*p*FedPrompt: Learning Personalized Prompt for Vision-Language Models in Federated Learning

Tao Guo
Hong Kong Polytechnic University
Hong Kong, China
20034349r@connect.polyu.hk

Song Guo*
Hong Kong Polytechnic University
Hong Kong, China
song.guo@polyu.edu.hk

Junxiao Wang*
Hong Kong Polytechnic University
Hong Kong, China
junxiao.wang@polyu.edu.hk

ABSTRACT

Pre-trained vision-language models like CLIP show great potential in learning representations that capture latent characteristics of users. A recently proposed method called Contextual Optimization (*CoOp*) introduces the concept of training prompt for adapting pre-trained vision-language models. Given the lightweight nature of this method, researchers have migrated the paradigm from centralized to decentralized system to innovate the collaborative training framework of Federated Learning (FL). However, current prompt training in FL mainly focuses on modeling user consensus and lacks the adaptation to user characteristics, leaving the personalization of prompt largely under-explored. Researches over the past few years have applied personalized FL (*p*FL) approaches to customizing models for heterogeneous users. Unfortunately, we find that with the variation of modality and training behavior, directly applying the *p*FL methods to prompt training leads to insufficient personalization and performance. To bridge the gap, we present *p*FedPrompt, which leverages the unique advantage of multimodality in vision-language models by learning user consensus from linguistic space and adapting to user characteristics in visual space in a non-parametric manner. Through this dual collaboration, the learned prompt will be fully personalized and aligned to the user's local characteristics. We conduct extensive experiments across various datasets under the FL setting with statistical heterogeneity. The results demonstrate the superiority of our *p*FedPrompt against the alternative approaches with robust performance.

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in ubiquitous and mobile computing**; *Empirical studies in collaborative and social computing*; • **Social and professional topics** → User characteristics.

KEYWORDS

user modeling and personalization, federated learning, prompt learning, vision-language models

*Corresponding authors: Song Guo, Junxiao Wang.

WWW '23, April 30–May 04, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9416-1/23/04…$15.00
https://doi.org/10.1145/3543507.3583518

ACM Reference Format:
Tao Guo, Song Guo, and Junxiao Wang. 2023. *p*FedPrompt: Learning Personalized Prompt for Vision-Language Models in Federated Learning. In *Proceedings of the ACM Web Conference 2023 (WWW '23), April 30–May 04, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3543507.3583518

1 INTRODUCTION

User modeling has been widely employed in Federated Learning (FL) by collaboratively capturing the latent characteristics of users from their behaviors with the exchange of locally obtained parameters [26, 55]. Meanwhile, such cooperative ecosystem has been applied in various scenarios to realize benefits, including recommendation [56], medicine [42], and finance [35].

However, with the significant increase in user data and model capacity, the communication and computational overhead generated by the FL co-modeling process will become increasingly unbearable for users [43]. Even worse, when the model is large, achieving the model's performance inherently requires the user to expose copious amounts of private data to the system [12]. This private information can be recovered from the exchanged parameters or intermediate results, raising potential privacy risks [38, 39, 61].

Fortunately, as pretrained vision-language models like CLIP [44] show great potential in learning representations, a recently proposed method called Contextual Optimization (*CoOp*) [60] introduces the concept of training prompt for adapting pretrained vision-language models. Based on the lightweight nature of this adaptation, researchers [17] have shifted the paradigm of *CoOp* to FL to overcome the problems outlined above. Their core idea is to use *CoOp* at each client to convert context words in prompt into a set of learnable vectors, and to optimize prompt via standard FL algorithm. According to [34, 60], activating the pre-trained knowledge via training prompt is both data- and parameter-efficient, thereby greatly benefiting FL over existing frameworks in terms of computation, communication, and privacy.

Although using prompt in FL to activate the pre-trained knowledge is a promising direction, a major challenge for deploying such approaches in FL is the heterogeneity of users. In this paper, we show that current prompt training is essential to model the user consensus. When the learned consensus is applied to the user's task, the significant gap between them will reduce the effectiveness of user modeling [30, 32, 59]. Research over the past few years has applied personalized FL (*p*FL) approaches to customizing models for heterogeneous users. These model-based *p*FL methods can be categorized into four types: local fine-tuning [8, 50, 53], parameter decomposition [1, 5, 10], regularization [18, 19, 31, 49], and clustering [24, 58]. We investigate a range of vanilla methods by directly applying ideas of personalized methods in the paradigm of

pre-trained models and prompt. Such vanilla methods easily inherit advances of *p*FL, yet are unable to capture the multimodality of vision-language models, thereby leading to insufficient personalization and performance.

As the first attempt to learn personalized prompt in FL, we propose *p*FedPrompt, which takes advantage of the multimodality of vision-language models through two components. Specifically, the Global User Consensus (GUC) component allows full exploration in the word embedding space by globally optimizing continuous vectors, which facilitates the learning of general user consensus. The Local Feature Attention (LFA) component leverages a local personalized attention module by interacting with the spatial visual features in the visual space to dynamically lookup user-relevant features, which adapts the consensus knowledge encoded in GUC by feature retrieval. By incorporating the knowledge retrieved from GUC and LFA, the learned prompt turns out to be personalized according to users' features, so that the user achieves improved accuracy in practical FL classification tasks. Since FL does not exist baseline against any such personalized approach, we implement *p*FedPrompt and other *p*FL methods in the framework as baselines. Extensive experiments spanning a range of popular image classification tasks are conducted under the FL setting. We find that *p*FedPrompt beats baselines with competitive and robust performance. To summarize, the main contributions of this paper are four-fold:

- We find that current prompt training in FL is essentially to model the user consensus and lacks adaptation to user characteristics. We thus propose the problem of learning personalized prompt in FL (see figure 1).
- We survey existing model-based approaches in *p*FL and adapt them into the prompt training manner. We find that these existing personalized techniques cannot capture the multimodality of vision-language models, thereby leading to insufficient personalization and performance.
- To unleash the multimodality, we present *p*FedPrompt, which learns user consensus in linguistic space and adapts to user features on each client in visual space respectively. By incorporating the knowledge retrieved from multimodality, the challenge of user statistical heterogeneity is addressed.
- We evaluate *p*FedPrompt against the existing personalized techniques on widely-adopted datasets. Extensive experiments and ablation studies demonstrate the superiority of our methods.

2 PRELIMINARIES

2.1 User Heterogeneity

The fundamental challenge in addressing user heterogeneity is the presence of non-IID data [26], so we begin by investigating this issue and highlight potential mitigations. While the meaning of IID is generally clear, data can be non-IID in many ways [30]. The most common sources of non-IID data are due to each user corresponding to a particular device, web service, geographic location, and/or time window. For example, users in different regions may have very different disease distributions. There may be more skin cancer patients in southern hemisphere countries than in the northern hemisphere due to the ozone hole. Thus, the label distribution varies from user to user. Another example is that users have different writing styles. In this case, the feature distributions of the users are different. In this paper, we consider differences in the data or feature distribution on each user. According to previous research [21, 27, 32], non-IID data settings reduce the effectiveness of user modeling in FL.

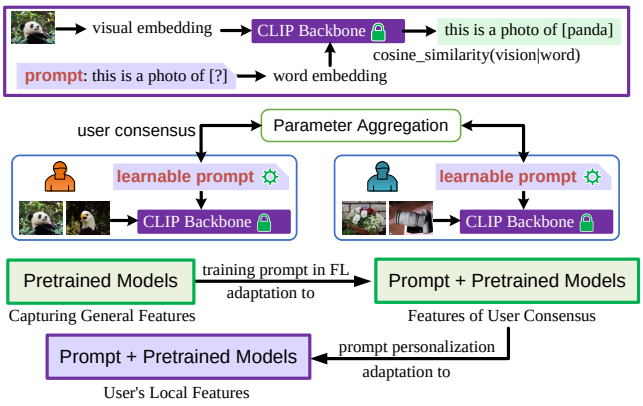

Figure 1: Stages of using pre-trained models with prompt in federated learning: (1) pre-trained vision-language models contain general knowledge that is transferable across a wide range of user modeling; (2) prior work activates the knowledge of pre-trained models by training prompt in the word embedding space so as to model user consensus; (3) our work aims to personalize prompt and further adapt the user consensus to the user's local features.

2.2 Personalized Federated Learning

Federated Learning. The term *federated learning* was introduced by [37]. In a centralized setting, the federated server initially sends global model parameters to each user. Then the server aggregates the user's parameters and transmits the updated model back to each user. In addition to centralized federated learning, there are also some implementations of federated learning based on decentralized frameworks, where the aggregation of parameters occurs in some users [22, 29, 46]. The utilization of stochastic gradient descent (SGD) [4, 11] in FL makes it prone to face statistical challenges, since IID sampling of the training data is important to ensure the unbiased estimate of the full gradient [45]. In practice, it is unrealistic to assume that each user's local data is always IID.

Personalized FL. In recent years, personalized federated learning has received increasing attention due to its potential in handling user statistical heterogeneity. The core idea of model-based *p*FL is to produce customized model structures or parameters for different users. Existing model-based *p*FL methods can be categorized into two types according to the number of global models applied in the server, i.e., single global model, and multiple global models. Single global model type is a close variant of conventional FL algorithms like FedAVG [37], that combines global model optimization process with additional local model customization, and consists of three different kinds of approaches: local fine-tuning [8, 50, 53], parameter decomposition [1, 5, 10], and regularization [18, 19, 31, 49]. These

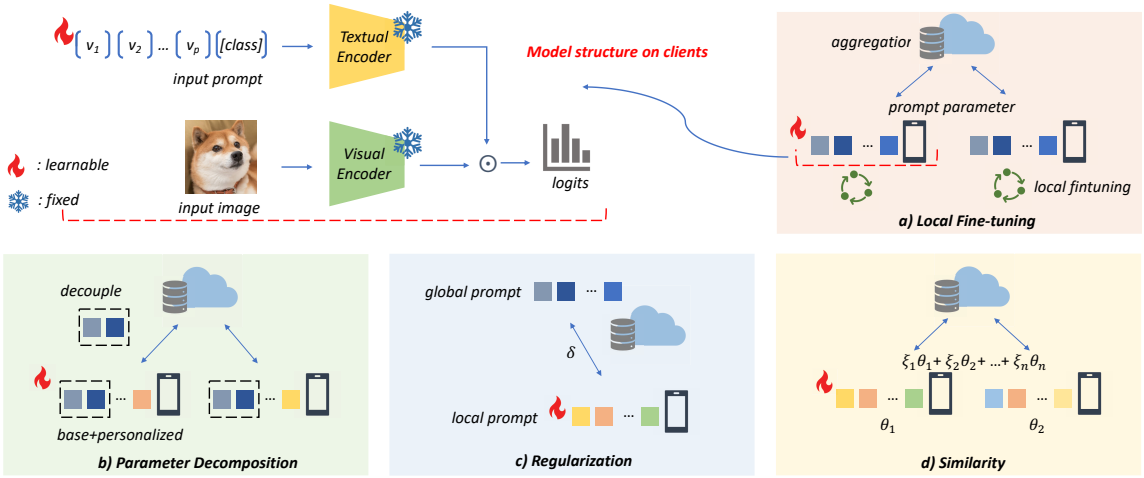

Figure 2: Illustration of baseline methods for personalized prompt for vision-language models in federated learning. The left above part shows the detailed structure of models on clients, which contains textual and visual encoders which are frozen and prompt which is learnable. To simplify the illustration of client model on for each method, we only utilize the learnable prompt to represent. Four personalized prompt learning techniques are introduced: a) local fine-tuning of prompt performed after obtaining global prompt, b) base vectors are aggregated while personalized vectors update locally, c) regularization is performed between global prompt and local prompt, d) clients relationship is leveraged for better personalization.

*p*FL methods apply a single global model and thus limit the customized level of the local model on the user side. Some researchers recommend training multiple global models on the server, where users are clustered into several groups according to their similarity and different global models are trained for each group [24, 51, 58]. We will discuss more details in section 3.1 how to directly apply existing *p*FL methods to prompt training.

2.3 Prompted Vision-Language Models

Vision-Language Models. The most trendy vision-language models like CLIP [44] and ALIGN [25] are neural networks pretrained on hundreds of millions of image and caption pairs. These models encode images and captions separately as vectors, enabling users with visual modality samples to retrieve, score, or classify samples from textual modalities. In other words, these models extend the knowledge of classification models to a wider range of things by leveraging semantic information in text.

Prompt Training. The pretrained vision-language models like CLIP consist of an image encoder and a text encoder to predict the pairing relationship between images and texts. Therefore, these models can be converted to an image classifier. As shown in figure 1, the users may convert all [class] to prompt such as "this is a photo of [class]" and predict the caption class the model estimates the best pairing with the given image. Previous research has involved prompt engineering [14, 33, 47, 57], in which human engineers or algorithms search for the best template for the classes. Prior work [17] proposes a federated prompt engineering framework and optimizes the prompt collaboratively via standard FL algorithm. The federated server aggregates only the parameter updates of the prompt across users, and keeps the CLIP backbone frozen locally. Therefore, using prompt training in FL incurs significant less computation and communication overhead than conventional FL. Nevertheless, the current prompt training in FL is essentially to train the user consensus (see figure 1). Different from previous research, our work aims to personalize prompt and further adapt the user consensus to the user's local features.

2.4 Attention Mechanism

Attention was first presented in the [2] and later emerged from [52] as an important component in the transformer architecture to decouple the long-range dependency of sequences, in the field of neural language processing. Nowadays, attention mechanism has developed vigorously in the field of computer vision by adaptively weighting features according to the importance of the input, and has shown its advantage in deep feature representation for visual tasks [16]. Other than the above training-based attention mechanisms which aims to select the important channels [23], branches [7] or spatial regions [6] inside the neural network, we inspire by [15] and propose a non-parametric attention module to capture the global data context for the local adaptation.

3 PROMPT PERSONALIZATION

In this section, we first investigate the under-explored methods of how to apply existing advances of *p*FL (as referred to in section 2.2) to prompt training in a straightforward manner. Unfortunately, these vanilla methods cannot capture the multimodality of vision-language models, thereby leading to insufficient personalization. We then present *p*FedPrompt, which can unleash and incorporate the knowledge retrieved from the multimodality.

3.1 *p*FL – Straightforward But Insufficient

Local Fine-tuning. *"FL training + local adaptation"* is usually regarded as a simple yet effective personalization paradigm by the FL

community [26, 36, 50]. After obtaining a collaboratively trained global model, each client adapts their local model through additional training with local datasets. Recently, the significance and effectiveness of this two-step paradigm have been brought up and emphasized by [8].

Similar in our case, when learning on heterogeneous data, all the clients train collaboratively by aggregating only the parameters of prompts but freezing the corresponding textual and visual backbone. After reaching a global user consensus prompt, each client fine-tunes the global prompt with its own few-shot data and obtains a personalized prompt. Personalized prompts are utilized with previously frozen backbones for further inference.

Parameter Decomposition. Parameter Decomposition is an architecture-based approach which aims to address the personalization problem by decoupling the personalized parameters from the global ones. [1] believes that deep learning model can be divided into two parts, *"base layers"* and *"personalization layers"*. Base layers are uploaded to join the formation of global model, while the personalized layers are kept locally by each client. [10] shares the same idea with different training procedures.

Inspired from [1, 10], here we intend to achieve personalization by viewing the learnable vector as *base + personalization vectors* and intend to decouple the personalized one from the base one. We presume that the former vectors act on common effects and intend to lead to a general performance, while the latter vectors which next to the class token emphasize on the specific performance related to the labels. Thus, during each iteration, we only transmit and aggregate the parameters of base vectors to the server and leave the personalized vectors on the local.

Regularization. Regularization is always employed in controlling the model expression ability during the training process [28]. In federated learning, regularization-based techniques are alleviated to address the client shift problem due to data heterogeneity by controlling the relationship between clients and global model. [31] introduced a proximal term on the local objective function to effectively limit the capability of local updates by restricting them to the current local model.

In our context, we aim to maintain the general instructive ability of prompt, but also allow them to approach the performance of their own local data distribution. Thus, we apply the method of [31] on learnable prompt by restricting the update of local prompt to not deviate too much from the current global prompt.

Similarity. Other than the above methods, similarity-based approaches are commonly used by leveraging the relationship and data distribution between clients. [24] propose a similarity-based mechanism to enforce that FL clients with similar data distributions collaborate intensely with each other, while clients with different data distributions have less impact on each other. Specifically, in each iteration, each client will maintain a cloud model which is the linear combination of the other clients, after obtaining the new model each client will perform local training with its private data.

Here we follow the idea of weight combination in prompt learning. Specifically, each client obtains a personalized prompt as a linear combination of the other local prompts, $u_c = \xi_{c,1}\theta_1 + \cdots + \xi_{c,m}\theta_m$ where $\sum_{m \in C} \xi_{c,m} = 1$. C is the set of local prompts, ξ is the coefficient that should be applied on θ, and θ is the weight parameter of other local prompts. For each round we obtain the above new prompt for each client and then update them locally, we perform this two-step interactively.

Limitations. The above methods are novel personalized prompt attempts for vision-language models when encountering data heterogeneity. However, problems may be encountered when transferring the setting from traditional model architecture to learnable prompt. First, ***parameters are few***. Compared with the backbone model, the amount of the learnable parameters is very small, which lets us think if the above personalized techniques suit well for models with small parameters? Second, ***shots are few***. Few-shot learning is employed in prompt learning instead of the traditional large amount of data, which might incur poor effect to the techniques that are data-driven. Third, ***two modalities exist***. Other than the single modality which only trains for images, vision-language models leverage the alignment of both textual and visual modality to enhance the performance of visual tasks with zero-shot or few-shot applications. However, the existing approaches only focus on the update of the prompt. i.e., the input of the text encoder, which raises a question that if it is enough to only adapt the single modality. Based on the above thinking, we employ several experiments in Sec. 4 and propose the following *p*FedPrompt approach.

3.2 *p*FedPrompt – Unleashing Multimodality

Motivation. As we survey the existing approaches on vision-language models in federated learning, several attempts have been explored to capture user consensus through prompt training [17, 54]. However, user characteristics, especially heterogeneous data distribution with real-world scenarios have been neglected so far. What's more, we observe that all the attempts, including the personalized techniques above, are conducted with prompts according to the existing paradigm, leaving both textual and visual encoder fixed.

Although achieving adequate performance by collaborative prompt training during the learning process, we notice that visual feature has not been leveraged for adaptation at all. This finding leads us to presume that there is still a room for improvement since the semantic gap incurred by local dataset is much larger between visual features than the one between text features.

Design. To capture the general features for all clients as well as adapting to the local personalization on each client, we present *p*FedPrompt, which contains two parts, Global User Consensus (GUC) and Local Feature Attention (LFA). GUC is captured through the textual space by global optimization of the learnable prompt. After obtaining the GUC, local personalization is realized on each client with the help of LFA through parameter-free attention to capture the additional personalized features and merge them to the final logits. The pipeline of *p*FedPrompt is shown in fig 3, which unleashes the modality in vision-language models. We will introduce GUC and LFA in detail as follows:

Global User Consensus (GUC). Each client is given a pre-trained vision-language model, with a fixed textual and visual encoder. Instead of the hand-craft prompt in[44], we introduce a set of p continuous vectors with d-dimension to form a learnable prompt that can be optimized through training. Here we use $p = 16$ and $d = 512$ as the word embedding in the text encoder.

Let $g(\cdot)$ and $h(\cdot)$ be the feature extraction function of the image and text encoder, k denotes the number of classes and each

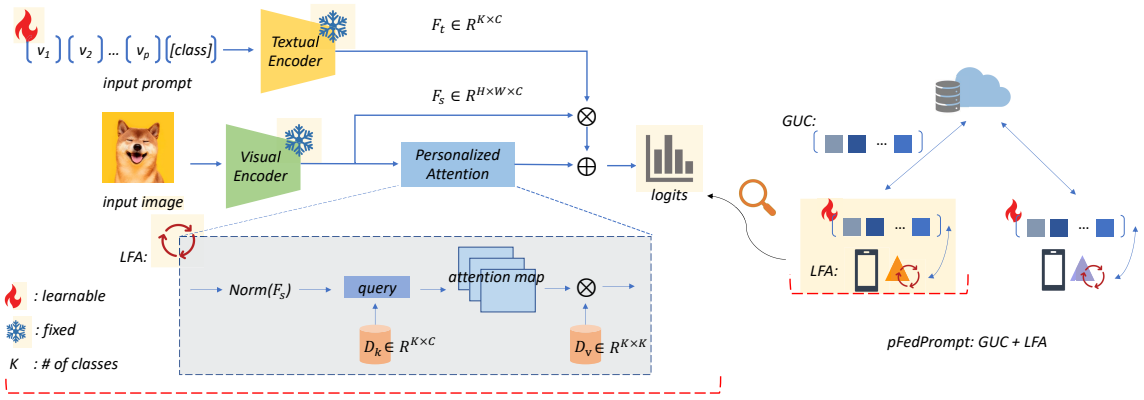

Figure 3: Illustration of *p*FedPrompt of personalized prompt for vision-language models in federated learning.The right part shows the workflow of *p*FedPrompt and the left part shows the detailed topography and pipeline of local model on the client. During the training process, only learnable prompt on each client is uploaded to capture the global user consensus. After obtaining the global prompt, textual encoder on each client is leveraged to generate the common textual features. On the other hand, each client maintains a non-parametric personalized attention module respectively, and combines with the visual encoder to generate the local personalized spacial visual features additionally. In this way, GUC and LFA work together to achieve superior performance for all clients under the heterogeneity setting.

$\mathbf{P}_i$ is derived from the prompt in the form of $[v_1][v_2]...[v_p][\text{class}]_i$, where $[\text{class}]_i$ is replaced by the word embedding vector of the corresponding i_{th} class label name. By forwarding the image-text pairs, each vision-language model will maximize the cosine similarity of the correct pairs and minimize the remaining incorrect pairs. The prediction probability on each client is computed as follows:

$$\mathrm{p}(\mathbf{y}=i|\mathbf{x}) = \frac{\exp\left(cos[g(\mathbf{x})|h(\mathbf{P}_i)]\right)}{\sum_{j=1}^{k}\exp\left(cos[g(\mathbf{x})|h(\mathbf{P}_j)]\right)}, \tag{1}$$

where P_i is the only part that can be updated during training. Each client optimizes local prompt for iterations between rounds.

After obtaining the latest updated prompt on each communication round, selected prompts will be uploaded to the server for aggregation. At each communication round $t+1$, $\mathbf{C}_k$ is the set of selected clients in joining in this round, and k is the client index. The aggregated prompt $\mathbf{P}_{t+1}$ for each round can be expressed as:

$$\mathbf{P}_{t+1} = \frac{1}{n_k}\sum_{k\in\mathbf{C}_k}\mathbf{P}_{t+1}^k. \tag{2}$$

Global prompt after aggregation will be downloaded to each client from the server. After several rounds between server and clients, global user consensus is captured collaboratively.

Local Feature Attention (LFA). After achieving global user consensus through the textual part, we leverage the visual counterpart to adapt personalization on each client, which precisely makes use of the modality advantages of vision-language model. For each input image, we obtain the intermediate spatial visual feature $F_s \in R^{H\times W\times C}$ extracted by visual encoder, and leverage the visual feature F_s to interact with a non-parametric attention module for the additional personalized logits.

We propose an external non-parametric attention module named *local personalization attention*, which computes the attention between the input visual feature F_s and an external memory unit M. M contains two parts, M_k and M_v, e.g., the key-value pairs, as our prior knowledge. To directly compensate the semantic gap in visual feature, we regard M as a memory of the local few-shots training data. For M_k, we first reshape the intermediate spatial visual feature F_s from $R^{H\times W\times C}$ into a 1D vector sequence $R^{HW\times C}$, and then use the normalized features as our keys in M_k. And as for M_v, we use the corresponding ground-truth label L_{train} after one-hot operation as our values in M_v. Given K class and N shots images per class, the dimension of M_k should be $NK\times C$. To maintain a stable and negligible overhead, we disentangle the buffer size with shots number and reshape the dimension of M_k to $K\times C$ on each client.

$$\mathrm{M}_k = Norm(Reshape(\mathrm{F}_{s(train)})), \tag{3}$$
$$\mathrm{M}_v = OneHot(\mathrm{L}_{train}). \tag{4}$$

After obtaining the external memory unit M, we calculate the pair-wise affinity between the input visual feature F_s and M_k to get the attention map A. To be concise, A is the additional attention map inferred from the affinity between the prior local knowledge and the current input features, which can be obtained through a query function. Here we use cosine similarity as our query function. Afterwards, the personalized feature can be generated as AM_v.

$$\mathrm{A} = cos(\mathrm{F}_s, \mathrm{M}_k), \tag{5}$$
$$\mathrm{F}_{personalized} = \mathrm{AM}_v. \tag{6}$$

Thus, the final logits can be expressed as the original logits obtained by the interaction between textual features $F_t \in R^{K\times C}$ and visual features $F_s \in R^{H\times W\times C}$, with the additional personalized features generated by the additional non-parametric attention:

$$\text{logits} = \mathrm{F}_s\mathrm{F}_t^t \cdot exp(t) + \alpha \cdot \mathrm{F}_{personalized}, \tag{7}$$

where α represents the weight for the additional personalized logits. If the local data generates a large semantic gap between the local and global prompt, the value should be large, otherwise, the value should be small. Specifically, in echo with the above consideration, the final logits is also composed of two parts: 1) the original logits represent the global user consensus (GUC) captured by the participation of collaborative trained prompt, and 2) the additional personalized logits realized by local feature attention (LFA) with the adaption of local data on each client.

4 EXPERIMENTS

In this section, we numerically evaluate our proposed method in the scenarios of heterogeneous data distribution and conduct comprehensive experiments.

4.1 Experimental Setup

Datasets. We select 6 representative image classification datasets used in CLIP [44] as our benchmark, which consists of various classification tasks. *General objects:* Caltech101 [13]. *Fine-grained Categories:* Flower102 [40], OxfordPets [41], Food101 [3]. *Action Recognition:* UCF101 [48]. *Texture Classification:* DTD [9].

Models. As for the local vision-language model, we use the same architecture with CLIP [44], which consists of an image encoder and a text encoder for feature extraction respectively. We use ResNet50 [20] as the backbone for image encoder and transformer [52] as the textual encoder. To quick-adjust and exploit the capacity of the pre-trained vision-language model, we follow the predecessor [60] to keep the prompt learnable and the two encoder freeze instead the complete zero-shot inference in CLIP.

Baselines. Since personalized techniques for the vision-language model is under-explored when encountering the heterogeneous scenarios [17]. We absorb the concept in traditional *p*FL techniques and adapt them to the scenarios of vision-language model as our baselines. We compare our method with the existing PromptFL and five adapted baseline methods, e.g., Local, PromptFL+Finetuning [8], PromptPer [1], PromptProx [31] and PromptAMP [24], as introduced in Sec 3.1.

Heterogeneity Simulation. Combined with previous works setting and practical situations of few-shot learning in our scenarios, we consider two pathological Non-IID settings in our experiments. In the pathological Non-IID setting, each client will be assigned only a specified number of labels, e.g. 5 random labels as shown in Tab. 2. Practical Non-IID setting with specific data distribution among clients is also a common setting in traditional personalized federated learning, however, since few-shots are employed here, this setting is not applicable in this scenario. Concerning different clients number n, participation rate r and Non-IID data distribution, we simulate the following two settings: 1) $n = 10$ clients with $r = 100\%$ participation, each client shares a completely disjoint random class with each other. 2) $n = 100$ clients with $r = 10\%$ participation, and $S = 5$ random classes are assigned to each client, thus repetition will appear in each class among clients.

Implementation Details. We implement all the methods in Pytorch and all experiments are conducted on GeForce RTX 3090 GPU. We use SGD optimizer with 0.001 learning rate with all methods except FedProx, which uses the corresponding modified optimizer with the stated best hyper-parameters reported in the preceding works. We set a local epoch $E = 5$ for both cases, while for the global communication round, we perform a $R = 10$ for $n = 10$ clients case and $R = 30$ for $n = 100$ clients case. For the fine-tuning baseline, we conduct an additional adaption epoch $AE = 10$ for $n = 10$ clients case and $AE = 30$ for $n = 100$ clients case.

For the setting of soft learnable prompt, we introduce a set of p continuous embeddings of dimension d in consist of the [prompt vectors]. d is the same as the dimension of word embeddings in the text encoder, i.e., 512 by default. p is a hyperparameter specifying the number of embeddings. Here we use $p = 16$ vectors as the best case shown in [17, 60]. We also follow the precedent to place the class token in the end of the of the prompt.

4.2 Performance Evaluation

Comparison with state-of-the-art. To show the effectiveness of our method, we compare our proposed *p*FedPrompt with corresponding adapted state-of-the-arts methods across six representative datasets. And as stated in the above section, both the selection of the baseline methods and datasets aims to guarantee the generosity and comprehensiveness of our evaluation. Due to the newly emergence of prompt training in FL, personalized problems and techniques in this scenario have not been considered and developed yet, which causes the lack of corresponding baselines for comparison. To make up for this deficiency, we utilize various state-of-the-art personalized techniques and adapt them in the form of prompt learning as our baseline. To enhance the persuasiveness of our proposed method, we select approaches from diversified categories for adaptation, e.g., Local Adaptation, Parameter Decoupling, Regularization-based and Similarity-based approaches. As for the datasets, we cover several categories including general objects, fine-grained objects, action recognition, and texture classification.

We report the performance of *p*FedPrompt against baselines for the two heterogeneous setting in Tab. 1 and 2. All the baselines are performed under their optimized setting. In almost all cases, *p*FedPrompt strongly outperforms the alternatives. Comparing the two tables, Tab. 2 shares more classes between clients since that classes are randomly shared, while classes are fully independent between clients in Tab. 1. As a result, the performance gap is wider in Tab. 1 than that of Tab. 2 as local data distribution become more extreme. When more classes are shared between clients, individual client possesses a greater possibility to benefit with each other. However, even though dissatisfaction appears in other approaches when heterogeneity increases, our proposed *p*FedPrompt remains robust performance across datasets. Within the Table, we observe that, for the other personalized approaches, the performance deteriorates strongly as datasets type change from general objects to fine-grained objects or other specific tasks. While for our method, the degradation process is comparatively slow, which on the other hand verifies the effectiveness and robustness of *p*FedPrompt.

Analysis of Number of Shots. Under the setting of few-shots learning, we also want to find out how is the number of shots in training data will affect the overall performance. To analyze the effect of the number of shots, we vary the shots in [1, 2, 4, 8, 16]. In

Table 1: Performance of *p*FedPrompt against adapted baselines on the pathological Non-IID Setting 1: The table reports the average test accuracy according to six diversified datasets. Six baselines are selected for comparison. Among them, PromptFL [17] is the novel paradigm for FL with vision-language model and the other four of them are adapted from the latest *p*FL researches. Here we use the extreme Non-IID setting, where 10 clients are simulated here with $r = 100\%$ participation rate and non-overlapping class on each client, which means that each class only appears once among clients. The best score of each group appears in bold. Compared with the adapted baseline methods, *p*FedPrompt outperforms other methods across datasets.

Dataset	Caltech101	Flowers102	OxfordPets	Food101	DTD	UCF101
(♯ Clients, Non-IID Setting)	(10 clients, non-overlapping)					
Local Training	87.37 ± 0.44	70.14± 0.76	83.21± 1.30	70.43±2.42	44.23±0.63	62.53± 0.09
PromptFL [17]	89.70 ± 1.99	72.80±1.14	90.79± 0.61	77.31±1.64	54.11± 0.22	67.87±0.74
PromptFL+FT (Adapted from [8])	89.70 ± 0.25	72.31± 0.91	91.23±0.50	77.16± 1.56	53.74±1.36	66.36±0.65
PromptPer (Adapted from [1])	86.72 ± 1.45	72.11± 1.35	89.50±1.62	71.29± 1.87	50.23±0.82	65.81±1.42
PromptProx (Adapted from [31])	89.41 ± 0.55	66.40± 0.29	89.24± 0.41	76.24± 1.94	44.26±1.11	63.27± 1.20
PromptAMP (Adapted from [24])	87.31± 1.60	69.10± 0.13	80.21±0.44	74.48±1.71	47.16±0.92	62.37±0.81
pFedPrompt (Ours)	**96.54 ± 1.31**	**86.46± 0.15**	**91.84± 0.41**	**92.26± 1.34**	**77.14±0.09**	**86.22±1.02**

Table 2: Performance of *p*FedPrompt against adapted baselines on the pathological Non-IID Setting 2: The table report the average test accuracy corresponding datasets and methods as stated in Tab. 1. Each baseline method is recorded with their optimal performance. 100 clients are simulated here and $r = 10\%$ of clients are selected to participate in each round. 5 random classes are selected on each client, which means that same classes may encounter overlapping on different clients. The best score of each group appears in bold. Compared with the adapted baseline methods, pFedPrompt not only reaches supreme performance on the extreme case with 10 clients setting in Tab. 1, but also outperforms other methods with more general case.

Dataset	Caltech101	Flowers102	OxfordPets	Food101	DTD	UCF101
(♯ Clients, Non-IID Setting)	(100 clients, 5 random class)					
Local Training	85.50± 0.32	72.8± 0.59	85.50 ±0.63	77.51±0.29	55.06±0.38	66.80±0.74
PromptFL [17]	82.92± 0.43	69.08±0.74	84.49±1.06	73.35±1.11	52.49±1.59	66.56±0.22
PromptFL+FT (Adapted from [8])	84.45± 0.29	71.04± 0.57	85.49± 0.49	74.61 ±0.82	56.20±0.51	68.40±0.21
PromptPer (Adapted from [1])	82.19±0.61	69.52± 0.23	83.66±0.85	73.72±0.84	53.34±1.44	66.78±0.53
PromptProx (Adapted from [31])	85.52± 0.42	67.63±1.10	85.76±0.80	73.36±2.12	46.23±0.18	62.31±0.11
PromptAMP (Adapted from [24])	88.30± 0.71	75.01±0.91	**87.50±0.51**	77.70±0.36	57.30±0.46	69.80±0.51
pFedPrompt (Ours)	**92.24 ± 0.31**	**85.72±0.18**	87.31±0.32	**90.11±0.60**	**73.44±0.96**	**85.97±0.42**

Fig. 4, we report six datasets and with each dataset, we record the performance of the five different shots setting. The horizontal axis shows the shots number and the vertical axis shows the average test accuracy. Heterogeneity simulation 2 is employed here.

We observe that in most cases, *p*FedPrompt already achieves promising performance when the number of training data is small, even within one shot. However, alternative approaches appear poor performance when shots are small. And as the number of shots increases, the corresponding performance will enhance gradually, but still can not beat the top-performance of *p*FedPrompt. Such phenomenon exactly verifies our concerns above that current personalization approaches only applies with the large amount training data. To be concise, *p*FedPrompt remains robustness against the variation of number of shots in comparing with the alternative methods, which exactly in accordance with the our purpose to propose a unique personalized technique which suits our particular few-shot scenarios.

Effects of Hyper-Parameter α. Apart from the above superiority of *p*FedPrompt, we also want to find out to what extent does the final performance benefit from the local feature attention. As mentioned above, α serves as the indicator to control the balance of the general user consensus(GUC) and local personalized feature(LFA). Larger value of α denotes to integrate more knowledge from local personalized feature and vice versa.

Within each number of shots, we vary the value of α from 0.0 to 5.0, and select the best α value that can produce the best performance. Tab. 3 reports the best α with the according number of shots, we can observe that as the number of shots increase, the demand for additional local feature information gradually decrease, which means that global prompt tuning can capture more user characteristics when given more data. And when the number of shots is small, the overall performance may benefit more from the local feature attention, which from the side proves that the local personalized attention module plays an important part in adjusting the personalization within the few-shot learning behavior.

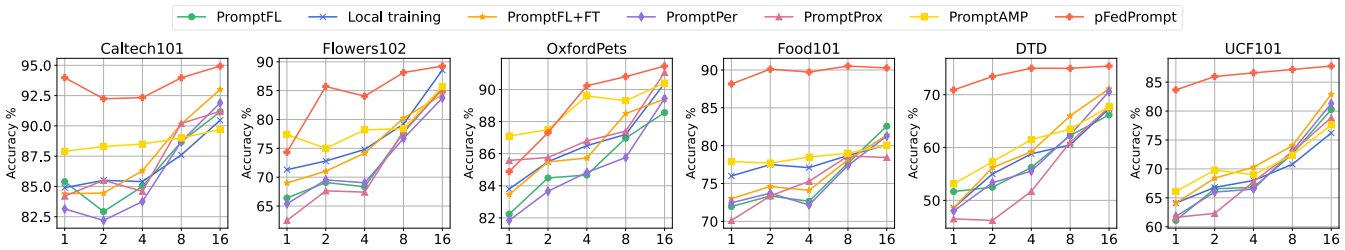

Figure 4: Peformance of different personalized approaches over six datasets. Average Local Test Accuracy is reported with different methods and number of shots. In each subplot, horizontal axis represents number of shots and vertical axis represents the corresponding test accuracy. We range the number of shots on each client from 1, 2, 4, 8 to 16. We observe that *p*FedPrompt strongly outperforms alternatives across datasets, as shown in the red line. Furthermore, when the number of shots decreases, the gap widens between *p*FedPrompt and other methods. Compared with the alternatives, *p*FedPrompt remains robust and outstanding performance against the variation of number of shots.

Effects of Buffer Size. To guarantee the additional memory overhead is limited and negligible, we resize the dimension of buffer size of M_k from $NK \times C$ to $K \times C$, which means that regardless of how many shots are utilized in training, the memory for the attention still remains for the same dimension of K, e.g, number of classes. To achieve this purpose, we aggregate and average the spatial visual feature of local training data to maintain that each class only exists one aggregated features as M_k. Tab. 3 shows the performance before and after reshape with different training shots of 1, 2, 4, 8 and 16. We can find out that even after reshape, the performance of *p*FedPrompt still outperforms other methods and the accuracy drop is negligible(around 1%) compared with the original one.

Computational Efficiency Analysis. We further analyse the computational overhead to ensure the efficiency of our method. We observe the required training epochs against the achieved test accuracy and compare it with the same local adaptation category method PromptFL+FT. However, unlike PromptFL+FT which needs extra local training for the adaptation, pFedPrompt adapts instantly during inference, which do not incur additional training overhead but still achieve extraordinary performance, as shown in Tab 4.

Table 3: Ablation results on effect of hyper-parameter α and buffer size. The best α for corresponding shots are reported. As shots decrease, model takes more advantage of the local personalized features for personalization. Compared with the performance before reshape of the buffer size, the decrease of average test accuracy after reshape is negligible.

♯ SHOTS	1	2	4	8	16
BEST α VALUE	3.77	2.55	2.06	1.08	0.59
ORIGINAL ACCURACY	93.98	92.24	92.33	93.97	94.94
RESHAPE ACCURACY	93.98	91.32	91.09	93.01	94.12

5 CONCLUSION

Large pre-trained vision-language models have shown great potential in federated learning. However, challenges when encountering real-world problems like data heterogeneity have not been well-addressed. To solve the problem, we first explore the existing personalized technique and adapt them to the prompt training manner. As few-shot learning and relatively limited learnable parameters are employed here, available methods achieve inadequate performance in this particular scenario. To bridge the gap, our research proposes *p*FedPrompt, which uniquely addresses the above challenge by leveraging both the textual and visual modality at the same time. *p*FedPrompt learns the global user consensus in the linguistic space with the collaboratively updating of prompt and adapts to user features in visual space with the local personalized attention module. Since the personalized attention module is non-parametric, the additional overhead is negligible. By incorporating the knowledge retrieved from multimodality, the challenge of user statistical heterogeneity is addressed. Our research provides a novel insight and direction in addressing the personalization problem in this scenario, which made an important step forward to the development and application of pre-trained vision-language models in federated learning.

Table 4: Efficiency Analysis for pFedPrompt. Training overhead with the corresponding accuracy of pFedPrompt against the two baselines are reported. The adaptation overhead is negligible while gain is considerable.

METHOD	Global (round)	Local (epoch)	Accuracy	Gain
PROMPTFL	30	0	82.92	-
PROMPTFL+FT	30	30	84.45	+ 1.53
PFEDPROMPT	30	0	92.24	+ 9.32

ACKNOWLEDGMENTS

This research was supported by fundings from the Key-Area Research and Development Program of Guangdong Province (No. 2021B0101400003), Hong Kong RGC Research Impact Fund (No. R5060-19), Areas of Excellence Scheme (No. AoE/E-601/22-R), General Research Fund (No. 152203/20E, 152244/21E, 152169/22E), and Shenzhen Science and Technology Innovation Commission (No. JCYJ20200109142008673).

REFERENCES

[1] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. 2019. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818* (2019).

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[3] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101–mining discriminative components with random forests. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[4] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer, 177–186.

[5] Duc Bui, Kshitiz Malik, Jack Goetz, Honglei Liu, Seungwhan Moon, Anuj Kumar, and Kang G Shin. 2019. Federated user representation learning. *arXiv preprint arXiv:1909.12535* (2019).

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 213–229.

[7] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. 2020. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11030–11039.

[8] G Cheng, K Chadha, and J Duchi. 2021. Fine-tuning in Federated Learning: A simple but tough-to-beat baseline. *arXiv* (2021).

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

[10] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. 2021. Exploiting shared representations for personalized federated learning. In *International Conference on Machine Learning*. PMLR, 2089–2099.

[11] Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Marc'aurelio Ranzato, Andrew Senior, Paul Tucker, Ke Yang, et al. 2012. Large scale distributed deep networks. *Advances in neural information processing systems* 25 (2012).

[12] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305* (2020).

[13] Li Fei-Fei, Rob Fergus, and Pietro Perona. 2004. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.

[14] Andreas Fürst, Elisabeth Rumetshofer, Viet Tran, Hubert Ramsauer, Fei Tang, Johannes Lehner, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto-Nemling, et al. 2021. Cloob: Modern hopfield networks with infoloob outperform clip. *arXiv preprint arXiv:2110.11316* (2021).

[15] Meng-Hao Guo, Zheng-Ning Liu, Tai-Jiang Mu, and Shi-Min Hu. 2022. Beyond self-attention: External attention using two linear layers for visual tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

[16] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. 2022. Attention mechanisms in computer vision: A survey. *Computational Visual Media* 8, 3 (2022), 331–368.

[17] Tao Guo, Song Guo, Junxiao Wang, and Wenchao Xu. 2022. PromptFL: Let Federated Participants Cooperatively Learn Prompts Instead of Models–Federated Learning in Age of Foundation Model. *arXiv preprint arXiv:2208.11625* (2022).

[18] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. 2020. Lower bounds and optimal algorithms for personalized federated learning. *Advances in Neural Information Processing Systems* 33 (2020), 2304–2315.

[19] Filip Hanzely and Peter Richtárik. 2020. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516* (2020).

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[21] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. 2019. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335* (2019).

[22] Chenghao Hu, Jingyan Jiang, and Zhi Wang. 2019. Decentralized federated learning: A segmented gossip approach. *arXiv preprint arXiv:1908.07782* (2019).

[23] Jie Hu, Li Shen, and Gang Sun. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7132–7141.

[24] Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. 2021. Personalized Cross-Silo Federated Learning on Non-IID Data.. In *AAAI*. 7865–7873.

[25] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*.

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

[27] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*. PMLR, 5132–5143.

[28] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.

[29] Anusha Lalitha, Shubhanshu Shekhar, Tara Javidi, and Farinaz Koushanfar. 2018. Fully decentralized federated learning. In *Proceedings of the NeurIPS Workshop on Bayesian Deep Learning*.

[30] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. 2022. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 965–978.

[31] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems* 2 (2020), 429–450.

[32] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2019. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations*.

[33] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. 2021. Supervision Exists Everywhere: A Data Efficient Contrastive Language-Image Pre-training Paradigm. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[34] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586* (2021).

[35] Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. 2020. Federated learning for open banking. In *Federated learning*. Springer, 240–254.

[36] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. 2020. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619* (2020).

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

[38] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 691–706.

[39] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 739–753.

[40] Maria-Elena Nilsback and Andrew Zisserman. 2008. Automated flower classification over a large number of classes. In *Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*.

[41] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. 2012. Cats and dogs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[42] Bjarne Pfitzner, Nico Steckhan, and Bert Arnrich. 2021. Federated learning in a medical context: a systematic literature review. *ACM Transactions on Internet Technology (TOIT)* 21, 2 (2021), 1–31.

[43] Liangqiong Qu, Yuyin Zhou, Paul Pu Liang, Yingda Xia, Feifei Wang, Ehsan Adeli, Li Fei-Fei, and Daniel Rubin. 2022. Rethinking architecture design for tackling data heterogeneity in federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10061–10071.

[44] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*. PMLR, 8748–8763.

[45] Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. 2012. Making gradient descent optimal for strongly convex stochastic optimization. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*. 1571–1578.

[46] Abhijit Guha Roy, Shayan Siddiqui, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. 2019. Braintorrent: A peer-to-peer environment for decentralized federated learning. *arXiv preprint arXiv:1905.06731* (2019).

[47] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. 2022. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[48] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).
[49] Canh T Dinh, Nguyen Tran, and Josh Nguyen. 2020. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems* 33 (2020), 21394–21405.
[50] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. 2022. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems* (2022).
[51] Xueyang Tang, Song Guo, and Jingcai Guo. 2022. Personalized federated learning with contextualized generalization. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*. 2241–2247.
[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
[53] Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. 2019. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252* (2019).
[54] Chuhan Wu, Fangzhao Wu, Tao Qi, Yongfeng Huang, and Xing Xie. 2022. Privacy-preserving, Efficient, and Effective Machine Learning. (2022).
[55] Jinze Wu, Qi Liu, Zhenya Huang, Yuting Ning, Hao Wang, Enhong Chen, Jinfeng Yi, and Bowen Zhou. 2021. Hierarchical personalized federated learning for user modeling. In *Proceedings of the Web Conference 2021*. 957–968.
[56] Liu Yang, Ben Tan, Vincent W Zheng, Kai Chen, and Qiang Yang. 2020. Federated recommendation systems. In *Federated Learning*. Springer, 225–239.
[57] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. 2021. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432* (2021).
[58] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M Alvarez. 2020. Personalized Federated Learning with First Order Model Optimization. In *International Conference on Learning Representations*.
[59] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582* (2018).
[60] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. 2022. Learning to prompt for vision-language models. *International Journal of Computer Vision* 130, 9 (2022), 2337–2348.
[61] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Advances in neural information processing systems* 32 (2019).

A APPENDIX

A.1 Ablation study on *alpha*

We show the best α for all six datasets. In most cases, the value of best α decreases as the number of shots increases, as shown in Tab. 5. Such behavior further validate that fewer shots may benefit more from local adaptaion, since that prompt might have difficulties in capturing user features from too little training data. We also observe that as for oxfordpets, since the GUC have already captured enough information from prompt learning, the best value of α is relatively small.

Table 5: Ablation results on effect of hyper-parameter α across six datasets.

Dataset	♯ shots	1	2	4	8	16
Caltech101	best α	3.77	2.55	2.06	1.08	0.59
	accuracy	93.98	92.24	92.33	93.97	94.94
Flowers102	best α	0.1	2.06	1.08	0.35	0.35
	accuracy	74.32	85.72	84.04	88.13	89.26
OxfordPets	best α	0.1	0.59	0.59	0.1	0.1
	accuracy	84.88	87.31	90.23	90.8	91.46
Food101	best α	4.75	4.51	2.06	2.06	3.53
	accuracy	88.15	90.11	89.72	90.52	90.27
DTD	best α	4.51	3.29	1.08	1.33	0.59
	accuracy	70.89	73.44	75.01	74.99	75.42
UCF101	best α	3.77	2.55	2.06	1.08	4.51
	accuracy	83.65	85.97	86.6	87.19	87.78

A.2 Ablation study on buffer size

We show the entire results of average test accuracy over six data, as shown in Tab. 6. The accuracy after reshaping the dimension of the buffer size only drops a little, around 1 %, which demonstrates the robustness and efficiency of our method with limited additional overhead.

Table 6: Ablation results on effect of buffer size across six datasets.

Dataset	♯ shots	1	2	4	8	16
Caltech101	original acc	93.98	92.24	92.33	93.97	94.94
	reshape acc	93.98	91.30	91.3	93.01	94.02
Flowers102	original acc	74.32	85.72	84.04	88.13	89.26
	reshape acc	74.32	84.6	83.01	87.4	88.2
OxfordPets	original acc	84.88	87.31	90.23	90.81	91.46
	reshape acc	84.88	86.40	89.5	89.16	90.41
Food101	original acc	88.15	90.11	89.72	90.52	90.27
	reshape acc	88.15	88.64	88.03	89.26	89.97
DTD	original acc	70.89	73.44	75.01	74.99	75.42
	reshape acc	70.89	73.31	73.78	74.37	75.8
UCF101	original acc	83.65	85.97	86.60	87.19	87.78
	reshape acc	83.65	84.70	85.45	86.44	86.97

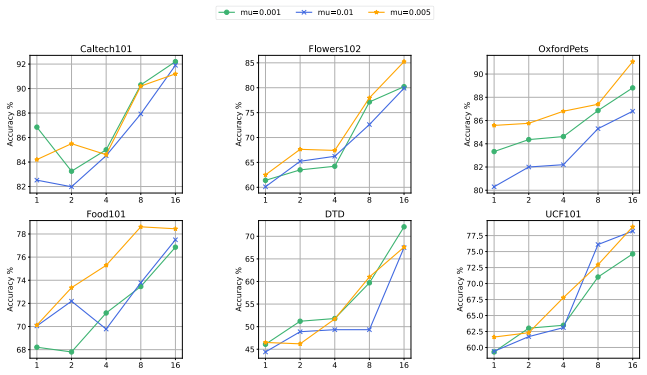

Figure 5: Effect of μ in PromptProx. μ is range from 0.001, 0.005 and 0.01.

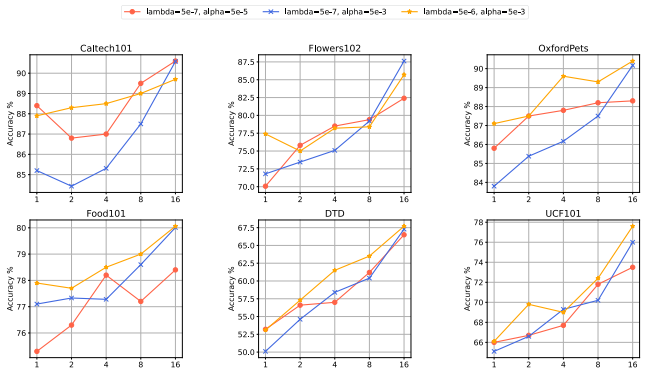

Figure 6: Effect of λ and α in PromptAMP. λ range from $5e-6$ to $5e-7$, α range from $5e-3$ to $5e-5$.

A.3 Detailed settings of PromptProx and PromptAMP

We show the detailed setting selection of hyper-parameters for the two adapted personalized methods, PromptProx and PromptAMP. For each method, we apply multiple settings and record three settings of the corresponding hyper-parameters here. We select the best performance as the baseline setting in the above main text. As shown in Fig. 5, the best setting for PromptProx is $\mu = 0.005$, and for PromptAMP, the best setting is $\lambda = 5e-6$ and $\alpha = 5e-3$.