2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Knowledge Distillation Based on Logit Ranking Alignment

Guoming Lu[1,2][0000-0001-7477-5800], Mingdong Zhang[1][0009-0000-0298-9754], Zihan Cheng[1,3][0009-0007-1689-6816], Jielei Wang[1,2][0000-0003-2882-7053](✉), Yu Peng[1,2][0000-0002-2949-9728], Kexin Li[1,2][0000-0003-3511-2582]

[1] The Institute of Intelligent Computing, University of Electronic Science and Technology of China (UESTC), Chengdu, 611731, China
[2] Ubiquitous Intelligence and Trusted Services Key Laboratory of Sichuan Province, Chengdu, 610000, China
[3] School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu, 610054, China
lugm@uestc.edu.cn, mdzhang445@foxmail.com, zihancheng1213@foxmail.com, jieleiwang_uestc@163.com, ypeng@uestc.edu.cn, likx@uestc.edu.cn

Abstract. With the development of deep learning, models have become increasingly large and difficult to deploy effectively on embedded devices, mobile applications, and edge computing environments. Knowledge distillation has become a mainstream approach to solving this problem due to its simplicity and efficiency. In the field of image classification, most knowledge distillation methods focus on how to enable the student model to learn more knowledge from the teacher model, but they introduce unnecessary strict constraints. To address this challenge, we propose a novel knowledge distillation paradigm based on logit ranking alignment, i.e., aligning the logit rankings of the teacher and student models. Since traditional hard ranking algorithm is non-differentiable, we introduce a fast differentiable soft ranking algorithm to obtain the soft logit rankings of the teacher and student models, and then we use an L2 loss to align them. Extensive experiments on CIFAR-100 and Tiny-ImageNet validate the effectiveness of our method.

Keywords: Knowledge Distillation, Logit Ranking Alignment, Fast Differentiable Soft Ranking, Image Classification.

1 Introduction

With the rise of machine learning, an increasing number of models have been developed for computer vision tasks, such as image classification [1, 2, 3], object detection [4, 5], and semantic segmentation [6, 7]. However, with the continuous development of deep learning, models have become larger, making it difficult to deploy them effectively on embedded devices, mobile applications, and edge computing environments. Knowledge distillation [8] has become a mainstream approach to address this issue. Knowledge distillation can transfer the knowledge from large models to small ones, thereby reducing the resource consumption required for model deployment.

With the growing research interest in knowledge distillation, an expanding array of distillation approaches have been developed for image classification tasks. From the perspective of knowledge definition, knowledge distillation methods can be broadly classified into three categories: logit distillation [8-17], feature distillation [18-25], and relation distillation [26-28]. Logit distillation has become a research hotspot in recent years due to its simplicity, efficiency, and low cost.

Traditional KD [8] requires the student model to match the probability distribution output by the teacher model. Logit standardization [13] uses a standardization method to eliminate the impact of different means and variances in logits. MLKD [15] employs prediction augmentation and multi-level alignment to allow the student model to learn more knowledge from the teacher model's logits. However, a common goal of these methods is to require the student to learn the probability distribution or certain latent characteristics of the teacher's logits, which introduces unnecessary strict constraints in the learning process of the student model.

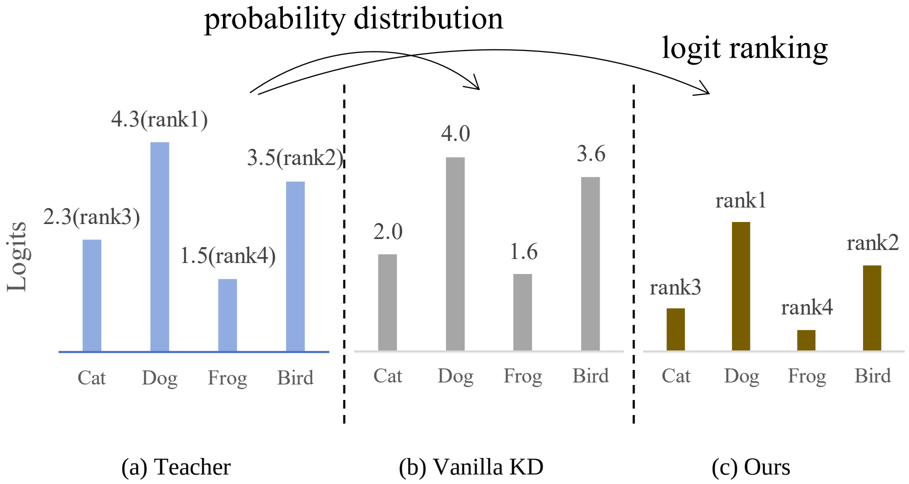


Fig. 1. Vanilla KD explicitly requires the student model to learn the probability distribution output by the teacher model, introducing unnecessary strict constraints. However, our method simply requires the student model to learn the logit ranking of the teacher model, reducing such constraints.

To address this challenge, we propose a novel knowledge distillation paradigm based on logit ranking alignment. As shown in Fig. 1, we simply require the student model to learn the logit ranking of the teacher model without forcing it to learn the probability distribution output by the teacher model. This allows the student to adaptively achieve the learning objective, reducing the learning difficulty and mitigating the impact of the capability gap between teacher and student models.

However, hard ranking is inherently non-differentiable. To overcome this, we introduce a fast differentiable soft ranking algorithm to achieve our goal. This method employs permutahedron projection, which transforms the discrete ranking problem into a continuous optimization problem, making the ranking process differentiable. We then directly align the soft logit rankings of the teacher and student using an L2 loss. Extensive experiments on standard benchmarks demonstrate that our method outperforms previous logit distillation and feature distillation methods, proving its effectiveness.

In summary, this paper makes the following three contributions:

- To avoid introducing unnecessary strict constraints to the student model, this paper proposes a novel knowledge distillation paradigm based on logit ranking alignment.
- To address the non-differentiability issue of traditional hard ranking algorithm, this paper introduces a fast differentiable soft ranking algorithm to obtain the soft logit rankings of the teacher and student, and uses an L2 loss to align them.
- Extensive experiments on CIFAR-100 and Tiny-ImageNet validate the effectiveness of our method.

2 Related Work

In the field of image classification, knowledge distillation is widely used to improve the performance of lightweight models. Traditional knowledge distillation [8] enables the student to learn the probability distribution output by the teacher by introducing soft labels and KL divergence loss. However, this method has several issues, including a large impact from the temperature parameter, difficulty in cross-architecture distillation, and a lack of adaptability in the student model. To address these issues, more and more knowledge distillation methods have been proposed, which can be mainly categorized into three types: logit distillation [8-17], feature distillation [18-25], and relation distillation [26-28].

Feature Distillation. These methods improve the student's understanding of the teacher by making the student learn the intermediate features of the teacher. In recent years, many related methods have been proposed. ReviewKD [23] proposed cross-stage connection paths in knowledge distillation for the first time; CAT-KD [22] improved the student model by transferring the teacher's class activation maps; FCFD [25] explicitly optimized the functional similarity between teacher and student features. Other representative works include [18-21], [24]. However, in general, feature distillation is heavily affected by the teacher-student architecture and has high computational costs.

Relation Distillation. These methods enhance the generalization ability of the student by making the student learn the relational information of the teacher. SP [26] enabled input pairs that produce similar (or dissimilar) activations in the teacher network also produce similar (or dissimilar) activations in the student network. ICKD [27] aligned the diversity and homology of the feature space of the student with that of the teacher. DIST [28] proposed a correlation-based loss to explicitly capture the teacher model's inter-class and intra-class relations. However, in general, relation distillation is computationally intensive and poses optimization challenges due to its dependence on structured inter-sample relationships.

Logit Distillation. These methods mainly allow the student to learn information related to the teacher’s logits, with traditional KD [8] belonging to this category. In recent years, many logit distillation methods have been proposed. TAKD [9] proposed multi-step knowledge extraction by employing a medium-scale network (teacher assistant) to bridge the gap between the student and teacher; DKD [12] rephrased the classic KD loss into two parts: target class knowledge distillation (TCKD) and non-target class knowledge distillation (NCKD); CTKD [10] dynamically modulated task difficulty throughout the student's learning process via a learnable temperature parameter; TTM [14] improved the generalization ability of the student by reformulating temperature scaling as a power transform of probability distribution. However, these methods still introduce unnecessary strict constraints in the learning process of the student. In contrast, a significant advantage of our method is that it reduces the difficulty of learning, allowing the student to adaptively achieve the learning objective.

3 Methodology

3.1 Preliminaries

Traditional KD [8] mainly relies on soft labels for training. Assuming there are C classification categories, let s represent the student's logits and t represent the teacher's logits, where both s and t belong to R^C. The softmax outputs of the student and teacher are then represented as:

$$p_{Stu}{}^{(i,T)} = \frac{e^{s_i/T}}{\sum_{j=1}^{C} e^{s_j/T}}, p_{\mathrm{Tea}}{}^{(i,T)} = \frac{e^{t_i/T}}{\sum_{j=1}^{C} e^{t_j/T}} \tag{1}$$

where s_i and t_i represent the values of the student's and teacher's logits for the i-th class, respectively. T is the temperature parameter used to control the smoothness. The larger the value of T, the smoother the softmax distribution; the smaller the value of T, the sharper the softmax distribution. Let y represent the one-hot encoded true label corresponding to s. The student's standard cross-entropy loss is then expressed as:

$$L_{CE} = -\sum_{i=1}^{C} y_i \, log \, p_{Stu}{}^{(i,1)} \tag{2}$$

where y_i represents the value of y for class i (either 0 or 1). The KL divergence loss between the student and teacher models can be expressed as:

$$L_{KL} = \sum_{i=1}^{C} p_{Tea}{}^{(i,T)} \, log \frac{p_{Tea}{}^{(i,T)}}{p_{Stu}{}^{(i,T)}} \tag{3}$$

Finally, the traditional KD loss combines the CE Loss and KL Loss, and can be expressed as:

$$L_{KD} = \alpha \cdot L_{CE} + \beta \cdot L_{KL} \tag{4}$$

where α and β are hyperparameters that control the weights of the CE Loss and KL Loss. Traditional KD can improve the generalization ability of the student model, and is easy to implement. However, it suffers from issues such as a large dependency on the temperature parameter, difficulty in cross-architecture distillation, and a lack of adaptability in the student model. Therefore, we propose a more robust logit distillation method.

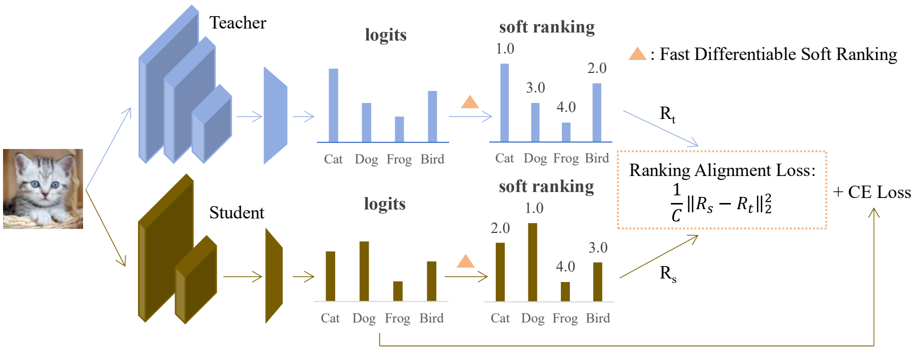


Fig. 2. Overall Framework. The image is processed through both the teacher and student models to produce logits. Then, through the fast differentiable soft ranking algorithm, the soft logit rankings R_t and R_s for the teacher and student are obtained. Finally, an L2 loss is used to align R_t and R_s, resulting in the Ranking Alignment Loss, which, along with the student's CE Loss, forms the final loss.

3.2 Soft Ranking Alignment

We propose a novel knowledge distillation paradigm based on logit ranking alignment, as shown in Fig. 2. First, the image is processed through both the teacher and student models to obtain logits. Then, the fast differentiable soft ranking algorithm is applied to obtain the soft logit ranking. Finally, an L2 loss is used to align the teacher's and student's soft logit rankings, resulting in the Ranking Alignment Loss, which, along with the student's CE Loss, forms the final loss.

Fast Differentiable Soft Ranking. As shown in Fig. 2, we need to generate corresponding ranking based on the logits of the teacher and student. Since traditional hard ranking algorithm is not differentiable, we adopt the fast differentiable soft ranking algorithm proposed in [29].

Let $\rho = (n, n-1, \cdots, 1)$. Let Σ represent the set of $n!$ permutations of ρ, and let $\theta \in R^n$ be the input to be ranked. First, present the discrete optimization criterion:

$$r(\theta) = \underset{\mu \in \Sigma}{argmax} \langle -\theta, \mu \rangle \tag{5}$$

where the symbol $\langle \cdot\,, \ \cdot \rangle$ represents the dot product. To make the problem continuously optimizable, this algorithm introduces the permutahedron induced by the vector $w \in R^n$, which is the convex hull formed by all permutations of w:

$$P(w) = conv(\{w_\sigma : \sigma \in \Sigma\}) \subset R^n \tag{6}$$

The permutahedron of w is a convex polyhedron whose vertices correspond to all permutations of w. Specifically, when $w = \rho$, $P(w) = conv(\Sigma)$. Based on this, the linear programming formulation for ranking can be derived:

$$r(\theta) = \underset{\mu \in P(\rho)}{argmax} \langle -\theta, \mu \rangle \tag{7}$$

To design an efficient approximation of the ranking operator and provide useful derivatives for the ranking operator, this algorithm introduces strong convex regularization in the linear programming formulation. Define $z \in R^n$, then, add the squared regularization $Q(\mu) = \frac{1}{2} \|\mu\|^2$ to the linear programming on the permutahedron:

$$P_Q(z, w) = \underset{\mu \in P(w)}{argmax} \langle z, \mu \rangle - Q(u) = \underset{\mu \in P(w)}{argmin} \frac{1}{2} \|\mu - z\|^2 \tag{8}$$

i.e., the Euclidean projection of z onto $P(w)$. To control the strength of the regularization, this algorithm introduces a parameter $\varepsilon > 0$ and multiply it by Q (equivalent to dividing z by), i.e.:

$$P_{\varepsilon Q}(z, w) = P_Q(z/\varepsilon, w) \tag{9}$$

Based on this, define the Q-regularized soft ranking as:

$$r_{\varepsilon Q}(\theta) = P_{\varepsilon Q}(-\theta, \rho) = P_Q(-\theta/\varepsilon, \rho) \tag{10}$$

As $\varepsilon \to 0$, $r_{\varepsilon Q}(\theta)$ converges to the corresponding "hard" counterpart; as $\varepsilon \to \infty$, $r_{\varepsilon Q}(\theta)$ shrinks to a constant. It is important to note that the resulting $r_{\varepsilon Q}(\theta)$ corresponds to a descending order ranking, meaning that the larger the value in θ, the smaller the corresponding ranking in $r_{\varepsilon Q}(\theta)$. To obtain an ascending order ranking, simply input $-\theta$, i.e., $r_{\varepsilon Q}(-\theta)$. Additionally, to speed up the process, [29] also provides methods for fast computation and differentiation.

Ranking Alignment. Given the logits s and $t \in R^C$ for the student and teacher models, we obtain the corresponding soft rankings $r_{\varepsilon Q}(s)$ and $r_{\varepsilon Q}(t)$ through the fast differentiable soft ranking algorithm. Then, we use an L2 loss to align $r_{\varepsilon Q}(s)$ and $r_{\varepsilon Q}(t)$ to obtain the Ranking Alignment Loss:

$$L_{RA} = \frac{1}{C} \cdot \sum_{i=1}^{C} \left(r_{\varepsilon Q}(s)_i - r_{\varepsilon Q}(t)_i \right)^2 \tag{11}$$

3.3 Total Loss

In summary, the total loss function can be expressed as:

$$L_{\text{total}} = L_{CE} + \lambda L_{RA} \tag{12}$$

where λ is a hyperparameter used to adjust the weight of the Ranking Alignment Loss. For convenience, in all subsequent experiments, the regularization strength parameter ε for the fast differentiable soft ranking is fixed to 1.

4 Experiments

In Section 4.1, we introduce the datasets and related settings used in this paper. In Section 4.2, we discuss the experimental results on CIFAR-100 and Tiny-ImageNet. In Section 4.3, we conduct ablation study, hyperparameter experiments, and visualization analysis.

4.1 Datasets and Settings

Datasets. We use the widely adopted CIFAR-100 and Tiny-ImageNet datasets to validate the effectiveness of our method. 1) CIFAR-100 [30] is a standard dataset for image classification, containing 100 categories, with 500 training images and 100 test images per category. Each image has a resolution of 32 × 32. 2) Tiny-ImageNet [31] is a smaller version of the ImageNet dataset, consisting of 200 categories, with 500 training images and 50 validation images per category. Each image has a resolution of 64 × 64.

Settings. We compare our method with other approaches on both homogeneous and heterogeneous teacher-student architectures. We use a variety of network architectures, including ResNet [32], WRN [33], VGG [34], MobileNetV2 [35], and ShuffleNet-V1 [36]/V2 [37].

Implementation Details. For CIFAR-100, we use a batch size of 64, an initial learning rate of 0.025, and train for 300 epochs. The learning rate decays by a factor of 0.1 at the 150th, 180th, and 210th epoch. We use SGD as the optimizer with a weight decay of 0.0005 and a momentum of 0.9. For Tiny-ImageNet, we use a batch size of 128 and an initial learning rate of 0.05, training for 240 epochs, while preserving other configurations. To ensure fair comparisons, we apply the identical data augmentation strategies as MLKD [15] on CIFAR-100. All experiments are conducted on a 1× NVIDIA Tesla T4 16GiB GPU.

4.2 Experimental Results

To validate the effectiveness of our method, we conduct extensive experiments on CIFAR-100 and Tiny-ImageNet, and compare it with logit distillation and feature distillation methods.

CIFAR-100. We compare our method with various logit distillation and feature distillation methods on CIFAR-100. The comparison results for homogeneous teacher-student architectures are reported in Table 1, while the results for heterogeneous teacher-student architectures are reported in Table 2. It can be observed that our method achieves the best performance in both homogeneous and heterogeneous teacher-student architectures, validating the superiority of our approach.

Tiny-ImageNet. We compare our method with other methods on Tiny-ImageNet, and the results of Top-1 accuracy are shown in Table 3. It can be observed that, compared to previous logit distillation and feature distillation methods, our method achieves the best performance, validating the effectiveness of our approach.

4.3 Analyses

The experimental results above validate the superiority and effectiveness of our method. In this section, we will further analyze our approach.

Ablation Study. As shown in Fig. 2, we align the soft logit rankings of the teacher and student models using a simple L2 loss. However, is there a better method for handling the soft logit rankings? We explore this question here. In Table 4, we compare the experimental results using different ranking alignment methods and also evaluate the impact of applying vanilla KD to our method. From Table 4, it can be seen that our proposed ranking alignment method with L2 loss achieves superior performance improvement compared to both L1 loss and Spearman's ρ-based loss (where Spearman's ρ [39] quantifies the monotonic relationship strength between teacher and student's soft logit rankings, and the loss is defined as $1 - \rho$). Moreover, applying vanilla KD to our method leads to a performance decline. One possible explanation is that our method avoids introducing unnecessary strict constraints to the student, whereas vanilla KD imposes such constraints, causing the performance to drop. The ablation study results further validate the effectiveness of our method.

Table 1. Results on CIFAR-100, Homogeneous Architectures. We compare our method with various logit distillation and feature distillation methods and report the Top-1 accuracy. The best and second-best results are highlighted in bold and underlined, respectively.

Teacher	resnet56 72.34	resnet32×4 79.42	wrn-40-2 75.61	wrn-40-2 75.61	vgg13 74.64	resnet110 74.31
Student	resnet20 69.06	resnet8×4 72.50	wrn-16-2 73.26	wrn-40-1 71.98	vgg8 70.36	resnet32 71.14
			Feature-based			
FitNets [18]	69.21	73.50	73.58	72.24	71.02	71.06
AT [19]	70.55	73.44	74.08	72.77	71.43	72.31
VID [20]	70.38	73.09	74.11	73.30	71.23	72.61
CRD [21]	71.16	75.51	75.48	74.14	73.94	73.48
CAT-KD [22]	71.05	76.91	75.60	74.82	74.65	73.62
ReviewKD [23]	71.89	75.63	76.12	75.09	74.84	73.89
NORM [24]	71.61	76.98	76.26	75.42	74.46	73.95
FCFD [25]	71.96	76.62	76.43	75.46	75.22	-
			Logit-based			
KD [8]	70.66	73.33	74.92	73.54	72.98	73.08
TAKD [9]	70.83	73.81	75.12	73.78	73.23	73.37
CTKD [10]	71.19	73.79	75.45	73.93	73.52	73.52
NKD [11]	70.40	76.35	75.24	74.07	74.86	72.77
DKD [12]	71.97	76.32	76.24	74.81	74.68	74.11
LSKD [13]	71.43	76.62	76.11	74.37	74.36	74.17
TTM [14]	71.83	76.17	76.23	74.32	74.33	73.97
MLKD [15]	72.19	77.08	76.63	75.35	75.18	74.11
CRLD [16]	72.10	77.60	76.45	75.58	75.27	74.42
CALD [17]	72.05	77.02	76.44	75.12	74.99	74.19
Ours	**72.39**	**77.97**	**77.13**	**75.98**	**75.84**	**75.29**

Table 2. Results on CIFAR-100, Heterogeneous Architectures. We compare our method with various logit distillation and feature distillation methods and report the Top-1 accuracy. The best and second-best results are highlighted in bold and underlined, respectively.

Teacher	resnet32×4 79.42	vgg13 74.64	resnet50 79.34	wrn-40-2 75.61	resnet32x4 79.42
Student	shufflenetv2 71.82	mobilenetv2 64.60	mobilenetv2 64.60	shufflenetv1 70.50	shufflenetv1 70.50
			Feature-based		
FitNets [18]	73.54	64.16	63.16	73.73	73.59
AT [19]	72.73	59.40	58.58	73.32	71.73
VID [20]	73.40	65.56	67.57	73.61	73.38
CRD [21]	75.65	69.63	69.11	76.05	75.11
CAT-KD [22]	78.41	69.13	71.36	77.35	78.26
Re-viewKD[23]	77.78	70.37	69.89	77.14	77.45
NORM [24]	78.32	69.38	71.17	77.63	77.79
FCFD [25]	78.18	70.65	71.00	77.99	78.12
			Logit-based		
KD [8]	74.45	67.37	67.35	74.83	74.07
TAKD [9]	74.82	67.91	68.02	75.34	74.53
CTKD [10]	75.31	68.46	68.47	75.78	74.48
NKD [11]	76.26	70.22	70.67	75.96	75.31
DKD [12]	77.07	69.71	70.35	76.70	76.45
LSKD [13]	75.56	68.61	69.02	76.62	75.62
TTM [14]	76.55	69.16	69.59	75.42	74.37
MLKD [15]	78.44	70.57	71.04	77.44	77.18
CRLD [16]	78.27	70.39	71.36	-	-
CALD [17]	77.89	70.42	71.19	77.25	77.63
Ours	**79.44**	**71.03**	**72.31**	**78.36**	**78.57**

Table 3. Results on Tiny-ImageNet. The Top-1 accuracy is reported, with the best results highlighted in bold.

Type			
	Teacher	resnet18 62.99	resnet18 62.99
	Student	mobilenetv2 56.28	shufflenetv2 60.78
Feature	AT [19]	57.18	62.45
	CRD [21]	61.18	63.97
Logit	KD [8]	58.35	62.26
	DKD [12]	62.04	65.06
	KD+DOT [38]	64.01	65.75
	LSKD [13]	64.20	65.97
	Ours	**64.30**	**66.58**

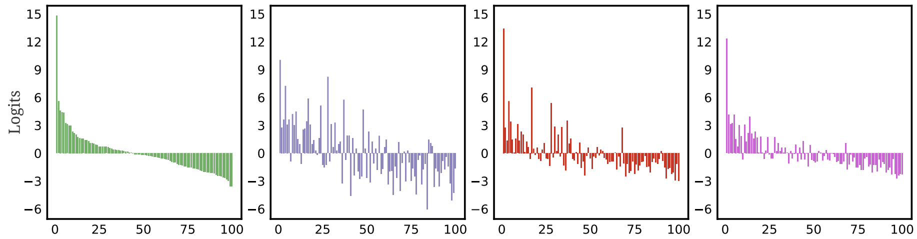
Table 4. Experiments conducted on CIFAR-100. The teacher model is resnet32x4, and the student model is resnet8x4. The Top-1 accuracy is reported.

CE	Vanilla KD	Ranking Alignment Method			Top-1 Acc
		L2	L1	Spearman’ ρ	
√	×	-	-	-	72.50
√	√	-	-	-	73.33
√	×	-	√	-	77.12
√	×	-	-	√	77.44
√	×	√	-	-	**77.97**
√	√	√	-	-	77.29

Analysis of Hyperparameter. We analyze the weight λ of Ranking Alignment Loss on CIFAR-100 and Tiny-ImageNet, with the results shown in Table 5. On CIFAR-100, the teacher model is resnet32×4, and the student model is resnet8x4. As λ increases, the Top-1 accuracy continuously rises, reaching the maximum value of 77.97 when λ is 5.0. On Tiny-ImageNet, the teacher model is resnet18, and the student model is mobilenetv2. As λ increases, the Top-1 accuracy continuously rises, reaching the maximum value of 64.30 when λ is 10.0. The hyperparameter experimental results further validate the effectiveness of our method.

Table 5. Analysis of the weight λ of Ranking Alignment Loss on different datasets. The Top-1 accuracy is reported.

Dataset	CIFAR-100	Dataset	Tiny-ImageNet
Teacher	resnet32×4 79.42	Teacher	resnet18 62.99
Student	resnet8×4 72.50	Student	mobilenetv2 56.28
$\lambda = 0.1$	75.59	$\lambda = 0.5$	62.15
$\lambda = 0.5$	76.74	$\lambda = 1.0$	63.16
$\lambda = 1.0$	77.17	$\lambda = 5.0$	64.02
$\lambda = 5.0$	**77.97**	$\lambda = 10.0$	**64.30**
$\lambda = 10.0$	77.71	$\lambda = 15.0$	64.03



(a) Teacher (b) Student (c) Vanilla KD (d) Ours

Fig. 3. In 3(a), the logits of the teacher are ranked; in 3(b), (c), and (d), the logits are assigned the same ranking as the teacher's logits, and the logit values corresponding to each ranking are shown. The teacher model is resnet32x4, and the student model is resnet8x4.

Visualization. To better illustrate the characteristics and advantages of our method, we randomly select an image from the CIFAR-100 test set, pass it through the teacher model, and rank the output logits, as shown in Fig. 3(a). Then, the same image is passed through the student trained with the standard approach, vanilla KD, and our method, respectively, and the corresponding logits are assigned the same ranking as the teacher's logits, and the logit values corresponding to each ranking are shown in Fig. 3(b), (c), and (d). We can observe that, compared to the student trained with the standard approach and vanilla KD, where the logit arrangement is sharp in some positions, the logit arrangement of the student trained with our method is relatively smoother. This is due to our method using the teacher's logit ranking as the learning objective, further validating the effectiveness of our approach.

5 Conclusion

To avoid introducing unnecessary strict constraints to the student model during knowledge distillation, we propose a novel knowledge distillation paradigm based on logit ranking alignment. To address the issue of non-differentiability in traditional hard ranking algorithm, we introduce a fast differentiable soft ranking algorithm to obtain the soft logit rankings of the teacher and student. We then use an L2 loss to align the soft logit rankings of the teacher and student. Experimental results on CIFAR-100 and Tiny-ImageNet validate that our method achieves superior performance improvement compared to previous logit distillation and feature distillation methods.

Acknowledgments. This work was supported by the Postdoctoral Fellowship Program of CPSF (under Grant No. GZB20240113), the Sichuan Science and Technology Program (granted No. 2024ZDZX0011 and No. 2025ZNSFSC1472), and the Sichuan Central-Guided Local Science and Technology Development Program (under Grant No. 2023ZYD0165).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Tokozume Y, Ushiku Y, Harada T. Between-class learning for image classification[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 5486-5494.
2. Rao Y, Zhao W, Zhu Z, et al. Global filter networks for image classification[J]. Advances in neural information processing systems, 2021, 34: 980-993.
3. Chen K, Chen B, Liu C, et al. Rsmamba: Remote sensing image classification with state space model[J]. IEEE Geoscience and Remote Sensing Letters, 2024.
4. Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
5. Wang A, Chen H, Liu L, et al. Yolov10: Real-time end-to-end object detection[J]. Advances in Neural Information Processing Systems, 2024, 37: 107984-108011.
6. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 3431-3440.
7. Sung C, Kim W, An J, et al. Contextrast: Contextual contrastive learning for semantic segmentation[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2024: 3732-3742.
8. Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network[J]. arXiv preprint arXiv:1503.02531, 2015.

9. Mirzadeh S I, Farajtabar M, Li A, et al. Improved knowledge distillation via teacher assistant[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(04): 5191-5198.
10. Li Z, Li X, Yang L, et al. Curriculum temperature for knowledge distillation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2023, 37(2): 1504-1512.
11. Yang Z, Zeng A, Li Z, et al. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 17185-17194.
12. Zhao B, Cui Q, Song R, et al. Decoupled knowledge distillation[C]//Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. 2022: 11953-11962.
13. Sun S, Ren W, Li J, et al. Logit standardization in knowledge distillation[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2024: 15731-15740.
14. Zheng K, Yang E H. Knowledge distillation based on transformed teacher matching[J]. arXiv preprint arXiv:2402.11148, 2024.
15. Jin Y, Wang J, Lin D. Multi-level logit distillation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 24276-24285.
16. Zhang W, Liu D, Cai W, et al. Cross-View Consistency Regularisation for Knowledge Distillation[C]//Proceedings of the 32nd ACM International Conference on Multimedia. 2024: 2011-2020.
17. Huang X, Chen W, Zhou W. Class-wise Adaptive Logits Distillation with Meta-Learning[C]//ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2025: 1-5.
18. Romero A, Ballas N, Kahou S E, et al. Fitnets: Hints for thin deep nets[J]. arXiv preprint arXiv:1412.6550, 2014.
19. Zagoruyko S, Komodakis N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer[J]. arXiv preprint arXiv:1612.03928, 2016.
20. Ahn S, Hu S X, Damianou A, et al. Variational information distillation for knowledge transfer[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 9163-9171.
21. Tian Y, Krishnan D, Isola P. Contrastive representation distillation[J]. arXiv preprint arXiv:1910.10699, 2019.
22. Guo Z, Yan H, Li H, et al. Class attention transfer based knowledge distillation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 11868-11877.
23. Chen P, Liu S, Zhao H, et al. Distilling knowledge via knowledge review[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 5008-5017.
24. Liu X, Li L, Li C, et al. Norm: Knowledge distillation via n-to-one representation matching[J]. arXiv preprint arXiv:2305.13803, 2023.
25. Liu D, Kan M, Shan S, et al. Function-consistent feature distillation[J]. arXiv preprint arXiv:2304.11832, 2023.
26. Tung F, Mori G. Similarity-preserving knowledge distillation[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 1365-1374.
27. Liu L, Huang Q, Lin S, et al. Exploring inter-channel correlation for diversity-preserved knowledge distillation[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 8271-8280.
28. Huang T, You S, Wang F, et al. Knowledge distillation from a stronger teacher[J]. Advances in Neural Information Processing Systems, 2022, 35: 33716-33727.

29. Blondel M, Teboul O, Berthet Q, et al. Fast differentiable sorting and ranking[C]//International Conference on Machine Learning. PMLR, 2020: 950-959.
30. Krizhevsky A, Hinton G. Learning multiple layers of features from tiny images[J]. 2009.
31. Le Y, Yang X. Tiny imagenet visual recognition challenge[J]. CS 231N, 2015, 7(7): 3.
32. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
33. Zagoruyko S, Komodakis N. Wide residual networks[J]. arXiv preprint arXiv:1605.07146, 2016.
34. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
35. Sandler M, Howard A, Zhu M, et al. Mobilenetv2: Inverted residuals and linear bottlenecks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 4510-4520.
36. Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6848-6856.
37. Ma N, Zhang X, Zheng H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 116-131.
38. Zhao B, Cui Q, Song R, et al. Dot: A distillation-oriented trainer[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 6189-6198.
39. Spearman C. The proof and measurement of association between two things[J]. 1961.