Learning Loss for Knowledge Distillation with Conditional Adversarial Networks

Zheng Xu
University of Maryland
College Park

Yen-Chang Hsu
Georgia Institute of Technology
Atlanta

Jiawei Huang
Honda Research Institute
Mountain View

Abstract

There is an increasing interest on accelerating neural networks for real-time applications. We study the student-teacher strategy, in which a small and fast student network is trained with the auxiliary information provided by a large and accurate teacher network. We use conditional adversarial networks to learn the loss function to transfer knowledge from teacher to student. The proposed method is particularly effective for relatively small student networks. Moreover, experimental results show the effect of network size when the modern networks are used as student. We empirically study trade-off between inference time and classification accuracy, and provide suggestions on choosing a proper student.

1 Introduction

Deep neural networks (DNNs) achieve massive success in artificial intelligence by substantially improving the state-of-the-art performance in various applications. For one of the core applications in computer vision, large-scale image classification (Russakovsky et al. 2015), the accuracy reached by DNNs has become comparable to humans on several benchmark datasets. The recent progress towards such impressive accomplishment is largely driven by exploring deeper and wider network architectures. Despite the clear performance boost of modern DNNs (He et al. 2016; Zagoruyko and Komodakis 2016; Xie et al. 2017), the heavy computation and memory cost of these deep and wide networks makes it difficult to directly deploy the trained networks on embedded system for real-time applications. In the meantime, the demands on low cost networks are increasing for applications on mobile devices and autonomous cars.

Do DNNs really need to be deep and wide? Early theoretical studies (Cybenko 1989; Hornik, Stinchcombe, and White 1989) suggest that shallow networks can approximate arbitrary functions. More recent theorems show depth is indeed beneficial for the expressive capacity of networks (Eldan and Shamir 2016; Telgarsky 2016; Liang and Srikant 2017; Safran and Shamir 2017). Moreover, the overparameterized and redundant networks, which can easily memorize and overfit the training data, surprisingly generalize well in practice (Zhang et al. 2017; Arpit et al. 2017). Various explanations have been investigated, but the secret of deep and wide networks remains an open problem.

On the other hand, empirical studies suggest that the performance of shallow networks can be improved by learning from large networks following the student-teacher strategy (Bucilu, Caruana, and Niculescu-Mizil 2006; Ba and Caruana 2014; Urban et al. 2017; Hinton, Vinyals, and Dean 2015). In these approaches, the student networks are forced to mimic the output probability distribution of the teacher networks to transfer the knowledge embedded in the soft targets. The intuition is that the *dark knowledge* (Hinton, Vinyals, and Dean 2015), which contains the relative probabilities of “incorrect” answers provided by deep and wide networks, is informative and representative. For example, an image of a dog may be mistakenly recognized as cat or sheep with small probability, but should be seldom recognized as car; the soft target of output distribution over categories for this image, $(0.7, 0.2, 0.1, 0)$, contains more information than the hard target of one-hot vector, $(1, 0, 0, 0)$.

In the previous works, (Ba and Caruana 2014; Urban et al. 2017) train shallow and wide student networks that potentially have more parameters than the deep teacher networks; (Hinton, Vinyals, and Dean 2015) use ensemble of networks as teacher, and train student network with similar architecture and capacity; particularly, (Romero et al. 2015) train a small deep and thin network to replace a shallow and wide network for acceleration, given the best teacher at that time is the shallow and wide VGGNet (Simonyan and Zisserman 2014). Since then, the design of network architecture has advanced. ResNet (He et al. 2016) has significantly deepened the networks by introducing residual connections, and wide residual networks (WRNs) (Zagoruyko and Komodakis 2016) suggest to widen the networks for better performance. It is unclear whether the dark knowledge from the state-of-the-art networks based on residual connections, which are both deep and wide, can help train a shallow and/or thin network for acceleration.

In this paper, we focus on the practical approach that can improve the performance of a shallow and thin modern network by learning from the dark knowledge of a deep and wide network. Both the student and teacher networks are convolutional neural networks (CNNs) with residual connections, and the student network is shallow and thin so that it can be much faster than the teacher network during inference. Instead of forcing the output of a student network to exactly mimic the soft targets produced by a teacher network, we introduce conditional adversarial networks to transfer the dark knowledge from teacher to student. We empirically show that the loss learned by the adversarial training has the advantage over the hand-engineered loss in the student-teacher strategy, especially when a relatively small student is used.

Our learning loss approach is inspired by the recent success of conditional adversarial networks for various image-to-image translation applications (Isola et al. 2017). We show that the generative adversarial nets (GAN) is capable of benefiting a task that is very different from image generation. In the student-teacher strategy, GAN can help preserve the multi-modal nature of the output distribution. Take the soft targets of a dog image over (dog, cat, sheep, car) as an example, both soft targets (0.7, 0.2, 0.1, 0) and (0.8, 0, 0.2, 0) can predict the correct label, dog. In fact, a teacher network trained multiple times would produce different while correct soft targets because of the randomness in the training. Forcing a student network to exactly mimic one of the soft targets (or the average/ensemble of several teacher networks) can be difficult when the student has smaller capacity than the teacher. By introducing the discriminator as in GAN, our learning loss approach transfers the correlation between classes, i.e., the dark knowledge from teacher, and also preserves the multi-modality.

2 Related work

Network acceleration has attracted increasing interest because the needs of real-time applications in artificial intelligence are growing . The techniques can be roughly divided into three categories, low precision, pruning and factorization, and knowledge distillation. Low precision methods use limited number of bits to store and operate the network weights, and the extreme case is binary networks that only use 1-bit to represent each number (Rastegari et al. 2016). The acceleration of these methods is sometimes conceptual because the low precision support of common GPU is still limited. Networks can also be directly modified by pruning and factorizing the redundant weights, either as a post-processing after training, or as a fine-tuning stage (Li et al. 2017b; Howard et al. 2017). These methods often assume network weights are sparse and/or low rank, and aim for efficient networks of similar architecture with reduced number of weights.

Knowledge distillation is a principled approach to train small neural networks for acceleration. We slightly abuse the name *knowledge distillation* to represent methods that train student networks by transferring knowledge from teacher networks. (Bucilu, Caruana, and Niculescu-Mizil 2006) pioneered this approach for model compression. (Ba and Caruana 2014; Urban et al. 2017) trained shallow but wide student by learning from a deep teacher, which were not primarily designed for acceleration. (Hinton, Vinyals, and Dean 2015) generalized the previous methods by introducing a new metric between the output distribution of teacher and student, as well as a tuning parameter. The variants of knowledge distillation has also been applied to many different tasks, such as semantic segmentation (Ros et al. 2016), pedestrian detection (Shen et al. 2016), face recognition (Luo et al. 2016), metric learning(Chen, Wang, and Zhang 2017), reinforcement learning (Teh et al. 2017) and for regularization(Sau and Balasubramanian 2016). A recent preprint (Kim and Kim 2017) presented promising preliminary results on CIFAR-10 by learning a small ResNet from a large ResNet. Another line of research focuses on transferring intermediate features instead of soft targets from teacher to student (Romero et al. 2015; Wang et al. 2016; Zagoruyko and Komodakis 2017; Yim et al. 2017; Huang and Wang 2017; Zhou et al. 2017; You et al. 2017). Our approach is complement to those methods by directly following (Hinton, Vinyals, and Dean 2015) to design a new metric between the output distribution of teacher and student, and adversarial networks are used to learn the metric to replace hand-engineering.

Generative adversarial networks (GAN) has been extensively studied over recent years since (Goodfellow et al. 2014). GAN trains two neural networks, the generator and the discriminator, in an adversarial learning process that alternatively updates the two networks. We apply GAN in the conditional setting (Mirza and Osindero 2014; Isola et al. 2017; Reed et al. 2016; Odena, Olah, and Shlens 2017), where the generator is conditioned on input images. Unlike the previous works focus on generating and editing images, we target on learning a loss for knowledge distillation, which requires quite different architecture choices for our generator and discriminator.

3 Learning loss for knowledge distillation

In this section, we introduce the learning loss approach based on conditional adversarial networks. We start from a recap of modern network architectures (section 3.1), and then describe the dark knowledge that can be transferred from teacher to student networks (section 3.2). The GAN-based approach for learning loss is detailed in section 3.3.

3.1 Neural networks with residual connection

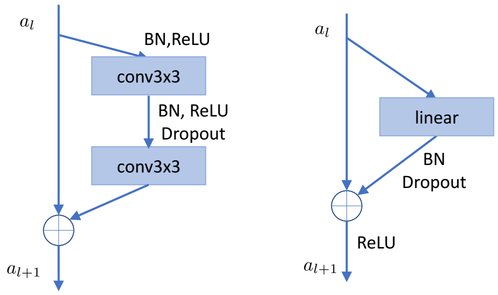


Figure 1: Blocks with residual connection for convolutional neural networks (Zagoruyko and Komodakis 2016) (left) and multi-layer perceptron (right). a_l represents the output of lth block. Each block is composed of batch normalization (BN), activation ReLU, weight layer and dropout.

The modern neural networks are built by stacking basic components. For computer vision tasks, residual blocks (He et al. 2016; Zagoruyko and Komodakis 2016) are the basic components to build deep neural networks to achieve state-of-the-art performance. Both student and teacher networks in this paper are based on the residual convolutional blocks shown in Figure 1 (left). The first layer is 16 filters of 3×3 convolution, followed by a stack of $6n$ layers, where n is the number of residual blocks and each block contains two convolution layers equipped with batch normalization (Ioffe and Szegedy 2015), ReLU (Krizhevsky, Sutskever, and Hinton 2012) and dropout (Srivastava et al. 2014). The output feature map is subsampled twice, and the number of filters are doubled when subsampling, as shown in Table 1. The widen factor m is used to increase the number of filters in each residual block. After the last residual block is the global average pooling, and then fully-connected layer and softmax. In the following sections, the architecture of wide residual networks (WRNs) is represented by WRN-d-m, where the total depth is $d = 6n + 4$. Our teacher network is deep and wide WRN with large d and m, while student network is shallow and thin WRN with small d and m.

Table 1: The architecture of wide residual networks (Zagoruyko and Komodakis 2016). n represents the number of residual blocks, m represents the widen factor.

output size	32×32	16×16	8×8
# layers	2n	2n	2n
# filters	16m	32m	64m

3.2 Knowledge distillation

The output of neural networks for image classification is a probability distribution over categories. The probability is generated by the softmax layer from *logits*, which represents the output of the last fully connected layer. The dimension of logits from student and teacher networks are both equal to the number of categories. Rich information is embedded in the output probability of a teacher network, and we can use logits to transfer the knowledge to student network (Bucilu, Caruana, and Niculescu-Mizil 2006; Ba and Caruana 2014; Urban et al. 2017; Hinton, Vinyals, and Dean 2015). We review the method in (Hinton, Vinyals, and Dean 2015), which provides a metric between student and teacher logits that generalized previous methods for *knowledge distillation.*

The logits vector generated by pre-trained teacher network for an input image $x_i, i = 1, \ldots, N$ is represented by t_i, where the dimension of vector $t_i = (t_i^1, \ldots, t_i^C)$ is the number of categories C. We now consider training a student network F to generate student logits $F(x_i)$. By introducing a parameter called temperature T, the generalized softmax layer converts logits vector t_i to probability distribution q_i,

$$M_T(t_i) = q_i, \text{ where } q_i^j = \frac{\exp(t_i^j/T)}{\sum_k \exp(t_i^k/T)}. \tag{1}$$

(Hinton, Vinyals, and Dean 2015) proposed to minimize the Kullback-Leibler divergence between teacher output and student output,

$$\mathcal{L}_{KD}(F,T) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{KL}(M_T(t_i)\|M_T(F(x_i))). \tag{2}$$

where higher temperature T produces softer probability over categories. As detailed in (Hinton, Vinyals, and Dean 2015), the Euclidean distance between teacher and student logits, $\|t_i - F(x_i)\|_2^2$, is a special case of $\mathcal{L}_{KD}$ when T is large .

The regular softmax for classification uses $T = 1$. When the image-label pairs $\{x_i, l_i\}$ are provided, the cross-entropy loss for supervised training of a neural network can be represented as

$$\mathcal{L}_S(F) = \frac{1}{N}\sum_{i=1}^{N} \mathcal{H}(l_i, M_1(F(x_i))). \tag{3}$$

$\mathcal{L}_S$ is the most frequently used loss for pure supervised learning in image classification. (Hinton, Vinyals, and Dean 2015) minimized the weighted combination of loss $\mathcal{L}_{KD}$ and loss $\mathcal{L}_S$ to train a student network,

$$\mathcal{L}_1(F,T) = \frac{1}{2}\mathcal{L}_S(F) + T^2\mathcal{L}_{KD}(F,T). \tag{4}$$

3.3 Learning loss with adversarial networks

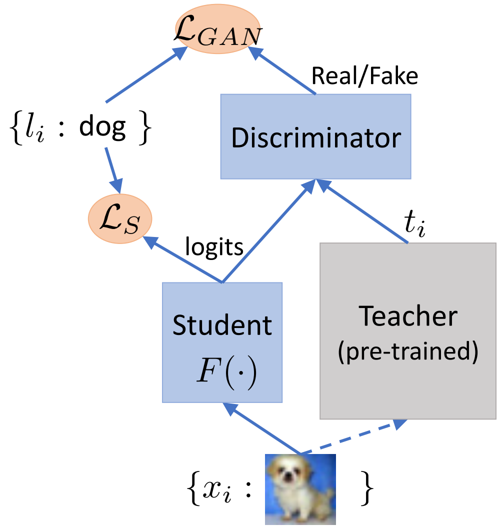


Figure 2: The GAN-based architecture to learn loss for knowledge distillation. The deep and wide teacher is pre-trained offline. The student network and discriminator are updated alternatively, where the discriminator aims to distinguish logits from student and teacher networks, and the student aims to fool the discriminator. Additional supervised loss is added for both student and discriminator.

Overview. The main idea of learning the loss for transferring knowledge from teacher to student is presented in Figure 2. Instead of forcing the student to exactly mimic the teacher by minimizing KL-divergence in $\mathcal{L}_1(F,T)$ of Equation (4), the knowledge is transferred from teacher to student through a discriminator in our GAN-based approach.

This discriminator is trained to distinguish the output logits of teacher and student, while the student is adversarially trained to fool the discriminator, i.e., output logits similar to the teacher logits so that the discriminator can not distinguish.

There are several benefits of the proposed method. First, the learned loss can be effective, as has already been demonstrated for several image to image translation tasks (Isola et al. 2017). Moreover, the GAN-based approach relieves the pain for hand-engineering the loss. Though the parameter tuning and hand-engineering of the loss is replaced by hand-engineering the discriminator networks in some sense, our empirical study shows that the performance is less sensitive to the discriminator architecture than the temperature parameter in knowledge distillation. The second benefit is closely related to the multi-modality of network output. Let us consider the previous example on classifying a dog image for labels (dog, cat, sheep, car). The relationship of the categories, such as dog looks more like cat than car, can be captured by the discriminator trained from the multi-modal logits of teacher. However, if we look at each individual output, both (0.7, 0.2, 0.1, 0) and (0.8, 0, 0.2, 0) are valid outputs, but it may not be plausible for the student with small capacity to exactly produce either one of the outputs. The student can still benefit from the knowledge transferred from discriminator, which suggests the output should be similar to the two vectors and different from a vector like (0.4, 0.1, 0.1, 0.4).

Discriminator update. We now describe the proposed method in a more rigorous way. The student and discriminator in Figure 2 are alternatively updated in the GAN-based approach. Let us first look at the update of the discriminator, which is trained to distinguish teacher and student logits. We use multi-layer perceptron (MLP) as discriminator, which is the stack of residual block shown in Figure 1 (right). The number of nodes in each layer is the same as the dimension of logits, i.e., the number of categories C. Representing the discriminator that predicts binary value "Real/Fake" as $D(\cdot)$ and fixing the student network $F(\cdot)$, we can maximize the log-likelihood, which is known as binary cross-entropy loss,

$$\mathcal{L}_A(D,F) = \frac{1}{N}\sum_{i=1}^{N}\Big(\log P(\text{Real}|D(t_i)) + \log P(\text{Fake}|D(F(x_i)))\Big).$$

The adversarial loss $\mathcal{L}_A$ for knowledge distillation, which follows the original GAN (Goodfellow et al. 2014), faces two major issues. First, the adversarial training is difficult. Even if we replace the log-likelihood with advanced techniques such as Wasserstein GAN (Arjovsky, Chintala, and Bottou 2017) and Least Squares GAN (Mao et al. 2016), the training is still slow and unstable in our experiments. Second, the discriminator captures the high-level statistics of teacher and student outputs, but the low-level alignment is missing. The student outputs $F(x_i)$ for x_i can learn from a completely unrelated teacher sample t_j by optimizing $\mathcal{L}_A$, which means a dog image can be mapped to a logits vector that predicts cat.

To tackle these problems, we modify the discriminator objective to also predict the class labels, inspired by (Chen et al. 2016; Odena, Olah, and Shlens 2017). In this case, the output of discriminator $D(\cdot)$ is a $2C$ dimensional vector and each entry corresponds to a tuple of $\{Label, Real/Fake\}$. We now maximize

$$\tilde{\mathcal{L}}_{GAN}(D,F) = \frac{1}{N}\sum_{i=1}^{N}\Big(\log P(\{l_i, \text{Real}\}|D(t_i)) + \log P(\{l_i, \text{Fake}\}|D(F(x_i)))\Big). \tag{5}$$

If we further assume *Label* and *Real/Fake* are conditional independent, we can simplify the loss as

$$\mathcal{L}_{GAN}(D,F) = \frac{1}{2}(\mathcal{L}_A(D,F) + \mathcal{L}_{DS}(D,F)), \tag{6}$$

where $\mathcal{L}_A$ is the previously defined adversarial loss, $\mathcal{L}_{DS}$ is the supervised log-likelihood of discriminator written as

$$\mathcal{L}_{DS}(D,F) = \frac{1}{N}\sum_{i=1}^{N}\Big(\log P(l_i|D(t_i)) + \log P(l_i|D(F(x_i)))\Big).$$

In (6), the output of discriminator $D(\cdot)$ is a $C+2$ dimensional vector and each entry corresponds to either *Label* or *Real/Fake*. In our experiments, optimizing the two forms of GAN-based loss, $\tilde{\mathcal{L}}_{GAN}$ in (5) and $\mathcal{L}_{GAN}$ in (6), achieves almost identical performance. Hence we will always use (6) in the following sections because of the simplicity and compactness of discriminator. Note that equation (6) has the same form as the auxiliary classifier GANs (Odena, Olah, and Shlens 2017).

The adversarial training becomes much more stable when the discriminator reconstructs category *Labels* besides *Real/Fake*. Moreover, the discriminator can provide category-level alignment between outputs of student and teacher. The student outputs of a dog image are more likely to learn from the teacher outputs that predict dogs.

To provide instance-level information for the discriminator, we investigate conditional discriminators, in which the input of discriminators are logits concatenate with a conditional vector. We tried the following conditional vectors: image with convolutional embedding; label one-hot vector with embedding; and the extracted teacher logits. The embedding includes several weight layers and outputs a vector that is the same size as the logits. However, it turns out the conditional vectors are easy to be ignored during the training of discriminator. The conditional discriminator does not help in practice and we introduce a more direct instance-level alignment for training student network below.

Student update. We update the student network after updating the discriminator in each iteration. When updating the student network $F(\cdot)$, we aim to fool the discriminator by fixing discriminator $D(\cdot)$ and minimizing the adversarial loss $\mathcal{L}_A$. In the meantime, the student network is also trained to satisfy the auxiliary classifier of discriminator $\mathcal{L}_{DS}$. Besides the category-level alignment provided by $\mathcal{L}_{DS}$, we introduce instance-level alignment between teacher and student outputs as

$$\mathcal{L}_{L_1}(F) = \frac{1}{N}\sum_{i=1}^{N}|F(x_i) - t_i|. \tag{7}$$

The L_1 norm alignment has been found helpful in the GAN-based approach for image to image translation (Isola et al.

2017). At last, we combine the learned loss with the supervised loss $\mathcal{L}_S$ in (3), and minimize the following objective for the student network $F(\cdot)$,

$$\begin{aligned}\mathcal{L}_2(D,F) =& \mathcal{L}_S(F) + \mathcal{L}_{L_1}(F) \\ &+ \frac{1}{2}(\mathcal{L}_A(D,F) - \mathcal{L}_{DS}(D,F)).\end{aligned} \tag{8}$$

The sign of $\mathcal{L}_{DS}$ is flipped in (6) and (8) because both the discriminator and student are trained to preserve the category-level information.

The final loss $\mathcal{L}_2(D,F)$ in (8) is a combination of the learned loss for knowledge distillation and the supervised loss for neural network, and may look complicated at the first glance. However, each component of the loss is relatively simple. Moreover, since both student F and discriminator D are learned, there is no explicit parameters to be tuned in the loss function. Our experiments in the next section suggest the performance is reasonably insensitive to the discriminator architecture and the proposed method can replace the hand-engineering loss for knowledge distillation.

4 Experiments

We present the experimental results in this section. The implementation details and experimental settings are provided in section 4.1. The benefits of the proposed method for training a small student network with the help of a large teacher network is presented in section 4.2. We then analyze the different components of the proposed methods in section 4.3. The effect of depth and width of the student network is presented in section 4.4, followed by the discussion of trade-off between classification accuracy and inference acceleration in section 4.5. At last, in section 4.6, we show the qualitative visualization on the output distribution for student, teacher, and knowledge distillation.

4.1 Experimental setting

We consider three benchmark datasets for image classification: ImageNet32 (Chrabaszcz, Loshchilov, and Hutter 2017), CIFAR-10 and CIFAR-100 (Krizhevsky and Hinton 2009). ImageNet32 is a downsampled version of the ImageNet2012 challenge dataset (Russakovsky et al. 2015), which contains 1.28M training images and 50K validation images for 1K classes; all images are downsampled to 32×32. The CIFAR datasets contain 50K training images and 10K validation images for 10 and 100 classes, respectively. The images are also 32×32. In all the experiments, light data augmentation with horizontal flip, padding and cropping is used for input images as in (He et al. 2016).

We use wide residual networks (WRNs) (Zagoruyko and Komodakis 2016) as both student and teacher networks. The residual blocks are shown in Figure 1 (right) and the network architectures are in Table 1. WRN-d-m represents network with $d = 6n + 4$ as depth and m as widen factor. The teacher network is always WRN-40-10, while the student network changes depth and width in different experiments. Dropout 0.3 is used for all WRNs. We use stochastic gradient descent (SGD) as optimizer, and set the initial learning rate as 0.1, momentum as 0.9, and weight decay as 1e-4. We use minibatch size 128 and train the WRNs for 200 epochs with learning rate divided by 10 at epoch 80 and 160 for CIFARs; we use minibatch size 256 and train the WRNs for 70 epochs with learning rate divided by 10 at epoch 25 and 50 for Imagenet32.

We use multi-layer preceptron (MLP) for the discriminator in the GAN-based approach. The number of nodes in each layer is the same as the length of logits, i.e., number of categories. 3-layer MLP is used for most of the experiments except for section 4.3, in which we study the effect of depth for discriminator. The logits of teacher are generated offline and stored in memory. The logits pass through a batch normalization layer before the MLP. Dropout 0.3 is also used for discriminator.

The implementation is in PyTorch. The results below present the median of five random runs.

4.2 Benefits of learning loss

Table 2: The error rate on benchmark datasets.

	CIFAR-10	CIFAR-100	ImageNet32
Student	7.46	28.52	48.2
Teacher	4.19	20.62	38.41
KD(T=1)	7.27	28.62	49.37
KD(T=2)	7.3	28.33	49.48
KD(T=5)	7.02	27.06	49.63
KD(T=10)	6.94	27.07	51.12
Ours	**6.09**	**25.75**	**47.39**

We first show the proposed method is good for transferring knowledge from teacher to student. Table 2 presents the error rate of classification on the three benchmark datasets. The teacher is the deep and wide WRN-40-10. The student is much shallower and thinner, WRN-10-4 for CIFARs, and WRN-22-4 for ImageNet32. We choose a larger student network for ImageNet32 because the dataset contains more training samples and categories. More discussion on wisely choosing the student architecture is in section 4.4 and 4.5.

The first two rows of Table 2 present the performance of pure supervised learning for student and teacher networks, without knowledge transfer by student-teacher strategy. We compare the GAN-based approach with knowledge distillation proposed in (Hinton, Vinyals, and Dean 2015) and reviewed in section 3.2. We choose the temperature parameter $T \in \{1, 2, 5, 10\}$ following (Hinton, Vinyals, and Dean 2015). The GAN-based approach is detailed in section 3.3 and no parameter is tuned.

We have several observations from Table 2. The deep and wide teacher performs much better than the shallow and thin student by pure supervised learning. The error rate of the small network trained with student-teacher strategy is lower bounded by the teacher performance, as expected. Baseline method (Hinton, Vinyals, and Dean 2015) helps the training of small networks for the two CIFARs, but does not help for ImageNet32. The difference may be because the capacity of the student is too small to learn from knowledge distillation for ImageNet32 that has more samples and categories. The temperature parameter T introduced in (Hinton, Vinyals, and Dean 2015) is useful. For CIFARs, (Hinton, Vinyals, and Dean 2015) performs better when T is large to some extent, and $T = 5$ and $T = 10$ performs similarly. The proposed method improves the performance of small network for all three datasets, and outperforms (Hinton, Vinyals, and Dean 2015).

4.3 Analysis of the proposed method

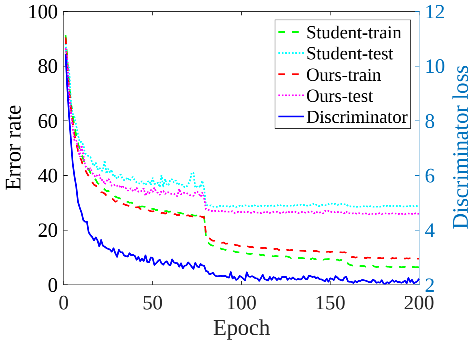


Figure 3: The training curve on CIFAR-100. We show the training/testing accuracy for supervised training of student network, and GAN-based training, as well as the loss of the discriminator.

We discuss the proposed method in more detail in this section. Figure 3 presents the training curve of the small student network, WRN-10-4, on CIFAR-100 dataset. The loss of the discriminator (blue solid line) is gradually decreasing, which suggests the adversarial training steadily makes progress. The error rates of GAN-based method for both training and testing data are decreasing. The testing error rate of GAN-based method is consistently better than the pure supervised training of the student model, and looks more stable between epoch 50-100. Surprisingly, the training error rate of pure supervised learning is slightly better than the GAN-based method, which suggests knowledge transfer is more beneficial for generalization.

Table 3: The effect of different components of the loss in the proposed method; the error rates on benchmark datasets.

	CIFAR-10	CIFAR-100
$\mathcal{L}_S$	7.46	28.52
$\mathcal{L}_{GAN}$	14.82	47.04
$\mathcal{L}_S + \mathcal{L}_{GAN}$	6.56	27.27
$\mathcal{L}_S + \mathcal{L}_{L_1}$	6.44	26.66
$\mathcal{L}_S + \mathcal{L}_{L_1} + \mathcal{L}_{GAN}$	**6.09**	**25.75**

Next, we look into different components of the GAN-based approach, as shown in Table 3. By optimizing the adversarial loss and the category-level knowledge transfer, the learned loss $\mathcal{L}_{GAN}$ performs reasonably well. However, the indirect knowledge provided by $\mathcal{L}_{GAN}$ alone is not as good as pure supervised learning $\mathcal{L}_S$. Both category-level knowledge transfer by $\mathcal{L}_{GAN}$ and instance-level knowledge transfer by $\mathcal{L}_{L_1}$ can improve the performance of training student network. The final approach combines these components and performs the best without parameter tuning.

Table 4: The effect of discriminator depth on CIFAR-100.

Depth	1	2	3	4
Error rate	26.13	25.88	**25.75**	27.42

Finally, we present the effect of the depth of MLP as discriminator in Table 4. The error rate is relatively insensitive to the depth of discriminator. The error rate slightly decreases as the depth increases when the discriminator is generally shallow. When the discriminator becomes deeper, the error rate increases as the adversarial training becomes unstable. Decreasing the learning rate of discriminator sometimes helps, but it may introduce parameter tuning for the proposed method. The 3-layer MLP works reasonably well and is used for all our experiments to keep the GAN-based method simple.

4.4 Do WRN need to be deep and wide?

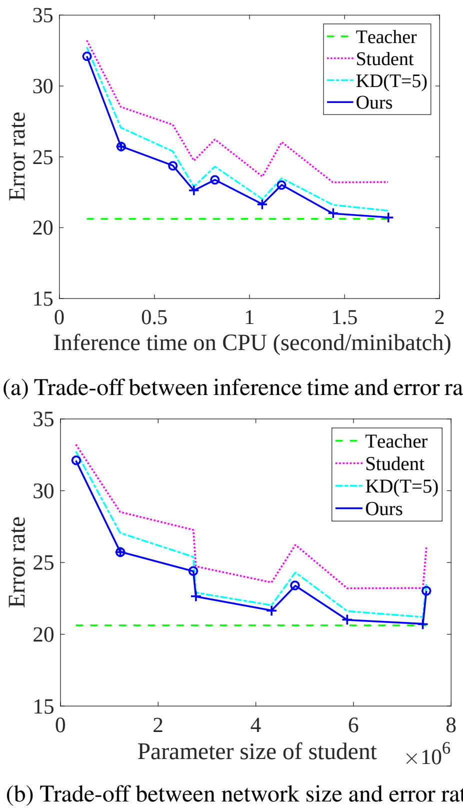
Table 5: The effect of depth and width in student network; the error rate on CIFAR-100.

WRN	Size (M)	Time (s)	Student	KD(T=5)	Ours
10-2	0.32	0.14	33.22	32.74	32.1
10-4	1.22	0.32	28.52	27.16	25.75
10-6	2.72	0.60	27.27	25.39	24.39
10-8	4.81	0.82	26.23	24.31	23.38
10-10	7.49	1.17	26.04	23.49	23.02
16-4	2.77	0.71	24.73	22.9	22.73
22-4	4.32	1.07	23.61	22.02	21.66
28-4	5.87	1.44	23.2	21.61	21
34-4	7.42	1.73	23.22	21.2	20.73
40-10	55.9	8.73	20.62	-	-

(Urban et al. 2017) asked similar question for convolutional neural networks and claimed the network should at least has a few layers of convolutions. We study the modern architecture WRN of residual blocks. Our empirical study suggests that even for the modern architecture WRN, the network has to be deep and wide to some extent.

Table 5 presents the results of pure supervised learning, knowledge distillation (Hinton, Vinyals, and Dean 2015) and the GAN-based approach for different student networks on CIFAR-100. We first fix the depth of WRN as 10, and change the widen factor from 2 to 10. 10 is the minimum depth for our WRN architecture as the depth has to be $6n+4$. We then fix the width as 4, and increase depth from 10 to 34. The parameter size is in millions, and the inference time is in seconds per minibatch of 100 samples on CPU.

When the student is very small, such as WRN-10-2, it is difficult to transfer knowledge from teacher to student because the student is limited by the network capacity. When the student is large, such as WRN-34-4, both knowledge distillation (Hinton, Vinyals, and Dean 2015) and GAN-based approach can improve the performance to almost as good



(a) Trade-off between inference time and error rate.

(b) Trade-off between network size and error rate.

Figure 4: Error rate to inference time and parameter size. The figure is generated from Table 5. Networks WRN-10-m are labeled as circles, and WRN-d-4 are labeled as crosses for the GAN-based approach. The largest student is 7x smaller and 5x faster than the teacher WRN-40-10.

as the teacher. The advantage of the proposed method is more obvious for relatively small student such as WRN-10-4. Increasing depth is more effective than increasing width for WRN. For example, WRN-34-4 has less parameter than WRN-10-10, but achieves lower error rate.

4.5 Training student for acceleration

The shallow and thin network is much easier to deploy in practice. We present the trade-off between error rate, inference time and parameter size in Figure 4. The figure is generated from Table 5 by changing the architecture of the student network. Larger student network is more accurate but also slower. For network with similar size, such as WRN-10-10 and WRN-34-4, deep network achieves lower error rate, while wide network runs slightly faster. The student-teacher strategy can help improve the classification performance of the student network. When the student network is relatively large, such as WRN-34-4, the student network trained by the GAN-based approach can achieve competitive error rate comparing to the teacher WRN-40-10. WRN-34-4 is 7x smaller and 5x faster than WRN-40-10, and the GAN-based approach decreases the absolute error rate by 2.5%.

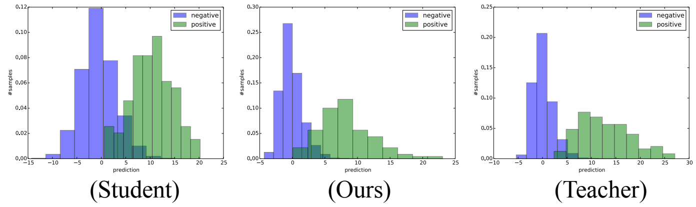


Figure 5: Qualitative visualization; the distribution of prediction for category 85 in CIFAR-100.

4.6 Visualization of distribution

In the last section of experimental results, we present qualitative visualization for the GAN-based approach. Figure 5 presents the scaled histogram for the prediction of category 85 in CIFAR-100. The histogram is counted on the 10K testing samples, in which 100 samples are from category 85 and labeled as positive (green in figure), and the other 9.9K are labeled as negative (blue in the figure). The histogram is scaled to sum up to one for positive and negative, respectively. The three plots represent the distribution predicted by student network trained by pure supervised learning , the teacher network, and the student network trained by GAN-based approach. The histogram in the middle is similar to the histogram in the right, which suggests the GAN-based approach transfers knowledge from teacher to student.

5 Conclusion and discussion

We study the student-teacher strategy for network acceleration in this paper. We propose a GAN-based approach to learn the loss for transferring knowledge from teacher to student. We empirically show the the GAN-based approach can improve the training of student network, especially when the student network is shallow and thin. Moreover, we empirically study the effect of capacity for modern network as student and provide guidelines for wisely choosing a student to balance error rate and acceleration. In specific setting, we can train a student that is 7x smaller and 5x faster than teacher without loss of accuracy.

The GAN-based approach is stable and easy to implement after applying several advanced techniques in the GAN literature. The current implementation uses the stored logtis from teacher network to save GPU memory and computation. Generating teacher logits on the fly with dropout can be more reliable for the adversarial training. At last, the GAN-based approach can be naturally extended for the ensemble of networks as teacher. The logits of multiple teacher networks can be fed into the discriminator for better performance. We will investigate these ideas for future work.

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein GAN. *ICML*.

Arpit, D.; Jastrzebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; et al. 2017. A closer look at memorization in deep networks. *ICML*.

Ba, J., and Caruana, R. 2014. Do deep nets really need to be deep? In *NIPS*, 2654–2662.

Bucilu, C.; Caruana, R.; and Niculescu-Mizil, A. 2006. Model compression. In *KDD*, 535–541. ACM.

Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; and Abbeel, P. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, 2172–2180.

Chen, Y.; Wang, N.; and Zhang, Z. 2017. Darkrank: Accelerating deep metric learning via cross sample similarities transfer. *arXiv preprint arXiv:1707.01220*.

Chrabaszcz, P.; Loshchilov, I.; and Hutter, F. 2017. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*.

Cybenko, G. 1989. Approximation by superpositions of a sigmoidal function. *MCSS* 2(4):303–314.

Eldan, R., and Shamir, O. 2016. The power of depth for feedforward neural networks. In *COLT*, 907–940.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*, 2672–2680.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *CVPR*, 770–778.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hornik, K.; Stinchcombe, M.; and White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural networks* 2(5):359–366.

Howard, A. G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; and Adam, H. 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Huang, Z., and Wang, N. 2017. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*.

Ioffe, S., and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 448–456.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. *CVPR*.

Kim, S. W., and Kim, H.-E. 2017. Transferring knowledge to smaller network with class-distance loss. *ICLR Workshop*.

Krizhevsky, A., and Hinton, G. 2009. Learning multiple layers of features from tiny images.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*, 1097–1105.

Li, H.; De, S.; Xu, Z.; Studer, C.; Samet, H.; and Goldstein, T. 2017a. Training quantized nets: A deeper understanding. *arXiv preprint arXiv:1706.02379*.

Li, H.; Kadav, A.; Durdanovic, I.; Samet, H.; and Graf, H. P. 2017b. Pruning filters for efficient convnets. *ICLR*.

Liang, S., and Srikant, R. 2017. Why deep neural networks for function approximation? *ICLR*.

Luo, P.; Zhu, Z.; Liu, Z.; Wang, X.; Tang, X.; et al. 2016. Face model compression by distilling knowledge from neurons. In *AAAI*, 3560–3566.

Mao, X.; Li, Q.; Xie, H.; Lau, R. Y.; Wang, Z.; and Smolley, S. P. 2016. Least squares generative adversarial networks. *arXiv preprint ArXiv:1611.04076*.

Mirza, M., and Osindero, S. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Odena, A.; Olah, C.; and Shlens, J. 2017. Conditional image synthesis with auxiliary classifier gans. *ICML*.

Rastegari, M.; Ordonez, V.; Redmon, J.; and Farhadi, A. 2016. XNOR-net: Imagenet classification using binary convolutional neural networks. In *ECCV*, 525–542. Springer.

Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; and Lee, H. 2016. Generative adversarial text to image synthesis. *ICML*.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2015. Fitnets: Hints for thin deep nets. *ICLR*.

Ros, G.; Stent, S.; Alcantarilla, P. F.; and Watanabe, T. 2016. Training constrained deconvolutional networks for road scene semantic segmentation. *arXiv preprint arXiv:1604.01545*.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *IJCV* 115(3):211–252.

Safran, I., and Shamir, O. 2017. Depth-width tradeoffs in approximating natural functions with neural networks. In *ICML*, 2979–2987.

Sau, B. B., and Balasubramanian, V. N. 2016. Deep model compression: Distilling knowledge from noisy teachers. *arXiv preprint arXiv:1610.09650*.

Shen, J.; Vesdapunt, N.; Boddeti, V. N.; and Kitani, K. M. 2016. In teacher we trust: Learning compressed models for pedestrian detection. *arXiv preprint arXiv:1612.00478*.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR* 15(1):1929–1958.

Teh, Y. W.; Bapst, V.; Czarnecki, W. M.; Quan, J.; Kirkpatrick, J.; Hadsell, R.; Heess, N.; and Pascanu, R. 2017. Distral: Robust multitask reinforcement learning. *arXiv preprint arXiv:1707.04175*.

Telgarsky, M. 2016. Benefits of depth in neural networks. *arXiv preprint arXiv:1602.04485*.

Urban, G.; Geras, K. J.; Kahou, S. E.; Aslan, O.; Wang, S.; Caruana, R.; Mohamed, A.; Philipose, M.; and Richardson, M. 2017. Do deep convolutional nets really need to be deep and convolutional? *ICLR*.

Wang, J.; Wei, Z.; Zhang, T.; and Zeng, W. 2016. Deeply-fused nets. *arXiv preprint arXiv:1605.07716*.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. *CVPR*.

Yadav, A.; Shah, S.; Xu, Z.; Jacobs, D.; and Goldstein, T. 2017. Stabilizing adversarial nets with prediction methods. *arXiv preprint arXiv:1705.07364*.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. *CVPR*.

You, S.; Xu, C.; Xu, C.; and Tao, D. 2017. Learning from multiple teacher networks. In *KDD*, 1285–1294. ACM.

Zagoruyko, S., and Komodakis, N. 2016. Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zagoruyko, S., and Komodakis, N. 2017. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *ICLR*.

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2017. Understanding deep learning requires rethinking generalization. *ICLR*.

Zhou, G.; Fan, Y.; Cui, R.; Bian, W.; Zhu, X.; and Gai, K. 2017. Rocket launching: A universal and efficient framework for training well-performing light net. *arXiv preprint arXiv:1708.04106*.