Towards a General Framework for Continual Learning with Pre-training

Liyuan Wang[1], **Jingyi Xie**[1], **Xingxing Zhang**[1]*, **Hang Su**[1], **Jun Zhu**[1]*

Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, THBI Lab,
Tsinghua-Bosch Joint Center for ML, Tsinghua University, Beijing, China.
wly19@tsinghua.org.cn, jingyi_xie96@163.com
xxzhang1993@gmail.com, {suhangss, dcszj}@tsinghua.edu.cn

Abstract

In this work, we present a general framework for continual learning of sequentially arrived tasks with the use of pre-training, which has emerged as a promising direction for artificial intelligence systems to accommodate real-world dynamics. From a theoretical perspective, we decompose its objective into three hierarchical components, including within-task prediction, task-identity inference, and task-adaptive prediction. Then we propose an innovative approach to explicitly optimize these components with parameter-efficient fine-tuning (PEFT) techniques and representation statistics. We empirically demonstrate the superiority and generality of our approach in downstream continual learning, and further explore the applicability of PEFT techniques in upstream continual learning. We expect this to provide an important technical foundation for intrinsically motivated open-ended learning. Our code is available at https://github.com/thu-ml/HiDe-Prompt.

1 Introduction

To cope with real-world dynamics, continual learning has received widespread attention, especially in the context of pre-training. Through adapting the pre-trained knowledge effectively to downstream tasks, it provides not only positive knowledge transfer but also robustness to catastrophic forgetting [10, 8, 16, 20]. An emerging direction is the implementation of parameter efficient fine-tuning (PEFT) techniques (e.g., Prompt [3], Adapter [11], LoRA [2], FiLM [9], etc.), which usually freeze a pre-trained transformer backbone and employ additionally a few parameters to steer representation learning. In particular, recent prompt-based approaches [19, 18, 17, 12, 15] focus on *construction* and *inference* of appropriate prompts for each task, and achieve outstanding performance under strong supervised pre-training. However, existing methods usually degrade in performance with challenges in upstream knowledge (e.g., different pre-training paradigms) and downstream tasks (e.g., out-of-distribution and fine granularity), with generality left to be desired [15].

In this work, we provide an in-depth theoretical analysis of the continual learning objective in the context of pre-training, which can be decomposed into hierarchical components such as *within-task prediction*, *task-identity inference* and *task-adaptive prediction*. By leveraging the well-distributed pre-trained representations, we then propose an innovative approach applicable to various PEFT techniques to optimize explicitly the hierarchical components. We perform extensive experiments on downstream continual learning to demonstrate the superiority and generality of our approach, and further explore the applicability of PEFT techniques in upstream continual learning. We also provide neuroscience insights into the proposed framework for acquisition of open-world knowledge.

*Corresponding authors.

Preprint. Under review.

2 Hierarchical Decomposition of Continual Learning Objective

Continual learning aims to master a sequence of tasks represented by their respective training sets $\mathcal{D}_1, ..., \mathcal{D}_T$ and excel on their corresponding test sets. Each training set $\mathcal{D}_t = \{(\boldsymbol{x}_{t,n}, y_{t,n})\}_{n=1}^{N_t}$, where $|\mathcal{D}_t| = N_t$ denotes the size of $\mathcal{D}_t$. $\boldsymbol{x}_{t,n} \in \mathcal{X}_t$ and $y_{t,n} \in \mathcal{Y}_t$ indicate the sample and label elements, respectively. Consider a neural network model with a backbone f_θ parameterized by θ, and an output layer h_ψ parameterized by ψ. This model seeks to learn the projection from $\mathcal{X} = \bigcup_{t=1}^{T} \mathcal{X}_t$ to $\mathcal{Y} = \bigcup_{t=1}^{T} \mathcal{Y}_t$, aiming to predict the label $y = h_\psi(f_\theta(\boldsymbol{x})) \in \mathcal{Y}$ of an unseen test sample $\boldsymbol{x}$ drawn from previous tasks. The backbone function f_θ is assumed to be pre-trained with a substantial quantity of additional training samples external to each $\mathcal{D}_t$. There are commonly three distinct settings for continual learning [13]: task-, domain-, and class-incremental learning (TIL, DIL, and CIL). Specifically, $\mathcal{Y}_1, ..., \mathcal{Y}_T$ are identical for DIL while disjoint for TIL and CIL. The task identity is provided for TIL at test time but is not available for DIL and CIL.

Here we take CIL as a typical scenario for theoretical analysis, where $\mathcal{Y}_t \cap \mathcal{Y}_{t'} = \emptyset, \forall t \neq t'$ (see Appendix A for DIL and TIL). Let $\mathcal{X}_t = \bigcup_j \mathcal{X}_{t,j}$ and $\mathcal{Y}_t = \{\mathcal{Y}_{t,j}\}$, where $j \in \{1, ..., |\mathcal{Y}_t|\}$ indicates the j-th class in task t. Now assume we have a ground event denoted as $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_t\}$ and a pre-trained model f_θ. For any sample $\boldsymbol{x} \in \bigcup_{k=1}^{t} \mathcal{X}_k$, a general goal of the CIL problem is to learn $P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D}, \theta)$, where $i \in \{1, ..., t\}$ and $j \in \{1, ..., |\mathcal{Y}_i|\}$. This can be decomposed into two probabilities, including task-identity inference (TII) and within-task prediction (WTP), denoted as $P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)$ and $P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta)$, respectively. Based on Bayes' theorem, we have

$$P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D}, \theta) = P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta) P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta). \tag{1}$$

Let $\bar{i} \in \{1, ..., t\}$ and $\bar{j} \in \{1, ..., |\mathcal{Y}_i|\}$ be the ground truth of an $\boldsymbol{x}$ w.r.t. the task identity and within-task index. Eq. (1) shows that if we can improve either the WTP performance $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)$, the TII performance $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}}|\mathcal{D}, \theta)$, or both, then the CIL performance $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D}, \theta)$ would be improved. However, such an improvement is limited since it is upper-bounded by WTP or TII. To further improve the CIL performance, we propose a hierarchical decomposition of its objective. That is, besides the improvement of $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D}, \theta)$, we also need to improve the performance of task-adaptive prediction (TAP), denoted as $P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta)$, where $\mathcal{X}^y$ represents the domain of class y in all previous tasks, and $y = \mathcal{Y}_{\bar{i},\bar{j}}$ is the ground truth label of $\boldsymbol{x}$. Then the final goal of CIL is formulated as a multi-objective optimization problem, i.e., $\max[P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D}, \theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta)]$. Notice that the WTP probability is a categorical distribution over all observed tasks $\{1 : t\}$, while the TAP probability is over all observed classes $\bigcup_{k=1}^{t} \mathcal{Y}_k$.

To resolve the problems above, we derive the sufficient and necessary conditions in the context of the widely-used cross-entropy loss.[2] Specifically, we define

$$H_{\mathrm{WTP}}(\boldsymbol{x}) = \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},j}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)\}_j), \tag{2}$$

$$H_{\mathrm{TII}}(\boldsymbol{x}) = \mathcal{H}(\mathbf{1}_{\bar{i}}, \{P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)\}_i), \tag{3}$$

$$H_{\mathrm{TAP}}(\boldsymbol{x}) = \mathcal{H}(\mathbf{1}_{\bar{c}}, \{P(\boldsymbol{x} \in \mathcal{X}^c|\mathcal{D}, \theta)\}_c), \tag{4}$$

where H_{WTP}, H_{TII}, and H_{TAP} are the cross-entropy values of WTP, TII, and TAP, respectively. The operation $\mathcal{H}(p, q) \triangleq -\mathbb{E}_p[\log q] = -\sum_i p_i \log q_i$. $\mathbf{1}.$ is a one-hot encoding function.

We now present two important theorems under the CIL scenario (see Appendix A for a detailed proof, as well as the counterpart of DIL and TIL), corresponding to the sufficient and necessary conditions for improving continual learning. First, Theorem 1 suggests that the good performances of WTP, TII and TAP are sufficient to guarantee a good performance of CIL:

Theorem 1 *For continual learning with pre-training, if* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \leq \delta$, $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] \leq \epsilon$, *and* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})] \leq \eta$, *we have the loss error* $\mathcal{L} \in [0, \max\{\delta + \epsilon, \eta\}]$, *regardless whether WTP, TII and TAP are trained together or separately.*

Then, Theorem 2 suggests that if a continual learning model is well trained (i.e., with low loss), then the WTP, TII and TAP for sequential tasks are always implied to be small:

Theorem 2 *For continual learning with pre-training, if the loss error* $\mathcal{L} \leq \xi$, *then there always exist (1) a WTP, s.t.* $H_{\mathrm{WTP}} \leq \xi$; *(2) a TII, s.t.* $H_{\mathrm{TII}} \leq \xi$; *and (3) a TAP, s.t.* $H_{\mathrm{TAP}} \leq \xi$.

[2] Note that our framework is applicable to TIL, DIL and CIL scenarios, detailed in Appendix A.

3 Optimization of Hierarchical Components

Motivated by these theoretical insights, we propose to optimize explicitly the hierarchical components (i.e., WTP, TII and TAP) for continual learning with pre-training. Our proposal stems from two particular advantages of pre-training: (1) the representations can be effectively adapted to downstream tasks through PEFT techniques, and (2) the distributions of unadapted and adapted representations (denoted as $\hat{\mathcal{G}}_c$ and $\mathcal{G}_c$ for each class $c \in \mathcal{Y}_i, i = 1, ...t-1$, respectively) can be effectively preserved through their statistical information. For efficiency and generality, here we employ multiple centroids obtained from K-Nearest Neighbor (KNN) and add Gaussian noise as a specific implementation.

First, we improve **WTP** through effectively incorporating task-specific knowledge from each $\mathcal{D}_t$. Specifically, we construct task-specific parameters $\boldsymbol{e}_t$ with a PEFT technique (e.g., Prompt [3], Adapter [11], LoRA [2], FiLM [9], etc.), and optimize H_{WTP} with cross-entropy (CE). $\boldsymbol{e}_1, ..., \boldsymbol{e}_{t-1}$ are frozen to avoid catastrophic forgetting, while $\boldsymbol{e}_t$ is initialized with $\boldsymbol{e}_{t-1}$ to transfer knowledge. Besides, the adapted representations of $\boldsymbol{e}_t$, although allowing the new task to be performed well, may overlap with that of the old tasks and thus affect TAP. To overcome this issue, we preserve statistics of adapted representations collected by f_θ and $\boldsymbol{e}_i, i = 1, ..., t-1$, where for classification we calculate the mean $\boldsymbol{\mu}_c$ of $\mathcal{G}_c$ for each class $c \in \mathcal{Y}_i$, and design a *contrastive regularization* (CR):

$$\mathcal{L}_{\mathrm{CR}}(\boldsymbol{e}_t) = \sum_{\boldsymbol{h} \in \mathcal{H}_t} \frac{1}{\sum_{i=1}^{t-1} |\mathcal{Y}_i|} \sum_{i=1}^{t-1} \sum_{c \in \mathcal{Y}_i} \log \frac{\exp(\boldsymbol{h} \cdot \boldsymbol{\mu}_c / \tau)}{\sum_{\boldsymbol{h}' \in \mathcal{H}_t} \exp(\boldsymbol{h} \cdot \boldsymbol{h}' / \tau) + \sum_{i=1}^{t-1} \sum_{c \in \mathcal{Y}_i} \exp(\boldsymbol{h} \cdot \boldsymbol{\mu}_c / \tau)}, \tag{5}$$

where $\mathcal{H}_t$ is the embedding transformation of $\mathcal{D}_t$ with f_θ and $\boldsymbol{e}_t$. τ is the temperature coefficient, which is insensitive and set to 0.8 in practice. Then, the loss function of WTP can be defined as

$$\mathcal{L}_{\mathrm{WTP}}(\psi, \boldsymbol{e}_t) = \mathcal{L}_{\mathrm{CE}}(\psi, \boldsymbol{e}_t) + \lambda \mathcal{L}_{\mathrm{CR}}(\boldsymbol{e}_t). \tag{6}$$

Therefore, the adapted representations of new classes can be well distinguished for WTP while avoiding overlap with the previous ones. λ is a hyperparamter to balance the impact of old classes.

Second, we improve **TII** and **TAP** through leveraging the approximated distributions of unadapted and adapted representations, respectively. For H_{TII}, we construct an auxiliary output layer $\hat{h}_\omega : \mathbb{R}^D \rightarrow \mathbb{R}^T$ parameterized by ω, learning explicitly the projection from unadapted representations to task identity via cross-entropy:

$$\mathcal{L}_{\mathrm{TII}}(\omega) = \frac{1}{\sum_{i=1}^{t} |\mathcal{Y}_i|} \sum_{i=1}^{t} \sum_{c \in \mathcal{Y}_i} \sum_{\hat{\boldsymbol{h}} \in \hat{\mathcal{H}}_{i,c}} - \log \frac{\exp(\hat{h}_\omega(\hat{\boldsymbol{h}})[i])}{\sum_{j=1}^{t} \exp(\hat{h}_\omega(\hat{\boldsymbol{h}})[j])}, \tag{7}$$

where $\hat{\mathcal{H}}_{i,c}$ is constructed by sampling an equal number of pseudo representations from $\hat{\mathcal{G}}_c$ for $c \in \mathcal{Y}_i$ and $i = 1, ..., t$. Similarly, the final output layer $h_\psi : \mathbb{R}^D \rightarrow \mathbb{R}^{|\mathcal{Y}|}$ is further optimized for H_{TAP}:

$$\mathcal{L}_{\mathrm{TAP}}(\psi) = \frac{1}{\sum_{i=1}^{t} |\mathcal{Y}_i|} \sum_{i=1}^{t} \sum_{c \in \mathcal{Y}_i} \sum_{\boldsymbol{h} \in \mathcal{H}_{i,c}} - \log \frac{\exp(h_\psi(\boldsymbol{h})[c])}{\sum_{j=1}^{t} \sum_{c' \in \mathcal{Y}_j} \exp(h_\psi(\boldsymbol{h})[c'])}, \tag{8}$$

where $\mathcal{H}_{i,c}$ is constructed by sampling an equal number of pseudo representations from $\mathcal{G}_c$ for $c \in \mathcal{Y}_i$ and $i = 1, ..., t$. As ω and ψ are usually *light-weight*, the optimization of TII and TAP is computationally efficient. At test time, our approach predicts the task identity $i = \hat{h}_\omega(f_\theta(\boldsymbol{x}))$ and then the label $y = h_\psi(f_\theta(\boldsymbol{x}; \boldsymbol{e}_i))$. Please refer to Appendix Algorithm 1 for more details.

4 Experiment

Experimental Setup: We consider two CIL benchmarks that are widely used for *downstream continual learning* [19, 18, 12], such as Split ImageNet-R [5] of 200-class natural images and Split CUB-200 [14] of 200-class bird images, randomly split into 10 incremental tasks. After learning multiple incremental tasks, we further evaluate *upstream continual learning* with the ability of few-shot learning, i.e., adapting the backbone to a N-way K-shot task [1] randomly sampled from subsequent unseen classes. We consider supervised and self-supervised pre-training on ImageNet-21K, denoted as Sup-21K and iBOT-21K, respectively. Our implementation is detailed in Appendix B.

Experimental Result: We implement two representative PEFT techniques as the task-specific parameters in our approach, such as Prompt [3] (adjusting intermediate inputs through prepending a short sequence of learnable prompt parameters) and LoRA [2] (adjusting backbone parameters

Table 1: Performance of downstream continual learning. PTM: pre-trained model. FAA: final average accuracy. CAA: cumulative average accuracy. FFM: final forgetting measure.

PTM	Method	Split ImageNet-R			Split CUB-200		
		FAA (↑)	CAA (↑)	FFM (↓)	FAA (↑)	CAA (↑)	FFM (↓)
Sup-21K	L2P [19]	63.65	67.25	7.51	75.58	80.32	6.38
	DualPrompt [18]	68.79	71.96	4.49	81.32	83.45	5.31
	S-Prompt [17]	69.68	72.50	3.29	81.51	83.24	4.48
	CODA-Prompt [12]	70.03	74.26	5.17	74.34	80.71	7.42
	Ours-Prompt	73.55	75.93	0.95	84.60	83.87	0.21
	Ours-LoRA	69.59	74.18	8.68	85.26	86.56	3.58
iBOT-21K	L2P [19]	55.35	58.62	3.73	45.93	56.02	9.20
	DualPrompt [18]	54.55	58.69	5.38	41.46	54.57	14.03
	S-Prompt [17]	55.16	58.48	4.07	39.88	53.71	13.15
	CODA-Prompt [12]	61.22	66.76	9.66	47.79	59.24	11.81
	Ours-Prompt	70.63	72.94	1.31	72.27	73.66	1.94
	Ours-LoRA	70.94	74.92	5.61	71.75	76.57	5.33

Table 2: Performance of upstream continual learning. After learning 8 incremental tasks, we present the accuracy of learning each N-way K-shot (NWKS) task sampled from subsequent classes.

PTM	Method	Split ImageNet-R		Split CUB-200		Split CUB & Cars	
		5W1S (↑)	5W5S (↑)	5W1S (↑)	5W5S (↑)	5W1S (↑)	5W5S (↑)
Sup-21K	Ours-Prompt	49.88	67.88	77.50	80.63	65.87	82.13
	Ours-LoRA	67.00	82.88	79.50	92.88	71.87	86.93
iBOT-21K	Ours-Prompt	42.87	64.63	36.50	62.88	34.87	58.93
	Ours-LoRA	58.38	78.87	53.63	79.75	40.60	68.40

through adding a learnable low-rank parameter matrix). We first evaluate the performance of *downstream continual learning* with different pre-training paradigms and CIL benchmarks. As shown in Table 1, the performance of state-of-the-art prompt-based approaches degrades remarkably under self-supervised pre-training (e.g., iBOT-21K) and fine-grained classification (e.g., Split CUB-200), while both versions of our approach outperform them significantly.

On the other hand, a potential limitation of prompt-based methodologies is that, the pre-trained knowledge in backbone parameters *cannot* be updated and enriched from incremental tasks, which has been rarely discussed in previous literature. Motivated by this, we then consider *upstream continual learning*, i.e., the ability of accumulating knowledge in backbone parameters. Specifically, after downstream continual learning of multiple incremental tasks, we evaluate the performance of the backbone to perform few-shot learning of an additional task randomly sampled from subsequent unseen classes. As shown in Table 2, the backbone adapted by the LoRA version of our approach acquires strong improvements in few-shot learning, compared to the unadapted backbone of the Prompt version. In addition to Split ImageNet-R and Split CUB-200 that split all tasks from the same dataset, we further consider a mixture of tasks sampled from CUB-200 [14] and Cars-196 [4] datasets, where the improvements remain significant. These results demonstrate the importance and feasibility of synchronizing upstream and downstream continual learning.

5 Discussion

In this work, we propose a general framework for continual learning in the context of pre-training, with decomposing the objective into three hierarchical components (i.e., WTP, TII and TAP) and optimizing them explicitly with PEFT techniques and representation statistics. Through extensive experiments, we demonstrate the superiority and generality of our approach in downstream continual learning, and further elaborate on the importance of upstream continual learning, which requires updating the backbone parameters rather than instructing only (intermediate) inputs. Interestingly, the proposed framework requires sequential invocation of the unadapted and (task-specific) adapted representations for inference, which is consistent with recent advances in biological learning and memory [7, 6] that the activation of non-memory cells and memory cells (as well as their specific sub-populations) is internally switched. This connection potentially bridges the intrinsic mechanisms of biological and artificial intelligence in acquisition of open-world knowledge. Based on the above discussion, we believe that the strong ability of continual learning would be a critical component in realizing intrinsically motivated open-ended learning.

Acknowledgements

This work was supported by the National Key Research and Development Program of China (No. 2020AAA0106302), NSFC Projects (Nos. 62061136001, 92248303, 62106123, 61972224), BNRist (BNR2022RC01006), Tsinghua Institute for Guo Qiang, and the High Performance Computing Center, Tsinghua University. L.W. is also supported by Shuimu Tsinghua Scholar, and J.Z. is also supported by the XPlorer Prize.

References

[1] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[3] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pages 709–727. Springer, 2022.

[4] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[5] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[6] Bo Lei, Bilin Kang, Wantong Lin, Haichao Chen, Yuejun Hao, Jian Ma, Songhai Shi, and Yi Zhong. Adult newborn granule cells confer emotional state–dependent adaptability in memory retrieval. *Science Advances*, 8(45):eabn2136, 2022.

[7] Bo Lei, Li Lv, Shiqiang Hu, Yikai Tang, and Yi Zhong. Social experiences switch states of memory engrams through regulating hippocampal rac1 activity. *Proceedings of the National Academy of Sciences*, 119(15):e2116844119, 2022.

[8] Sanket Vaibhav Mehta, Darshan Patil, Sarath Chandar, and Emma Strubell. An empirical investigation of the role of pre-training in lifelong learning. *arXiv preprint arXiv:2112.09153*, 2021.

[9] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[10] Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*, 2021.

[11] Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. Learning multiple visual domains with residual adapters. *Advances in Neural Information Processing Systems*, 30, 2017.

[12] 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.

[13] Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

[14] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[15] Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *arXiv preprint arXiv:2310.07234*, 2023.

[16] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*, 2023.

[17] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam's razor for domain incremental learning. *Advances in Neural Information Processing Systems*, 35:5682–5695, 2022.

[18] 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 *European Conference on Computer Vision*, pages 631–648. Springer, 2022.

[19] 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.

[20] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. *arXiv preprint arXiv:2303.05118*, 2023.

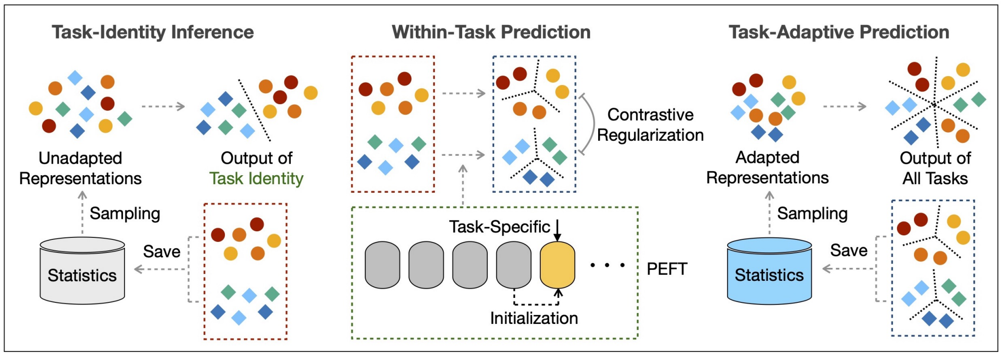

Figure 1: The proposed framework for continual learning with pre-training.

Algorithm 1 Training Algorithm of Our Approach

Input: Pre-trained transformer backbone f_θ, training sets $\mathcal{D}_1, ..., \mathcal{D}_T$, number of tasks T, number of epochs E, hyperparameters τ and λ.
Output: Parameters $\boldsymbol{e}_1, ..., \boldsymbol{e}_T$, ω and ψ

```
Initialize e_1, ω and ψ
for t = 1, ..., T do
    for c ∈ Y_t do
        Obtain Ĝ_c from f_θ and D_t                    ▷ Unadapted Representations
    if t > 1 then
        Initialize e_t ← e_{t−1}
    for epoch = 1, ..., E do
        Optimize e_t and ψ with L_WTP in Eq. (6)        ▷ Within-Task Prediction
        Optimize ω with L_TII in Eq. (7)                ▷ Task-Identity Inference
        Optimize ψ with L_TAP in Eq. (8)                ▷ Task-Adaptive Prediction
    for c ∈ Y_t do
        Obtain G_c from f_θ, e_t and D_t                ▷ Adapted Representations
return (e_1, ..., e_T, ω, ψ)
```

A Theoretical Foundation

A.1 Class-Incremental Learning (CIL)

Proof of Theorem 1

Proof. For class-incremental learning (CIL) with pre-training, assume $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \leq \delta$, $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] \leq \epsilon$, and $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})] \leq \eta$. Let $y = \mathcal{Y}_{\bar{i},\bar{j}}$ be the ground truth of an $\boldsymbol{x}$, where $\bar{i} \in \{1, ..., t\}$ and $\bar{j} \in \{1, ..., |\mathcal{Y}_i|\}$ denote the task identity and within-task index, respectively.

As we defined,

$$\begin{aligned} H_{\mathrm{WTP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},j} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)\}_j) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta), \end{aligned} \tag{9}$$

$$\begin{aligned} H_{\mathrm{TII}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{i}}, \{P(\boldsymbol{x} \in \mathcal{X}_i | \mathcal{D}, \theta)\}_i) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}} | \mathcal{D}, \theta), \end{aligned} \tag{10}$$

$$\begin{aligned} H_{\mathrm{TAP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{c}}, \{P(\boldsymbol{x} \in \mathcal{X}^c | \mathcal{D}, \theta)\}_c) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^{\bar{c}} | \mathcal{D}, \theta) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^y | \mathcal{D}, \theta). \end{aligned} \tag{11}$$

Then, we have

$$\begin{aligned}
&\mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D},\theta)\}_{i,j}) \\
&= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta) \\
&= -\log(P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}},\mathcal{D},\theta)P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}}|\mathcal{D},\theta)) \\
&= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}},\mathcal{D},\theta) - \log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}}|\mathcal{D},\theta) \\
&= H_{\mathrm{WTP}}(\boldsymbol{x}) + H_{\mathrm{TII}}(\boldsymbol{x}).
\end{aligned} \tag{12}$$

Taking expectations on Eq. (11), we have

$$\mathcal{L}_1 = \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})] \leq \eta. \tag{13}$$

Taking expectations on both sides of Eq. (12), we have

$$\begin{aligned}
\mathcal{L}_2 &= \mathbb{E}_{\boldsymbol{x}}[\mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D},\theta)\}_{i,j})] \\
&= \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] \\
&\leq \delta + \epsilon.
\end{aligned} \tag{14}$$

Since our objective of CIL with pre-training is $\max[P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta)]$, then we have the loss error

$$\begin{aligned}
\mathcal{L} &= \max\{\mathbb{E}_{\boldsymbol{x}}[\mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D},\theta)\}_{i,j})], \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})]\} \\
&= \max\{\mathcal{L}_2, \mathcal{L}_1\} \\
&= \max\{\delta + \epsilon, \eta\}.
\end{aligned} \tag{15}$$

This finishes the proof.

Proof of Theorem 2

Proof. For CIL with pre-training, its loss error $\mathcal{L} \leq \xi$. Assume $\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} \subseteq \mathcal{X}_{\bar{i}}$. According to the proof of Theorem 1, we have

$$\begin{aligned}
H_{\mathrm{WTP}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}},\mathcal{D},\theta) \\
&= -\log \frac{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta)}{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}}|\mathcal{D},\theta)} \\
&\leq -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta) \\
&= \mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D},\theta)\}_{i,j}) \\
&= \mathcal{L}_2 \leq \xi.
\end{aligned} \tag{16}$$

Likewise, we have

$$\begin{aligned}
H_{\mathrm{TII}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}}|\mathcal{D},\theta) \\
&= -\log \frac{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta)}{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\boldsymbol{x} \in \mathcal{X}_{\bar{i}},\mathcal{D},\theta)} \\
&\leq -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta) \\
&= \mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\mathcal{D},\theta)\}_{i,j}) \\
&= \mathcal{L}_2 \leq \xi.
\end{aligned} \tag{17}$$

In addition, we have formulated the final goal of CIL as a multi-objective optimization problem, i.e., $\max[P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}}|\mathcal{D},\theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta)]$. Then, each objective must guarantee the loss error less than ξ, i.e.,

$$\begin{aligned}
H_{\mathrm{TAP}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta) \\
&= \mathcal{L}_1 \leq \xi.
\end{aligned} \tag{18}$$

This finishes the proof.

A.2 Domain-Incremental Learning (DIL)

For domain-incremental learning (DIL), Let $\mathcal{X}_t = \bigcup_j \mathcal{X}_{t,j}$ and $\mathcal{Y}_t = \{\mathcal{Y}_{t,j}\}$, where $j \in \{1, ..., |\mathcal{Y}_t|\}$ denotes the j-th class in task t. Now assume we have a ground event denoted as $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_t\}$ and a pre-trained model f_θ. For any sample $\boldsymbol{x} \in \bigcup_{k=1}^{t} \mathcal{X}_k$, a general goal of the DIL problem is to learn $P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D}, \theta)$, where $\mathcal{X}_{*,j}$ represents the j-th class domain in any task. Of note, $\mathcal{Y}_t = \mathcal{Y}_{t'}$, $\forall t \neq t'$ for DIL. This can be decomposed into two probabilities, including task-identity inference (TII) and within-task prediction (WTP), denoted as $P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)$ and $P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta)$, respectively. Based on Bayes' theorem, we have

$$P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D}, \theta) = \sum_i P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta) P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta), \tag{19}$$

where $\{*, j\}$ represents the j-th class in each domain.

Then we have the following theorems in terms of the sufficient and necessary conditions for improving DIL with pre-training.

Theorem 3 *For domain-incremental learning with pre-training, if* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \leq \delta$, $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] \leq \epsilon$, *and* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})] \leq \eta$, *we have the loss error* $\mathcal{L} \in [0, \max\{\delta + \epsilon + \log t, \eta\}]$, *regardless whether WTP, TII and TAP are trained together or separately.*

Proof of Theorem 3

Proof. Let $\bar{j} \in \{1, ..., |\mathcal{Y}_t|\}$ and $y \in \mathcal{Y}_t$ be the ground truth of an $\boldsymbol{x}$ w.r.t. the within-task index and class label, and $y = \mathcal{Y}_{i,\bar{j}}$ for any $i \in \{1, ..., t\}$. Eq. (19) suggests that if we can improve either the WTP performance $P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta)$, the TII performance $P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)$, or both, then the DIL performance $P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta)$ would be improved. However, such an improvement is limited since it is upper-bounded by WTP or TII. To further improve the DIL performance, we propose a hierarchical decomposition of the objective. That is, besides the improvement of $P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D}, \theta)$, we also need to directly improve the performance of task-adaptive prediction (TAP), denoted as $P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta)$, where $y \in \{1, ..., |\mathcal{Y}_t|\}$, $\mathcal{X}^y$ represents the domain of class y in all observed domains, and $\mathcal{X}^y = \bigcup_i \mathcal{X}_{i,\bar{j}}$. Then the final goal of DIL is formulated as a multi-objective optimization problem, i.e., $\max[P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D}, \theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta)]$. Notice that the WTP probability is a categorical distribution over all observed domains $\{1 : t\}$, while the TAP probability is over all observed classes $\bigcup_{k=1}^t \mathcal{Y}_k$.

As similarly defined in CIL, here

$$\begin{aligned} H_{\mathrm{WTP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta)\}_j) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta), \end{aligned} \tag{20}$$

$$\begin{aligned} H_{\mathrm{TII}}(\boldsymbol{x}) &= \mathcal{H}(\gamma, \{P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)\}_i) \\ &= -\gamma_i \log P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta), \end{aligned} \tag{21}$$

$$\begin{aligned} H_{\mathrm{TAP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{c}}, \{P(\boldsymbol{x} \in \mathcal{X}^c|\mathcal{D}, \theta)\}_c) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^{\bar{c}}|\mathcal{D}, \theta) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D}, \theta), \end{aligned} \tag{22}$$

where $\gamma = \{\gamma_i\}_{i=1}^t$ represents the possibility of $\boldsymbol{x}$ belonging to each observed domain, $\gamma_i \in [0, 1]$ and $\sum_i \gamma_i = 1$.

Then, for any simplex γ, we have

$$\begin{aligned} &\mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D}, \theta)\}_j) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D}, \theta) \\ &= -\log(\sum_i P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta) P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)) \\ &\leq -\sum_i \gamma_i \log(\frac{P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta) P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta)}{\gamma_i}) \\ &= -\sum_i \gamma_i \log P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D}, \theta) - \sum_i \gamma_i \log P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D}, \theta) + \sum_i \gamma_i \log(\gamma_i) \\ &= \sum_i \gamma_i H_{\mathrm{WTP}} + H_{\mathrm{TII}} + \mathcal{H}(\gamma). \end{aligned} \tag{23}$$

Taking expectations on Eq. (22), we have

$$\mathcal{L}_1 = \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})] \le \eta. \tag{24}$$

Taking expectations on both sides of Eq. (23), we have

$$\begin{aligned}\mathcal{L}_2 &= \mathbb{E}_{\boldsymbol{x}}[\mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D},\theta)\}_j]\\ &\le \sum_i \gamma_i \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] + \mathcal{H}(\gamma)\\ &\le \delta + \epsilon + \log t.\end{aligned} \tag{25}$$

Since our objective of DIL with pre-training is $\max[P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D},\theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta)]$, then we have the loss error

$$\begin{aligned}\mathcal{L} &= \max\{\mathbb{E}_{\boldsymbol{x}}[\mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D},\theta)\}_j)], \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TAP}}(\boldsymbol{x})]\}\\ &= \max\{\mathcal{L}_2, \mathcal{L}_1\}\\ &= \max\{\delta + \epsilon + \log t, \eta\}.\end{aligned} \tag{26}$$

This finishes the proof.

Theorem 4 *For domain-incremental learning with pre-training, if the loss error $\mathcal{L} \le \xi$, then there always exist (1) a WTP, s.t. $H_{\mathrm{WTP}} \le \xi$; (2) a TII, s.t. $H_{\mathrm{TII}} \le \xi$; and (3) a TAP, s.t. $H_{\mathrm{TAP}} \le \xi$.*

Proof of Theorem 4 For DIL with pre-training, its loss error $\mathcal{L} = \max[\mathcal{L}_1, \mathcal{L}_2] \le \xi$. Assume $\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}} \subseteq \mathcal{X}^y$. According to the proof of Theorem 3, if we define $P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\mathcal{D},\theta) = P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D},\theta)$, we have

$$\begin{aligned}H_{\mathrm{WTP}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D},\theta)\\ &= -\log \frac{P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\mathcal{D},\theta)}{P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D},\theta)}\\ &\le -\log P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\mathcal{D},\theta)\\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D},\theta)\\ &= \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D},\theta)\}_j)\\ &= \mathcal{L}_2 \le \xi.\end{aligned} \tag{27}$$

Likewise, if we define $\gamma_i = 1$ and $\gamma_{i'} = 0$ for $i' \neq i$, we have

$$\begin{aligned}H_{\mathrm{TII}}(\boldsymbol{x}) &= -\sum_i \gamma_i \log P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D},\theta)\\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_i|\mathcal{D},\theta)\\ &= -\log \frac{P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\mathcal{D},\theta)}{P(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\boldsymbol{x} \in \mathcal{X}_i, \mathcal{D},\theta)}\\ &\le -\log(\boldsymbol{x} \in \mathcal{X}_{i,\bar{j}}|\mathcal{D},\theta)\\ &= -\log(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D},\theta)\\ &= \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{*,j}|\mathcal{D},\theta)\}_j)\\ &= \mathcal{L}_2 \le \xi.\end{aligned} \tag{28}$$

In addition, we have formulated the final goal of DIL as a multi-objective optimization problem, i.e., $\max[P(\boldsymbol{x} \in \mathcal{X}_{*,\bar{j}}|\mathcal{D},\theta), P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta)]$. Then, each objective must guarantee the loss error less than ξ, i.e.,

$$\begin{aligned}H_{\mathrm{TAP}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}^y|\mathcal{D},\theta)\\ &= \mathcal{L}_1 \le \xi.\end{aligned} \tag{29}$$

This finishes the proof.

A.3 Task-Incremental Learning (TIL)

For task-incremental learning (TIL), let $\mathcal{X}_t = \bigcup_j \mathcal{X}_{t,j}$ and $\mathcal{Y}_t = \{\mathcal{Y}_{t,j}\}$, where $j \in \{1, ..., |\mathcal{Y}_t|\}$ indicates the j-th class in task t. Now assume we have a ground event denoted as $\mathcal{D} = \{\mathcal{D}_1, ..., \mathcal{D}_t\}$

and a pre-trained model f_θ. For any sample $\boldsymbol{x} \in \bigcup_{k=1}^{t} \mathcal{X}_k$, a general goal of the TIL problem is to learn $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},j} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)$, where $\bar{i} \in \{1, ..., t\}$ and $j \in \{1, ..., |\mathcal{Y}_{\bar{i}}|\}$. This can be equivalent to within-task prediction (WTP). Different from CIL, the task identity is provided in TIL. Unlike DIL, $\mathcal{Y}_t \cap \mathcal{Y}_{t'} = \emptyset, \forall t \neq t'$. Then we have the following theorems in terms of the sufficient and necessary conditions for improving TIL with pre-training.

Theorem 5 *For task-incremental learning with pre-training,* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{TII}}(\boldsymbol{x})] = 0$*, and task-adaptive prediction (TAP) is degraded into within-task prediction (WTP). If* $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \leq \delta$*, we have the loss error* $\mathcal{L} \in [0, \delta]$.

Proof of Theorem 5

Proof. For task-incremental learning (TIL) with pre-training, assume $\mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \leq \delta$. Given an $\boldsymbol{x}$ with the task identity $\bar{i} \in \{1, ..., t\}$, let $\bar{j} \in \{1, ..., |\mathcal{Y}_i|\}$ be the ground truth of $\boldsymbol{x}$ w.r.t. the within-task index, and $y = \mathcal{Y}_{\bar{i},\bar{j}}$ be the ground truth label of $\boldsymbol{x}$.

As similarly defined in CIL, here

$$\begin{aligned} H_{\mathrm{WTP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},j} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)\}_j) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta), \end{aligned} \tag{30}$$

$$\begin{aligned} H_{\mathrm{TII}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{i}}, \{P(\boldsymbol{x} \in \mathcal{X}_i | \mathcal{D}, \theta)\}_i) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}} | \mathcal{D}, \theta) \\ &= -\log 1 = 0, \end{aligned} \tag{31}$$

$$\begin{aligned} H_{\mathrm{TAP}}(\boldsymbol{x}) &= \mathcal{H}(\mathbf{1}_{\bar{c}}, \{P(\boldsymbol{x} \in \mathcal{X}^c | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)\}_c) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^{\bar{c}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}^{y} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta) \\ &= H_{\mathrm{WTP}}(\boldsymbol{x}). \end{aligned} \tag{32}$$

Then, we have

$$\begin{aligned} &\mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j} | \mathcal{D}, \theta)\}_{i,j}) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \mathcal{D}, \theta) \\ &= -\log(P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta) P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}} | \mathcal{D}, \theta)) \\ &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta) - \log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}} | \mathcal{D}, \theta) \\ &= H_{\mathrm{WTP}}(\boldsymbol{x}) + H_{\mathrm{TII}}(\boldsymbol{x}) \\ &= H_{\mathrm{WTP}}(\boldsymbol{x}). \end{aligned} \tag{33}$$

Taking expectations on both sides of Eq. (33), we have

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_{\boldsymbol{x}}[\mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j} | \mathcal{D}, \theta)\}_{i,j})] \\ &= \mathbb{E}_{\boldsymbol{x}}[H_{\mathrm{WTP}}(\boldsymbol{x})] \\ &\leq \delta. \end{aligned} \tag{34}$$

Since our objective of TIL with pre-training is $P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta)$, then we have the loss error $\mathcal{L} \leq \delta$. This finishes the proof.

Theorem 6 *For task-incremental learning with pre-training, if the loss error* $\mathcal{L} \leq \xi$*, then there always exists a WTP, s.t.* $H_{\mathrm{WTP}} \leq \xi$.

Proof of Theorem 6

For TIL with pre-training, its loss error $\mathcal{L} \leq \xi$. Assume $\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} \subseteq \mathcal{X}_{\bar{i}}$. According to the proof of Theorem 5, we have

$$\begin{aligned} H_{\mathrm{WTP}}(\boldsymbol{x}) &= -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \boldsymbol{x} \in \mathcal{X}_{\bar{i}}, \mathcal{D}, \theta) \\ &= -\log \frac{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \mathcal{D}, \theta)}{P(\boldsymbol{x} \in \mathcal{X}_{\bar{i}} | \mathcal{D}, \theta)} \\ &\leq -\log P(\boldsymbol{x} \in \mathcal{X}_{\bar{i},\bar{j}} | \mathcal{D}, \theta) \\ &= \mathcal{H}(\mathbf{1}_{\bar{i},\bar{j}}, \{P(\boldsymbol{x} \in \mathcal{X}_{i,j} | \mathcal{D}, \theta)\}_{i,j}) \\ &\leq \xi. \end{aligned} \tag{35}$$

This finishes the proof.

B Implementation Details

In this section, we describe the implementation details of all experiments.

Baseline: We follow the same implementations for all baselines as their original papers [19, 18, 12] (except S-Prompt that was originally designed for domain-incremental learning), which have been shown to yield strong performance. Specifically, L2P [19] is implemented with $M = 30$ for the total number of prompts ($M = 10$ [19] and $M = 30$ [18] have similar results), $L_{\boldsymbol{p}} = 5$ for the prompt length, and $N = 5$ for the Top-N keys. DualPrompt [18] is implemented with $L_{\boldsymbol{g}} = 5$ for the prompt length of task-sharing prompts $\boldsymbol{g}$ inserted into layers 1-2 and $L_{\boldsymbol{e}} = 20$ for the prompt length of task-specific prompts $\boldsymbol{e}$ inserted into layers 3-5. S-Prompt [17] is implemented similarly to DualPrompt but replaces all task-sharing prompts with task-specific prompts, i.e., the task-specific prompts are inserted into layers 1-5 with prompt length $L_{\boldsymbol{e}} = 20$. CODA-Prompt [12] is implemented with $M = 100$ for the total number of prompts and $L_{\boldsymbol{p}} = 8$ for the prompt length, inserted into the same layers 1-5 as DualPrompt and S-Prompt. Ours-Prompt adopts a similar architecture as S-Prompt, but replaces the task-specific keys with an auxiliary output layer $\hat{h}_\omega$ to predict the task identity and further preserves statistics of unadapted and adapted representations. Compared with Ours-Prompt, Ours-LoRA replaces the prompt parameters with LoRA parameters inserted into corresponding layers. The hyperparameters of LoRA are set to $\mathrm{rank} = 8$ and $\mathrm{scaling} = 1/\mathrm{rank}$, with only the value projection parameters of each layer implemented LoRA. We additionally implement a dataset-shared LoRA updated by sequential fine-tuning, adding it to the backbone for few-shot learning. For Split CUB & Cars that includes multiple datasets, we use the TII function to predict which dataset-shared LoRA to use. The hyperparameters of contrastive regularization in our approach are set to $\tau = 0.8$ and $\lambda = 0.1$.

Training Regime: Following the implementations of previous work [19, 18], we employ a pre-trained ViT-B/16 backbone, an Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and a batch size of 128. The learning rate is set to 0.001 with cosine decay for CODA-Prompt, compared to 0.005 for other approaches. The total number of epcohs is set to 50 for all approaches.

Evaluation Metric: We focus on three evaluation metrics for downstream continual learning, including the final average accuracy (FAA), cumulative average accuracy (CAA) and final forgetting measure (FFM). Specifically, we define the accuracy on the i-th task after learning the t-th task as $A_{i,t}$, and define the average accuracy of all seen tasks as $AA_t = \frac{1}{t}\sum_{i=1}^{t} A_{i,t}$. After learning all T tasks, we report FAA $= AA_T$, CAA $= \frac{1}{T}\sum_{t=1}^{T} AA_t$, and FFM $= \frac{1}{T-1}\sum_{i=1}^{T-1} \max_{t\in\{1,\ldots,T-1\}}(A_{i,t} - A_{i,T})$. FAA is the primary metric to evaluate the final performance of continual learning, CAA further reflects the historical performance, and FFM serves as a measure of catastrophic forgetting. For upstream continual learning, we randomly sample multiple N-way K-shot tasks to perform few-shot learning and present their average accuracy.