the EMA parameters of the teacher to compute the artificial labels $\hat{y}_c$ of real images, *i.e.*, we compute

$$\hat{y}_c = \arg\max_i C_i(\alpha(x_r); \bar{\theta}_C). \quad (8)$$

Second, because the artificial labels $\hat{y}_c$ are inaccurate especially during the early epochs of the training, we introduce a selection mechanism called self-attention. We first define the *reliability* of the artificial labels via the softmax of the classifier output

$$p_c = \frac{\exp(C_{\hat{y}_c}(\alpha(x_r); \bar{\theta}_C))}{\sum_{i=1}^K \exp(C_i(\alpha(x_r); \bar{\theta}_C))}, \quad (9)$$

where K is the number of the artificial classes. As we show in the experiments, the reliability yields a high value with real images that are distinctively similar to generated fake

Table 1. **Ablation study on the unlabeled CIFAR-10.** We show that both self-labeling and self-attention are necessary to improve the GAN training.

	SELF-LABELING	SELF-ATTENTION	FID-5EP	$\overline{\text{FID}}$
(A)	-	-	7.45 ± 0.17	6.96 ± 0.20
(B)	✓	-	7.74 ± 0.20	7.00 ± 0.34
(C)	✓	✓	7.28 ± 0.11	6.81 ± 0.13

images, and when these fake images are well separated into different clusters. Then, self-attention selects real images x_r such that $p_c \geq Th$, where the threshold $Th \in [0, 1]$.

3.2. Training with Artificial (and Real) Labels

The conditional adversarial loss for conventional cGANs uses the supervised class labels y and artificial labels c for real images and fake images respectively as shown in Eq. (3). With real images without supervised class labels y, we use instead the artificial labels $\hat{y}_c$.

The discriminator has 2 heads, one for the unconditional fake/real adversarial loss and another for the conditional adversarial loss. The losses for the discriminator L_D and the generator L_G are simply the sum of the corresponding conditional and unconditional losses

$$L_D = L_D^U + L_D^C, \quad L_G = L_G^U + L_G^C. \tag{10}$$

The loss functions L_D^U, L_G^U, and L_G^C are shown in Eqs. (1), (2) and (4). The loss function L_D^C instead is defined so that it can be applied to a dataset with any degree of labeling (from 0% to 100%) as

$$\begin{aligned} L_D^C = & -E_{\{x_r, y | \text{with label}\}}[\log P(\text{real}, y | x_r)] \\ & -E_{\{x_r, \hat{y}_c | \text{no label} \wedge p_c \geq Th\}}[\log P(\text{real}, \hat{y}_c | x_r)] \\ & -E_{x_f, c}[\log P(\text{fake}, c | x_f)]. \end{aligned} \tag{11}$$

The loss function uses artificial labels $\hat{y}_c$ obtained from the teacher as shown in Eq. (8). As explained in subsection 3.1, we calculate the loss only on images where the reliability p_c is higher than a threshold Th, because unreliable labels have an adverse effect on the training of the discriminator. We update the teacher, the discriminator, and the generator simultaneously via Eqs. (7), (9) and (10). We can train cGAN on unlabeled dataset, because these loss terms are well-defined even in the absence of real labels.

We show the network architecture of our method in Figure 2. The components were already introduced in Figure 1 (d).

3.3. Implementation

Teacher: We employ Resnet18 (He et al., 2016) with a head of K outputs as the backbone of the teacher. We use the multi-class cross entropy loss for the classification loss. The strong image augmentation function A is RandAugment (Cubuk et al., 2020) and the weak image augmentation function α consists of random horizontal-flips and shifts between ± 4 pixels.

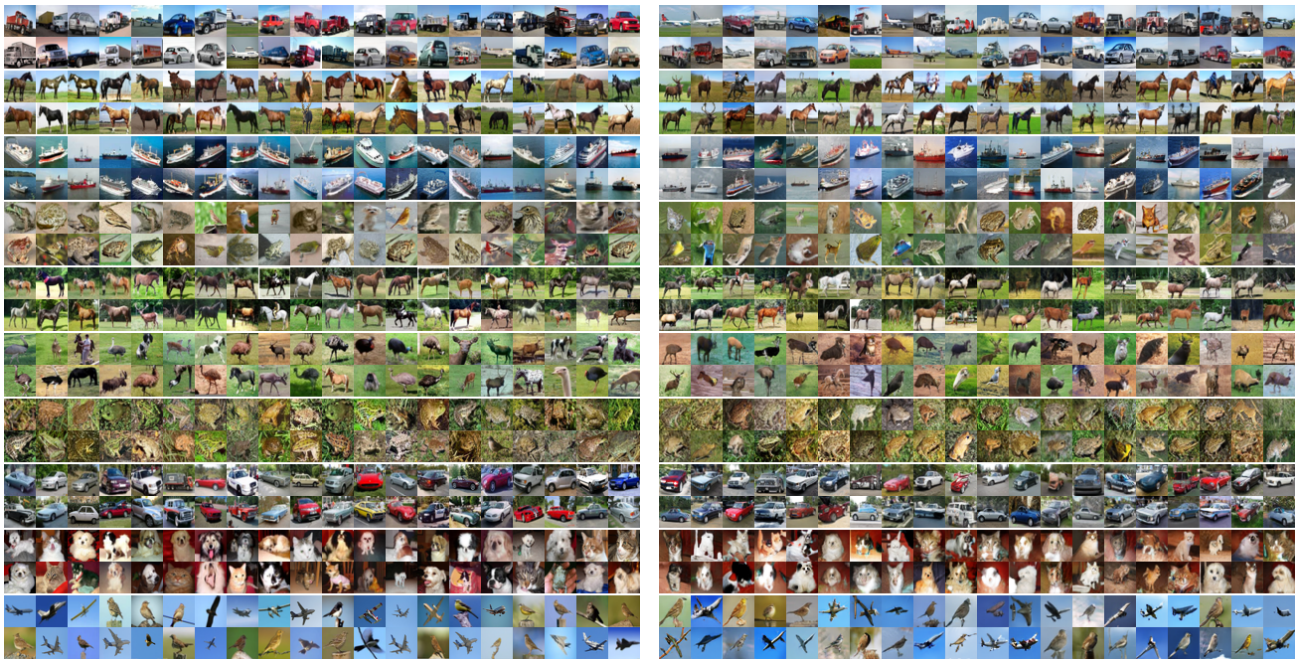
Algorithm 1 Unified GAN Training

Input: Parameters of the generator θ_G, the discriminator θ_D, the teacher θ_C, the number of artificial classes K, and the threshold Th
for the number of training iterations **do**
  Sample batch $z \sim p(z)$, $c \sim p(c)$, $x_r \sim p_{real}(x_r)$
  Step1. Update teacher: subsection 3.1
  $L_C \leftarrow \text{SoftmaxCrossEntropy}(c, C(A(x_f))$ Eq. (7)
  $\theta_C \leftarrow \text{MomentumOptimizer}(L_C)$
  $\bar{\theta}_C \leftarrow \text{ExponentialMovingAverage}(\theta_C)$
  $\hat{y}_c \leftarrow \arg\max_i C_i(\alpha(x_r); \bar{\theta}_C)$ (self-labeling) Eq. (8)
  $p_c \leftarrow \text{Softmax}(C_{\hat{y}_c}(\alpha(x_r); \bar{\theta}_C))$ Eq. (9)
  Step 2. Update cGAN: subsection 3.2
  $x_f \leftarrow G(c, z)$
  $L_D^U \leftarrow \text{Hinge}(D^U(x_r)) + \text{Hinge}(-D^U(x_f))$ Eq. (1)
  $S \leftarrow$ **if** $p_c \geq Th$ **then** 1 **else** 0 (self-attention)
  $L_D^C \leftarrow S \cdot \text{MultiClassHinge}(\hat{y}_c, D^C(x_r))$
    $+ \text{MultiClassHinge}(c + K, D^C(x_f))$ Eq. (11)
  $\theta_D \leftarrow \text{AdamOptimizer}(L_D^U + L_D^C)$
  $L_G^U \leftarrow \text{Hinge}(D^U(G(c, z)))$ Eq. (2)
  $L_G^C \leftarrow \text{MultiClassHinge}(c, D^C(G(c, z))$ Eq. (4)
  $\theta_G \leftarrow \text{AdamOptimizer}(L_G^U + L_G^C)$
end for

Table 2. Comparison on the unlabeled CIFAR-10.

METHOD	$\overline{\text{FID}}$
BIGGAN (BROCK ET AL., 2019)	14.73
SS-GAN (CHEN ET AL., 2019A)	15.60
MS-GAN (TRAN ET AL., 2019)	11.40
TOP-K GAN (SINHA ET AL., 2020)	13.34
ICR-GAN (ZHAO ET AL., 2020C)	9.21
SLCGAN (NOROOZI, 2020)	8.95
TOP-K ICR-GAN (SINHA ET AL., 2020)	8.57
OURS	**6.81**

Conditional GAN: We employ BigGAN (Brock et al., 2019) for the backbones of the generator (to which we add the conditional label input) and the discriminator. We also add a fully connected layer of $2K$ outputs and a U-Net decoder (Schönfeld et al., 2020) of $W \times H$ outputs, *i.e.*, the same size of the training image after the global pooling layer of the discriminator, as the heads for conditional adversarial loss and unconditional adversarial loss respectively. We use the hinge loss for the unconditional adversarial loss and the multi-class hinge loss for the conditional adversarial loss. To stabilize the training, we apply the following GAN training techniques: differentiable augmentation (Zhao et al., 2020a),



(a) Real images (b) Fake images

Figure 3. **(a) Real images and (b) Fake images on the unlabeled CIFAR-10.** From the top to the bottom, every pair of image rows corresponds to the same artificial label, which was estimated by the teacher. We observe that images in the same pair of rows contain semantically similar objects.

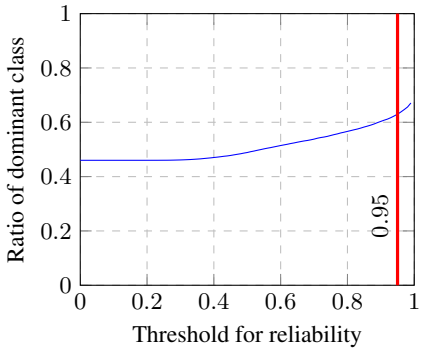


Figure 4. **On the role of self-attention.** The plot shows the average ratio of the dominant class contained in the set of selected real images grouped through the artificial labels. A high rate means that each artificial label is consistent with one real label on average.

Table 3. FID on unlabeled CIFAR-10

	BASELINE	K=2	K=5	K=10	K=50
FID	6.96	6.07	**5.97**	6.81	7.54

R1 gradient penalty (Mescheder et al., 2018), spectral normalization (Miyato et al., 2018), and the U-Net CutMix consistency regularization (Schönfeld et al., 2020).

We show the pseudo code of the training in Algorithm 1. The parameters of the generator θ_G, the discriminator θ_D, and the teacher θ_C are initialized with Xavier uniform initialization (Glorot & Bengio, 2010). Firstly, we update the parameters θ_C of the teacher and assign the artificial labels for the real images. Secondly, we update the parameters of the cGAN (θ_G and θ_D) by using the artificial labels obtained in the first step. D^U and D^C represent the discriminator's heads for the unconditional/conditional adversarial loss. Then, we repeat the above 2 steps for the number of training iterations.

4. Experiments

We evaluate our method on CIFAR-10, STL-10, and SVHN by using FID scores as a quantitative measure and also visualize samples for a qualitative assessment. The FID scores are computed by using the official implementation (Heusel et al., 2017) on 50,000 generated samples. Moreover, we use the FID-5ep, which is the FID averaged over the last 5 epochs of 5 evaluation runs and also the $\overline{\text{FID}}$, which is the average of the lowest FIDs in 5 evaluation runs. We train the network on a single GPU: GeForce RTX 2080ti for CIFAR-10 and SVHN and Quadro RTX 6000 for STL-10.

To train the teacher we use Nesterov's momentum optimizer with a batch size of 64, momentum 0.9, $K = 10$, EMA decay of 0.999, and the number of epochs is 40 and 60 on the unlabeled and labeled datasets respectively. To train the cGAN we use Adam's optimizer with a batch size of 128 for the loss with the artificial labels and of 64 for the other losses, $Th = 0.95$, and the gradient penalty weight is 10. We use the large batch size for the unlabeled real images,

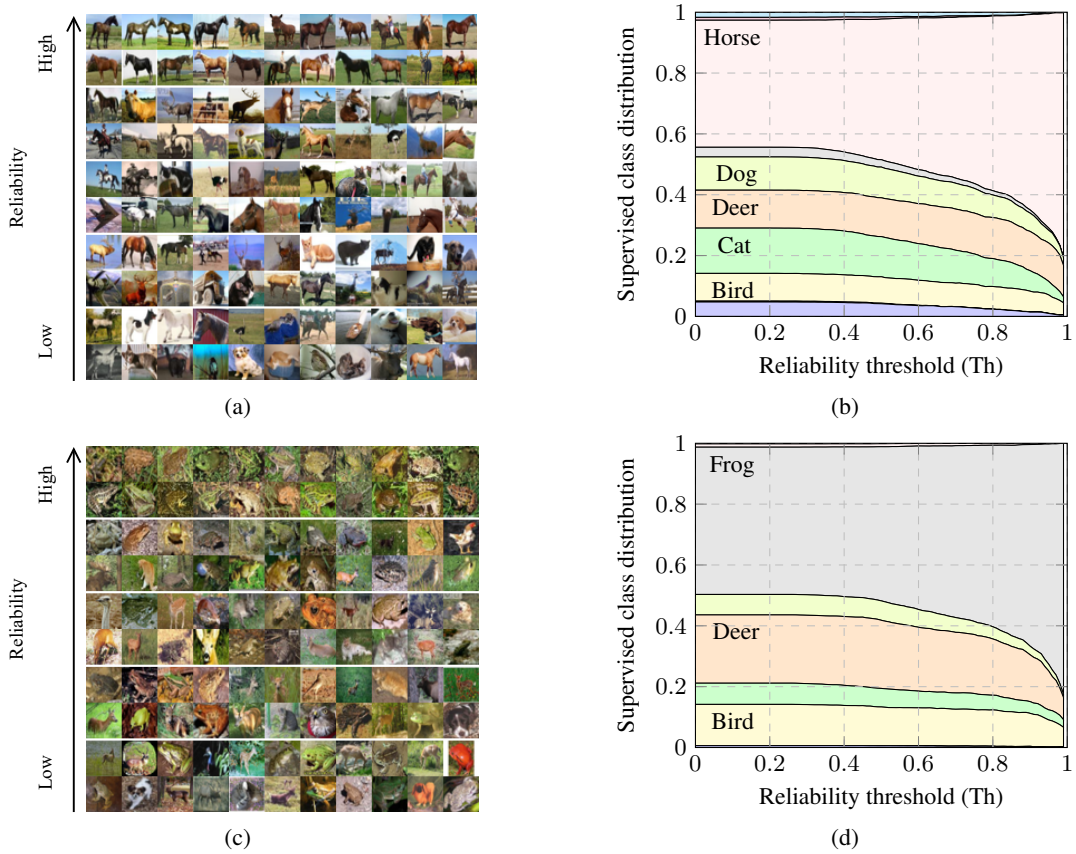


Figure 5. **Two examples of the reliability of the artificial labels.** (a) & (c): Pairs of rows with samples of real images assigned to *a single artificial label* and selected via self-attention (for reliability thresholds $Th = 0$ (bottom), $0.25, 0.5, 0.75$, and 1 (top)). The top and bottom rows correspond to high and low reliability of the artificial labels respectively. The top row pair contains images that are more consistent with a single real class (`Horse` in (a) and (b), and `Frog` in (c) and (d)) than images in the bottom row pair. (b) & (d): The relative distribution of real classes on images from (a) and (c) respectively. The ratio of the dominant real class (`Horse` in (b) and `Frog` in (d)) grows as the threshold for the reliability p_c increases. This shows a desirable alignment between the artificial labels and the real labels especially when the threshold Th is high.

Table 4. **Ablation study on the labeled CIFAR-10.** We calculate the conditional adversarial loss with (A) the artificial labels, (B) the real labels, and (C) both labels.

	LABELS	FID-5EP	$\overline{\text{FID}}$
(A)	ARTIFICIAL	7.28 ± 0.11	6.81 ± 0.13
(B)	REAL	5.04 ± 0.08	4.57 ± 0.10
(C)	ARTIFICIAL & REAL	4.72 ± 0.06	4.35 ± 0.05

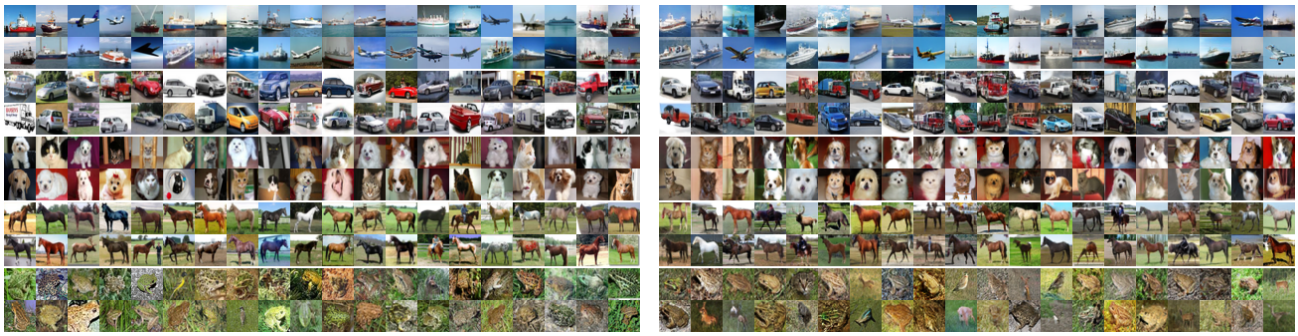
Table 5. Comparison on the labeled CIFAR-10.

METHOD	$\overline{\text{FID}}$
BIGGAN (BROCK ET AL., 2019)	9.06
DIFFAUG GAN (ZHAO ET AL., 2020A)	8.56
MHINGE GAN (KAVALEROV ET AL., 2019)	6.40
FQ-GAN (ZHAO ET AL., 2020B)	5.39
OURS	**4.35**

because we select a subset via self-attention.

Unlabeled CIFAR-10. We evaluate the effectiveness of self-labeling and self-attention on CIFAR-10 and summarize the results in Table 1. The results show that (B) unreliable artificial labels hurt the performance and (C) refined artificial labels help the training of the GAN compared to using (A) no artificial labels. Our method improves the FID score averaged over last 5 epochs from 7.45 to 7.28 and the best FID score averaged over 5 runs from 6.96 to 6.81.

In Figure 3a, we show real images grouped by their artificial labels, which were learned without supervision. The images are selected via self-attention. By starting from the top, every pair of rows corresponds to one artificial label. We can see that every artificial label identifies images with similar objects, but also that objects across separate labels differ substantially. For example, the first, the third and the last groups contain an object on the ground, in the sea, and in the sky respectively, and the third and seventh groups contain ships and cars respectively. In Figure 3b, we also show in



(a) Real images (b) Fake images

Figure 6. **(a) Real images and (b) Fake images on the unlabeled CIFAR-10 with $K = 5$ artificial classes, which is half of the actual number of supervised classes.** From the top to the bottom, every pair of image rows corresponds to the same artificial label, which was estimated by the teacher. We observe that the teacher assigns the same artificial label to visually similar objects such as `Airplane` and `Ship`, `Automobile` and `Truck`, and `Cat` and `Dog`.

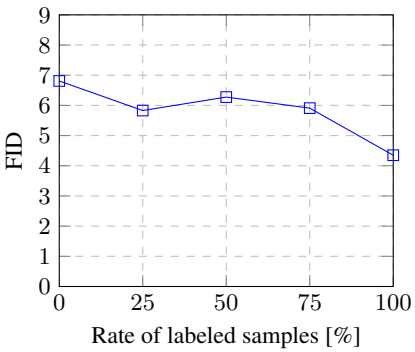


Figure 7. **FID for different real label rates (in percent with respect to the complete set of labels) in CIFAR-10.** Our method can handle different levels of labeling in a unified manner. We find that the FID tends to improve as more real labels are available.

Table 6. **FID on STL-10.** The STL-10 dataset contains 100,000 unlabeled images and 500 labeled images.

DATASET	BASELINE	+OURS
UNLABELED	32.85	30.91
+0.5% LABELED	48.49	43.29

the same manner images generated using the artificial labels (as input to the generator). Notice the broad diversity and the strong similarity between fake and real images in terms of artificial categories.

To explain the role of the proposed reliability measure p_c (see Eq. (9)) for self-attention, we sort real images with the same dominant artificial label based on the magnitude of the reliability. We show in Figure 5 an evaluation of the consistency between real and artificial labels for two randomly chosen artificial labels. Through visual inspection we find that these labels correspond mostly to the `Horse` (Figure 5a) and `Frog` (Figure 5c) categories. The top and bottom rows correspond to images of high and low reliability respectively. One can observe the higher semantic class consistency (*i.e.*, more `Horse` images in Figure 5a and more `Frog` images in Figure 5c), when the reliability is high. For a more quantitative measure of this consistency, in Figure 5b and Figure 5d we use labeled data to show the relative distribution of real samples selected with different reliability thresholds (Th). We notice that the ratio of the dominant classes `Horse` and `Frog` grows as the threshold is increased, which suggests a consistency between the clustering induced by artificial labels and real labels as well as a correlation between the reliability p_c and the purity measure of the selected samples. Moreover, in Figure 4, we show the rate of the dominant class averaged over all artificial labels. The plot shows that the ratio of the dominant class (that is consistent with a manually defined class) grows with the reliability threshold. We also indicate with a red vertical bar the chosen threshold in our implementation. The threshold defines a trade-off between high class consistency

Table 7. **Comparison on the unlabeled STL-10.** The methods below do not use the same generator, but the number of the convolutional layers is the same. Our FID is comparable to that of other methods, although we did not use network architecture search.

METHOD	FID
SNGAN (MIYATO ET AL., 2018)	40.1
AUTOGAN (GONG ET AL., 2019)	31.01
DEGAS (DOVEH & GIRYES, 2019)	28.76
OURS	30.91

Table 8. **FID on the unlabeled STL-10 and SVHN.** Our method improves the FID scores on both datasets.

DATASET	BASELINE	+OURS
STL-10	32.85	30.91
SVHN	2.44	2.19

(a) Real images (b) Fake images

Figure 8. **(a) Real images and (b) Fake images on the labeled CIFAR-10.** From the top to the bottom, every pair of image rows corresponds to the same artificial label, which was estimated by the teacher. Images in the same pair of rows contain semantically similar objects and are aligned with the real labels (the real labels corresponding to the same artificial label index are shown on the left).

and number of selected real samples. As also shown previously in Table 1, this selection process is quite essential to self-labeling.

Finally, we compare our method with the state-of-the-art methods for unsupervised GAN training on CIFAR-10 in Table 2. The methods use different loss functions, but share the same BigGAN generator. Although our generator uses the conditional label input, the basic backbone is the same. Our proposed training shows a significant $\overline{\text{FID}}$ improvement over the previous state-of-the-art (from 8.57 to 6.81).

To evaluate the effect of the number of the artificial classes K, in Table 3 we show the results learned with several settings of K on CIFAR-10. The best FID 5.97 is obtained with $K = 5$, which is smaller than the number of real classes (10). In Figure 6, we show real and fake images learned with $K = 5$, similarly to what shown in Figure 3. We can see that real classes that are difficult to tell apart are grouped together. For example, the teacher clusters the following pairs `Airplane` and `Ship`, `Automobile` and `Truck`, and `Cat` and `Dog`. We find that the confidence of the teacher on artificial labels is higher with $K = 5$ than with $K = 10$, which is the true number of real classes. Also, there is much less confidence with $K = 50$. This further supports the argument that reducing the ambiguities of the artificial label helps the discriminator, and then in turn the generator. The artificial labels contain less information with $K = 2$ than with $K = 5$, but the difference of the FID is small. The result suggests that a large K is suitable as long as the estimation of the artificial labels is confident.

Labeled CIFAR-10. As shown in Table 4, our method also improves the FID score when training on labeled datasets. The first column shows the type of labels used in the conditional adversarial loss. We calculate the conditional adversarial loss with either (a) the artificial labels, (b) the real labels, or (c) both of them. The result using both labels yields the best FID. Our method improves the baseline cGAN on the FID averaged on the last 5 epochs from 5.04 to 4.72 and the best $\overline{\text{FID}}$ from 4.57 to 4.35. The results show that the artificial labels integrate naturally with the real labels and further boost the performance of the generator.

In Figure 7, we also evaluate our method in the semi-supervised learning settings, by using a partially labeled CIFAR-10 dataset. We randomly select 25%, 50%, and 75% of the available real labels to train the cGAN with our method. The rates of 0% and 100% mean unlabeled and fully labeled respectively. We observe that the FID tends to improve with an increasing number of real labels.

In Figure 8, we show the real and generated images grouped by the artificial labels $\hat{y}_c$ (estimated by the teacher) and the artificial conditional labels c respectively. Figure 8a shows qualitatively that every group corresponds to some meaningful real class although the teacher is trained only on fake images. Figure 8b shows that the conditional labels c seem to seamlessly align with the available real labels. To measure the degree of alignment between artificial and real labels, we compute the classification accuracy on test real images after finding the optimal correspondence between the predicted and ground truth labels. We find that the

(a) Real images (b) Fake images

Figure 9. **(a) Real images and (b) Fake images on the unlabeled STL-10.** From the top to the bottom, every pair of image rows corresponds to the same artificial label, which was estimated by the teacher. As with CIFAR-10, we observe that images in the same pair of rows contain semantically similar objects.

accuracy of the classifier averaged over 5 runs is 43% in the unsupervised training case and 83% in the fully supervised training case. We attribute this alignment to the fact that artificial labels are arbitrarily defined by the generator and the generator can adapt to the backpropagation from the discriminator, which in turn is influenced by the real labels.

In Table 5, we compare our method with the state-of-the-art cGAN methods on CIFAR-10. As in the unsupervised case, our proposed training shows a significant $\overline{\text{FID}}$ improvement over the previous state-of-the-art (from 5.39 to 4.35).

Evaluation on STL-10 and SVHN. In Table 8, we compare our method to a baseline without self-labeling and self-attention on STL-10 and SVHN. As we can see from the range of the FID scores, the STL-10 dataset is more complex and the SVHN dataset is simpler than the CIFAR-10 dataset. Another difference is that we use an image size of 48×48 pixels for the experiments on STL-10, while in the CIFAR-10 and SVHN datasets it is of 32×32 pixels. The results show that our method improves the FID score on both datasets compared to the baseline (from 32.85 to 30.91 and from 2.44 to 2.19). In Figure 9, we show samples of real and generated images.

In Table 6, we evaluate our method on STL-10 in the semi-supervised setting. The STL-10 dataset contains $100{,}000$ images of various classes and 500 images of 10 classes with supervised labels. By adding extremely few real labels to the cGAN training, the FID score of the baseline worsens (from 32.85 to 48.49). However, our method improves the FID score with respect to the baseline and even more substantially when labels are available (from 48.49 to 43.29). Finally, in Table 7, we compare our method with the state-of-the-art unsupervised GANs. We should point out that these methods use different generators, as a result of network architecture search (Gong et al., 2019; Doveh & Giryes, 2019). Despite the non optimality of our generator architecture, our approach yields a very competitive FID.

5. Conclusions

We proposed a novel GAN training scheme that can handle different levels of labeling in a unified manner. Our approach is based on using the generator to implicitly define artificial labels and then to train a classifier on purely synthetic data and labels. This classifier can then be used to self-label real data. Its class-consistency is found to correlate well with its classification probability score, which we then use to select samples with a reliable label (self-attention). We evaluated our approach on CIFAR-10, STL-10 and SVHN, and showed that both self-labeling and self-attention consistently improve the quality of generated data.

References

Berthelot, D., Carlini, N., Cubuk, E. D., Kurakin, A., Sohn, K., Zhang, H., and Raffel, C. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *CoRR*, abs/1911.09785, 2019.

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Chen, T., Zhai, X., Ritter, M., Lucic, M., and Houlsby, N. Self-supervised GANs via auxiliary rotation loss. In *2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 12154–12163. IEEE, 2019a.

Chen, T., Zhai, X., Ritter, M., Lucic, M., and Houlsby, N. Self-supervised GANs via auxiliary rotation loss. In *2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 12154–12163. IEEE, 2019b.

Coates, A., Ng, A. Y., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In Gordon, G. J., Dunson, D. B., and Dudík, M. (eds.), *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, volume 15 of *JMLR Proceedings*, pp. 215–223. JMLR.org, 2011.

Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. Randaugment: Practical automated data augmentation with a reduced search space. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Doveh, S. and Giryes, R. DEGAS: differentiable efficient generator search. *CoRR*, abs/1912.00606, 2019.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Teh, Y. W. and Titterington, D. M. (eds.), *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pp. 249–256. JMLR.org, 2010.

Gong, X., Chang, S., Jiang, Y., and Wang, Z. AutoGAN: Neural architecture search for generative adversarial networks. In *2019 IEEE International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 3223–3233. IEEE, 2019.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. C., and Bengio, Y. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 2672–2680, 2014.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778. IEEE, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 9726–9735. IEEE, 2020.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6626–6637, 2017.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *2019 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 4401–4410. IEEE, 2019.

Kavalerov, I., Czaja, W., and Chellappa, R. cGANs with multi-hinge loss. *CoRR*, abs/1912.04216, 2019.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. In *Technical report*. Citeseer, 2009.

Lucic, M., Tschannen, M., Ritter, M., Zhai, X., Bachem, O., and Gelly, S. High-fidelity image generation with fewer labels. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4183–4192. PMLR, 2019.

Mescheder, L. M., Geiger, A., and Nowozin, S. Which training methods for GANs do actually converge? In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15,*

2018, volume 80 of *Proceedings of Machine Learning Research*, pp. 3478–3487. PMLR, 2018.

Mirza, M. and Osindero, S. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

Miyato, T. and Koyama, M. cGANs with projection discriminator. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Noroozi, M. Self-labeled conditional GANs. *CoRR*, abs/2012.02162, 2020.

Odena, A., Olah, C., and Shlens, J. Conditional image synthesis with auxiliary classifier GANs. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 2642–2651. PMLR, 2017.

Reed, S. E., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. Generative adversarial text to image synthesis. In Balcan, M. and Weinberger, K. Q. (eds.), *Proceedings of the 33rd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 1060–1069. JMLR.org, 2016.

Schönfeld, E., Schiele, B., and Khoreva, A. A u-net based discriminator for generative adversarial networks. In *2020 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 8204–8213. IEEE, 2020.

Sinha, S., Zhao, Z., Goyal, A., Raffel, C., and Odena, A. Top-k training of GANs: Improving GAN performance by throwing away bad samples. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C., Cubuk, E. D., Kurakin, A., and Li, C. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Tarvainen, A. and Valpola, H. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1195–1204, 2017.

Tran, N., Tran, V., Nguyen, N., Yang, L., and Cheung, N. Self-supervised GAN: analysis and improvement with multi-class minimax game. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 13232–13243, 2019.

van Engelen, J. E. and Hoos, H. H. A survey on semi-supervised learning. *Mach. Learn.*, 109(2):373–440, 2020.

Zhao, S., Liu, Z., Lin, J., Zhu, J., and Han, S. Differentiable augmentation for data-efficient GAN training. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020a.

Zhao, Y., Li, C., Yu, P., Gao, J., and Chen, C. Feature quantization improves GAN training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 11376–11386. PMLR, 2020b.

Zhao, Z., Singh, S., Lee, H., Zhang, Z., Odena, A., and Zhang, H. Improved consistency regularization for GANs. *CoRR*, abs/2002.04724, 2020c.