<Deep Learning : Statistical Perspective – Final Project>

Improved structure for Knowledge Distillation : Growing Student Knowledge Distillation

Chanhwa Lee 2014-18323
Seung Hoon Paik 2015-12868

Abstract. Knowledge Distillation is one of model compression methods, which uses the soft label of a large model to train a small but sufficiently accurate model. After the popularization of Knowledge Distillation (KD), there have been many papers looking for better KD frameworks. In this work, we propose a new, improved method of KD, Growing Student Knowledge Distillation (GSKD), which is motivated by the human life cycle. We justified the effect of "young" network in GSKD with ANOVA. The behavior of hyperparameters and maximal accuracy of models is clearly different from other KD methods previously introduced. Furthermore, we showed the superiority of GSKD with multiple experiments on CIFAR-10 and various ResNet architectures.

I. Introduction

After remarkable development of Deep Learning, other issues besides accuracy become important. One of these issues is computation, not only for model training but also for the computation of a well-trained model. A high cost of time and computation is naturally required for an accurate neural network. The critical problem is that a more substantial model gives better accuracy, but has more limitations in deployment. Unlike developers, most users have tight restrictions (e.g., cell phones and IoT devices). Also, to use the model in real-time, reducing the computation is essential. Therefore, model compression is a significant issue.

Knowledge Distillation (KD) is one of the popular ways that compress model. It aims to create a small model emulating the accuracy of the larger one as possible. The idea of KD was first introduced by Bucilua et al. [1] and popularized by Hinton et al. [2]. The pre-trained large model is called "Teacher," and the small objective model is called "Student." In Hinton's model, the "soft label" of the teacher for each data is transferred to the student, which takes a vital role in the student's loss function. Soft label, or soft probability, refers to the class score, which is the result of softmax at the last layer. After Hinton et al., finding better student consturction method has been a popular topic for KD related works.

We propose a new type of distillation structure called Growing Student Knowledge

Distillation (GSKD), which is inspired by the principles of learning in the real world.

The main contributions of this paper are as follows :
- Proposed a new KD method, Growing Student Knowledge Distillation (GSKD)
- Using ANOVA, justified the effect of "young" network KD
- Showed the superiority of GSKD with experiments
- Observed the characteristic of hyperparameters in GSKD and compared to BLKD

II. Related Works and Background

Model Compression

The methods for model compression can be summarized as four main strategies [3]; Parameter pruning and sharing(reducing redundant parameters which are not sensitive to the performance), Low-rank factorization(using matrix/tensor decomposition to estimate the informative parameters), Transferred/compact convolutional filters(designing special structural convolutional filters to save parameters), Knowledge Distillation(training a compact neural network with distilled knowledge of a large model).

BLKD

Baseline Knowledge Distillation (BLKD) is the simplest KD framework. It starts from one trained large model called teacher, and create one smaller model called student. It becomes popular through Hinton et al. [2]. The main idea is that not only the true label of data but also 'soft labels' of the teacher provide information to the student. Interestingly, for the same network structure, a student trained by BLKD performs better than a student trained using a standard loss, cross-entropy.

TAKD

Mirzadeh et al. [4] found that the size gap between teacher and student is important in training. The main point is, for fixed student size, the larger teacher does not always guarantee a better student. To reduce the negative effect of the size gap, they proposed "Teacher Assistant Knowledge Distillation (TAKD)" with adding a mid-size network called "Teacher Assistant (TA)" between teacher and student. With experiments in CIFAR-10 and CIFAR-100 dataset, they reported the positive effect of TA in accuracy increasing. Moreover, they compared the experimental result of TA paths for the same teacher and student; for example, Path1: 10 layers CNN(Teacher) → 6 layers CNN(TA) → 4 layers CNN(Student) / Path2: 10 layers CNN(Teacher) → 8 layers CNN(TA) → 6 layers CNN(TA)→ 4 layers CNN(Student).

ReKD / DeKD / TfKD

Yuan et al. [5] suggested three other variations of KD. Two of them are named as "Reversed Knowledge Distillation (ReKD)" that student teaches teacher, and "Defective Knowledge Distillation (DeKD)" that poorly trained teacher teaches student. Both models are quite counterintuitive since insufficient, sometimes wrong, information is provided for training. However, the interesting result is that two models enhance the accuracy of student in certain conditions. Yuan explained these phenomena with Label Smoothing Regularization (LSR). LSR is introduced by Szegedy et al. [6] to replace the hard label (one-hot vector output) with smoothened labels. With LSR viewpoint, KD can be considered as a variation of LSR where the smoothing distribution is learned but not pre-defined [5]. Through this explanation, Yuan lastly proposed Teacher-free KD (TfKD) that student trains itself, so-called self-training, and checked the positive effect through the experiment.

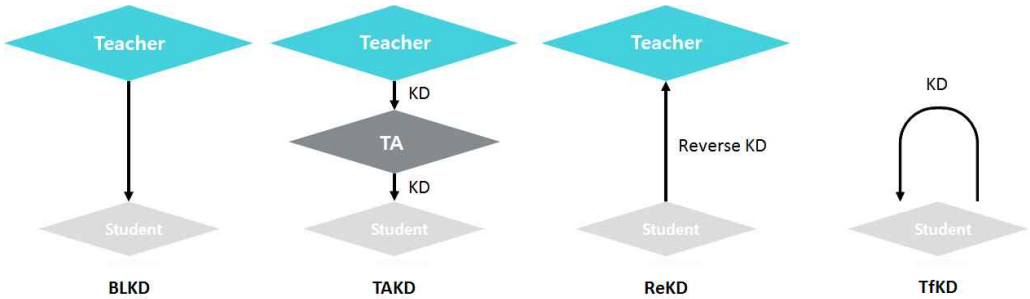

Figure 1. Concept of BLKD, TAKD, ReKD, TfKD

III. Growing Student Knowledge Distillation (GSKD)

Model description

The idea of GSKD comes from the human life cycle. It reflects the academic level and brain growth of corresponding stages in the cycle. Teacher will train the smallest student. Then larger student of the next stage is affected by both teacher and the previous stage's student. A detailed description of the structure is as follows:

Step 1 Teacher trains the smallest size student with BLKD.
Step 2 Train the student of the next stage, which is larger than the previous one. It is trained by using the softened label of pre-trained teacher network and the previous stage student, say, "young" network.
Step 3 Repeat Step 2 until we get an objective size of the student.

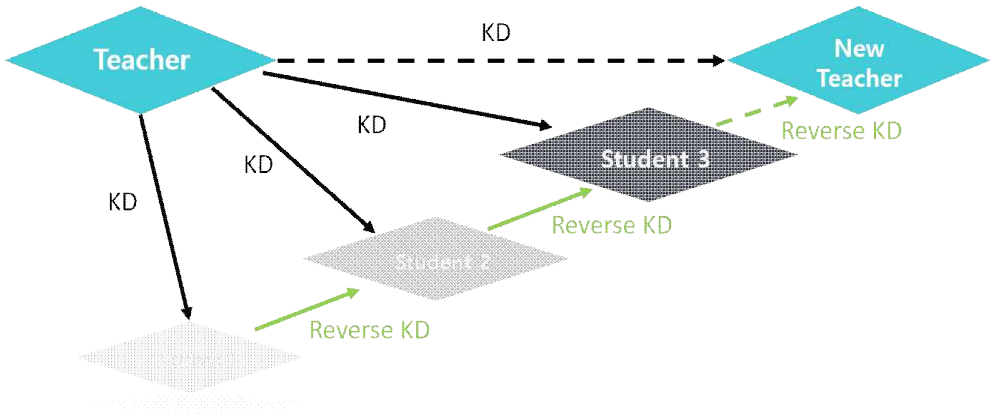

Figure 2. Structure of Growing Student KD model

At Step 2, we used the loss function similar to BLKD, adding KL-divergence between student and young as well as KL-divergence between student and teacher. We repeated Step 2 until we reached the size of student becomes equal to that of teacher, and let this student be a new teacher.

Notations

Neural networks typically produce class probability, or class score, by applying a softmax function to the logit(the value of the last layer) denoted by $a(i)$, where $i \in \{1, \cdots, K\}$ represents the class. Then, class probability, $p^{\tau}(i)$, is given by :

$$p^{\tau}(i) = \frac{\exp(a(i)/\tau)}{\Sigma_j \exp(a(j)/\tau)}$$

where τ is a temperature that is typically set to 1. Using a higher value for τ produces a softer probability distribution $p^{\tau}(i)$ over classes. Denote $p^{\tau} = (p^{\tau}(1), \cdots, p^{\tau}(K))^{\top}$ for softened probability distribution over K classes.

In classical supervised learning, the mismatch between output of the neural network and ground-truth label is usually computed by cross-entropy loss

$$H(p, q) = -\sum_{i=1}^{K} q(i) \log(p(i))$$

where q is ground truth distribution over labels. For a single ground-truth label, $q(i) = 1$ for the true label i and $q(j) = 0$ otherwise.

In knowledge distillation, the information provided by teacher network is used to train student network, through KL-divergence loss with respect to temperature

$$\tau^2 KL(p_s^{\tau}, p_t^{\tau})$$

where p_s^τ, p_t^τ are the softened probability distribution of student and teacher networks, respectively.

The loss function of BLKD model consists of two parts. The first part is the standard loss between p_s and q, say, $L_{SL} = H(p_s, q)$. The second part is KL-divergence of the soft label, say, $L_{KL}^{teacher} = \tau^2 KL(p_s^\tau, p_t^\tau)$. Then the BLKD loss function is:

$$L_{KD} = (1-\lambda) L_{SL} + \lambda L_{KL}^{teacher}$$

where λ is a hyperparameter controlling trade-off between losses.

In our proposed model, GSKD, student network is trained under the following loss function:

$$L_{KD} = (1-\lambda-\gamma) L_{SL} + \lambda L_{KL}^{teacher} + \gamma L_{KL}^{young}$$

where $L_{KL}^{young} = \tau^2 KL(p_s^\tau, p_y^\tau)$, p_y^τ is softened probability distribution of young network, and λ, γ are hyperparameters controlling the trade-off between losses.

IV. Experiment

Dataset

We performed a set of experiments on CIFAR-10. CIFAR-10 is a standard image classification dataset with 10 classes and 32x32 resolution, consisting of 50k training images and 10k test images.

Network Architectures

We used the structures proposed in the original paper of He et al. [7], ResNet8, ResNet14, ResNet20, and ResNet26. The number of blocks in the ResNet architecture is served as a proxy for the size of the network. We trained ResNet26 20 times and obtained the maximal accuracy, 89.71%, and this network was used as a teacher classifier. ResNet8, ResNet14, ResNet20, and ResNet26 were trained as student classifiers. All classifiers were trained over 80 epochs.

Training methods

We trained the student classifiers in three different ways. The first method is NOKD (No KD), which uses only the standard cross-entropy loss for training. The second method is BLKD, which is implemented in the same way as Hinton et al. [2] did. The last method is GSKD we propose. The performance of all classifiers was measured in terms of accuracy.

Implementation details

We followed the basic implementation setting of Heo et al. [8]. We used the random crop for data augmentation and normalized an input image based on the mean and the variance of the dataset. The temperature(τ) of the KD loss was fixed to 3 in all experiments. The learning process was performed with 128 batch size, with a learning rate that started at 0.1 and decreased to 0.01 at half of the maximum epoch, and to 0.001 in 3/4 of the maximum epoch. The momentum used in the study was 0.9, and the weight decay was 0.0001.

The setting for hyperparameters λ and γ is as shown in Table 1 and 2.

Table 1. Hyperparameter setting for BLKD

Setting no.	λ	**# of repetition**
1	0.01	4
2	0.025	4
3	0.05	4
4	0.1	8
5	0.4	8
6	0.8	4
7	0.9	4
total		36

Table 2. Hyperparameter setting for GSKD

Setting no.	λ	γ	**# of repetition**
1	0.01	0.01	8
2	0.025	0.025	8
3	0.05	0.05	8
4	0.1	0.05	8
5	0.1	0.8	8
6	0.4	0.1	8
7	0.4	0.4	8
8	0.8	0.1	8
9	0.9	0.05	8
total			72

The numbers of experiments for each ResNet model and KD method are as shown in Table 3.

Table 3. The numbers of experiments

	ResNet8	**ResNet14**	**ResNet20**	**ResNet26**
NOKD	10	10	10	20†
BLKD	72††	36	36	36
GSKD	72	72	72	72

† To get the best teacher, we trained ResNet26 more than other structures with NOKD

†† Since ResNet8 is the 'youngest' model, BLKD is equal to GSKD

Note that 72 experiments for GSKD are composed of 8 experiments per each hyperparameter setting in Table 2, and BLKD for ResNet26 is the same as TfKD in [5]. We used PyTorch framework for the implementation and as a preprocessing step on CIFAR-10 dataset. Utilizing Google Colaboratory, we used NVIDIA Tesla P100-PCIE-16GB GPU to train networks. We referred PyTorch implementation source code used in Mirzadeh et al. [4].

V. Results and Observations

1. Justification of young network knowledge distillation

In BLKD, accuracies of trained student networks did not vary widely except for hyperparameter $\lambda = 0.8$ & 0.9 (setting no. 6 & 7). Indeed, networks trained by BLKD showed lower accuracy than networks trained by NOKD in these settings. However, in GSKD, the obtained accuracies varied with hyperparameter settings. Since we considered each hyperparameter setting as treatment, we applied a one-way classification model to the obtained accuracies. Excepting the accuracies from these inefficient hyperparameter $\lambda = 0.8$ & 0.9, we performed ANOVA to check whether there is a difference among hyperparameter settings. The result of the analysis is shown in Table 4.

$\mathbf{H_0}$: setting (treatment) effects on accuracy are equal ***vs*** $\mathbf{H_1}$: not H_0

Table 4. p-value of ANOVA for testing the effect of young network KD

	ResNet14	**ResNet20**	**ResNet26**
BLKD	0.492	0.479	0.304
GSKD	< 0.001	< 0.001	< 0.001

Since p-values for every network in BLKD are significantly high, it meant that there was no difference in the effect of hyperparameter λ on accuracy. In contrast, p-values are smaller than 0.001 in case of GSKD, which indicated that the hyperparameter settings influenced accuracies of networks. Since BLKD and GSKD shared the same λ values, hypothesis rejection in GSKD and failure to reject H_0 in BLKD demonstrated that Knowledge Distillation from young network directly impacted the accuracy of student networks. Therefore, our proposal that exploited the knowledge distillation from young network is a plausible idea, and the superiority of GSKD is described in the next section.

2. Accuracy improvement via GSKD

Maximal accuracy of trained students varied with the size of networks and training methods. In Table 5, GSKD consistently showed the highest accuracy among the three methods, excepting in case of ResNet8. Since ResNet8 is the "youngest" network in our GSKD structure, ResNet8 was trained only through BLKD. It seems that the size gap of teacher and student affected negatively in KD. The size gap issue has been reported in previous works [4].

Table 5. Maximal accuracy of trained student networks

	ResNet8	ResNet14	ResNet20	ResNet26
NOKD	**85.21**	88.56	89.2	89.71
BLKD	84.89	88.73	89.49	**89.85**
GSKD	84.89	**88.74**	**89.76**	**89.85**

3. Behaviour of hyperparameters

We observed the relation between hyperparameters and maximal model accuracy with interpolated graphs. The following graphs show the maximal model accuracy with respect to λ, which means we chose smaller γ for same λ according to Observation 1. In the graphs, the x-axis is for $\lambda \in [0,1]$ and the y-axis is for maximal model accuracy(%).

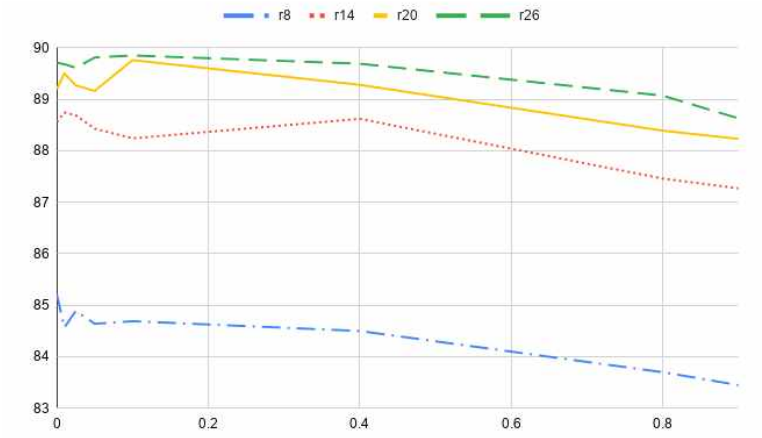

Figure 4. Overall graph for all ResNet trained by GSKD

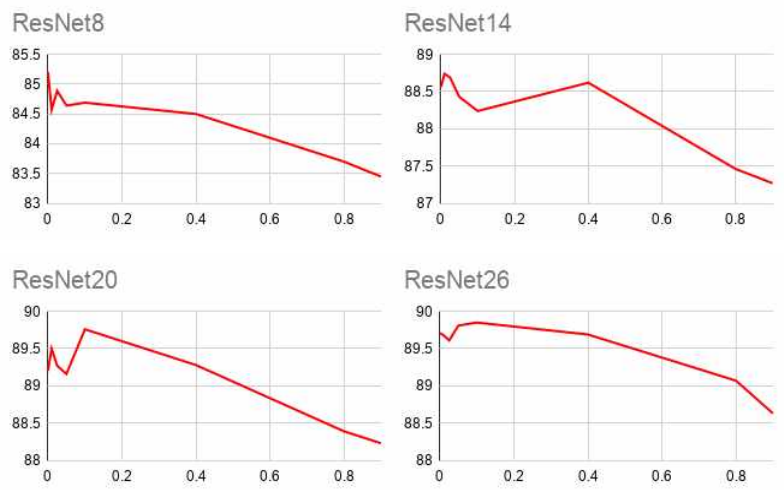

Figure 5. Maximum model accuracy for each ResNet

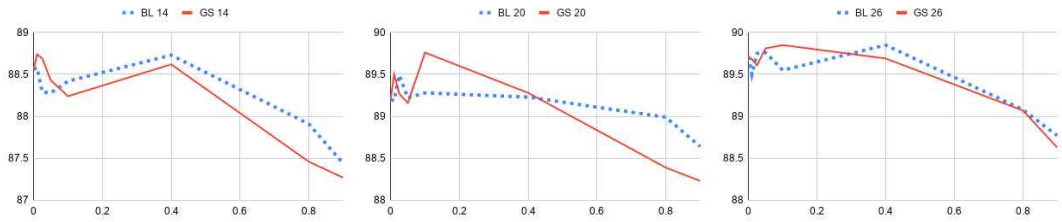

Figure 6. Comparing maximum accuracy of BLKD and GSKD (From left to right, ResNet14, ResNet20, ResNet26)

Note that the difference between the loss functions of BLKD and GSKD is γ and L_{KL}^{young}, and BLKD can be considered as a GSKD with $\gamma = 0$. It means that our proposed GSKD model conceptually contains BLKD method, and thus BLKD is a special version of GSKD. Figure 5 shows the maximal accuracy of each ResNet model trained by GSKD, and Figure 6 shows the differences between BLKD and GSKD in ResNet 14, 20, and 26.

The followings are observations from the figures.

Observation 1 In GSKD, it seems that a model with smaller γ gives higher accuracy in the same λ setting (refer to Appendix; setting no. 4, 5 and no. 6, 7 in the first table). It means that young model's information should not affect a lot to the next stage for better accuracy. It implies that the truth-like teacher's information is more critical than the childish student's information.

Observation 2 In the same lambda, the maximal accuracy of given model can not exceed those of larger models (See Figure 4).

Observation 3 In each model, the maximal accuracy is not a decreasing function of λ. Since increasing λ means that the effect of ground-truth is decreasing, in some cases teacher's soft label helps more than the ground-truth label, which is a counterintuitive result. Similar results have been reported in other KD works, and we checked that it appears in GSKD also.

4. Relation to Label Smoothing Regularization

Yuan et al. [5] introduced an probable relation between KD and Line Smoothing Regularization (LSR). LSR works quite well in several applications. However, it is still experimental and does not have a theoretical background. A plausible explanation is that LSR is just an appropriate regularization. In this viewpoint, GSKD is a specialized extended version of LSR, with additional KL-divergence term and parameter γ for young network's knowledge distillation.

VI. Conclusion and Future Work

Conclusion

We proposed a new Knowledge Distillation method, GSKD. We showed that GSKD achieves higher maximal accuracy than NOKD and BLKD. We checked the effect of young network in GSKD by ANOVA. The behaviour of hyperparmeters (λ, γ) were counterintuitive and engaging. In particular, as shown in the graphs, the behavior of hyperparameter in GSKD is clearly different from that of BLKD.

Future work

We trained more than 400 ResNets using Google Colab. However, if we performed more experiments, we could have more valuable and clear observations. Hyperparameter tuning was much more complicated than we expected. By using the Microsoft NNI toolkit, optimal hyperparameters would be more easily found. We did our research based on CIFAR-10 dataset. However, there are many other image datasets such as CIFAR-100. Applying GSKD and verifying its effect in other datasets will be meaningful.

VII. References

[1] Buciluǎ, Cristian, Rich Caruana, and Alexandru Niculescu-Mizil. "Model compressio n." Proceedings of the 12th ACM SIGKDD international conference on Knowledge disco very and data mining. ACM, 2006.

[2] Hinton, Geoffrey, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neura l network." arXiv preprint arXiv:1503.02531 (2015).

[3] Cheng, Yu, et al. "A survey of model compression and acceleration for deep neural networks." arXiv preprint arXiv:1710.09282 (2017).

[4] Mirzadeh, Seyed-Iman, et al. "Improved knowledge distillation via teacher assistant: Bridging the gap between student and teacher." arXiv preprint arXiv:1902.03393 (2019).

[5] Yuan, Li, et al. "Revisit Knowledge Distillation: a Teacher-free Framework." arXiv p reprint arXiv:1909.11723 (2019).

[6] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[7] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of t he IEEE conference on computer vision and pattern recognition. 2016.

[8] Heo, B., et al. "Improving knowledge distillation with supporting adversarial sample s." arXiv preprint arXiv:1805.05532 3 (2018).

A. Appendix

1. maximal accuracy of ResNet trained by GSKD

setting no.	λ	γ	ResNet8	ResNet14	ResNet20	ResNet26
1	0.01	0.01	84.57	88.74	89.50	89.68
2	0.025	0.025	84.89	88.69	89.27	89.61
3	0.05	0.05	84.64	88.43	89.16	89.81
4	0.1	0.05	84.69	88.24	89.76	89.85
5	0.1	0.8	84.6	85.36	85.85	86.4
6	0.4	0.1	84.5	88.62	89.28	89.69
7	0.4	0.4	84.19	87.06	88.29	88.96
8	0.8	0.1	83.7	87.46	88.39	89.07
9	0.9	0.05	83.45	87.27	88.23	88.63

2. maximal accuracy of ResNet trained by BLKD

setting no.	λ	ResNet8	ResNet14	ResNet20	ResNet26
1	0.01	-	88.58	89.19	89.5
2	0.025	-	88.29	89.49	89.75
3	0.05	-	88.28	89.22	89.76
4	0.1	-	88.42	89.28	89.55
5	0.4	-	88.73	89.23	89.85
6	0.8	-	87.91	88.99	89.08
7	0.9	-	87.44	88.64	88.77