

Densely Guided Knowledge Distillation using Multiple Teacher Assistants

Wonchul Son, Jaemin Na, Junyong Choi, Wonjun Hwang
Ajou University, Republic of Korea

{dnjscjf92, osial46, chldusxkr, wjhwang}@ajou.ac.kr

Abstract

With the success of deep neural networks, knowledge distillation which guides the learning of a small student network from a large teacher network is being actively studied for model compression and transfer learning. However, few studies have been performed to resolve the poor learning issue of the student network when the student and teacher model sizes significantly differ. In this paper, we propose a densely guided knowledge distillation using multiple teacher assistants that gradually decreases the model size to efficiently bridge the large gap between the teacher and student networks. To stimulate more efficient learning of the student network, we guide each teacher assistant to every other smaller teacher assistants iteratively. Specifically, when teaching a smaller teacher assistant at the next step, the existing larger teacher assistants from the previous step are used as well as the teacher network. Moreover, we design stochastic teaching where, for each mini-batch, a teacher or teacher assistants are randomly dropped. This acts as a regularizer to improve the efficiency of teaching of the student network. Thus, the student can always learn salient distilled knowledge from the multiple sources. We verified the effectiveness of the proposed method for a classification task using CIFAR-10, CIFAR-100, and ImageNet. We also achieved significant performance improvements with various backbone architectures such as ResNet, WideResNet, and VGG.¹

1. Introduction

While deep learning-based methods [11, 16, 10, 2], e.g., convolutional neural networks (CNNs), have achieved very impressive results in terms of accuracy, there have been many trials [9, 15, 34, 14] to apply them to many applications such as classification, detection, and segmentation. Among these attempts, Knowledge Distillation (KD) [14, 28] transfers the knowledge of a teacher model (e.g., a deeper or wider network) in the form of soft prob-

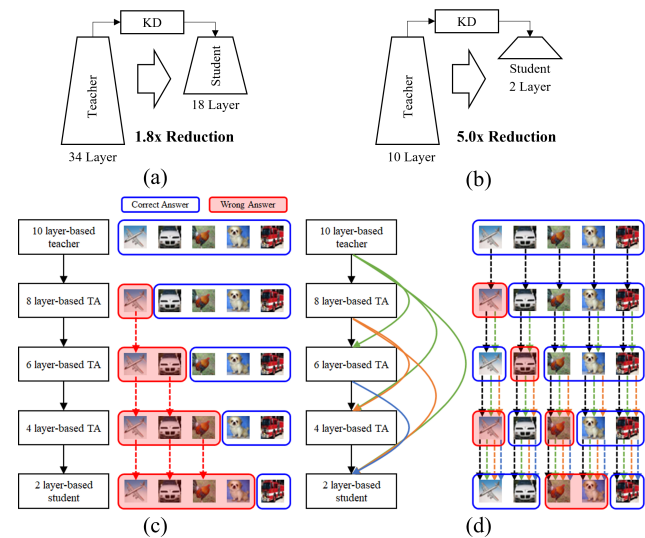


Figure 1. **Problem definition of the large gap between a teacher and a student network.** (a) In general, the difference between layers at KD is approximately 1.8 times, but (b) we are interested in the challenging problem of layer differences of more than 5 times. For solving this problem, TAKD [23] has been proposed. However, (c) TAKD has a fundamental limitation such as the error avalanche problem. Assuming that a unique error occurs one by one when a higher-level teacher assistant (TA) teaches a lower-level TA. The error case continues to increase whenever teaching more TAs. Meanwhile, in (d), the proposed densely guided knowledge distillation can be relatively free from this error avalanche problem because it does not teach TAs at each level alone.

ability (e.g., logits) to improve the accuracy of a less-parameterized student model (e.g., a shallower network) during a learning procedure. Specifically, the soft logits of the teacher network can train the student network more efficiently than the softmax based on the class label of the student network itself. Many studies [14, 28, 40, 36, 42, 33, 3] on the KD method have been proposed, most of which focused on effectively guiding a teacher’s soft probability or outputs to a student. Recently, there have been ensemble-based attempts [4, 41, 38, 22] to train a student network based on many peers or students without considering the single teacher network, which is a slight lack of consider-

¹Our code is available at <https://github.com/wonchulSon/DGKD>.

ation for the diversity of ensemble classifiers teaching the student, especially when the gap between a teacher and a student is large like Figure 1 (a) and (b).

In [5, 18], it was shown that a teacher does not always have to be smart for a student to learn effectively. The KD can not succeed when the student’s capacity is too low to successfully mimic the teacher’s knowledge. Recently, to overcome this problem, TA-based knowledge distillation (TAKD) [23] using intermediate-sized assistant models was introduced to alleviate the poor learning of a student network when the size gap between a student and a teacher is large. It achieved an effective performance improvement in the case of a large gap in teacher and student sizes. However, further studies are required to determine whether using middle-sized assistant models in series is the most efficient KD method for bridging the gap between a teacher and a student. For example, TAKD tends to cause the error avalanche problem, as shown in Figure 1 (c). It sequentially trains the multiple TA models by decreasing the capacity of their own assistant models for efficient student learning. If an error occurred during a specific TA model learning, this TA model will teach the next level assistant models including the same error. From then on, each time a TA is trained, the error snowballs gradually, as illustrated in Figure 1 (c). This error avalanche problem becomes an obstacle to improving the student model’s performance as the total number of TAs increases.

In this paper, we propose a novel densely guided knowledge distillation (DGKD) using multiple TAs for efficient learning of the student model despite the large size gap between a teacher and a student model. As shown in Figure 1 (d), unlike TAKD, when learning a TA whose model size gradually decreases for the target student, the knowledge is not distilled only from the higher-level TA but guided from all previously learned higher-level TAs including the teacher. Thus, a trainee had distilled knowledge by considering the relationship between the multiple trainers (e.g., teacher and TAs) with complementary characteristics. The error avalanche problem could be alleviated successfully. It is largely because the distilled knowledge previously used for the teaching of models disappears in TAKD, but the proposed method densely guides the whole distilled knowledge to the target network. In the end, the closer we are to student learning, the more educators we have, e.g., TAs and the teacher. Therefore, the final student model can get more opportunities to achieve better results.

For stochastic learning of a student model, we randomly remove a fraction of the guided knowledge from trainers during the student training, which is inspired from [31, 17]. Eventually, the student network is taught from trainers ensembled slightly different for each iteration; this acts as a kind of regularization to solve the problem of overfitting which can often occur when a simple student learns from a

complex teacher groups.

The major contributions of this paper are summarized as follows:

- We propose a DGKD that densely guides each TA network with the higher-level TAs as well as the teacher and it helps to alleviate the error avalanche problem whose probability of occurrence increases as the number of TAs increases.
- We revise a stochastic DGKD learning algorithm to train the student network from the teacher and multiple TAs efficiently.
- We demonstrate the significant accuracy improvement gained by the proposed method over well-known KD methods through extensive experiments on various datasets and network architectures.

2. Related Work

Knowledge Distillation: Knowledge distillation is a popular research topic in the field of model compression [9, 37]. We can extract distilled knowledge from a teacher network and transfer it to a student network to mimic the teacher network. The basic concept of knowledge distillation [14] is to compress the knowledge of a deeper or larger model to a single computational efficient neural network. After this study, extensive research was conducted on knowledge distillation. Remero et al. [28] introduced the transfer of a hidden activation output and Zagoruyko et al. [40] proposed transferring attention information as knowledge. Yim et al. [36] defined distilled knowledge from the teacher network as the flow of the solution process (FSP), which is calculated as the inner product between feature maps from two selected layers.

Recently, Tung et al. [33] introduced similarity-preserving knowledge distillation guided training of a student network such that input pairs that produce similar activations in the teacher network produce similar activations in the student network. Zhang et al. [41] proposed self-distillation in which student networks train the knowledge by themselves from deeper to shallower layers so that a teacher network is not required. Because the training of the pre-trained teacher model is not required, the time for training the student model can be reduced. Contrarily, Shen et al. [29] believed that a student network can learn knowledge efficiently from an ensemble of teacher networks; they proposed the use of an adversarial-based learning strategy with a block-wise training loss. For an online distillation framework, Zhang et al. [42] suggested that peer students learn from each other through the cross-entropy loss between each pair of students. Chen et al. [4] also suggested using peers where multiple student models train each student model based on auxiliary peers and one group leader.

Guo et al. [8] proposed collaborative learning-based online knowledge distillation that trained students without a teacher where knowledge is transferred among arbitrary students during collaborative training.

There have also been recent attempts to break away from the traditional method. Xu et al. [35] showed that contrastive learning as a self-supervision task helps to gain more rounded knowledge from a teacher network. Yuan et al. [38] addressed the time-consuming training procedure of the traditional knowledge distillation methods using the few-sample where the teacher model was compressed and the student-teacher model was aligned and merged with additional layers.

Large Capacity Gap Between Teacher and Student:

There are contrasting views regarding whether a good teacher always teaches a student well. Cho and Hariharan [5] found that knowledge distillation cannot succeed when the student model capacity is too low to mimic the teacher model; they presented an approach to mitigate this issue by stopping teacher training early to recover a solution more amenable for the student model. Similar to [7], Jin et al. [18] also constructed a gradually mimicking sequence of learning the teacher, which supervised the student with the teacher’s optimization route. From a similar perspective, Mirzadeh et al. [23] insisted that the student network performance may decrease when the capacity gap between the teacher and the student is large and introduced the multi-step knowledge distillation that employs the intermediate TA to bridge the gap between the teacher and student network. They showed that more distillation steps make a better student, up to three steps. However, considering resource constraints, they added that even one step can be effective.

Our proposed method differs from existing methods as we densely guide the student network using all assistant models generated along the way from the teacher to the student for overcoming the large gap between them. Note that our approach does not simply rely on a single model to teach the student, but uses all models that gradually become similar to the teacher’s characteristics; it will be helpful to avoid the error avalanche problem.

3. Densely Guided Knowledge Distillation using Teacher Assistants

3.1. Background

The key concept of knowledge distillation [14] is training the student network to mimic the teacher network output. To achieve this goal, the fully connected layer output, logits, is used as the knowledge of the network. The loss L_{KD} of the Kullback–Leibler (KL) divergence consists of the softened output of the teacher and student networks, de-

finied as follows:

$$L_{KD} = \tau^2 KL(y_S, y_T), \quad (1)$$

where τ is the temperature parameter to control the softening of signals. z_S and z_T refer to the teacher and student logits, respectively and each network’s output is $y_S = \text{softmax}(z_S/\tau)$ and $y_T = \text{softmax}(z_T/\tau)$.

With the distillation loss, from equation (1), to learn the original supervision signal, the cross-entropy loss L_{CE} needs to be added with the label y as follows:

$$L_{CE} = H(\text{softmax}(z_S), y). \quad (2)$$

As a result, the final loss function of the conventional KD is written with the balancing parameter λ as follows:

$$L = (1 - \lambda)L_{CE} + \lambda L_{KD}. \quad (3)$$

3.2. Proposed Method

When the teacher’s performance is good, the main purpose of KD is basically a trial to guide the student knowledge at a level close to that of a teacher. However, when the gap between a student and a teacher is large in terms of the size of weight parameter or the number of layers, the best teacher does not always guide the student properly [5]. To solve this problem, TAKD [23] was proposed using intermediate-sized networks such as a TA to bridge the large gap between a teacher and a student. TAKD improved the student learning efficiency by sequentially deploying the TAs from the teacher to the student. However, TAs are smarter than the student but worse than the teacher; this eventually becomes an obstacle to further student learning due to the limited knowledge of TAs. In the end, for teaching the student well, we need a smart teacher but it is a contradiction that most of good teacher networks have many parameters, which creates a network gap with the student at the same time.

Densely Guided Knowledge Distillation: In this paper, to overcome this contradiction and achieve better performance of a shallow student network, we propose densely guided knowledge distillation using multiple TAs which are trained sequentially. As shown in Figure 2, we make use of the distilled knowledge from intermediate-sized TAs and the teacher for teaching the student network. Moreover, this densely connected distillation form is also used when teaching TAs. Note that when designing the proposed training framework for KD, the underlying idea came from DenseNet [16] which is a densely connected residual network architecture for a classification task.

We can use several distillation losses between the assistant models and the student. For easy understanding, if there are two TA models A_1 and A_2 with the teacher model T , the loss of each TA can be written as follows:

$$\begin{aligned} L_{A_1} &= L_{T \rightarrow A_1}, \\ L_{A_2} &= L_{T \rightarrow A_2} + L_{A_1 \rightarrow A_2}, \end{aligned} \quad (4)$$

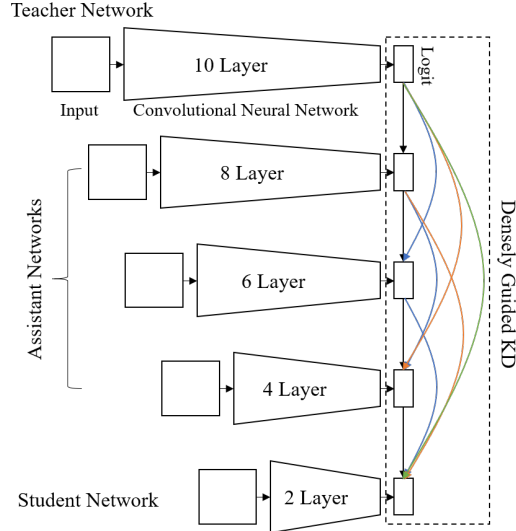


Figure 2. **Overview of the proposed method.** Our densely guided knowledge distillation using multiple teacher assistant networks is able to train a small-sized student network from a large-sized teacher network efficiently through the multiple teacher assistant networks.

where the right arrow at the subscript indicates the teaching direction. The student’s loss L_S is finally derived as follows:

$$L_S = L_{T \rightarrow S} + L_{A_1 \rightarrow S} + L_{A_2 \rightarrow S}, \quad (5)$$

where we guide the student network using the distilled knowledge from two TAs and a teacher. Equation (5) can be expressed in the same form as equation (3) as follows:

$$L_S = (1 - \lambda_1)L_{CE_S} + \lambda_1 L_{KD_{T \rightarrow S}} + (1 - \lambda_2)L_{CE_S} + \lambda_2 L_{KD_{A_1 \rightarrow S}} + (1 - \lambda_3)L_{CE_S} + \lambda_3 L_{KD_{A_2 \rightarrow S}}. \quad (6)$$

If there are n assistant models and assuming that λ retains the same value for simplicity, the general form of the total loss is derived as the follows:

$$L_S = (n+1)(1-\lambda)L_{CE_S} + \lambda(L_{KD_{T \rightarrow S}} + \sum_{i=1}^n L_{KD_{A_i \rightarrow S}}). \quad (7)$$

We distill complementary knowledge from each assistant network that has been previously learned and teach the student network with combined knowledge from all teacher assistant models. Consequently, the student network tries to mimic the various logit distributions ranging from the larger teacher network to the smaller TA network, resulting in improvement of the learning efficiency of the student network even with a large gap.

Stochastic DGKD: For efficient learning with the proposed DGKD, we adopt a stochastic learning strategy [31,

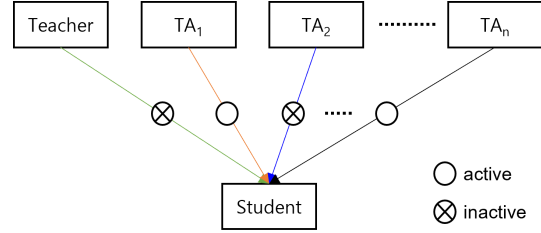


Figure 3. **Concept of the stochastic DGKD method.** When having n TAs and the teacher network, the student network is distilled by them. Depending on the survival probability, some of them could be randomly dropped out from a group of teachers at every training iteration.

[17] that randomly cuts the knowledge connections between many TAs and a student for each sample or mini-batch; we named our method as stochastic DGKD. Learning with a stochastic strategy is based on a simple intuition. We have multiple assistant models for teaching the shallow student network with a large gap, which would cause an overfitting problem due to the complex logit distribution of the TA ensemble as well as the teacher. It is necessary that the knowledge connection from the teachers is irregularly altered during the training, thus we randomly select the combination of distilled knowledge from the complex teacher and teacher assistant models. This process acts as a regularization function and it alleviates the overfitting problem. If the number of teacher and teacher assistants increases, it can relatively make the learning procedure simple.

For this purpose, we set $b_i \in \{0, 1\}$ as a Bernoulli random variable and $b_i = 1$ is active and $b_i = 0$ is inactive during KD learning. The survival probability for Bernoulli random variable is noted by $p_i = Pr(b_i = 1)$. Eventually, Equation (7) is updated by replacing L_{KD} with $b_i \cdot L_{KD_i}$. In this paper, we use a simple dropping rule as shown in Figure 3. The possibility of dropping knowledge is equal from the teacher to the last TA and it is only applied when teaching the student because the last student has the most enough candidates to teach the distilled knowledge to the target. In the experiment section, we perform a more detailed empirical comparison of the different survival probabilities.

4. Experimental Setting

Datasets: For fair comparisons, we evaluate the KD methods using CIFAR [20] and ImageNet [6] datasets, which are widely used as benchmarks for image classification. The CIFAR datasets comprise RGB images of size 32×32 with 50,000 training images and 10,000 testing images. There are two kinds of datasets, CIFAR-10 and CIFAR-100, which have 10 and 100 classes, respectively. The ImageNet dataset contains 1,281,167 images from 1,000 classes for training and 50,000 images for validation.

Networks: We performed experiments using various

Table 1. Accuracy comparison with all distillation steps using plain CNNs (e.g., teacher T_{10} , teacher assistants A_8, A_6, A_4 , student S_2) on CIFAR-100. The numbers denoted by * method came from the corresponding paper.

Step	TAKD*	DGKD
Teacher (10 layer)	56.19	56.15
Student (2 layer)	41.09	41.06
$T_{10} \rightarrow A_8$	56.75	56.72
$T_{10} \rightarrow A_8 \rightarrow A_6$	57.53	60.15
$T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4$	52.87	57.63
$T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$	45.14	48.92

networks: plain CNN and ResNet [11], WideResNet (WRN) [39], and VGG [30]. In this paper, the baseline architecture is a plain CNN, which is a VGG-like network. Following the experimental protocols of TAKD, we use a plain CNN architecture with a 10 layer-based teacher model, TA models with 8, 6, 4 layers, and a 2 layer-based student model. For a detailed comparison, we use 9, 7, 5, and 3 layer-based assistant models by removing the last convolution layer of layers 10, 8, 6, and 4.

Implementation Details: We set the implementation setting with a preprocessing, optimization, training plan, and learning rate, etc., using PyTorch [25]. We first used a random crop and random horizontal flip. We normalized for ResNet, WRN, and VGG except plain CNNs as done by TAKD. We used an initial learning rate of 0.1, stochastic gradient descent (SGD) optimizer with nesterov momentum 0.9, and weight decay $1e-4$ for 160 epochs. For plain CNN, we maintained the same learning rate for all epochs, but for ResNet, WRN, and VGG, we divided the learning rate by 0.1 on epochs 80 and 120. To obtain optimal performance, we use a hyperparameter optimization toolkit² using the same hyperparameter and seed setting, as done by TAKD. The survival probability p for the stochastic DGKD is added to balance parameter λ and temperature parameter τ . We reported the performance of the classification task for all experiments.

5. Result and Discussion

5.1. Ablation Study: Comparison with TAKD

In this section, we conduct comprehensive ablation studies to demonstrate the superiority of the proposed method by directly comparing it with TAKD. Basically, we retrain the whole network and follow the experimental protocol same to [23]. Tables 1 and 2 show that our method DGKD achieves better results compared with TAKD in all cases. For example, as shown in Table 1, the student model using the plain CNN network on CIFAR-100 shows 3.78% better performance than TAKD for the $T_{10} \rightarrow A_8 \rightarrow$

²Microsoft’s neural network intelligence toolkit

Table 2. Accuracy comparison with all distillation steps using ResNet (e.g., teacher T_{26} , teacher assistants A_{20}, A_{14} , and student S_8) on CIFAR-10. The numbers denoted by * method came from the corresponding paper.

Step	TAKD*	DGKD
Teacher (26 layer)	92.48	92.44
Student (8 layer)	86.61	86.56
$T_{26} \rightarrow A_{20}$	-	92.57
$T_{26} \rightarrow A_{20} \rightarrow A_{14}$	91.23	92.15
$T_{26} \rightarrow A_{20} \rightarrow A_{14} \rightarrow S_8$	88.01	89.02

$A_6 \rightarrow A_4 \rightarrow S_2$ path. Likewise, for ResNet on CIFAR-10, shown in Table 2, over 1% improvement is achieved for the $T_{26} \rightarrow A_{20} \rightarrow A_{14} \rightarrow S_8$ path. Moreover, we verified the steady improvements for the other steps. In particular, when there is just one TA, such as the $T_{10} \rightarrow A_8 \rightarrow A_6$ path in Table 1 and the $T_{26} \rightarrow A_{20} \rightarrow A_{14}$ path in Table 2, we confirm the good improvements by our method. Note that TAKD bridges the large gap between a teacher and a student through TAs and it is a good choice for this case. This is largely because it can play a positive role as a bridge to transfer dark knowledge sequentially. Simultaneously, it can play a negative role in error accumulation, such as error avalanche. In the extreme case, if only one TA is present, there is no way to overcome the TA’s inherent error. Even if the student can learn from the label, owing to its low capacity, it cannot get through this error by itself.

We conducted another comparison experiment using the large-scale dataset, ImageNet [6], presented in Table 3. In this experiment, we used a 34 layer-based teacher, a 18 layer-based student, and a single TA with 26 layers using the ResNet. Our method achieves over 1% better accuracy than Hinton’s KD and shows the best performance among CC [27], SP [33], online KD [21], KD [14], AT [40], CRD [32], SSKD [35], and TAKD [23] when the 34 layer teacher teaches the 18 layer student. As a result, we can conclude that our method works efficiently regardless of the database.

5.2. Ablation Study: Classifier Ensemble

We make experimental results regarding the ensemble classifiers in Table 4. The first ensemble is made by four different 10 layer-based teachers from scratch independently and the second ensemble is built by 10 layer-based teacher T_{10} , 8 layer-based teacher T_8 , 6 layer-based teacher T_6 , and 4 layer-based teacher T_4 . Note that these teachers are trained independently. Both these ensembles achieved better results than the KD [14] but they fail to overcome the large gap between the teacher and student models. On the other hand, TA-based methods including ours solve this problem successfully. From this result, we can infer that the simple ensemble method is not a good solution for the large gap between the teacher and student models in a KD

Table 3. Top-1 accuracy (%) on ImageNet. The teacher network is 34 layer-based ResNet (T_{34}) and the student network is 18 layer-based ResNet (S_{18}). We only use a single TA using 26 layer-based ResNet (A_{26}) for simplicity.

	Teacher	Student	CC [27]	SP [33]	Online KD [21]	KD [14]	AT [40]	CRD [32]	SSKD [35]	TAKD [23]	DGKD
Top-1	73.3	69.75	69.96	70.62	70.55	70.66	70.7	71.38	71.62	71.37	71.73
Top-5	91.42	89.07	89.17	89.8	89.59	89.88	90	90.49	90.67	90.27	90.82

Table 4. Comparison results with the ensemble methods using plain CNNs on CIFAR-100.

Method	Accuracy
Teacher (10 layer)	56.15
Student (2 layer)	41.06
KD [14]	42.56
TAKD using three TAs [23]	45.14
Ensemble using four T_{10} s	42.57
Ensemble using T_{10} , T_8 , T_6 , and T_4	43.25
DGKD	48.92

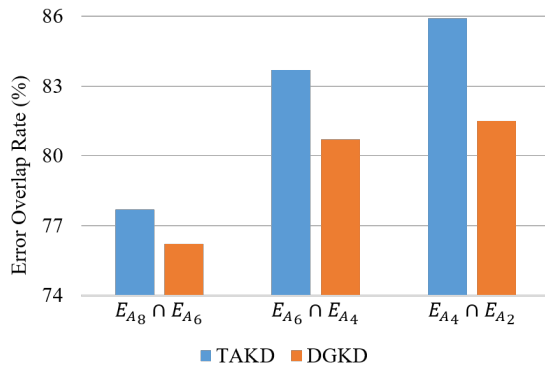


Figure 4. **Error avalanche problem.** The error overlap rate indicates the intersection proportion of a higher-level model’s incorrect answer and a lower level model’s incorrect answer when we have the teacher T_{10} , the student S_2 , and three TAs (e.g., A_8 , A_6 , and A_4). $E_i \cap E_j$ where E_i is the i th plain CNN model’s error examples on CIFAR-100.

task.

5.3. Error Avalanche Problem of TAKD

An example of an error avalanche problem is briefly explained in Figure 1 (c). In the case of TAKD, the student can learn from only an upper TA independently; the TA also learns from an upper TA sequentially following the distillation path. Thus, if an upper TA model transfers incorrect knowledge to the next model, this incorrect knowledge continuously accumulates following the sequential distillation path of TAKD.

We conducted an experiment to check the error overlap rate between two neighboring models using the full distillation path (e.g., $T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$), as shown in Figure 4, and observe that the error overlap rate of TAKD is much higher than that of DGKD in all cases. In particular,

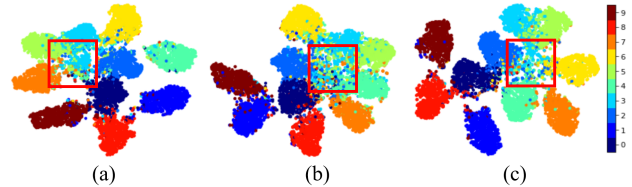


Figure 5. **t-SNE visualizations** of (a) KD for $T_{26} \rightarrow A_{20}$, (b) TAKD for $A_{20} \rightarrow S_{14}$, and (c) our DGKD for $A_{20} \rightarrow S_{14}$ using ResNet on CIFAR-10. Looking at the class distribution in the red box, we can see the different error accumulation rates of (b) TAKD and (c) our DGKD.

Table 5. Extensive distillation path with plain CNNs on CIFAR-100 dataset by adding TAs intermediately; n is the number of TAs used.

Step	n	TAKD	DGKD
Teacher (10 layer)	-	56.19	
Student (2 layer)	-	41.09	
$T_{10} \rightarrow A_6 \rightarrow S_2$	1	44.57	45.85
$T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$	3	45.14	48.92
$T_{10} \rightarrow A_9 \rightarrow A_8 \rightarrow A_7 \rightarrow A_6 \rightarrow A_5 \rightarrow A_4 \rightarrow A_3 \rightarrow S_2$	7	44.07	49.56

we can see that the closer to the student, the larger the gap in error overlap rates between TAKD and DGKD. From this viewpoint, we conclude that the teacher and TAs can help the student avoid the error avalanche problem by the proposed method.

We also verify this problem by t-SNE visualization at Figure 5, where it could be seen that the error accumulation increased more when going from TA to student networks than when going from teacher to TA networks. In this case, the student model deployed at the bottom would suffer from the accumulated errors due to the error avalanche problem. The student has a chance to learn from the cross-entropy supervision loss, but it is not very helpful to resolve the error avalanche problem because the 2 layer-based student’s learning capacity for the supervision signal is not enough to overcome this problem. Therefore, TAs sequentially deployed based on TAKD could be an insufficient solution when there is a large gap between the teacher and the student. Due to the fundamental limitations, it is difficult to expect the best performance improvement at KD. However, our DGKD teaches the student from the teacher to the TAs at the same time and this error avalanche problem could be alleviated properly.

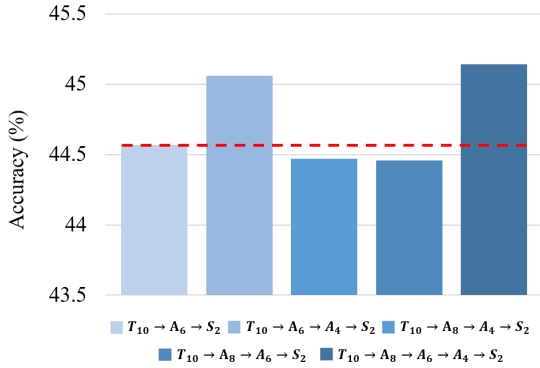


Figure 6. **Various TAKD distillation paths and the corresponding results.** Different distillation paths result in different accuracies at CIFAR-100, but the deeper path based on many TAs does not always guarantee the best performance owing to the error avalanche problem.

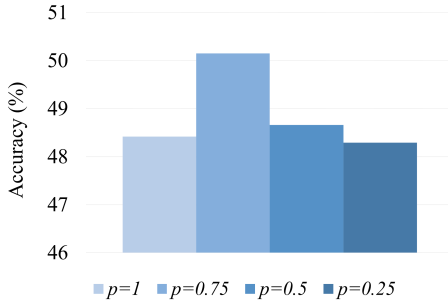


Figure 7. **Stochastic DGKD.** Performances of plain CNNs with $T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$ by the different survival probability p at CIFAR-100.

5.4. Knowledge Distillation Path

We also investigated the assertion made by [23] that a full distillation path is optimal for KD. As shown in Figure 6, the deepest model (e.g., $T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$) path is the best as described by [23]; however, the models with two TAs (e.g., $T_{10} \rightarrow A_8 \rightarrow A_4 \rightarrow S_2$ and $T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow S_2$) show lower performances than the model with one TA (e.g., $T_{10} \rightarrow A_6 \rightarrow S_2$) indicated by the red dash line. In this respect, we can infer that TAKD does not always improve the performance even if there are multiple TAs.

We attempted to determine whether TAKD can achieve the best accuracy even when the distillation path is extended to the maximum. As shown in Table 5, when all TAs are added intermediately for extending the distillation path to the maximum possible levels, the full distillation path model (e.g., $n = 7/44.07\%$) of TAKD shows an even lower performance than the other models (e.g., $n = 3/45.14\%$ and $n = 1/44.57\%$). In this respect, we could be sure that the

Table 6. Stochastic learning-based DGKD ($p = 0.75$) comparison results using a 2 layer-based plain CNN student with path $T_{10} \rightarrow A_8 \rightarrow A_6 \rightarrow A_4 \rightarrow S_2$ on CIFAR-100, and an 8 layer-based ResNet student with path $T_{26} \rightarrow A_{20} \rightarrow A_{14} \rightarrow S_8$ on CIFAR-10.

Model	Dataset	TAKD	Ours	
			DGKD	Stochastic DGKD
PlainCNN	CIFAR-100	45.14	48.92	50.15
ResNet	CIFAR-10	88.01	89.02	89.66

error avalanche problem would occur in this case. As a result, when the path is deeper and deeper, the error avalanche problem can get worse than expected. However, using the proposed DGKD, the accuracy gradually improves when the distillation path becomes deeper, ranging from $n = 1$ to $n = 7$; a full distillation path (e.g., $n = 7$) achieves the best accuracy of 49.56%. Note that as the path lengthens, the performance of our DGKD method improves unlike TAKD, and it was over approximately 5% better in accuracy compared with TAKD.

In summary, by adapting our proposed DGKD method which uses all higher-path model teacher and TAs together, a low capacity student can overcome the error avalanche problem with the proper trainers.

5.5. Stochastic DGKD

We proposed the stochastic learning-based DGKD for further performance improvements of a student. Specifically, our stochastic DGKD is directly inspired by the dropout concept. If there are n TAs, the student can learn from the $n + 1$ knowledge corpus including a teacher. We randomly removed the connections between the student and the trainers (the teacher and TAs) based on the survival probability p . To figure out the tendency of the survival probability, we performed the experiments shown in Figure 7. When the survival probability is $p = 0.75$, the student performance shows the best accuracy. Note that the stochastic DGKD with the survival probability p from 0.75 to 0.5 shows the better accuracy than the vanilla DGKD $p = 1$ but at $p = 0.25$, the accuracy is slightly below than the vanilla DGKD because we only have four knowledge connections in this experiment and dropping three results in losing the chance to learn the proper knowledge from the number of trainers. In this respect, we conclude that the survival probability $p = 0.75$ is the optimal hyper-parameter when we have three TAs and a teacher for the stochastic DGKD.

Table 6 shows a performance comparison among TAKD, DGKD, and stochastic DGKD. As expected, the stochastic DGKD shows the best performance on the other networks and datasets. Specifically, the proposed stochastic DGKD using the 2 layer-based plain CNN student network achieves 1.23% better result than the original DGKD and 5.01% bet-

Table 7. Comparison with well-known KD methods using ResNet on CIFAR-10. We used 26 layer-based ResNet as the teacher model which teaches two different student models, e.g., 8 layer-based ResNet and 14 layer-based ResNet, respectively. For TAKD and our DGKD, we used knowledge distillation paths, $T_{26} \rightarrow A_{20} \rightarrow A_{14} \rightarrow S_8$ and $T_{26} \rightarrow A_{20} \rightarrow S_{14}$, respectively.

	Student	KD [14]	FitNet [28]	AT [40]	FSP [36]	BSS [12]	Mutual [42]	TAKD [23]	Ours
ResNet8	86.02	86.66	86.73	86.86	87.07	87.32	87.71	88.01	89.66
ResNet14	89.11	89.75	89.82	89.84	89.92	90.34	90.54	91.23	92.34

Table 8. Comparison with the previously published KD methods using WRN, ResNet, and VGG on CIFAR-100. A bold number is a best accuracy, and an underlined number is a second best one.

Teacher	WRN40×2	ResNet56	VGG13
Student	WRN16×2	ResNet20	VGG8
Teacher	76.46	73.44	75.38
Student	73.64	69.63	70.68
KD [14]	74.92	70.66	72.98
FitNet [28]	75.75	71.60	73.54
AT [40]	75.28	<u>71.78</u>	73.62
SP [33]	75.34	71.48	73.44
VID [1]	74.79	71.71	73.96
RKD [24]	75.40	71.48	73.72
PKT [26]	76.01	71.44	73.37
AB [13]	68.89	71.49	74.27
FT [19]	75.15	71.52	73.42
CRD [32]	<u>76.04</u>	71.68	74.06
SSKD [35]	<u>76.04</u>	71.49	75.33
TAKD [23]	75.04	70.77	73.67
Ours	76.24	71.92	<u>74.40</u>

ter result than TAKD on the CIFAR-100. This improvement is the same on 8 layer-based ResNet student on CIFAR-10. From this, we can conclude that the proposed method works successfully in the case of a large gap between the teacher and student network.

5.6. Comparison with The State-of-the-art Methods

To verify the generality of the proposed method, we compared its performance with the well-known KD methods [14, 28, 40, 36, 12, 42, 23]. As summarized in Table 7, the proposed DGKD achieves the best performances compared with the well-known KD methods such as KD [14], FitNet [28], AT [13], FSP [36], BSS [12], Mutual [42], and TAKD [23] at both ResNet-based student models with 8 and 14 layers, respectively at CIFAR-10. For comparison results with various backbone architectures, for example, WRN [39], ResNet [11], and VGG [30], Table 8 shows that our proposed method performed favorably against the state-of-the-art KD methods. Specifically, a 40×2 layer-based WRN, a 56 layer-based ResNet, and a 13 layer-based VGG as teachers, and the corresponding students are a 16×2 layer-based WRN, a 20 layer-based ResNet, and an 8 layer-based VGG, respectively. In this experiment, we also use different numbers of the teacher assistant models for the

different networks. In detail, we used the following knowledge distillation paths: $T_{40 \times 2} \rightarrow A_{34 \times 2} \rightarrow A_{28 \times 2} \rightarrow A_{22 \times 2} \rightarrow S_{16 \times 2}$ for WRN, $T_{56} \rightarrow A_{44} \rightarrow A_{32} \rightarrow S_{20}$ for ResNet, and $T_{13} \rightarrow A_{11} \rightarrow S_8$ for VGG. As shown in Table 8, when the difference between the teacher and student networks is large (e.g., from WRN40-2 to WRN16-2 and from ResNet56 to ResNet20) our method shows the best accuracy among the state-of-the-art methods, but when the difference is not large (e.g., from VGG13 to VGG8), our method shows the second-best accuracy.

From the results of all these experiments, we can conclude that the proposed method shows the best performance not only when the gap between the teacher and the student is large but also compared with the general KD methods.

6. Conclusion

In this paper, we proposed a densely guided knowledge distillation using multiple assistants to improve the performance of the student with low capacity compared to the teacher. Empirically, we found that the error avalanche problem can easily occur as the assistant knowledge distillation paths deepen. When there are multiple assistants, if the upper assistant transfers the wrong answers to the next assistant and it continues recursively, it can be difficult for the student to avoid error avalanche problems because of its low capacity. Thus, we proposed the novel method using the knowledge of the teacher and the whole assistants to provide more opportunities for the student to learn the right answers during the training. Our experiments demonstrate that our proposed method can play a key role in resolving the error avalanche problem. Moreover, for efficient student learning, we adapted the stochastic learning role by randomly abandoning the teacher or assistant knowledge. Using this strategy, our proposed method achieves the state-of-the-art among the well-known distillation methods. We believe that our proposed method can boost planning the distillation path deeper and deeper using the multiple TAs, which improving the performances of the low capacity-based student networks in the real world applications.

Acknowledgement: This work was partially supported by IITP grant funded by the Korea government(MSIT) (No.2021-0-00951, Development of Cloud based Autonomous Driving AI learning Software), partially by IITP grant funded by MSIT (Artificial Intelligence Innovation Hub) under Grant 2021-0-02068, and partially by the BK21 FOUR Program under Grant NRF5199991014091.

References

- [1] S. Ahn, S. X., Hu, A. Damianous, N. D. Lawrence, and Z. Dai. Variational information distillation for knowledge transfer. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jun. 2019. 8
- [2] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, Dec. 2017. 1
- [3] J.-H. Bae, D. Yeo, J. Yim, N.-S. Kim, C.-S. Pyo, and J. Kim. Densely distilled flow-based knowledge transfer in teacher-student framework for image classification. *IEEE Trans. on Image Processing*, 29:5698–5710, Apr. 2020. 1
- [4] D. Chen, J. Mei, C. Wang, Y. Feng, and C. Chen. Online knowledge distillation with diverse peers. *34th AAAI Conf. on Artificial Intelligence*, Feb. 2020. 1, 2
- [5] J. Cho and B. Hariharan. On the efficacy of knowledge distillation. *IEEE International Conf. on Computer Vision*, pages 4794–4802, Oct. 2019. 2, 3
- [6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jun. 2009. 4, 5
- [7] T. Furlanello, Z. C. Lipton, M. Tschannen, L. Itti, and A. Anandkumar. Born-again neural networks. *International Conf. on Machine Learning*, Jul. 2018. 3
- [8] Q. Guo, X. Wang, Y. Wu, Z. Yu, D. Liang, X. Hu, and P. Luo. Online knowledge distillation via collaborative learning. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 4320–4328, Jun. 2020. 3
- [9] S. Han, H. Mao, and W. J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *International Conf. on Learning Representations*, May 2016. 1, 2
- [10] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. *IEEE International Conf. on Computer Vision*, Oct. 2017. 1
- [11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 770–778, Jun. 2016. 1, 5, 8
- [12] B. Heo, M. Lee, S. Yun, and J. Choi. Improving knowledge distillation with supporting adversarial samples. *33rd AAAI Conf. on Artificial Intelligence*, pages 3771–3778, Feb. 2019. 8
- [13] B. Heo, M. Lee, S. Yun, and J. Choi. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. *34th AAAI Conf. on Artificial Intelligence*, pages 3779–3787, Jan. 2019. 8
- [14] G. Hinton, O. Yinyals, and J. Dean. Distillation the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, Mar. 2015. 1, 2, 3, 5, 6, 8
- [15] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, Apr. 2017. 1
- [16] G. Huang, Z. Liu, L. van der Maaten, and K. Weinberger. Densely connected convolutional networks. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2261–2269, Jul. 2017. 1, 3
- [17] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger. Deep networks with stochastic depth. *European Conf. on Computer Vision*, Sept. 2016. 2, 4
- [18] X. Jin, B. Peng, Y. Wu, Y. Liu, J. Liu, D. Liang, J. Yan, and X. Hu. Knowledge distillation via route constrained optimization. *IEEE International Conf. on Computer Vision*, Oct. 2019. 2, 3
- [19] J. Kim, S. Park, and N. Kwak. Paraphrasing complex network: Network compression via factor transfer. *Advances in Neural Information Processing Systems*, pages 2760–2769, Dec. 2018. 8
- [20] A. Krizhevsky. Learning multiple layers of features from tiny images. *Tech. Rep*, Apr. 2009. 4
- [21] X. Lan, X. Zhu, and S. Gong. Knowledge distillation by on-the-fly native ensemble. *Advances in Neural Information Processing Systems*, pages 7528–7538, Dec. 2018. 5, 6
- [22] A. Malinin, B. Mlodozienec, and M. Gales. Ensemble distribution distillation. *International Conf. on Learning Representations*, Apr. 2020. 1
- [23] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh. Improved knowledge distillation via teacher assistant. *34th AAAI Conf. on Artificial Intelligence*, pages 5191–5198, Feb. 2020. 1, 2, 3, 5, 6, 7, 8
- [24] W. Park, D. Kim, Y. Lu, and M. Cho. Relational knowledge distillation. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jun. 2019. 8
- [25] A. Paszke, S. Gross, S. Chintala, and et. al. Automatic differentiation in pytorch. *NIPS Autodiff Workshop*, 2017. 5
- [26] D. Pathak, P. Kahenbuhl, J. Donahue, T. Darrell, and A. Efros. Context encoders: Feature learning by inpainting. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jun. 2016. 8
- [27] B. Peng, X. Jin, J. Liu, D. Li, Y. Wu, Y. Liu, S. Zhou, and Z. Zhang. Correlation congruence for knowledge distillation. *IEEE International Conf. on Computer Vision*, Oct. 2019. 5, 6
- [28] A. Romero, N. Ballas, S. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. *International Conf. on Learning Representations*, 2015. 1, 2, 8
- [29] Z. Shen, Z. He, and X. Xue. Meal: Multi-model ensemble via adversarial learning. *33th AAAI Conf. on Artificial Intelligence*, Feb. 2019. 2
- [30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5, 8
- [31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. 2, 4
- [32] Y. Tian, D. Krishnan, and P. Isola. Contrastive representation distillation. *International Conf. on Learning Representations*, Apr. 2020. 5, 6, 8
- [33] F. Tung and G. Mori. Similarity-preserving knowledge distillation. *IEEE International Conf. on Computer Vision*, pages 1365–1374, Oct. 2019. 1, 2, 5, 6, 8

- [34] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng. Quantized convolutional neural networks for mobile devices. *IEEE International Conf. on Computer Vision*, pages 4820–4828, Jun. 2016. [1](#)
- [35] G. Xu, Z. Liu, X. Li, and C. C. Loy. Knowledge distillation meets self-supervision. *European Conf. on Computer Vision*, Aug. 2020. [3](#), [5](#), [6](#), [8](#)
- [36] J. Yim, D. Joo, J. Bae, and J. Kim. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jul. 2017. [1](#), [2](#), [8](#)
- [37] R. Yu, A. Li, C. Chen, J. Lai, V. Morariu, X. Hand, M. Gao, C. Lin, and L. Davis. Nisp: Pruning networks using neuron importance score propagation. *IEEE International Conf. on Computer Vision*, Jun. 2018. [2](#)
- [38] L. Yuan, F. E. Tay, G. Li, T. Wang, and J. Feng. Revisit knowledge distillation: a teacher-free framework. *IEEE Conf. on Computer Vision and Pattern Recognition*, Jun. 2020. [1](#), [3](#)
- [39] S. Zagoruyko and N. Komodakis. Wide residual networks. *British Machine Vision Conference*, pages 87.1–87.12, Sept. 2016. [5](#), [8](#)
- [40] S. Zagoruyko and N. Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *International Conf. on Learning Representations*, May 2017. [1](#), [2](#), [5](#), [6](#), [8](#)
- [41] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. *IEEE International Conf. on Computer Vision*, pages 3713–3722, Oct. 2019. [1](#), [2](#)
- [42] Y. Zhang, T. Xiang, T. Hospedales, and H. Lu. Deep mutual learning. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 4320–4328, Jun. 2018. [1](#), [2](#), [8](#)