Knowledge Distillation:

Current Understanding and Future Directions in an Adversarial Context

- Luke Zhuo, Haoran Zhang, Martin Kuo

Duke PRATT SCHOOL of ENGINEERING

Abstract

Convolutional neural networks have been widely employed in graphic classification tasks. However, these heavily trained networks are too large to run on devices with limited computing resources like wearable devices and mobile phones. Consequently, researchers have sought to boost model accuracy while minimizing model size. Knowledge distillation (KD) achieves this goal by distilling the knowledge from a large model (or an ensemble of models) to a single small model. By reproducing, analyzing, and experimenting with reputed Knowledge Distillation works, this paper intends to provide additional insights into this field, particularly on student-teacher reverse distillation, self-distillation, and an approach to adversarial distillation training we devised which has shown promise.

Vanilla Distillation

The network architecture and the accuracy of the MNIST dataset with and without self-distillation are included in **Table 1** [1]. It is clear that the soft labels generated by the larger model do help to improve the accuracy of the smaller model (from 98.06% to 98.73%).

Table 1: Accuracy & Architecture comparison between large model and small model on MNIST

Model	Shape			
Large Model	(fc1) Linear(in_features=28*28, out_features=1200, bias=True) (fc2): Linear(in_features=1200, out_features=1200, bias=True) (fc3): Linear(in_features=1200, out_features=10, bias=True))			
	FLOPS	**# of Parameters**	**Accuracy**	
	2,392,800	940800+1440000+12000	0.9898	
Small Model	(fc1): Linear(in_features=28*28, out_features=400, bias=True) (fc2): Linear(in_features=400, out_features=10, bias=True)			
	FLOPS	**# of Parameters**	**Accuracy**	
	317,600	313600+4000	0.9806	0.9873

* Small Model accuracy trained w/o distillation (left) and with Distillation (right)

Afterwards, we randomly omit one digit from the MNIST datasets and see if the teacher can transfer knowledge to the student on digits it has not seen in the training process. Higher accuracy on seen and omitted data suggest transference of "dark knowledge" - privileged information without parameter gain - to the student in **Table 2**.

Table 2: Accuracy comparison with Distillation and digit omitted

Digit Omitted	Accuracy			
	Without Distillation (T = 1)		**Accuracy with Distillation (T = 10)**	
One	0.9803 (w/o one)	0.00 (only one)	0.9825 (w/o one)	0.9850 (only one)
Two	0.9812 (w/o two)	0.00 (only two)	0.9851 (w/o two)	0.9273 (only two)

Reversed Distillation

Table 3: Reversed Knowledge Distillation (Re-KD) Accuracy (mean ± std over 3 runs, accuracy in %)

Teacher: baseline	Student: baseline	Re-KD (S→T)
95.29±0.99	94.9±0.10	95.46±0.12

Re-KD [2] is a network architecture in which a student teaches a teacher. To implement Re-KD, we use ResNet18 as the student and ResNet50 as the teacher [6]. In **Table 3**, we can see that ResNet50 improved by learning from ResNet18 by 0.17%. Our experiment results of Re-KD can be be explained as the soft targets providing model label smoothing regularization for the teacher.

Teacher-free Distillation

Table 4: Teacher-free Knowledge Distillation Accuracy (mean ± std over 3 runs, accuracy in %)

ResNet18: baseline	Tf KD-reg	Tf KD-self
94.98±0.10	95.39±0.15	95.42±0.42

Distillation via Manually Designed Regularization (Tf KD-reg):

For K classes, we assign probability $\delta \in [0.9, 1]$ to correct class, and $(1 - \delta) / (K - 1)$ to incorrect classes, before applying softmax to form a "virtual teacher" from the softmax outputs, which are shown in **Figure 1** [2]. We perform knowledge distillation from manually-designed virtual teacher to ResNet18 student.

Knowledge Distillation via Self-Training (Tf KD-self):

We perform distillation from a pretrained ResNet18 to another ResNet18 (self-distillation) [2]. Tf KD-reg and Tf KD-self both have higher accuracy than ResNet18 baseline, as evident in **Table 4**. With either the model itself as the teacher or no teacher entirely, teacher-free methods resemble Re-KD and benefit from the aforementioned label smoothing regularization.

Examining gradients for teacher-free approaches also indicates knowledge distillation mitigates the vanishing gradient problem known to plague deep neural networks [3]. Average gradients for layers in the Tf KD-reg and Tf KD-self models are significantly larger than those for a standard ResNet18, as shown in **Figure 2**.

Figure 1: Manually Designed Virtual Teacher Probabilities. Probability of the correct class and incorrect classes (for δ=0.9) at each temperature, the parameter controlling softmax confidence

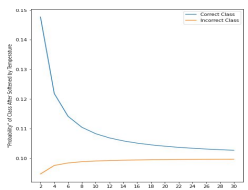
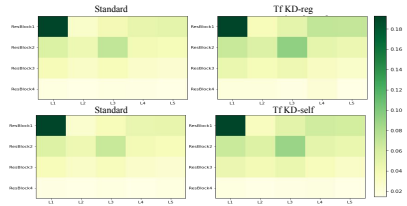


Figure 2: Heatmap Comparison of Average Gradients between Teacher-free Knowledge Distillation Approaches and ResNet18 without Knowledge Distillation. Average gradients for each layer at each stage (ResBlock) of the models



Adversarial Teacher-free Distillation

Table 5: Adversarially Trained Tf-KD experiment results on CIFAR=10 (accuracy in %)

Method of Knowledge Distillation	Clean Accuracy	AutoAttack Accuracy
ResNet18 (AT) Student, No Teacher	84.22	46.99
ResNet18 Student, ResNet18 (AT) Teacher	82.70	47.66
ResNet18 (AT) Tf KD-reg	82.02	48.28
ResNet18 (AT) Tf KD-self	81.53	48.83

We used Projected Gradient Descent attack with 7 iterations (PGD-7) to add intentional noise to data to lead the model to misclassify data.

To implement Adversarial Training Manually Designed Regularization (AT Tf KD-reg), we applied distillation from the virtual teacher's soft outputs to an adversarially trained ResNet18. To implement Adversarial Self-Training (AT Tf KD-self), we applied distillation from an adversarially pretrained ResNet18 to another ResNet18 (self-distillation) being trained on adversarial data. Both adversarially trained teacher-free distillation approaches introduced showed promise. **Table 5** shows both approaches had higher robustness to AutoAttack than the adversarially trained ResNet18 baseline and the standard ResNet18 student with adversarially trained ResNet18 teacher highlighted in [4]. We posit this greater robustness stems from label smoothing, which penalizes over-confidently and badly classified points [5].

Conclusions and Further Investigation

Knowledge distillation enables improved accuracy in a student model. This stems from factors which include dark knowledge, mitigating of the vanishing gradient problem, and label smoothing regularization, as evidenced by the results of vanilla, reversed, and teacher-free self-training and manually designed regularization approaches to knowledge distillation. The approaches devised to adversarially train teacher-free knowledge distillation models have showed initially promising results in promoting robustness against adversarial noise. Delving into the use of teacher-free knowledge distillation in an adversarial setting is an area in which we hope to both further examine with greater computation resources and open up to others to explore.

References

[1] Geoffrey Hinton, Jeff Dean, and Oriol Vinyals. Distilling the knowledge in a neural network. pages 1–9, 03133 2014.

[2] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisit knowledge distillation: a teacher-free framework. 2019.

[3] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. pages 3712–3721, 10 2019.

[4] Javier Maroto, Guillermo Ortiz-Jimenez, and Pascal Frossard. On the benefits of knowledge distillation for adversarial robustness. 03 2022.

[5] Morgane Goibert and Elvis Dohmatob. Adversarial robustness via label-smoothing. arXiv: Learning, 2020.

[6] Pytorch Vision.(2015). ResNet-18 and ResNet-50 from `Deep Residual Learning for Image Recognition. https://github.com/pytorch/vision/blob/main/torchvision/models/resnet.py