C^2Prompt: Class-aware Client Knowledge Interaction for Federated Continual Learning

Kunlun Xu*
Wangxuan Institute of Computer Technology
Peking University
Beijing, China
xkl@stu.pku.edu.cn

Yibo Feng*
Wangxuan Institute of Computer Technology
Peking University
Beijing, China
2022090917012@std.uestc.edu.cn

Jiangmeng Li
University of Chinese Academy of Sciences
Beijing, China
jiangmeng2019@iscas.ac.cn

Yongsheng Qi
Inner Mongolia University of Technology
Hohhot, Inner Mongolia Autonomous Region
qys@imut.edu.cn

Jiahuan Zhou †
Wangxuan Institute of Computer Technology
Peking University
Beijing, China
jiahuanzhou@pku.edu.cn

Abstract

Federated continual learning (FCL) tackles scenarios of learning from continuously emerging task data across distributed clients, where the key challenge lies in addressing both temporal forgetting over time and spatial forgetting simultaneously. Recently, prompt-based FCL methods have shown advanced performance through task-wise prompt communication. In this study, we underscore that the existing prompt-based FCL methods are prone to class-wise knowledge coherence between prompts across clients. The class-wise knowledge coherence includes two aspects: (1) intra-class distribution gap across clients, which degrades the learned semantics across prompts, (2) inter-prompt class-wise relevance, which highlights cross-class knowledge confusion. During prompt communication, insufficient class-wise coherence exacerbates knowledge conflicts among new prompts and induces interference with old prompts, intensifying both spatial and temporal forgetting. To address these issues, we propose a novel **C**lass-aware **C**lient Knowledge Interaction (**C^2Prompt**) method that explicitly enhances class-wise knowledge coherence during prompt communication. Specifically, a local class distribution compensation mechanism (LCDC) is introduced to reduce intra-class distribution disparities across clients, thereby reinforcing intra-class knowledge consistency. Additionally, a class-aware prompt aggregation scheme (CPA) is designed to alleviate inter-class knowledge confusion by selectively strengthening class-relevant knowledge aggregation. Extensive experiments on multiple FCL benchmarks demonstrate that C^2Prompt achieves state-of-the-art performance. Our source code is available at https://github.com/zhoujiahuan1991/NeurIPS2025-C2Prompt

*Equal contribution
†Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

With the proliferation of edge computing and IoT devices [1], federated continual learning (FCL) has emerged as a critical paradigm for enabling intelligent systems to continuously learn from decentralized data streams while preserving data privacy [2; 3; 4; 5]. However, this setting presents a dual challenge: models must overcome catastrophic forgetting across sequential tasks (temporal dimension) while adapting to heterogeneous data distributions among clients (spatial dimension) [6; 7]. While traditional continual learning methods [8; 9; 10; 11; 12] and federated learning approaches [13; 14; 15; 16; 17] have made significant progress independently, their combined formulation in FCL struggles to address the superimposed forgetting effectively [6; 18; 19].

Existing FCL methods predominantly address the challenges of spatio-temporal knowledge transfer through data synthesis and parameter regularization [20]. However, data synthesis approaches [21; 22] typically depend on deep generative models trained on raw data, raising concerns regarding data privacy. In contrast, parameter regularization methods [3] attempt to balance learning and forgetting but often suffer from limited capacity to acquire new knowledge effectively. Recently, prompt-based learning [19; 23; 24] has emerged as a promising solution for FCL by maintaining task-specific prompts that store knowledge representations while leveraging a frozen pre-trained model [25; 26]. To overcome the overfitting to local distribution, some methods introduce inter-client prompt communication to improve the robustness [19]. Despite their potential, these approaches are prone to class-wise knowledge coherence during prompt communication, which comprises two aspects.

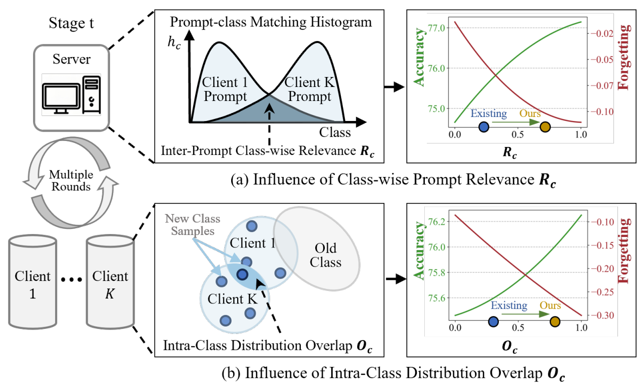

Figure 1: In FCL, class-wise knowledge coherence includes two aspects: (a) inter-prompt class-wise relevance which influences prompt aggregation in the server, (b) intra-class distribution gap (overlap) across clients which influences the locally learned semantics of each class.

First, as illustrated in Figure 1 (a), class-wise knowledge across prompts from different clients inherently varies to some extent. During prompt communication (aggregation), this divergence often results in knowledge conflicts, degrading the model's acquisition capacity. Moreover, these conflicts exacerbate forgetting, as the aggregated prompts may conflict with the historical prompts. Second, as depicted in Figure 1 (b), the intra-class distribution disparity across clients often affects the learned semantics of prompts. Certain features, although locally discriminative, may prove suboptimal from a global perspective, leading to further knowledge conflicts during prompt communication. Additionally, these locally discriminative features may be confused with historical data representations, resulting in degraded performance on previously learned tasks.

To address these challenges, we propose a novel Class-aware Client Knowledge Interaction (C^2Prompt) approach to improve inter-prompt class-wise relevance and intra-class distribution overlap simultaneously, as shown in Figure 1 (a)-(b). To achieve this, we first collect the local class distributions across clients and estimate the class-wise global distribution according to probability theory. Then, a local class distribution compensation mechanism (LCDC) is developed, which learns a set of class prompts to transfer the local semantics to the global domain, significantly improving intra-class knowledge consistency across clients. Additionally, each local prompt is recorded with its affinity with different classes. Then, a class-aware prompt aggregation scheme (CPA) is designed to exploit the class affinities to estimate the class-wise knowledge relevance across prompts and generate dynamic weights to enhance class-relevant knowledge aggregation, effectively alleviating the confusion caused by class knowledge conflict. Extensive experiments on multiple FCL benchmarks demonstrate that C^2Prompt outperforms state-of-the-art methods by large margins.

To summarize, the contributions of our paper are three-fold: (1) We present the C^2Prompt, an exemplar-free method that achieves Class-aware Client Knowledge Interaction to mitigate the temporal and spatial forgetting simultaneously. (2) A local class distribution compensation mechanism is developed to complement local distribution to improve cross-client semantic consistency. (3) A class-aware prompt aggregation scheme is proposed to enhance intra-class knowledge aggregation and alleviate the knowledge conflict via a class-wise knowledge relevant estimation mechanism. (4) The superiority of C^{2}Prompt is validated on the challenging FCL benchmarks, where our method consistently achieves remarkable state-of-the-art performance.

2 Related Work

In this section, we review three research directions and discuss the state-of-the-art works that are most relevant to this paper.

2.1 Federated Learning

FL considers a distributed machine learning paradigm where decentralized data resources are modeled collaboratively [27; 28; 29; 30]. Each client trains with its corresponding data locally, and a server aggregates the client knowledge to obtain a global model [31; 32; 33; 34]. The key challenge in FL is the data heterogeneity problem, where the data are not independently and identically distributed (non-IID) on different clients [35; 36; 37; 38]. Current FL approaches can be primarily categorized into three branches, *i.e.*, client-side regularization, server-side regularization, and synthetic data generation [39]. Client-side regularization methods aim to improve the alignment with the global model by refining local updates [40; 41; 42; 43; 44; 45; 46; 47]. Server-side regularization approaches focus on achieving better aggregation to maximize the performance of the global model [48; 49; 50; 51]. Synthetic data generation methods rely on MixUp or training a deep generation model to generate synthetic data to approximate IID conditions [52; 53] or post-train the global model [54; 55; 56]. However, these FL methods assume that all the training data are available at the same time and neglect the practical condition that the training data occur sequentially.

2.2 Continual Learning

Continual Learning (CL) aims to learn with non-stationary data and generate a unified model that can address multiple tasks [9; 57; 58]. Current CL methods are mainly divided into two categories: rehearsal-based and rehearsal-free. Rehearsal-based methods [20; 21; 22] save a subset of learned samples into a memory buffer and replay them when learning a new task. While promising performance has been achieved, they usually require a large memory cost and raise privacy concerns during long-term learning. Rehearsal-free methods dynamically expand the network or isolate parameters for different tasks, regularize the network parameters that are important to learned tasks. Recently, freezing the pre-trained backbone model and only training a subset of learnable parameters is the current mainstream approach [59; 60; 61; 8]. L2P [62] pioneeringly introduced prompt learning to CL and proposed a key-query similarity method to select prompts for each task data from a prompt pool. CODAPrompt [11] transforms prompt selection into a differential process with an attention mechanism. However, these approaches only consider alleviating temporal forgetting and struggle to address the non-IID data in the federated learning scenario [63; 64; 65].

2.3 Federated Continual Learning

In FCL, each client continuously learns from a private and incremental task stream locally and a global model aims to aggregate the spatial-temporal knowledge in a unified model [3]. Existing FCL methods primarily focus on generative replay to address the spatial and temporal forgetting [66; 18; 6; 67; 68]. However, due to the slow convergence of generation training, training a generative model introduces massive training overheads [69]. Besides, the generative models typically risk privacy leakage of local information [6]. Recently, efficient tuning-based methods have shown advanced performance in FCL. PILoRA and LoRM introduced LoRA to address FCL by learning low-rank parameters in each client and aggregates them in the server. Besides, prompt learning has shown remarkable anti-forgetting capacity due to the parameter-matching mechanism that enables mutli-task knowledge co-consistency [70; 71; 19]. However, existing methods typically neglect the knowledge conflict between individual prompts during server-side aggregation, leading to significant knowledge loss.

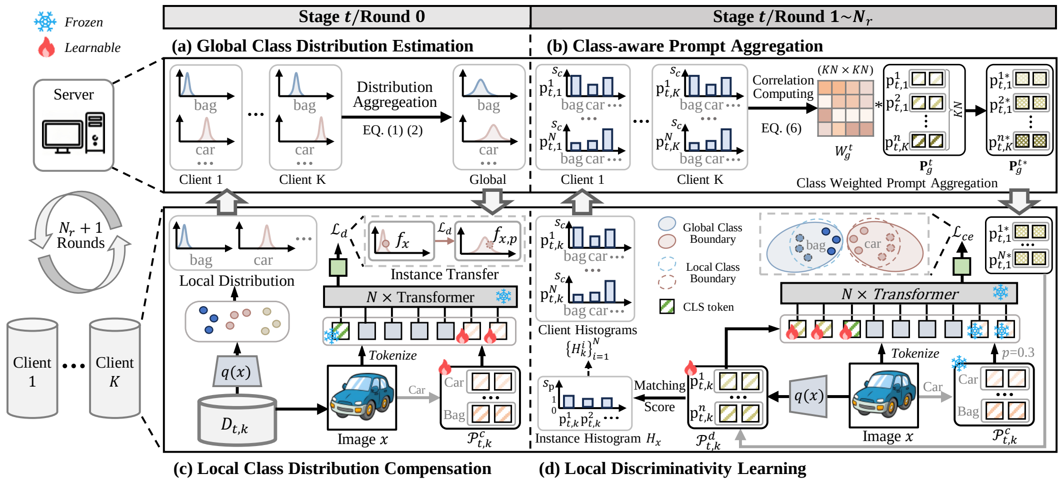

Figure 2: Overview of our C²Prompt approach. During the training stage t, given the data $D_{t,k}$ at each client k, the local class-aware feature distribution is collected and uploaded to the server to estimate the global distribution of each class. Then, the global distribution is distributed to the local clients to train client-specific class-distribution compensation prompts $\mathcal{P}^c_{t,k}$. During the later process, the discriminativity prompts $\mathcal{P}^d_{t,k}$ are introduced to learn classification-relevant knowledge, which are iteratively aggregated in the server according to the class knowledge relevance for N_r rounds.

3 Preliminaries

Problem Formulation: In FCL, a collection of K clients collaboratively learn under the coordination of a central server. Each client $k \in \{1, 2, \ldots, K\}$ sequentially learns a series of T tasks. Let $\mathcal{T}^t_k$ denote the t-th task of the k-th client, and D^t_k be its corresponding dataset. The model parameters of client k during the learning of $\mathcal{T}^t_k$ are represented as θ^t_k.

Baseline: Following recent FCL methods [19; 72; 71; 19], we adopt CODAPrompt [11] as the basic architecture for both clients and the server. For each local task $\mathcal{T}^t_k$, a set of prompts $\mathbf{P}^t_k \in \mathbb{R}^{N \times L_p \times D}$ is learned, where N is the number of local prompts, L_p is the length of each prompt, and D is the input dimension of the Vision Transformer (ViT) encoder. On the server side, a global prompt pool $\mathbf{P}_g \in \mathbb{R}^{N_g \times L_p \times D}$ is maintained, containing prompts from both previous tasks $\mathcal{T}^{pre}$ and the current task $\mathcal{T}^{cur}$. The total number of prompts in the pool is denoted as $N_g = M \times N$, where M is the number of seen tasks. For an image $\boldsymbol{x}$, its associated prompt $\mathbf{p}_x \in \mathbb{R}^{L_g \times D}$ is generated through a weighted sum of the prompts in $\mathbf{P}_g$:

$$\mathbf{p}_x = \sum_{i}^{N_g} \alpha_i [\mathbf{P}_g]_i \,, \tag{1}$$

where the weights $\boldsymbol{\alpha_x} = \{\alpha_1, \alpha_2, \ldots, \alpha_{M_g}\}$ are computed based on query-key similarity:

$$\boldsymbol{\alpha_x} = \{\gamma(q(x) \odot [\mathbf{A}_g]_1, [\mathbf{K}_g]_1), \gamma(q(x) \odot [\mathbf{A}_g]_2, [\mathbf{K}_g]_2), \ldots, \gamma(q(x) \odot [\mathbf{A}_g]_{N_g}, [\mathbf{K}_g]_{N_g})\}, \tag{2}$$

where $\gamma(\cdot, \cdot)$ represents the cosine similarity function, $\mathbf{K}_g \in \mathbb{R}^{M_g \times D}$ and $\mathbf{A}_g \in \mathbb{R}^{M_g \times D}$ are the learnable keys and attention weights of the prompts in $\mathbf{P}_g$. $\odot$ denotes the Hadamard product. For simplicity, given the one-to-one correspondence among $\mathbf{A}_g, \mathbf{K}_g$, and $\mathbf{P}_g$, we represent the global prompt pool as $\mathbf{P}_g$, encapsulating both attention and key representations.

In addition to the basic architecture of CODAPrompt, we also incorporate the knowledge distillation loss introduced by Powder [19] to enhance knowledge retention across tasks. The distillation loss is formulated as follows:

$$\mathcal{L}_{kd}(\hat{y}_{cu}, \hat{y}_{tr}) = - \sum_{k=0, k \neq y}^{K} [\hat{y}_{tr}]_k \log \frac{[\hat{y}_{cu}]_k}{[\hat{y}_{tr}]_k}, \tag{3}$$

where $\hat{y}_{cu}$ denotes the output logits of the current model, and $\hat{y}_{tr}$ represents the logits of the model at the beginning of the current communication round, containing the latest knowledge transferred from other tasks.

4 Proposed Method

In this section, we elaborate on our C^{2}Prompt which primarily consist of four modules, *i.e.*, Global Class Distribution Estimation, Class-aware Prompt Aggregation, Local Class Distribution Compensation, and Local Discriminativity Learning. An overview of C^{2}Prompt is illustrated in Figure 2, and the procedure is summarized in Algorithm 1 of Appendix B.

4.1 Global Distribution Generation

When the new stage data of a client D_k^t is given, the local distribution for each class is first computed, resulting in a distribution set $\mathcal{D}_k^t = \{(\mu_{k,i}^t, \sigma_{k,i}^t)\}_{i=1}^{|\mathcal{C}_k^t|}$, where $(\mu_{k,i}^t, \sigma_{k,i}^t)$ denotes the class center and standard deviation, and $|\mathcal{C}_k^t|$ is the number of classes for client k. For each class i, the data distribution on client k is approximated by a Gaussian distribution $\mathcal{N}(\mu_{i,k}^t, (\sigma_{i,k}^t)^2)$. Furthermore, the proportion of samples for class i at client k relative to the global sample size of class i is represented as $p_{k,i}^t$. The global mean and standard deviation are then aggregated across all clients as follows:

$$\mu_i^g = \sum_{k=1}^{K} \mu_{i,k}^t p_{k,i}^t, \tag{4}$$

$$(\sigma_i^g)^2 = \sum_{k=1}^{K} \left((\mu_{i,k}^t)^2 + (\sigma_{i,k}^t)^2\right) p_{k,i}^t - (\mu_i^g)^2 \tag{5}$$

The **theoretical derivations** of Equation 4 and Equation 5 are provided in the Appendix A.1. Once calculated, the global distribution of each class is sent back to each client for further processing.

4.2 Local Class Distribution Compensation

Upon receiving the global class distribution, local class distribution compensation prompts $\mathcal{P}_{t,k}^c = \{\mathbf{p}_i^c\}_{i=1}^{|\mathcal{C}_k^t|}$ are introduced to address the issue of local undersampling by transferring the local samples to align with the global distribution. Specifically, for each class i, a local class distribution compensation prompt is denoted as $\mathbf{p}_i^c \in \mathbb{R}^{L_c \times d}$, where L_c represents the length of $\mathbf{p}_i$. Given an input image $\boldsymbol{x}$, it is first tokenized [73] into a sequence representation $\boldsymbol{h}_{\boldsymbol{x}} \in \mathbb{R}^{L_{\boldsymbol{h}} \times d}$, where $L_{\boldsymbol{h}}$ is the sequence length. The associated local class distribution compensation prompt of $\boldsymbol{x}$, denoted $\mathbf{p}_{\boldsymbol{x}}^c$, is obtained by indexing from $\mathcal{P}_{t,k}^c$ with its label. Both $\mathbf{p}_{\boldsymbol{x}}^c$ and $\boldsymbol{h}_{\boldsymbol{x}}$ are then fed into transformer layers:

$$f_{x,p} = \boldsymbol{f}_\theta([\boldsymbol{h}_{\boldsymbol{x}}, \mathbf{p}_{\boldsymbol{x}}^c, cls]), \tag{6}$$

where $\boldsymbol{f}_\theta$ is the parameters of the pretrained ViT, cls is the [CLS] token [59], and $f_{x,p} \in \mathbb{R}^c$ is the generated feature. To ensure that $f_{x,p}$ aligns with the global class distribution, we assume that the global distribution for class i follows a Gaussian parameterization $\mathcal{N}(\mu_i^g, (\sigma_i^g)^2)$. The alignment is enforced through a distribution-based cross-entropy loss that maximizes the likelihood of $f_{x,p}$ under the global distribution:

$$\mathcal{L}_c = -\frac{1}{2}(f_{x,p} - \boldsymbol{\mu}_i^g)^\top (\boldsymbol{\Sigma}_i^g)^{-1} (f_{x,p} - \boldsymbol{\mu}_i^g), \tag{7}$$

where $\boldsymbol{\Sigma}_i^g$ is a diagonal covariance matrix with its diagonal entries equal to $(\sigma_i^g)^2$. The **theoretical derivations** of Equation 7 are provided in the Appendix A.2. Note that once $\mathcal{P}_{t,k}^c = \{\mathbf{p}_i^c\}_{i=1}^{|\mathcal{C}_k^t|}$ is trained, it is frozen during the sequential rounds of training in the current stage.

4.3 Local Discriminativity Learning

When $\mathcal{P}^c_{t,k}$ is learned, we introduce the local discriminativity prompts $\mathcal{P}^d_{t,k} = \{\mathbf{p}^i_{t,k}\}^N_{i=1}$, matrixed as $\mathbf{P}^d_{t,k}$, which corresponds to prompts of original CODAPrompt, to enable new knowledge learning. Given $\boldsymbol{x}$ and its label y, we generate instance-specific discriminativity prompt $\mathbf{p}^d_{\boldsymbol{x}}$ form $\mathbf{P}^d_{t,k}$ according to Equation 1. Besides, the local class distribution compensation prompt $\mathbf{p}^c_{\boldsymbol{x}}$ is indexed from $\mathcal{P}^c_{t,k}$ using y. Then, a cross entropy loss (CE) is introduced to optimize $\mathbf{p}^d_{\boldsymbol{x}}$:

$$\mathcal{L}_{ce} = CE\big(\mathbf{W}_k \boldsymbol{f}_\theta([\boldsymbol{h_x}, \mathbf{p}^c_{\boldsymbol{x}}, \mathbf{p}^d_{\boldsymbol{x}}, cls]), y\big), \tag{8}$$

where $\mathbf{W}_k$ is the learnable weight of the classifier of client k. Note that $\mathbf{p}^c_{\boldsymbol{x}}$ is exploited with $p = 0.5$ to sufficiently utilize the information of both local original data and the completed distributions.

At the same time, a instance histogram $H_{\boldsymbol{x}} = \{s^i_p\}^N_{i=1}$ for $\boldsymbol{x}$ is generated where $\{s^i_p\}$ denotes the similarity score between $\boldsymbol{x}$ and $\mathbf{p}^i_{t,k}$. During one round of training, we introduce a client histogram $H^i_k = \{s^j_c\}^{|\mathcal{C}^t_k|}_{j=1}$ for each prompt of stage t that records the cumulative prompt-class matching scores s^j_c, which is mathematically calculated by:

$$s^j_c = \sum_{n=1}^{|D_{t,k}|} [H_{\boldsymbol{x}_n}]_j, \tag{9}$$

where s^j_c represents the affinity between the prompt and class. Note that H^i_k can be generated online during training and requires almost no additional computing overhead. When one round of discriminative prompt training is finished, a set of client histograms $\{H^i_k\}^N_{i=1}$ for the new stage prompts is uploaded to the server.

4.4 Class-aware Prompt Aggregation

When a round of local discriminativity learning is finished, the local client histograms are collected to form a set $\mathcal{H}^t_g = \{H^i_1\}^N_{i=1} \cup \{H^i_2\}^N_{i=1} \cup \cdots \cup \{H^i_K\}^N_{i=1}$, which is matrixed as $\mathbf{H}^t_g \in \mathbb{R}^{KN\times|\mathcal{C}_t|}$. Then, an inter-prompt correlation matrix $W^t_g \in \mathbb{R}^{KN\times KN}$ is computed by

$$W^t_g = \gamma(\mathbf{H}^t_g {\mathbf{H}^t_g}^\top / \tau), \tag{10}$$

where γ is the softmax function that is conducted row-wise here, and τ is a hyperparameter to scale the similarity scores. Besides, the prompts of stage t are also collected from the clients to form a set $\mathcal{P}^t_g = \{p^i_1\}^N_{i=1} \cup \{p^i_2\}^N_{i=1} \cup \cdots \{p^i_K\}^N_{i=1}$, which is matrixed as $\mathbf{P}^t_g \in \mathbb{R}^{KN\times L_p\times d}$. Then, a Class Weighted Prompt Aggregation process is conducted by:

$$\mathbf{P}^{t*}_g = W^t_g \mathbf{P}^t_g, \tag{11}$$

where $\mathbf{P}^{t*}_g \in \mathbb{R}^{KN\times L_p\times d}$ is the updated prompts that have collected the most relevant knowledge from prompts of different clients. Then, $\mathbf{P}^{t*}_g$ is split into K prompt sets and distributed to the corresponding clients.

Training and Inference: As shown in Figure 2, during stage t, the training process consists of two phases. Firstly, Global Class Distribution Estimation and Local Class Distribution Compensation are conducted at round 0. The local distribution compensation loss $\mathcal{L}_c$ is adopted to train $\mathcal{P}^c_{t,k}$. Then, from round 1 to N_r, Class-aware Prompt Aggregation and Local Discriminativity Learning are conducted in turn. The model is optimized by an overall loss:

$$\mathcal{L}_d = \mathcal{L}_{ce} + \beta\mathcal{L}_{kd}, \tag{12}$$

where β is a hyperparameter to balance the loss components.

During inference, following previous works [19], the prompts learned on all the seen local tasks are collected to generate a prompt $\mathbf{p}_{\boldsymbol{x}}$ which is exploited to generate predictions by

$$\hat{y} = \gamma\big(\mathbf{W}_g \boldsymbol{f}_\theta([\boldsymbol{h_x}, \boldsymbol{p_x}, cls]), y\big), \tag{13}$$

where $\mathbf{W}_g$ is the global classifier by concentrates the local classifiers learned from different tasks following [19].

5 Experiments

5.1 Experimental Setups

Datasets and Metrics: We conduct the experiments on three widely used benchmarks in FCL, *i.e.*, ImageNet-R[74], DomainNet[75] and CIFAR-100[76]. To evaluate the effectiveness of different FCL methods, 6 metrics are adopted in this paper, including Average Accuracy (Avg), Average Incremental Accuracy (AIA), Forgetting Measure (FM), Forward Transfer (FT), Backward Transfer (BT), Combined Transfer (CT). The configurations of the benchmarks and the details of the metrics are presented in Appendix C.

Table 1: Result comparison on the ImageNet-R and DomainNet benchmark

Methods	Pub.	ImageNet-R						DomainNet					
		Avg↑	AIA↑	FM↓	FT↑	BT↑	CT↑	Avg↑	AIA↑	FM↓	FT↑	BT↑	CT↑
FedWEIT	*ICML2021*	71.10	74.30	1.80	-2.39	-1.83	-3.86	67.84	69.63	1.91	-2.92	-3.11	-4.97
CFeD	*IJCAI2022*	47.93	59.79	3.81	-17.67	-14.92	-29.60	42.85	60.19	1.65	-4.98	-13.32	-15.64
GLFC	*CVPR 2022*	72.96	75.21	1.10	-3.87	-1.55	-5.11	69.75	70.34	1.23	-4.08	-2.46	-6.04
Fedspace	*CVPR 2023*	72.27	73.36	2.01	-2.60	-4.91	-5.95	68.98	70.71	1.80	1.87	-4.16	-1.45
Fed-L2P	*CVPR2022*	77.88	75.03	0.41	-2.79	-0.17	-2.92	70.98	72.36	0.16	-2.18	0.10	-2.09
Fed-Dual	*ECCV2022*	76.85	74.91	0.49	-3.12	0.22	-2.95	71.90	72.15	0.16	-1.82	0.41	-1.49
Fed-CODA	*CVPR2023*	79.65	75.14	**-0.68**	-2.53	1.69	-1.18	72.47	72.84	0.01	-0.82	**0.83**	-0.15
Fed-CP	*ICML2023*	76.75	72.59	0.63	-3.16	0.00	-3.16	71.28	69.92	0.18	-2.78	0.00	-2.78
Powder	*ICML2024*	84.69	84.08	-0.54	4.48	**1.95**	6.04	75.98	77.28	0.10	1.28	0.14	1.40
PILoRA	*ECCV2024*	45.43	48.72	0.92	-5.75	-7.32	-12.54	31.22	40.76	0.55	-0.12	-0.74	-1.81
Fed-MOS	*AAAI2025*	47.67	47.08	1.40	-3.30	-0.12	-3.37	40.37	45.22	0.31	-1.43	-1.21	-2.50
LoRM	*ICLR2025*	58.00	67.78	8.71	-4.67	-9.22	-13.70	23.18	28.49	5.72	-1.32	-0.11	-1.40
C^{2}Prompt	*This Paper*	**87.20**	**85.93**	-0.36	**7.63**	1.12	**8.52**	**78.88**	**77.55**	**-0.02**	**3.87**	0.23	**4.05**

Compared Methods: We compare our proposed C^{2}Prompt with the following methods: (1) Fully-Tuning-based (FULLY) federated continual learning methods, including FedWEIT [77], CFeD [78], GLFC [7] and FedSpace [79]. (2) Efficient-Tuning-based (EFFICIENT) methods, including prompt learning approaches, FedCPrompt [71] and Powder [19]. Besides, the state-of-the-art prompt-based continual learning methods, *i.e.*, L2P [62], DualPrompt [8], CODAPrompt [11], are integrated with the well-known FedAvg [48] algorithm to make a comprehensive comparison (Fed-L2P, L2P-Dual, Fed-CODAP, Fed-CPrompt). Additionally, the LoRA-based FCL methods, including PILoRA [80] and LoRM [81], and the adapter-based continual learning method MOS [82] is integrated with FedAvg to form Fed-MOS. All experiments are implemented using official code, with the ViT-B/16 pre-trained on ImageNet-21k serving as the backbone network.

Implementation Details The settings of our discriminativity prompts follow the configuration of previous works [19], where L_p, N and d are set to 10, 8 and 768, respectively. For our class distribution compensation prompt, the prompt length L_c is set to 3 by default. The Adam optimizer with a learning rate of 0.01 is adopted during training. For all the experiments, the training and testing images are resized to 224×224. The client number K and round number for each task are set to 5 and 3, respectively. All experiments are conducted on a single Nvidia 4090 GPU.

5.2 Comparison Results

We follow the experimental setting of the previous methods [19] and the comparison results on the ImageNet-R and DomainNet are represented in Table 1, where Avg and AIA are the most important metrics indicating the long-term knowledge accumulation and progressive performance, respectively. The best and second best methods are highlighted in **Bold** and Underlined, separately.

Avg Comparison: Our C^{2}Prompt outperforms the state-of-the-art Powder, achieving improvements of **2.51%** and **2.90%** on ImageNet-R and DomainNet, respectively. These results demonstrate the superiority of our method in long-term knowledge consolidation. This is because the new knowledge acquisition capacity is significantly improved with our local class distribution compensation and class-aware discriminativity prompt aggregation designs. Besides, the accurate knowledge communication mechanism avoids the irrelevant prompts fusion that generate invalid prompts which not only semantically away from new prompts, but also conflict with historical prompts.

AIA Comparison: Our C^{2}Prompt achieves improvements of **1.85%** on ImageNet-R and also outperforms all existing approaches on DomainNet, verifying our method consistently obtains superior performance compared the existing methods across different training stages. This is attributed to the local class distribution compensation and class-aware discriminativity prompt aggregation designs that enhance robust local knowledge acquisition and improve distributed knowledge collection.

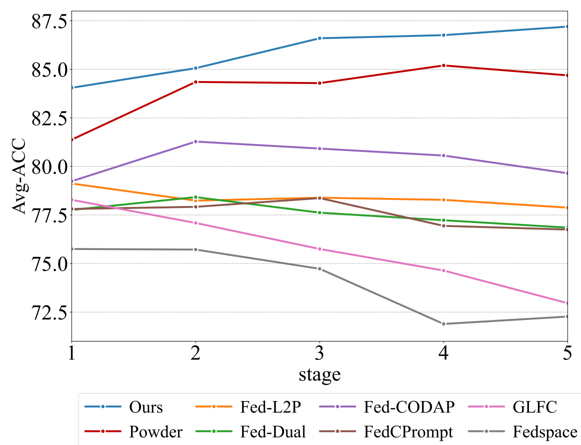

Figure 3: Avg-ACC curves on the seen tasks across training stages .

FM Comparison: Fed-CODAP, Powder and our C^{2}Prompt show a negative forgetting rate on the small-scale dataset ImageNet-R. This indicates the new tasks can facilitate historical task learning when training samples are limited. On the large-scale dataset DomainNet, only our C^{2}Prompt shows a negative forgetting rate. These results verify the effective antiforgetting capacity of our method under different conditions.

FT Comparison: Our C^{2}Prompt shows advanced forward-transfer capacity, outperforming existing methods by at least **3.15%** and **2.59%** on ImageNet-R and DomainNet, respectively. This is primarily attributed to the Global Class Distribution Estimation and Local Class Distribution Compensation designs, where the former can effectively exploit the asynchronously arriving data of the same class to generate reliable global distribution estimation. Then the estimated global distribution is exploited by the later to achieve data-level information compensation, thereby significantly improving subsequent data learning.

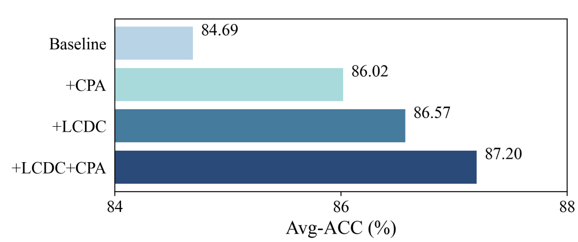

Figure 4: Ablation on the model components.

BT Comparison: Fed-CODAP, Powder and our C^{2}Prompt consistently show positive backward-transfer across both ImageNet-R and DomainNet datasets. This is because the asynchronously arriving data enable the later tasks to enhance the knowledge of previously seen classes. We observe that the backward-transfer results of C^{2}Prompt are relatively inferior to Fed-CODAP and Powder. This is because the Local Class Distribution Compensation and Class-aware Prompt Aggregation designs significantly improve the distributed data learning capacity at each stage, leaving less improvement space for seen tasks.

CT Comparison: As for the combined transfer of forward and backward, our C^{2}Prompt outperforms all existing approaches with **2.48%** and **2.65%** improvements on ImageNet-R and DomainNet, respectively. These results demonstrate that the class-aware client knowledge interaction designs in this paper effectively boost the overall learning capacity in the temporal dimension in FCL. Specifically,

Figure 5: Visualization of attention across prompt and image regions.

the Global Class Distribution Estimation effectively aggregates the distributional information across spatial and temporal data sources. Local Class Distribution Compensation module leverages the global distributional image to overcome the non-IID phenomenon across clients. Finally, Local Discriminativity Learning and Class-aware Prompt Aggregation modules effectively integrate the distributional knowledge into the prompts.

Performance Tendency Analysis: To further analyze the model learning process, we visualize the Average Accuracy (Avg-ACC) across the seen tasks during the FCL stages in Fig. 3 on the ImageNet-R benchmark. The results show that our method consistently outperforms state-of-the-art approaches across all stages. Furthermore, the Avg-ACC of C^2-Prompt exhibits a stable upward trajectory throughout the training process, whereas other methods display either declining or fluctuating trends. This advantage is attributed to our class-aware client knowledge interaction designs, which effectively extract and preserve robust knowledge over long-term training. In contrast, existing methods are more prone to knowledge conflicts during parameter aggregation in FCL, leading to performance degradation as training progresses.

5.3 Ablation Study and Additional Analysis

Ablation on components. Since Fig. 2 (a) and Fig. 2 (c) rely on each other, we present them as a unified component termed LCDC. Besides, Fig. 2 (b) and Fig. 2 (d) also rely on each other, we present them as a unified component termed CPA.

The ablation studies on LCDC and CPA are illustrated in Fig. 4, which are conducted with the ImageNet-R benchmark. When using LCDC module alone, our method obtains **1.88%** improvement compared to the baseline, verifying the effectiveness of the Global Class Distribution Estimation and Local Class Distribution Compensation mechanism. Besides, CPA achieves **1.33%** improvement compared to the baseline, demonstrating the effectiveness of our Class-aware Prompt Aggregation design. When all our modules are used together, the model performance is further improved with **2.51%** improvement. This is because LCDC and CPA achieve input-level class information compensation and feature extraction parameter-level knowledge communication, respectively. Therefore, they are complementary to each other.

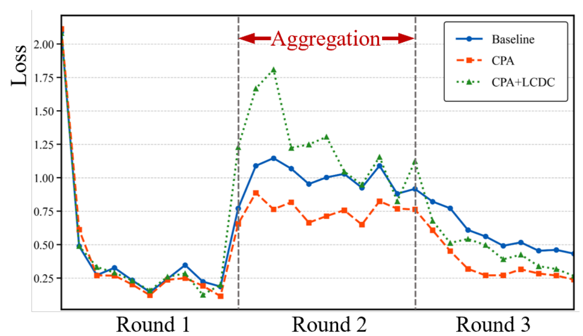

Figure 6: Visualization of loss curves.

We also visualize the loss of Baseline, CPA and CPA+LCDC in Figure 6. During round 1, different methods primarily learn with new data and converge similarly. When the first aggregation is conducted, all the methods show improved loss due to the parameter drift. CPA shows the least loss improvement due to the class-aware prompt fusion design that mitigates the knowledge conflict issue. CPA+LCDC shows a large loss improvement because the class distribution Compensation design guides the model in the early stage. During the second aggregation, the loss improvement of CPA+LCDC is significantly reduced since knowledge correlation between prompts is improved after round 2. After the training of round 3, both CPA and CPA+LCDC show significantly lower loss compared to the

Baseline. Although CPA and CPA+LCDC obtain similar final losses, the performance of CPA+LCDC is superior to CPA since LCDC improves the robustness of the learned knowledge.

Visualization of learned prompts: Figure 5 illustrates the prompt attention maps of our C^2-Prompt in comparison with state-of-the-art Powder [19]. Specifically, the prompts generated by Powder are largely dominated by class-irrelevant knowledge and exhibit limited discriminative feature extraction capacity. In contrast, the prompts generated by our method effectively focus on the discriminative regions and influence less on the class-irrelevant knowledge. These improvements are primarily attributed to our Class-aware Prompt Aggregation mechanism, which effectively alleviates conflicted knowledge fusion during prompt aggregation.

Communication Overhead: Table 2 compares the communication and parameter overhead of C^2-Prompt with state-of-the-art methods. Our approach demonstrates comparable communication and parameter costs with Powder, with only 0.6% and 6.8% increases, respectively. The additional communication overhead stems from the exchange of class distribution information between the server and clients. However, due to the sparse distribution of classes, this overhead remains minimal. The slight increase in training parameter count is attributed to the introduction of local class distribution compensation prompts, which are significantly fewer than the discriminative prompts commonly used in existing methods. Note that C^2-Prompt **does not** introduce any additional parameters or computational overhead during inference, as only the discriminative prompts are employed for testing.

Table 2: Comparison of communication and additional parameter overhead.

Methods	Communication	Parameter	
		Training	Inference
Fed-L2P	686.69MB	3.96MB	3.96MB
Fed-Dual	621.78MB	4.73MB	4.73MB
Fed-CODAP	815.63MB	11.43MB	11.43MB
Fed-CPrompt	815.63MB	11.43MB	11.43M
Powder	**493.08MB**	**2.64MB**	**2.64MB**
C^2Prompt	<u>496.01MB</u>	<u>2.82MB</u>	**2.64MB**

6 Conclusion

In this paper, we propose Class-aware Client Knowledge Interaction (C^2Prompt), which enhances the class-wise knowledge coherence between prompts across clients, significantly alleviating both temporal and spatial forgetting by mitigating the potential knowledge conflict during prompt communication. C^2Prompt introduces two kinds of prompts, local class distribution compensation prompt and local discriminativity prompt. The former transfers local class features to a global class-wise distribution to improve the intra-class semantic consistency across clients. The latter learn discrimination capacity with local data and aggregated with the ones from other clients in the server according to the class-wise affinity, enabling class-wise knowledge enhancement while alleviating conflicts. Extensive experiments on the challenging FCL benchmarks demonstrate that our method significantly outperforms the state-of-the-art, validating the effectiveness of our approach.

Limitation Discussion: Our approach requires class distribution communication at the initial round. Although this operation incurs minimal overhead due to the sparsity of distributional parameters, it introduces a minor communication cost. Furthermore, the prompt aggregation process generates client-specific prompts, slightly increasing computing and storage overhead compared to existing prompt-based methods. Nevertheless, this remains significantly more efficient than full fine-tuning approaches, as the number of learnable parameters in prompts is substantially smaller than that of the entire feature extractor.

Acknowledgement

This work was supported by the National Key R&D Program of China (2024YFA1410000), the National Natural Science Foundation of China (62376011), and the China Postdoctoral Science Foundation (2025T180417).

References

[1] T. Zhang, C. He, T. Ma, L. Gao, M. Ma, and S. Avestimehr, “Federated learning for internet of things,” in *Proceedings of the ACM conference on embedded networked sensor systems*, 2021, pp. 413–419.

[2] X. Gao, X. Yang, H. Yu, Y. Kang, and T. Li, “Fedprok: Trustworthy federated class-incremental learning via prototypical feature knowledge transfer,” in *CVPR*, 2024, pp. 4205–4214.

[3] Y. Li, W. Xu, H. Wang, Y. Qi, J. Guo, and R. Li, “Personalized federated domain-incremental learning based on adaptive knowledge matching,” in *ECCV*. Springer, 2024, pp. 127–144.

[4] M.-T. Tran, T. Le, X.-M. Le, M. Harandi, and D. Phung, “Text-enhanced data-free approach for federated class-incremental learning,” in *CVPR*, 2024, pp. 23 870–23 880.

[5] M. A. Khan, Y. Chandio, and F. Anwar, “Hydra-fl: Hybrid knowledge distillation for robust and accurate federated learning,” *NeurIPS*, vol. 37, pp. 50 469–50 493, 2024.

[6] Y. Li, Q. Li, H. Wang, R. Li, W. Zhong, and G. Zhang, “Towards efficient replay in federated incremental learning,” in *CVPR*, 2024, pp. 12 820–12 829.

[7] J. Dong, L. Wang, Z. Fang, G. Sun, S. Xu, X. Wang, and Q. Zhu, “Federated class-incremental learning,” in *CVPR*, 2022, pp. 10 164–10 173.

[8] Z. Wang, Z. Zhang, S. Ebrahimi, R. Sun, H. Zhang, C.-Y. Lee, X. Ren, G. Su, V. Perot, J. Dy *et al.*, “Dualprompt: Complementary prompting for rehearsal-free continual learning,” in *ECCV*. Springer, 2022, pp. 631–648.

[9] Q. Li, Y. Peng, and J. Zhou, “Fcs: Feature calibration and separation for non-exemplar class incremental learning,” in *CVPR*, 2024, pp. 28 495–28 504.

[10] K. Xu, C. Jiang, P. Xiong, Y. Peng, and J. Zhou, “Dask: Distribution rehearsing via adaptive style kernel learning for exemplar-free lifelong person re-identification,” in *AAAI*, vol. 39, no. 9, 2025, pp. 8915–8923.

[11] J. S. Smith, L. Karlinsky, V. Gutta, P. Cascante-Bonilla, D. Kim, A. Arbelle, R. Panda, R. Feris, and Z. Kira, “Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning,” in *CVPR*, 2023, pp. 11 909–11 919.

[12] J. Zhou, K. Xu, F. Zhuo, X. Zou, and Y. Peng, “Distribution-aware knowledge aligning and prototyping for non-exemplar lifelong person re-identification,” *IEEE TPAMI*, 2025.

[13] M. Li, X. Zhang, Q. Wang, T. Liu, R. Wu, W. Wang, F. Zhuang, H. Xiong, and D. Yu, “Resource-aware federated self-supervised learning with global class representations,” *NeurIPS*, vol. 37, pp. 10 008–10 035, 2024.

[14] W. Huang, M. Ye, and B. Du, “Learn from others and be yourself in heterogeneous federated learning,” in *CVPR*, 2022, pp. 10 143–10 153.

[15] J. Shi, S. Zheng, X. Yin, Y. Lu, Y. Xie, and Y. Qu, “Clip-guided federated learning on heterogeneity and long-tailed data,” in *AAAI*, vol. 38, no. 13, 2024, pp. 14 955–14 963.

[16] L. Wang, J. Bian, L. Zhang, C. Chen, and J. Xu, “Taming cross-domain representation variance in federated prototype learning with heterogeneous data domains,” in *NeurIPS*.

[17] M. Morafah, V. Kungurtsev, H. M. Chang, C. Chen, and B. Lin, “Towards diverse device heterogeneous federated learning via task arithmetic knowledge integration,” in *NeurIPS*.

[18] S. Babakniya, Z. Fabian, C. He, M. Soltanolkotabi, and S. Avestimehr, “A data-free approach to mitigate catastrophic forgetting in federated class incremental learning for vision tasks,” *NeurIPS*, vol. 36, pp. 66 408–66 425, 2023.

[19] H. Piao, Y. Wu, D. Wu, and Y. Wei, “Federated continual learning via prompt-based dual knowledge transfer,” in *ICML*, 2024.

[20] R. Aljundi, E. Belilovsky, T. Tuytelaars, L. Charlin, M. Caccia, M. Lin, and L. Page-Caccia, "Online continual learning with maximal interfered retrieval," in *NeurIPS*, vol. 32. Curran Associates, Inc., 2019.

[21] R. Aljundi, M. Lin, B. Goujaud, and Y. Bengio, "Gradient based sample selection for online continual learning," in *NeurIPS*, vol. 32. Curran Associates, Inc., 2019.

[22] J. Bang, H. Kim, Y. Yoo, J.-W. Ha, and J. Choi, "Rainbow memory: Continual learning with a memory of diverse samples," in *CVPR*, 2021, pp. 8214–8223.

[23] K. Xu, X. Zou, G. Hua, and J. Zhou, "Componential prompt-knowledge alignment for domain incremental learning," in *ICML*, 2025.

[24] M. A. Ma'sum, M. Pratama, L. Liu, H. Habibullah, and R. Kowalczyk, "Federated few-shot class-incremental learning," in *ICLR*, 2025.

[25] C. Zhang, K. Xu, Z. Liu, Y. Peng, and J. Zhou, "Scap: Transductive test-time adaptation via supportive clique-based attribute prompting," in *CVPR*, 2025.

[26] Z. Liu, K. Xu, B. Su, X. Zou, Y. Peng, and J. Zhou, "Stop: Integrated spatial-temporal dynamic prompting for video understanding," in *CVPR*, 2025.

[27] X. Liao, W. Liu, P. Zhou, F. Yu, J. Xu, J. Wang, W. Wang, C. Chen, and X. Zheng, "Foogd: Federated collaboration for both out-of-distribution generalization and detection," *NeurIPS*, vol. 37, pp. 132 908–132 945, 2024.

[28] J. Chen, J. Xue, Y. Wang, Z. Liu, and L. Huang, "Classifier clustering and feature alignment for federated learning under distributed concept drift," in *NeurIPS*.

[29] Y. Allouah, A. El Mrini, R. Guerraoui, N. Gupta, and R. Pinot, "Fine-tuning personalization in federated learning to mitigate adversarial clients," *NeurIPS*, vol. 37, pp. 100 816–100 844, 2024.

[30] Z. Qi, L. Meng, Z. Li, H. Hu, and X. Meng, "Cross-silo feature space alignment for federated learning on clients with imbalanced data," in *AAAI*, 2025, pp. 19 986–19 994.

[31] Z. Qi, L. Meng, Z. Chen, H. Hu, H. Lin, and X. Meng, "Cross-silo prototypical calibration for federated learning with non-iid data," in *ACM MM*, 2023, pp. 3099–3107.

[32] B. Pan, W. Huang, and Y. Shi, "Federated learning from vision-language foundation models: Theoretical analysis and method," *NeurIPS*, vol. 37, pp. 30 590–30 623, 2024.

[33] P.-Y. Weng, M. Hoang, L. Nguyen, M. T. Thai, L. Weng, and N. Hoang, "Probabilistic federated prompt-tuning with non-iid and imbalanced data," *NeurIPS*, vol. 37, pp. 81 933–81 958, 2024.

[34] F. Wu, X. Wang, Y. Wang, T. Liu, L. Su, and J. Gao, "Fiarse: Model-heterogeneous federated learning via importance-aware submodel extraction," in *NeurIPS*.

[35] H. Zhang, C. Li, N. Kan, Z. Zheng, W. Dai, J. Zou, and H. Xiong, "Improving generalization in federated learning with model-data mutual information regularization: A posterior inference approach," *NeurIPS*, vol. 37, pp. 136 646–136 678, 2024.

[36] C. Mclaughlin and L. Su, "Personalized federated learning via feature distribution adaptation," *NeurIPS*, vol. 37, pp. 77 038–77 059, 2024.

[37] P. M Ghari and Y. Shen, "Personalized federated learning with mixture of models for adaptive prediction and model fine-tuning," *NeurIPS*, vol. 37, pp. 92 155–92 183, 2024.

[38] H. Li, W. Huang, J. Wang, and Y. Shi, "Global and local prompts cooperation via optimal transport for federated learning," in *CVPR*, 2024, pp. 12 151–12 161.

[39] J. Zhang, C. Shan, and J. Han, "Fedgmkd: An efficient prototype federated learning framework through knowledge distillation and discrepancy-aware aggregation," *NeurIPS*, vol. 37, pp. 118 326–118 356, 2024.

[40] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[41] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "Scaffold: Stochastic controlled averaging for federated learning," in *ICML*. PMLR, 2020, pp. 5132–5143.

[42] M. Mendieta, T. Yang, P. Wang, M. Lee, Z. Ding, and C. Chen, "Local learning matters: Rethinking data heterogeneity in federated learning," in *CVPR*, 2022, pp. 8397–8406.

[43] D. A. E. Acar, Y. Zhao, R. M. Navarro, M. Mattina, P. N. Whatmough, and V. Saligrama, "Federated learning based on dynamic regularization," *arXiv preprint arXiv:2111.04263*, 2021.

[44] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, and C.-Z. Xu, "Feddc: Federated learning with non-iid data via local drift decoupling and correction," in *CVPR*, 2022, pp. 10 112–10 121.

[45] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in *CVPR*, 2021, pp. 10 713–10 722.

[46] T. Lin, L. Kong, S. U. Stich, and M. Jaggi, "Ensemble distillation for robust model fusion in federated learning," *NeurIPS*, vol. 33, pp. 2351–2363, 2020.

[47] L. Meng, Z. Qi, L. Wu, X. Du, Z. Li, L. Cui, and X. Meng, "Improving global generalization and local personalization for federated learning," *TNNLS*, vol. 36, 2024.

[48] H. B. McMahan, E. Moore, D. Ramage, and B. A. y Arcas, "Federated learning of deep networks using model averaging," *arXiv preprint arXiv:1602.05629*, vol. 2, no. 2, pp. 15–18, 2016.

[49] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.

[50] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *NeurIPS*, vol. 33, pp. 7611–7623, 2020.

[51] X. Li, M. JIANG, X. Zhang, M. Kamp, and Q. Dou, "Fedbn: Federated learning on non-iid features via local batch normalization," in *ICLR*, 2021.

[52] Z. Zhu, J. Hong, and J. Zhou, "Data-free knowledge distillation for heterogeneous federated learning," in *ICML*. PMLR, 2021, pp. 12 878–12 889.

[53] T. Guo, S. Guo, and J. Wang, "Pfedprompt: Learning personalized prompt for vision-language models in federated learning," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 1364–1374.

[54] T. Yoon, S. Shin, S. J. Hwang, and E. Yang, "Fedmix: Approximation of mixup under mean augmented federated learning," in *ICLR*.

[55] S. Hu, J. Goetz, K. Malik, H. Zhan, Z. Liu, and Y. Liu, "Fedsynth: Gradient compression via synthetic data in federated learning," in *NeurIPS*.

[56] Y. Xiong, R. Wang, M. Cheng, F. Yu, and C.-J. Hsieh, "Feddm: Iterative distribution matching for communication-efficient federated learning," in *CVPR*, 2023, pp. 16 323–16 332.

[57] K. Xu, X. Zou, Y. Peng, and J. Zhou, "Distribution-aware knowledge prototyping for non-exemplar lifelong person re-identification," in *CVPR*, 2024, pp. 16 604–16 613.

[58] K. Xu, Z. Liu, X. Zou, Y. Peng, and J. Zhou, "Long short-term knowledge decomposition and consolidation for lifelong person re-identification," *TPAMI*, 2025.

[59] Z. Liu, Y. Peng, and J. Zhou, "Compositional prompting for anti-forgetting in domain incremental learning," *IJCV*, pp. 1–18, 2024.

[60] L. Wang, J. Xie, X. Zhang, M. Huang, H. Su, and J. Zhu, "Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality," *NeurIPS*, vol. 36, 2023.

[61] K. Xu, H. Zhang, Y. Li, Y. Peng, and J. Zhou, "Mitigate catastrophic remembering via continual knowledge purification for noisy lifelong person re-identification," in *ACM MM*, 2024, pp. 5790–5799.

[62] Z. Wang, Z. Zhang, C.-Y. Lee, H. Zhang, R. Sun, X. Ren, G. Su, V. Perot, J. Dy, and T. Pfister, "Learning to prompt for continual learning," in *CVPR*, 2022, pp. 139–149.

[63] S. Bai, J. Zhang, S. Guo, S. Li, J. Guo, J. Hou, T. Han, and X. Lu, "Diprompt: Disentangled prompt tuning for multiple latent domain generalization in federated learning," in *CVPR*, 2024, pp. 27 284–27 293.

[64] K. Xu, X. Zou, and J. Zhou, "Lstkc: Long short-term knowledge consolidation for lifelong person re-identification," in *AAAI*, vol. 38, no. 14, 2024, pp. 16 202–16 210.

[65] C.-M. Feng, B. Li, X. Xu, Y. Liu, H. Fu, and W. Zuo, "Learning federated visual prompt in null space for mri reconstruction," in *CVPR*, 2023, pp. 8064–8073.

[66] J. Liang, J. Zhong, H. Gu, Z. Lu, X. Tang, G. Dai, S. Huang, L. Fan, and Q. Yang, "Diffusion-driven data replay: A novel approach to combat forgetting in federated class continual learning," in *ECCV*. Springer, 2024, pp. 303–319.

[67] D. Qi, H. Zhao, and S. Li, "Better generative replay for continual federated learning," in *ICLR*, 2023.

[68] J. Zhang, C. Chen, W. Zhuang, and L. Lyu, "Target: Federated class-continual learning via exemplar-free distillation," in *ICCV*, 2023, pp. 4782–4793.

[69] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *ICML*. PMLR, 2017, pp. 2642–2651.

[70] H. Yu, X. Yang, X. Gao, Y. Kang, H. Wang, J. Zhang, and T. Li, "Personalized federated continual learning via multi-granularity prompt," in *SIGKDD*, 2024, pp. 4023–4034.

[71] G. Bagwe, X. Yuan, M. Pan, and L. Zhang, "Fed-CPrompt: Contrastive prompt for rehearsal-free federated continual learning," in *Federated Learning and Analytics in Practice: Algorithms, Systems, Applications, and Opportunities*, 2023.

[72] S. Halbe, J. S. Smith, J. Tian, and Z. Kira, "Hepco: Data-free heterogeneous prompt consolidation for continual federated learning," in *NeurIPS*.

[73] Q. Li, K. Xu, Y. Peng, and J. Zhou, "Exemplar-free lifelong person re-identification via prompt-guided adaptive knowledge consolidation," *IJCV*, pp. 1–16, 2024.

[74] D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo *et al.*, "The many faces of robustness: A critical analysis of out-of-distribution generalization," in *ICCV*, 2021, pp. 8340–8349.

[75] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *ICCV*, 2019, pp. 1406–1415.

[76] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[77] J. Yoon, W. Jeong, G. Lee, E. Yang, and S. J. Hwang, "Federated continual learning with weighted inter-client transfer," in *ICML*. PMLR, 2021, pp. 12 073–12 086.

[78] Y. Ma, Z. Xie, J. Wang, K. Chen, and L. Shou, "Continual federated learning based on knowledge distillation." in *IJCAI*, 2022, pp. 2182–2188.

[79] D. Shenaj, M. Toldo, A. Rigon, and P. Zanuttigh, "Asynchronous federated continual learning," in *CVPR*, 2023, pp. 5055–5063.

[80] H. Guo, F. Zhu, W. Liu, X.-Y. Zhang, and C.-L. Liu, "Pilora: Prototype guided incremental lora for federated class-incremental learning," in *ECCV*. Springer, 2024, pp. 141–159.

[81] R. Salami, P. Buzzega, M. Mosconi, J. Bonato, L. Sabetta, and S. Calderara, "Closed-form merging of parameter-efficient modules for federated continual learning," *arXiv preprint arXiv:2410.17961*, 2024.

[82] H.-L. Sun, D.-W. Zhou, H. Zhao, L. Gan, D.-C. Zhan, and H.-J. Ye, "Mos: Model surgery for pre-trained model-based class-incremental learning," in *AAAI*, vol. 39, no. 19, 2025, pp. 20 699–20 707.

[83] A. Douillard, M. Cord, C. Ollion, T. Robert, and E. Valle, "Podnet: Pooled outputs distillation for small-tasks incremental learning," in *ECCV*. Springer, 2020, pp. 86–102.

[84] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *ECCV*, 2018, pp. 532–547.

[85] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *NeurIPS*, vol. 30, 2017.

[86] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[87] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE TPAMI*, vol. 40, no. 12, pp. 2935–2947, 2017.

[88] M. S. Matena and C. A. Raffel, "Merging models with fisher-weighted averaging," *NeurIPS*, vol. 35, pp. 17 703–17 716, 2022.

[89] X. Jin, X. Ren, D. Preotiuc-Pietro, and P. Cheng, "Dataless knowledge fusion by merging weights of language models," *arXiv preprint arXiv:2212.09849*, 2022.

[90] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-iid data," *NeurIPS*, vol. 34, pp. 5972–5984, 2021.

[91] Y. Tan, G. Long, L. Liu, T. Zhou, Q. Lu, J. Jiang, and C. Zhang, "Fedproto: Federated prototype learning across heterogeneous clients," in *AAAI*, vol. 36, no. 8, 2022, pp. 8432–8440.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have analyzed the limitations of this method over computing and communication overheads.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We have provided a theoretical analysis of our distribution operation with a complete theoretical derivation proof.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper fully discloses all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have attached our source code and data access links in the supplementary materials, with sufficient instructions to faithfully reproduce the main experimental results.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We have specified all the training and test details necessary to understand the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported due to computational constraints. However, all experiments are conducted with consistent benchmark configurations and fixed random seeds, ensuring reproducibility. The significant and consistent improvements across multiple benchmarks substantiate the effectiveness of our method.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provide sufficient information on the computer resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The Broader impacts has been discussed in the Appendix.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: They are properly credited and respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: This core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Theoretical Justification of Distribution Operations

A.1 Server-side Distribution Aggregation

Federated continual learning (FCL) involves decentralized clients collaboratively learning over sequential tasks. To enhance cross-client coherence, we estimate the global class distributions on the server by aggregating local class distributions from each client. Given local class distributions $\mathcal{N}(\mu_1, \sigma_1^2), \mathcal{N}(\mu_2, \sigma_2^2), \ldots, \mathcal{N}(\mu_n, \sigma_n^2)$ with probability density functions $f_1(x), f_2(x), \ldots, f_n(x)$, the global distribution is defined using the clients' sample frequencies $p_1, p_2, \ldots, p_n$, satisfying $\sum_{i=1}^{n} p_i = 1$.

The global class mean is calculated as:

$$\mu = \int \left[p_1 f_1(x) + p_2 f_2(x) + \cdots + p_n f_n(x)\right] x \, dx = p_1 \mu_1 + p_2 \mu_2 + \cdots + p_n \mu_n \tag{14}$$

The global class variance is expressed as:

$$\sigma^2 = \int \left[p_1 f_1(x) + p_2 f_2(x) + \cdots + p_n f_n(x)\right] x^2 \, dx - \mu^2 \tag{15}$$

Then we have

$$\sigma^2 = p_1 \left(\sigma_1^2 + \mu_1^2\right) + p_2 \left(\sigma_2^2 + \mu_2^2\right) + \cdots + p_n \left(\sigma_n^2 + \mu_n^2\right) - \mu^2 \tag{16}$$

This aggregation provides a comprehensive global distribution for each class, continuously updated as new tasks arrive. These global statistics are then communicated back to clients to guide local prompt optimization, enhancing semantic consistency.

A.2 Local Class Distribution Compensation Loss

Upon receiving global class distributions, each client optimizes local class distribution compensation prompts. The derivation of the loss function for the prompt is as follows:

For class i, the global class distribution is represented as $\mathcal{N}(\mu_g^i, \mathbf{\Sigma}_g^i)$. Specifically, for feature vector $X \in \mathbb{R}^d$ of class i, its probability density is:

$$p(X \mid \mu_i, \mathbf{\Sigma}_i) = \frac{1}{(2\pi)^{d/2} |\mathbf{\Sigma}_i|^{1/2}} \cdot e^{-\frac{1}{2}(X-\mu_i)^\top \mathbf{\Sigma}_i^{-1}(X-\mu_i)} \tag{17}$$

where $\mu_i \in \mathbb{R}^d$ is the mean vector and $\mathbf{\Sigma}_i \in \mathbb{R}^{d \times d}$ is a positive definite symmetric covariance matrix. When assuming independent feature dimensions, the covariance matrix reduces to diagonal form:

$$\mathbf{\Sigma}_i = \text{diag}(\sigma_{i1}^2, \sigma_{i2}^2, \ldots, \sigma_{id}^2) \tag{18}$$

with determinant and inverse matrix given by:

$$|\mathbf{\Sigma}_i| = \prod_{j=1}^{d} \sigma_{ij}^2 \quad \text{and} \quad \mathbf{\Sigma}_i^{-1} = \text{diag}\left(\frac{1}{\sigma_{i1}^2}, \frac{1}{\sigma_{i2}^2}, \ldots, \frac{1}{\sigma_{id}^2}\right) \tag{19}$$

According to (17), the exponent expands to:

$$(X - \mu_i)^\top \mathbf{\Sigma}_i^{-1} (X - \mu_i) = \sum_{j=1}^{d} \frac{(X_j - \mu_{ij})^2}{\sigma_{ij}^2} \tag{20}$$

Thus, the probability density function decomposes as:

$$p_i(X|\mu_i, \sigma_i^2) = \frac{1}{(2\pi)^{d/2} \left(\prod_{j=1}^{d} \sigma_{ij}^2\right)^{1/2}} \cdot e^{-\frac{1}{2} \sum\limits_{j=1}^{d} \frac{(X_j - \mu_{ij})^2}{\sigma_{ij}^2}} \tag{21}$$

In federated learning, client-generated features $f_{x,p}$ should align with the server's global class distribution $\mathcal{N}(\mu_g^i, \mathbf{\Sigma}_g^i)$. The optimization objective becomes maximizing the log-likelihood:

$$\mathcal{L}_c = \log p(f_{x,p}|\mu_g^i, \mathbf{\Sigma}_g^i) \tag{22}$$

Substituting (17) and expanding:

$$\log p(f_{x,p}|\mu_g^i, \mathbf{\Sigma}_g^i) = -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}_g^i| - \frac{1}{2}(f_{x,p} - \mu_g^i)^\top(\mathbf{\Sigma}_g^i)^{-1}(f_{x,p} - \mu_g^i) \tag{23}$$

Minimizing the negative log-likelihood loss:

$$\mathcal{L}_c = -\log p(f_{x,p}|\mu_g^i, \mathbf{\Sigma}_g^i) = \frac{d}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{\Sigma}_g^i| + \frac{1}{2}(f_{x,p} - \mu_g^i)^\top(\mathbf{\Sigma}_g^i)^{-1}(f_{x,p} - \mu_g^i) \tag{24}$$

Since the first two constant terms can be omitted for optimization, $\mathcal{L}_c$ can be simplified to:

$$\mathcal{L}_c = \frac{1}{2}(f_{x,p} - \mu_g^i)^\top(\mathbf{\Sigma}_g^i)^{-1}(f_{x,p} - \mu_g^i) \tag{25}$$

This objective encourages local features to match the global class distribution, effectively reducing inter-client distribution gaps and enhancing semantic coherence during federated updates.

A.3 Theoretical Implications

Overall, the proposed distribution operations theoretically guarantee smoother knowledge alignment across clients by optimizing class-wise distribution coherence. This process stabilizes global knowledge representations throughout continual learning, improving the robustness of the learned knowledge and improving the aggregation compatibility between prompts from different clients, effectively mitigating spatial and temporal forgetting.

B Algorithm of the proposed approach

The overall process of our C^{2}Prompt is shown in Algorithm 1.

C Details of the datasets and evaluation metrics

C.1 Datasets

We use three image datasets commonly utilized in Federated Continual Learning (FCL) to evaluate our method: ImageNet-R, DomainNet, and CIFAR-100. ImageNet-R consists of 30,000 images from 200 categories, including challenging samples from ImageNet and newly collected samples with various styles. The dataset is divided into a training set with 24,000 images and a test set with 6,000 images, and 20% of the training set is selected as a validation set for tuning model parameters. DomainNet is a large dataset containing 600,000 images and 345 categories, spanning six different domains. CIFAR-100 contains 50,000 training and 10,000 test-colored images for 100 classes, respectively.

C.2 Configuration of Federated Continual Learning Benchmarks

The benchmark configuration of this paper follows previous FCL method Powder [19]. Based on transferability (tasks have class overlap and each task contains only a small portion of each class's data) and asynchrony, for ImageNet-R, each task randomly selects 20 classes (20% samples per class), distributed randomly across clients with varying round durations. For DomainNet, each task randomly selects 35 classes (2% samples per class due to closeness to pre-trained distribution, others same as ImageNet-R). We control task overlap by randomly selecting classes with the least overlap to

Algorithm 1 Local Class Distribution Compensation and Global Class Distribution Estimation

Input: Stage t data $\mathcal{D}^t = \{\mathcal{D}_k^t\}_{k=1}^K$

Output: Global prompt pool $\mathbf{P}_g^t$

Initialize $\mathcal{P}_d^{t,k*} = None$
for each round $r = 0$ to N_r **do**
  if round r = 0 **then**

    # Local Class Distribution Compensation ($\boldsymbol{LCDC}$)
    for each client k **do**
      Compute local class statistics $(\mu_{k,i}^t, \sigma_{k,i}^t)$ for each class i
      Upload $(\mu_{k,i}^t, \sigma_{k,i}^t)$ to the server
    end for

    # Global Class Distribution Estimation
    Estimate global class center $\mu_i^g = \sum_{k=1}^K \mu_{i,k}^t p_{k,i}^t$, Eq. 4
    Estimate global class variance $(\sigma_i^g)^2 = \sum_{k=1}^K \left((\mu_{i,k}^t)^2 + (\sigma_{i,k}^t)^2\right) p_{k,i}^t - (\mu_i^g)^2$, Eq. 5
    Distribute the global distribution to the corresponding clients

    # Back to ($\boldsymbol{LCDC}$)
    for each client $k = 1$ to K **do**
      Initialize local class distribution complension prompts $\mathcal{P}_{t,k}^c = \{p_i^c\}_{i=1}^{|C_t^k|}$
      For input x, obtain $f_{x,p} = f_\theta([\boldsymbol{h}_x, \mathbf{p}_{\boldsymbol{x}}^c, \text{[CLS]}])$, Eq. 6
      Update $\mathcal{P}_{t,k}^c$ using $\mathcal{L}_c = -\frac{1}{2}(f_{x,p} - \boldsymbol{\mu}_i^g)^\top (\boldsymbol{\Sigma}_i^g)^{-1}(f_{x,p} - \boldsymbol{\mu}_i^g)$, Eq. 7
    end for
    Froze prompt $\mathcal{P}_{t,k}^c$
  end if

  # Local Discriminativity Learning
  for each client $k = 1$ to K **do**
    if $\mathcal{P}_{t,k}^{d*} \neq None$ **then**
      Initialize local discriminativity prompts $\mathcal{P}_{t,k}^d = \{p_i^d\}_{i=1}^N$ with $\mathcal{P}_{t,k}^{d*}$
    end if
    For each input x, pbtain $\mathbf{p}_{\boldsymbol{x}}^d$, $\mathbf{p}_{\boldsymbol{x}}^c$ and $H_{\boldsymbol{x}}$
    Update using $\mathcal{L}_{ce} = CE\big(\mathbf{W}_k \boldsymbol{f}_\theta([\boldsymbol{h}_{\boldsymbol{x}}, \mathbf{p}_{\boldsymbol{x}}^c, \mathbf{p}_{\boldsymbol{x}}^d, cls]), y\big)$, Eq. 8
    Obtain client histogram $H_k^i = \{s_c^j\}_{j=1}^{|\mathcal{C}_k^t|}$ for prompt p_d^i, where $s_c^j = \sum_{n=1}^{|D_{t,k}|}[H_{\boldsymbol{x}_n}]_j$, Eq. 9
    Upload $\{H_k^i\}_{i=1}^N$ and $\mathcal{P}_{t,k}^d$ to the server
  end for

  # Class-aware Prompt Aggregation ($\boldsymbol{CPA}$)
  Server collects all client histograms to form $\mathbf{H}_g^t \in \mathbb{R}^{KN \times |\mathcal{C}_t|}$
  Server collects all client discriminativity prompts to form $\mathbf{P}_g^t$
  Compute inter-prompt attention: $W_g^t = \gamma(\mathbf{H}_g^t \mathbf{H}_g^{t^\top} / \tau)$, Eq. 10
  Update prompts: $\mathbf{P}_g^{t*} = W_g^t \mathbf{P}_g^t$, Eq. 11
  Distribute $\mathcal{P}_{t,k*}^d$ to corresponding clients
end for
Return $\mathbf{P}_g^t$

study FCL performance under different task correlations. Unlike the common Dirichlet distribution method in FL, we avoid it here because in FCL, class sets of different tasks vary greatly, making it hard to control similarity with it. The setup details for the CIFAR-100 dataset are the same as those for ImageNet-R. Simultaneously, we set five clients for training, each executing distinct tasks. Furthermore, we organize the training process into phases, each consisting of three communication rounds. At the beginning of each phase, 40% of the clients are selected to initiate learning on new tasks. To ensure the fairness of the results, we keep the optimizer, learning rate, and local training epochs consistent with those of the Powder method when training the classification prompts. Our LCDC is trained using the Adam optimizer when new tasks arrive, and the trained prompt is only used for the training of the formal classification prompt and not for the testing phase.

C.3 Evaluation Metrics

We evaluate the effectiveness of our method by adapting seven metrics, including the Average accuracy of all tasks (Avg), Average Incremental Accuracy (AIA)[83], Forgetting Measure (FM)[84], Forward Transfer (FT)[85], Backward Transfer (BT)[85], Combined Transfer (CT), Final Average Accuracy (FAA).

Average accuracy of all tasks (Avg) This metric measures the average accuracy of the final model across all tasks, computed as

$$\text{Avg} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}_c^t \in \mathcal{T}} a_{c,\max(\mathcal{R})}^t$$

where $\mathcal{T}$ denotes the set of all tasks during the Federated Continual Learning (FCL) process, and $a_{c,\max(\mathcal{R})}^t$ denotes the final accuracy of task $\mathcal{T}_c^t$ (i.e., the accuracy on this task when training concludes).

Average Incremental Accuracy (AIA) This metric measures the average accuracy over the FCL process, computed as

$$\text{AIA} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \sum_{\mathcal{T}_c^t \in \mathcal{T}_r} a_{c,r}^t$$

where $\mathcal{R}$ denotes the set of rounds with task switch, $\mathcal{T}_r$ denotes the set of existing tasks at round r, and $a_{c,r}^t$ denotes the accuracy of $\mathcal{T}_c^t$ at round r.

Forgetting Measure (FM) Forgetting is measured by the difference between the highest historical accuracy and the current accuracy of a task. This metric quantifies the model's memory stability by the average forgetting over the FCL process, computed as

$$\text{FM} = \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \left(\sum_{\mathcal{T}_c^t \in \mathcal{T}_r} a_{c,r}^t - \tilde{a}_{c,r}^t \right)$$

where $\tilde{a}_{c,r}$ denotes the max accuracy of $\mathcal{T}_c^t$ before round r.

Forward Transfer (FT) This metric assesses the model's ability to transfer knowledge into a task, from both previously learned tasks and other currently learned tasks, computed as

$$\text{FT} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}_c^t \in \mathcal{T}} \left(\acute{a}_c^t - \hat{a}_c^t \right)$$

where $\mathcal{T}$ denotes all tasks during the FCL process, $\acute{a}_c^t$ denotes the accuracy of $\mathcal{T}_c^t$ when it finished, and $\hat{a}_c^t$ denotes the accuracy of single-task training.

Backward Transfer (BT) This metric evaluates the model's ability to transfer knowledge from new tasks back to previously learned tasks, computed as

$$\text{BT} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}_c^t \in \mathcal{T}} \left(a_{c,\max(\mathcal{R})}^t - \acute{a}_c^t \right)$$

where $a_{c,\max(\mathcal{R})}^t$ denotes the final accuracy of $\mathcal{T}_c^t$.

Combined Transfer (CT) This metric is a combination of FT and BT, evaluating the amount of information that a task $\mathcal{T}_c^t$ acquires from other tasks. The other tasks can have any sequence relationship with task $\mathcal{T}_c^t$ in terms of temporal dimension. It is computed as

$$\mathrm{CT} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}_c^t \in \mathcal{T}} \left(a^t_{c,\max(\mathcal{R})} - \hat{a}^t_c \right)$$

Final Average Accuracy (FAA) FAA is a standard metric used in FCIL to measure knowledge retention and accumulation. Let a^t denote the test accuracy on the t-th task after the final incremental step. FAA is defined as:

$$\mathrm{FAA} = \frac{1}{T} \sum_{t=1}^{T} a^t$$

where T is the total number of tasks. A higher FAA indicates better overall performance across all tasks and stronger continual learning ability.

D Results under Other Federated Incremental Learning Experimental Settings

In addition to the FCL setting proposed by Powder (ICML 2024)[19], which considers task overlaps over time, we evaluate our approach under the federated class incremental learning (FCIL) setting, as investigated by PILoRA (ECCV 2024)[80] and LoRM (ICLR 2025) [81].

FCIL Setting: The FCIL setting divides the learning process into 10 incremental tasks, where class distributions are disjoint across tasks. For each task, training data is distributed among 10 clients following a Dirichlet distribution with parameter $\beta \in \{0.5, 0.1, 0.05\}$ to simulate non-IID scenarios. A smaller β value represents a stronger data imbalance among clients.

Training Details: A ViT-B/16 backbone pre-trained on ImageNet-21K is adopted. Each communication round consists of 5 training epochs, with a total of 5 communication rounds. Data augmentation for the training set includes random horizontal flipping and normalization. For the test set, preprocessing involves resizing with bicubic interpolation to 256×256, followed by center cropping to 224×224 and normalization.

Comparison Results: We compare our C^{2}Prompt with state-of-the-art FCIL methods [86; 87; 62; 11; 88; 89; 90; 91; 68; 80; 81] and the FCL method Powder [19] on the ImageNet-R benchmark. The Final Average Accuracy (FAA) [81] results are presented in Table 3. The results demonstrate that our C^{2}Prompt surpasses the state-of-the-art FCIL method LoRM, achieving improvements of **2.75%/5.91%/2.29%** at $\beta = 0.5/0.1/0.05$, respectively. Furthermore, compared to the state-of-the-art prompt-based FCL method Powder, our approach achieves **0.61%/2.60%/6.48%** improvements at $\beta = 0.5/0.1/0.05$, respectively. These increasing advantages under lower β values are attributed to the effective inter-client intra-class distribution knowledge compensation mechanism, which significantly enhances model acquisition capacity and mitigates inter-client knowledge conflicts. These findings, alongside the experiments reported in the main paper, validate the adaptability of our approach to diverse practical federated continual learning scenarios.

E Experimental comparison on Cifar100

Avg Comparison: Only our C^{2}Prompt outperforms the state-of-the-art Powder, achieving improvements of **1.54%** on Cifar100. This finding highlights the advantage of our approach in long-term knowledge consolidation. This can be attributed to the substantial enhancement of new knowledge acquisition capability achieved through our local class distribution compensation and class-aware discriminativity prompt aggregation strategies. Additionally, the precise knowledge communication mechanism prevents the fusion of irrelevant prompts, which would otherwise produce invalid prompts that are not only semantically divergent from new prompts but also clash with historical prompts.

AIA Comparison: Our C^{2}Prompt achieves improvements of **0.86%** on Cifar100, confirming that our approach consistently outperforms existing methods across various training stages. These improvements are due to the local class distribution compensation and class-aware discriminativity

Table 3: Performance comparison on ImageNet-R with different β values.

Method	Publication	ImageNet-R (FAA)		
		$\beta = 0.5$	$\beta = 0.1$	$\beta = 0.05$
EWC [86]	*NAS 2017*	58.93	48.15	43.68
LwF [87]	*PAMI 2017*	54.03	41.02	46.07
FisherAVG [88]	*NeurIPS 2022*	58.68	50.82	47.33
RegMean [89]	*ICLR 2023*	61.18	57.00	55.80
CCVR [90]	*NeurIPS 2021*	70.00	62.60	60.38
L2P [62]	*CVPR 2022*	42.08	23.85	16.98
CODA-P [11]	*CVPR 2023*	61.18	36.73	25.82
FedProto [91]	*AAAI 2022*	58.52	47.30	52.93
TARGET [68]	*ICCV 2023*	54.65	45.83	41.32
PILoRA [80]	*ECCV 2024*	53.67	51.62	49.37
Powder [19]	*ICML 2024*	<u>74.62</u>	<u>67.14</u>	62.26
LoRM [81]	*ICLR 2025*	72.48	63.83	<u>66.45</u>
C^{2}Prompt	*This Paper*	**75.23**	**69.74**	**68.74**

Table 4: Performance comparison on CIFAR-100 with different β values.

Method	Publication	CIFAR-100 (FAA)		
		$\beta = 0.5$	$\beta = 0.1$	$\beta = 0.05$
EWC [86]	*NAS 2017*	78.46	72.42	64.51
LwF [87]	*PAMI 2017*	62.87	55.56	47.09
FisherAVG [88]	*NeurIPS 2022*	76.10	74.43	65.31
RegMean [89]	*ICLR 2023*	59.80	45.88	39.08
CCVR [90]	*NeurIPS 2021*	79.95	75.14	65.30
L2P [62]	*CVPR 2022*	83.88	61.54	55.00
CODA-P [11]	*CVPR 2023*	82.25	61.82	46.74
FedProto [91]	*AAAI 2022*	75.79	70.02	60.55
TARGET [68]	*ICCV 2023*	74.72	72.32	62.60
PILoRA [80]	*ECCV 2024*	76.48	75.81	74.80
Powder [19]	*ICML 2024*	<u>87.46</u>	<u>85.33</u>	82.03
LoRM [81]	*ICLR 2025*	86.95	81.76	<u>82.76</u>
C^{2}Prompt	*This Paper*	**89.93**	**87.67**	**83.25**

prompt aggregation designs, which strengthen robust local knowledge acquisition and enhance distributed knowledge collection.

FM Comparison: Our C^{2}Prompt shows a negative forgetting rate on the small-scale dataset Cifar100. This suggests that new tasks can facilitate the learning of historical tasks when training samples are limited. These results confirm the effective antiforgetting capability of our method.

FT Comparison: Our C^{2}Prompt shows advanced forward-transfer capacity, outperforming existing methods on Cifar100, respectively. This can be primarily attributed to two key components: the Global Class Distribution Estimation and Local Class Distribution Compensation mechanisms. Specifically, the former effectively leverages asynchronously arriving data from the same class to generate reliable global distribution estimates, while the latter utilizes these estimated global distributions to implement data-level information compensation, thereby significantly enhancing the learning efficiency of subsequent data.

BT Comparison: Our C^{2}Prompt consistently demonstrates positive backward transfer capability on the Cifar100 dataset. This arises from the fact that asynchronously arriving data allow subsequent tasks to enhance knowledge of previously seen classes. We observe that C²Prompt's backward-transfer results relatively outperform those of Fed-CODAP and Powder. This is because the Local Class Distribution Compensation and Class-aware Prompt Aggregation designs significantly boost

Table 5: Result comparison on the CIFAR-100 benchmark

Methods	Publication	Avg↑	AIA↑	FM↓	FT↑	BT↑	CT↑
FedWEIT	*ICML 2021*	95.17	95.61	0.48	3.76	-0.91	3.04
CFeD	*IJCAI 2022*	73.87	79.06	2.07	-11.01	-4.71	-14.78
GLFC	*CVPR 2022*	95.35	95.92	0.35	**5.51**	-0.54	**5.08**
Fedspace	*CVPR 2023*	94.17	94.87	1.03	0.37	-2.46	-1.60
Fed-L2P	*CVPR 2022*	95.65	95.68	0.08	0.89	0.08	0.95
Fed-Dual	*ECCV 2022*	95.35	95.08	0.27	0.70	-0.24	0.51
Fed-CODAP	*CVPR 2023*	82.05	55.71	13.77	-30.25	-18.60	-45.13
FedCPrompt	*ICML 2023*	94.22	94.04	0.08	0.32	0.00	0.32
Powder	*ICML 2024*	95.78	95.83	0.35	2.03	-0.36	1.74
PILoRA	*ECCV 2024*	76.21	82.31	0.32	0.01	-0.42	-0.40
Fed-MOS	*AAAI 2025*	85.11	87.23	0.20	-0.31	-0.11	-0.45
LoRM	*ICLR 2025*	77.42	80.11	0.74	0.07	0.20	0.22
Ours	*This Paper*	**97.32**	**96.78**	**-0.05**	2.57	**0.31**	2.82

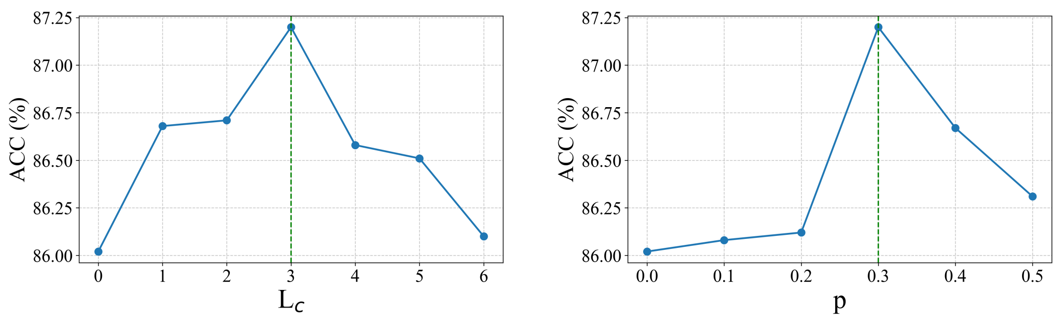

Figure 7: Ablation studies on the hyper-parameters under ImageNet-R dataset.

the distributed data learning capability at each stage, thereby leaving less improvement space for seen tasks.

CT Comparison: In terms of the comprehensive performance of forward and backward transfer, our C^{2}Prompt overall outperforms other existing methods that employ efficient fine-tuning. These results demonstrate that the class-aware client knowledge interaction designs proposed in this paper effectively enhance the overall learning capability of Federated Continual Learning (FCL) in the temporal dimension. Specifically: the Global Class Distribution Estimation module efficiently aggregates distributional information across spatial and temporal data sources; the Local Class Distribution Compensation module utilizes global distribution representations to overcome the non-IID (non-independent and identically distributed) phenomenon across clients; and the Local Discriminativity Learning and Class-aware Prompt Aggregation modules effectively integrate distributional knowledge into prompts.

F Analysis on the hyper-parameters

In Figure 7, we evaluate the performance of C^{2}Prompt under different values of the hyper-parameters L_c and p. The parameter L_c represents the length of the local class distribution compensation prompts. When the prompt length is less than or equal to 3, a larger value of parameter a enables the trained prompts to better fit the central distribution of the class, thereby improving the model's performance. Meanwhile, p serves as the usage probability of local class distribution compensation prompts, is used to determine the number of generated new central distribution samples. Based on experimental analysis, the optimal hyperparameter value for p is set to 0.3.

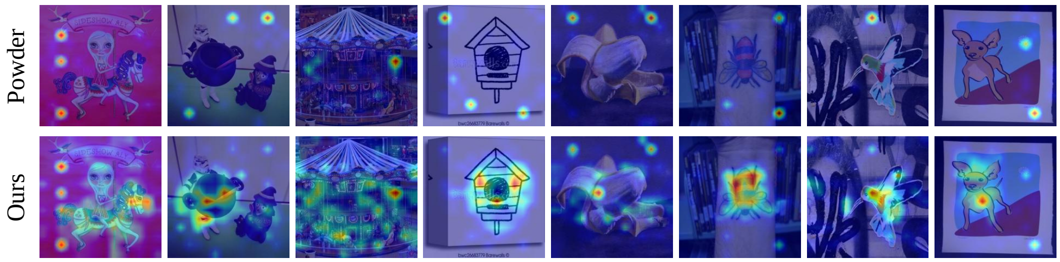

Figure 8: Prompt attention visualization on ImageNet-R.

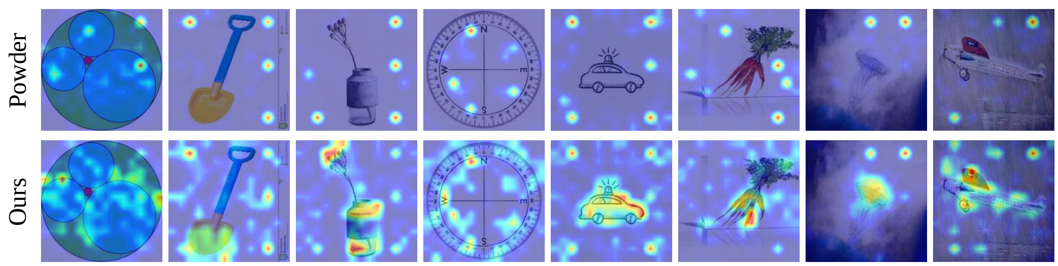

Figure 9: Prompt attention visualization on DomainNet.

G Prompt attention visualization comparison on different benchmarks

Figure 8 and Figure 9 present the visualization comparison of prompt attention maps between our C^{2}Prompt framework and the state-of-the-art Powder method across the challenging ImageNet-R and DomainNet benchmarks. The results demonstrate that prompts generated by Powder are predominantly influenced by class-irrelevant knowledge, leading to limited discriminative feature extraction capabilities. In contrast, our method effectively focuses prompt activations on discriminative regions while significantly reducing interference from class-agnostic knowledge. These improvements are primarily attributed to the proposed Class-aware Prompt Aggregation mechanism, which systematically alleviates the fusion of knowledge conflicts during prompt aggregation through explicit semantic alignment.

H Broader Impacts

Our method tackles a practical federated continual learning (FCL) problem and introduces a novel approach that effectively improves the local parameter learning in the client side and enhances knowledge aggregation capacity on the server side.

The Potential Positive Societal Impacts of this research include:

1. Enhanced Privacy Preservation in Decentralized Learning

Federated continual learning (FCL) inherently avoids centralized data collection. C^{2}Prompt further eliminates reliance on raw data or generative models for knowledge retention, reducing risks of sensitive data leakage. This is critical for applications like healthcare (e.g., personalized disease prediction across hospitals) or finance (e.g., fraud detection without sharing transaction details).

2. Improved Adaptability in Dynamic Environments

By addressing both temporal and spatial forgetting, C^{2}Prompt enables models to continuously adapt to evolving data streams. This ensures long-term reliability in scenarios where data distributions shift over time or vary across regions.

3. Democratization of AI in Resource-Constrained Settings

The lightweight prompt-based framework reduces computational and communication overhead compared to traditional methods. This democratizes access to AI for edge devices with limited resources (e.g., rural IoT sensors, low-power medical devices), fostering equitable technological progress.

4. Mitigation of Model Bias via Class-Aware Aggregation

The class-aware prompt aggregation (CPA) mechanism explicitly accounts for inter-client class relevance, potentially reducing biases arising from skewed local data distributions. For instance, in facial recognition systems deployed across diverse demographics, CPA could improve fairness by ensuring minority groups' features are adequately represented.

The Potential Negative Societal Impacts of this research include:

1. Energy Consumption

Additional distributional information communication and client-wise aggregation across distributed clients may increase energy consumption, particularly in large-scale deployments.