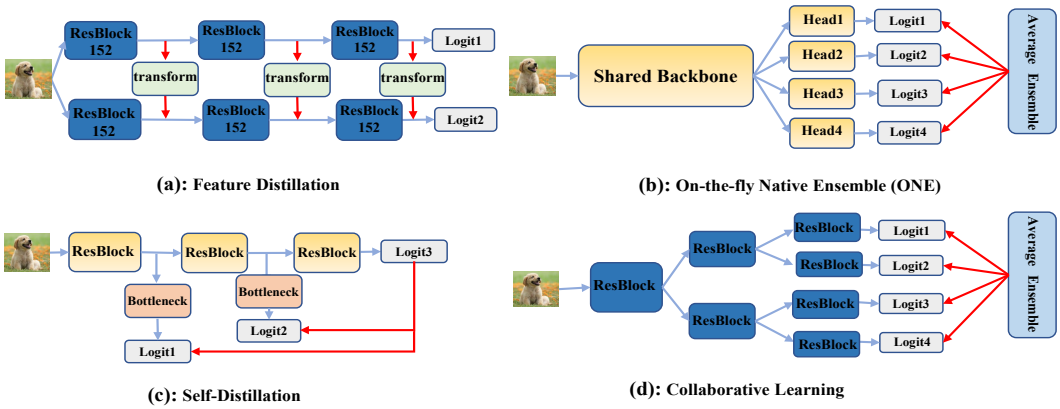


Figure 4: **Several popular frameworks for distillation**. (a) [43] extends the feature maps as the learning target instead of logits. (b) [69] adds several individual auxiliary heads on the backbone, and all peers share the same early backbone. (c) [59] adds early-exit bottlenecks to the shallow blocks to get multiple peer models, and the shallow peers can only obtain teacher knowledge from the deepest one. (d) [46] proposes a hierarchical structure so that the peers can share different parts in the early blocks, which is also adopted by our WML.

A Implementation Details

In this paper, we conducted a series of experiments on several image classification tasks to evaluate the effectiveness of our framework. There are three types of networks, including ResNet18, ResNet50, and ResNet101 on CIFAR10, CIFAR100, and ImageNet to compare with self-distillation methods and knowledge distillation baselines; ResNet32 and ResNet56 on CIFAR10 to compare with network pruning baselines; and also the MobileNetV2 on ImageNet dataset to compare with self-distillation baseline and also network pruning baselines. CIFAR10 contains 50K training images and 10K test images with 10 classes and CIFAR100 is similar with CIFAR100 with only has 100 classes. ImageNet contains over 1.2 million training images and 50K test images with 1000 classes. We follow the typical data augmentation of these three datasets, where only random resized cropping and horizontal flipping are applied on CIFAR10 and CIFAR100 to get 32×32 images. In contrast, on ImageNet, we also use the typical random resized crop and horizontal flipping, and the brightness, contrast, saturation data augmentation, to get 224×224 images.

In the CIFAR experiments, the number of initial filters is set as 64 for ResNet18, ResNet50, and ResNet101, while 16 for ResNet32 and ResNet56. The SGD optimizer is selected with a momentum of 0.9 and a weight decay of 0.0005. The batch size is set at 128 and the model is trained with 200 epochs. For the MobileNet on ImageNet, the model is also trained by SGD with 90 epochs with 256 batch size. Following [59], we set α 0.1 during the knowledge distillation in all experiments. The default step size η to update ω is also set as 0.1.

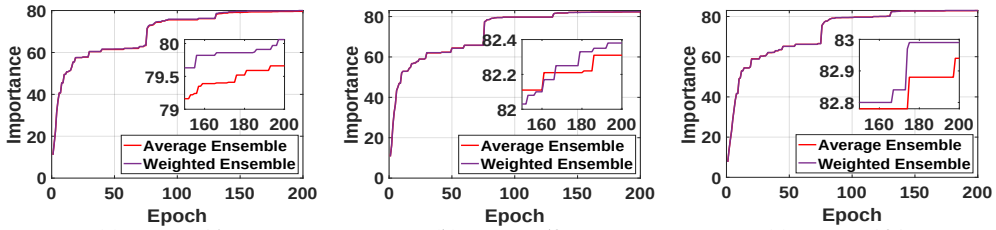
In our WML, we specify different pruning ratios to increase the diversity among students, as it is supposed to yield more informative knowledge in online distillation[9, 16], where similar phenomenon also exists in the neural architecture search [63–65]. We default pruned peer models to be of similar size as Classifiers in self-distillation [59] for fair comparison in the main experiments. For example, the Classifier1-4 contain $\{32.2\%, 53.5\%, 76.9\%, 100\%\}$ parameters as the baseline ResNet50, so we set the pruning ratio as $P = \{0.7, 0.5, 0.3, 0\}$ in our WML in the main experiments to get four peer models. In our channel pruning, we leverage SNIP at initialization to get the relative importance of each layer in Eq.(2), and the number of to be pruned parameters in each layer is calculated as $p_l = P \times N \times \tilde{\mathcal{I}}_l$, where $\tilde{\mathcal{I}}_l$ is a normalization of $\frac{1}{\mathcal{I}_l}$ and N is the total number of parameters in the model. However, when the pruning ratio is too high, e.g., when P is larger than 0.8, the supposed pruned parameter p_l may be larger than N_l (the total number of parameters in this layer), which will result in layer collapse [49]. In our practical implementation, we set an upper-bounded ratio P_{up}=0.9 that only maximal 90% parameters in a layer can be pruned.

B Ablation Study on the Structure

The Figure 1 and Figure 4 summarized existing popular paradigm of knowledge distillation. Sec. 2 has detailed described the difference between the proposed framework and the existing works. Apart from the self-distillation, the peers in On-the-fly Native Ensemble (ONE) [69] and collaborative learning (CL) [46] also partially share blocks. In this paper, each head in ONE contains the last block and fully-connected layer in the ResNet, and CL

Table 6: Comparison results with online distillation on CIFAR100.

Networks	Methods	Baseline	Model1	Model2	Model3	Model4	Ensemble
ResNet18	ONE [69]	77.09	77.12	77.08	77.16	77.09	77.51
	CL [46]	77.09	77.34	77.53	77.62	77.57	78.16
	WML-P	77.09	77.05	77.65	77.89	78.01	78.45
	WML-S	77.09	77.40	78.69	79.05	79.50	81.41
ResNet50	ONE [28]	77.68	78.95	78.82	78.81	79.03	79.48
	CL [46]	77.68	79.33	79.46	79.01	79.27	80.51
	WML-P	77.68	79.02	79.16	79.66	79.78	80.53
	WML-S	77.68	80.16	80.58	80.85	81.25	82.78
ResNet101	ONE [28]	77.98	79.49	79.39	79.46	79.59	79.77
	CL [46]	77.98	80.49	80.26	80.42	80.24	81.37
	WML-P	77.98	80.18	80.35	80.65	80.43	81.44
	WML-S	77.98	80.73	81.23	81.34	81.43	83.12



(a) ResNet18 (b) ResNet50 (c) ResNet101

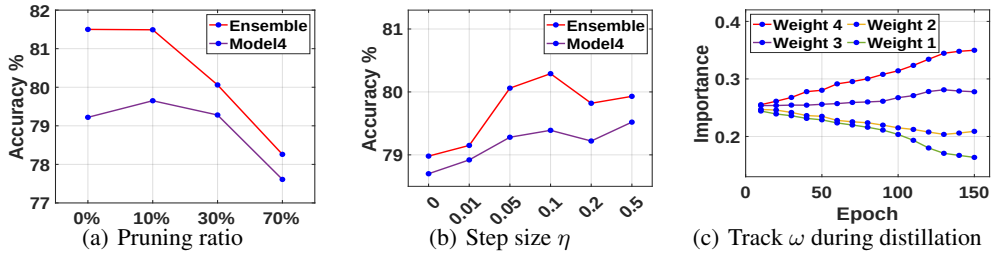
Figure 5: Track the average-ensemble and weighted-ensemble for WML.

share the same hierarchy structure as our WML. From the structure perspective, the main difference between ONE and our WML is that the backbone of ONE is the same for all peers, which means that it is impossible to prune peers under different ratios. Although collaborative learning also considers the same hierarchy structure as us, its peers are with the same structure, while WML introduces channel pruning to diversify the structures of peers. From the optimization perspective, the primary difference between our WML with all existing frameworks is that we propose a novel weighted mutual learning strategy in which each peer dynamically learns from the remaining peers, while the average ensemble is the most popular paradigm to get the knowledge source during the distillation for existing works. Figure 4 (a) presents another paradigm in distillation, feature distillation, which extends the intermediate feature map as the learning target instead of logits, so as steer the student's intermediate feature map towards the teacher rather than only the output logits. Recent work on feature distillation [61] verified the superiority of this learning paradigm. In this paper, we focus on knowledge distillation and leave feature distillation to our future work.

This section conducts experiments to show the superiority of our hierarchy structure with pruning to diversify the peers. In this section, we consider four peers and replace the weighted mutual learning with the average ensemble for distillation, as denoted by WML-P, where all peers are under $\{0.3, 0.2, 0.1, 0\}$ pruning ratio. Different from CL, which is additionally equipped with a backpropagation rescaling, we empirically found that it could only bring ignorable improvements to our hierarchy structure with pruning. In this way, we dismiss this operator in WML-P. Table 6 presents the reproduced comparison between WML-P and two online distillation competitors, ONE and CL, under the same experimental settings. As shown, we can find that WML-P and CL both outperform ONE. One potential reason is that the frameworks of WML-P and CL both contain more parameters than ONE, which further enhance the discriminative ability of the online distillation framework. More interesting, we found that our WML-P, which is also the pruned version of CL with fewer parameters, can achieve slightly better results, especially in larger networks, e.g., ResNet101. One potential reason is that those larger networks can maintain their discriminative ability under slight pruning. In contrast, the diversity that the pruning brings can further enhance the ensemble performance so as to improve the distillation.

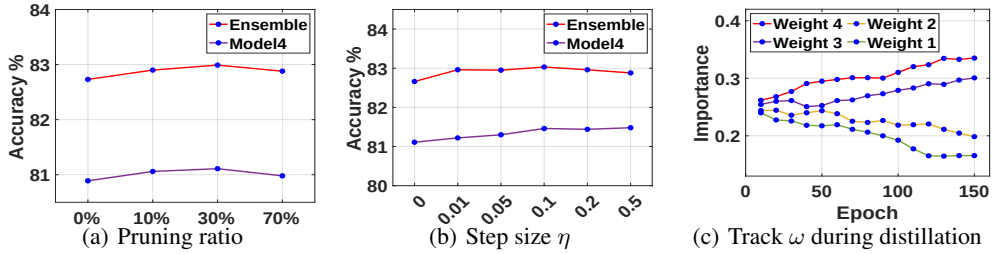
C Discussion on the Generalization of Weighted Ensemble

One of the key contributions of this paper is that we proposed weighted mutual learning, which adaptively distills knowledge from the peer models to a student by assigning different weights to different peers under



(a) Pruning ratio (b) Step size η (c) Track ω during distillation

Figure 6: Ablation studies on pruning ratio, step size η and peer importance on ResNet18.



(a) Pruning ratio (b) Step size η (c) Track ω during distillation

Figure 7: Ablation studies on pruning ratio, step size η and peer importance on ResNet101.

different capacities. To further verify the effectiveness and generalization of the proposed weighted mutual learning, we conduct a series of experiments to compare WML-P and WML-S, where WML-P only leverages the pruning to diversify the peers without the weighted mutual learning. For fair comparison, in this section, we only slightly prune the neural networks, e.g. with $\{0.3, 0.2, 0.1, 0\}$ pruning ratio for four peers in WML-P and WML-S (the pruning ratio here is smaller than the pruning ration in the maintext), and Table 6 presents the comparison results. Compared with results in Table 1, which pruned much more parameter, we can find that WML-S can obtain higher accuracy in Table 6, especially for small networks, e.g. ResNet18. A similar finding can be found in the next section on the ablation study on pruning ratio. As shown in Table 6, WML-S gains performance improvements over WML-P for all peers in ResNet18 from 0.35% to 1.49%, and increase the ensemble performance by 2.96%. Similar results can also be found in larger neural networks, e.g., ResNet50 and ResNet101. However, we can see a decrease in the performance gains when network becomes larger, which shares a similar trend as the number of heads. A similar finding can be found in the analysis of η.

In the following, we further investigate whether the learned weights capture the importance of each peer by comparing the optimization trajectory of average-ensemble and weighted-ensemble. In this experiment, we consider a similar experimental setting to our main experiments, so making the peers with the identical model sizes as the self-distillation. As shown in Figure 5, we can find that leveraging the learned weight can further enhance the performance of the ensemble by tuning the importance of each peer. This advantage is more obvious in the small ResNet18. The above phenomenons imply that tuning the weights is another option to improve the discriminative ability of the ensemble, especially when each peer model is not powerful enough.

D More Ablation Study on Peer Pruning and Dynamic Weighting Strategy

This section gives the complementary results of the ablation study on the dynamic weighting strategy and peer pruning. Figure 2 in the Sec.4.4 only reports the results on ResNet50, and we release more results of ResNet18 and ResNet101 in Figure 6 and 7. Interesting, we can find a controversial phenomenon in ResNet18 and ResNet101 on the pruning ratio, as shown in Figure 6 (a) and Figure 7 (a), where we can see a sharp drop on performance with the higher pruning ratio in ResNet18 while ResNet101 can maintain or even improve the performance with the pruning. The potential reason is that the ResNet101 is much larger than Resnet18, and it also contains much more redundant parameters that pruning 30% parameters will not deteriorate its discriminative ability. On the contrary, pruning 30% parameters clearly affect the discriminative ability of ResNet18 so as deteriorate the performance. In addition, for a larger neural network, our WML is more robust to the η, as shown in Figure 6 (b) and Figure 7 (b).

In Figure 6 (c) and Figure 7 (c), we also prune Model 1-4 with different pruning ratio, e.g., $\{0.7, 0.45, 0.25, 0\}$ for ResNet18 and $\{0.9, 0.75, 0.1, 0\}$ for ResNet101, and the four lines in each figure track the change of weights ω for each peer model in ResNet18 and ResNet101, respectively. As shown, the weights of Model4 and Model3

Table 7: Comparison with channel pruning methods for ResNet-32 and ResNet-56 on CIFAR10

Models	Method	Baseline Acc	Pruned Acc	Acc Drop↓	FLOPs Drop↓
ResNet32	WML_SNIP	92.63%	92.23%	0.40%	45.4%
	WML_SynFlow	92.63%	92.21%	0.42%	42.6%
	WML_GraSP	92.63%	90.67%	1.96%	44.8%
ResNet56	WML_SNIP	93.26%	93.46%	-0.20%	41.7%
	WML_SynFlow	93.26%	93.39%	-0.13%	43.8%
	WML_GraSP	93.26%	93.70%	0.56%	42.4%

Table 8: Top-1 accuracy (%) comparison results with additional knowledge distillation on CIFAR100.

Models	WML	FRSKD[26]	CS-KD[57]	TESKD [33]	LS-GDFD[11]	DDGSD[53]	BAKE[13]	MUSE[14]	FitNet[43]	AT[58]
Resnet18	79.38%	77.71%	77.26%	79.15%	77.04%	78.53%	78.72%	78.75 %	78.21%	78.54%
Resnet34	80.78%	77.58%	76.82%	79.60%	77.61%	79.25%	79.06%	80.11%	79.83%	79.05%

consistently increase since they contain much more parameters and with better generalization ability in ResNet18 and ResNet101. In contrast, the weights for the other two models both decrease. Results in ResNet18 are very similar to ResNet50, while ResNet101 presents a bit different results. We can find that Model3 gets the highest weight in ResNet101, while Model4 gets the highest weight for both ResNet18 and ResNet50, which contains the most parameters. This interesting phenomenon also suggests that moderate pruning of an overparameterized model could hardly affect its performance, but further improve its discriminative ability.

In addition, we also investigate the effects of channel pruning methods. In Table 7, we consider another two training-free network pruning methods, SynFlow [48] and GraSP [49], for the channel pruning in our WML, called as WML_SynFlow and WML_GraSP, respectively. The two pruning approaches are similar with SNIP, with only replace the parameter score for θ_i in to $\frac{\partial \mathcal{R}_{\text{SF}}}{\partial \theta_i} \odot \theta_i$ and $-(H \frac{\partial \mathcal{L}}{\partial \theta_i}) \odot \theta_i$, where H is Hessian matrix and $\mathcal{R}_{\text{SF}}$ is obtained by setting the input as the all-ones vector $\mathbf{1}$ and applying an element-wise absolute on all network parameters θ. Similarly, we also set a upper bound of the pruned ratio for each layer to avoid layer collapse as WML_SNIP. Table 7 compares the three different training-free pruning approaches in channel pruning for our WML, where we set the pruning ratio as about 40%. As shown, we find that WML_SynFlow achieves similar results as our WML_SNIP for both ResNet32 and ResNet56. In contrast, GraSP obtains much worse results, showing that an appropriate channel pruning method is also important to enhance the performance of our WML.

E More Comparison with Knowledge Distillation

In this subsection, we compare our WML with several recent knowledge distillation methods, including FitNet[43], AT[58], and LS-GDFD[11], and more recent self-distillation approaches, including FRSKD[26], CS-KD[57], TESKD [33],DDGSD[53], BAKE[13], and MUSE[14]. We conduct comparison experiments on CIFAR100 with ResNet18 and ResNet34, respectively. We report the results from the cited papers and [33]. We can see that our proposed WML outperforms these knowledge distillation methods, especially those recent self-distillation approaches with large margins, again verifying the effectiveness of our weighted mutual learning paradigm.