Improved Knowledge Distillation via Teacher Assistant

Seyed Iman Mirzadeh,*[1] Mehrdad Farajtabar,*[2] Ang Li,[2]
Nir Levine,[2] Akihiro Matsukawa,†[3] Hassan Ghasemzadeh[1]
[1]Washington State University, WA, USA
[2]DeepMind, CA, USA
[3]D.E. Shaw, NY, USA
[1]{seyediman.mirzadeh, hassan.ghasemzadeh}@wsu.edu
[2]{farajtabar, anglili, nirlevine}@google.com
[3]akihiro.matsukawa@gmail.com

Abstract

Despite the fact that deep neural networks are powerful models and achieve appealing results on many tasks, they are too large to be deployed on edge devices like smartphones or embedded sensor nodes. There have been efforts to compress these networks, and a popular method is knowledge distillation, where a large (teacher) pre-trained network is used to train a smaller (student) network. However, in this paper, we show that the student network performance degrades when the gap between student and teacher is large. Given a fixed student network, one cannot employ an arbitrarily large teacher, or in other words, a teacher can effectively transfer its knowledge to students up to a certain size, not smaller. To alleviate this shortcoming, we introduce multi-step knowledge distillation, which employs an intermediate-sized network (teacher assistant) to bridge the gap between the student and the teacher. Moreover, we study the effect of teacher assistant size and extend the framework to multi-step distillation. Theoretical analysis and extensive experiments on CIFAR-10,100 and ImageNet datasets and on CNN and ResNet architectures substantiate the effectiveness of our proposed approach.

Introduction

Deep neural networks have achieved state of the art results in a variety of applications such as computer vision (Huang et al. 2017; Hu, Shen, and Sun 2018), speech recognition (Han et al. 2017) and natural language processing (Devlin et al. 2018). Although it is established that introducing more layers and more parameters often improves the accuracy of a model, big models are computationally too expensive to be deployed on devices with limited computation power such as mobile phones and embedded sensors. Model compression techniques have emerged to address such issues, *e.g.*, parameter pruning and sharing (Han, Mao, and Dally 2016), low-rank factorization (Tai et al. 2015) and knowledge distillation (Bucila, Caruana, and Niculescu-Mizil 2006; Hinton, Vinyals, and Dean 2015). Among these approaches, knowledge distillation has proven a promising way to obtain a small model that retains the accuracy of a large one. It works by adding a term to the usual classification loss that encourages the student to mimic the teacher's behavior.

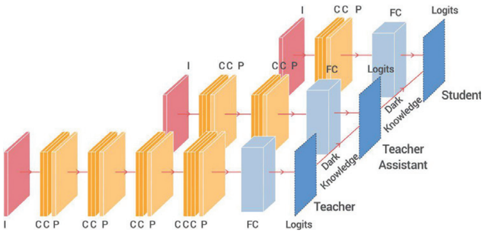


Figure 1: TA fills the gap between student & teacher

However, we argue that knowledge distillation is not always effective, especially when the gap (in size) between teacher and student is large. To illustrate, we ran experiments that show surprisingly a student model distilled from a teacher with more parameters(and better accuracy) performs worse than the same one distilled from a smaller teacher with a smaller capacity. Such scenarios seem to impact the efficacy of knowledge distillation where one is given a small student network and a pre-trained large one as a teacher, both fixed and (wrongly) presumed to form a perfect transfer pair.

Inspired by this observation, we propose a new distillation framework called Teacher Assistant Knowledge Distillation (TAKD), which introduces intermediate models as teacher assistants (TAs) between the teacher and the student to fill in their gap (Figure 1). TA models are distilled from the teacher, and the student is then only distilled from the TAs.

Our contributions are: (1) We show that the size (capacity) gap between teacher and student is important. To the best of our knowledge, we are the first to study this gap and verify that the distillation performance is not at its top with the largest teacher; (2) We propose a teacher assistant based knowledge distillation approach to improve the accuracy of student network in the case of extreme compression; (3) We extend this framework to include a chain of multiple TAs from teacher to student to further improve the knowledge transfer and provided some insights to find the best one; (4) Through extensive empirical evaluations and a theoretical justification, we show that introducing intermediary TA networks improves the distillation performance.

*Equal Contribution

†Work Done at DeepMind

Related Work

We discuss in this section related literature in knowledge distillation and neural network compression.

Model Compression. Since our goal is to train a small, yet accurate network, this work is related to model compression. There has been an interesting line of research that compresses a large network by reducing the connections based on weight magnitudes (Han, Mao, and Dally 2016; Li et al. 2016) or importance scores (Yu et al. 2018). The reduced network is fine-tuned on the same dataset to retain its accuracy. Another line of research focuses on distilling the original (large) network to a smaller network (Polino, Pascanu, and Alistarh 2018; Wang et al. 2018a), in which case the smaller network is more flexible in its architecture design does not have to be a sub-graph of the original network.

Knowledge Distillation. Originally proposed by Bucila, Caruana, and Niculescu-Mizil (2006) and popularized by Hinton, Vinyals, and Dean (2015) knowledge distillation compress the knowledge of a large and computational expensive model (often an ensemble of neural networks) to a single computational efficient neural network. The idea of knowledge distillation is to train the small model, the student, on a transfer set with soft targets provided by the large model, the teacher. Since then, knowledge distillation has been widely adopted in a variety of learning tasks (Yim et al. 2017; Yu et al. 2017; Schmitt et al. 2018; Chen et al. 2017). Adversarial methods also have been utilized for modeling knowledge transfer between teacher and student (Heo et al. 2018; Xu, Hsu, and Huang 2018; Wang et al. 2018b; 2018c).

There have been works studying variants of model distillation that involve multiple networks learning at the same time. Romero et al. (2014) proposed to transfer the knowledge using not only the logit layer but earlier ones too. To cope with the difference in width, they suggested a regressor to connect teacher and student's intermediate layers. Unfortunately, there is not a principled way to do this. To solve this issue, Yim et al.; Yu et al. (2017; 2017) used a shared representation of layers, however, it's not straightforward to choose the appropriate layer to be matched. Czarnecki et al. (2017) minimized the difference between teacher and student derivatives of the loss combined with the divergence from teacher predictions while Tarvainen and Valpola (2017) uses averaging model weights instead of target predictions. Urban et al. (2017) trained a network consisting of an ensemble of 16 convolutional neural networks and compresses the learned function into shallow multilayer perceptrons. To improve the student performance, Sau and Balasubramanian (2016) injected noise into teacher logits to make the student more robust. Utilizing multiple teachers were always a way to increase robustness. Zhang et al. (2017) proposed deep mutual learning which allows an ensemble of student models to learn collaboratively and teach each other during training. KL divergences between pairs of students are added into the loss function to enforce the knowledge transfer among peers. You et al. (2017) proposed a voting strategy to unify multiple relative dissimilarity information provided by multiple teacher networks. Anil et al. (2018) introduced an efficient distributed online distillation framework called co-distillation and argue that distillation can even work when the teacher and student are made by the same network architecture. The idea is to train multiple models in parallel and use distillation loss when they are not converged, in which case the model training is faster and the model quality is also improved.

However, the effectiveness of distilling a large model to a small model has not yet been well studied. Our work differs from existing approaches in that we study how to improve the student performance given fixed student and teacher network sizes, and introduces intermediate networks with a moderate capacity to improve distillation performance. Moreover, our work can be seen as a complement that can be combined with them and improve their performance.

Distillation Theory. Despite its huge popularity, there are few systematic and theoretical studies on how and why knowledge distillation improves neural network training. The so-called *dark knowledge* transferred in the process helps the student learn the finer structure of teacher network.

Hinton, Vinyals, and Dean (2015) argues that the success of knowledge distillation is attributed to the logit distribution of the incorrect outputs, which provides information on the similarity between output categories. Furlanello et al. (2018) investigated the success of knowledge distillation via gradients of the loss where the soft-target part acts as an importance sampling weight based on the teachers confidence in its maximum value. Zhang et al. (2017) analyzed knowledge distillation from the posterior entropy viewpoint claiming that soft-targets bring robustness by regularizing a much more informed choice of alternatives than blind entropy regularization. Last but not least, Lopez-Paz et al. (2015) studied the effectiveness of knowledge distillation from the perspective of learning theory (Vapnik 1998) by studying the estimation error in empirical risk minimization framework.

In this paper, we take this last approach to support our claim on the effectiveness of introducing an intermediate network between student and teacher. Moreover, we empirically analyze it via visualizing the loss function.

Assistant based Knowledge Distillation

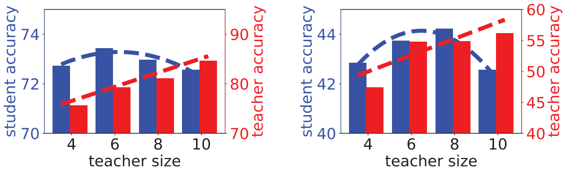
Background and Notations

The idea behind knowledge distillation is to have the student network (S) be trained not only via the information provided by true labels but also by observing how the teacher network (T) represents and works with the data. The teacher network is sometimes deeper and wider (Hinton, Vinyals, and Dean 2015), of similar size (Anil et al. 2018; Zhang et al. 2017), or shallower but wider (Romero et al. 2014).

Let a_t and a_s be the logits (the inputs to the final softmax) of the teacher and student network, respectively. In classic supervised learning, the mismatch between output of student network softmax(a_s) and the ground-truth label y_r is usually penalized using cross-entropy loss

$$\mathcal{L}_{SL} = \mathcal{H}(\text{softmax}(a_s), y_r). \tag{1}$$

In knowledge distillation, originally proposed by Bucila, Caruana, and Niculescu-Mizil; Ba and Caruana (2006; 2014) and popularized by Hinton, Vinyals, and Dean (2015),



a) CIFAR-10

b) CIFAR-100

Figure 2: Distillation performance with increasing teacher size. The number of convolutional layers in student is 2.

one also tries to match the softened outputs of student $y_s =$ softmax(a_s/τ) and teacher y_t =softmax(a_t/τ) via a KL-divergence loss

$$\mathcal{L}_{KD} = \tau^2 KL(y_s, y_t) \tag{2}$$

Hyperparameter τ referred to temperature is introduced to put additional control on softening of signal arising from the output of the teacher network. The student network is then trained under the following loss function:

$$\mathcal{L}_{\text{student}} = (1-\lambda)\mathcal{L}_{SL} + \lambda\mathcal{L}_{KD} \tag{3}$$

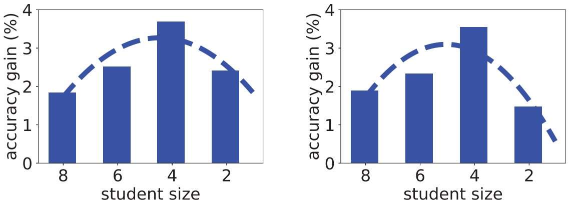
where λ is a second hyperparameter controlling the trade-off between the two losses. We refer to this approach as Baseline Knowledge Distillation (BLKD) through the paper.

The Gap Between Student and Teacher

Given a fixed student network, *e.g.*, a Convolutional Neural Network (CNN) with 2 layers to be deployed on a small embedded device, and a pool of larger pre-trained CNNs, which one should be selected as the teacher in the knowledge distillation framework? The first answer is to pick the strongest which is the biggest one. However, this is not what we observed empirically as showing in Figure 2. Here, a plain CNN student with 2 convolutional layers is being trained via distillation with similar but larger teachers of size 4, 6, 8, and 10 on both CIFAR-10 and CIFAR-100 datasets. By size, we mean the number of convolutional layers in the CNN. This number is roughly proportional to the actual size or number of parameters of the neural network and proxy its capacity. Note that they are usually followed by max-pooling or fully connected layers too. We defer the full details on experimental setup to experiments section.

With increasing teacher size, its own (test) accuracy increases (plotted in red on the right axis). However, the trained student accuracy first increases and then decreases (depicted in blue on the left axis). To explain this phenomenon, we can name a few factors that are competing against each other when enlarging the teacher:

1. Teacher's performance increases, thus it provides better supervision for the student by being a better predictor.
2. The teacher is becoming so complex that the student does not have the sufficient capacity or mechanics to mimic her behavior despite receiving hints.
3. Teacher's certainty about data increases, thus making its logits (soft targets) less soft. This weakens the knowledge transfer which is done via matching the soft targets.



a) CIFAR-10

b) CIFAR-100

Figure 3: Percentage of distilled student performance increase over the performance when it learns from scratch with varying student size. The teacher has 10 layers.

Factor 1 is in favor of increasing the distillation performance while factors 2 and 3 are against it. Initially, as the teacher size increases, factor 1 prevails; as it grows larger, factors 2 and 3 dominate.

Similarly, imagine the dual problem. We are given a large teacher network to be used for training smaller students, and we are interested in knowing for what student size this teacher is most beneficial in the sense of boosting the accuracy against the same student learned from scratch. As expected and illustrated in Figure 3, by decreasing student size, factor 1 causes an increase in the student's performance boost while gradually factors 2 and 3 prevail and worsen the performance gain.

Teacher Assistant Knowledge Distillation (TAKD)

Imagine a real-world scenario where a pre-trained large network is given, and we are asked to distill its knowledge to a fixed and very small student network. The gap discussed in the previous subsection makes the knowledge distillation less effective than it could be. Note that we cannot select the teacher size or the student size to maximize the performance. Both are fixed and given.

In this paper, we propose to use intermediate-size networks to fill in the gap between them. The teacher assistant (TA) lies somewhere in between teacher and student in terms of size or capacity. First, the TA network is distilled from the teacher. Then, the TA plays the role of a teacher and trains the student via distillation. This strategy will alleviate factor 2 in the previous subsection by being closer to the student than the teacher. Therefore, the student is able to fit TA's logit distribution more effectively than that of the teacher's. It also alleviates factor 3 by allowing softer (and maybe) less confident targets. In terms of factor 1, a TA may degrade the performance, however, as we will see in experiments and theoretical analysis sections, both empirical results and theoretical analyses substantiate the effectiveness (improved performance) of TAKD. This happens because encouraging positively correlated factors (like 2 and 3) outweighs the performance loss due to negative ones (like 1).

It will be demonstrated in experiments that TA with any intermediate size always improves the knowledge distillation performance. However, one might ask what the optimal TA size for the highest performance gain is? If one TA improves the distillation result, why not also train this TA via another distilled TA? Or would a TA trained from scratch

Table 1: Comparison on evaluation accuracy between our method (TAKD) and baselines. For CIFAR, plain (S=2, TA=4, T=10) and for ResNet (S=8, TA=20, T=110) are used. For ImageNet, ResNet (S=14, TA=20, T=50) is used. Higher numbers are better.

Model	Dataset	NOKD	BLKD	TAKD
CNN	CIFAR-10	70.16	72.57	73.51
	CIFAR-100	41.09	44.57	44.92
ResNet	CIFAR-10	88.52	88.65	88.98
	CIFAR-100	61.37	61.41	61.82
ResNet	ImageNet	65.20	66.60	67.36

Table 2: Student's accuracy given varied TA sizes for (S=2, T=10)

Model	Dataset	TA=8	TA=6	TA=4
CNN	CIFAR-10	72.75	73.15	73.51
	CIFAR-100	44.28	44.57	44.92

Table 3: Student's accuracy given varied TA sizes for (S=8, T=110)

Model	Dataset	TA=56	TA=32	TA=20	TA=14
ResNet	CIFAR-10	88.70	88.73	88.90	88.98
	CIFAR-100	61.47	61.55	61.82	61.5

be as effective as our approach? In the following sections, we try to study and answer these questions from empirical perspectives complemented with some theoretical intuitions.

Experimental Setup

We describe in this section the settings of our experiments.

Datasets. We perform a set of experiments on two standard datasets CIFAR-10 and CIFAR-100 and one experiment on the large-scale ImageNet dataset. The datasets consist of 32×32 RGB images. The task for all of them is to classify images into image categories. CIFAR-10, CIFAR-100 and ImageNet contain 10 and 100 and 1000 classes, respectively.

Implementation. We used PyTorch (Paszke et al. 2017) framework for the implementation[1] and as a preprocessing step, we transformed images to ones with zero mean and standard deviation of 0.5. For optimization, we used stochastic gradient descent with Nesterov momentum of 0.9 and learning rate of 0.1 for 150 epochs. For experiments on plain CNN networks, we used the same learning rate, while for ResNet training we decrease learning rate to 0.01 on epoch 80 and 0.001 on epoch 120. We also used weight decay with the value of 0.0001 for training ResNets. To attain reliable results, we performed all the experiments with a hyper-parameter optimization toolkit (Microsoft-Research 2018) which uses a tree-structured Parzen estimator to tune hyper-parameters as explained in (Bergstra et al. 2011). Hyper-parameters include distillation trade-off λ and temperature τ explained in the previous section. It's notable that all the accuracy results reported in this paper, are the top-1 test accuracy reached by the hyper-parameter optimizer after running each experiment for 120 trials.

Network Architectures. We evaluate the performance of the proposed method on two architectures. The first one is a VGG like architecture (plain CNN) consists of convolutional cells (usually followed by max pooling and/or batch normalization) ended with fully connected layer. We take the number of convolutional cells as a proxy for size or capacity of the network. The full details of each plain CNN network is provided in the appendix[2]. We also used ResNet as a more advanced CNN architecture with skip connections. We used the structures proposed in the original paper (He et al. 2016). The number of blocks in the ResNet architecture is served as a proxy for the size or flexibility of the network.

Results and Analysis

In this section, we evaluate our proposed Teacher Assistant Knowledge Distillation (TAKD) and investigate several important questions related to this approach. Throughout this section, we use S=i to represent the student network of size i, T=j to represent a teacher network of size j and TA=k to represent a teacher assistant network of size k. As a reminder by size we mean the number of convolutional layers for plain CNN and ResNet blocks for the case of ResNet. These serve as a proxy for the size or the number of parameters or capacity of the network.

Will TA Improve Knowledge Distillation?

First of all, we compare the performance of our Teacher Assistant based method (TAKD) with the baseline knowledge distillation (BLKD) and with training normally without any distillation (NOKD) for the three datasets and two architectures. Table 1 shows the results. It is seen the proposed method outperforms both the baseline knowledge distillation and the normal training of neural networks by a reasonable margin. We include ImageNet dataset only for this experiment to demonstrate TAKD works for the web-scale data too. For the rest of the paper we work with CIFAR10 and CIFAR100.

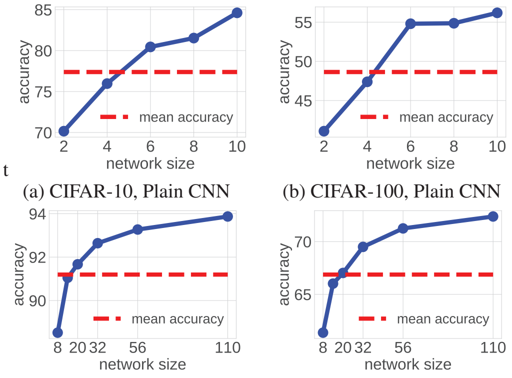
What is the Best TA Size?

The benefits of having a teacher assistant as an intermediary network for transferring knowledge comes with an essential burden – selecting the proper TA size. We evaluate the student's accuracy given varied TA sizes for plain CNN in Table 2 and for ResNet in 3, respectively.

The first observation is that having a TA (of any size) improves the result compared to BLKD and NOKD reported in Table 1. Another observation is that for the case of CNN,

[1] Codes and Appendix are available at the following address: https://github.com/imirzadeh/Teacher-Assistant-Knowledge-Distillation

[2] Appendix is available along with the code repository.



(a) CIFAR-10, Plain CNN

(b) CIFAR-100, Plain CNN

(c) CIFAR-10, ResNet

(d) CIFAR-100, ResNet

Figure 4: Accuracy of training from scratch for different network sizes. The dashed red line shows the average performance of the teacher and student.

TA=4 performs better than TA=6 or TA=8. One might naturally ask why 4 is the best while 6 seems to be better bridge as it is exactly lies between 2 and 10? Alternatively, we note that for both CIFAR-10 and CIFAR-100, the optimal TA size (4) is actually placed close to the middle in terms of average accuracy rather than the average of size. Figure 4-a,b depicts the accuracy of a trained neural network with no distillation in blue while the mean accuracy between S=2 and T=10 is depicted in red dashed line. The figure shows that for both of them, size 4 is closer to the mean value compared to 6 or 8. For ResNet in Table 3 for CIFAR-10, TA=14 is the optimum, while, for CIFAR-100, TA=20 is the best. Interestingly, Figure 4-c,d confirms that for CIFAR-10, TA=14 is closer to the mean performance of size 8 and 110 while TA=20 is so for CIFAR-100. Incorporating a TA with size close to the average performance of teacher and student seems to be a reasonable heuristic to find the optimal TA size, however, more systematic theoretical and empirical investigation remains an interesting venue for future work.

Why Limiting to 1-step TA?

We have seen that for CNN networks on CIFAR-100, incorporating a TA=4 between S=2 and T=10 improves the student. However, to train TA=4 via distillation from T=10, one may propose to put another TA (Say TA =6) in between to enhance the TA training via another distillation. Using a simplified notation we represent the above sequential distillation process by the *distillation path* $10 \rightarrow 6 \rightarrow 4 \rightarrow 2$. Even, one could go further and do a distillation via the path $10 \rightarrow 8 \rightarrow 6 \rightarrow 4 \rightarrow 2$.

To investigate this extension we evaluate all the possible distillation paths and show their outcomes in a single graph in Figure 5. To simplify the presentation we only include networks with even numbers of layers. The numbers in each oval are the accuracy on CIFAR-100 trained on CNN network using the corresponding distillation paths. A benefit of this visualization is not only that we can study the transfer results to S=2, but also for intermediate sizes. Given n possible intermediate networks, there are 2^n possible paths. For example, the 4 possible paths to transfer from T=10 to S=4 are shown in column associated to size 4. For better comparison, the direct transfer (associated to BLKD) are colored in green while the performance without distillation (NOKD) is shown in the last row.

By studying this figure we get interesting insights. Firstly, it is clear that, for all the student sizes (S=2,4,6), TAKD works better than BLKD or NOKD. No matter how many TAs are included in the distillation path, one can obtain better students compared to BLKD and NOKD. Secondly, the column associated with size 2 reveals that all multi-step TAKD variants work comparably good and considerably better than BLKD and NOKD. Thirdly, for S=2 and S=4, a full path going through all possible intermediate TA networks performs the best. According to these observations, one can choose a distillation path based on the time and computing resources available. Without any constraint, a full distillation path is optimal (refer to appendix for details). However, an interesting extension is to limit the number of intermediate TA networks. Can we find the best path in that setting? Given a student and teacher is there a way to automatically find the best path given the constraints? In the appendix section we provide a discussion for these problems.

Comparison with Other Distillation Methods

Since the rediscovery of the basic knowledge distillation method (Hinton, Vinyals, and Dean 2015) many variants of it has been proposed. In Fig 6-right we have compared the performance of our proposed framework via a single TA with some of the most recent state-of-the-art ones reported and evaluated by Heo et al. (2018). The 'FITNET' (Romero et al. 2014) proposed to match the knowledge in the intermediate layers. The method denoted as 'AT' proposed spatial transfer between teacher and student (Z and K 2016). In the 'FSP' method, a channel-wise correlation matrix is used as the medium of knowledge transfer (Yim et al. 2017). The method 'BSS' (Heo et al. 2018) trains a student classifier based on the adversarial samples supporting the decision boundary. For these the numbers are reported from the paper (Heo et al. 2018). To make a fair comparison, we used exactly the same setting for CIFAR-10 experiments. In addition to 50K-10K training-test division, all classifiers were trained 80 epochs. Although we found that more epochs (e.g. 160) further improves our result, we followed their setting for a fair comparison. ResNet26 is the teacher and ResNet8 and ResNet14 are the students. In addition, we compared with deep mutual learning, 'MUTUAL', with our own implementation of the proposed algorithm in (Zhang et al. 2017) where the second network is the teacher network. Also, since deep mutual learning needs an initial training phase for both networks, we did this initialization phase for 40 epochs for both networks and then, trained both networks mutually for 80 epochs, equal to other modes. For our method, we used TAs ResNet20 and ResNet14 for students ResNet14 and ResNet8, respectively. It's seen that our TA-trained student outperforms all of them. Note that our proposed framework can be combined with all these variants to improve them too.

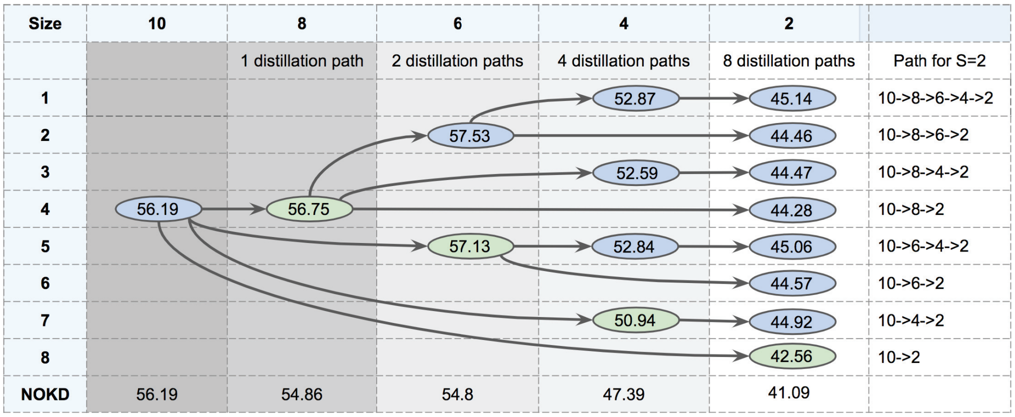


Figure 5: Distillation paths for plain CNN on CIFAR-100 with T=10

Why Does Distillation with TA work?

In this section, we try to shed some light on why and how our TA based knowledge distillation is better than the baselines.

Theoretical Analysis

According to the VC theory (Vapnik 1998) one can decompose the classification error of a classifier f_s as

$$R(f_s) - R(f_r) \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{sr}}}\right) + \epsilon_{sr}, \tag{4}$$

where, the $O(\cdot)$ and ϵ_{sr} terms are the estimation and approximation error, respectively. The former is related to the statistical procedure for learning given the number of data points, while the latter is characterized by the capacity of the learning machine. Here, $f_r \in \mathcal{F}_r$ is the real (ground truth) target function and $f_s \in \mathcal{F}_s$ is the student function, R is the error, $|\cdot|_C$ is some function class capacity measure, n is the number of data point, and finally $\frac{1}{2} \leq \alpha_{sr} \leq 1$ is related to the learning rate acquiring small values close to $\frac{1}{2}$ for difficult problems while being close to 1 for easier problems. Note that ϵ_{sr} is the approximation error of the student function class $\mathcal{F}_s$ with respect to $f_r \in \mathcal{F}_r$. Building on the top of Lopez-Paz et al. (2015), we extend their result and investigate why and when introducing a TA improves knowledge distillation. In Equation (4) student learns from scratch (NOKD). Let $f_t \in \mathcal{F}_t$ be the teacher function, then

$$R(f_t) - R(f_r) \leq O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}}\right) + \epsilon_{tr}, \tag{5}$$

where, α_{tr} and ϵ_{tr} are correspondingly defined for teacher learning from scratch. Then, we can transfer the knowledge of the teacher directly to the student and retrieve the baseline knowledge distillation (BLKD). To simplify the argument we assume the training is done via pure distillation ($\lambda = 1$):

$$R(f_s) - R(f_t) \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{st}}}\right) + \epsilon_{st}, \tag{6}$$

where α_{st} and ϵ_{st} are associated to student learning from teacher. If we combine Equations (5) and (6) we get

$$O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{st}}}\right) + \epsilon_{tr} + \epsilon_{st} \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{sr}}}\right) + \epsilon_{sr} \tag{7}$$

to hold for BLKD to be effective. In line with our finding, but with a little different formulation, Lopez-Paz et al. (2015) pointed out $|\mathcal{F}_t|_C$ should be small, otherwise the BLKD would not outperform NOKD. We acknowledge that similar to Lopez-Paz et al. (2015), we work with the upper bounds not the actual performance and also in an asymptotic regime. Here we built on top of their result and put a (teacher) assistant between student and teacher

$$R(f_s) - R(f_a) \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{sa}}}\right) + \epsilon_{sa}, \tag{8}$$

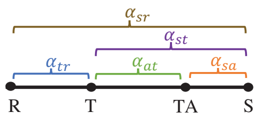
and, then the TA itself learns from the teacher

$$R(f_a) - R(f_t) \leq O\left(\frac{|\mathcal{F}_a|_C}{n^{\alpha_{at}}}\right) + \epsilon_{at}, \tag{9}$$

where, α_{sa}, ϵ_{sa}, α_{at}, and ϵ_{at} are defined accordingly. Combing Equations (5), (8), and (9) leads to the following equation that needs to be satisfied in order to TAKD outperforms BLKD and NOKD, respectively:

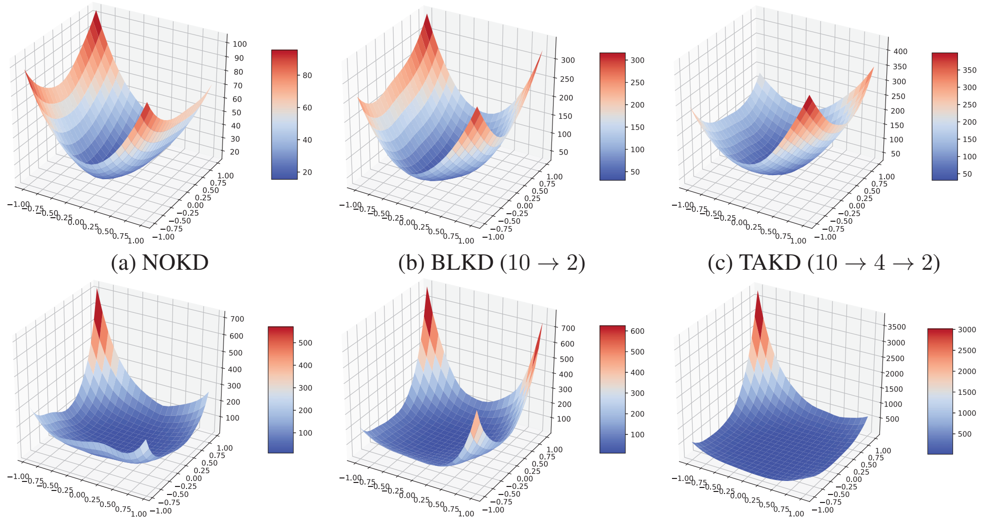
$$\begin{aligned} O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}} + \frac{|\mathcal{F}_a|_C}{n^{\alpha_{at}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{sa}}}\right) + \epsilon_{tr} + \epsilon_{at} + \epsilon_{sa} && (10) \\ \leq O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{st}}}\right) + \epsilon_{tr} + \epsilon_{st} && (11) \\ \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{sr}}}\right) + \epsilon_{sr}. && (12) \end{aligned}$$

We now discuss how the first inequality (eq. (10) $\leq$ eq. (11)) holds which entails TAKD outperforms BLKD. To do so, first note that $\alpha_{st} \leq \alpha_{sa}$ and $\alpha_{st} \leq \alpha_{at}$ (the larger the gap means the lower rate of learning or smaller $\alpha_{..}$). Figure 6-left shows their differences. Student learning directly from teacher is certainly more difficult than either student learning from TA or TA learning from teacher. Therefore, asymptotically speaking, $O\left(\frac{|\mathcal{F}_a|_C}{n^{\alpha_{at}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{sa}}}\right) \leq O\left(\frac{|\mathcal{F}_s|_C}{n^{\alpha_{st}}}\right)$ which in turn leads to $O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}} + \frac{|\mathcal{F}_a|_C}{n^{\alpha_{at}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{sa}}}\right) \leq O\left(\frac{|\mathcal{F}_t|_C}{n^{\alpha_{tr}}} + \frac{|\mathcal{F}_s|_C}{n^{\alpha_{st}}}\right)$. Moreover, according to assumption of Hinton, Vinyals, and Dean (2015) we know $\epsilon_{at} + \epsilon_{sa} \leq$



Student	NOKD	BLKD	FITNET	AT	FSP	BSS	MUTUAL	TAKD
ResNet8	86.02	86.66	86.73	86.86	87.07	87.32	87.71	**88.01**
Resnet14	89.11	89.75	89.72	89.84	89.92	90.34	90.54	**91.23**

Figure 6: Left) rate of learning between different targets (longer distance means lower α..); Right) Table for Comparison of TAKD with distillation alternatives on ResNet8 and ResNet14 as student and ResNet26 teacher



(a) NOKD (b) BLKD (10 → 2) (c) TAKD (10 → 4 → 2)

(d) NOKD (e) BLKD (110 → 8) (f) TAKD (110 → 20 → 8)

Figure 7: Loss landscape around local minima. Top) plain CNN for student of size 2. Bottom: ResNet for student of size 8.

ϵ_{st}. These two together establish eq. (10) $\leq$ eq. (11), which means that the upper bound of error in TAKD is smaller than its upper bound in BLKD.

Similarly, for the second inequality (eq. (11) $\leq$ eq. (12)) one can use $\alpha_{sr} \leq \alpha_{st}$ and $\alpha_{sr} \leq \alpha_{tr}$ and $\epsilon_{tr} + \epsilon_{st} \leq \epsilon_{sr}$. Note that, these are asymptotic equations and hold when $n \to \infty$. In the finite sample regime, when $|\mathcal{F}_t|_C$ is very large, then the inequality eq. (11) $\leq$ eq. (12) may not be valid and BLKD fails. Another failure case (in the finite sample regime) for BLKD happens when the student and teacher differ greatly in the capacity (i.e. α_{st} is very small and close to α_{sr}). In this case, the error due to transfer from real to teacher outweigh (11) in comparison to (12) and the inequality becomes invalid. In this case TAKD turns to be the key. By injecting a TA between student and teacher we break the very small α_{st} to two larger components α_{sa} and α_{at} which makes the second inequality (eq. (10) $\leq$ eq. (11)) a game changer for improving knowledge distillation.

Empirical Analysis

Whether or not a smooth (or sharp) loss landscape is related to the generalization error, is under an active debate in the general machine learning community (Li et al. 2018). However, for the case of knowledge distillation it seems to have connections to better accuracy. It's believed that softened targets provide information on the similarity between output categories (Hinton, Vinyals, and Dean 2015). Furlanello et al. (2018) connected the knowledge distillation to a weighted/smoothed loss over classification labels. Importantly, Zhang et al. (2017) used posterior entropy and its flatness to make sense of the success of knowledge distillation. Supported by these prior works we propose to analyze the KD methods through loss landscape. In Figure 7, using a recent state of the art landscape visualization technique (Li et al. 2018) the loss surface of plain CNN on CIFAR-100 is plotted for student in three modes: (1) no knowledge distillation (NOKD), (2) baseline knowledge distillation (BLKD), (3) the proposed method (TAKD). It's seen that our network has a flatter surface around the local minima. This is related to robustness against noisy inputs which leads to better generalization.

Summary

We studied an under-explored yet important property in Knowledge Distillation of neural networks. We showed that the gap between student and teacher networks is a key to the efficacy of knowledge distillation and the student network performance may decrease when the gap is larger. We proposed a framework based on Teacher Assistant knowledge Distillation to remedy this situation. We demonstrated the effectiveness of our approach in various scenarios and studied its properties both empirically and theoretically. Designing a fully data-driven automated TA selection is an interesting venue for future work. We also would like to make a call for research on deriving tighter theoretical bounds and rigorous analysis for knowledge distillation.

Acknowledgement

Authors Mirzadeh and Ghasemzadeh were supported in part through grant CNS-1750679 from the United States National Science Foundation. The authors would like to thank Luke Metz, Rohan Anil, Sepehr Sameni, Hooman Shahrokhi, Janardhan Rao Doppa, and Hung Bui for their review and feedback.

References

Anil, R.; Pereyra, G.; Passos, A.; Ormándi, R.; Dahl, G.; and Hinton, G. 2018. Large scale distributed neural network training through online distillation. *CoRR* abs/1804.03235.

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

Bergstra, J.; Bardenet, R.; Bengio, Y.; and Kégl, B. 2011. Algorithms for hyper-parameter optimization. In *NIPS*.

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

Chen, G.; Choi, W.; Yu, X.; Han, T.; and Chandraker, M. 2017. Learning efficient object detection models with knowledge distillation. In *NIPS*, 742–751.

Czarnecki, W.; Osindero, S.; Jaderberg, M.; Swirszcz, G.; and Pascanu, R. 2017. Sobolev training for neural networks. In *NIPS*, 4278–4287.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR* abs/1810.04805.

Furlanello, T.; Lipton, Z.; Tschannen, M.; Itti, L.; and Anandkumar, A. 2018. Born again neural networks. *arXiv preprint arXiv:1805.04770*.

Han, K. J.; Chandrashekaran, A.; Kim, J.; and Lane, I. 2017. The capio 2017 conversational speech recognition system. *CoRR* abs/1801.00059.

Han, S.; Mao, H.; and Dally, W. J. 2016. Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding. In *4th International Conference on Learning Representations, ICLR 2016*.

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

Heo, B.; Lee, M.; Yun, S.; and Choi, J. 2018. Improving knowledge distillation with supporting adversarial samples. *arXiv preprint arXiv:1805.05532*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *CVPR*.

Huang, G.; Liu, Z.; Maaten, L.; and Weinberger, K. 2017. Densely connected convolutional networks. *CVPR* 2261–2269.

Li, H.; Kadav, A.; Durdanovic, I.; Samet, H.; and Graf, H. P. 2016. Pruning filters for efficient convnets. *CoRR* abs/1608.08710.

Li, H.; Xu, Z.; Taylor, G.; Studer, C.; and Goldstein, T. 2018. Visualizing the loss landscape of neural nets. In *NIPS*, 6391–6401.

Lopez-Paz, D.; Bottou, L.; Schölkopf, B.; and Vapnik, V. 2015. Unifying distillation and privileged information. *arXiv preprint arXiv:1511.03643*.

Microsoft-Research. 2018. Neural network intelligence toolkit.

Paszke, A.; Gross, S.; Chintala, S.; and et. al. 2017. Automatic differentiation in pytorch. In *NIPS Autodiff Workshop*.

Polino, A.; Pascanu, R.; and Alistarh, D. 2018. Model compression via distillation and quantization. In *ICLR*.

Romero, A.; Ballas, N.; Kahou, S.; Chassang, A.; Gatta, C.; and Bengio, Y. 2014. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*.

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

Schmitt, S.; Hudson, J.; Zidek, A.; and et. al. 2018. Kickstarting deep reinforcement learning. *CoRR* abs/1803.03835.

Tai, C.; Xiao, T.; Wang, X.; and Weinan, E. 2015. Convolutional neural networks with low-rank regularization. *CoRR* abs/1511.06067.

Tarvainen, A., and Valpola, H. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NIPS*.

Urban, G.; Geras, K. J.; Kahou, S. E.; and et. al. 2017. Do deep convolutional nets really need to be deep and convolutional? In *5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings*.

Vapnik, V. 1998. *Statistical learning theory. 1998*, volume 3. Wiley, New York.

Wang, J.; Bao, W.; Sun, L.; Zhu, X.; Cao, B.; and Yu, P. S. 2018a. Private model compression via knowledge distillation. *CoRR* abs/1811.05072.

Wang, X.; Zhang, R.; Sun, Y.; and Qi, J. 2018b. Kdgan: Knowledge distillation with generative adversarial networks. In *NIPS*, 783–794.

Wang, Y.; Xu, C.; Xu, C.; and Tao, D. 2018c. Adversarial learning of portable student networks. In *AAAI*.

Xu, Z.; Hsu, Y.; and Huang, J. 2018. Training shallow and thin networks for acceleration via knowledge distillation with conditional adversarial networks. In *6th International Conference on Learning Representations, ICLR 2018*.

Yim, J.; Joo, D.; Bae, J.; and Kim, J. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *CVPR*, 7130–7138.

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

Yu, R.; Li, A.; Morariu, V. I.; and Davis, L. 2017. Visual relationship detection with internal and external linguistic knowledge distillation. In *ICCV*.

Yu, R.; Li, A.; Chen, C.; Lai, J.; Morariu, V.; Han, X.; Gao, M.; Lin, C.; and Davis, L. 2018. Nisp: Pruning networks using neuron importance score propagation. In *CVPR*.

Z, S., and K, N. 2016. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*.

Zhang, Y.; Xiang, T.; Hospedales, T.; and Lu, H. 2017. Deep mutual learning. *CoRR* abs/1706.00384.