erative setting. Serving as motivation for these adaptations is the idea that large over-parameterized networks have nicer loss landscapes than smaller ones, and are thus able to learn better quality mappings, regardless of whether an approximately equivalent mapping exists for smaller networks. We experimentally validate these methods on several datasets and via a number of objective measurements. Lastly, we discuss the limit of compression in the GAN setting and how it appears in empirical results.

2. Background

2.1. Generative Adversarial Networks

GAN was first proposed as a two player min-max optimization problem between a discriminator $f_w(.)$ and a generator $g_\theta(.)$ as in (1) (Goodfellow et al., 2014). The generator is tasked with generating realistic examples that fool the discriminator while the discriminator learns how to differentiate between the real and the generated samples.

$$\min_{\theta\in\Theta}\max_{w\in W}\mathbb{E}_{\mathbf{x}\sim p_{data}}[\log(f_w(\mathbf{x})] + \mathbb{E}_{\mathbf{z}\sim p_z}[\log(1-f_w(g_\theta(\mathbf{z})))] \tag{1}$$

The optimization in (1) has a global minimum and the system converges when $p_g = p_{data}$ at which point, $f_w(.)$ cannot classify a sample as being generated from p_g or from p_{data}. Further, the optimal solution to (1) corresponds to minimizing the Jensen-Shannon (JS) divergence between the two distributions p_{data} and p_g (Goodfellow et al., 2014). However, training of GANs is often unstable because JS divergence is not well defined when p_g and p_{data} do not have the same support (Arjovsky et al., 2017). To solve the problem with using JS divergence, WGAN minimizes the Wasserstein's distance between p_g and p_{data} in place of the JS divergence(Arjovsky et al., 2017), which is well defined even when p_g and p_{data} have disjoint support. Specifically, WGAN attempts to solve the optimization problem in (2), where $f_w(.)$ is a Lipschitz bounded function. (Arjovsky et al., 2017).

$$\min_{\theta\in\Theta}\max_{w\in W}\mathbb{E}_{\mathbf{x}\sim p_{data}}[f_w(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim p_z}[f_w(g_\theta(\mathbf{z}))] \tag{2}$$

WGAN was used in place of regular GAN for most of our experiments due to its favorable characteristics. However, empirically, we noticed that WGAN does not work as well for the Celeb-A dataset, so we reverted to using regular GAN for all Celeb-A related experiments.

2.2. Knowledge Distillation

Knowledge distillation refers to the technique of transferring the knowledge learned, from an ensemble of networks to a single network, or from a network with high number of parameters, to a network with relatively low number of parameters. We refer to the bigger network as the *teacher* network and the smaller network as the *student* network.

A student can learn to match any activation layer in the teacher network. Learning parameters from the final layer, called hard targets, lends itself to shorter training time but increased chance of over-fitting. The inputs to the softmax layer (logits) of the teacher network, referred to as soft targets, on the other hand, have more descriptive information about the samples and give better generalization characteristics to the student network (Hinton et al., 2015), which makes training on soft targets more beneficial.

2.3. Over-parameterization of Networks

An over-parameterized network is described as one whose number of hidden units is polynomially large relative to the number of training samples (Allen-Zhu et al., 2018b). It has been shown that training a significantly over-parameterized GAN yields dramatically better results than those generated from a smaller network (Brock et al., 2018). This may be explained by a finding that showed that the over-parameterization of neural networks creates optimized loss functions with many good minima spread throughout the entire loss landscape allowing for efficient training with alternating gradient descent (Allen-Zhu et al., 2018b) (Allen-Zhu et al., 2018a). This theory was bolstered by recent empirical studies of loss functions using visualization methods (Li et al., 2018). Therefore, it is necessary that a bigger network learn these mappings in a hyper-parameterized space before it can be distilled to a simpler model. Likewise, there has been empirical evidence that knowledge distillation, or model compression, is successful (Hinton et al., 2015) (Bucilu et al., 2006) (Yim et al., 2017). This success may be attributed to the aforementioned phenomena. Although training a teacher network might require a higher number of parameters, a reduced number of parameters is sufficient to describe the model with high fidelity.

3. Methods

The teacher (large, over-parameterized network) and student (small, few parameter network) GANs used either the original DCGAN architecture or a slightly modified DCGAN architecture (Radford et al., 2015), more closely resembling the WGAN (Arjovsky et al., 2017), referenced as the W-DCGAN.

The number of parameters in our networks is controlled by the depth scale factor, referenced throughout the paper as d. The overall number of parameters increases approximately linearly to d^2.

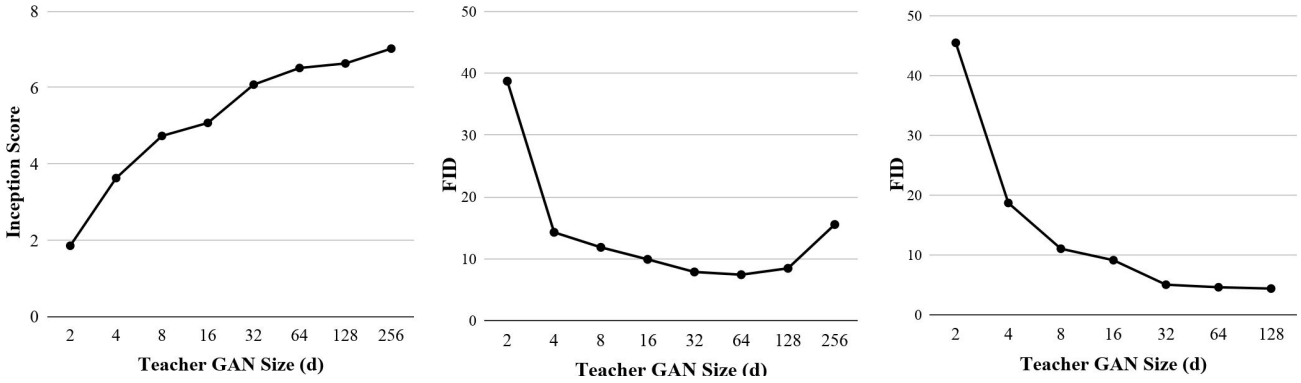

Figure 2. The Inception Score and Frechet Inception Distance was used to evaluate the best teacher GAN, parameterized by the depth scale factor d. A high Inception Score is good and a low Frechet Inception Distance is good. From these results, we selected a teacher GAN size of $d = 256$ for MNIST, $d = 64$ for CIFAR-10 and $d = 128$ for Celeb-A (left to right).

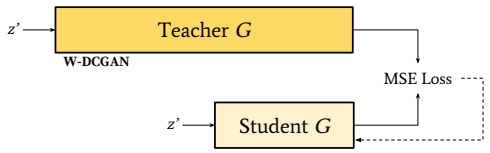

Figure 3. Student-teacher training framework with mean squared error (MSE) loss for student training. The teacher generator was trained using DCGAN framework (Radford et al., 2015) including WGAN modifications (Arjovsky et al., 2017). A mathematical analogy is shown in (3).

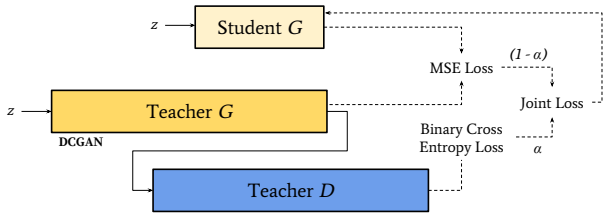

Figure 4. Student-teacher training framework with joint loss for student training. The teacher generator was trained using DCGAN framework (Radford et al., 2015). A mathematical analogy is shown (4).

4. Analysis

In the case of classification networks, the performance can be measured by the classification accuracy. Unlike classification networks, GANs do not have an explicit measure for performance. The performance of GANs could be naively measured by human judgment of visual quality (Goodfellow et al., 2014). For example, one could collect scores (1 to 10) of visual quality from various subjects and average the scores to understand the performance of GANs. However, the method is very expensive. The score could also vary significantly based on the design of the interface used to collect the data (Goodfellow et al., 2014). To evaluate the performance of GANs more systematically, the field has developed several quantitative metrics. Some of the popular metrics are Inception Score and Frechet Inception Distance (FID). Additionally, we used Variance of Laplacian to evaluate the blurring artifacts inherent to compressing GANs trained on complex datasets.

4.1. Inception Score (IS)

There are two important things that we would like to see in images generated from good GANs. First, we would like it to generate diverse images. We would like $p(\mathbf{y})$ to be relatively equal across different classes (Goodfellow et al., 2014). Secondly, given a generated image, we would like to be confident of the class in which the image belongs. Given a generated image x, we would like $p(\mathbf{y}|\mathbf{x})$ to be very concentrated in a particular class (Goodfellow et al., 2014). To take both of the desired qualities into account, the cross entropy, $H(\cdot, \cdot)$, between $p(\mathbf{y})$ and $p(\mathbf{y}|\mathbf{x})$ can be taken, otherwise known as the Inception Score.

$$IS = H(p(\mathbf{y}), p(\mathbf{y}|\mathbf{x})) \tag{5}$$

If $p(\mathbf{y})$ is similar across classes and $p(\mathbf{y}|\mathbf{x})$ is very concentrated in a particular class, then the cross entropy between the two distributions will be high. Consequently, the Inception Score will be high.

The Inception Scores makes a few assumptions. First, it assumes that the image can be classified yielding $p(\mathbf{y})$ and $p(\mathbf{y}|\mathbf{x})$, but not all images can be classified. For example, in our experiments with the Celeb-A dataset, we could not use Inception Score because the data set does not have labels associated with them. Second, the Inception Score is