SIMILARITY-AWARE PROMPT ROUTING FOR DATA-EFFICIENT CONTINUAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual Learning (CL) enables models to adapt to new data after deployment. Recent approaches increasingly use representations from pretrained or foundation models to reduce catastrophic forgetting as new tasks arrive. Yet, recent approaches often assume abundant per-task data or strictly non-overlapping tasks. In this paper, we address a more realistic setting where tasks have limited data and may contain overlapping concepts. In this setting, knowledge transfer must be carefully controlled to overcome low-sample noise while mitigating interference between tasks. To address this, we propose Similarity-Aware Prompt Routing (SPR), an adaptive routing framework for prompt-based continual learning that progressively builds task-similarity awareness over a pool of prompts. SPR relies on two key components: incremental global pooling, which mitigates prompt association noise through gradual prompt introduction, and instance-wise prompt masking, which separates incoming samples into those aligning with current prompts and those requiring new ones. Experiments across varying data volumes and inter-task similarities demonstrate that our method actively prevents negative transfer, enhances sample efficiency, and is broadly applicable.

1 INTRODUCTION

Machine learning models often need to adapt to new data after deployment due to complex and intermittent distribution shifts. Continual learning (CL) addresses this setting by updating a model as new data arrive while retaining useful knowledge from earlier data, often with limited or no access to past examples. Its central challenge is to learn from distribution shifts without catastrophically forgetting prior capabilities (McCloskey & Cohen, 1989; Wang et al., 2024). In many real-world applications, newly arriving tasks may overlap with previously seen tasks in complex ways, while labeled examples for each new task remain scarce because collecting and annotating data is costly (Qin & Joty, 2022; Li et al., 2025). For instance, in fraud detection, each fraud type may be represented by only a few cases (data scarcity), but fraudulent patterns often share features across transaction types (task overlap). Similarly, in image classification, tasks of interest evolve but may share similar concepts, as illustrated in Figure 1.

Recent continual learning methods increasingly use pretrained or foundation models to adapt to new tasks while mitigating catastrophic forgetting, often through prompt-based updates (Zhou et al., 2024; Wang et al., 2022c; Smith et al., 2023). Somewhat surprisingly, although earlier work has studied task relatedness and few-shot learning (Thrun & O'Sullivan, 1996; Lee et al., 2021; Tao et al., 2020; Chi et al., 2022), prominent prompt-based methods are primarily evaluated on sequences with disjoint class splits and substantial training data per task (Wang et al., 2022c; Smith et al., 2023). This motivates studying how to adapt pretrained models when new tasks have few labeled examples and share an unknown degree of structure with earlier tasks.

One approach to mitigating forgetting is to isolate task-specific knowledge in separate models or adapters (Rusu et al., 2016; Mallya & Lazebnik, 2018). This idea has grown in popularity with pre-trained transformer models, as lightweight learnable prompts may be injected into the model for parameter-efficient adaptation to new tasks (Wang et al., 2022c;b;a; Smith et al., 2023). Because only a small set of prompt parameters is learned while the backbone stays frozen, these methods can adapt from fewer samples, making them a natural starting point when per-task data are limited. Within this line of work, two prompting strategies are prominent. Task-expert methods (Wang et al., 2022b;a)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

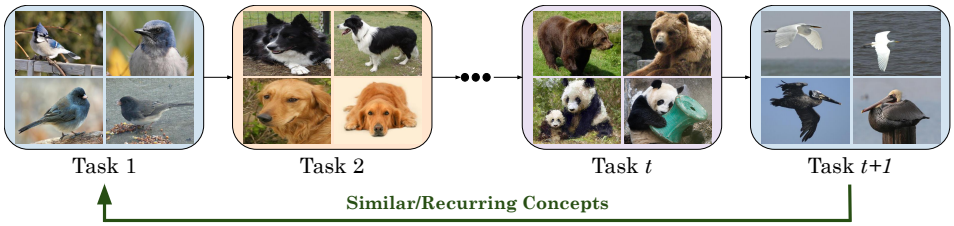

Figure 1: As tasks of interest evolve over time, the learner may encounter similar or recurring concepts. For example, Task 1 and Task $t + 1$ may both involve the high-level category of birds.

assign each task its own prompt trained on task-specific data. While effective, it relies upon sufficient supervision and clear task boundaries. Global prompt pools (Wang et al., 2022c; Smith et al., 2023) share prompts across all tasks and learn which prompts to select for each input; this encourages reuse but introduces noise as the selection of prompts for each sample, i.e., the *prompt routing*, may be suboptimal.

These limitations are compounded when data are scarce and tasks overlap. Task-specific prompts prompts cannot draw on related tasks when their own data are limited, while global pools, which could in principle exploit shared structure, become less reliable as scarce data make prompt selection noisier. When task data are scarce, a model must lean more heavily on knowledge from the pre-trained backbone and from related tasks. Yet the same scarcity makes routing less reliable: with few samples, routing errors are harder to correct, and a prompt may be updated with data from unrelated tasks. Because task overlap is unknown in advance, sharing prompts can therefore cause negative transfer as easily as it enables positive transfer.

This tension between knowledge sharing and routing noise calls for more adaptive and reliable prompt allocation. We therefore introduce Similarity-aware Prompt Routing (SPR), which balances knowledge sharing with robustness to routing noise. Our key idea is to use an out-of-distribution (OOD) detector to decide how incoming data should be used. Samples deemed in-distribution are used to refine the current prompt pool, while OOD samples are assigned to a new task-specific prompt. We additionally reduce the amount of early routing errors through an incrementally grown prompt pool.

Contributions. Our contributions can be summarized as follows.

- We identify a general, yet critically overlooked, setting in continual learning which motivates a new perspective on prompt routing.
- To enable effective knowledge sharing despite data-scarcity, we challenge the rigid routing structures of prior prompt-based approaches and propose an adaptive and similarity-aware routing mechanism. Our design contains two innovative algorithmic components: incremental global pooling and instance-wise prompt masking. Together, they leverage potential task overlaps while preventing negative mutual interference in the presence of per-task data scarcity.
- Our method establishes a connection between CL and out-of-distribution detection (Kim et al., 2022; 2025): by reliably measuring the similarity between incoming data and previously encountered tasks, the model can more effectively allocate prompts. We hope that this connection can inspire the design of future CL methods by drawing on tools and insights developed in the out-of-distribution detection community.
- Experiments across varying data volumes and inter-task similarity show that our method enhances sample efficiency and is broadly applicable.

2 Related Work

Prior work has recognized that tasks may be related in different ways: some methods infer relatedness to select useful prior tasks for transfer, while theoretical studies examine how prescribed feature or readout similarity affects forgetting (Thrun & O'Sullivan, 1996; Lee et al., 2021). Few-shot class-incremental learning also addresses scarce labeled data, but typically assumes that successive

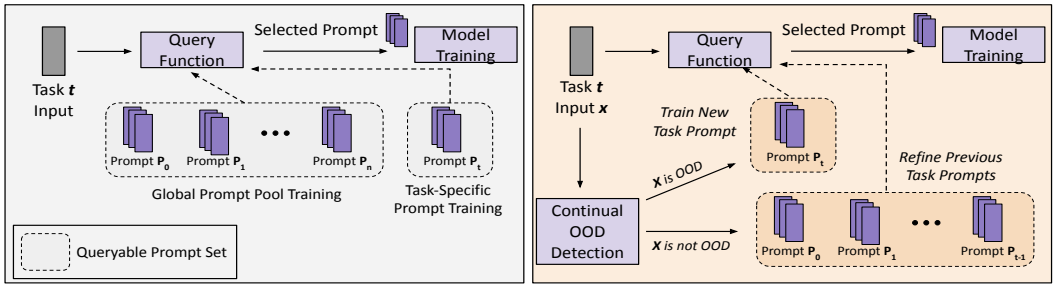

Figure 2: Comparison of Prompt Routing Strategies. (Left) When training on task t, typical methods select prompt models in one of two ways: either by querying an unstructured global pool or by rigidly training a prompt designated for task t. (Right) We employ an out-of-distribution (OOD) detector to dynamically determine which prompts may be queried. Inputs detected as novel (OOD) are routed to a task-specific prompt to learn new concepts, whereas for familiar samples, we query the set of previous task prompts to facilitate knowledge reuse.

tasks introduce disjoint classes (Tao et al., 2020; Chi et al., 2022). Several recent surveys provide comprehensive overviews of the foundations, challenges, and recent advances in CL (Wang et al., 2024; De Lange et al., 2021; Shi et al., 2025). In the following, we briefly touch on a few lines of work that are closely related to our setting.

Rehearsal-based/rehearsal-free methods. Rehearsal-based methods (Rebuffi et al., 2017; Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2020; Buzzega et al., 2020; Verwimp et al., 2021; Yoon et al., 2021; Zhang et al., 2022) generally yield the strongest performance in class incremental learning settings. These methods maintain a memory buffer of samples from past distributions (experience replay) or generate them via generative models (pseudo-rehearsal). While rehearsal-based methods (Rebuffi et al., 2017; Buzzega et al., 2020) remain strong baselines, their reliance on raw data storage raises significant privacy concerns, motivating the shift toward rehearsal-free alternatives.

Prompt-based CL. Prompt-tuning (Lester et al., 2021) has emerged as a compelling parameter-efficient alternative to traditional fine-tuning. Originating as an alternative to text-based prompts for conditioning language models, prompting attaches a set of learnable vectors (prompts) to a frozen pre-trained model to be trained via backpropagation. The core challenge in this domain is designing mechanisms to select and update prompts dynamically. L2P (Wang et al., 2022c) introduced the concept of a shared prompt pool, utilizing a key-query mechanism to retrieve relevant prompts for a given input. DualPrompt (Wang et al., 2022b) refined this by architecturally separating prompts into task-agnostic and task-specific components to better balance stability and plasticity. Other variants, such as S-Prompts (Wang et al., 2022a), focused purely on task-specific learning, while CODA-Prompt (Smith et al., 2023) employed an attention-based weighted summation to construct prompts from a pool. Consistent Prompting (Gao et al., 2024) further emphasizes the stability of prompt behavior across distributions.

Soft-boundary CL. Several works have studied continual learning with soft or noisy task boundaries (Wang et al., 2024). Ke et al. (2020) approximate task similarity by comparing the performance of the previous model with that of a model trained from scratch, and use gradient masking and knowledge distillation to control knowledge transfer. Bang et al. (2021) introduce a rehearsal method when a fraction of each task consists of previously encountered classes. Koh et al. (2022) propose a rehearsal memory manager for streaming data with both novel and known classes. Buzzega et al. (2020) study a task-overlap setting and propose distilling from previous models by storing predicted logits alongside buffered examples. Departing from these works, our method is based on prompt-tuning, is rehearsal-free, and accommodates data scarcity (i.e., insufficient data to re-train a reference model).

3 Problem Formulation

We study the problem of rehearsal-free CL, where a classification model $\mathcal{M}$ is trained sequentially on a series of tasks without storing past data. Formally, the learner observes a sequence of T datasets $\mathfrak{S} = \{\mathcal{S}_0, \mathcal{S}_1, \ldots, \mathcal{S}_{T-1}\}$, with each $\mathcal{S}_t = \{(x_t^i, y_t^i)\}_{i=1}^{n_t}$ where $x_t^i \in \mathcal{X}$ is the input feature vector, $y_t^i \in \mathcal{C}$ is a label. Let $\mathcal{C}_t = \{y_t^i : i = 1, \cdots, n_t\}$ denote the collection of distinct labels of task $\mathcal{S}_t$. As

mentioned in Section 1, we consider the general setting wherein n_t may be small, and $\mathcal{C}_t \cap \mathcal{C}_{t'} \neq \emptyset$ for some $t' \neq t$.

To avoid overfitting and to better capture the generalization capability of the trained model, we measure its performance in terms of the average generalization error of the tasks in T. Formally, let d_t denote the underlying distribution for which $(x_t^i, y_t^i) \overset{i.i.d.}{\sim} d_t$. We would like to obtain a model $\mathcal{M}$ that minimizes the average generation error

$$\min_{\mathcal{M}} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_{(x,y)\sim d_t}[\mathbf{1}_{\{\mathcal{M}(x)\neq y\}}], \tag{1}$$

where $\mathcal{M}(x)$ denotes the label prediction, and $\mathbf{1}_{\{\mathcal{M}(x)\neq y\}}$ is an indicator function which takes value 1 if $\mathcal{M}(x) \neq y$, and 0 otherwise. At the inference time, the trained model is not told to which task the input feature encountered x belongs. We focus on rehearsal-free methods, i.e., no subset of samples from previous tasks is stored.

4 Preliminaries: Prompt-Based CL

We introduce the relevant backgrounds and notations for the prompt-based continual learning and prompt routing that our method builds upon.

Model Structure. In the prompt-based CL, model to be learned is decomposed into three components: (1) a pre-trained backbone model f_θ (e.g., ViT or BERT), which encodes the general knowledge obtained in the pre-training; (2) a collection of prompts $\mathcal{P}$ that can be flexibly combined with f_θ to extract knowledge that are specific to the given task sequence $\mathfrak{S}$, and (3) classification head g_ψ, where ψ denotes learnable parameters (Dosovitskiy et al., 2020; Wang et al., 2022c).

For any given x, its label prediction is determined as

$$\widehat{y}(x) = g_\psi \circ (f_\theta(x) \oplus p(x)), \tag{2}$$

where $p(x)$ is a set of prompts that are relevant to input x, and, with slight abuse of notation, $\oplus$ represents the chosen modulation operation between the prompts and the pre-trained backbone.

Throughout the CL process, the backbone model f_θ remains fixed, whereas ψ is repeatedly updated for each task. In addition, for any given task, the prompts are updated according to some rules of determining $p(x)$, which we will elaborate on soon.

Pre-trained Transformer f_θ. Let $E : \mathcal{X} \to \mathbb{R}^{N\times d}$ denote an embedding function that tokenizes an input $x \in \mathcal{X}$ and maps it to a sequence of N d-dimensional token embeddings $[e^1; \ldots; e^N] = E(x)$. Depending on the modality, tokens may correspond to word-fragments or image patches (Dosovitskiy et al., 2020). Together with a class token e_{cls} and positional embeddings e_{pos}, the input to the first layer is

$$h^0 = [e_{\text{cls}}; e_p^1; \ldots; e_p^N] + e_{\text{pos}} \in \mathbb{R}^{L^h \times d}, \tag{3}$$

where $L^h = N + 1$ is the sequence length.

A transformer encoder layer applies multi-head self-attention (MSA) followed by an MLP. For the l-th layer, with input $h^l \in \mathbb{R}^{L^h \times d}$,

$$\text{MSA}(h^Q, h^K, h^V) = \text{Concat}(h^1, \ldots, h^m)W^O, \tag{4}$$

$$h^i = \text{Attention}(h^Q W_i^Q, h^K W_i^K, h^V W_i^V) \tag{5}$$

where W^O, W_i^Q, W_i^K, W_i^V are learned projections, m is the number of heads.

Prompt Structure. Our framework supports common modulation strategies such as prompt tuning (Lester et al., 2021), where prompts are prepended to the input sequence, and prefix tuning (Li & Liang, 2021), where they are prepended to keys and values in MSA layers. The choice of modulation and layers is treated as a hyperparameter.

Query-Based Selection. To handle non-stationary data distributions, prompts are typically selected dynamically based on input relevance. We focus on instance-wise query mechanisms (Wang et al.,

2022c;b), where a query function $q(\cdot, \cdot)$ measures the similarity between an input sample x and a prompt's key $\phi(p)$. Commonly, the query operates in the pre-trained embedding space (e.g., using the [CLS] token $z(x) = f_\theta(x)[0]$), and $\phi(p)$ is a learnable key vector associated with prompt p. The relevance score is defined via cosine similarity:

$$q(x, \phi(p)) := \frac{\langle z(x), \phi(p) \rangle}{\|z(x)\| \|\phi(p)\|}. \tag{6}$$

For any input x, we select a subset of prompts $p(x)$ by retrieving the top-K matches from the available pool $\mathcal{P}$:

$$p(x) := \underset{\mathcal{R} \subseteq \mathcal{P}, |\mathcal{R}|=K}{\arg\max} \sum_{p \in \mathcal{R}} q(x, \phi(p)). \tag{7}$$

While soft aggregation (Smith et al., 2023) is an alternative, we focus on hard selection for clarity.

Optimization. Let $\mathcal{L}$ denote the prediction loss (e.g., cross-entropy). The prediction is given by $\widehat{y}(x) = g_\psi(f_\theta(x) \oplus p(x))$, where $\oplus$ denotes the prompt modulation. The prompts and the classification head g_ψ are updated by minimizing the empirical risk over the current task samples $\mathcal{S}_t$:

$$\min_{p \in \mathcal{P},\, \psi} \frac{1}{n_t} \sum_{i=1}^{n_t} \mathcal{L}(y_i,\, \widehat{y}(x_i))\,. \tag{8}$$

Intuitively, frequently selected prompts (Eq. 7) contribute more to the task loss and model updates.

Existing Strategies. Existing prompt-based CL methods generally fall into two categories based on how they manage the pool $\mathcal{P}$:

- **Global Prompt Pool** (Wang et al., 2022c; Smith et al., 2023): The pool size $|\mathcal{P}| = N_p$ is fixed. Prompts are initialized once (randomly or via heuristics) and shared across all tasks. During training, the entire pool is selectable, and updates are driven purely by the query mechanism defined above.
- **Task-Specific Prompts** (Wang et al., 2022b; Zhang et al., 2022): A new, disjoint prompt set $\mathcal{P}_t$ is created for each task t. During training, the search space is restricted from $\mathcal{P}$ to $\mathcal{P}_t$, preventing interference.

5 Methodology

To enable effective knowledge transfer under per-task data scarcity, we propose Similarity-Aware Prompt Routing (SPR), which consists of two complementary components. First, incremental global pooling controls routing errors during early CL stages. Second, instance-wise prompt masking decomposes the current task $\mathcal{S}_t$ into samples that are well-explained by prior prompts versus those that are "out-of-distribution."

5.1 Incremental Global Pooling

We maintain a global prompt pool that grows incrementally as we encounter more and more tasks.

Let $\widetilde{\mathcal{P}}_t$ denote the incrementally constructed pool up to task t, such that $|\widetilde{\mathcal{P}}_t| \leq |\widetilde{\mathcal{P}}_{t'}|$ for any $t' > t$. While the prompt indices of $\widetilde{\mathcal{P}}_t$ form a subset of $\widetilde{\mathcal{P}}_{t'}$, the prompts themselves are continuously updated, so strictly speaking $\widetilde{\mathcal{P}}_t \not\subseteq \widetilde{\mathcal{P}}_{t'}$. At the start of task t, new prompts are incorporated, i.e., $\widetilde{\mathcal{P}}_t = \widetilde{\mathcal{P}}'_{t-1} \cup \{\text{new prompts}\}$, where $\widetilde{\mathcal{P}}'_{t-1}$ is the collection of updated prompts at the end $(t-1)$-th task training. We initialize these new prompts to be orthogonal to the current prompt set $\widetilde{\mathcal{P}}'_{t-1}$, following recent practice (Smith et al., 2023). This initialization minimizes the initial overlap between the new prompt training and past knowledge. All prompts in $\widetilde{\mathcal{P}}_t$ are available for routing the samples in $\mathcal{S}_t$, and may possibly be updated during training.

This component significantly mitigates negative interference, especially when per-task data is scarce. Figure 3 compares global and task-specific prompt selection on ImageNet-R across varying data scales. We observe that existing global pooling only outperforms task-specific learning in extremely low-data

regimes (e.g., 1%), indicating ineffective knowledge sharing otherwise. Our approach addresses this by dynamically sizing the pool. In early tasks (small t), most prompts remain uninitialized. When data is scarce, the few updated prompts retain significant noise, making them difficult to distinguish from unmodified ones. By initially restricting the pool size, we force the reuse and refinement of previously updated prompts. Continuously introducing new prompts then ensures fresh capacity for novel knowledge in later tasks.

5.2 Instance-Wise Prompt Masking.

Task overlap is both an enabler and a threat: it enables knowledge transfer yet, in the presence of unknown relatedness and dataset noise, it inherently risks negative interference. We introduce a novel masking strategy that acts on each sample in the current tasks $\mathcal{S}_t$.

For an input $x \in \mathcal{S}_t$, we define a binary mask $m(x, \mathcal{H}_{t-1}) \in \{0, 1\}$ based on lightweight historical summaries $\mathcal{H}_{t-1}$. This mask filters the search space of the underlying query mechanism (e.g., learnable keys) to determine the available prompt pool $\mathcal{P}_t(x)$:

$$\mathcal{P}_t(x) = \begin{cases} \widetilde{\mathcal{P}}'_{t-1}, & \text{if } m(x, \mathcal{H}_{t-1}) = 0 \\ \mathcal{P}_t^{\text{new}}, & \text{if } m(x, \mathcal{H}_{t-1}) = 1. \end{cases} \quad (9)$$

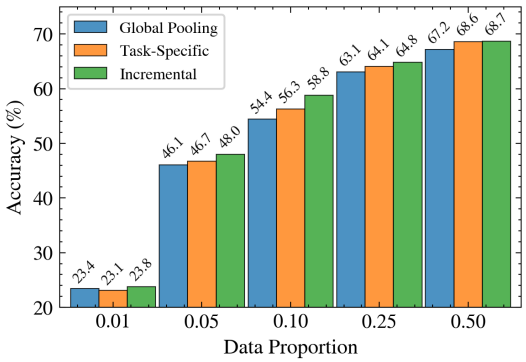

Figure 3: Average Accuracy on ImageNet-R across varying training set sizes (expressed as a fraction of the full dataset).

Importantly, this masking strategy does not replace the underlying query mechanism (e.g., learnable keys) but *filters* its search space. Intuitively, this decomposes $\mathcal{S}_t$ into samples that can be well-explained by existing prompts and the ones that are nearly "orthogonal". This orthogonality arises either from persistent statistical heterogeneity or from random sampling noise, which naturally diminishes as more data is observed.

5.3 Masking Function Design

The effectiveness of the instance-wise prompt masking in Section 5.2 depends crucially on the design of the masking functions, which is the focus on this subsection. A poor design of m can yield inferior overall performance compared to using incremental pooling alone.

The concept of history-auxiliary sample decomposition mirrors the central challenge of out-of-distribution (OOD) detection: samples well-explained by existing prompts are treated as in-distribution (ID), while those that are not are considered OOD. To implement the masking function m, we adopt the Relative Mahalanobis Distance (RMD) (Ren et al., 2021) as the discriminative metric.

Definition 1 *Let $f : \mathcal{X} \to \mathbb{R}^d$ be a given feature extractor. Let $\mathcal{D} = \{(x^i, y^i)\}_{i=1}^n$ be a given classification dataset with label set $\mathcal{C}$. For each $c \in \mathcal{C}$, define the class-wise feature mean as $\mu_c := \frac{1}{|\{i \in [n]:\, y^i = c\}|} \sum_{i=1}^n f(x^i) \mathbf{1}_{\{y^i = c\}}$. Additionally, define the background feature mean across the entire dataset as $\mu_{bg} := \frac{1}{n} \sum_{i=1}^n f(x^i)$. Let Σ denote the pooled class covariance matrix (computed from features centered around μ_c), and let Σ_{bg} denote the global covariance matrix (computed from features centered around μ_{bg}). Suppose that both Σ and Σ_{bg} are of full rank. The Mahalanobis distance of a given input x to class c is defined as*

$$MD_c^f(x) = (f(x) - \mu_c)\, \Sigma^{-1}\, (f(x) - \mu_c)\,.$$

Furthermore, let $MD_{bg}^f(x)$ be the Mahalanobis distance of the sample x to the distribution fitted to the entire training data, without considering class labels, defined as

$$MD_{bg}^f(x) = (f(x) - \mu_{bg})^\top\, \Sigma_{bg}^{-1}\, (f(x) - \mu_{bg})\,.$$

The Relative Mahalanobis Distance of a given x is defined as

$$RMD^f(x) = -\min_{c \in \mathcal{C}} \left(MD_c^f(x) - MD_{bg}^f(x) \right).$$

We select RMD for its state-of-the-art performance in *near-OOD* detection. This setting is critical because realistic open-world inputs are often perceptually similar to previous samples (causing spurious high query similarities) but semantically distinct (Mukhoti et al., 2023), making near-OOD more practical than *far*-OOD detection (e.g., noise vs. images) (Fort et al., 2021). Furthermore, RMD is efficient for prompt-based CL because it leverages pre-computed backbone features. We detail our implementation below.

Feature Statistics. In our rehearsal-free CL, dataset $\mathcal{D}$ is not given all at once, and cannot be stored along the task sequence $\mathfrak{S}$. To obtain the necessary means (μ_c, μ_{bg}) and covariance matrices (Σ, Σ_{bg}) under this data access constraint, we maintain task summary statistics $\mathcal{H}_{t-1} = \{\mathcal{H}^{\text{class}}_{t-1}, \mathcal{H}^{\text{global}}_{t-1}\}$ for all tasks up to t, estimated in the feature space of the pre-trained model f_θ.

- *Per-class statistics:* $\mathcal{H}^{\text{class}}_{t-1} = \{(\mu_c^{t-1}, \Sigma_c^{t-1}, n_c^{t-1}) : c \in \cup_{r=0}^{t-1}\mathcal{C}_r\}$, where $\mu_c^{t-1} \in \mathbb{R}^d$ and $\Sigma_c^{t-1} \in \mathbb{R}^{d\times d}$ denote the empirical mean and covariance of the extracted features for class c, and n_c is the total number of samples with label c in all previous tasks.
- *Global background statistics:* $\mathcal{H}^{\text{global}}_{t-1} = (\mu_{bg}^{t-1}, \Sigma_{bg}^{t-1})$, computed over all observed data.

At the end of task t, the per-class statistics $(\mu_c^t, \Sigma_c^t, n_c^t)$ and global statistics $(\mu_{bg}^t, \Sigma_{bg}^t)$ are updated using extracted features from $\mathcal{S}_t$ via standard online mean and covariance updates. The pooled class covariance matrix Σ is then derived from the per-class covariances Σ_c. When estimated covariances are not full rank, we apply a standard regularization technique by adding $\epsilon\mathbf{I}$ with a small ϵ.

Masking Function. Let

$$\text{RMD}_t(x) := \min_{c\in\cup_{r=0}^{t-1}\mathcal{C}_r} \left(\text{MD}_c^{f_\theta,t-1}(x) - \text{MD}_{bg}^{f_\theta,t-1}(x)\right),$$

where $\text{MD}^{f_\theta,t-1}$ denotes the Mahalanobis distance computed using the corresponding feature statistics from time $t-1$ (i.e., $\mathcal{H}_{t-1}$). We propose the following masking function:

$$m(x, \mathcal{H}^{\text{class}}_{t-1}, \mathcal{H}^{\text{global}}_{t-1}) = \begin{cases} 0, & \text{if } \text{RMD}_t(x) \le \tau_t \\ 1, & \text{if } \text{RMD}_t(x) > \tau_t, \end{cases}$$

where the task-specific decision threshold $\tau_t = Q_q(\{\text{RMD}_{t'}(x) : x \in \mathcal{D}_{t'},\ t' \le t\})$ is the empirical q-th quantile (e.g., $q = 0.99$) of the historical buffer.

Table 1 validates our RMD-based masking against existing routing methods by analyzing prompt retrieval scores for seen versus unseen tasks. We use the Area Under the Curve (AUC) to quantify the separability of these distributions. Ideally, unseen tasks should yield distinctly lower scores. However, standard query mechanisms exhibit poor separation (AUC as low as 0.602), demonstrating that raw similarity scores are unreliable for global prompt routing.

Table 1: Task Separability Analysis. Average AUC scores for discriminating between samples of seen and unseen tasks on the ImageNet-R and 5-Dataset benchmarks.

Query Mechanism	Task Separability (AUC ↑)	
	ImageNet-R	5-Dataset
Learnable Key (Wang et al., 2022c;b)	.602	.749
Task Centroids (Wang et al., 2022a)	.633	.698
Task Classifier (Le et al., 2025)	.749	.791
Online RMD (Ours)	**.877**	**.819**

6 Experiments

6.1 Evaluation Setup

Benchmarks: We evaluate our method on several standard continual learning benchmarks: Split CIFAR-100 (Krizhevsky et al., 2009) (10 tasks), Split ImageNet-R (Ridnik et al., 2021b) (10 tasks), 5-Datasets (Ebrahimi et al., 2020) (5 tasks), and DomainNet (Peng et al., 2019). We also evaluate on a mixed-dataset protocol, M-EMNIST and M-CifarCeleb (Ke et al., 2020).

Compared Methods. Our evaluation compares with five state-of-the-art prompting methods: L2P (Wang et al., 2022c), DualPrompt (Wang et al., 2022b), S-Prompt (Wang et al., 2022a), CODA-Prompt (Smith et al., 2023), and HiDe-Prompt (Wang et al., 2023). Following (Wang et al., 2023), we modernize the S-Prompt and use a shared classification head, denoting the resulting method as

S-Prompt++. We report Final Average Accuracy (FAA), Cumulative Average Accuracy (CAA), and Final Forgetting Measure (FFM).

Implementation: We instantiate SPR using HiDe-Prompt (Wang et al., 2023) as our primary base recipe for main result comparisons, and utilize S-Prompt++ (Wang et al., 2022a) as a simplified baseline for ablation studies. All experiments employ an ImageNet-21k pre-trained ViT-B (Ridnik et al., 2021a) backbone. Additional details are provided in in Appendix A.

6.2 Results and Analysis

Analysis of Prompt Routing Policies. We first evaluate our dynamic routing policy against the two predominant paradigms in prompt-based continual learning: global pooling and task-specific allocation. Figure 4 illustrates their comparative performance across varying training data proportions.

Our adaptive routing policy is particularly effective in data-constrained regimes, outperforming baselines by dynamically modulating parameter sharing. Task-specific methods require high data volumes to optimize disjoint prompts, failing to exploit cross-task redundancies. Conversely, unstructured global pooling causes inter-task interference that outweighs the benefits of sharing. By selectively retrieving relevant features and isolating task-specific refinements, our method maintains high plasticity for data-efficient learning, achieving the most significant improvements when training data is scarce.

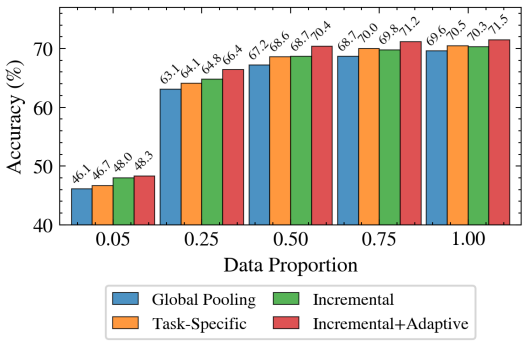

Figure 4: Impact of training data availability on routing method performance on ImageNet-R.

Extended Task Streams. We evaluate model scalability over extended horizons using the 5-Datasets and DomainNet benchmarks (Table 2), which introduce diverse inputs like new classes or shared classes across visual domains. Across both benchmarks, our method achieves the highest Final Average Accuracy (FAA). While all methods experience performance decline as sequence length increases, our improvement margin grows with task length. For example, while the baseline HiDe-Prompt performs best on the 5-task DomainNet scenario, its performance degrades much faster than ours on longer sequences. By accurately distinguishing novel information from recurring concepts, our model balances parameter isolation with knowledge reuse for long-term scalability.

Robustness on Mixed Data Streams To further evaluate the flexibility of our routing mechanism, we utilize the mixed-dataset protocol (Ke et al., 2020), denoted as M-EMNIST and M-CifarCeleb. In this setting, the task sequence is heterogeneous: some tasks introduce entirely new classes, while others present samples from the same overall distribution as previously encountered tasks, such as handwritten samples from a previously seen author or photos of a known individual.

Table 2: Performance on 5-Datasets and DomainNet at varying task scales.

Method	5-Datasets (FAA ↑)			DomainNet (FAA ↑)		
	5	10	15	5	10	15
L2P	81.84	78.45	75.22	69.58	66.82	64.15
DualPrompt	77.91	74.33	71.15	70.73	68.11	65.48
S-Prompt++	86.06	82.71	79.58	72.35	69.54	66.88
CODA-Prompt	64.18	61.52	58.83	73.24	70.62	68.03
HiDe-Prompt	93.83	90.25	87.12	**78.92**	75.68	72.85
Ours	**93.92**	**91.18**	**88.45**	78.65	**76.54**	**73.91**

Verification on Standard Benchmarks. Finally, we evaluate our method on the standard Class-Incremental Learning (CIL) benchmarks, Split CIFAR-100 and Split ImageNet-R. In these protocols, the task sequence is strictly disjoint and sequential, with ample training data for each task.

Table 3: Evaluation on task sequences with a mixture of similar and dissimilar tasks.

Method	M-EMNIST			M-CifarCeleb		
	FAA (↑)	CAA (↑)	FFM (↓)	FAA (↑)	CAA (↑)	FFM (↓)
L2P	93.12	94.50	2.11	72.44	76.88	6.40
DualPrompt	95.88	96.92	1.85	78.10	81.55	5.11
CODA-Prompt	95.50	97.10	1.90	77.65	82.40	5.45
S-Prompt++	96.12	97.25	1.40	76.90	80.22	4.92
HiDe-Prompt	98.45	98.90	0.75	86.55	88.92	2.85
Ours	**99.31**	**99.55**	**0.48**	**90.11**	**92.70**	**2.38**

Finally, we evaluate on standard Class-Incremental Learning (CIL) benchmarks (Split CIFAR-100 and Split ImageNet-R), where task sequences are disjoint

Table 4: Overall performance of continual learning. We present the final average accuracy (FAA), cumulative average accuracy (CAA) and final forgetting measure (FFM).

Method	Split CIFAR-100			Split ImageNet-R		
	FAA (↑)	CAA (↑)	FFM (↓)	FAA (↑)	CAA (↑)	FFM (↓)
L2P	83.06 ±0.17	88.25 ±0.01	6.58 ±0.40	63.65 ±0.12	67.25 ±0.02	7.51 ±0.17
DualPrompt	86.60 ±0.19	90.64 ±0.01	4.45 ±0.16	68.79 ±0.31	71.96 ±0.04	4.49 ±0.14
S-Prompt++	88.81 ±0.18	92.25 ±0.03	3.87 ±0.05	69.68 ±0.12	72.50 ±0.04	3.29 ±0.05
CODA-Prompt	86.94 ±0.63	91.57 ±0.75	4.04 ±0.18	70.03 ±0.47	74.26 ±0.24	5.17 ±0.22
HiDe-Prompt	**92.61** ±0.28	**94.03** ±0.01	**3.16** ±0.10	75.06 ±0.12	76.60 ±0.01	2.17 ±0.19
Ours	92.53 ±0.22	93.80 ±0.03	3.18 ±0.12	**75.10** ±0.15	**78.19** ±0.02	**2.15** ±0.17

and data is ample (Table 4). Our method performs comparably to the state-of-art HiDEe-prompt, verifying the stability of our routing to more generous data streams.

6.3 Prompt Selection Visualization

To investigate our performance gains, we visualize prompt selection during training and inference (Figure 5). Task-specific methods route rigidly by task ID, resulting in a uniform distribution during training. However, their inference-time selection is sparse, leaving several updated prompts unused.

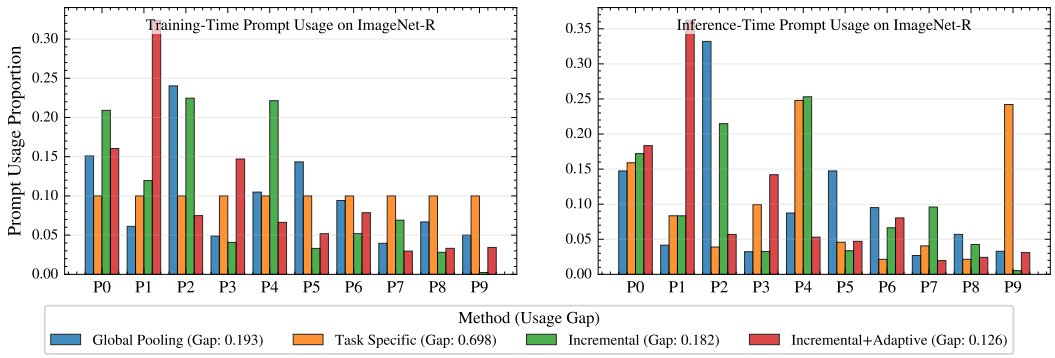

Figure 5: Prompt allocation during training (left) and inference (right) on ImageNet-R (10-Tasks).

We quantify this inefficiency via the *Usage Gap* (Δ_{usage}) defined as the L_1 distance between the proportion of samples assigned to each prompt during training (p_{train}) and inference (p_{val}):

$$\Delta_{\text{usage}} = \sum_{i=1}^{|\mathcal{P}|} |p_{\text{train},i} - p_{\text{val},i}|, \tag{10}$$

Task-specific methods exhibit a high Δ_{usage}. Conversely, global pooling baselines show a low Δ_{usage} but over-utilize a small subset of prompts, leading to capacity collapse and inter-task interference. Our adaptive routing avoids both extremes: it minimizes Δ_{usage} to ensure full parameter utilization while distributing capacity to prevent interference. This structural efficiency keeps learned representations accessible and task-appropriate, driving our demonstrated scalability.

7 Conclusion

In this work, we addressed the challenges of continual learning under per-task data scarcity and unstructured task overlap. We investigated the inherent trade-offs between global pooling and task-specific training, specifically the delicate balance between facilitating knowledge sharing and avoiding inter-task interference.

Our empirical evaluation demonstrates that SPR yields consistent improvements over state-of-the-art baselines, particularly in low-data and extended-stream scenarios. These results indicate the efficacy of OOD-informed knowledge sharing. Future work could explore integrating distribution-aware metrics directly into the query-key matching process to further improve parameter efficiency.

AI USE STATEMENT

In this work, we have not used generative AI tools for generating or cleaning datasets, developing models, formulating mathematical claims, designing research methodology or experiments, translation, supporting data analysis, implementing methods, or interpreting results. We used generative AI tools for editing software code and formatting references. We have verified all AI-assisted work, through testing and using refchecker to avoid hallucinations.

REFERENCES

Jihwan Bang, Heesu Kim, YoungJoon Yoo, Jung-Woo Ha, and Jonghyun Choi. Rainbow memory: Continual learning with a memory of diverse samples. In *2021 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 8214–8223. IEEE, 2021.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911, 2020.

Zhixiang Chi, Li Gu, Huan Liu, Yang Wang, Yuanhao Yu, and Jin Tang. Metafscil: A meta-learning approach for few-shot class incremental learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14146–14155. IEEE, 2022.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3366–3385, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *European conference on computer vision*, pp. 386–402. Springer, 2020.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in neural information processing systems*, 34:7068–7081, 2021.

Zhanxin Gao, Jun Cen, and Xiaobin Chang. Consistent prompting for rehearsal-free continual learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28463–28473. IEEE, 2024.

Zixuan Ke, Bing Liu, and Xingchang Huang. Continual learning of a mixed sequence of similar and dissimilar tasks. *Advances in neural information processing systems*, 33:18493–18504, 2020.

Gyuhak Kim, Sepideh Esmaeilpour, Changnan Xiao, and Bing Liu. Continual learning based on ood detection and task masking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3856–3866, 2022.

Gyuhak Kim, Changnan Xiao, Tatsuya Konishi, Zixuan Ke, and Bing Liu. Open-world continual learning: Unifying novelty detection and continual learning. *Artificial Intelligence*, 338:104237, 2025.

Hyunseo Koh, Dahyun Kim, Jung-Woo Ha, and Jonghyun Choi. Online continual learning on class incremental blurry task configuration with anytime inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nrGGfMbY_qK.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Minh Le, An Nguyen, Huy Nguyen, Trang Nguyen, Trang Pham, Linh Van Ngo, and Nhat Ho. Mixture of experts meets prompt-based continual learning, 2025. URL https://arxiv.org/abs/2405.14124.

Sebastian Lee, Sebastian Goldt, and Andrew Saxe. Continual learning in the teacher-student setup: Impact of task similarity. In *International Conference on Machine Learning*, pp. 6109–6119. PMLR, 2021.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

Yujie Li, Xiangkun Wang, Xin Yang, Marcello Bonsangue, Junbo Zhang, and Tianrui Li. Improving open-world continual learning under the constraints of scarce labeled data. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 1647–1658, 2025.

David Lopez-Paz and Marc-Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pp. 6467–6476, 2017.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Jishnu Mukhoti, Tsung-Yu Lin, Bor-Chun Chen, Ashish Shah, Philip HS Torr, Puneet K Dokania, and Ser-Nam Lim. Raising the bar on the evaluation of out-of-distribution detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4365–4375, 2023.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Chengwei Qin and Shafiq Joty. Continual few-shot relation learning via embedding space regularization and data augmentation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2776–2789, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.198. URL https://aclanthology.org/2022.acl-long.198/.

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*, pp. 2001–2010, 2017.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-ood detection, 2021. URL https://arxiv.org/abs/2106.09022.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. *arXiv preprint arXiv:2104.10972*, 2021a.

Tal Ridnik, Emanuel Ben-Baruch, Asaf Noy, and Lihi Zelnik-Manor. Imagenet-21k pretraining for the masses. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021b.

Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. In *arXiv preprint arXiv:1606.04671*, 2016.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, Zifeng Wang, Sayna Ebrahimi, and Hao Wang. Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 58(5):1–42, 2025.

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 *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11909–11919. IEEE, 2023.

Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12180–12189. IEEE, 2020.

Sebastian Thrun and Joseph O'Sullivan. Discovering structure in multiple learning tasks: The tc algorithm. In *ICML*, volume 96, pp. 489–497, 1996.

Eli Verwimp, Matthias De Lange, and Tinne Tuytelaars. Rehearsal revealed: The limits and merits of revisiting samples in continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9385–9394, 2021.

Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *Advances in Neural Information Processing Systems*, 36:69054–69076, 2023.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024.

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, 2022a.

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*, pp. 631–648. Springer, 2022b.

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*, pp. 139–149, 2022c.

Jaehong Yoon, Divyam Madaan, Eunho Yang, and Sung Ju Hwang. Online coreset selection for rehearsal-based continual learning. *arXiv preprint arXiv:2106.01085*, 2021.

Yaqian Zhang, Bernhard Pfahringer, Eibe Frank, Albert Bifet, Nick Jin Sean Lim, and Yunzhe Jia. A simple but strong baseline for online continual learning: Repeated augmented rehearsal. *Advances in Neural Information Processing Systems*, 35:14771–14783, 2022.

Da-Wei Zhou, Hai-Long Sun, Jingyi Ning, Han-Jia Ye, and De-Chuan Zhan. Continual learning with pre-trained models: a survey. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24, 2024. ISBN 978-1-956792-04-1. doi: 10.24963/ijcai.2024/924. URL https://doi.org/10.24963/ijcai.2024/924.

A Evaluation Setup and Implementation

A.1 Benchmark Characteristics

Split CIFAR-100 (Krizhevsky et al., 2009): Consists of 100 classes partitioned into 10 incremental tasks.

Split ImageNet-R (Ridnik et al., 2021b): Contains 200 classes featuring diverse artistic renditions, split into 10 tasks to test robustness against significant intra-class style variations.

5-Datasets (Ebrahimi et al., 2020): Contains 5 distinct datasets (CIFAR-10, MNIST, Fashion-MNIST, SVHN, and notMNIST) with a disjoint label set to measure performance under shifts in domain and label.

DomainNet (Peng et al., 2019): Contains 6 stylistically distinct datasets (Clipart, Infograph, Quickdraw, Real, Painting, Sketch) with a common label set to further benchmark performance under domain shift. Following (Smith et al., 2023), we do not include the Quickdraw dataset, and samples from different domains are shuffled.

Mixed-Dataset Protocols (M-EMNIST and M-CifarCeleb) (Ke et al., 2020): These benchmarks present a sequence of both similar and dissimilar tasks, where task similarity is emulated by sampling data from the same "client" in datasets commonly used for federated learning. We evaluate both datasets on a 20-task setting, featuring 10 similar and 10 dissimilar tasks.

A.2 Training Details and Hyperparameters

Following (Wang et al., 2023), we modernize the baseline S-Prompt (Wang et al., 2022a) architecture by utilizing a shared classification head across tasks. We denote this resulting modernized method as S-Prompt++ in our experimental comparisons.

Unless otherwise specified, we follow the public implementations and hyperparameters set by prior work (Wang et al., 2023; Smith et al., 2023). All experiments utilize an ImageNet-21k pre-trained ViT-B/16 (Ridnik et al., 2021a) backbone. The network is optimized via Adam without weight decay and batch size of 128. We use a constant learning rate of 0.005 for all methods except for CODA-prompt, which uses a cosine-decaying schedule following (Wang et al., 2023). We use a quantile threshold of $q = 0.95$ when determining the OOD threshold. We perform a grid search over the number of epochs $E \in \{5, 10, 20, 30\}$ for our low-data and mixed-data stream benchmarks, and select $E = 10$. When reporting the performance of baseline methods on preexisting benchmarks, we report the higher accuracy of our reproduced and prior published results.