ANCHOR-GUIDED PERSONALIZATION FOR FEDERATED PROMPT LEARNING IN VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated prompt learning provides a parameter-efficient way to adapt vision-language models while keeping the pre-trained backbone frozen. However, under heterogeneous client distributions, a single global prompt may be too generic to capture client specialization, while sharing personalized prompts keeps knowledge tied to individual clients and requires communicating client-specific representations. This raises a key challenge: how can clients benefit from specialized knowledge learned across the federation while keeping personalized prompts local? To address this challenge, we propose FedAGP (Federated Anchor-Guided Personalization), which enables specialized knowledge sharing through a compact set of group anchors. From these anchors, each client retrieves a class-dependent mixture according to its local characteristics. The resulting retrieval weights are accumulated into anchor usage that further guides server-side aggregation, allowing the same client–anchor relevance to govern both retrieval and refinement without communicating personalized prompts. To improve global coordination, we further introduce consensus-guided global aggregation, which adjusts the contribution of global prompt updates according to their directional agreement. Extensive experiments on 11 heterogeneous federated benchmarks demonstrate that FedAGP achieves a strong balance between personalization and generalization over existing federated prompt learning methods.

1 INTRODUCTION

Large vision-language models (VLMs) have become a strong foundation for visual recognition, enabling broad transfer through aligned image and text representations Radford et al. (2021); Shu et al. (2022); Wang et al. (2023); Zhang et al. (2024); Laurençon et al. (2024); Yang et al. (2025). In real-world deployments, however, data are naturally distributed across users and devices, while privacy and communication constraints often make centralized collection impractical. Federated learning (FL) provides a natural paradigm for such settings by enabling multiple clients to train collaboratively while keeping their data local McMahan et al. (2017); Li et al. (2020; 2021). Yet, directly adapting large VLM backbones in FL is costly: full-model fine-tuning places a substantial computational burden on clients and incurs prohibitive communication overhead.

Federated prompt learning (FPL) has therefore emerged as an appealing alternative Guo et al. (2023b;a); Li et al. (2024); Su et al. (2024); Luo et al. (2025); Bo et al. (2025). By optimizing only a small set of prompt parameters while keeping the pre-trained backbone frozen, FPL enables parameter-efficient adaptation with lightweight communication. Recent methods further introduce personalized prompts to cope with heterogeneous client distributions. For example, FedPGP Cui et al. (2024) combines a shared prompt with low-rank client-specific adaptation, while pFedMoAP Luo et al. (2025) allows each client to combine its local prompt with non-local prompt experts. However, personalized prompts do not fully resolve the challenge of knowledge sharing across heterogeneous clients. Clients may differ substantially in label distribution, domain, and data quantity Kairouz & McMahan (2021); Zhang et al. (2022); Li et al. (2020); Ye et al. (2023a); Le et al. (2026); consequently, knowledge that is useful to one client may be less relevant to another, while updates produced from different local objectives may push the shared prompt in conflicting directions.

This observation raises two fundamental questions, as shown in Figure 1. First, *how can a client benefit from specialized knowledge learned by other clients without exchanging personalized prompts*

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

or forcing all shared knowledge into a single global prompt? Existing approaches mainly rely on these two forms of knowledge sharing. A global prompt provides common knowledge across clients, but may be too generic to capture the specialization of heterogeneous clients. In contrast, sharing personalized prompts preserves client-specific knowledge but keeps collaboration at the individual-client level and requires communicating personalized prompts. This motivates sharing specialized knowledge across clients in a way that is more flexible than a single global prompt while avoiding the direct exchange of personalized prompts. Second, *how should global knowledge be consolidated under heterogeneous updates?* Standard averaging treats local updates uniformly, although each update can contain both a direction consistently supported across clients and a deviation induced by the client's own data distribution. Repeatedly accumulating these client-specific deviations may gradually bias the global prompt toward particular clients and weaken the broad generalization inherited from the pre-trained model.

Motivated by these challenges, we propose **FedAGP** (**Fed**erated **A**nchor **G**uided **P**ersonalization), which enables clients to access and collaboratively refine shared prompt knowledge through a compact set of group anchors. Each client uses a signature derived from its private prompt to perform class-dependent retrieval from the group anchors, refines the most relevant anchor using local visual prototypes, and contributes to server-side anchor aggregation according to its accumulated usage. In this way, FedAGP enables clients to share specialized knowledge across the federation while keeping personalized prompts local. To coordinate the global level, we further introduce consensus-guided aggregation, which weights global prompt updates according to their directional agreement with the collective update.

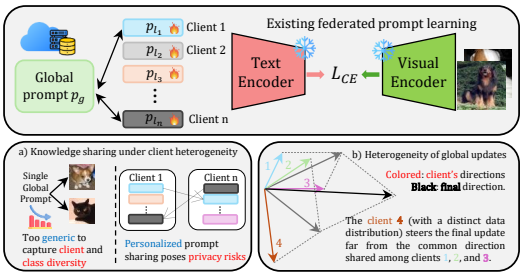

Figure 1: **Background and motivation.** (a) Existing federated prompt learning mainly shares knowledge through a global prompt or client-specific prompts, which may be too generic or remain tied to individual clients. (b) Under client heterogeneity, local global-prompt updates may disagree, causing uniform aggregation to deviate from directions shared across multiple clients.

Our main contributions are summarized as follows:

- We investigate how clients can benefit from specialized knowledge learned across the federation without exchanging personalized prompts or relying on a single global prompt for all shared knowledge.
- We propose FedAGP, a novel framework that enables specialized knowledge sharing through a compact set of group anchors. Clients perform class-dependent anchor retrieval using private signatures, refine relevant anchors with local visual prototypes, and contribute to their aggregation according to anchor usage, while personalized prompts remain local. We further introduce consensus-guided aggregation to coordinate heterogeneous global-prompt updates.
- We conduct extensive experiments on 11 heterogeneous federated benchmarks, demonstrating consistent improvements in personalization and generalization and validating the contribution of the proposed components through comprehensive ablation studies.

2 Related Works

2.1 Heterogeneous Federated Learning

Federated learning (FL) McMahan et al. (2017); Mohri et al. (2019); Xu et al. (2021); Quan et al. (2025) enables multiple clients to collaboratively train a model while keeping their data decentralized. However, heterogeneous client distributions can significantly degrade the performance of conventional aggregation methods such as FedAvg Wang et al. (2021); Tan et al. (2023); Huang et al. (2022); Wang et al. (2025). To address this issue, personalized federated learning learns client-specific models through local fine-tuning, regularization, parameter decomposition, or personalized aggregation (Tan et al., 2022; Collins et al., 2021; T Dinh et al., 2020; Acar et al., 2021; Chen et al., 2024; Yi et al., 2026). Other approaches exploit client similarity or clustering to promote knowledge sharing among

clients with related distributions Guo et al. (2024); Yang et al. (2024), while recent studies improve global aggregation by accounting for heterogeneous or conflicting local updates Ye et al. (2023b); Xia et al. (2025). Different from these methods that mainly consider conventional model training, our work investigates heterogeneous knowledge sharing in federated prompt learning for vision-language models.

2.2 Federated Prompt Learning

Prompt tuning adapts pre-trained vision-language models through a small set of learnable parameters while keeping the backbone frozen Radford et al. (2021); Zhou et al. (2022); Khattak et al. (2023); Hao et al. (2025); Xiao et al. (2025), making it particularly suitable for federated learning. PromptFL Guo et al. (2023b) and pFedPrompt Guo et al. (2023a) establish early federated prompt learning frameworks, while subsequent methods further improve personalization under heterogeneous client distributions. FedPGP Cui et al. (2024) combines global and low-rank personalized prompts, FedOTP Li et al. (2024) models global-local prompt interaction through optimal transport, and FedPHA Fang et al. (2025) aligns global and flexible personalized prompts to reduce their discrepancy. Among the most related methods, pFedMoAP (Luo et al., 2025) combines local prompts with non-local prompt experts, while FedMGP (Bo et al., 2025) maintains multiple paired textual and visual prompt groups within each client. In both cases, the shared knowledge remains tied to individual clients. In contrast, FedAGP introduces a shared group level between global and personalized prompts. It learns a common set of M anchors that can be shared across multiple clients, performs class-wise retrieval from these anchors, and reuses the resulting retrieval weights to guide anchor aggregation without communicating personalized prompts. Section 4.3 further evaluates the contribution of these design choices.

3 Methodology

3.1 Preliminaries

We consider a federated learning system with N clients. Each client i owns a private dataset $\mathcal{D}_i$, whose labels belong to a local class set $\mathcal{C}_i \subseteq \mathcal{C}$, where $\mathcal{C}$ denotes the global class set. We refer to classes in $\mathcal{C}_i$ as *observed* and those in $\mathcal{C} \setminus \mathcal{C}_i$ as *unobserved* for client i. Due to data heterogeneity, both the observed classes and their data distributions may differ across clients. At communication round t, the server selects a subset of participating clients $\mathcal{S}_t$, broadcasts the shared parameters, and aggregates their locally updated parameters after training. Following federated prompt learning, we adopt a frozen vision-language model with image encoder $f(\cdot)$ and text encoder $g(\cdot)$. A learnable prompt $p \in \mathbb{R}^{L\times d}$ consists of L context tokens. For a class c, its text representation is obtained as $t^{(c)} = g([p, c])$. Given an image $\mathbf{x}$, prediction probabilities can be calculated using the similarity scores between image feature and text feature:

$$q(\hat{y} = c \mid \mathbf{x}) = \frac{\exp\big(\mathrm{sim}(f(\mathbf{x}), t^{(c)})/\tau\big)}{\sum_{k=1}^{C} \exp\big(\mathrm{sim}(f(\mathbf{x}), t^{(k)})/\tau\big)}, \tag{1}$$

where $\mathrm{sim}(\cdot,\cdot)$ denotes cosine similarity and τ is a temperature parameter. Only the prompt parameters are optimized using the cross-entropy loss $\mathcal{L}_{\mathrm{CE}} = -\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}[\log q(\hat{y} = y \mid \mathbf{x})]$, while the pre-trained image and text encoders remain frozen. FedAGP addresses the two questions raised in Section 1 through two complementary components: *anchor-guided personalization*, which enables knowledge sharing at the intermediate level, and *consensus-guided global aggregation*, which coordinates global knowledge under heterogeneous local updates, as shown in Figure 2.

3.2 Anchor-Guided Personalization

A single global prompt provides a common representation across all clients, whereas a personalized prompt reflects only the specialization of one client. To bridge these two extremes, FedAGP introduces shared group anchors that can be utilized by multiple clients according to their local characteristics. We represent prompt knowledge at three complementary levels:

$$\underbrace{p_{\ell_i}}_{\text{client-specific}}, \qquad \underbrace{\mathcal{A} = \{a_m\}_{m=1}^{M}}_{\text{group-level}}, \qquad \underbrace{p_g}_{\text{global}}, \tag{2}$$

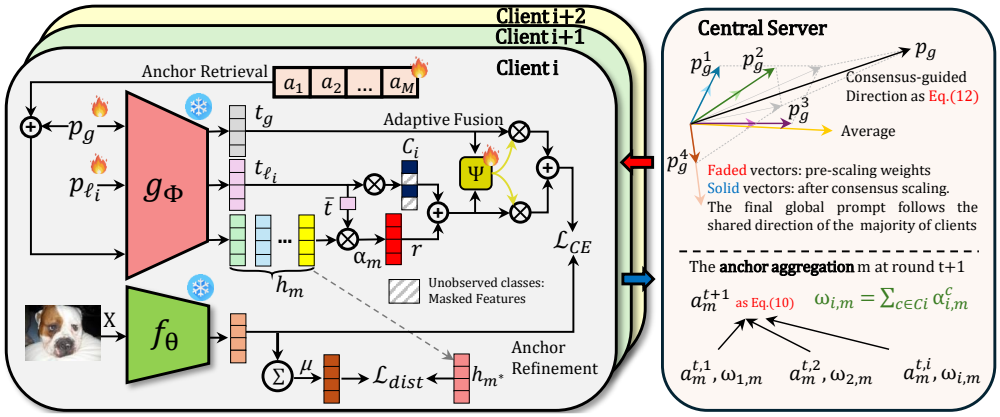

Figure 2: **Overview of FedAGP.** Each client performs class-dependent retrieval from the shared group anchors using its private prompt, refines the most relevant anchor with local visual prototypes, and contributes to server-side anchor aggregation according to its anchor usage (Sec. 3.2). Personalized prompts remain local, while consensus-guided aggregation coordinates the global prompt. (Sec. 3.3).

where $p_{\ell_i}, a_m, p_g \in \mathbb{R}^{L\times d}$. The personalized prompt p_{ℓ_i} captures the client-specific knowledge and remains local to client i. In contrast, the global prompt p_g captures knowledge shared across the federation, while the M group anchors capture intermediate knowledge shared among related clients. Both p_g and $\mathcal{A}$ are communicated between the server and participating clients. Instead of storing and sharing personalized prompts from individual clients, FedAGP uses the group anchors as shared representations to capture recurring knowledge across clients.

Anchor retrieval and fusion. Each group anchor is parameterized as a residual around the current global prompt, so that it can preserve globally shared knowledge while capturing group-specific variation. For class c, anchor m produces *anchor text representation* $h_m^{(c)} = g([p_g + a_m, c])$, where p_g provides the common reference and a_m models the specialization around it. However, the relevance of each anchor may vary across both clients and classes. Assigning each client to a fixed anchor assumes that all classes within the client benefit from the same shared knowledge, which may not hold under heterogeneous label distributions. To capture the client-specific component, client i first summarizes its local specialization through $\bar{t}_i = \frac{1}{|\mathcal{C}_i|}\sum_{c\in\mathcal{C}_i} t_{\ell_i}^{(c)}$, where $t_{\ell_i}^{(c)} = g([p_{\ell_i}, c])$. The resulting $\bar{t}_i$ serves as a private *client signature* and remains on-device. For each class c, client i first computes the relevance of anchor m as the similarity between $h_m^{(c)}$ and $\bar{t}_i$. The resulting scores are then normalized across all anchors to obtain the class-wise *retrieval weights*:

$$s_{i,m}^{(c)} = \left\langle h_m^{(c)}, \bar{t}_i \right\rangle, \qquad \alpha_{i,m}^{(c)} = \frac{\exp(s_{i,m}^{(c)}/\tau_a)}{\sum_{m'=1}^{M}\exp(s_{i,m'}^{(c)}/\tau_a)}, \tag{3}$$

where τ_a controls the concentration of the *retrieval weights* $\alpha_{i,m}^{(c)}$ over the group anchors. Based on these weights, the retrieved group-level representation for class c is computed as:

$$r_i^{(c)} = \sum_{m=1}^{M} \alpha_{i,m}^{(c)} h_m^{(c)}. \tag{4}$$

This soft and class-wise retrieval allows different classes from the same client to use different mixtures of shared anchors, rather than assigning the entire client to a single group. The retrieved group representation captures knowledge shared across related clients, but it does not replace the knowledge specific to each client. Therefore, FedAGP combines $r_i^{(c)}$ with the client's personalized representation $t_{\ell_i}^{(c)}$. Since the personalized prompt is trained only on classes observed by client i, we introduce a *class-support constraint* that conditions its contribution on the client's observed class set during both training and inference:

$$\tilde{t}_{\ell_i}^{(c)} = r_i^{(c)} + \delta_i^{(c)} t_{\ell_i}^{(c)}, \qquad \delta_i^{(c)} = \mathbb{I}[c \in \mathcal{C}_i], \tag{5}$$

where $\delta_i^{(c)}$ indicates whether class c is observed by client i. For an observed class, $\tilde{t}_{\ell_i}^{(c)}$ combines the retrieved group knowledge with the client's own specialization, promoting personalization on local classes. For an unobserved class, the personalized representation is excluded and $\tilde{t}_{\ell_i}^{(c)}$ reduces to the retrieved group representation, which is shared across clients and therefore transfers beyond the local class set. The global prompt provides a complementary representation that captures knowledge shared across the federation. To adaptively balance globally shared and client-specialized knowledge, we introduce a lightweight client-specific gating function $\Psi_i(\cdot)$. Given the global representation $t_g^{(c)}$ and the personalized/group representation $\tilde{t}_{\ell_i}^{(c)}$, the gate produces two fusion weights:

$$\boldsymbol{\lambda}_i^{(c)} = [\lambda_i^{g,(c)}, \lambda_i^{\ell,(c)}] = \mathrm{softmax}\Big(\Psi_i\Big([t_g^{(c)}; \tilde{t}_{\ell_i}^{(c)}]\Big)\Big), \tag{6}$$

where $t_g^{(c)} = g([p_g, c])$, and $\lambda_i^{g,(c)}$ and $\lambda_i^{\ell,(c)}$ denote the contributions of the global and personalized/group representations, respectively. The final text representation is then computed by:

$$t_i^{(c)} = \lambda_i^{g,(c)} t_g^{(c)} + \lambda_i^{\ell,(c)} \tilde{t}_{\ell_i}^{(c)}. \tag{7}$$

This adaptive fusion allows each client to balance globally shared knowledge with locally relevant knowledge for each class.

Anchor refinement and aggregation. Optimizing the anchors only through the local classification objective may make them overly dependent on the locally learned text representations. To provide an auxiliary semantic reference for anchor refinement, we use local visual prototypes obtained from the frozen image encoder, which provides a common visual embedding space across clients. For each observed class c, client i maintains an exponential-moving-average visual prototype

$$\mu_i^{(c)} \leftarrow (1-\beta)\mu_i^{(c)} + \beta \frac{1}{|\mathcal{B}_i^{(c)}|} \sum_{\mathbf{x} \in \mathcal{B}_i^{(c)}} f(\mathbf{x}), \tag{8}$$

where $\mathcal{B}_i^{(c)}$ contains samples of class c in the current mini-batch and β controls the update rate. Since the image encoder is frozen across clients, $\mu_i^{(c)}$ provides a stable visual reference for refining the group anchor in the shared vision-language embedding space. The retrieval weights indicate how strongly each client relies on the shared anchors. We compute the *usage* of anchor m by client i across the observed classes as $\omega_{i,m} = \sum_{c \in \mathcal{C}_i} \alpha_{i,m}^{(c)}$ and select the *dominant anchor* as $m_i^* = \arg\max_m \omega_{i,m}$. We refine this dominant anchor using the local visual prototypes:

$$\mathcal{L}_{\mathrm{dist}}^{(i)} = \frac{1}{|\mathcal{P}_i|} \sum_{c \in \mathcal{P}_i} \Big[1 - \cos\Big(h_{m_i^*}^{(c)}, \mathrm{sg}[\mu_i^{(c)}]\Big)\Big], \tag{9}$$

where $\mathcal{P}_i \subseteq \mathcal{C}_i$ contains classes with valid visual prototypes and $\mathrm{sg}[\cdot]$ denotes stop-gradient. This objective aligns the dominant anchor with the client's local visual semantics. By applying this visual supervision only to the dominant anchor, FedAGP concentrates local refinement on the group-level knowledge most relevant to the client while avoiding unnecessary updates to less relevant anchors.

We also use $\omega_{i,m}$ to determine the contribution of each client's local anchor update during server aggregation. Let $a_{i,m}^t$ denote the locally updated copy of anchor m returned by client i. The server updates each anchor as:

$$a_m^{t+1} = \frac{\sum_{i \in \mathcal{S}_t} \omega_{i,m} a_{i,m}^t}{\sum_{i \in \mathcal{S}_t} \omega_{i,m}}, \qquad m = 1, \ldots, M. \tag{10}$$

Overall, the class-wise retrieval weights $\alpha_{i,m}^{(c)}$ determine how strongly client i relies on each shared anchor, while their accumulated usage $\omega_{i,m}$ determines the client's contribution to that anchor during server aggregation. Thus, the same client-anchor relevance guides both local retrieval and global anchor refinement. Over communication rounds, this consistent use of anchor relevance allows the shared anchors to capture recurring knowledge across related clients, while personalized prompts remain local.

3.3 CONSENSUS-GUIDED GLOBAL AGGREGATION

The group anchors capture knowledge shared among subsets of clients, while the global prompt is designed to preserve knowledge that is broadly useful across the federation. Under heterogeneous local objectives, however, an update to the global prompt can contain both a direction supported by multiple and a client-specific deviation induced by the local data distribution. Uniform averaging does not distinguish between these effects. As client-specific deviations accumulate over communication rounds, they may bias the global prompt toward the particular clients and weaken its generalization. To address this issue, FedAGP weights each global-prompt update according to its agreement with the collective update direction. Let $\Delta_i^t = p_g^{t,i} - p_g^t$ denote the global-prompt update returned by client i at round t. We estimate the collective direction from the mean update and measure the directional agreement of each client using cosine similarity:

$$\bar{\Delta}^t = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \Delta_i^t, \qquad \rho_i^t = \frac{\langle \Delta_i^t, \bar{\Delta}^t \rangle}{\|\Delta_i^t\|_2 \|\bar{\Delta}^t\|_2}. \tag{11}$$

The agreement scores are converted into normalized aggregation weights, and the global prompt is updated as:

$$\gamma_i^t = \frac{\exp(\rho_i^t / \tau_g)}{\sum_{j \in \mathcal{S}_t} \exp(\rho_j^t / \tau_g)}, \qquad p_g^{t+1} = p_g^t + \sum_{i \in \mathcal{S}_t} \gamma_i^t \Delta_i^t, \tag{12}$$

where τ_g controls the sensitivity to directional agreement. Because ρ_i^t is based on cosine similarity, the weighting reflects the direction of an update rather than its magnitude. Updates that are more aligned with the collective direction receive larger weights, while conflicting updates receive less. The global prompt is therefore shaped primarily by directions that recur across clients, complementing the group-level knowledge sharing provided by the anchors.

3.4 TRAINING OBJECTIVE AND COMMUNICATION

For the final text representation $t_i^{(c)}$, prediction follows Eq. 1, and client i minimizes the following local objective:

$$\mathcal{L}_i = \mathcal{L}_{\text{CE}}^{(i)} + \lambda_{\text{dist}} \mathcal{L}_{\text{dist}}^{(i)}, \tag{13}$$

where λ_{dist} controls the contribution of visual prototype distillation. The classification loss jointly optimizes the global prompt, personalized prompt, group anchors, and local gate, while the distillation term provides complementary visual supervision for anchor refinement. At communication round t, the server broadcasts the global prompt p_g^t and group anchors $\mathcal{A}^t$ to the participating clients. After local optimization, client i returns its updated global prompt $p_g^{t,i}$, locally updated group anchors $\mathcal{A}^{t,i}$, and anchor-usage weights $\{\omega_{i,m}\}_{m=1}^M$. The personalized prompt p_{ℓ_i}, client-specific gate Ψ_i remain on-device. The server then updates the group anchors through Eq. 10 and the global prompt through Eq. 12 before beginning the next communication round.

4 EXPERIMENT

4.1 EXPERIMENTAL SETUP

Datasets and Data Heterogeneity. We conduct experiments across 11 public benchmark datasets under various settings to simulate various types of data heterogeneity. Following previous works Guo et al. (2023b); Cui et al. (2024), we select 7 datasets to evaluate the base-to-novel class generalization ability including: OxfordPets Parkhi et al. (2012), Flowers102 Nilsback & Zisserman (2008), DTD Cimpoi et al. (2014), Caltech101 Fei-Fei et al. (2004), Food101 Bossard et al. (2014), UCF101 Soomro et al. (2012), and Stanford Cars Krause et al. (2013). These datasets span a wide range of visual recognition tasks including fine-grained classification, texture classification, action recognition, and general object recognition. Classes in each dataset are equally divided into base and novel splits, and non-overlapping base classes are distributed to different clients to form a pathological non-IID setting. We train each client on its local classes and evaluate on three test sets: local classes, base classes held by other clients, and novel classes which are unseen during the training process. To evaluate performance under label heterogeneity, we use CIFAR-10 and CIFAR-100 Krizhevsky et al. (2009) partitioned across clients using a Dirichlet distribution, which creates varying levels of label

Table 1: Performance comparison of clients' local accuracy and generalization across 7 datasets.

Method	Average on 7 datasets				OxfordPets				Flowers102				DTD			
	Local	Base	Novel	HM	Local	Base	Novel	HM	Local	Base	Novel	HM	Local	Base	Novel	HM
Zero-Shot CLIP Radford et al. (2021)	72.51	75.49	81.20	76.10	89.07	89.54	96.85	91.69	67.87	67.98	76.89	70.67	43.02	54.26	59.55	51.31
PromptFL Guo et al. (2023b)	80.56	82.24	79.10	80.49	95.30	95.28	92.57	94.37	87.49	82.02	75.34	81.31	54.13	67.20	57.91	59.26
FedOTP Li et al. (2024)	93.34	94.87	37.83	60.92	98.63	99.82	52.22	76.32	98.55	98.20	35.24	61.59	90.40	96.94	36.56	61.56
FedPGP Cui et al. (2024)	89.20	85.97	77.53	83.54	97.29	94.37	97.09	96.23	96.52	86.96	72.96	84.35	65.30	76.71	51.22	62.66
FedPHA Fang et al. (2025)	96.63	98.05	58.67	78.31	99.28	99.94	81.46	92.72	99.18	99.76	46.94	72.45	91.80	97.46	42.16	66.86
pFedMoAP Luo et al. (2025)	96.68	79.89	78.76	83.90	99.32	91.09	96.57	95.54	99.20	75.38	75.03	81.80	90.80	60.92	54.55	65.56
FedMGP Bo et al. (2025)	94.27	95.82	67.26	82.20	97.33	99.59	94.06	96.94	97.84	98.55	56.65	78.91	90.27	95.41	38.91	63.48
FedAGP (Ours)	96.60	97.75	79.91	**90.21**	99.40	99.90	94.32	**97.80**	99.16	99.53	75.24	**89.76**	91.21	96.61	60.36	**79.19**
Δ		+6.31				+0.86				+5.41				+12.33		

Method	Caltech101				Food101				UCF101				Stanford Cars			
	Local	Base	Novel	HM	Local	Base	Novel	HM	Local	Base	Novel	HM	Local	Base	Novel	HM
Zero-Shot CLIP Radford et al. (2021)	91.91	96.03	94.03	93.96	85.24	89.41	90.69	88.38	65.00	67.84	74.98	69.03	65.45	63.36	75.43	67.69
PromptFL Guo et al. (2023b)	93.51	96.64	94.02	94.70	87.00	90.14	89.99	89.02	77.84	77.84	72.55	75.99	68.67	66.57	71.35	68.81
FedOTP Li et al. (2024)	98.00	99.29	62.94	82.96	92.99	94.25	39.82	64.55	86.86	89.10	17.94	38.23	87.95	86.51	20.08	41.25
FedPGP Cui et al. (2024)	97.29	96.24	93.46	95.64	87.19	90.25	89.11	88.83	91.93	76.93	72.55	79.66	88.87	80.36	66.34	77.38
FedPHA Fang et al. (2025)	98.60	99.75	82.41	92.87	96.20	97.01	68.65	85.06	95.07	95.76	48.97	72.50	96.25	96.65	40.11	65.69
pFedMoAP Luo et al. (2025)	99.09	98.01	93.58	96.83	96.40	89.01	90.14	91.74	95.64	80.39	69.69	80.55	96.32	64.42	71.73	75.29
FedMGP Bo et al. (2025)	98.09	99.12	81.95	92.35	93.78	94.55	83.04	90.14	93.55	93.75	59.56	78.65	89.01	89.77	56.67	74.96
FedAGP (Ours)	99.05	99.73	95.57	**98.08**	96.94	97.74	90.61	**94.99**	95.20	95.87	71.91	**86.10**	95.23	94.86	71.34	**85.57**
Δ		+1.25				+3.25				+5.55				+8.19		

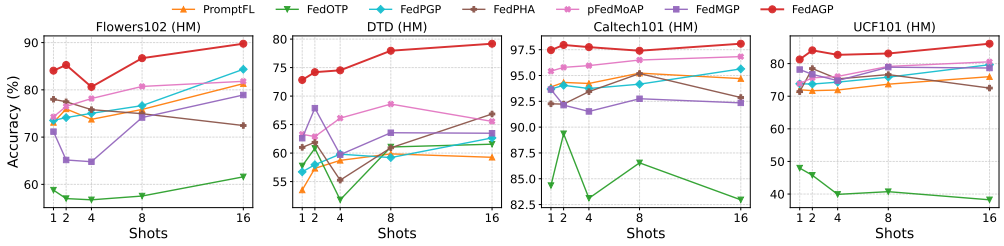

Figure 3: Few-shot analysis with HM accuracy across four datasets.

heterogeneity across clients. For feature and label heterogeneity, we adopt DomainNet dataset Peng et al. (2019) and OfficeHome dataset Venkateswara et al. (2017), with 6 and 4 domains, respectively. Dataset details are provided in Appendix B.

Baselines. We compare FedAGP against seven methods: Zero-shot CLIP (ICML'21) Radford et al. (2021), PromptFL (IEEE TMC'23) Guo et al. (2023b), FedOTP (CVPR'24) Li et al. (2024), FedPGP (ICML'24) Cui et al. (2024), FedPHA (ICML'25) Fang et al. (2025), pFedMoAP (ICLR'25) Luo et al. (2025), and FedMGP (NeurIPS'25) Bo et al. (2025). All baselines are re-implemented from their official repositories under the same training configuration for fair comparison. Detailed descriptions of each baseline are provided in Appendix E.

Table 2: Performance comparison on CIFAR-10 and CIFAR-100 with label shift with $\mathrm{Dir}(\alpha = 0.5)$ partition.

Methods	CIFAR-10	CIFAR-100
Zero-Shot CLIP Radford et al. (2021)	87.85	64.85
PromptFL Guo et al. (2023b)	92.35	74.47
FedOTP Li et al. (2024)	94.24	76.75
FedPGP Cui et al. (2024)	91.99	74.73
FedPHA Fang et al. (2025)	93.76	75.90
pFedMoAP Luo et al. (2025)	95.01	79.45
FedMGP Bo et al. (2025)	94.15	78.42
FedAGP	**95.38**	**80.30**

Implementation Details. All methods are implemented using a frozen CLIP model with ViT-B/16 Dosovitskiy et al. (2021) as the backbone. We use the SGD optimizer with a learning rate of 0.001 and batch sizes of 32 for training and 128 for testing. Local training epochs are set to $E = 1$, and communication rounds are set to $T = 50$ for all datasets except CIFAR-10 and CIFAR-100, where $T = 100$. Final performance is averaged over the last 5 communication rounds. The number of clients varies by dataset: for single-domain CLIP datasets (OxfordPets, Flowers102, DTD, Caltech101, Food101, UCF101, Stanford Cars), we use $N = 10$ with each client holding a distinct class subset and 16-shot samples per class; for multi-domain datasets (OfficeHome, DomainNet), we set N to twice the number of domains, assigning two clients per domain, with data within each domain partitioned using Dirichlet distribution $\mathrm{Dir}(\alpha = 0.5)$; for CIFAR-10 and CIFAR-100, we use $N = 100$ with 10% client participation per round, where data is partitioned across clients using $\mathrm{Dir}(\alpha = 0.5)$.

4.2 Performance Evaluation

Base-to-Novel class generalization. We evaluate each method's ability to generalize from trained local classes to base classes held by other clients and novel classes unseen during training. We report top-1 accuracy on each set: Local (personalization), Base (cross-client transfer), and Novel (generalization). To capture overall performance, we use the harmonic mean $\mathrm{HM} = \frac{3}{\frac{1}{\mathrm{Local}} + \frac{1}{\mathrm{Base}} + \frac{1}{\mathrm{Novel}}}$,

Table 3: Performance comparison on multi-domain datasets with domain and label heterogeneity.

Methods	OfficeHome					DomainNet						
	A	C	P	R	Avg	C	I	P	Q	R	S	Avg
Zero-Shot CLIP Radford et al. (2021)	84.45	66.25	89.50	89.66	82.47	72.38	53.30	66.53	13.19	83.51	66.55	59.24
PromptFL Guo et al. (2023b)	86.42	76.49	94.53	92.96	87.60	86.09	70.56	79.76	33.36	91.40	80.39	73.59
FedOTP Li et al. (2024)	81.11	76.81	92.70	87.65	84.57	86.28	72.30	81.69	48.85	90.97	84.09	77.36
FedPGP Cui et al. (2024)	88.35	78.46	95.35	93.66	88.96	89.58	79.59	88.07	51.66	94.54	87.49	81.89
FedPHA Fang et al. (2025)	88.62	79.12	**96.18**	93.49	89.35	90.24	80.08	89.11	51.97	94.67	**88.63**	82.45
pFedMoAP Luo et al. (2025)	88.45	78.65	95.75	93.24	89.02	90.03	78.68	87.84	**56.42**	94.59	88.03	82.60
FedMGP Bo et al. (2025)	87.52	76.59	93.52	92.02	87.41	87.37	76.77	84.79	52.55	92.46	84.55	79.75
FedAGP (Ours)	**89.80**	**79.72**	96.02	**94.65**	**90.05**	**91.04**	**80.84**	**89.62**	52.74	**94.69**	88.38	**82.88**

which jointly measures personalization and generalization. A method that achieves high Local but low Base or Novel will be penalized, encouraging balanced performance across all three sets. As shown in Table 1, FedAGP achieves the best average HM of 90.21% across the seven datasets, outperforming the second-best pFedMoAP by 6.31%. More importantly, the improvement is driven by a better transfer of knowledge beyond each client's locally observed classes. FedAGP maintains Local accuracy comparable to the strongest personalized baselines, while achieving substantially stronger Base accuracy than methods such as pFedMoAP and retaining competitive Novel performance. FedPHA exhibits strong Local and Base accuracy but degrades considerably on Novel classes, whereas pFedMoAP preserves Novel performance but transfers less effectively to Base classes held by other clients. FedAGP balances these three settings, indicating that the shared-anchor mechanism improves access to cross-client knowledge without sacrificing local adaptation or novel-class transfer. We report variability over random seeds in Appendix C.1.

Evaluation on label heterogeneity. We evaluate on CIFAR-10 and CIFAR-100 partitioned across 100 clients with 10% client participation per round using $\mathrm{Dir}(\alpha = 0.5)$. As shown in Table 2, FedAGP achieves the best accuracy on both datasets with 95.38% on CIFAR-10 and 80.30% on CIFAR-100, respectively, demonstrating its effectiveness under severe label heterogeneity.

Evaluation on feature and label heterogeneity. As shown in Table 3, FedAGP achieves the highest average accuracy across different domains on both datasets with 90.05% on OfficeHome and 82.88% on DomainNet. Under the conditions where clients differ in both visual appearance and class proportions, methods relying on uniform aggregation such as PromptFL and FedOTP illustrate the limited cross-domain transfer. FedAGP addresses this heterogeneity through class-dependent retrieval, which allows clients to draw on different shared representations, while consensus-guided aggregation reduces the influence of conflicting cross-domain updates on the global prompt.

Few-shot analysis. We evaluate FedAGP under varying shot settings from 1 to 16 shots per class on four datasets. As shown in Figure 3, FedAGP achieves the best HM accuracy across all settings. Even in the extreme 1-shot regime, our proposed method maintains competitive performance.

4.3 Ablation Study

To better understand the design choices of FedAGP, we conduct ablation studies on its main components and key variants. All results are reported as the average performance across Flowers102, DTD, Caltech101, and UCF101. Further ablation studies are reported in Appendix C.

Effect of key components. Table 4(a) evaluates the contribution of the main components of FedAGP. AGP provides the largest improvement, particularly for transfer beyond locally observed classes. Removing either the group anchors or the prototype distillation objective degrades the overall personalization–generalization balance, showing the importance of both collaborative anchor sharing and anchor refinement. To further isolate the effect of cross-client anchor sharing, *Local-only anchors* retains the same anchor capacity, class-dependent retrieval, prototype supervision, class-support constraint, and adaptive fusion as FedAGP, but keeps each client's anchors local without server-side anchor aggregation. This variant maintains comparable Local and Base accuracy, but reduces Novel accuracy by 4.58% and HM by 2.32%. This performance gap highlights the benefit of cross-client group-anchor sharing. CGA provides a smaller but complementary improvement.

Retrieval weights. Table 4(b) compares different ways of constructing $\alpha_{i,m}^{(c)}$, while keeping the read operation in Eq. (4) unchanged. *Random* assigns each class to a single anchor through a fixed mapping shared by all clients, making the weights class-dependent but neither learned nor client-specific. *Uniform* sets $\alpha_{i,m}^{(c)} = 1/M$, reducing the group branch to an equal mixture of all anchors. *Client-wise* computes one retrieval distribution for each client and reuses it across all classes. In contrast, our

Table 4: Ablation study on FedAGP design choices, including key components, retrieval weights, feature fusion, and anchor aggregation.

(a) Components

Variant	Local	Base	Novel	HM
Local-only anchors	96.66	97.53	71.19	85.97
w/o AGP	96.60	98.12	48.04	65.32
w/o anchors	96.58	98.08	48.33	67.74
w/o $\mathcal{L}_{\text{dist}}$	96.35	97.98	63.64	81.31
w/o CGA	95.89	97.89	73.09	86.84
FedAGP (ours)	96.15	97.94	75.77	**88.29**

(b) Retrieval weights

Variant	Local	Base	Novel	HM
Random	95.75	97.43	58.84	77.20
Uniform	96.45	98.09	70.52	85.70
Client-wise	96.15	97.87	73.08	86.91
Class-wise (ours)	96.15	97.94	75.77	**88.29**

(c) Feature fusion

Variant	Local	Base	Novel	HM
w/o δ	95.40	97.04	70.10	84.97
Fixed gate	95.97	97.27	72.50	86.42
w/o both	95.24	96.57	70.21	84.81
FedAGP (ours)	96.15	97.94	75.77	**88.29**

(d) Anchor aggregation

Variant	Local	Base	Novel	HM
Uniform	96.22	98.04	72.00	86.41
Random	96.23	98.02	73.26	87.07
Averaging	96.26	98.09	73.45	87.19
Usage-weighted (ours)	96.15	97.94	75.77	**88.29**

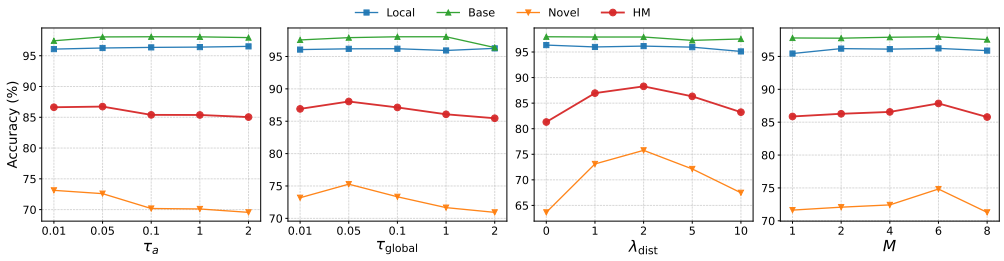

Figure 4: Sensitivity to different hyperparameters across four datasets.

Class-wise strategy preserves both client- and class-specific relevance, allowing different classes from the same client to retrieve different mixtures of group anchors and improves HM over *Client-wise*.

Feature fusion. Table 4(c) examines whether personalized knowledge should be conditioned on local class support and whether the final fusion should adapt to each representation. *w/o* $\delta_i^{(c)}$ removes the class-support constraint, allowing the personalized representation to contribute to both observed and unobserved classes. *Fixed gate* replaces the learnable gate with equal fusion weights, $\lambda_i^{g,(c)} = \lambda_i^{\ell,(c)} = 0.5$, while *w/o both* removes the class-support constraint and simultaneously uses the fixed $0.5/0.5$ fusion.

Anchor aggregation. Table 4(d) compares usage-weighted aggregation in Eq. 10 with three variants: *Uniform* assigns equal weight to all clients, *Averaging* uses local data size, and *Random* samples Dirichlet weights for each anchor. Usage-weighted aggregation achieves the best HM of 88.29%, outperforming data-size weighting (87.19%) and uniform aggregation (86.41%). Unlike these variants, our method accounts for how strongly each client actually uses a given anchor, reducing the influence of clients with less relevance to that anchor.

Hyperparameter sensitivity. Figure 4 analyzes the retrieval temperature τ_a, consensus temperature τ_g, distillation weight λ_{dist}, and number of group anchors M. Local and Base accuracy remain relatively stable across most settings, while Novel accuracy shows greater variation and consequently drives much of the change in HM. Moderate values of the two temperatures avoid overly concentrated or overly uniform weighting, while a moderate distillation strength and number of anchors provide a suitable balance between anchor specialization and shared knowledge. We use $\tau_a = 0.05$, $\tau_g = 0.05$, $\lambda_{\text{dist}} = 2$, and $M = 6$ by default.

5 Conclusion

In this paper, we proposed FedAGP, a federated prompt learning framework for sharing specialized knowledge across clients under heterogeneous data distributions. FedAGP uses shared group anchors to enable class-dependent retrieval and refinement of knowledge learned across the federation, while keeping personalized prompts local. The accumulated anchor usage further guides anchor aggregation, and consensus-guided global aggregation coordinates heterogeneous global prompt updates according to their directional agreement. Experiments on 11 heterogeneous benchmarks show that FedAGP improves access to knowledge beyond locally observed classes while maintaining strong local adaptation and novel-class transfer.

AI USE STATEMENT

In this work, we used generative AI tools solely to aid and polish the writing of the manuscript, including grammar correction, improving clarity and conciseness. We have not used generative AI tools for research ideation, methodology design, experimental design or execution, code development, or data analysis; the proposed method, all experiments, and all reported results are the work of the authors. Retrieval and literature search were likewise conducted by the authors. We have reviewed all AI-assisted writing: every suggested edit was checked by the authors for factual accuracy and consistency with the reported results, and no claim, number, or citation was adopted without verification against the underlying experiments and original sources. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021.

Weihao Bo, Yanpeng Sun, Yu Wang, Xinyu Zhang, and Zechao Li. FedMGP: Personalized federated learning with multi-group text-visual prompts. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=E6SFbnPiVP.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

Haokun Chen, Yao Zhang, Denis Krompass, Jindong Gu, and Volker Tresp. Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 11285–11293, 2024.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3606–3613, 2014.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pp. 2089–2099. PMLR, 2021.

Tianyu Cui, Hongxia Li, Jingya Wang, and Ye Shi. Harmonizing generalization and personalization in federated prompt learning. In *International Conference on Machine Learning*, pp. 9646–9661. PMLR, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Chengying Fang, Wenke Huang, Guancheng Wan, Yihao Yang, and Mang Ye. Fedpha: Federated prompt learning for heterogeneous client adaptation. In *Forty-second International Conference on Machine Learning*, 2025.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pp. 178–178. IEEE, 2004.

Tao Guo, Song Guo, and Junxiao Wang. Pfedprompt: Learning personalized prompt for vision-language models in federated learning. In *Proceedings of the ACM Web Conference 2023*, pp. 1364–1374, 2023a.

Tao Guo, Song Guo, Junxiao Wang, Xueyang Tang, and Wenchao Xu. Promptfl: Let federated participants cooperatively learn prompts instead of models–federated learning in age of foundation model. *IEEE Transactions on Mobile Computing*, 23(5):5179–5194, 2023b.

Yongxin Guo, Xiaoying Tang, and Tao Lin. Fedrc: Tackling diverse distribution shifts challenge in federated learning by robust clustering. In *International Conference on Machine Learning*, pp. 16910–16944. PMLR, 2024.

Fusheng Hao, Fengxiang He, Fuxiang Wu, Tichao Wang, Chengqun Song, and Jun Cheng. Task-aware clustering for prompting vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 14745–14755, 2025.

Wenke Huang, Mang Ye, and Bo Du. Learn from others and be yourself in heterogeneous federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10143–10153, 2022.

Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and trends in machine learning*, 14(1-2):1–210, 2021.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19113–19122, 2023.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pp. 554–561, 2013.

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

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models? *Advances in Neural Information Processing Systems*, 37:87874–87907, 2024.

Huy Q. Le, Loc X. Nguyen, Yu Qiao, Seong Tae Kim, Eui-Nam Huh, and Choong Seon Hong. Feddap: Domain-aware prototype learning for federated learning under domain shift. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3390–3399, June 2026.

Hongxia Li, Wei Huang, Jingya Wang, and Ye Shi. Global and local prompts cooperation via optimal transport for federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12151–12161, 2024.

Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10713–10722, 2021.

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE signal processing magazine*, 37(3):50–60, 2020.

Jun Luo, Chen Chen, and Shandong Wu. Mixture of experts made personalized: Federated prompt learning for vision-language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=xiDJaTim3P.

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*, pp. 1273–1282. PMLR, 2017.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International conference on machine learning*, pp. 4615–4625. PMLR, 2019.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pp. 722–729. IEEE, 2008.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

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

Minh K Quan, Pubudu N Pathirana, Mayuri Wijayasundara, Sujeeva Setunge, Dinh C Nguyen, Christopher G Brinton, David J Love, and H Vincent Poor. Federated learning for cyber physical systems: a comprehensive survey. *IEEE Communications Surveys & Tutorials*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. *Advances in Neural Information Processing Systems*, 35:14274–14289, 2022.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Shangchao Su, Mingzhao Yang, Bin Li, and Xiangyang Xue. Federated adaptive prompt tuning for multi-domain collaborative learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 15117–15125, 2024.

Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in neural information processing systems*, 33:21394–21405, 2020.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE transactions on neural networks and learning systems*, 34(12):9587–9603, 2022.

Yue Tan, Chen Chen, Weiming Zhuang, Xin Dong, Lingjuan Lyu, and Guodong Long. Is heterogeneity notorious? taming heterogeneity to handle test-time shift in federated learning. *Advances in neural information processing systems*, 36:27167–27180, 2023.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5018–5027, 2017.

Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. Addressing class imbalance in federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 10165–10173, 2021.

Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19175–19186, 2023.

Zheng Wang, Zihui Wang, Xiaoliang Fan, and Cheng Wang. Federated learning with domain shift eraser. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4978–4987, 2025.

Endre Weiszfeld. Sur le point pour lequel la somme des distances de n points donnés est minimum. *Tohoku Mathematical Journal, First Series*, 43:355–386, 1937.

Zeke Xia, Ming Hu, Dengke Yan, Ruixuan Liu, Anran Li, Xiaofei Xie, and Mingsong Chen. Multisfl: Towards accurate split federated learning via multi-model aggregation and knowledge replay. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 914–922, 2025.

Zehao Xiao, Shilin Yan, Jack Hong, Jiayin Cai, Xiaolong Jiang, Yao Hu, Jiayi Shen, Cheems Wang, and Cees G. M. Snoek. Dynaprompt: Dynamic test-time prompt tuning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=EFZEdHB3Mp.

Jie Xu, Benjamin S Glicksberg, Chang Su, Peter Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of healthcare informatics research*, 5(1):1–19, 2021.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19792–19802, 2025.

Xiyuan Yang, Wenke Huang, and Mang Ye. Fedas: Bridging inconsistency in personalized federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11986–11995, 2024.

Mang Ye, Xiuwen Fang, Bo Du, Pong C Yuen, and Dacheng Tao. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56(3):1–44, 2023a.

Rui Ye, Mingkai Xu, Jianyu Wang, Chenxin Xu, Siheng Chen, and Yanfeng Wang. Feddisco: Federated learning with discrepancy-aware collaboration. In *International Conference on Machine Learning*, pp. 39879–39902. PMLR, 2023b.

Liping Yi, Han Yu, Gang Wang, Xiaoguang Liu, and Qinghua Hu. pfedmoe: Data-level personalization with mixture of experts in model-heterogeneous personalized federated learning. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5625–5644, 2024.

Tuo Zhang, Lei Gao, Chaoyang He, Mi Zhang, Bhaskar Krishnamachari, and A Salman Avestimehr. Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet of Things Magazine*, 5(1):24–29, 2022.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International journal of computer vision*, 130(9):2337–2348, 2022.

A FedAGP Algorithm

We illustrate the process of FedAGP in Algorithm 1.

B Experimental Details

Table 5: Dataset statistics and federated setup used in our experiments.

Dataset	Task	Train	Test	Classes	Domains	# Clients	Client Participation Rate
Base-to-Novel Generalization							
OxfordPets Parkhi et al. (2012)	Fine-grained Recognition	2,944	3,669	37	1	10	100%
Flowers102 Nilsback & Zisserman (2008)	Fine-grained Recognition	4,093	2,463	102	1	10	100%
DTD Cimpoi et al. (2014)	Texture Classification	2,820	1,692	47	1	10	100%
Caltech101 Fei-Fei et al. (2004)	Object Recognition	4,128	2,465	101	1	10	100%
Food101 Bossard et al. (2014)	Fine-grained Recognition	50,500	30,300	101	1	10	100%
UCF101 Soomro et al. (2012)	Action Recognition	7,639	3,783	101	1	10	100%
Stanford Cars Krause et al. (2013)	Fine-grained Recognition	6,509	8,041	196	1	10	100%
Label Heterogeneity							
CIFAR-10 Krizhevsky et al. (2009)	Object Recognition	50,000	10,000	10	1	100	10%
CIFAR-100 Krizhevsky et al. (2009)	Object Recognition	50,000	10,000	100	1	100	10%
Feature and Label Heterogeneity							
OfficeHome Venkateswara et al. (2017)	Domain Adaptation	12,475	3,113	65	4	8	100%
DomainNet Peng et al. (2019)	Domain Adaptation	111,132	47,937	100	6	12	100%

All methods are implemented using a frozen CLIP model with ViT-B/16 Dosovitskiy et al. (2021) as the backbone in PyTorch, and all experiments are conducted on NVIDIA A6000 GPUs. We use SGD optimizer with a learning rate of 0.001 and batch sizes of 32 for training and 128 for testing. Local training epochs are set to $E = 1$, and communication rounds are set to $T = 50$ for all datasets except CIFAR-10 and CIFAR-100, where $T = 100$. Final performance is averaged over the last 5 communication rounds.

Algorithm 1 FedAGP

Input: Communication rounds T, local epochs E, clients N, anchors M, local dataset $\mathcal{D}_i$, retrieval temperature τ_a, consensus temperature τ_g, distillation weight λ_{dist}, prototype momentum β.
Output: Global prompt p_g, group anchors $\mathcal{A} = \{a_m\}_{m=1}^{M}$, personalized prompts $\{p_{\ell_i}\}_{i=1}^{N}$ (kept on-device)

Server Execution:
Initialize p_g^1 and anchors $a_m^1 \sim \mathcal{N}(0, \sigma^2)$ for $m = 1, \dots, M$
for $t = 1, \dots, T$ **do**
  Select a subset $\mathcal{S}_t$ of clients and broadcast p_g^t, $\mathcal{A}^t$
  for each client $i \in \mathcal{S}_t$ **in parallel do**
    $p_g^{t,i}, \mathcal{A}^{t,i}, \{\omega_{i,m}^t\} \leftarrow$ **LocalUpdate**$(i, p_g^t, \mathcal{A}^t)$
  end for
  /* Usage-weighted anchor aggregation (Sec. 3.2) */
  for $m = 1, \dots, M$ **do**
    $a_m^{t+1} \leftarrow \big(\sum_{i\in\mathcal{S}_t} \omega_{i,m}^t a_{i,m}^t\big) / \big(\sum_{i\in\mathcal{S}_t} \omega_{i,m}^t\big)$ via Eq. (10)
    if $\sum_{i\in\mathcal{S}_t} \omega_{i,m}^t = 0$ **then** $a_m^{t+1} \leftarrow a_m^t$ /* unused slot */
  end for
  /* Consensus-guided global aggregation (Sec. 3.3) */
  $\Delta_i^t \leftarrow p_g^{t,i} - p_g^t$ for each $i \in \mathcal{S}_t$
  $\bar{\Delta}^t \leftarrow \frac{1}{|\mathcal{S}_t|} \sum_{i\in\mathcal{S}_t} \Delta_i^t$; $\rho_i^t \leftarrow \cos(\Delta_i^t, \bar{\Delta}^t)$ via Eq. (11)
  $\gamma_i^t \leftarrow \text{softmax}_i(\rho_i^t/\tau_g)$; $p_g^{t+1} \leftarrow p_g^t + \sum_{i\in\mathcal{S}_t} \gamma_i^t \Delta_i^t$ via Eq. (12)
end for

LocalUpdate$(i, p_g, \mathcal{A})$:
for $e = 1, \dots, E$ **do**
  for each batch $(\mathbf{x}, y) \in \mathcal{D}_i$ **do**
    /* Class-conditional retrieval */
    $h_m^{(c)} \leftarrow g([p_g + a_m, c])$; $\bar{t}_i \leftarrow$ mean of $t_{\ell_i}^{(c)}$ over $c \in \mathcal{C}_i$
    $\alpha_{i,m}^{(c)} \leftarrow \text{softmax}_m(\langle h_m^{(c)}, \bar{t}_i \rangle / \tau_a)$; $r_i^{(c)} \leftarrow \sum_m \alpha_{i,m}^{(c)} h_m^{(c)}$ via Eqs. (3), (4)
    /* Class-support constraint */
    $\tilde{t}_{\ell_i}^{(c)} \leftarrow r_i^{(c)} + \delta_i^{(c)} t_{\ell_i}^{(c)}$, $\delta_i^{(c)} = \mathbb{I}[c \in \mathcal{C}_i]$ via Eq. (5)
    $\hat{t}_i^{(c)} \leftarrow \lambda_i^{g,(c)} t_g^{(c)} + \lambda_i^{\ell,(c)} \tilde{t}_{\ell_i}^{(c)}$ via Eqs. (6), (7)
    /* Visual prototype refinement */
    Update prototypes $\mu_i^{(c)}$ by EMA via Eq. (8)
    $m_i^* \leftarrow \arg\max_m \omega_{i,m}^t$, where $\omega_{i,m}^t = \sum_{c\in\mathcal{C}_i} \alpha_{i,m}^{(c)}$
    Update p_g, p_{ℓ_i}, $\mathcal{A}$, Ψ_i by minimizing $\mathcal{L}_{\text{CE}}^{(i)} + \lambda_{\text{dist}} \mathcal{L}_{\text{dist}}^{(i)}$ via Eq. (13)
  end for
end for
return p_g, $\mathcal{A}$, $\{\omega_{i,m}^t\}_{m=1}^{M}$ /* p_{ℓ_i}, Ψ_i, $\mu_i^{(c)}$ stay local */

Base-to-Novel Generalization. We select 7 datasets spanning diverse visual recognition tasks: OxfordPets Parkhi et al. (2012), Flowers102 Nilsback & Zisserman (2008), DTD Cimpoi et al. (2014), Caltech101 Fei-Fei et al. (2004), Food101 Bossard et al. (2014), UCF101 Soomro et al. (2012), and Stanford Cars Krause et al. (2013). Classes in each dataset are equally divided into base and novel splits, and non-overlapping base classes are distributed across $N = 10$ clients to form a pathological non-IID setting, with each client holding 16-shot samples per class. Each client is trained on its local classes and evaluated on three test sets: local classes, base classes held by other clients, and novel classes unseen during training.

Label Heterogeneity. We use CIFAR-10 and CIFAR-100 Krizhevsky et al. (2009) partitioned across $N = 100$ clients using a Dirichlet distribution Dir($\alpha = 0.5$), with 10% client participation per round. This setting creates varying degrees of label imbalance across clients, simulating realistic non-IID conditions.

Feature and Label Heterogeneity. We adopt OfficeHome Venkateswara et al. (2017) and DomainNet Peng et al. (2019) to evaluate performance under the combined challenge of feature and label

Table 6: Local accuracy (%) across 7 datasets, reported as mean ± standard deviation over three random seeds.

Methods	OxfordPets	Flowers102	DTD	Caltech101	Food101	StanfordCars	UCF101	Avg
Zero-Shot CLIP Radford et al. (2021)	88.67±0.25	67.81±0.46	44.76±2.18	92.01±0.10	85.26±0.03	65.34±0.30	64.99±0.16	72.69±0.31
PromptFL Guo et al. (2023b)	92.93±0.34	85.76±1.11	55.29±2.70	93.60±0.73	86.80±0.15	69.10±0.27	74.33±0.19	79.69±0.50
FedOTP Li et al. (2024)	98.21±0.53	98.14±0.62	89.16±1.29	98.67±0.32	92.44±0.49	85.47±1.33	87.39±0.73	92.78±0.37
FedPGP Cui et al. (2024)	93.77±0.29	96.46±1.97	62.39±3.00	95.48±1.26	88.05±0.51	86.09±1.81	89.48±2.55	87.39±1.32
FedPHA Fang et al. (2025)	**99.35**±0.48	99.26±0.29	91.29±1.07	99.43±0.32	96.08±0.23	**95.72**±0.30	**95.25**±0.56	96.63±0.21
pFedMoAP Luo et al. (2025)	99.27±0.30	99.40±0.11	89.89±1.14	**99.44**±0.21	96.40±0.21	95.59±0.49	95.21±0.62	96.46±0.20
FedMGP Bo et al. (2025)	99.12±0.60	99.10±0.17	90.25±0.85	99.10±0.34	95.81±0.31	92.51±0.30	94.67±0.43	95.78±0.33
FedAGP (Ours)	99.20±0.38	**99.42**±0.10	**91.79**±0.83	99.24±0.18	**96.69**±0.30	95.46±1.19	94.90±0.49	**96.74**±0.25

heterogeneity. OfficeHome contains 65 classes across 4 domains: Art-Painting, Clipart, Photo, and Real-World, with $N = 8$ clients and two clients assigned per domain. DomainNet spans 6 domains: Clipart, Infograph, Painting, Quickdraw, Real, and Sketch, from which we select the first 100 classes out of 345, with $N = 12$ clients and two clients per domain. Within each domain, data is further partitioned using Dirichlet distribution Dir($\alpha = 0.5$) to introduce additional label heterogeneity.

C Additional Results

C.1 Evaluation on Personalization

Table 6 reports Local accuracy over three seeds under the pathological non-IID setting, where each client holds a disjoint subset of classes. Methods with client-specific components substantially outperform Zero-Shot CLIP and PromptFL, highlighting the limitation of a single shared prompt under disjoint label spaces. Among the personalized methods, FedAGP achieves the highest average accuracy of 96.74%, followed closely by FedPHA and pFedMoAP with 96.63% and 96.46%, respectively. Their small deviations and overlapping intervals indicate comparable personalization performance. Together with Table 1, these results show that FedAGP maintains competitive performance on locally observed classes while improving the harmonic mean by 6.31%. This suggests that introducing group-level sharing improves generalization without sacrificing local personalization.

C.2 Analysis on Global Prompt Aggregation

We compare consensus-guided aggregation with prompt averaging, the geometric median using Weiszfeld's algorithm Weiszfeld (1937) ($\delta = 10^{-5}$, 50 iterations), and random Dirichlet weights resampled at each round. As shown in Table 8, consensus-guided aggregation achieves the best HM of 88.29%, outperforming prompt averaging by 1.45 points and random weighting by 1.37 points. This indicates that the gain does not arise from non-uniform weighting alone, but from weighting updates according to their agreement with the collective direction. The geometric median performs worst, with an HM of 85.25%. While it is effective for reducing the influence of outlying updates, client disagreement in our setting mainly arises from heterogeneous label distributions rather than corrupted updates. Suppressing updates that deviate from the central tendency may therefore remove useful client-specific information.

C.3 Analysis on distillation target

Table 7 compares different targets for visual prototype distillation. *All anchors* aligns every group anchor with the client's visual prototypes, while *Usage-weighted* distributes the distillation signal across anchors according to their accumulated usage weights $\omega_{i,m}$. Our *Dominant* strategy instead selects m_i^*, the anchor most consistently used by the client, and applies visual supervision only to that anchor. This comparison evaluates whether concentrating the local visual information on the most relevant anchor provides a cleaner group-level refinement than distributing the same supervision across multiple anchors.

C.4 Analysis on the distillation objective

Table 9 compares different objectives for aligning the dominant anchor with the local visual prototype. KL divergence preserves strong performance on observed classes but generalizes poorly to Novel classes, suggesting that distribution-level matching may be less suitable for transferring visual

Table 7: Ablation study on the distillation target.

Setting	Local	Base	Novel	HM
Usage-weighted	96.15	97.73	72.22	86.41
All anchors	96.14	97.81	73.40	87.08
Dominant (ours)	96.15	97.94	75.77	**88.29**

Table 8: Ablation study on global prompt aggregation.

Setting	Local	Base	Novel	HM
Geometric median	96.07	96.93	70.45	85.25
Prompt averaging	95.89	97.89	73.09	86.84
Random	96.21	97.73	72.99	86.92
Consensus (ours)	96.15	97.94	75.77	**88.29**

Table 9: Ablation on the distillation objective, averaged over four datasets.

Objective	Local	Base	Novel	HM
KL divergence	96.42	98.02	68.12	84.28
MSE	95.68	97.57	73.84	87.10
Wasserstein distance	95.64	97.36	73.90	87.02
Cosine distance (ours)	96.15	97.94	75.77	**88.29**

semantics to the anchor representation. MSE and Wasserstein distance provide more balanced results, but remain below our cosine-based objective. Cosine distance achieves the best Novel accuracy and HM, indicating that aligning the direction of the visual and text representations is more effective than directly matching their magnitudes or distributions. This is also consistent with the cross-modal setting, where the relative feature direction provides a natural measure of semantic alignment.

C.5 Evaluation under Varying Label Heterogeneity Level

We further evaluate all methods under different degrees of label heterogeneity by varying the Dirichlet concentration parameter $\alpha \in \{0.1, 0.3, 1.0\}$ on CIFAR-10 and CIFAR-100, as shown in Table 10. FedAGP achieves the best performance across most settings on both datasets, demonstrating its robustness to varying degrees of label heterogeneity.

Table 10: Performance comparison on CIFAR-10 and CIFAR-100 under varying levels of label heterogeneity with Dirichlet partition into 100 clients.

Methods	CIFAR-10			CIFAR-100		
	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 1.0$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 1.0$
PromptFL Guo et al. (2023b)	88.12	91.46	92.88	70.76	73.58	74.82
FedOTP Li et al. (2024)	96.84	95.46	92.75	86.32	79.88	73.82
FedPGP Cui et al. (2024)	87.57	91.74	92.02	79.81	76.45	74.08
FedPHA Fang et al. (2025)	97.16	95.25	92.45	86.79	79.84	73.05
pFedMoAP Luo et al. (2025)	**97.44**	96.24	94.11	88.37	82.69	76.98
FedMGP Bo et al. (2025)	96.03	94.89	92.75	80.90	75.05	75.45
FedAGP (Ours)	97.25	**96.27**	**94.33**	**88.56**	**82.72**	**77.22**

C.6 Effect of the number of group anchors

Table 11 studies the effect of the number of group anchors M with $N = 100$ clients. Notably, FedAGP already achieves strong performance with only a small number of anchors. With only $M = 2$ anchors, FedAGP reaches 95.38% on CIFAR-10 and 80.16% on CIFAR-100, outperforming the compared federated prompt learning baselines on both datasets. This shows that the performance gain does not require a large anchor set, and that a small number of shared anchors can already capture useful knowledge across many clients. Increasing M further results in only moderate performance variation, while the number of anchor parameters and communication cost grow linearly. We therefore use $M = 6$ as the default setting, which provides a favorable balance between performance and communication cost.

D Communication and Computation Efficiency

Table 12 compares the communication and computation costs of FedAGP with existing prompt-based methods. FedAGP introduces $66{,}560$ local trainable parameters, which remains lightweight compared with full-model fine-tuning, although larger than most prompt-based baselines except pFedMoAP. Its communication cost is 112 KB for both upload and download per round, reflecting

Table 11: Effect of the number of anchors M with $N = 100$ clients.

M	Anchor params	Uplink (KB)	CIFAR-10	CIFAR-100
1	8,192	32	94.90	80.07
2	16,384	48	95.38	80.16
4	32,768	80	95.17	79.86
6	49,152	112	95.21	80.30
8	65,536	144	95.29	80.36

the additional transmission of the shared group anchors. In terms of computation, FedAGP requires 25.43 seconds per round and $4{,}556$ MiB of GPU memory, remaining comparable to most baselines and substantially more memory-efficient than FedMGP. Despite this moderate overhead, FedAGP achieves the comparable Local accuracy of 96.60% and the best HM accuracy of 90.21%, indicating a favorable trade-off between computational cost and performance. More details are provided in Appendix E.

Table 12: Comparison of communication and computation costs.

Methods	Local Trainable Params	Communication Cost (per round) Upload (KB)	Download (KB)	Training Time (s)	GPU mem. (MiB)	Avg Local	Avg HM
PromptFL Guo et al. (2023b)	8,192	16	16	12.30	1,972	80.56	80.49
FedOTP Li et al. (2024)	16,384	16	16	22.40	3,648	93.34	60.92
FedPGP Cui et al. (2024)	10,304	16	16	22.84	4,910	89.20	83.54
FedPHA Fang et al. (2025)	16,384	16	16	25.07	4,911	96.63	78.31
pFedMoAP Luo et al. (2025)	74,240	16	160	16.23	2,324	96.68	83.90
FedMGP Bo et al. (2025)	12,800	25	10	32.13	17,239	94.27	82.20
FedAGP	66,560	112	112	25.43	4,556	**96.60**	**90.21**

E Baseline Details and Training Cost Analysis

We provide detailed descriptions of each baseline method along with an analysis of their training costs in terms of locally trainable parameters and communication cost per round, as reported in Table 12. All methods use a frozen CLIP ViT-B/16 backbone, so only prompt-related parameters are counted. Communication cost is computed as the sum of transmitted tensor sizes, where all prompt tensors are in fp16 (2 bytes per parameter).

PromptFL Guo et al. (2023b) is a standard federated prompt learning method that collaboratively learns a single shared text prompt context of shape $[16, 512]$ across clients using FedAvg, yielding 8,192 trainable parameters. The full prompt is uploaded to the server for aggregation and broadcast back, resulting in 16 KB in both directions.

FedOTP Li et al. (2024) applies unbalanced optimal transport to align global knowledge with client-specific prompt representations. It learns two context tensors of the same shape $[16, 512]$: a global prompt and a local prompt, totaling 16,384 parameters. Only the global prompt is communicated, resulting in 16 KB upload and 16 KB download per round.

FedPGP Cui et al. (2024) introduces prompt-wise contrastive learning to balance consistency between global and local prompts. It decomposes the prompt context into a global prompt of shape $[16, 512]$ and two local low-rank factors $\mathbf{U} \in \mathbb{R}^{16\times 4}$ and $\mathbf{V} \in \mathbb{R}^{4\times 512}$ with bottleneck rank $r = 4$, totaling 10,304 parameters. Only the global prompt is communicated, keeping communication at 16 KB per direction.

FedPHA Fang et al. (2025) proposes a global-local prompt architecture with SVD-based projection to resolve conflicts between shared and client-specific knowledge. It learns a multi-prompt context of shape $[2, 16, 512]$, where the first prompt serves as the shared global component and the second as the local component, totaling 16,384 parameters. Only the global prompt slice is communicated, resulting in 16 KB upload and 16 KB download.

pFedMoAP Luo et al. (2025) adopts a mixture-of-experts design that enables each client to access both local and non-local prompt experts through an attention-based gating mechanism. It learns a local prompt context of shape $[16, 512]$ along with a local attention-based gating network (66,048 parameters), totaling 74,240 parameters. The local prompt is uploaded (16 KB), while the download

includes the globally averaged prompt plus $K = 10$ selected expert prompts, resulting in 160 KB download per round.

FedMGP Bo et al. (2025) assigns each client multiple groups of paired textual and visual prompts, together with a similarity-guided dynamic aggregation strategy. It introduces a text context of shape $[5, 2, 512]$ (5,120 parameters) and visual prompt tokens of shape $[5, 2, 768]$ (7,680 parameters), totaling 12,800 parameters in fp16. Both modalities are uploaded, giving $12{,}800 \times 2 = 25{,}600$ bytes $\approx$ 25 KB upload. The server selects the top-$k = 2$ most similar prompt slices per modality and broadcasts them back, resulting in approximately 10 KB download.

FedAGP learns a global prompt p_g and a personalized prompt p_{ℓ_i}, each of shape $[16, 512]$ (8,192 parameters), an anchor $\mathcal{A} = \{a_m\}_{m=1}^{M}$ with M=6 of shape $[6, 16, 512]$ (49,152 parameters), and a lightweight gating network $\Psi\phi$ (1,024 parameters), totaling 66,560 parameters. The personalized prompt and the gating parameters remain local and are never communicated. Each client uploads p_g together with its updated anchor copy and the accompanying anchor-usage mass $\omega^{(i)}$ (112 KB), and downloads the refined consensus prompt and the aggregated anchor (112 KB). Despite the moderate communication overhead, FedAGP achieves the best Avg HM of 90.21%, demonstrating a favorable trade-off between cost and performance.