International Core Journal of Engineering Volume 11 Issue 3, 2025
ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

Multi-knowledge Collaborative Distillation Framework based on an Encoder-decoder Feature Projector

Kangping Chen, Hong Zhao*

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

Abstract

Knowledge distillation as a flexible model compression technique, is widely applied in various computer vision tasks to transfer knowledge from large-scale models to lightweight, small-scale models. However, existing knowledge distillation methods, particularly feature-based distillation approaches, often require the alignment of heterogeneous features, which can lead to a decline in student model performance due to misalignment issues. To address this, we propose a multi-knowledge collaborative distillation framework based on an encoder-decoder feature projector. To avoid the computational overhead introduced by complex feature alignment mechanisms, we reuse the teacher classifier and design an encoder-decoder-based feature projector to facilitate the alignment of deep features between the student and teacher models. Furthermore, considering the progressive learning process of the student model and reducing the additional workload caused by tuning distillation temperature parameters, we introduce a progressive distillation temperature adjustment mechanism. Extensive experiments on the benchmark dataset CIFAR-100 validate the effectiveness of our distillation method, achieving outstanding performance across various teacher-student architecture combinations.

Keywords

Multi-Knowledge Distillation; Encoder-Decoder Mechanisms; Feature Projector; Model Compression.

1. Introduction

In recent years, the rapid development of deep learning technology has led to groundbreaking advancements in computer vision tasks. However, as model architectures grow increasingly complex, the number of parameters in these models has surged, imposing stringent requirements on hardware environments during real-world deployment. Consequently, addressing the challenge of deploying deep learning models on resource-constrained devices has become an urgent issue.

Knowledge distillation (KD), as a flexible model compression method, effectively transfers knowledge from larger, high-performing teacher models to smaller, more lightweight student models. This approach was first proposed by Hinton et al.[1] Since then, KD has achieved significant progress in computer vision tasks. Nevertheless, a noticeable performance gap persists between teacher and student models. To bridge this gap, numerous KD methods have emerged in recent years, enhancing KD performance from various perspectives. For instance, Yang et al.[2] utilized the logits of intermediate teacher models, updated at each training epoch, to guide student models. To better exploit label knowledge and eliminate noise, Huang et al.[3] proposed a correlation-based logits distillation approach, replacing Kullback-Leibler (KL) divergence with Pearson correlation coefficients to extract intra-class relationships, thereby improving distillation performance and narrowing the performance gap between teacher and student models. Li et al.[4] designed a dynamic

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

learning temperature module that adaptively adjusts the distillation temperature, enabling the student model to maximally absorb knowledge transferred from the teacher.

Logits-based KD methods are simple and easy to use but are limited to leveraging only the information in model logit outputs and are typically applicable solely to classification tasks. Feature-based KD methods address these limitations by utilizing intermediate knowledge to enhance the transfer of knowledge to student models. Intermediate knowledge represents the high-dimensional features extracted from the middle layers of neural networks. Since the knowledge expressed by teacher neurons shares similar patterns with that of student models, teacher features can directly guide student training. For example, Tung et al.[5] proposed a similarity-preserving KD method by extracting pairwise feature activations from teacher and student models to generate similarity matrices, guiding the student model to learn the teacher's similarity structure and thereby improving its performance. Similarly, Zagoruyko et al.[6] extracted attention maps from teacher models and compelled students to mimic these maps, aligning the attention regions of student models with those of teachers.

To address performance degradation caused by feature mismatches in heterogeneous KD, Romero et al.[7] trained student models by minimizing the differences between intermediate feature maps of teacher and student models, thereby enhancing the student model's generalization and reliability by imitating the teacher's learning paradigm. Heo et al.[8] transferred knowledge through activation boundary distillation, which distilled the activation boundaries formed by hidden neurons, mitigating feature mismatches in heterogeneous KD processes.

While these methods have improved distillation performance by optimizing various aspects, the growing complexity of model architectures and the increasing number of parameters in computer vision tasks present new challenges. To further reduce the performance gap between teacher and student models and alleviate performance degradation caused by heterogeneous feature mismatches, we propose a collaborative multi-knowledge distillation framework. To address the issue of unaligned heterogeneous features, we employ feature projector to align the deep features of student and teacher models. Additionally, by leveraging the teacher classifier, we map the aligned student and teacher models to a unified probabilistic space, where the similarity between distilled features of teachers and students can be measured. To reduce the extra workload associated with hyperparameter tuning, we introduce a progressive distillation temperature adjustment module, which gradually increases the learning difficulty, guiding the student model to better assimilate the teacher model's knowledge.

2. Related Work

2.1 Logits-based Knowledge Distillation

Knowledge distillation is a method that transfers knowledge from larger and deeper neural networks to smaller, lightweight models with fewer parameters. This effective knowledge transfer technique has garnered significant attention from researchers. Logits-based KD primarily focuses on extracting and leveraging the "dark knowledge" embedded in the teacher model's logit outputs to guide the training of student models. For example, Zhao et al.[9] proposed the method of decomposition of the vanilla KD approach by separating the teacher model's logits into target and non-target classes. The authors argue that non-target class responses contain critical "dark knowledge," and by adjusting the weight of non-target class knowledge, the efficiency of knowledge transfer within logits is improved. To further explore the latent knowledge within logit outputs, Jin et al.[10] introduced a more robust logit extraction method. By aligning predictions at multiple levels-not only at the instance level but also at batch and class levels-the framework enables the student model to simultaneously learn instance-level predictions, batch-level predictions, and class-level correlations. This approach reduces discrepancies between teacher and student models across instances, batches, and classes. Considering inter-class relationships within sample data, Yun et al.[11] proposed an inter-class self-distillation approach. During training, the method extracts prediction distributions among different samples of the same label, forcing the network to generate more meaningful predictions through class-

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

level regularization. This reduces overconfident predictions, decreases intra-class variance, and enhances correlations between similar classes, ultimately improving the prediction accuracy for fine-grained categories. As teacher models become increasingly powerful, student models often lack sufficient capacity to fully absorb the knowledge from the teacher model.

Although logits-based KD has made significant progress in extracting dark knowledge from teacher logits and guiding student model training, it has inherent limitations. Since it relies on classifier outputs, logits-based KD is less applicable to tasks driven by semantic understanding, such as object detection, semantic segmentation, and person re-identification.

2.2 Feature-based Knowledge Distillation

Feature knowledge refers to the semantic features extracted from intermediate layers of deep learning models. Compared to label knowledge, intermediate-layer knowledge contains richer information, significantly enhancing the capacity and informativeness of knowledge transfer, thereby improving the effectiveness of distillation training. The variety of knowledge types that can be extracted from intermediate layers provides greater richness and flexibility.

In recent years, numerous effective feature knowledge distillation methods have been proposed. For instance, Ji et al.[12] introduced a similarity-matching method for distilled feature pairs, utilizing attention vectors to capture the relationships between teacher and student features. This approach selectively transfers teacher feature knowledge to the student model. Unlike methods that compute similarity between distilled feature pairs, Xu et al.[13] proposed a distillation method that constrains intermediate-layer features. By adding an additional fully connected classifier to each intermediate layer and using the teacher's logits to regulate the intermediate features, this method effectively mitigates issues like gradient explosion or vanishing gradients. Recognizing that the feature knowledge of a single teacher model may not be suitable for guiding the student model throughout the entire training process, Cao et al.[14] proposed a progressive distillation method with multiple teachers. This approach constructs a teacher pool and employs a greedy search algorithm to identify a sequence of teachers, enabling the student model to adapt gradually during training. A backward greedy selection (BGS) mechanism was developed to automatically determine an optimal sequence of teacher models for distillation. While these methods demonstrate excellent distillation performance, they often require significant computational resources. To address this, Chen et al.[15] introduced a method that transfers the teacher's classifier to the student model as a replacement for the original classifier, employing a simple feature mapping mechanism to adapt the student's features for use with the teacher's classifier. This straightforward mechanism effectively enhances the student model's performance. Additionally, some researchers have leveraged self-distillation mechanisms to transfer feature knowledge. Jang et al.[16] designed a feature knowledge-based distillation method that uses contrastive loss to encourage intermediate-layer features to learn from the final layer. By maximizing the mutual information between intermediate-layer and final-layer features, this approach significantly improves the performance of the student model.

Existing feature-based knowledge distillation methods have been widely applied, but most rely on complex feature transformations or sophisticated representation-matching loss calculations. These approaches often impose higher demands on the hardware environment for model distillation. Moreover, in many distillation methods, careful tuning of distillation hyperparameters is required to achieve the desired performance, leading to additional workload. Direct alignment of teacher and student representations through feature mapping may reduce distillation effectiveness in heterogeneous distillation scenarios and could even adversely affect the student model's performance.

3. Method

This paper proposes a multi-knowledge collaborative distillation framework based on an encoder-decoder feature projector, as illustrated in Figure 1. Specifically, we introduce an encoder-decoder structured feature projector module designed to assist the student model in aligning with the teacher’s features. The intermediate features of the student model, after being encoded and aligned to match

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

the dimensions of the teacher's features, are fed into the teacher model's classifier. The KL divergence is then used to measure the discrepancy between the teacher and student features within the unified probability space. Furthermore, to optimize the parameters of the feature projector module and effectively model the relationship between the student and teacher features, we calculate the Mean Squared Error (MSE) loss between the final-layer features of the teacher model and the deep-layer features of the student model. Additionally, to address potential feature decorrelation during the mapping process, we employ a reconstruction loss within the encoder-decoder structure to constrain the correlation between the student's features before and after mapping.

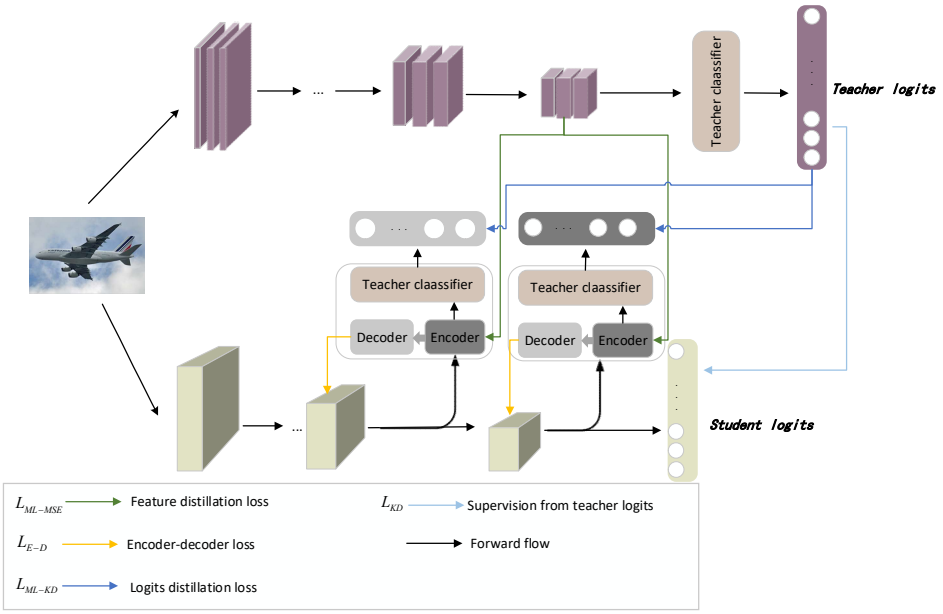

Figure 1. Details of our framework. The framework consists of two offline networks: teachers and students.

3.1 Encoder-decoder Feature Projector

We introduce an encoder-decoder feature projector, as shown in Figure 2. The encoding module is used to align the student's features with the teacher's features, while the decoding module is introduced to maintain the correlation of the student's features before and after mapping. To facilitate the training of the encoder-decoder module, we use MSE loss to measure the similarity between the features before and after encoding and decoding, providing gradient information for optimization.

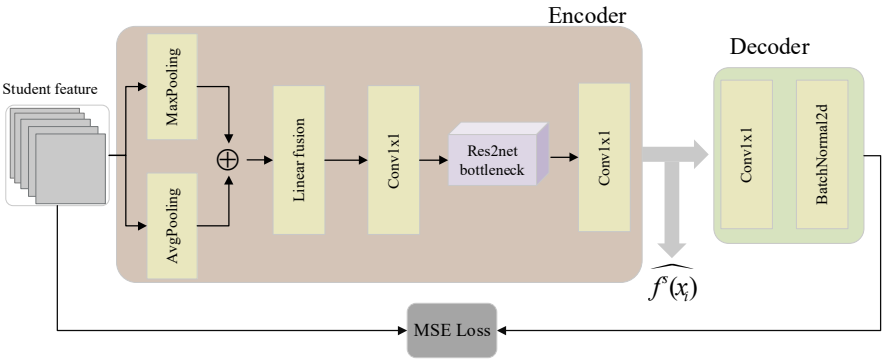

Figure 2. Encoder-decoder feature projector. $f^s(x_i)$ is the deep features of the student model aligned to the teacher features after the feature projector.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

Let the given dataset be $D=\{(x_i,y_i)\}_{i=1}^{I}$, with samples x_i and corresponding labels y_i. The teacher and student models share the same input x_i. The student model produces feature outputs $f_n^t(x_i)$ from the final layers, and student model produces feature outputs $f_n^s(x_i)$ and $f_{n-1}^s(x_i)$ from the penultimate and final layers, respectively, with the final-layer output represented as $f_n^t(x_i)$. Due to inherent differences between the teacher's and student's features, feature projector alignment is performed as expressed in Equation (1).

$$\widehat{f_n^s(x_i)} = Encoder(f_n^s(x_i)) \tag{1}$$

The aligned student feature: $\widehat{f_n^s(x_i)}$, matches the teacher's feature dimensions after the mapping. Similarly, the student's penultimate-layer feature is aligned with the teacher's final-layer feature to produce the aligned student feature $\widehat{f_{n-1}^s(x_i)}$.

Considering the potential decorrelation caused by the feature projector module during the alignment process, the correlation between the student's features before and after mapping may significantly degrade. To address this, a decoding module is introduced to constrain the feature projector module, ensuring that the alignment with the teacher's features does not compromise the structural information of the student's original features. The decoded student feature is represented as $f^{s'}(x_i)=Decoder(\widehat{f^s(x_i)})$. We use MSE loss to measure the correlation between the student's features before and after encoding-decoding. We use small loss weights to preserve this correlation without affecting the student model's ability to learn the teacher feature. The computation process is described in Equation (2):

$$L_{E-D}=\sum_{l=n-1}^{n}\sum_{i=1}^{I}\| f_l^s(x_i)-f_l^{s'}(x_i)\| \tag{2}$$

Here, n is the number of strata in the student model, I is the number of samples.

3.2 Multi-knowledge Distillation

To enhance the performance of the distilled model without relying on complex feature mapping mechanisms, we align the deep features of the teacher and student models within a unified probabilistic space by reusing the teacher classifier. This approach alleviates the decline in distillation performance caused by differences in directly matched features. Specifically, we freeze the parameters of the teacher model, where the classifier weights are denoted as W_t. The mapped deep features of the student model $\widehat{f^s(x_i)}$, passing through the teacher classifier to obtain logits output $\widehat{p_i^s}=W_t\widehat{f^s(x_i)}$. The logit outputs of the teacher, and the student: p_i^t, are softened using a distillation temperature. The softened logits are then compared using KL divergence to measure their difference. The computation process is detailed in Equation (3):

$$L_{MLKD}=\sum_{j=n-1}^{n}\sum_{i=1}^{I}\tau^2 L_{KL}(\widehat{p_i^{s_j}},p_i^t) \tag{3}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

Here, τ represents the distillation temperature, n denotes the number of layers in the student model, and I indicates the number of samples. To constrain the parameter optimization of the encoder-decoder feature projector module and establish an effective feature mapping relationship, we measure the similarity between the teacher's deep features and the student's encoded features using MSE. The specific calculation process is shown in Equation (4):

$$L_{MSE} = \sum_{j=n-1}^{n} \sum_{i=1}^{I} \| f_n^t(x_i) - \widehat{f_j^s(x_i)} \|_2^2 \tag{4}$$

The overall loss is expressed in Equation (5):

$$L_{total} = L_{CE} + L_{KL} + \alpha L_{E-D} + \beta L_{ML-KD} + \gamma L_{ML-MSE} \tag{5}$$

Here, L_{CE} represents the task training loss of the model itself, and L_{KL} is the vanilla KD loss between the logit outputs of the teacher and the student. α, β, and γ are hyperparameters used to adjust the weights of different losses.

3.3 Progressive Distillation Temperature Adjustment Mechanism

Considering that the learning process of the student model during distillation is continuously progressive, maintaining a constant distillation temperature may lead to suboptimal results. We introduce a progressive distillation temperature adjustment mechanism, allowing the student model to learn knowledge in a gradual manner, from simple to complex. We use a difficulty level coefficient μ to adjust the distillation temperature τ, $\tau = \tau_{init} + \mu\tau_{init}$. The update process for the difficulty level coefficient is shown in Equation (6):

$$\mu = \mu_{\min} + \frac{(\mu_{\max} - \mu_{\min})(1 + \cos((1 + \frac{\min(E, E_l)}{E_l})\pi))}{2} \tag{6}$$

Here, $\mu_{\min}$ and $\mu_{\max}$ represent the range of μ, E is the total number of training epochs, and E_l is a hyperparameter used to gradually adjust the difficulty coefficient μ. $\mu_{\min}$ and $\mu_{\max}$ are set to 0 and 1, respectively, and E_l is set to 100, meaning that μ changes from 0 to 1 over 100 training epochs and remain constant thereafter until the end of training.

The training process of the model is detailed in Algorithm 1:

Algorithm 1: Multi-knowledge collaborative distillation method under an encoder-decoder feature projector

Input: Teacher network T, Student network S, input image x, ground truth label y, hyper-parameters $\alpha, \beta, \gamma, \tau_{init}, \mu_{\min}, \mu_{\max}$

Output: the updated S.

For epoch in range(epochs):

For x, y in loader:

1. Input x to T to obtain teacher feature: $f_n^t(x_i)$, and student feature: $f_n^s(x_i)$, $f_{n-1}^s(x_i)$.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

2. Computation of student feature $\widehat{f^s(x_i)}$ aligned to teacher feature
3. Input $\widehat{f^s(x_i)}$ to $Decoder(\cdot)$ obtain $f^{s'}(x_i)$. Calculate L_{E-D} from Eq. (2).
4. Input $\widehat{f_n^s(x_i)}$ to W_t obtain $\widehat{p_i^{s_n}}$. Calculate L_{ML-KD} from Eq. (3)
5. Calculate L_{ML-MSE} from Eq. (4)
6. Total loss from Eq. (5)
7. Update hyper-parameters τ.
8. Update S by optimizing total loss

End

End

4. Experiment

In this section, we validate the performance of our distillation method through experiments on benchmark datasets and compare it with representative distillation methods to further demonstrate the superiority of our approach.

Experimental Setup: The server used for experiments is equipped with a 48-core Intel(R) Xeon(R) Gold 5218 CPU. The experiments are conducted using the PyTorch 2.0 deep learning framework with Python 3.8 on a Linux operating system. For GPU acceleration, we use a Tesla A100 PCIe GPU with 40 GB of memory, configured with NVIDIA CUDA 12.2 and cuDNN V11.6.55 deep learning libraries.

Dataset and Benchmark: We use CIFAR-100 as the benchmark dataset and apply standard data augmentation techniques[17], all images are normalized using channel-wise mean and standard deviation. We compare our method with several other approaches, including vanilla KD[1], FitNet[7], RKD[18], CRD[19], OFD[20], SRRL[21], ReviewKD[22], and DKD[9].

4.1 Model Compression Experiment

Table 1. Same-architecture model distillation, Δ representing accuracy improvement over vanilla KD

Teacher	ResNet32x4	ResNet110	ResNet56	Wrn-40-2	Wrn-40-2	Vgg13
Top-1	79.42	74.31	72.33	75.6	75.61	74.64
Student	ResNet8x4	ResNet32	ResNet20	Wrn-40-1	Wrn-16-2	Vgg8
Top-1	72.5	71.14	69.06	71.98	73.26	70.36
FitNet	73.5	71.06	69.21	72.24	73.58	71.02
RKD	71.9	71.82	69.61	72.22	73.35	71.48
CRD	75.51	73.48	71.16	74.14	75.48	73.94
OFD	74.95	73.23	70.98	74.33	75.24	73.95
ReviewKD	75.63	**73.89**	<u>71.89</u>	**75.09**	<u>76.12</u>	**74.84**
SSRL	75.39	73.54	71.31	74.64	75.79	74.4
KD	73.33	73.08	70.66	73.54	74.92	72.98
DKD	<u>76.32</u>	73.66	71.43	74.54	75.7	74.41
Ours	**77.26**	<u>73.87</u>	**72.01**	<u>74.98</u>	**76.71**	<u>74.5</u>
Δ	3.93	0.79	1.35	1.44	1.79	1.52

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

To evaluate the effectiveness of our proposed distillation method, we conducted experiments on the CIFAR-100 dataset. Tables 1 and 2 present the comprehensive performance comparison of various distillation methods across 11 network combinations. In all experiments, the student models were trained from scratch.

Table 1 shows the results of distillation within models of the same architecture. Our proposed method demonstrates outstanding performance on CIFAR-100. For instance, in the distillation combination where the teacher model is ResNet32x4 and the student model is ResNet8x4, our method achieves a 5.3% improvement compared to vanilla KD. Across other same-architecture distillation combinations, our method consistently yields significant performance improvements for the student models. This can be attributed to the use of our encoder-decoder feature projector, which allows the student model to retain its unique characteristics while learning the representative information from the teacher's deep features, rather than merely minimizing the feature differences between the teacher and student models. Furthermore, the multi-knowledge distillation loss supervises the student model's learning from the teacher model at multiple levels, effectively ensuring the completeness of knowledge transfer.

Table 2. Different-architecture model distillation, Δ representing accuracy improvement over vanilla KD

Teacher	ResNet32x4	ResNet32x4	ResNet50	Vgg13	Wrn-40-2
Top-1	79.42	79.42	79.34	74.64	75.61
Student	Shufflev1	Shufflev2	Mobilenetv2	Mobilenetv2	Shufflev1
Top-1	70.5	71.82	64.6	64.6	70.5
FitNet	73.59	73.54	63.16	64.14	73.73
RKD	72.28	73.21	64.43	64.52	72.21
CRD	75.11	75.65	69.11	69.73	76.05
OFD	75.98	76.82	69.04	69.48	75.85
ReviewKD	77.45	77.78	69.89	**70.37**	77.14
SSRL	75.66	76.4	69.45	69.14	76.61
KD	74.07	74.45	67.35	67.37	74.83
DKD	75.44	76.48	**70.35**	69.71	76.70
Ours	**77.51**	**77.8**	70.28	69.78	**77.81**
Δ	3.44	3.35	2.93	2.41	2.98

From the heterogeneous distillation results, it is evident that our method achieves the best or second-best performance among all comparison methods. On the one hand, this may be due to the reuse of the teacher classifier to align the deep features of the teacher and student models, avoiding the performance degradation caused by directly matching heterogeneous distillation feature pairs. By comparing the differences in a unified probability space, the alignment reduces biases introduced by architectural differences. On the other hand, the encoder-decoder feature projector structure helps retain the intrinsic properties of the student model's features, enabling the learned semantic representations to better adapt to the student's architecture.

4.2 Ablation Study

In this section, we validate the effectiveness of the different modules proposed in our method through ablation experiments. Specifically, we conduct experiments on the following variants: Variant A:

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

$L_{CE}+L_{KL}+L_{MSE}$. Uses the bottleneck structure of the ResNet model as the feature projector module to align student and teacher features, with MSE used to measure the discrepancy between the mapped student and teacher deep features. Variant B: $L_{CE}+L_{KL}+L_{E-D}$. L_{E-D} loss optimizing the parameters of the encoder-decoder module. Variant C: $L_{CE}+L_{KL}+L_{E-D}+L_{ML-KD}$. L_{ML-KD} loss term from the multi-knowledge distillation mechanism to measure the differences in the probability space between the distillation feature pairs. Variant D: $L_{CE}+L_{KL}+L_{E-D}+L_{ML-KD}+L_{ML-MSE}$. L_{ML-MSE} loss from the multi-knowledge distillation mechanism to supervise the parameter optimization of the encoder module within the feature projector module.

Table 3. Ablation experiments. We use ResNet32x4 as the teacher model, ResNet8x4 and Shufflev2 as the student model, and the hyperparameter settings in the experiment are consistent with the experiment in section 4.1

Student	ResNet-8x4		Shufflev2	
	Acc1(%)	Acc5(%)	Acc1(%)	Acc5(%)
$L_{CE}+L_{KL}+L_{MSE}$	74.12	92.81	74.2	92.87
$L_{CE}+L_{KL}+L_{E-D}$	74.52	93.11	74.68	93.17
$L_{CE}+L_{KL}+L_{E-D}+L_{ML-KD}$	76.59	93.74	76.94	93.87
$L_{CE}+L_{KL}+L_{E-D}+L_{ML-KD}+L_{ML-MSE}$	**77.26**	94.2	**77.8**	**94.23**

The results of Table 3 show that our feature projector mechanism outperforms the bottleneck-based feature projector structure in terms of feature alignment performance. Furthermore, compared to Variant A, Variant C exhibits a significant improvement in distillation performance. Variants incorporating the multi-knowledge distillation loss deliver substantial performance gains. Variant D achieves the best distillation performance, which may be attributed to the alignment of distillation feature pairs via the teacher classifier and the use of the encoder-decoder feature projector mechanism to guide the student model in constructing effective deep representations. This also demonstrates that the multi-knowledge collaborative distillation loss effectively supervises the encoder-decoder feature projector, establishing robust mappings between student and teacher distillation feature pairs.

5. Conclusion

We propose a multi-knowledge collaborative distillation method based on an encoder-decoder feature projector mechanism. We designed an encoder-decoder feature projector module that retains the unique characteristics of the student model to a certain extent, guiding the student model to reconstruct representations adapted to its own architecture. Additionally, we designed a multi-knowledge distillation mechanism that reuses the teacher classifier to measure the differences in the probability space between distillation feature pairs, mitigating the impact of model architecture differences. Finally, we introduced a self-adjusting distillation temperature mechanism, reducing the additional workload associated with tuning the distillation temperature hyperparameter. Our experiments show that by adjusting the weights of L_{ML-MSE} and L_{ML-KD} loss, superior distillation results can be achieved in both same and different architecture distillation combinations. This indicates that directly matching features after feature mapping is unsuitable for certain distillation scenarios. By appropriately balancing the weights of different distillation losses, better distillation performance can be obtained. The experiments on the CIFAR-100 dataset further validate the superiority of our proposed method.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62166025).

References

[1] HINTON G. Distilling the Knowledge in a Neural Network[J]. arXiv preprint arXiv:1503.02531, 2015.

[2] YANG C, XIE L, SU C, et al. Snapshot distillation: Teacher-student optimization in one generation[C/OL]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 2859-2868[2024-03-12]. http://openaccess.thecvf.com/content_CVPR_2019/html/Yang_Snapshot_Distillation_Teacher-Student_Optimization_in_One_Generation_CVPR_2019_paper.html.

[3] 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.

[4] LI Z, LI X, YANG L, et al. Curriculum temperature for knowledge distillation[C/OL]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 37. 2023: 1504-1512[2024-04-11]. https://ojs.aaai.org/index.php/AAAI/article/view/25236.

[5] TUNG F, MORI G. Similarity-preserving knowledge distillation[C/OL]//Proceedings of the IEEE/CVF international conference on computer vision. 2019: 1365-1374[2024-03-12]. http://openaccess.thecvf.com/content_ICCV_2019/html/Tung_Similarity-Preserving_Knowledge_Distillation_ICCV_2019_paper.html.

[6] ZAGORUYKO S, KOMODAKIS N. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer[A/OL]. arXiv, 2017[2024-03-12]. http://arxiv.org/abs/1612.03928.

[7] ROMERO A, BALLAS N, KAHOU S E, et al. Fitnets: Hints for thin deep nets. arXiv 2014[J]. arXiv preprint arXiv:1412.6550, 2014.

[8] HEO B, LEE M, YUN S, et al. Knowledge transfer via distillation of activation boundaries formed by hidden neurons[C/OL]//Proceedings of the AAAI conference on artificial intelligence: vol. 33. 2019: 3779-3787[2024-03-12]. https://aaai.org/ojs/index.php/AAAI/article/view/4264.

[9] 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.

[10] JIN Y, WANG J, LIN D. Multi-level logit distillation[C/OL]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 24276-24285[2024-08-05]. http://openaccess.thecvf.com/content/CVPR2023/html/Jin_Multi-Level_Logit_Distillation_CVPR_2023_paper.html.

[11] YUN S, PARK J, LEE K, et al. Regularizing class-wise predictions via self-knowledge distillation[C/OL]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 13876-13885[2024-03-12]. http://openaccess.thecvf.com/content_CVPR_2020/html/Yun_Regularizing_Class-Wise_Predictions_via_Self-Knowledge_Distillation_CVPR_2020_paper.html.

[12] JI M, HEO B, PARK S. Show, attend and distill: Knowledge distillation via attention-based feature matching[C/OL]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 35. 2021: 7945-7952[2024-03-12]. https://ojs.aaai.org/index.php/AAAI/article/view/16969.

[13] XU C, GAO W, LI T, et al. Teacher-student collaborative knowledge distillation for image classification[J/OL]. Applied Intelligence, 2023, 53(2): 1997-2009. DOI:10.1007/s10489-022-03486-4.

[14] CAO S, LI M, HAYS J, et al. Learning lightweight object detectors via multi-teacher progressive distillation[C/OL]//International Conference on Machine Learning. PMLR, 2023: 3577-3598[2025-01-16]. https://proceedings.mlr.press/v202/cao23c.html.

[15] CHEN D, MEI J P, ZHANG H, et al. Knowledge distillation with the reused teacher classifier[C/OL]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 11933-11942[2024-03-12]. http://openaccess.thecvf.com/content/CVPR2022/html/Chen_Knowledge_Distillation_With_the_Reused_Teacher_Classifier_CVPR_2022_paper.html.

[16] JANG J, KIM S, YOO K Y, et al. Self-Distilled Self-Supervised Representation Learning.[J/OL]. 2021[2024-03-12]. http://arxiv.org/abs/2111.12958v1. DOI:10.48550/arXiv.2111.12958.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202503_11(3).0014

[17] HE K, ZHANG X, REN S, et al. Deep Residual Learning for Image Recognition[C]//IEEE Conference on Computer Vision and Pattern Recognition. 2016.

[18] PARK W, KIM D, LU Y, et al. Relational knowledge distillation[C/OL]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 3967-3976[2024-03-12]. http://openaccess.thecvf.com/content_CVPR_2019/html/Park_Relational_Knowledge_Distillation_CVPR_2019_paper.html.

[19] TIAN Y, KRISHNAN D, ISOLA P. Contrastive Representation Distillation[A]. arXiv, 2022.

[20] HEO B, KIM J, YUN S, et al. A comprehensive overhaul of feature distillation[C/OL]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 1921-1930[2024-04-02]. http://openaccess.thecvf.com/content_ICCV_2019/html/Heo_A_Comprehensive_Overhaul_of_Feature_Distillation_ICCV_2019_paper.html.

[21] YANG J, MARTINEZ B, BULAT A, et al. Knowledge distillation via softmax regression representation learning[C/OL]//International Conference on Learning Representations. 2020[2024-03-12]. https://openreview.net/forum?id=ZzwDy_wiWv.

[22] 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.