Technical Report: Data-Free Knowledge Distillation

Junhee Heo, Jinbae Im, Hyunsouk Cho

***Abstract*—Knowledge distillation (KD) is in the limelight for recent years among model compression and knowledge transfer methods. Similar with many real-world applications, KD also struggles with the data-sparse problems from the privacy, cost, and legal issue. To solve this problem, many data-free knowledge distillation (DF-KD) approaches are also proposed. In this paper, we categorized the intuition of the previous DF-KD models by analyzing their loss functions. By empirical testing, we found that the effect of the entropy of teacher model is crucial in the DF-KD training. From this observation, we propose a novel loss function based on Minkowski distance which can control the impact of the teacher network's entropy. From the real-data experiments, we demonstrate the proposed model outperforms the-state-of-the-art baseline.**

I. Introduction

Knowledge distillation (KD) is in the limelight for recent years among model compression and knowledge transfer methods. Knowledge Distillation is expected to be used on edge devices like mobile phones which have small computational capacities. The main idea is to use the outputs of a large pre-trained network(teacher network) to train another smaller network (student network). To overcome the smaller network, the student network trains from both the real data and the knowledge from the teacher. Therefore, student loss contains not just the difference between student's output and real label but also the difference between student's output and teacher's output, as below:

$$L_{total} = L_{error} + L_{distillation}$$

where L_{error} represents the loss with the real label and $L_{distillation}$ represents the loss with the knowledge from the teacher.

To distill the knowledge from the teacher network into the student network effectively, Hinton et al. [1] introduced knowledge distillation paradigm as a transferring information method. From the technique, many studies [2]–[4] have been conducted and shown good performance.

Previous approaches have used the original training data that was used for training the teacher network. However, it is usually hard to get training data due to many real-world problems (e.g. privacy, cost, and legal issue). Because of this sparse-data problem, models outside of the lab often need to learn in a data-free condition. In these situations, when model compression of a pre-trained model is needed for resource-limited platforms like mobiles, the original training data could be restricted.

To overcome this issue, Srinivas and Babu [5] merged similar neurons directly in fully connected layers and Lopes et al. [6] used meta-data to reconstruct the training samples. However, the former's approach was impossible to apply in convolutional layers and the latter's approach was not completely data-free. To overcome those limitations, researchers started to synthesize data whose distribution is similar to that of the original training data by adopting GAN concepts [7].

To adversarially learn a generator that mimics the original training data distribution while learning the student network, many researchers [8]–[11] have used student network as discriminator. They trained the student network so that the logits of the teacher and student were similar for the data examples generated by the generator. Adversarially, they trained the generator network so that the logits of the teacher and student were different for the data examples generated by the generator. By adversarially learning with each other, the students network learns the knowledge of the teacher network and the generator generates examples that follow a distribution similar to that of the original training data.

Most researchers have focused on the loss function representing the difference between the distributions of the teacher network and student network. Furthermore, some researchers used additional loss function for the generator network. However, the studies were conducted unorganized, and some studies also showed inconsistent results.

In this paper, we organized the recent data-free knowledge distillation studies and the loss terms used in those studies based on key concepts. Through the experiments, we found that the change in the entropy of the teacher network for the generated examples was crucial. From the findings, we propose a loss function based on Minkowski distance. With the loss function, we can freely control the impact of the teacher network's entropy. We have demonstrated the excellence of our proposed loss function on the CIFAR-10 dataset.

II. Related Work

In this section, we briefly introduce the concept of GAN-based Data-Free Knowledge Distillation (DF-KD) studies, and the intuition of the previous models by analyzing their loss functions.

The goal of GAN in DF-KD is different from the basic GAN. In the basic GAN, the generator usually aims to synthesize data that can fool the discriminator. In DF-KD, the generator aims to disturb the training of the discriminator (the student model). To hinder the student model, the generator will generate data which maximize the gap of the knowledge between the student model and the teacher model. On the contrary, the student model struggles to follow the teacher model with the generated data.

TABLE I: Loss Taxonomy of Knowledge Distillation

	Ground-truth loss	Distillation loss	Additional loss		
			Valid features	Instance entropy	Batch entropy
[1]	Cross-entropy	Cross-entropy	-	-	-
[8]	-	Cross-entropy	activation	one-hot	batch class
[9]	-	KL-divergence	-	-	-
[11]	-	KL-divergence	batch norm.	instance categorical	batch class
[10]	-	MAE	-	-	-

A. Loss of Student (Discriminator)

As it is impossible to use true labels in the data free setting, the loss function of the student network only contains the distillation loss as follows:

$$L_{student} = L_{distillation}$$

In DF-KD studies, $L_{distillation}$ is simply the difference between the output class probabilities of the teacher network and the student network. Many studies have been conducted to find the optimal difference measure.

1) Kullback-Leibler(KL) Divergence: KL divergence has been actively used for DF-KD because it is a representative measure of the difference between the two distributions [9], [11]. KL divergence from student distribution S to teacher distribution T is decomposed as follows:

$$D_{KL}(T||S) = H(T(x_G), S(x_G)) - H(T(x_G)),$$

where $H(p, q)$ stands for cross entropy of the distribution q relative to a distribution p and $H(p)$ is the entropy of the distribution p. x_G represents examples generated by generator G. As $H(T(x_G))$ does not affect the learning of the student network, KL divergence simply plays a role in reducing the cross entropy between $T(x_G)$ and $S(x_G)$.

2) Mean Absolute Error (MAE): [10] uses Mean Absolute Error (MAE) for the difference measure. MAE is a measure of errors between paired observations in statistics. In machine learning, it has been mainly used as a loss function for prediction problems. [10] proved the superiority of MAE theoretically and experimentally in DF-KD, and especially showed that MAE performs better than KL divergence. The loss function using MAE is as the following:

$$MAE(T, S) = \frac{1}{n}\|T(x_G) - S(x_G)\|,$$

where n is the number of examples generated by G.

B. Loss of Generator

In adversarial training, the role of the generator is to interrupt the student. Therefore, generator loss is simply expressed as minus student loss. However, unlike the discriminator, learning of the generator is more difficult and unstable. Additional loss terms have been used to compensate for the generator loss. The loss function for generator in DF-KD is expressed as follows:

$$L_{generator} = -L_{student} + \alpha \cdot L_{others},$$

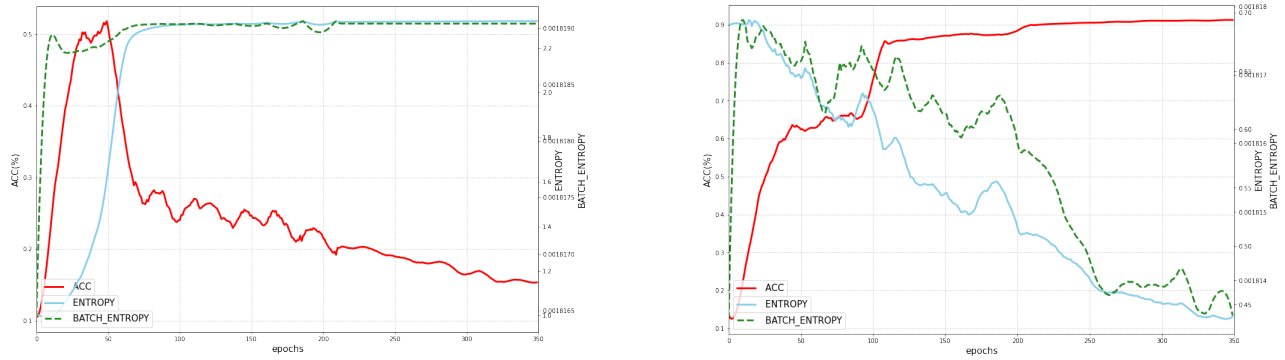
where L_{others} is collectively referred to as additional loss terms. Various additional loss terms have been used, and we have summarized them into three concepts.

1) Valid features: The generator is prone to generating meaningless examples particularly early in learning. To compensate for this, additional loss terms that make examples generated from generator have valid features in the teacher network have been proposed.

- **Activation loss function:** Chen et al. [8] proposed activation loss function.
  $$L_a = -\frac{1}{n}\sum_i \|f_T^i\|_1$$
  f_T^i is the output before the fully-connected layer of the teacher network. The intuition contained in the loss function is that the feature maps of teacher model would have higher activation value when the input is close to real data because the teacher model was trained to extract innate patterns of real data.
- **Batch normalization statistics:** Choi et al. [11] proposed the loss function using batch normalization statistics.
  $$L_{bn} = \sum_{l,c} D_{KL}((\hat{\mu_G}(l,c), \hat{\sigma_G^2}(l,c)), (\mu(l,c), \sigma^2(l,c)))$$
  $\mu(l,c), \sigma^2(l,c)$ is the mean and variance of l-th layer for channel c stored in the teacher network. $\hat{\mu_G}(l,c), \hat{\sigma_G^2}(l,c)$ is the empirical statistics of the teacher network from the data synthesized by generator G. Since the stored batch normalization statistics were obtained from real training data, comparing batch normalization statistics could be helpful to the generator to produce real-like data. This loss term has a very strong effect, however it is difficult to say that the loss is truly data-free because the loss is based on meta information. Furthermore, it has a disadvantage that it cannot be used in a model without batch normalization.

2) Decreasing instance class entropy: Additional loss terms have been proposed to reduce the entropy of the output probability of the teacher network for generated examples. The intuition behind this is that the entropy of the teacher network should be small if generated examples are meaningful, because original teacher network was trained from hard label with minimum entropy.

- **One-hot loss:** Chen et al. [8] proposed one-hot loss.
  $$L_{oh} = \frac{1}{n}\sum_i H(T(x_G), \text{argmax}(T(x_G)))$$
  This loss encourages the output probability of the teacher network from synthetic data to be close to one-hot vector.



(a) Cross-entropy only

(b) Cross-entropy + high entropy

Fig. 1: Accuracy changes of the Cross-entropy based loss model

This has a similar meaning to minimizing the entropy of the teacher network from synthetic data.

- **Instance categorical entropy loss:** Choi et al. [11] proposed instance categorical entropy loss.

$$L_{ce} = \frac{1}{n}\sum_i H(T(x_G))$$

This loss simply decreases the instance categorical entropy of the teacher network with generated examples.

3) Increasing batch class entropy: Additional loss terms have also been proposed to increase the batch entropy of the output probability of the teacher network for generated examples. As the original teacher network was trained from examples with various classes, the entropy value obtained from batch examples should be sufficiently large if examples generated by generator follow training data distribution.

- **Batch class entropy loss:** Both [8], [11] used batch class entropy loss in common.

$$L_{bce} = -H(E(T(x_G)))$$

This loss increases batch class entropy to make generated data to have a uniformly distributed category.

III. Method

A. Entropy in data-free KD

We hypothesize that the entropy of the teacher network would play an important role in data-free knowledge distillation. Most of recent studies use the $KL(T\|S)$ and it is equivalent to $H(T,S) - H(T)$. We observed the change in distillation performance by adjusting the weight of $H(T)$ in two ways.

First, we set the weight of $H(T)$ as 0. It is equivalent to use the cross entropy loss as the distillation loss. When the entropy of the teacher network was removed, it diverged and the generator could not generate useful examples for training student networks. Fig. 1a shows the accuracy changes when a cross-entropy loss is the only loss. As the loss function that adjusts the entropy for the examples generated by the generator does not exists, the entropy diverges and the accuracy has declined.

Second, we set the weight of $H(T)$ as large. The loss was $H(T,S) - 1.5H(T)$. As intended, the entropy of the teacher network was greatly reduced. However, as a side effect of this, the diversity of examples generated by the generator has decreased. In other words, as the individual instance entropy in the batch was greatly reduced, the entropy of the entire batch was also greatly reduced. As shown in Fig. 1b, the individual instance entropy was greatly reduced as intended, however the entropy of the entire batch was also greatly reduced. It did not fully reflect the distribution of the original training data and caused the limited performance.

From the experiments, we conclude that both instance entropy and batch entropy of the teacher network is important for data-free KD. This finding corresponds to the use of additional loss term in addition to the KL divergence in previous studies.

B. MAE in data-free KD

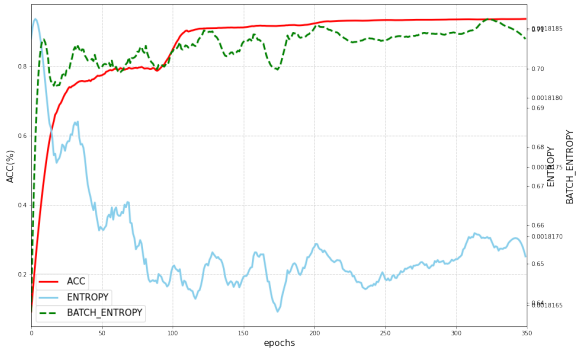


Fig. 2: Accuracy changes of the MAE loss model

In the very recent study [10], MAE was used for the distillation loss on behalf of the KL divergence. They theoretically and experimentally demonstrated that MAE was superior to KL divergence. Even MAE alone showed good performance without additional loss to the generator.

We hypothesize that the success of the MAE would be related to the entropy of the teacher network. We observed the changes in the instance entropy and batch entropy of the teacher network in the distillation process using MAE. As shown in Fig. 2, the instance entropy gradually decreased and the batch entropy gradually increased, as the accuracy

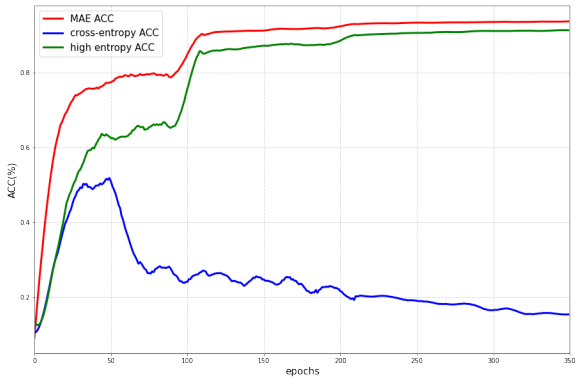


Fig. 3: Accuracy comparison of baseline models

of the student network gradually increased. In conclusion, the MAE effectively performs the knowledge distillation by controlling instance entropy and batch entropy. MAE based model showed better performance compared to the Cross-entropy based models as shown in Fig. 3.

C. Proposed Method: Minkowski Distance

In the previous section, it was observed that the change in entropy directly affects the performance of the data-free knowledge distillation. In order to perform knowledge distillation more effectively by controlling the entropy, we focus on the Minkowski distance. Minkowski distance is a metric in a normed vector space which can be considered as a generalization of many distance metrics. When distance order p is 1, the Minkowski distance is the same as Manhattan distance that is equal to MAE times the number of class labels, and it is equivalent to the Chebyshev distance In the limiting case of p reaching infinity.

$$Minkowski(x, y) = (\sum_{i=1}^{n} |x_i - y_i|^p)^{1/p}$$

We hypothesize that an optimal p that can better control entropy and perform the knowledge distillation more effectively exists. In particular, when p is set large, the entropy of the teacher network is reduced and it would help early in learning when learning is unstable. Therefore, we set the Minkowski distance as the distillation loss. More specifically, in order to generalize MAE, we use the loss obtained from dividing the Minkowski by the number of class labels C as our loss function.

$$\textit{Minkowski loss} = \frac{(\sum_{i=1}^{n} |T(x_G) - S(x_G)|^p)^{1/p}}{C}$$

IV. Experiments

A. Experimental Settings

We used CIFAR10 as our dataset. It contains 60,000 RGB images (50,000 for training and 10,000 for testing) of 10 classes. For this experiment, only the test set is used to get accuracy of student model. ResNet-34 with accuracy of 95.54 was our teacher model and ResNet-18 was our student model.

The batch size was 128 for every experiment. For student, SGD of momentum 0.9 and weight decay 5e-4 was used and for generator, and Adam was used. To make student model stable and ensure convergence, we updated student model for 5 times during one epoch iteration. Trained models for 500 epochs and 50 iterations.

To find the optimal distance order p of the Minkowski loss, we set p as 1.1, 1.5 and 2.0. For each p, equal environment and training parameter was used.

B. Results

Table II shows the accuracy of the student networks with Minkowski loss function for various p. When $p = 1.0$, the accuracy was similar to that of MAE loss function as we intended. The accuracy recorded the best performance at $p = 1.5$. This proved the assumption that an optimal p for improving the distillation performance would exists. When p was outside of the range $1.0 \leq p \leq 2.0$, the accuracy decreased rapidly because it constrained the teacher entropy for generated examples too much.

TABLE II: Accuracy of Students with Minkowski Losses

Model	Accuracy(%)
Teacher	95.54
MAE	94.08
$p = 1.0$	94.07
$p = 1.1$	94.19
$p = 1.5$	94.25
$p = 2.0$	93.95

Fig. 4 shows the change pattern of the accuracy of the student network and the instance entropy of the teacher network with generated examples as the distillation progresses. As we intended, the instance entropy was controlled by adjusting the distance order p of the Minkowski loss function. This had a big impact at the beginning of learning. When p was greater than 2.0, the instance entropy became too small, and it showed lower performance than the baseline MAE.

V. Conclusion

In this study, we organized previous GAN-based data-free knowledge distillation studies focusing on the loss function and the intuition contained in it. Through experiments, we found that the pattern of changes in the entropy of the teacher network has a profound effect on the distillation performance. Furthermore, we also found that the loss function using MAE shows an ideal entropy change pattern when used alone without any additional loss term. From the findings, we propose a Minkowski distance-based loss function that generalizes the MAE loss function. With the loss function, the entropy of the teacher network can be freely controlled by adjusting the distance order p of the Minkowski distance. For further research, we will adopt curriculum learning to DF-KD by adjusting p of the Minkowski distance.

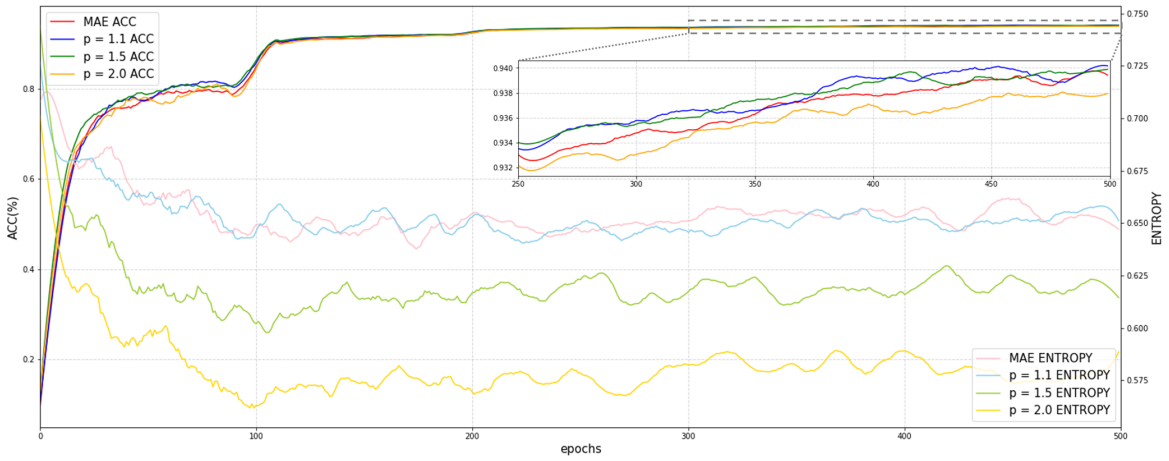


Fig. 4: Accuracy comparison of various Minkowski loss models

REFERENCES

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

[2] J. Tang, R. Shivanna, Z. Zhao, D. Lin, A. Singh, E. H. Chi, and S. Jain, "Understanding and improving knowledge distillation," *arXiv preprint arXiv:2002.03532*, 2020.

[3] J. Yim, D. Joo, J. Bae, and J. Kim, "A gift from knowledge distillation: Fast optimization, network minimization and transfer learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4133–4141.

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

[5] S. Srinivas and R. V. Babu, "Data-free parameter pruning for deep neural networks," *arXiv preprint arXiv:1507.06149*, 2015.

[6] R. G. Lopes, S. Fenu, and T. Starner, "Data-free knowledge distillation for deep neural networks," *arXiv preprint arXiv:1710.07535*, 2017.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[8] H. Chen, Y. Wang, C. Xu, Z. Yang, C. Liu, B. Shi, C. Xu, C. Xu, and Q. Tian, "Data-free learning of student networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3514–3522.

[9] P. Micaelli and A. J. Storkey, "Zero-shot knowledge transfer via adversarial belief matching," in *Advances in Neural Information Processing Systems*, 2019, pp. 9551–9561.

[10] G. Fang, J. Song, C. Shen, X. Wang, D. Chen, and M. Song, "Data-free adversarial distillation," *arXiv preprint arXiv:1912.11006*, 2019.

[11] Y. Choi, J. Choi, M. El-Khamy, and J. Lee, "Data-free network quantization with adversarial knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 710–711.