Semantic Prompting with Image-Token for Continual Learning - Supplementary Materials -

Jisu Han[1], Jaemin Na[2*], and Wonjun Hwang[1*]
[1]Ajou University, Korea, [2]Tech. Innovation Group, KT, Korea
{jisu3709, wjhwang}@ajou.ac.kr, jaemin.na@kt.com

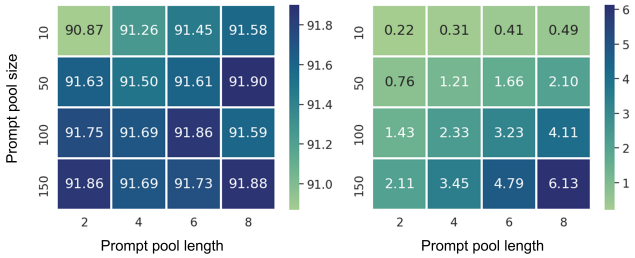

Figure 1. **Hyperparameter analysis.** Result of the grid search for prompt pool size and length, **Left:** average accuracy ↑ (%), **Right:** tuning parameter ratio ↓ (%).

Table 1. **Variance on accuracy.** Average and standard deviation of accuracy for 5 trials

Method	CIFAR-100 B0-Inc10		ImageNet-R B0-Inc20	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	88.09 ± 1.01	82.82 ± 1.52	77.66 ± 0.73	72.36 ± 0.67
DualPrompt	86.44 ± 1.16	80.94 ± 1.27	74.72 ± 0.29	68.37 ± 0.72
CODA-Prompt	90.78 ± 0.72	86.36 ± 0.78	79.77 ± 0.62	74.99 ± 0.41
I-Prompt (Ours)	$\mathbf{91.63 \pm 0.39}$	$\mathbf{87.11 \pm 0.44}$	$\mathbf{80.21 \pm 0.78}$	$\mathbf{75.43 \pm 0.44}$

A. Hyperparameter analysis

Figure 1 presents the average accuracy and number of training parameters obtained through grid search with prompt pool sizes [10, 50, 100, 150] and prompt pool lengths [2, 4, 6, 8]. The hyperparameters of our method include the prompt pool size and prompt length. The prompt pool size represents the total number of prompts, and the prompt length denotes the number of prompts that are added to the image token. The prompt is separately applied to the self-attention key and value, making the prompt pool length twice the prompt length. To strike a balance between average accuracy and the number of training parameters, we empirically selected a prompt pool size of 100 and a prompt length of 2. Our method also achieves a high performance of 90.87% with only 0.22% of the training parameters, similar to those of L2P [7] and DualPrompt [6].

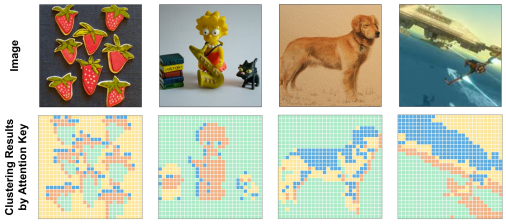

Figure 2. **clustering result. Above:** Input images, **Below:** Image token clustering results for first layer embedding.

Table 2. **Effect of Prompt location.** Average accuracy (%) comparison based on prompt location on CIFAR-100 B0-Inc10.

Lower layers (1-5)	Middle layers (4-8)	Higher layers (8-12)
91.75	91.62	87.98

B. Variance on accuracy

We conducted our experiment following previous works [3, 5, 10] with a random seed of 1993. In Table 2, we present the results of five trials using shuffled class orders with random seeds from {0, 10, 20, 30, 40}. We achieved the highest average performance on both CIFAR-100 and ImageNet-R across five trials. Additionally, we achieved the lowest variance in CIFAR-100 and the second-lowest variance in final accuracy for ImageNet-R.

C. Effect of prompt location

Previous studies [2, 9] demonstrate that the representation capability of deep learning models is more pronounced in higher layers. Therefore, while applying prompts to higher layers might seem appropriate to leverage the representation of model, we show through empirical investigation and Figure 2 that the pre-trained model also possesses significant representation and classification capabilities in lower layers. Since prompts change the overall output by altering the input or layers, we apply prompts to the lower layers for plasticity. we present the performance of the prompts according to their application location in Table 2.

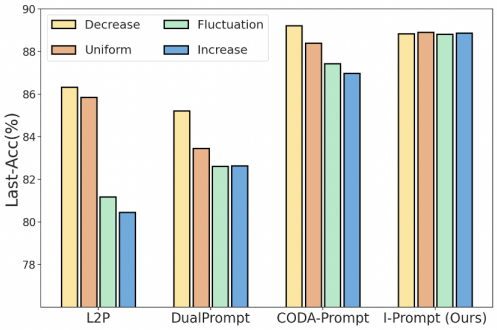

Figure 3. **Performance on various task distribution.** We report the final accuracy in the uniform setting, representing the task-balanced scenario, and in three distinct task-imbalanced scenarios.

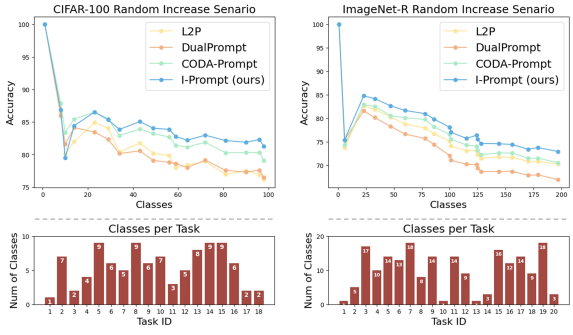

Figure 4. **Task-wise accuracy in random increase scenario.** We report the performance of the prompt-based method in a random increase scenario. The line plot and bar plot show the accuracy and the distribution of classes per task, respectively.

D. Various task distribution

In Figure 3, we present the last accuracy across a variety of task distributions to demonstrate that our task-agnostic method consistently provides robust performance for each distribution. We set up the experiments with four different task distributions: decreasing, uniform, increasing, and fluctuating, according to the number of classes per task. The uniform distribution has a uniform number of classes per task and is equal to B0-Inc20. The decreasing distribution has a decreasing number of classes per task and is equal to B0-Inc(30-5t). The increasing distribution has classes per task increase and is equal to B0-Inc(10+5t). Finally, the fluctuating distribution has increasing and decreasing classes per task and has 10,30,5,40,15 classes for each task. In experiments with uniform and decreasing task distributions, L2P [7], DualPrompt [6], and CODA-Prompt [4] demonstrate high performance levels; however, their effectiveness diminishes in scenarios involving increasing or fluctuating distributions. It is highly optimized for the initial task during the prompt selection process, resulting in high performance when the number of initial training classes is large, but due to task dependency, the final accuracy changes significantly as the task changes. On the other hand, the proposed method showed equivalent performance in all experiments. This shows that our method is not task-dependent by fully exploits information about the classes, making it task agnostic.

E. Random increase scenario

As a further task imbalance scenario, we report the task-wise accuracy of recent continual learning methods in Figure 4 for a random increase scenario with dynamically changing incremental steps. Each task is assigned a number of classes generated from the same random seed, and the histogram below shows the distribution of classes per task. Unlike the fluctuation scenario, it does not ensure fluctuations, but is randomly seeded to validate robustness. In experiments on random increase scenario, we observe a trend that as the number of tasks increases, the performance gap between our method and the comparison methods also widens. This trend highlights the effectiveness of our task-agnostic approach, particularly in addressing the existing challenge where task prediction becomes more difficult with an increasing number of tasks.

F. Task-wise accuracy

We further report the task-wise accuracy of recent continual learning methods [4, 6, 7] in Figure 5. In this experiment, we observe a trend where as the number of tasks increases, the performance gap between our method and the comparison methods also widens. This trend shows that we achieve our goal of addressing the performance decrease as task prediction becomes more difficult. Moreover, since the performance gain grows with the number of tasks, our approach has a large advantage in final accuracy over average accuracy.

G. Effect of pre-trained model

We show the results of the ViT-B/16 pre-trained on ImageNet-21k in Tables 3, 4 and the results of the ViT-L/16 in Tables 5, 6 to confirm the performance improvement according to the architecture of the pre-trained model and the pre-training data. Our method achieves the highest performance in most results even with models utilizing extensive pre-training data and larger models. We observe consistent trends, similar to previous experiments, showing significant performance gains in scenarios with many tasks and task imbalanced scenarios. By experimenting with variations in pre-trained models, we demonstrate consistent performance improvements of the proposed method from both the model and data perspectives.

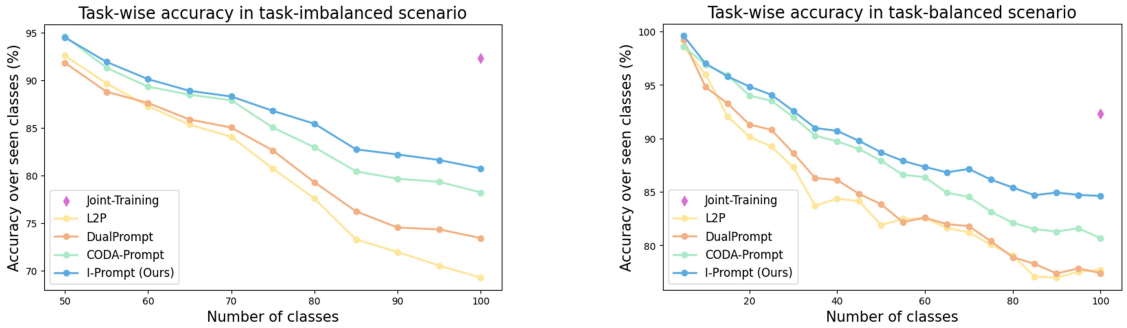

Figure 5. **Task-wise accuracy on CIFAR-100.** We report the performance of prompt-based methods in task-imbalanced and balanced scenarios. The mask with purple diamonds represents the upper bound of performance achieved through joint-training.

Table 3. **Comparison results (%) in task-imbalanced scenario on ImageNet-R and CIFAR-100.** Average and final accuracy on a ViT-B/16 pre-trained on **ImageNet-21k**. The best performance is in bold.

Method	ImageNet-R						CIFAR-100					
	B100-Inc5		B100-Inc10		B100-Inc20		B50-Inc2		B50-Inc5		B50-Inc10	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	61.73	49.95	66.49	57.55	71.37	64.38	66.95	49.92	81.89	71.55	87.85	82.68
DualPrompt	56.77	43.92	62.35	51.78	68.15	61.80	69.72	52.24	83.79	74.94	88.14	83.60
CODA-Prompt	66.12	58.69	68.49	61.00	72.55	67.23	71.24	56.89	82.34	72.52	88.55	82.49
I-Prompt (Ours)	**69.05**	**59.56**	**73.78**	**68.07**	**76.68**	**73.07**	**72.64**	**59.25**	**84.87**	**78.23**	**90.21**	**86.49**

Table 4. **Comparison results (%) in task-balanced scenario on ImageNet-R and CIFAR-100.** Average and final accuracy on a ViT-B/16 pre-trained on **ImageNet-21k**.

Method	ImageNet-R						CIFAR-100					
	B0-Inc10		B0-Inc20		B0-Inc40		B0-Inc5		B0-Inc10		B0-Inc20	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	74.38	67.80	76.07	69.85	77.29	72.90	85.00	78.16	89.53	84.95	90.93	86.67
DualPrompt	69.21	62.27	71.68	66.60	73.24	69.25	87.15	79.77	89.98	84.72	90.65	86.81
CODA-Prompt	74.97	68.10	79.30	73.28	79.76	74.70	87.58	80.06	91.46	86.74	92.79	88.91
I-Prompt (Ours)	**76.88**	**69.47**	**79.58**	**73.38**	**80.53**	**75.65**	**89.76**	**83.31**	**92.33**	**88.04**	**93.07**	**89.55**

Table 5. **Comparison results (%) in task-imbalanced scenario on ImageNet-R and CIFAR-100.** Average and final accuracy on a **ViT-L/16** pre-trained on ImageNet-1K.

Method	ImageNet-R						CIFAR-100					
	B100-Inc5		B100-Inc10		B100-Inc20		B50-Inc2		B50-Inc5		B50-Inc10	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	69.93	60.32	73.92	67.02	77.19	72.65	71.29	54.29	85.20	77.46	90.26	86.33
DualPrompt	67.20	59.87	71.35	65.97	74.25	70.25	76.43	64.80	87.24	82.54	89.97	87.15
CODA-Prompt	74.71	68.68	78.70	74.47	80.46	77.40	80.57	68.91	87.97	81.87	91.94	88.65
I-Prompt (Ours)	**75.63**	**70.03**	**79.69**	**76.45**	**81.74**	**79.32**	**80.89**	**69.90**	**90.55**	**86.69**	**92.75**	**90.34**

Table 6. **Comparison results (%) in task-balanced scenario on ImageNet-R and CIFAR-100.** Average and final accuracy on a **ViT-L/16** pre-trained on ImageNet-1K.

Method	ImageNet-R						CIFAR-100					
	B0-Inc10		B0-Inc20		B0-Inc40		B0-Inc5		B0-Inc10		B0-Inc20	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	80.88	75.53	81.32	76.53	81.64	77.43	89.53	84.39	91.97	88.07	92.41	89.10
DualPrompt	77.02	70.78	77.84	72.18	77.65	73.90	87.89	82.74	90.05	85.36	91.33	87.48
CODA-Prompt	81.59	76.32	**84.52**	**79.83**	84.31	80.42	90.49	84.82	93.20	88.96	**94.04**	**90.86**
I-Prompt (Ours)	**82.39**	**77.05**	84.08	79.60	**84.60**	**80.80**	**91.40**	**86.51**	**93.65**	**89.87**	**94.04**	90.82

Table 7. **Results (%) on Large-scale datasets.** Landmark-v2-1k indicates a subset of the Google landmark dataset v2, consisting of 1000 randomly selected classes.

Method	Landmark-v2-1k B0-Inc100		Landmark-v2-1k B500-Inc100	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	70.63	63.09	74.08	69.98
DualPrompt	73.51	65.33	75.07	70.03
CODA-Prompt	76.85	69.60	77.30	73.45
I-Prompt	**78.37**	**71.63**	**80.51**	**76.42**
Method	iNaturalist-19 B0-Inc100		iNaturalist-19 B500-Inc100	
	Avg-Acc	Last-Acc	Avg-Acc	Last-Acc
L2P	60.11	61.43	67.16	66.44
DualPrompt	58.01	57.18	64.77	62.70
CODA-Prompt	62.80	62.89	62.79	61.21
I-Prompt	**66.75**	**66.20**	**70.74**	**69.58**

H. Results on Large-scale datasets

Prompt-based methods leverage pre-trained knowledge to adapt to new tasks with a small number of parameters. We present experiments on a subset of the Google Landmark Dataset v2 [8] and iNaturalist 2019 [1], consisting of 1000 classes, to verify that these methods work effectively on large-scale datasets in Table 7. Our experiments show that it is still effective for datasets with a large number of classes, achieving the best performance on both 2 datasets and 4 experiments.

- **Google Landmarks Dataset v2** is a large-scalse dataset for fine-grained instance recognition and image retrieval containing various landmarks and famous places around the world and contains 200k classes. The training and test sets are divided into 4.1M and 118k images respectively, and we use a subset of 1,000 classes for our experiments.
- **iNaturalist 2019** is a subset of iNaturalist introduced at the 2019 CVPR Fine-Grained Visual Categorisation Workshop and consists of 1010 classes containing various species of plants, animals, insects, etc. observed in nature. The train set consists of 265,213 images and has no annotations for the test set, therefore we split the train set 8:2 to construct the training and test data.

I. Algorithm for I-Prompt

We demonstrate the procedure for continual learning and the operation of I-Prompt in Algorithms 1 and 2.

Algorithm 1 Continual learning procedure

Input: training set D^t, test set $\mathcal{D}^{1:t}$, total tasks T, training epochs E, pre-trained model parameters θ, classifier parameters ϕ, prompt key k, prompt value P, learning rate α
Output: θ, k, P
for t in $\{1, \cdots, T\}$ **do**
  for e in $\{1, \cdots, E\}$ **do**
    Sample a batch $(x, y) \sim D^t$
    Forward pass through the model θ with prompt
    Compute gradient $\nabla \mathcal{L}_{cls}$ (see Eq. (9))
    Update parameters $[k, P, \phi] \leftarrow [k, P, \phi] - \alpha \nabla \mathcal{L}_{cls}$
  end for
  Evaluate task accuracy on test set $D^{1:t}$
end for
Evaluate last accuracy on test set $D^{1:T}$

Algorithm 2 I-Prompt Algorithm

Input: input image x, pre-trained model parameters θ, classifier parameters ϕ, prompt key k, prompt value P, model layers L, tuning layers I
Output: prediction $\hat{y}$
for l in $\{1, \cdots, I\}$ **do**
  Forward through intermediate layers: $h^l = f(x, \theta^l)$
  Calculate token-key similarity: $\gamma(h^l, k)$
  Prompt matching (see Eq. (5))
  Add prompt to image token embedding (see Eq. (8))
end for
for l in $\{I+1, \cdots, L\}$ **do**
  Forward through attention layers: $h^l = f(x, \theta^l)$
end for
Prediction: $\hat{y} = \phi^T h^L$

References

[1] Yin Cui, Yang Song, Chen Sun, Andrew Howard, and Serge Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4109–4118, 2018. 4

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012. 1

[3] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1

[4] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11909–11919, 2023. 2

[5] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *In Proceedings of the European conference on computer vision*, pages 398–414. Springer, 2022. 1

[6] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *In Proceedings of the European Conference on Computer Vision*, pages 631–648. Springer, 2022. 1, 2

[7] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 1, 2

[8] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2575–2584, 2020. 4

[9] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, pages 818–833. Springer, 2014. 1

[10] Da-Wei Zhou, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *arXiv preprint arXiv:2303.07338*, 2023. 1