FedVaccine: Knowledge Recall after Spatial-Temporal Catastrophic Forgetting in Federated Continual Learning via Gradient-Based Vaccine

Hao Yu
Southwestern University of Finance and Economics (SWUFE)
yuhao2033@163.com

Xin Yang*
SWUFE
yangxin@swufe.edu.cn

Boyang Fan
SWUFE
fanby2014@126.com

Xuemei Cao
SWUFE
cxm_swpu@163.com

Linfei Ren
SWUFE
renlf@swufe.edu.cn

Hanlin Gu
Hong Kong University of Science and Technology
ghltsl123@gmail.com

Yang Liu
Hong Kong Polytechnic University
yang-veronica.liu@polyu.edu.hk

Qiang Yang
Hong Kong Polytechnic University
qyang@cse.ust.hk

Abstract

Federated Continual Learning (FCL) suffers from spatial-temporal catastrophic forgetting due to sequential task learning on individual clients and the aggregation of heterogeneous knowledge. Existing methods either overemphasize preserving old knowledge, limiting plasticity for new tasks, or rely on computationally expensive generative replay. In the human immune system, immune memory is preserved implicitly and can be reactivated through vaccination even after long periods of dormancy. Motivated by this, we propose ***FedVaccine***, a novel framework that explicitly permits forgetting to maintain high plasticity, while enabling efficient knowledge recovery via a compact vaccine set. For each task, the vaccine set is synthesized by compressing task gradients guided by the fine-tuned local model, capturing representative data characteristics and task-specific discriminative patterns. We further introduce two key components: (1) Priming & Boosting, which encodes task knowledge in a compact spectral form to support efficient reactivation after forgetting, and (2) Vaccination Aggregation, which aggregates vaccine sets across clients to train a globally generalized model without accessing raw data. Extensive experiments on multiple benchmarks demonstrate that FedVaccine achieves competitive performance, enables rapid knowledge recovery, and avoids the overhead of training generative models. The code is available at: https://github.com/SkyOfBeginning/FedVaccine.

1 Introduction

Federated Learning (FL) enables collaborative training across distributed clients without sharing raw data [53, 16, 17, 20, 21, 29, 30, 36, 31]. However, its assumption of relatively static data distributions limits its applicability in dynamic environments. This motivates Federated Continual

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Learning (FCL), which extends FL to sequentially evolving tasks [54, 26, 51]. In FCL, clients must accumulate knowledge over time while integrating information from heterogeneous peers [24]. This joint temporal–spatial setting substantially increases learning difficulty. Under strict privacy constraints, FCL faces two intertwined challenges: 1) temporal forgetting caused by continual learning, and 2) spatial forgetting induced by aggregating irrelevant or conflicting client knowledge [58, 60]. Furthermore, as clients repeatedly initialize new tasks from aggregated global models, these two types of forgetting reinforce each other over time, making spatial-temporal forgetting a fundamental challenge [11].



Figure 1: Overview of Motivation.

Table 1: Analogy for immune system and FedVaccine.

Immune System	FedVaccine	Function
Antibodies	Spectral memory $\mathcal{M}_i^t = \{V_i^t\}$.	Store compressed task knowledge.
Vaccine	Recall vaccine set $\mathcal{R}_i^t$ generated via gradient alignment.	Triggers for knowledge recall.
Immune priming	Task-specific LoRA training and SVD-based decomposition.	Encode task-specific information.
Immune boosting	Recall phase on θ_i^h triggered by $\mathcal{R}_i^t$ and regularized by $\mathcal{M}_i^t$.	Recover task performance.

Leveraging pre-trained foundation models (FMs) to tackle challenges in FCL has emerged as a prevailing trend recently. They provide a unified feature space across diverse clients, thereby facilitating the seamless transfer and long-term accumulation of spatial-temporal knowledge [59, 3, 52]. For instance, POWDER facilitates the transfer of knowledge encapsulated in prompts across diverse sequential tasks and clients [37]. DDDR utilizes a pre-trained conditional diffusion model to synthesize representative samples for each class, thereby mitigating catastrophic forgetting [28]. FedCBDR mitigates catastrophic forgetting in FCIL by preserving more balanced knowledge of previously learned classes during continual local updates [40].

Despite the success of methods in leveraging Parameter-Efficient Fine-Tuning (PEFT) or diffusion-based techniques, significant limitations remain. ***One primary issue is the sacrifice of plasticity*** *[7]. Most existing approaches focus heavily on preserving old knowledge while learning new tasks, which significantly hinders the model's ability to fit new information* [56, 61, 8, 23, 57]. Furthermore, although replay strategies impact plasticity less than regularization-based techniques, they incur prohibitive computational overhead, as they require training additional generative models to synthesize samples [62, 39]. **Therefore, developing methods that can alleviate spatial-temporal forgetting while maintaining plasticity and avoiding extra overhead remains a formidable challenge.**

Inspired by the human immune system, **priming** forms initial immunity upon first exposure, while **boosting** enables rapid reactivation of prior immunity with minimal stimulation, even after long periods [19, 46]. This remarkable ability to retain and recall past experiences motivates us to endow deep models with a similar mechanism. Specifically, we aim to enable models to construct an internal "vaccine" that supports efficient knowledge recall after long-term FCL. This raises a fundamental question: ***how can a model construct an effective internal vaccine that supports fast and reliable knowledge recall after spatial-temporal forgetting in FCL?***

Deep Leakage from Gradients (DLG) reconstructs private training samples by minimizing the gradient discrepancy between dummy and true data [67, 27, 55, 4]. We observe that such a gradient alignment strategy can be repurposed for constructive purposes rather than just privacy attacks. Specifically, by aligning the gradients of a small set of dummy data with those of the original training set, rich information from real samples can be effectively compressed into a compact synthetic dataset. It then functions as an "internal vaccine" to facilitate rapid knowledge recall (Figure 1).

To this end, we propose *FedVaccine*, a novel framework that re-purposes the gradient alignment mechanism as a generative tool to activate knowledge recall after forgetting. ***Vaccine Generation mechanism ensures that the intrinsic knowledge acquired by the model is implicitly encoded within the synthesized pseudo-samples.*** Moreover, the fine-tuned model serves as a discriminator, enabling

the recall samples to capture both training data information and the model's internal discriminative patterns. Consequently, they allow the model to swiftly recover its capabilities for old tasks.

Moreover, *FedVaccine* introduces a ***'Priming & Boosting'*** module to stabilize and accelerate knowledge retrieval. In the priming phase, we apply Singular Value Decomposition (SVD) to the PEFT parameters, retaining only the most significant singular values as the anti-bodies. These values are used as regularization terms when the model needs to re-activate previously learned knowledge in the boosting phase. Finally, ***the server fine-tunes the global model using synthesized vaccine samples from all clients.*** It yields a more generalized global model that serves as a robust starting point for subsequent client learning. Our main contributions are summarized as follows:

- Unlike conventional approaches that strive to strictly preserve old knowledge at the cost of plasticity, our method is the first to embrace a "forget-and-recall" paradigm. By generating recall vaccine set through gradient inversion, we demonstrate that the model can retrieve forgotten information, effectively overcoming forgetting without requiring additional generative models or plasticity loss.
- Vaccine Generation implicitly encodes intrinsic knowledge into synthesized pseudo-samples and utilizes SVD to provide a stable initialization for recalling. The server fine-tunes the global model using collected recall samples to yield a generalized global model for subsequent learning tasks.
- Extensive experiments on multiple datasets demonstrate the competitive performance of *FedVaccine*. Even after spatial-temporal catastrophic forgetting occurs, FedVaccine remains effective in helping client-side local models recover previously learned knowledge. To the best of our knowledge, it is the first to explore allowing forgetting in FCL to preserve the model's plasticity for new knowledge, while leveraging a recall set to quickly recover previously learned knowledge when needed.

2 Related Work

2.1 Regularization-based Federated Continual Learning

Regularization-based methods aim to mitigate forgetting by constraining parameter updates when learning new tasks. Early explorations such as FedWeiT [56], introduced methods for Task-IL scenarios that employ dedicated task IDs for different tasks to avoid interference.

More recent and representative works explore regularization in the context of parameter-efficient fine-tuning (PEFT). FedMGP [59], FedTA [60] and POWDER [37] leverage prompts and apply structured regularization to stabilize knowledge across clients and tasks. Other methods [42, 13] combine LoRA [15] with regularization to mitigate forgetting. pFedMxF [64] reduces update conflicts and communication overheads by assigning orthogonal frequency components in the LoRA matrix.

Despite their effectiveness, regularization-based approaches often suffer from a fundamental limitation: they enforce stability by restricting parameter updates, which can significantly hinder model plasticity when learning new tasks [7]. This trade-off becomes more severe in FCL due to spatial-temporal heterogeneity, where overly strong constraints may prevent the model from adapting to diverse client distributions. In contrast, our method explicitly allows forgetting to maximize plasticity, and instead relies on an on-demand recall mechanism to recover past knowledge.

2.2 Replay-based Federated Continual Learning

Replay-based methods constitute a dominant paradigm in FCL. Existing approaches can be broadly categorized into generative replay and exemplar replay. Generative replay methods leverage generative models to synthesize samples or latent representations of previously learned classes, which are rehearsed alongside current-task data to preserve prior knowledge [41, 28, 47, 38, 34, 6, 62]. This paradigm further enhances privacy by exchanging generators or synthetic features instead of raw data [43, 18, 10, 1]. Representative works, like LANDER [45], exploit pretrained label-text embeddings as semantic anchors for client training and generation.

In contrast, exemplar replay methods maintain a compact memory buffer of real samples from previous tasks and jointly rehearse them with new data, thereby preserving authentic data distributions and avoiding the complexity of generator training [8, 24, 32, 25]. FEAT improves exemplar replay under continual dynamic heterogeneity by preserving cross-task feature geometry and reducing majority-class bias [39]. They typically rely on importance-based sample selection or feature reconstruction strategies to improve memory efficiency and class balance.

However, both paradigms suffer from inherent limitations. Generative replay methods are prone to generator forgetting, limited sample fidelity and diversity, as well as significant computational and communication overhead. Exemplar replay methods, while more stable, often lack principled guarantees for sample selection and are constrained by memory budgets and privacy restrictions.

Unlike prior replay-based approaches, our method does not aim to approximate the original data distribution at the pixel level. Instead, it focuses on preserving the *optimization dynamics* induced by the data via gradient matching, enabling efficient knowledge recall with minimal samples. Furthermore, by combining gradient-based recall with a spectral memory mechanism derived from singular value decomposition, it provides a structured and lightweight alternative to both paradigms.

3 Problem Definition

We consider a standard centralized FL framework with a central server $\mathcal{S}$ and a clients, denoted as $\mathcal{A} = \{A_1, A_2, \ldots, A_a\}$ [63]. Each client A_i owns a local dataset $\mathcal{D}_i$, which consists of two disjoint sets of classes: *private classes* $\mathcal{C}_v^i$ and *public classes* $\mathcal{C}_p$. The private classes $\mathcal{C}_v^i$ contain data exclusively available on client A_i, ensuring data locality and privacy. The public classes $\mathcal{C}_p$ are shared across all clients; however, samples are strictly non-overlapping among clients to avoid duplication. Extreme spatial heterogeneity can be characterized by setting $|\mathcal{C}_p| = 0$, where clients share no common classes. To model *temporal heterogeneity*, each client A_i receives a task sequence $\mathcal{T}_i = \{T_i^1, T_i^2, \ldots, T_i^N\}$, where N denotes the total number of tasks assigned to A_i. Each task contains the same number of classes, while class sets are mutually exclusive across different tasks.

Given a global model θ_g^{t-1} aggregated at time $t-1$, each client A_i receives this model as the initialization for its local training on the current task T_i^t. To maximize model plasticity, the client directly optimizes the model on task T_i^t without imposing any constraints for historical knowledge:

$$\theta_i^t = \arg\min_{\theta} \mathcal{L}_{\text{task}}\big(f_{\theta_g^{t-1}}; T_i^t\big). \tag{1}$$

When the knowledge of a past task T_i^r ($r < t$) needs to be recalled, client A_i activates a task-specific *recall vaccine set* $\mathcal{R}_i^r$, which is generated via gradient inversion from task T_i^r. Instead of revisiting the original data, the model is re-optimized using the recall set to re-awaken the corresponding knowledge. The recall-aware objective is formulated as

$$\theta_i^{r\prime} = \arg\min_{\theta} \mathcal{L}_{\text{recall}}\big(f_{\theta_i^t}; \mathcal{R}_i^r\big). \tag{2}$$

The model θ_i^t is guided to regain its capability on task T_i^r, yielding a recalled model denoted as $\theta_i^{r\prime}$.

4 Methodology

To maximize the model plasticity for learning new knowledge, we propose **FedVaccine**. Instead of explicitly retaining or constraining previously learned knowledge, FedVaccine allows local models to naturally undergo *spatial-temporal forgetting* during FCL.

When knowledge from previous tasks is required, FedVaccine efficiently *recalls* past knowledge by re-learning from a small set of representative synthetic samples, referred to as the *recall vaccine set*. Through this lightweight replay, the model can rapidly re-activate previously acquired task-specific knowledge. We argue that this *forget-then-recall* paradigm is more realistic in practical federated scenarios, where long-term knowledge preservation is costly or infeasible, while on-demand knowledge recovery is sufficient and more aligned with real-world deployment constraints.

The framework of FedVaccine is shown in Figure 2 and the algorithm is summarized in Appendix B.

4.1 Vaccine Generation

Under the ***forget–then–recall*** paradigm, a central challenge lies in constructing an effective recall set. We propose to deliberately synthesize a compact recall vaccine set of pseudo samples that not only encapsulates the gradient information induced by historical data, but also reflects the model state at the time those data were learned. By forcing synthetic samples to fit the current model through gradient matching, the resulting recall set serves as an implicit "model vaccine," enabling efficient knowledge recall without additional generative models.

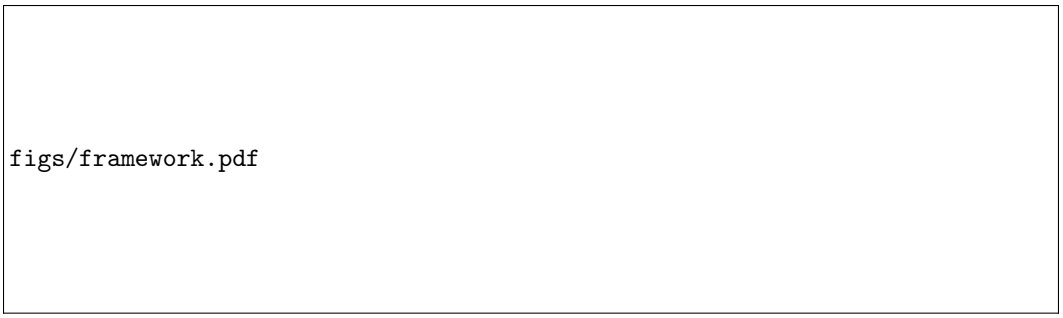


Figure 2: Overview of FedVaccine. (1) Vaccine Generation: a compact recall set is synthesized via gradient matching and discriminative guidance, where a LoRA-adapted model serves as discriminator and dummy samples are optimized to match gradients from real data. (2) Priming & Boosting: task-specific LoRA parameters are spectrally decomposed and stored to regularize recall training, enabling recovery of forgotten knowledge. (3) Vaccination Aggregation: the server aggregates client-generated synthetic data to update a global model, which is redistributed for subsequent tasks.

For the recall vaccine set $\mathcal{R}_i^t$, our objective is twofold. **First,** we require that training the local model on $\mathcal{R}_i^t$ induces gradients that closely match those generated by the original task dataset T_i^t. **Second,** the recall set is constructed to preserve sufficient information about the local model state θ_i^t, such that optimizing the model on $\mathcal{R}_i^t$ can approximately recover its task-specific behavior.

For the first objective, we initialize the recall samples from Gaussian noise. To eliminate the influence introduced by LoRA fine-tuning, we re-initialize a random classification head h_{r} and only employ the frozen feature extractor Θ together with h_{r} during recall set optimization.

Rather than performing one-to-one gradient matching, we adopt a **batch-to-set gradient alignment strategy**. Specifically, at each iteration, a mini-batch $B_{\mathrm{real}} \subset T_i^t$ is sampled from the full training data, while the recall set $\mathcal{R}_i^t$ remains a **compact and fixed-size synthetic set** with $|\mathcal{R}_i^t| \ll |T_i^t|$. The objective is to make this small recall set approximate the gradient induced by arbitrary real-data batches, effectively serving as a **compressed surrogate of the task dataset in the gradient space**.

$$\ell_{real} = \mathcal{L}_{ce}(h_r(\Theta(B_{real})), Y_{real}), \quad \ell_{recall} = \mathcal{L}_{ce}(h_r(\Theta(\mathcal{R}_i^t)), Y_{recall}). \tag{3}$$

The alignment loss is then the cosine distance between them:

$$\mathcal{L}_{\text{align}} = 1 - \cos\left(\nabla_{h_r}\ell_{real}, \nabla_{h_r}\ell_{recall}\right). \tag{4}$$

It enforces that a **small set of synthetic samples** produces gradients that are consistent with those of full-data mini-batches, achieving **gradient-level data compression and maintaining high efficiency.**

For the second objective, FedVaccine leverages the task-adapted local model θ_i^t (the ViT fine-tuned via LoRA on task T_i^t) as a discriminator during the iterative optimization of recall samples. By encouraging the recall samples to fit the decision behavior of θ_i^t, the current state of θ_i^t is implicitly encoded into the synthesized data, enabling the recall set to capture both task-specific knowledge and the temporal state of the model. The overall optimization objective for the recall samples is:

$$\min_{X_{\text{recall}}} \quad \mathcal{L}_{\text{vac}} = \mathcal{L}_{\text{align}} + \lambda_1 \mathcal{L}_{\text{disc}} = \mathcal{L}_{\text{align}} + \lambda_1 \mathcal{L}_{ce}(\theta_i^t(\mathcal{R}_i^t)), Y_{recall}). \tag{5}$$

4.2 Priming & Boosting

4.2.1 Priming

Viewing the model as an immune system, the recall vaccine set functions as a vaccine that **boosts** previously learned knowledge. However, effective recall is only possible if the model has been properly **primed**, i.e., if it has retained intrinsic memory of past tasks. In FedVaccine, such intrinsic memory is implicitly stored in task-specific low-rank subspaces, allowing the recall vaccine set to precisely and efficiently re-awaken the corresponding knowledge.

Therefore, when the training of θ_i^t is completed, besides serving as the discriminator to guide the generation of the recall set, the task-specific LoRA parameters are further decomposed to form the

model's intrinsic memory. Specifically, let W_i^t denote the LoRA-induced weight increment learned on task T_i^t, then we perform singular value decomposition (SVD) on W_i^t:

$$W_i^t = U_i^t \Sigma_i^t (V_i^t)^\top, \tag{6}$$

where $U_i^t \in \mathbb{R}^{d_{\text{out}} \times r}$ and $V_i^t \in \mathbb{R}^{d_{\text{in}} \times r}$ are the left and right singular vectors, respectively, and $\Sigma_i^t \in \mathbb{R}^{r \times r}$ is a diagonal matrix containing the singular values. To form a compact and effective task memory, FedVaccine retains only the right singular vectors:

$$\mathcal{M}_i^t = \{V_i^t\}. \tag{7}$$

The right singular vectors V_i^t characterize the dominant *input-sensitive* directions of the task-specific adaptation, encoding how task-relevant features are extracted from the input space. In contrast, the left singular vectors U_i^t mainly reflect output-side mappings and can be implicitly re-induced through replaying the vaccine set. In this way, the recall vaccine set boosts knowledge previously primed in the spectral memory $\mathcal{M}_i^t$, serving as the model's intrinsic immune memory for efficient recall.

4.2.2 Boosting

When the θ_i^h tries to recall the memory of T_i^t $(t < h)$ after spatial-temporal forgetting, it only needs to be "boosted" by retraining on the corresponding recall vaccine set $\mathcal{R}_i^t$. Meanwhile, the preserved $\mathcal{M}_i^t$ of the task-specific LoRA matrix is imposed as a spectral constraint, guiding the parameter updates to suppress task-irrelevant components and concentrate updates within the task-relevant spectral subspace, thus efficiently re-activate the forgotten knowledge. The recall loss is defined as:

$$\min_{\theta_i^h} \mathcal{L}_{\text{recall}}^{(t)} = \mathcal{L}_{\text{cls}}^{(t)} + \lambda_2 \mathcal{L}_{\text{spectral}}^{(t)}, \quad \mathcal{L}_{\text{spectral}}^{(t)} = \left\| W_i^h \left(I - V_i^t (V_i^t)^\top \right) \right\|_F^2. \tag{8}$$

Specifically, we penalize the component of the current weight matrix that lies in the orthogonal complement of the task subspace $\mathcal{S}_t = \text{span}(V_i^t)$. By projecting out components orthogonal to $\mathcal{S}_t = \text{span}(V_i^t)$, this regularization encourages updates to remain predominantly within the task-relevant spectral subspace. In combination with the recall vaccine set, which provides positive gradients for knowledge reactivation, the proposed boosting mechanism ensures both efficient and stable recovery of forgotten knowledge.

4.2.3 Vaccine-Inspired Memory Reactivation

Theorem 1 (Fast Recall in the Primed Subspace). Let $\mathcal{S}_t = \text{span}(V_i^t)$ denote the task-specific spectral subspace and let $\Pi_{\mathcal{S}_t}$ be the orthogonal projection onto $\mathcal{S}_t$. Assume that (i) the task-optimal W_i^t lies in $\mathcal{S}_t$, and (ii) the recall vaccine set $\mathcal{R}_i^t$ provides approximated stochastic gradients of the task loss within $\mathcal{S}_t$ with bounded variance σ^2. Suppose the task loss is μ-strongly convex and L-smooth in a neighborhood of W_i^t within $\mathcal{S}_t$. For step size $\eta \leq \mu/L^2$, the recall dynamics satisfy:

$$\mathbb{E}\left[\|\Pi_{\mathcal{S}_t}(W^{(k)} - W_i^t)\|_F^2\right] \leq (1 - \eta\mu)^k \|\Pi_{\mathcal{S}_t}(W^{(0)} - W_i^t)\|_F^2 + \frac{\eta\sigma^2}{\mu}. \tag{9}$$

where $W^{(k)}$ denotes the weight matrix after k steps of the recall optimization, $W^{(0)}$ is the initial weight at the start of recall (after spatial-temporal forgetting) and $\mu > 0$ is the strong convexity constant of the task loss within $\mathcal{S}_t$. *It explains why rapid recall is achieved: the primed spectral subspace preserves task-specific structure, while the vaccine gradients act as a targeted stimulus to quickly re-activate forgotten knowledge.*

Proof Sketch. The spectral regularizer $\mathcal{L}_{\text{spectral}}^{(t)}$ exponentially suppresses parameter components orthogonal to $\mathcal{S}_t$, effectively confining the optimization to the primed subspace. Within $\mathcal{S}_t$, the recall vaccine set provides controlled stochastic gradients, reducing the dynamics to standard stochastic gradient descent under strong convexity. The complete proof is provided in Appendix A.

4.3 Vaccination Aggregation

The server aims to form a globally generalized model that can help clients overcome the limitations caused by the scarcity of local data. FedVaccine compresses each client's local data into recall vaccine sets via gradient alignment. They produce gradients highly similar to those generated by the real data

while avoiding privacy leakage. Therefore, the server can safely collect the recall sets from all clients and use them to train a global model. The resulting global model is then distributed back to the clients to serve as the initial model for subsequent tasks, effectively enhancing local and global learning.

Formally, let $\mathcal{R}_i$ denote the recall set collected from client A_i. The global model θ_g is optimized by minimizing the cross-entropy loss over all clients' recall sets ($|\mathcal{A}|$ is the number of clients):

$$\mathcal{L}_{\text{global}}(\theta_g) = \frac{1}{|\mathcal{A}|} \sum_{i \in \mathcal{A}} \mathbb{E}_{(x,y)\sim\mathcal{R}_i} \left[\mathcal{L}_{\text{CE}}(\theta_g(x), y) \right]. \tag{10}$$

5 Experiments

5.1 Experimental Setup

Datasets. To evaluate FedVaccine under extreme spatial-temporal data heterogeneity, we follow the commonly adopted FCL protocol in [60, 59], which employs private and public classes to model label-space heterogeneity and client personalization. This design is also widely adopted in federated learning to simulate scenarios where clients possess both shared and client-specific knowledge [35, 44, 65]. Specifically, we consider 5 clients. For **ImageNet-R** [14] and **CUB-200** [50], each client is assigned 40 private classes that are disjoint across clients, with no public classes. For **CIFAR-100** [22], each client is assigned 15 private classes and 25 public classes shared across clients. The data associated with public classes are distributed according to a Dirichlet distribution to introduce statistical heterogeneity among clients.

For the evaluation metrics, we assess FedVaccine using two standard metrics: Average Accuracy (A_N) and Average Forgetting Rate (F_N), as defined in Appendix C.1. Specifically, A_N denotes the average accuracy on all previously seen classes after completing N tasks, reflecting the overall performance. F_N quantifies the performance degradation over N tasks, providing insight into the trade-off between plasticity and stability during FCL.

For the experimental protocols, we evaluate FedVaccine under a task-aware class-incremental learning setting. Each client receives a client-specific sequence of 5 tasks, with 8 classes per task. The task sequences may differ across clients, and each new task introduces classes that have not been observed by that client in previous tasks. During recall, task identity is available, allowing FedVaccine to activate the corresponding task-specific vaccine sets and spectral memories. This setting enables us to investigate federated continual learning under both client-specific label spaces and temporal distribution shifts.

Baselines. We compare FedVaccine with following methods: FedAvg [33], FedAvg+L2P [49], FedAvg+DualP [48], GLFC [8], TARGET [62], FedCprompt [3], MFCL [2], FedMGP [59], PILORA [13], LoRM [42], DDDR [28], CAN [41] and FedTA [60]. Brief introduction of baseline methods is provided in Appendix C.3. Note that, for fairness, we replace all backbone networks with ViT-B/16 [9] pre-trained on ImageNet-21k. For methods that are originally rehearsal-free, **we equip them with a replay buffer to ensure fairness and consistency.**

Implementation Details. All experiments are conducted on an Ubuntu 22.04 operating system with 64GB RAM, utilizing PyTorch=2.4.0. Training was performed sequentially on a single NVIDIA RTX-4080 GPU. For robustness, all results are averaged over three random seeds (42, 1999, 2025). Input images were resized to $3 \times 224 \times 224$ to match the input requirement of ViT-B/16 backbone, which was self-supervised and pre-trained on ImageNet-21K. Optimization was performed using the Adam optimizer with a learning rate of 10^{-3}, weight decay is set to 10^{-4}. To ensure fairness, all implementations are kept consistent across methods. Other important hyper-parameters are set as: the rank of LoRA matrix is set to 4, and it is applied to transformer blocks with indices [2, 3]; and for recall set construction, we generate 5 synthetic samples per class. Refer to Appendix C.2 for details.

5.2 Main Results

Table 2 demonstrates that FedAvg-based methods are highly vulnerable to spatial–temporal data heterogeneity, resulting in degraded accuracy and severe forgetting across all datasets. While replay-based methods such as TARGET, MFCL, DDDR, and CAN alleviate forgetting to some extent, their performance heavily depends on the quality of generated samples and degrades notably on

Table 2: Performance (%) on three datasets. The best results in each setting are **underlined in bold**, while the second-best results are underlined.

Algorithm	Source	CIFAR-100		ImageNet-R		CUB-200	
		$A_N(\uparrow)$	$F_N(\downarrow)$	$A_N(\uparrow)$	$F_N(\downarrow)$	$A_N(\uparrow)$	$F_N(\downarrow)$
FedAvg	AISTIS 2017	62.82±6.78	11.42±5.33	53.64±3.82	18.66±4.32	58.05±3.38	9.90±5.28
FedAvg+L2P	CVPR 2022	41.38±4.66	13.22±4.27	37.23±5.03	19.63±6.57	45.63±5.97	12.59±6.70
FedAvg+DualP	ECCV 2022	48.23±5.47	19.92±4.51	45.60±3.79	22.72±6.34	51.37±5.54	16.17±6.96
GLFC	CVPR 2022	81.40±5.62	8.72±4.13	68.45±6.90	12.00±5.61	84.30±4.32	7.66±2.07
TARGET	ICCV 2023	74.03±3.23	13.60±3.15	65.09±2.74	15.33±4.05	78.13±4.75	12.15±3.40
Fed-Cprompt	ICML 2023	87.45±1.34	7.16±2.70	71.05±1.55	5.94±2.30	82.22±1.72	10.83±2.45
MFCL	NeurIPS 2024	79.62±3.24	12.45±3.57	69.43±3.14	15.56±2.89	79.37±4.67	8.85±2.24
FedMGP	KDD 2024	90.07±1.13	**2.08±0.95**	79.64±1.74	5.25±2.87	88.13±1.96	4.16±3.47
LoRM	ICLR 2025	88.27±0.77	2.53±0.31	73.68±0.55	3.93±0.82	81.50±1.33	5.71±0.51
PILoRA	ECCV 2024	84.75±0.72	4.50±1.35	76.97±0.73	7.28±1.46	80.73±0.88	5.37±0.41
DDDR	ECCV 2024	82.35±1.45	7.85±2.42	72.36±1.30	8.92±2.19	84.18±1.64	3.49±0.99
CAN	ICML 2025	84.43±1.27	5.56±0.88	73.53±2.53	7.68±2.54	87.53±1.90	4.70±1.25
FedTA	CVPR 2025	**92.27±0.73**	4.24±0.52	81.14±0.88	5.32±0.60	90.33±1.28	4.68±0.52
FedVaccine	—	91.88±0.63	3.04±0.25	**84.51±1.10**	**3.70±0.98**	**91.94±0.92**	**0.82±0.15**

more complex datasets such as ImageNet-R. This limitation becomes more pronounced for higher-resolution images ($3 \times 224 \times 224$), where generating high-quality samples is substantially more challenging, further constraining the effectiveness of replay-based approaches.

In contrast, PEFT-based FCL methods (FedTA, FedMGP, PILoRA, LoRM, etc.) achieve more stable and competitive results; however, their plasticity on new tasks is often limited due to constrained parameter updates. FedVaccine maintains robust performance across all datasets while keeping the forgetting rate within a satisfactory range. By allowing controlled forgetting and enabling rapid knowledge recall, it effectively balances knowledge retention and adaptability under severe heterogeneity.

5.3 Recall Performance

Figure 3 reports three types of accuracy: (i) the instantaneous accuracy achieved immediately after training on each current task, (ii) the accuracy on previous tasks evaluated after completing the entire FCL process, and (iii) the accuracy of the final model after recall mechanism. As shown in Figure 3, FedVaccine is able to quickly restore the performance on past tasks, significantly narrowing the gap to the original post-training accuracy, which verifies its capability of efficient knowledge recovery.

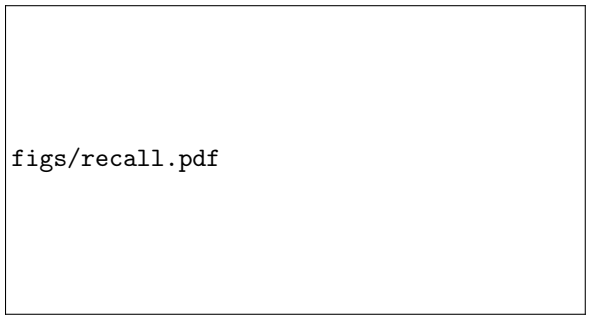


Figure 3: Forgetting and recall behavior.

Table 3: Memory retention analysis of LoRA parameters and classifier head across forgetting and recall.

Metric	**Origin**	**Forgetting**	**Recall**
Cosine ↑	1.00	0.28	0.71
Distance ↓	0.00	3.80	1.52
Subspace Sim ↑	1.00	0.41	0.73
Subspace Recovery ↑	1.00	0.36	0.69
Head Cosine ↑	1.00	0.13	0.54

We analyze the forgetting and recall behaviors in the parameter space by measuring the similarity between LoRA updates and Head updates. Specifically, we report **the cosine similarity, Frobenius distance, subspace alignment, and subspace recovery ratio**. The results in Table 3 demonstrate that severe forgetting leads to substantial degradation across all metrics, while our recall mechanism effectively restores both the magnitude and the structural alignment of the LoRA updates.

5.4 Ablation Studies & Scalability Analysis

We study the contributions of three key components in FedVaccine. Specifically, $\mathcal{L}_{disc}$ is the discriminator loss in Vaccine Generation that enforces discriminative behavior in the recall set; $\mathcal{L}_{spect}$ is the spectral regularization in *Priming & Boosting* that aligns updates with task-relevant subspaces; and **GMT** denotes the global model training strategy that aggregates recall sets across clients. The setting ($\times$ $\times$ $\times$) uses only gradient-aligned recall sets without these components. Table 4 shows that removing any component degrades performance, while removing all leads to a significant drop, demonstrating their complementary roles in mitigating forgetting and enabling knowledge recovery.

Table 4: Ablation study of three key components.

$\mathcal{L}_{disc}$	$\mathcal{L}_{spect}$	GMT	CIFAR-100	ImageNet-R	CUB-200
✓	✓	✓	91.88	84.51	91.94
✓	✓	×	89.32	81.66	89.80
✓	×	✓	82.53	76.40	84.35
×	✓	✓	84.71	78.14	87.53
×	×	×	72.19	63.84	75.27

Table 5: Scalability on 50 clients.

Sample Ratio		0.1	0.2	0.5
ImageNet-R	$A_N\uparrow$	79.56	81.24	81.77
	$F_N\downarrow$	5.82	6.10	6.04
CUB-200	$A_N\uparrow$	90.45	91.35	92.81
	$F_N\downarrow$	3.56	4.73	5.45

We further conduct experiments on ImageNet-R and CUB-200 with 50 clients, where each client sequentially learns 10 tasks of 8 classes each. We simulate partial participation by sampling 10%, 20%, and 50% of clients per round. Due to the increased number of clients, it becomes impractical to enforce strictly disjoint class partitions across all clients. Therefore, we adopt a Dirichlet distribution with a concentration parameter of $\alpha = 0.1$ for data partitioning. The results in Table 5 demonstrate that our method maintains strong performance and stable recall ability.

5.5 Sensitivity Analysis

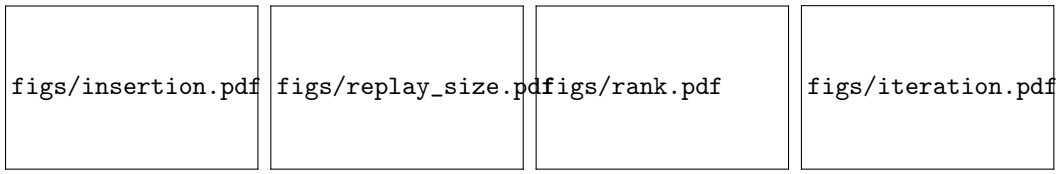


Figure 4: Sensitivity analysis on four important hyper-parameters on ImageNet-R.

To assess robustness, we conduct a sensitivity analysis on four key factors: the ViT block for LoRA insertion, recall set size per class, LoRA rank r, and the number of updates for Vaccine Generation. Results in Figure 4 report the final local accuracy after recall.

FedVaccine achieves the most compact and stable performance when LoRA is inserted into blocks [2,3], while deeper insertions lead to slightly less compact distributions but still strong performance, indicating low sensitivity. Increasing the recall set size consistently improves performance, though even 5 samples per class are sufficient for acceptable results. For the LoRA rank, $r = 4$ already performs well on ImageNet-R, while larger ranks increase cost and slightly degrade performance. Finally, 1000 updates yield the worst results, whereas 1200 updates already achieve satisfactory performance. Additional visualizations are provided in the next subsection.

5.6 Visualization of Recall Set

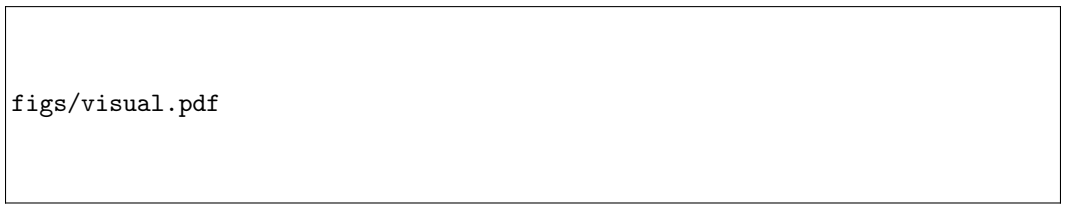


Figure 5: Illustration of recall vaccine samples from ImageNet-R using different pre-trained FMs.

Figure 5 shows recall samples generated under different optimization updates (1000 vs. 2000), as well as results using different pre-trained FMs (MoCo-v3 [5] and EVA-02 [12]).

FedVaccine maintains strong performance across different FMs. Although the generated samples are not human-interpretable, they capture label-relevant patterns guided by the discriminator. **Importantly, these samples are synthesized from compressed representations rather than raw data, effectively preserving task knowledge while avoiding exposure of private training data (Appendix H).**

5.7 Plasticity-Stability Analysis

CL and FCL methods are explicitly motivated by balancing stability and plasticity. FedVaccine challenges this balancing paradigm by decoupling stability from plasticity: it intentionally relaxes stability constraints to maximize plasticity during adaptation, and then restores forgotten knowledge through recall. To better demonstrate the plasticity–stability trade-off of FedVaccine, we include additional comparisons with representative FCL baselines on ImageNet-R.

We evaluate (1) the average learning time on new tasks ($t > 1$), (2) the total training time, and (3) the relative performance on new tasks, defined as the relative performance change between the accuracy on newly learned tasks and the final average accuracy ($ACC_{\text{relative}} = \frac{ACC_{\text{new}} - A_N}{A_N} \times 100$), (4) the average immediate new-task accuracy ($ACC_{new} = \frac{1}{T-1} \sum_{t=2}^{T} ACC_t$).

Table 6: Plasticity and efficiency comparison on ImageNet-R.

Type	Reg.	Reg.	Reg.	Reg. + Replay	Replay	Recall
Method	LoRM	FedMGP	FedTA	GLFC	CAN	FedVaccine
time on new task	110.32s	122.25s	100.62s	113.83s	124.36s	90.20s
total time	4.52h	5.13h	3.94h	4.53h	5.44h	4.27h
$ACC_{relative}$	-7.23%	+2.52%	-4.35%	-6.50%	-2.27%	+6.83%
ACC_{new} of 15 epochs	42.75	54.82	58.57	41.08	48.66	64.36
ACC_{new} of 25 epochs	73.64	80.53	83.27	65.31	75.92	88.42

These results suggest that the limitation of existing methods lies not in neglecting plasticity, but in preserving old knowledge during adaptation. By decoupling forgetting and recalling into two stages, FedVaccine avoids this conflict and provides a more effective continual learning paradigm. Further analysis of the plasticity–stability trade-off is provided in Appendix E.

5.8 Overheads Analysis

For efficiency, we summarize the computational overhead and communication cost in Table 7. Note that *Recall Samples per Class* refers to the total number of parameters that need to be optimized during the construction of the recall set. Among all components of our framework, vaccine generation is the most time-consuming, yet it requires only $\sim$0.12 GPU hours per task (see Appendix D).

Table 7: Computational overhead & Communication cost.

	Component Name	Parameter Counts
Computational Overhead	LoRA Matrix	3072 (A) + 9216 (B)
	Classification Head	496,328
	Recall Samples per Class	$5 \times 3 \times 224 \times 224$
Communication Cost	Recall Samples per Class	$5 \times 3 \times 224 \times 224$
	Global LoRA Matrix	3072 (A) + 9216 (B)
Memory Cost	Peak Memory (per client)	5033.31 MB
	Spectral memory per task (V)	1,044 parameters

6 Conclusion

We propose *FedVaccine*, which aims to recall learned knowledge by re-activating it through a recall vaccine set after spatial-temporal forgetting in FCL. FedVaccine allows the model to forget and later recall knowledge via synthesized samples. FedVaccine generates recall samples that compress the original gradients while capturing the model's optimal decision boundaries, effectively serving as a vaccine that primes the model for efficient knowledge recovery. Experiments show that it successfully maintains the model's flexibility to learn new tasks and enable knowledge recall after forgetting.

Acknowledgment

This work was supported by the National Natural Science Foundation of china (Nos. 62476228, 62506308) and Chengdu Science and Technology Program (No.2025-YF12-00030-RC).

References

[1] Ons Aouedi, Alexandre Boissel, and Kandaraj Piamrat. Gaia-fl: Generative ai-augmented federated learning for intrusion detection system. In *2025 International Wireless Communications and Mobile Computing (IWCMC)*, pages 192–197. IEEE, 2025.

[2] Sara Babakniya, Zalan Fabian, Chaoyang He, Mahdi Soltanolkotabi, and Salman Avestimehr. A data-free approach to mitigate catastrophic forgetting in federated class incremental learning for vision tasks. *Advances in Neural Information Processing Systems*, 36, 2024.

[3] Gaurav Bagwe, Xiaoyong Yuan, Miao Pan, and Lan Zhang. Fed-cprompt: Contrastive prompt for rehearsal-free federated continual learning. In *International Conference on Machine Learning (ICML)*. International Conference on Machine Learning (ICML), 2023.

[4] George Cazenavette, Antonio Torralba, and Vincent Sitzmann. Dataset distillation for pre-trained self-supervised vision models. *Advances in Neural Information Processing Systems*, 38:60361–60402, 2026.

[5] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9640–9649, 2021.

[6] Nikhil Churamani, Saksham Checker, Fethiye Irmak Dogan, Hao-Tien Lewis Chiang, and Hatice Gunes. Feature aggregation with latent generative replay for federated continual learning of socially appropriate robot behaviours. *arXiv preprint arXiv:2405.15773*, 2024.

[7] Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026):768–774, 2024.

[8] Jiahua Dong, Lixu Wang, Zhen Fang, Gan Sun, Shichao Xu, Xiao Wang, and Qi Zhu. Federated class-incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10164–10173, 2022.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[10] Gokberk Elmas, Salman UH Dar, Yilmaz Korkmaz, Emir Ceyani, Burak Susam, Muzaffer Ozbey, Salman Avestimehr, and Tolga Çukur. Federated learning of generative image priors for mri reconstruction. *IEEE Transactions on Medical Imaging*, 42(7):1996–2009, 2022.

[11] Tao Fan, Hanlin Gu, Xuemei Cao, Chee Seng Chan, Qian Chen, Yiqiang Chen, Yihui Feng, Yang Gu, Jiaxiang Geng, Bing Luo, et al. Ten challenging problems in federated foundation models. *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[12] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024.

[13] Haiyang Guo, Fei Zhu, Wenzhuo Liu, Xu-Yao Zhang, and Cheng-Lin Liu. Pilora: Prototype guided incremental lora for federated class-incremental learning. In *European Conference on Computer Vision*, pages 141–159. Springer, 2024.

[14] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021.

[15] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[16] Wenke Huang, Mang Ye, Zekun Shi, Guancheng Wan, He Li, Bo Du, and Qiang Yang. Federated learning for generalization, robustness, fairness: A survey and benchmark. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9387–9406, 2024.

[17] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.

[18] Honggu Kang, Seohyeon Cha, and Joonhyuk Kang. Gefl: Model-agnostic federated learning with generative models. *IEEE Transactions on Mobile Computing*, 2025.

[19] Kimia Kardani, Azam Bolhassani, and Sepideh Shahbazi. Prime-boost vaccine strategy against viral infections: Mechanisms and benefits. *Vaccine*, 34(4):413–423, 2016.

[20] Zhiqiang Kou, Junxiang Wu, Wenke Huang, Wenwen He, Ming-Kun Xie, Changwei Wang, Yuheng Jia, Di Jiang, Yang Liu, Xin Geng, and Qiang Yang. Fedharmony: Harmonizing heterogeneous label correlations in federated multi-label learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

[21] Zhiqiang Kou, Yucheng Xie, Hailin Wang, Jing Wang, Mingkun Xie, Shuo Chen, Yuheng Jia, Tongliang Liu, and Xin Geng. Rankmatch: A novel approach to semi-supervised label distribution learning leveraging rank correlation between labels. In *Proceedings of the 39th Conference on Neural Information Processing Systems (NeurIPS 2025)*, San Diego, California, USA, December 2025. Neural Information Processing Systems Foundation.

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

[23] Qiwei Li, Kunlun Xu, Yuxin Peng, and Jiahuan Zhou. Exemplar-free lifelong person re-identification via prompt-guided adaptive knowledge consolidation. *IJCV*, 132(11):4850–4865, 2024.

[24] Yichen Li, Qunwei Li, Haozhao Wang, Ruixuan Li, Wenliang Zhong, and Guannan Zhang. Towards efficient replay in federated incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12820–12829, 2024.

[25] Yichen Li, Haozhao Wang, Yining Qi, Wei Liu, and Ruixuan Li. Re-fed+: A better replay strategy for federated incremental learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[26] Yichen Li, Haozhao Wang, Wenchao Xu, Tianzhe Xiao, Hong Liu, Minzhu Tu, Yuying Wang, Xin Yang, Rui Zhang, Shui Yu, et al. Unleashing the power of continual learning on non-centralized devices: A survey. *IEEE Communications Surveys & Tutorials*, 2025.

[27] Zhuohang Li, Jiaxin Zhang, Luyang Liu, and Jian Liu. Auditing privacy defenses in federated learning via generative gradient leakage. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10132–10142, 2022.

[28] Jinglin Liang, Jin Zhong, Hanlin Gu, Zhongqi Lu, Xingxing Tang, Gang Dai, Shuangping Huang, Lixin Fan, and Qiang Yang. Diffusion-driven data replay: A novel approach to combat forgetting in federated class continual learning. In *European Conference on Computer Vision*, pages 303–319. Springer, 2024.

[29] Tianchi Liao, Xiaojun Deng, Lele Fu, Sheng Huang, Bowen Deng, Hong-Ning Dai, and Chuan Chen. Learning in the right subspace: Personalized differential private federated learning with noise filtering. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 3039–3050, 2026.

[30] Tianchi Liao, Ziyue Xu, Qing Hu, Hong-Ning Dai, Huaiwei Huang, Zibin Zheng, and Chuan Chen. Fedbrb: A solution to the small-to-large scenario in device-heterogeneity federated learning. *IEEE Transactions on Mobile Computing*, 25(3):3036–3050, 2026.

[31] Jianfeng Lu, Haibo Liu, Riheng Jia, Zhao Zhang, Xiong Wang, and Jiangtao Wang. Incentivizing proportional fairness for multi-task allocation in crowdsensing. *IEEE Transactions on Services Computing*, 17(3):990–1000, 2024.

[32] Yanyan Lu, Lei Yang, Hao-Rui Chen, Jiannong Cao, Wanyu Lin, and Saiqin Long. Federated class-incremental learning with dynamic feature extractor fusion. *IEEE Transactions on Mobile Computing*, 23(12):12969–12982, 2024.

[33] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[34] Yongsheng Mei, Liangqi Yuan, Dong-Jun Han, Kevin S Chan, Christopher G Brinton, and Tian Lan. Using diffusion models as generative replay in continual federated learning–what will happen? *arXiv preprint arXiv:2411.06618*, 2024.

[35] Yifan Niu and Weihong Deng. Federated learning for face recognition with gradient correction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 1999–2007, 2022.

[36] Bangqi Pan, Jianfeng Lu, Shuqin Cao, Jing Liu, Wenlong Tian, and Minglu Li. Bilateral pricing for dynamic association in federated edge learning. *IEEE Transactions on Mobile Computing*, 24(6):4684–4697, 2025.

[37] Hongming Piao, Yichen Wu, Dapeng Wu, and Ying Wei. Federated continual learning via prompt-based dual knowledge transfer. In *Forty-first International Conference on Machine Learning*, 2024.

[38] Daiqing Qi, Handong Zhao, and Sheng Li. Better generative replay for continual federated learning. *arXiv preprint arXiv:2302.13001*, 2023.

[39] Zhuang Qi, Ying-Peng Tang, Lei Meng, Guoqing Chao, Lei Wu, Han Yu, and Xiangxu Meng. From selection to scheduling: Federated geometry-aware correction makes exemplar replay work better under continual dynamic heterogeneity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17494–17504, June 2026.

[40] Zhuang Qi, Ying-Peng Tang, Lei Meng, Han Yu, Xiaoxiao Li, and Xiangxu Meng. Class-wise balancing data replay for federated class-incremental learning. *Advances in Neural Information Processing Systems*, 38:145309–145334, 2026.

[41] Xuankun Rong, Jianshu Zhang, Kun He, and Mang Ye. Can: Leveraging clients as navigators for generative replay in federated continual learning. In *Forty-second International Conference on Machine Learning*, 2025.

[42] Riccardo Salami, Pietro Buzzega, Matteo Mosconi, Jacopo Bonato, Luigi Sabetta, and Simone Calderara. Closed-form merging of parameter-efficient modules for federated continual learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

[43] Riccardo Salami, Pietro Buzzega, Matteo Mosconi, Mattia Verasani, and Simone Calderara. Federated class-incremental learning with hierarchical generative prototypes. *arXiv preprint arXiv:2406.02447*, 2024.

[44] Hideaki Takahashi, Jingjing Liu, and Yang Liu. Breaching fedmd: image recovery via paired-logits inversion attack. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12198–12207. IEEE, 2023.

[45] Minh-Tuan Tran, Trung Le, Xuan-May Le, Mehrtash Harandi, and Dinh Phung. Text-enhanced data-free approach for federated class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23870–23880, 2024.

[46] Jeffrey B Ulmer, Ulrich Valley, and Rino Rappuoli. Vaccine manufacturing: challenges and solutions. *Nature biotechnology*, 24(11):1377–1383, 2006.

[47] Naibo Wang, Yuchen Deng, Wenjie Feng, Jianwei Yin, and See-Kiong Ng. Data-free federated class incremental learning with diffusion-based generative memory. *arXiv preprint arXiv:2405.17457*, 2024.

[48] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648. Springer, 2022.

[49] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022.

[50] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010.

[51] Kunlun Xu, Yibo Feng, Jiangmeng Li, Yongsheng Qi, and Jiahuan Zhou. C^2 prompt: Class-aware client knowledge interaction for federated continual learning. In *NeurIPS*, 2025.

[52] Kunlun Xu, Zichen Liu, Xu Zou, Yuxin Peng, and Jiahuan Zhou. Long short-term knowledge decomposition and consolidation for lifelong person re-identification. *TPAMI*, 2025.

[53] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[54] Xin Yang, Hao Yu, Xin Gao, Hao Wang, Junbo Zhang, and Tianrui Li. Federated continual learning via knowledge fusion: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 36(8):3832–3850, 2024.

[55] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M Alvarez, Jan Kautz, and Pavlo Molchanov. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16337–16346, 2021.

[56] Jaehong Yoon, Wonyong Jeong, Giwoong Lee, Eunho Yang, and Sung Ju Hwang. Federated continual learning with weighted inter-client transfer. In *International conference on machine learning*, pages 12073–12086. PMLR, 2021.

[57] Hao Yu, Xin Yang, Boyang Fan, Yonghao Li, Hanlin Gu, Junbo Zhang, Tianrui Li, and Qiang Yang. Robust personalized federated continual learning via explainable multi-granularity prompt. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[58] Hao Yu, Xin Yang, Xin Gao, Yihui Feng, Hao Wang, Yan Kang, and Tianrui Li. Overcoming spatial-temporal catastrophic forgetting for federated class-incremental learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 5280–5288, 2024.

[59] Hao Yu, Xin Yang, Xin Gao, Yan Kang, Hao Wang, Junbo Zhang, and Tianrui Li. Personalized federated continual learning via multi-granularity prompt. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4023–4034, 2024.

[60] Hao Yu, Xin Yang, Le Zhang, Hanlin Gu, Tianrui Li, Lixin Fan, and Qiang Yang. Handling spatial-temporal data heterogeneity for federated continual learning via tail anchor. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4874–4883, 2025.

[61] Chengchao Zhang, Fanhua Shang, Hongying Liu, Liang Wan, and Wei Feng. Fedagc: Federated continual learning with asymmetric gradient correction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3841–3850, 2025.

[62] Jie Zhang, Chen Chen, Weiming Zhuang, and Lingjuan Lyu. Target: Federated class-continual learning via exemplar-free distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4782–4793, 2023.

[63] Xiangtao Zhang, Sheng Li, Ao Li, Yipeng Liu, Fan Zhang, Ce Zhu, and Le Zhang. Subspace constraint and contribution estimation for heterogeneous federated learning. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20632–20642. IEEE, 2025.

[64] Yifei Zhang, Hao Zhu, Alysa Ziying Tan, Dianzhi Yu, Longtao Huang, and Han Yu. pfedmxf: Personalized federated class-incremental learning with mixture of frequency aggregation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30640–30650, 2025.

[65] Zhengyi Zhong, Weidong Bao, Ji Wang, Shuai Zhang, Jingxuan Zhou, Lingjuan Lyu, and Wei Yang Bryan Lim. Unlearning through knowledge overwriting: Reversible federated unlearning via selective sparse adapter. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 30661–30670. IEEE, 2025.

[66] Zhengyi Zhong, Weidong Bao, Ji Wang, Xiaomin Zhu, and Xiongtao Zhang. Flee: A hierarchical federated learning framework for distributed deep neural network over cloud, edge, and end device. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 13(5):1–24, 2022.

[67] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019.

A Proof of Theorem 1 (Fast Recall in the Primed Subspace)

We provide the complete proof of Theorem 1 stated in the main text.

A.1 Notation and Setup

Let $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ be the weight matrix updated during recall. For task T_i^t, let W_i^t be the task-optimal LoRA matrix and $V_i^t \in \mathbb{R}^{d_{\text{in}} \times r}$ its right singular vectors, with $(V_i^t)^\top V_i^t = I_r$.

Define the task-specific spectral subspace

$$\mathcal{S}_t = \text{span}(V_i^t) = \{W : W = M(V_i^t)^\top,\ M \in \mathbb{R}^{d_{\text{out}} \times r}\}. \tag{11}$$

Its orthogonal complement is $\mathcal{S}_t^\perp = \{W : WV_i^t = 0\}$. Let $\Pi_{\mathcal{S}_t}$ and $\Pi_{\mathcal{S}_t^\perp}$ be the orthogonal projections onto $\mathcal{S}_t$ and $\mathcal{S}_t^\perp$, respectively (acting row-wise). For any W, we decompose $W = W_\parallel + W_\perp$ with $W_\parallel = \Pi_{\mathcal{S}_t}(W)$ and $W_\perp = \Pi_{\mathcal{S}_t^\perp}(W)$.

A.2 Assumptions

The theorem is built on the following assumptions, as stated in the main text:

(A1) **Subspace containment:** The task-optimal LoRA matrix satisfies $W_i^t \in \mathcal{S}_t$.

(A2) **Strong convexity and smoothness:** Within $\mathcal{S}_t$, the task loss $\mathcal{L}_{\text{task}}^{(t)}$ is μ-strongly convex and L-smooth in a neighborhood of W_i^t. That is, for any $W_\parallel = \Pi_{\mathcal{S}_t}(W)$ in this neighborhood,

$$\langle \nabla \mathcal{L}_{\text{task}}^{(t)}(W_\parallel),\, W_\parallel - W_i^t \rangle \geq \mu \|W_\parallel - W_i^t\|_F^2, \tag{12}$$

$$\|\nabla \mathcal{L}_{\text{task}}^{(t)}(W_\parallel)\|_F \leq L \|W_\parallel - W_i^t\|_F. \tag{13}$$

(A3) **Controlled vaccine gradient:** Let $\tilde{g}^{(k)} = \nabla_W \mathcal{L}_{\text{cls}}^{(t)}(W^{(k)})$ be the gradient computed from the recall vaccine set $\mathcal{R}_i^t$ at iteration k. Its projection onto $\mathcal{S}_t$ is an estimator of the true task gradient with bounded variance:

$$\mathbb{E}_k \left[\|\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)}) - \nabla \mathcal{L}_{\text{task}}^{(t)}(W_\parallel^{(k)})\|_F^2 \right] \leq \sigma^2, \tag{14}$$

A.3 Recall Objective and Update Rule

The recall objective is

$$\mathcal{L}_{\text{recall}}^{(t)}(W) = \mathcal{L}_{\text{cls}}^{(t)}(W) + \lambda_2\, \mathcal{L}_{\text{spectral}}^{(t)}(W), \tag{15}$$

with the spectral regularizer now defined as

$$\mathcal{L}_{\text{spectral}}^{(t)}(W) = \left\| W\big(I - V_i^t (V_i^t)^\top\big) \right\|_F^2. \tag{16}$$

Let $P_\perp = I - V_i^t (V_i^t)^\top$. Since V_i^t has orthonormal columns, $P_\perp$ is the orthogonal projector onto $\mathcal{S}_t^\perp$, satisfying $P_\perp^2 = P_\perp$ and $P_\perp^\top = P_\perp$.

The gradient of the spectral regularizer is

$$\nabla_W \mathcal{L}_{\text{spectral}}^{(t)}(W) = 2WP_\perp. \tag{17}$$

The gradient descent update at iteration k is

$$W^{(k+1)} = W^{(k)} - \eta\big(\tilde{g}^{(k)} + \lambda_2 \nabla_W \mathcal{L}_{\text{spectral}}^{(t)}(W^{(k)})\big), \tag{18}$$

where $\tilde{g}^{(k)} = \nabla_W \mathcal{L}_{\text{cls}}^{(t)}(W^{(k)})$ is the gradient from the recall vaccine set.

Decomposing:

$$W_\parallel^{(k+1)} = W_\parallel^{(k)} - \eta\, \Pi_{\mathcal{S}_t}(\tilde{g}^{(k)}), \tag{19}$$

$$W_\perp^{(k+1)} = W_\perp^{(k)} - \eta\, \Pi_{\mathcal{S}_t^\perp}(\tilde{g}^{(k)}) - 2\eta\lambda_2\, W_\perp^{(k)}. \tag{20}$$

A.4 Step 1: Exponential Suppression of $W_\perp$

From the definition, $W_\parallel \in \mathcal{S}_t$ implies $W_\parallel P_\perp = 0$. Therefore, $\nabla_W \mathcal{L}^{(t)}_{\text{spectral}}(W) = 2W_\perp P_\perp = 2W_\perp$, which is entirely in $\mathcal{S}_t^\perp$.

The orthogonal update simplifies to

$$W_\perp^{(k+1)} = W_\perp^{(k)} - \eta\, \Pi_{\mathcal{S}_t^\perp}(\tilde{g}^{(k)}) - 2\eta\lambda_2 W_\perp^{(k)}. \tag{21}$$

Taking norms and applying the triangle inequality:

$$\|W_\perp^{(k+1)}\|_F \le (1 - 2\eta\lambda_2)\|W_\perp^{(k)}\|_F + \eta\|\Pi_{\mathcal{S}_t^\perp}(\tilde{g}^{(k)})\|_F. \tag{22}$$

Choosing $\eta\lambda_2 < 1/2$ ensures $\rho = 1 - 2\eta\lambda_2 \in (0, 1)$. Under the uniform boundedness of $\|\tilde{g}^{(k)}\|_F$,

$$\|W_\perp^{(k)}\|_F \le \rho^k \|W_\perp^{(0)}\|_F + \frac{\eta G_\perp}{1-\rho}. \tag{23}$$

Thus, the orthogonal component is exponentially suppressed and bounded asymptotically by $G_\perp/(2\lambda_2)$. The residual bound depends on the magnitude of the orthogonal vaccine gradients and the spectral regularization strength, rather than the learning rate alone.

A.5 Step 2: Linear Convergence within the Primed Subspace

We now analyze the evolution of $W_\parallel^{(k)}$. Define the in-subspace error $e^{(k)} = W_\parallel^{(k)} - W_i^t$. Since $W_i^t \in \mathcal{S}_t$ by Assumption (A1), we have $\Pi_{\mathcal{S}_t}(W_i^t) = W_i^t$.

From (19),

$$e^{(k+1)} = e^{(k)} - \eta\, \Pi_{\mathcal{S}_t}(\tilde{g}^{(k)}). \tag{24}$$

Let $g^{(k)} = \nabla \mathcal{L}^{(t)}_{\text{task}}(W_\parallel^{(k)})$ denote the true task gradient at $W_\parallel^{(k)}$. By Assumption (A3), $\mathbb{E}_k[\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})] = g^{(k)}$. Taking the conditional expectation of the squared Frobenius norm of (24):

$$\begin{aligned}
\mathbb{E}_k[\|e^{(k+1)}\|_F^2] &= \mathbb{E}_k\big[\|e^{(k)} - \eta\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})\|_F^2\big] \\
&= \|e^{(k)}\|_F^2 - 2\eta\langle e^{(k)}, \mathbb{E}_k[\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})]\rangle + \eta^2\mathbb{E}_k[\|\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})\|_F^2] \\
&= \|e^{(k)}\|_F^2 - 2\eta\langle e^{(k)}, g^{(k)}\rangle + \eta^2\|g^{(k)}\|_F^2 + \eta^2\mathbb{E}_k[\|\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)}) - g^{(k)}\|_F^2] \\
&\le \|e^{(k)}\|_F^2 - 2\eta\mu\|e^{(k)}\|_F^2 + \eta^2 L^2\|e^{(k)}\|_F^2 + \eta^2\sigma^2,
\end{aligned} \tag{25}$$

where the inequality uses the strong convexity bound $\langle e^{(k)}, g^{(k)}\rangle \ge \mu\|e^{(k)}\|_F^2$ from (12), the smoothness bound $\|g^{(k)}\|_F \le L\|e^{(k)}\|_F$ from (13), and the variance bound σ^2 from (14).

For step size $\eta \le \mu/L^2$, we obtain $-2\eta\mu + \eta^2 L^2 \le -\eta\mu$. Indeed, the inequality $-2\eta\mu + \eta^2 L^2 \le -\eta\mu$ is equivalent to $-\eta\mu + \eta^2 L^2 \le 0$, i.e., $\eta L^2 \le \mu$, which holds precisely when $\eta \le \mu/L^2$. Thus, under this step-size condition,

$$\mathbb{E}_k[\|e^{(k+1)}\|_F^2] \le (1 - \eta\mu)\|e^{(k)}\|_F^2 + \eta^2\sigma^2. \tag{26}$$

Taking total expectation and unrolling the recursion over k iterations:

$$\begin{aligned}
\mathbb{E}[\|e^{(k)}\|_F^2] &\le (1-\eta\mu)^k\|e^{(0)}\|_F^2 + \eta^2\sigma^2\sum_{j=0}^{k-1}(1-\eta\mu)^j \\
&\le (1-\eta\mu)^k\|e^{(0)}\|_F^2 + \frac{\eta^2\sigma^2}{1-(1-\eta\mu)} \\
&= (1-\eta\mu)^k\|e^{(0)}\|_F^2 + \frac{\eta\sigma^2}{\mu}.
\end{aligned} \tag{27}$$

Recalling $e^{(k)} = \Pi_{\mathcal{S}_t}(W^{(k)} - W_i^t)$ and $e^{(0)} = \Pi_{\mathcal{S}_t}(W^{(0)} - W_i^t)$ yields exactly the bound stated in Theorem 1.

A.6 Step 3: Combined Effect and Final Remarks

Combining Step 1 and Step 2, the total error decomposes as

$$\|W^{(k)} - W_i^t\|_F^2 = \|\Pi_{\mathcal{S}_t}(W^{(k)} - W_i^t)\|_F^2 + \|W_\perp^{(k)}\|_F^2, \tag{28}$$

where the cross-term vanishes due to orthogonality of $\mathcal{S}_t$ and $\mathcal{S}_t^\perp$. Step 1 shows that the orthogonal component is exponentially suppressed and admits an asymptotic upper bound of $G_\perp/(2\lambda_2)$ under bounded orthogonal vaccine gradients. Step 2 establishes the convergence bound for the in-subspace component. The overall convergence behavior is determined by the combination of these two components.

A.7 Remark on the Error Floor

The asymptotic error floor $\eta\sigma^2/\mu$ is governed by three factors:

- η (step size): smaller η reduces the noise floor but slows convergence;
- σ^2 (gradient variance): the quality of the recall vaccine set directly controls this term through the gradient matching loss $\mathcal{L}_{\text{align}}$ used during vaccine generation;
- μ (strong convexity): well-conditioned task losses with larger μ yield better recovery accuracy.

In the noiseless limit $\sigma^2 \to 0$ (i.e., exact gradient recovery from the vaccine set), the bound guarantees exact convergence to W_i^t within $\mathcal{S}_t$.

A.8 Extension: Convergence under Biased Vaccine Gradients

We relax Assumption (A3) to allow a bounded bias in the vaccine gradient. Specifically, assume there exists $\zeta \geq 0$ such that

$$\left\|\mathbb{E}_k[\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})] - \nabla\mathcal{L}_{\text{task}}^{(t)}(W_\parallel^{(k)})\right\|_F \leq \zeta, \tag{29}$$

while the variance bound σ^2 remains unchanged.

Under this assumption, the error dynamics (24) give

$$\begin{aligned}\mathbb{E}_k[\|e^{(k+1)}\|_F^2] &\leq \|e^{(k)}\|_F^2 - 2\eta\langle e^{(k)}, \mathbb{E}_k[\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})]\rangle + \eta^2\mathbb{E}_k[\|\Pi_{\mathcal{S}_t}(\tilde{g}^{(k)})\|_F^2] \\ &\leq \|e^{(k)}\|_F^2 - 2\eta\langle e^{(k)}, g^{(k)}\rangle + 2\eta\|e^{(k)}\|_F\zeta + \eta^2\|g^{(k)}\|_F^2 + \eta^2\sigma^2. \end{aligned} \tag{30}$$

Using strong convexity $\langle e^{(k)}, g^{(k)}\rangle \geq \mu\|e^{(k)}\|_F^2$, smoothness $\|g^{(k)}\|_F \leq L\|e^{(k)}\|_F$, and the inequality $2\eta\zeta\|e^{(k)}\|_F \leq \eta\mu\|e^{(k)}\|_F^2/2 + 2\eta\zeta^2/\mu$, we obtain for $\eta \leq \mu/(2L^2)$,

$$\mathbb{E}_k[\|e^{(k+1)}\|_F^2] \leq (1 - \tfrac{\eta\mu}{2})\|e^{(k)}\|_F^2 + \eta^2\sigma^2 + \frac{2\eta\zeta^2}{\mu}. \tag{31}$$

Unrolling the recursion yields

$$\mathbb{E}[\|e^{(k)}\|_F^2] \leq (1 - \tfrac{\eta\mu}{2})^k\|e^{(0)}\|_F^2 + \frac{2\eta\sigma^2}{\mu} + \frac{4\zeta^2}{\mu^2}. \tag{32}$$

Compared with Theorem 1, an additional $\mathcal{O}(\zeta^2/\mu^2)$ term appears, which does not decay with k. This term originates from the systematic bias of the vaccine gradients and quantifies the price paid for imperfect gradient matching during vaccine generation. Importantly, the exponential convergence rate is preserved, and the extra error floor vanishes as the gradient matching quality improves ($\zeta \to 0$).

End of Proof.

B Pseudo Code of FedVaccine

Algorithm 1 Pseudo Code of FedVaccine Algorithm.

Input: a clients $\mathcal{A} = \{A_i\}_{i=1}^{a}$ with their own task sequence $\mathcal{T}_i = \{T_i^n\}_{n=1}^{N}$, a pre-trained frozen ViT Θ without classification head, local classification head H_i, max iteration counts MAX.
Output: Local model θ_i^N and recall sets $\mathcal{R}_i = \{\mathcal{R}_i^n\}_{n=1}^{N}$ of client A_i.

while task number n $\leq$ N **do**
  Assign global model $\theta_g^{n-1} = \{W_g^{n-1}, H_g\}$ to each clients.
  for each client $A_i, 1 \leq i \leq a$ **do**
    $\{\theta_i^n, W_i^n\} \leftarrow$ LoRA-Fine-Tuning($\Theta, W_i^{n-1}, H_i; T_i^t$);
    1. Vaccine Generation:
    Initialize recall set $\mathcal{R}_i^n = \{X_{recall}, Y_{recall}\}$ with Gaussian Noise.
    Add random classification head H_r to Θ.
    while iter $\leq MAX$ **do**
      for every batch $\{X_{real}, Y_{real}\}$ **do**
        $\ell_{real} = \mathcal{L}_{ce}(h_r(\Theta(X_{real})), Y_{real})$;
        $\ell_{recall} = \mathcal{L}_{ce}(h_r(\Theta(X_{recall})), Y_{recall})$;
        $\mathcal{L}_{align} = 1 - \cos(\frac{\partial \ell_{real}}{\partial h_r}, \frac{\partial \ell_{recall}}{\partial h_r})$;
        $\mathcal{L}_{disc} = \mathcal{L}_{ce}(\theta_i^n(X_{recall}), Y_{recall})$;
        Optimize X_{recall} with the loss: $\lambda_1 \mathcal{L}_{disc} + \mathcal{L}_{align}$;
        iter = iter + 1;
      end for
    end while
    Recall set $\mathcal{R}_i^n = \{X_{recall}, Y_{recall}\}$ is obtained;
    $U_i^t \Sigma_i^t (V_i^t)^\top \leftarrow \mathrm{SVD}(\mathrm{W_i^t})$;
    $\mathcal{M}_i^t \leftarrow \{V_i^t, \Sigma_i^t\}$; {*Priming.*}
  end for
  2. Global Model Training:
  $\mathcal{R}_g^n \leftarrow \bigcup_{i=1}^{a} \bigcup_{j=1}^{n} \mathcal{R}_i^j$;
  Update global model θ_g^n using $\mathcal{R}_g^n$;
end while
3. Boosting:
When the final local model $\theta_i^N = \{H_i, W_i^N\}$ tries to recall the memory of T_i^n:
$\{U_i^n, \Sigma_i^n\} \leftarrow \mathcal{M}_i^n$;
for each $\{x, y\} \in \mathcal{R}_i^{\setminus}$ **do**
  $\mathcal{L}_{\text{spectral}}^{(n)} = \left\| W_i^h \left(I - V_i^t \left(V_i^t \right)^\top \right) \right\|_F^2$;
  $\mathcal{L}_{cls} = \mathcal{L}_{ce}(\theta_i^N(X_{recall}), Y_{recall})$;
  Optimize θ_i^N with $\lambda_2 \mathcal{L}_{\text{spectral}}^{(n)}$ and $\mathcal{L}_{ce}$;
end for
$\theta_i^{n\prime}$ is obtained.

C Experiment Details

C.1 Evaluation Metrics

We employ two metrics to assess the performance of FedVaccine in the FCL.

Average Accuracy (A_N): This metric evaluates the final global model's classification performance on all the test sets after completing the final task, reflecting its ability to learn new tasks while retaining previously acquired knowledge. It is computed as:

$$A_N = \frac{1}{|\mathcal{A}|} \sum_{i=1}^{|\mathcal{A}|} \left(\frac{1}{N} \sum_{t=1}^{N} A_{i,t}^N \right), \tag{33}$$

where $A_{i,t}^N$ denotes the classification accuracy of the final global model θ_g^N on the test set of task T_i^t of client i.

Average Forgetting Rate (F_N): This metric measures the extent to which previously learned knowledge is forgotten after completing the final task. For each client, we compute the average performance drop on all past tasks by comparing the best historical accuracy with the accuracy obtained after learning N tasks. The metric is defined as:

$$F_N = \frac{1}{|\mathcal{A}|} \sum_{i=1}^{|\mathcal{A}|} \left(\frac{1}{N-1} \sum_{t=1}^{N-1} \left(a_{i,t}^t - A_{i,t}^N \right) \right), \tag{34}$$

where $a_{i,t}^t$ denotes the accuracy of client A_i's local model θ_i^t after learning T_i^t when evaluated on the test set of task T_i^t.

C.2 Implementation Details

To ensure reproducibility and clarify the experimental setup, we present the default key hyper-parameter configurations of FedVaccine adopted in the main experiments.

Table 8: Hyper-parameter configurations.

Hyper-parameter	Value
Total number of clients	5
Local training epoch	25
Global epoch per task	3
Rank of local LoRA	4
Block index of LoRA Insertion	[2,3]
Learning rate	0.001
Weight decay	0.0001
Max iteration for Vaccine Generation	2000
Size of recall set	5 per class
λ_1 in Vaccine Generation	0.2
λ_2 in Priming & Boosting	0.2

For completeness, we summarize the detailed architecture of the Vision Transformer (ViT-B/16) backbone used in our experiments, as illustrated in Table 9. ViT-B/16 divides the input image into non-overlapping patches of size 16×16 and processes them using a stack of Transformer encoder blocks. The model is pre-trained on ImageNet-21k and serves as the frozen foundation model throughout the FCL process.

C.3 Brief Introduction of Baselines

FedAvg (AISTATS 2017). FedAvg is the seminal federated learning algorithm that performs local training on each client followed by parameter averaging on the server. While effective in static settings, FedAvg does not account for task sequences or temporal distribution shifts, making it vulnerable to catastrophic forgetting in federated continual learning scenarios.

FedAvg+L2P (CVPR 2022). FedAvg+L2P extends the standard FedAvg framework by incorporating the Learning-to-Prompt (L2P) mechanism. By learning task-specific prompts while keeping the backbone largely frozen, it mitigates forgetting in continual learning. However, its reliance on prompt preservation may restrict model plasticity under highly dynamic task distributions.

FedAvg+DualP (ECCV 2022). Combines the DualPrompt continual learning method with federated averaging. DualPrompt learns complementary prompts for task-invariant and task-specific knowledge, which can mitigate forgetting without rehearsal.

GLFC (CVPR 2022). An FCL method that uses a class-aware gradient compensation loss and class-semantic relation distillation to balance retention of old tasks and learning of new ones. It also introduces privacy-preserving gradient communication to mitigate catastrophic forgetting across tasks and clients.

Table 9: Architecture of Vision Transformer ViT-B/16.

Component	Configuration
Input Image Size	$224 \times 224 \times 3$
Patch Size	16×16
Number of Patches	$14 \times 14 = 196$
Patch Embedding Dimension	768
Positional Embedding	Learnable, 197 tokens (incl. [CLS])
Transformer Encoder	
Number of Layers	12
Hidden Dimension	768
MLP Dimension	3072
Number of Attention Heads	12
Attention Head Dimension	64
Activation Function	GELU
Layer Normalization	Pre-LN
Classification Head	
[CLS] Token	Yes
Output Dimension	Task-dependent
Total Parameters	$\sim$86M

TARGET (ICCV 2023). TARGET addresses catastrophic forgetting in federated class-continual learning without storing replay samples. It preserves historical class knowledge by distilling predictions from previous global models during local training, thereby enabling clients to retain old-class information under strict privacy constraints.

Fed-Crompt (ICML 2023). Fed-CPrompt (Contrastive Prompt for Rehearsal-Free Federated Continual Learning) learns task prompts collaboratively across clients without storing past data. It introduces a contrastive objective to align prompt representations and improve generalization, effectively handling heterogeneous task sequences without explicit replay.

MFCL (NeurIPS 2024). A generative replay FCL method where a server-side generative model synthesizes representative samples from past distributions to be replayed during local training, reducing catastrophic forgetting without client data sharing.

FedMGP (KDD 2024). A personalized federated continual learning approach that learns both coarse-grained global prompts and fine-grained local prompts, using selective prompt fusion to retain spatial and temporal knowledge while personalizing models for each client.

LORM (ICLR 2025). LORM (Closed-form Merging of Parameter-Efficient Modules for Federated Continual Learning) focuses on efficiently merging parameter-efficient modules, such as LoRA adapters, across clients and tasks. It derives a closed-form solution for aggregating task-specific low-rank updates, enabling stable knowledge consolidation without accessing historical data or retraining previous modules.

PILoRA (ECCV 2024). PILoRA (Prototype-Guided Incremental LoRA) is a parameter-efficient federated class-incremental learning approach that integrates low-rank adaptation (LoRA) with class prototype guidance. By updating only task-specific LoRA modules and utilizing aggregated class prototypes to guide incremental learning, PILoRA achieves an effective balance between stability and plasticity while significantly reducing communication and optimization overhead.

DDDR (ECCV 2024). Diffusion-Driven Data Replay (DDDR) is a generative replay-based method designed for federated class-incremental learning. It leverages a diffusion model trained on the server to synthesize representative samples of previously learned classes. These synthetic samples are replayed during local training to mitigate catastrophic forgetting, without requiring access to real historical data, thereby preserving privacy while maintaining knowledge retention.

CAN (ICML 2025). CAN (Clients As Navigators) proposes a novel generative replay framework for federated continual learning, where clients act as navigators to guide the training of a generative model. By exploiting the evolution trajectories of client models across tasks, CAN synthesizes replay samples that better approximate historical data distributions, thereby alleviating forgetting under severe spatial and temporal heterogeneity.

FedTA (CVPR2025). A framework addressing spatial-temporal data heterogeneity by incorporating trainable tail anchors in representation space, combining input enhancement, selective knowledge fusion, and global prototype selection to mitigate both spatial and temporal catastrophic forgetting in FCL.

These baselines cover representative categories including rehearsal-based, regularization-based, prompt-based, and parameter-efficient FCL methods.

D Efficiency Analysis

In FedVaccine, each vaccine recall sample is implemented as a dummy input with a size of 3 ×224×224. In our experimental setting, we store 5 vaccine recall samples per class, and each task contains 8 classes, resulting in a vaccine recall set of size 8×5×3×224×224 for each task, i.e., 40 recall samples in total. These recall samples can be naturally stacked into a single batch and processed in parallel on the GPU during the vaccine construction stage. In our experimental environment, using a single NVIDIA RTX 4080 GPU, with approximately 1,600 training samples and 2,000 optimization updates, the total GPU time required to generate all 40 vaccine recall samples is $\sim$0.12 GPU hours.

Table 10: GPU resource consumption for vaccine recall set construction in FedVaccine.

Item	**Value**
Dummy data size (per sample)	$3 \times 224 \times 224$
Vaccine recall samples per class	5
Number of classes per task	8
Recall samples per task	40
Recall set size per task	$8 \times 5 \times 3 \times 224 \times 224$
Batch processing	Single batch on GPU
Training samples	$\sim$1600
Optimization updates	2000
GPU model	NVIDIA RTX 4080
GPU time for recall generation per task	$\sim$0.12 GPU hours
Peak Memory for each task per client	5033.313 MB

E Plasticity–Stability Analysis

To better demonstrate the plasticity–stability trade-off of FedVaccine, we include additional comparisons with representative FCL baselines, covering different methodological paradigms on ImageNet-R. Specifically, we consider **FedTA (CVPR 2025)**, a regularization-based method that introduces trainable tail anchors along with selective knowledge fusion to mitigate spatial–temporal forgetting; **GLFC (CVPR 2022)**, a hybrid regularization-based approach that combines class-aware gradient compensation with class-semantic relation distillation; and **CAN (ICML 2025)**, a generative replay-based method that leverages a learned generator to synthesize historical data via client-guided trajectories.

We evaluate (1) the average learning time on new tasks ($t > 1$), (2) the total training time, and (3) the relative performance on new tasks, defined as the relative performance change between the accuracy on newly learned tasks and the final average accuracy ($ACC_{\text{relative}} = \frac{ACC_{\text{new}} - A_N}{A_N} \times 100$), (4) the average immediate new-task accuracy ($ACC_{new} = \frac{1}{T-1} \sum_{t=2}^{T} ACC_t$).

These metrics are designed to jointly characterize the plasticity–stability trade-off from both efficiency and performance perspectives.

- The average learning time on new tasks directly reflects the plasticity efficiency, i.e., how quickly the model can adapt to incoming tasks.
- The total training time further captures the overall computational overhead introduced by different mechanisms, which is particularly important when comparing with methods that rely on additional regularization terms or auxiliary models.
- The relative performance on new tasks ($ACC_{relative}$) measures the degree to which new task learning is preserved under stability constraints, providing an indicator of whether maintaining past knowledge comes at the expense of adapting to new tasks. Therefore, negative values indicate that the model performs worse on newly learned tasks compared with its final average performance, while positive values indicate better adaptation to new tasks.
- The average immediate new-task accuracy measures the model's adaptation ability on the newly arrived task.

Together, these metrics offer a comprehensive view of how effectively a method balances fast adaptation to new tasks with the retention of previously acquired knowledge.

Table 11: Plasticity and efficiency comparison on ImageNet-R.

Type	Reg.	Reg.	Reg.	Reg. + Replay	Replay	Recall
Method	LoRM	FedMGP	FedTA	GLFC	CAN	FedVaccine
time on new task	110.32s	122.25s	100.62s	113.83s	124.36s	90.20s
total time	4.52h	5.13h	3.94h	4.53h	5.44h	4.27h
$ACC_{relative}$	-7.23%	+2.52%	-4.35%	-6.50%	-2.27%	+6.83%
ACC_{new} of 15 epochs	42.75	54.82	58.57	41.08	48.66	64.36
ACC_{new} of 25 epochs	73.64	80.53	83.27	65.31	75.92	88.42

As shown in Table 11, FedVaccine achieves a favorable balance between plasticity and stability across all three metrics.

First, regarding adaptation efficiency, FedVaccine achieves the lowest learning time on new tasks (90.20s), outperforming all baselines. It suggests that existing preservation-based mechanisms introduce additional constraints during adaptation. For example, FedMGP and LoRM preserve historical knowledge through prompt optimization and LoRA consolidation, while replay-based methods require knowledge retrieval or generation. In contrast, FedVaccine allows controlled forgetting during adaptation, enabling faster acquisition of new knowledge.

Second, regarding total training cost, FedVaccine achieves a competitive overhead of 4.27h, substantially lower than GLFC (4.53h) and CAN (5.44h), while remaining comparable to FedTA (3.94h), LoRM (4.52h), and FedMGP (5.13h). This shows that the proposed recall mechanism avoids the high cost of continually maintaining historical knowledge.

Finally, FedVaccine achieves the highest (+6.83%), outperforming all baselines. Although FedMGP demonstrates that regularization-based methods can balance stability and plasticity to some extent, and CAN alleviates forgetting through replay, both still constrain efficient adaptation to new tasks.

These results suggest that the limitation of existing methods lies not in neglecting plasticity, but in preserving old knowledge during adaptation. By decoupling forgetting and recalling into two stages, FedVaccine avoids this conflict and provides a more effective continual learning paradigm.

F Vaccine Samples V.S. Real Buffer

We include an additional ablation where each client stores a small buffer of real samples per class and replays them during recall (ignoring privacy constraints, as suggested). We ensure a fair comparison by matching the memory budget with the size of the recall vaccine set.

The results in Table 12 show that, although naive replay with real samples provides a reasonable baseline, it is consistently outperformed by FedVaccine on both accuracy and forgetting.

Specifically, FedVaccine achieves significantly higher final average accuracy (A_N) and substantially lower forgetting (F_N), indicating that the proposed recall vaccine set is more effective at both

Table 12: Comparison between the proposed recall vaccine set and naive replay with a real sample buffer. For fair comparison, both methods use the same memory budget per client.

Method	$A_N\uparrow$	$F_N\downarrow$
Naive Replay (real buffer)	73.15±2.38	8.67±1.36
FedVaccine (ours)	84.51±1.10	3.70±0.98

preserving past knowledge and supporting continual adaptation. This is noteworthy given that naive replay directly stores real data samples, which are often considered an upper bound for replay-based methods when privacy constraints are ignored.

The performance gap suggests that the recall vaccine set is not merely acting as a compressed surrogate of stored data. Instead, it captures more task-relevant and transferable information by explicitly encoding gradient-aligned knowledge for recall. In contrast, naive replay relies on raw sample storage, which may be redundant or suboptimal under limited memory budgets, and lacks an explicit mechanism to emphasize task-critical directions in the parameter space.

G Plug-in Ability

We further investigate the compatibility and generality of the proposed vaccine set mechanism by integrating it into representative FCL frameworks with different underlying paradigms.

Specifically, for replay-based methods (CAN and MFCL), we replace their original replay buffers with the generated vaccine sets, removing the need for storing raw or generated samples. For LoRA-based methods (LoRM), we incorporate our "Forget & Recall" mechanism as an additional module to enhance their ability to recover previously learned knowledge without modifying their core architecture.

Table 13: Plug-in ability of the proposed vaccine set mechanism on representative FCL methods.

Method		Original	with Vaccine-set
CAN	$A_N\uparrow$	73.53	77.82
	$F_N\downarrow$	7.68	4.35
MFCL	$A_N\uparrow$	69.43	72.82
	$F_N\downarrow$	15.56	12.38
LoRM	$A_N\uparrow$	73.68	76.13
	$F_N\downarrow$	3.93	3.73

Experimental results on ImageNet-R are summarized in Table 13. Overall, the proposed vaccine-based mechanism can be seamlessly integrated into all considered methods and consistently improves their performance, demonstrating its effectiveness as a plug-and-play component.

For replay-based methods, replacing the original buffers with vaccine sets leads to notable improvements in both A_N and F_N. This suggests that the vaccine sets provide a more informative and compact representation of historical knowledge compared to conventional replay strategies, which often rely on raw or generated samples without explicitly encoding task-relevant update directions.

For LoRA-based methods, incorporating the "Forget & Recall" mechanism also improves the final accuracy, indicating enhanced knowledge recovery capability. Although a slight increase in forgetting is observed, the overall performance gain suggests a better balance between plasticity and stability, as the model becomes more capable of adapting to new tasks while still retaining useful prior knowledge.

These results highlight that the proposed vaccine mechanism is architecture-agnostic and can effectively complement existing strategies.

H Privacy Discussion

FedVaccine avoids storing or transmitting raw training data by constructing a compact recall vaccine set through gradient matching. Instead of preserving individual samples, the proposed method compresses the information of an entire training dataset into a small number of synthetic samples whose objective is to approximate the aggregate gradient behavior of real data. This many-to-few compression inherently reduces the identifiability of any single training example, as each recall sample corresponds to a mixture of gradients induced by multiple data points rather than a one-to-one reconstruction. Consequently, recovering original data from the recall set becomes significantly more ill-posed compared to directly accessing gradients or real samples.

Moreover, the recall vaccine set is optimized with respect to task-level objectives and model states, rather than explicitly reconstructing input distributions. This further limits the fidelity of potential reconstructions, as the optimization prioritizes gradient alignment over pixel- or instance-level accuracy. In this sense, the recall samples can be viewed as task-relevant surrogates rather than faithful replicas of the underlying data distribution.

From a system perspective, the compact size of the recall vaccine set also enables the adoption of stronger cryptographic protections with reduced overhead. For instance, encrypting a small number of synthetic samples using secure aggregation or homomorphic encryption is significantly more efficient than protecting full datasets or dense gradient updates, making privacy-preserving deployment practical.

In addition, the generation process allows for explicit control over the privacy-utility trade-off. By introducing controlled noise into the gradient matching objective or limiting the alignment precision, the recall samples can be regularized to avoid overfitting to sensitive patterns, thereby reducing the risk of information leakage.

Nevertheless, we acknowledge that gradient-based representations may still carry residual information about the training data under strong adversarial assumptions. A rigorous analysis of privacy leakage (e.g., under gradient inversion attacks) remains an important direction for future work. Incorporating formal privacy mechanisms, such as differential privacy during vaccine generation, could further strengthen the theoretical guarantees.

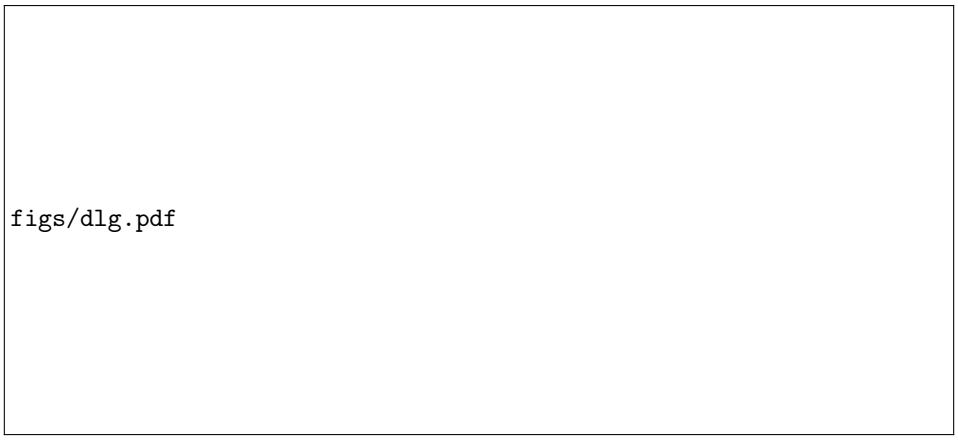


Figure 6: illustration of DLG samples.

To empirically validate the privacy advantages of the proposed recall vaccine set, we conduct gradient inversion attacks using Deep Leakage from Gradients (DLG). Specifically, we apply DLG under a unified setting with the same pre-trained model ViT B/16, and compare reconstruction results from gradients computed on (i) real training samples and (ii) the recall vaccine set. This controlled setup ensures that the only difference lies in the source of gradients.

The results in Figure 6 show that gradients derived from real samples can recover visually recognizable structures, whereas those obtained from the recall vaccine set lead to severely degraded reconstructions with significant distortion and loss of semantic content. This performance gap indicates that the many-to-few compression induced by gradient matching not only reduces data dimensionality but

also amplifies the ill-posedness of inversion. Consequently, even under strong attackers equipped with gradient inversion techniques, the recall vaccine set provides substantially improved resistance against data reconstruction, offering empirical evidence for its privacy-preserving properties.

We further conduct quantitative evaluations using PSNR and SSIM to measure reconstruction fidelity.

Table 14: Quantitative evaluations for the privacy leakage.

Attack Source	PSNR ↑	SSIM ↑
Raw Gradient	24.83	0.812
Vaccine Set	13.47	0.326

The results show that DLG attacks on raw gradients can recover substantially more visual information, achieving higher reconstruction quality. In contrast, reconstructions from vaccine sets exhibit significantly lower PSNR and SSIM values, indicating reduced sample-level information leakage.

I Limitations

Despite its promising performance, FedVaccine has several limitations.

First, the current framework is primarily designed and evaluated on visual recognition tasks, leveraging representations from pre-trained vision transformers. Although the proposed vaccine-based recall mechanism is conceptually general, its applicability to other modalities, such as natural language processing or multimodal learning, remains underexplored. Extending the framework to these domains may require redesigning the vaccine generation process to account for modality-specific structures and training dynamics. **Second,** FedVaccine is well-suited for the task-incremental setting, where recall can be explicitly triggered based on task identity. In more challenging scenarios such as class-incremental or task-agnostic settings, where task boundaries are unknown or blurred, determining when and what to recall becomes non-trivial. This may limit its effectiveness in fully online or highly dynamic environments.

J Broader Impacts

This work presents FedVaccine, a federated continual learning framework that enables efficient knowledge recall through compact synthetic data (recall vaccine sets) without storing raw historical data. The method is primarily intended for privacy-sensitive distributed learning scenarios.

Positive impacts. FedVaccine may contribute to privacy-preserving machine learning by reducing the need to store or transmit raw data across clients. This is particularly relevant in domains such as healthcare or finance, where data retention is costly or restricted. In addition, the compactness of the recall vaccine set can lower communication and storage overhead, improving the accessibility of continual learning for resource-constrained devices.

Potential risks and negative impacts. Since the recall process is designed to approximate task-specific gradients, it may inherit and preserve biases present in local client data. When deployed in real-world systems, this could lead to unfair or skewed model behavior across different user groups.

Another limitation arises when the method is used in task-agnostic or highly dynamic environments. Incorrect recall triggering or inaccurate vaccine sets may lead to degraded predictions, which could negatively impact downstream decision-making in safety-critical applications.

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 contributions and scope are clearly stated in the Abstract and Introduction, consistent with the experimental and theoretical results.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: Appendix I clearly discusses the limitation that the current method is only applicable to vision tasks and has not been validated on other modalities, as well as the fact that it performs best on task-incremental tasks.

   Guidelines:

   - The answer [N/A] 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: Theorem 1 (Fast Recall) and its complete proof can be found in Appendix A (pages 13–15). The assumptions (strong convexity, smoothness, unbiased gradients, etc.) are clearly listed.

Guidelines:

- The answer [N/A] 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: Section 5.1 provides the dataset splits and baseline settings; Appendix C.2 (Table 6) provides all key hyper-parameters (learning rate, LoRA rank, recall set size, etc.); Appendix B provides the algorithm pseudo-code. Meanwhile, we provide the code in the supplemental materials.

Guidelines:

- The answer [N/A] 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 provide the code in the supplemental materials.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5.1 details the number of training epochs, optimizer, learning rate, weight decay, LoRA insertion positions, the number of recall set generation iterations, etc.; Appendix C.2 provides additional configurations.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: All experimental results report the mean and standard deviation over three random seeds (see Tables 2, 3, etc.), conforming to the standard practice for statistical significance reporting.

   Guidelines:

   - The answer [N/A] 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: Section 5.1 specifies the use of a single NVIDIA RTX 4080 GPU, Ubuntu 22.04, and PyTorch 2.4.0; Appendix D (Table 8) provides the GPU time ($\tilde{0}$.12 hours) and peak memory (5033 MB) for generating the recall set for each task.

   Guidelines:

   - The answer [N/A] 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 paper adheres to the NeurIPS ethical guidelines.

   Guidelines:

   - The answer [N/A] 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: Appendix J.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not involve releasing high-risk models (e.g., pre-trained language models, image generators) or crawled datasets, and thus no special safeguards are required.

    Guidelines:

    - The answer [N/A] 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: The paper cites the datasets used (CIFAR-100, ImageNet-R, CUB-200) and pre-trained models (ViT-B/16, MoCo-v3, EVA-02), and the original papers and common licenses (e.g., CC BY, Apache 2.0, etc.) of these assets are properly cited.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: [N/A]

    Guidelines: The paper does not release any new datasets or models.

    - The answer [N/A] 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: [N/A]

    Justification: [N/A]

    Guidelines: The paper does not involve crowdsourcing or experiments with human subjects.

    - The answer [N/A] 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: [N/A]

    Justification: [N/A]

Guidelines: The paper does not involve human subjects research and therefore does not require IRB approval.

- The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: Only for writing and editing.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.