SEC-Prompt:SEmantic Complementary Prompting for Few-Shot Class-Incremental Learning

Supplementary Material

A. Additional Experimental Details

A.1. Dataset Details

We follow [25] and use the CIFAR100 [21], ImageNet-R [12], and CUB200-2011 [40] datasets. CIFAR100 consists of 100 classes, each with 500 training images and 100 testing images. The CUB200 dataset contains 11,788 high-resolution bird images categorized into 200 species, with 5,994 samples for training and 5,794 samples for testing. The ImageNet-R dataset is a subset of ImageNet that includes 30,000 images across 200 object categories, focusing on artistic renditions such as paintings, cartoons, sculptures, and sketches.

A.2. Additional Results

In the main paper, we presented the Top-1 accuracy of CUB200 and CIFAR100 in Figure 2. Here, we provide a detailed comparison on CUB200 in Table 4 and CIFAR100 in Table 5.

Table 4. Comparison with SOTA methods on CUB200 dataset for FSCIL.

Method	Acc. in each session ↑											PD ↓	HAcc ↑
	0	1	2	3	4	5	6	7	8	9	10		
TEEN [42]	**88.8**	86.2	**85.5**	83.0	83.0	81.7	81.5	79.7	79.9	79.5	80.0	8.8	80.2
PriViLege [30]	82.3	81.3	80.5	77.8	77.8	76.0	75.7	76.0	75.2	75.2	75.1	7.2	72.3
ASP [25]	87.1	86.0	84.9	83.4	83.6	82.4	82.6	83.0	82.6	83.0	83.5	3.6	83.4
L2P [44]	82.4	81.2	79.0	76.8	76.2	74.7	74.1	74.1	72.7	73.0	73.6	8.7	73.6
DualP [43]	83.5	82.2	80.9	79.5	78.6	77.0	76.3	77.0	75.7	76.1	76.5	7.0	76.3
Coda-P [34]	79.6	78.1	76.4	75.6	75.0	73.1	72.5	72.8	72.0	72.4	72.9	6.7	72.5
Ours-F	87.5	**86.6**	**85.5**	**84.6**	**84.9**	83.8	83.3	83.8	83.7	83.8	84.4	3.2	84.3
Ours	87.5	**86.6**	**85.5**	**84.6**	**84.9**	**84.0**	**83.7**	**84.2**	**83.8**	**84.0**	**84.8**	**2.6**	**84.7**

Table 5. Comparison with SOTA methods on CIFAR100 dataset for FSCIL.

Method	Acc. in each session ↑									PD ↓	HAcc ↑
	0	1	2	3	4	5	6	7	8		
TEEN [42]	**92.9**	90.2	88.4	86.8	86.4	86.0	85.8	85.1	84.0	8.9	81.2
PriViLege [30]	90.9	89.4	89.0	87.6	87.9	87.4	87.6	87.2	86.1	4.8	84.7
ASP [25]	92.2	90.7	90.0	88.7	88.7	88.2	88.2	87.8	86.7	5.5	85.3
L2P [44]	84.7	82.3	80.1	77.5	77.0	76.0	75.6	74.1	72.3	12.4	68.0
DualP [43]	86.0	83.6	82.9	80.2	80.6	80.2	80.5	79.0	77.4	8.5	75.3
Coda-P [34]	86.0	83.6	81.6	79.2	79.1	78.5	78.3	77.0	75.4	10.6	72.2
Ours-F	92.0	90.5	90.2	88.8	88.9	88.4	88.4	88.0	86.8	5.2	85.7
Ours	92.0	**90.8**	**90.6**	**89.1**	**89.2**	**89.1**	**89.2**	**88.7**	**87.5**	**4.5**	**87.1**

Our SEC-Prompt framework achieves the best PD and HAcc on both datasets. Specifically, on CUB200, SEC-Prompt achieves a PD of 2.6% and an HAcc of 24.7%, surpassing the second-best method by 1.0% and 1.3%, respectively. On CIFAR100, SEC-Prompt achieves a PD of 4.5% and an HAcc of 87.1%, outperforming the second-best method by 0.3% and

1.8%, respectively. The performance advantage of SEC-Prompt on CUB200 and CIFAR100 is less pronounced compared to ImageNet-R. This is likely due to the closer distributional similarity of these datasets to ImageNet, which enables the pretrained ViT generalization capability to sufficiently balance stability and plasticity. Conversely, ImageNet-R, with its diverse styles such as cartoons, sculptures, and paintings, poses greater learning challenges and highlights the significance of non-discriminative information.

Method	Trainable param	Final param
TEEN [42]	100	100
PriViLege [30]	16.3	102.3
ASP [25]	3.0	103.0
Ours-F	1.1	101.1
Ours	1.5	101.5

(a) CIFAR100

Method	Trainable param	Final param
TEEN [42]	100	100
PriViLege [30]	16.3	102.3
ASP [25]	3.0	103.0
Ours-F	1.7	101.7
Ours	2.8	102.8

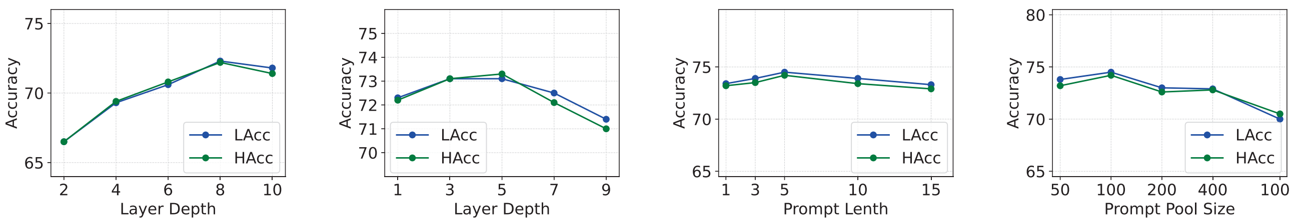
(b) CUB200

Table 6. Comparison of parameter effectiveness with SOTA FSCIL method. Trainable parameters and Final parameters represent the percentages of trainable and total parameters relative to the base ViT pre-trained model.

Additionally, as shown in Table 6b and Table 6a, we report the parameter efficiency analysis on CUB200 and CIFAR100. Notably, the parameter gap between SEC-Prompt and SOTA methods narrows on CUB200, leading to a reduced difference in base class accuracy as well.

B. Additional Implementation Details

In the main paper, we introduced the key experimental details. In this section, we provide an analysis of the selection of parameters, including the prompt insertion depth of the network, prompt length, and prompt pool size. In Figure 6, we present the results of our parameter selection experiments conducted on ImageNet-R, where LAcc represents the accuracy of the final session, and HAcc denotes the harmonic accuracy.



(a) Depth of D-Prompt (b) Depth of ND-Prompt (c) Prompt Lenth (d) Prompt Pool Size

Figure 6. Selection of prompt parameters, including Layer Depth, prompt length, and prompt pool size. LAcc represents the accuracy of the last session, while HAcc stands for Harmonic Accuracy.

B.1. Decoupled prompt positions

Visual Prompt Tuning [15] involves concatenating the prompt with the input of a specified layer in the backbone to fine-tune the model. Since different layers of the backbone exhibit varying levels of feature abstraction, we conduct experiments to determine the optimal layer for prompt insertion. Specifically, as in [34, 43], we begin by inserting the prompt at Layer_1 and search for the optimal final layer for prompt insertion. Discriminative information is the dominant feature in classification tasks, while non-discriminative information assists the classification task by complementing the discriminative features. Therefore, we first select the insertion layer for D-Prompt independently, and then, based on this, search for the optimal insertion layer for ND-Prompt. As shown in Figures 6a and 6b, D-Prompt achieves the highest accuracy at Layer 8, while ND-Prompt achieves the highest accuracy at Layer 5, aligning with the motivation behind the design of SEC-Prompt. D-Prompt captures high-level discriminative features, whereas ND-Prompt captures low-level non-discriminative features. Similar to findings in [43], we observe that model performance decreases with increasing depth. Specifically, while base class accuracy improves, new class accuracy declines. We hypothesize that this is due to the semantic information in deep features causing overfitting, which adversely affects the model's generalization ability.

B.2. Prompt Lenth

Prompt Length refers to the number of prompt tokens selected from the prompt pool for each instance query. As shown in 6c, increasing the prompt length initially improves accuracy but no longer contributes once the accuracy peaks, consistent with the findings in [34]. To achieve the optimal balance between parameter efficiency and model performance, we set the prompt length to 5.

B.3. Prompt Pool Size

Coda-Prompt [34] learns task-specific prompts added to the prompt pool, selecting prompt pool sizes arbitrarily to validate model performance and parameter efficiency. In contrast, our SEC-Prompt framework divides prompts into D-Prompt and ND-Prompt to learn semantic feature knowledge, making it class-dependent. Thus, we select prompt pool sizes proportional to the number of classes. As shown in Figure 6d, for ImageNet-R with 200 classes, we choose [50, 100, 200, 400, 1000] as prompt pool sizes and set the prompt pool size to 100, allocating one prompt component for every two classes. Unlike Coda-Prompt, our model does not continuously benefit from increasing the prompt pool size, instead, performance declines after reaching its peak. We observe that a larger prompt pool size results in more severe forgetting. We hypothesize that while our method enhances learning capability in few-shot scenarios, the quality of learned knowledge for new classes remains suboptimal. A larger prompt pool size amplifies the impact on old knowledge, leading to greater forgetting. This raises an important future research question: how to further improve the quality of features learned in few-shot scenarios.