Federated Adaptive Prompt Tuning for Multi-Domain Collaborative Learning

Shangchao Su, Mingzhao Yang, Bin Li*, Xiangyang Xue*

Shanghai Key Laboratory of Intelligent Information Processing
School of Computer Science, Fudan University
{scsu20, mzyang20, libin, xyxue}@fudan.edu.cn

Abstract

Federated learning (FL) enables multiple clients to collaboratively train a global model without disclosing their data. Previous researches often require training the complete model parameters. However, the emergence of powerful pre-trained models makes it possible to achieve higher performance with fewer learnable parameters in FL. In this paper, we propose a federated adaptive prompt tuning algorithm, FedAPT, for multi-domain collaborative image classification with powerful foundation models, like CLIP. Compared with direct federated prompt tuning, our core idea is to adaptively unlock specific domain knowledge for each test sample in order to provide them with personalized prompts. To implement this idea, we design an adaptive prompt tuning module, which consists of a meta prompt, an adaptive network, and some keys. The server randomly generates a set of keys and assigns a unique key to each client. Then all clients cooperatively train the global adaptive network and meta prompt with the local datasets and the frozen keys. Ultimately, the global aggregation model can assign a personalized prompt to CLIP based on the domain features of each test sample. We perform extensive experiments on two multi-domain image classification datasets across two different settings — supervised and unsupervised. The results show that FedAPT can achieve better performance with less than 10% of the number of parameters of the fully trained model, and the global model can perform well in diverse client domains simultaneously.

Introduction

As privacy protection gains increasing attention, federated learning (FL) (McMahan et al. 2017), a special machine learning paradigm, becomes more popular. FL has been applied to mobile phone album classification, automatic driving, and medical image analysis. FL enables multiple parties to cooperatively train a global model without sharing their training data. In each round of communication, the server sends the global model to the clients, and each client updates the global model with the private dataset to obtain the local model. The server collects the local models for aggregation, and obtains the global model for the next communication round. In computer vision, FL has been applied to classification (Hsu, Qi, and Brown 2020; Li, He, and Song 2021; Yang et al. 2023),

detection (Liu et al. 2020; Su et al. 2023), ReID (Zhuang, Wen, and Zhang 2021; Zhuang et al. 2020), etc.

Currently used FL techniques (McMahan et al. 2017; Li et al. 2020; Karimireddy et al. 2020) require all model parameters to be updated from the clients and shared on the server. As a result, collaborative learning incurs substantial client training costs, as well as communication costs for both clients and the server. The non-IID problem of various client data also poses significant difficulties for the federated learning algorithm. The performance of the global aggregation model will be impacted when the client datasets come from diverse data domains. Thus, training a more effective global model with fewer communication costs by adjusting only a few parameters remains an open research question.

The emergence of powerful vision-language pre-trained models, such as CLIP (Radford et al. 2021), opens up new avenues for solving this problem. CLIP formalizes object recognition into image-text matching, rather than an image-label mapping problem. Text encoder and image encoder are two of the Transformer (Vaswani et al. 2017) branches that are included in CLIP. About 400 million web-crawled image-text pairs are used for contrastive learning to establish semantic associations between images and texts. Pre-trained CLIP performs admirably in the zero-shot image classification task and demonstrates high transferability in a number of downstream applications. With the powerful representation ability of CLIP, it becomes possible to fine-tune only a few parameters for each client in the federated learning scenarios.

Inspired by the promising capabilities of CLIP, in this research, we propose a federated prompt tuning algorithm, FedAPT, for multi-domain collaborative image classification in cross-silo federated learning. Consider that there are multiple participants in diverse data domains; the ultimate objective is to develop a global classification model through federated prompt tuning that can perform well in all data domains. We employ a prompt tuning based strategy to provide learnable prompts for the CLIP. To make the global prompt adapt to all domains, we propose an adaptive method to set customized prompts for various test images in order to address the cross-domain challenge.

The underlying intuition is that images contain domain-specific information. By incorporating this domain-specific information into prompts, we can better guide the pre-trained CLIP model to activate relevant knowledge to that domain,

then the classification ability of the corresponding domain can be improved. The adaptive prompt tuning module includes a meta prompt, an adaptive network, and some frozen keys. Before federated learning, the server randomly assigns each client a frozen key. In local training, each client trains the adaptive network and the prompt using local data and the frozen key. Considering that the client may not have labels in the real scene, we also design an unsupervised training method for FedAPT. After local training, all updated parameters and prompts are sent to the server for aggregation. During inference, we leverage an adaptive network to select a specific key for each test image, so as to generate a specific prompt from the meta prompt.

We conduct extensive experiments on two multi-domain image classification datasets, Office-Caltech10 and Domain-Net, across two different settings: supervised and unsupervised. The results show that FedAPT can build a more powerful global model than fully-trained ResNet50 or ViT with less than 10% of the number of parameters, and demonstrates its ability to perform well in a variety of scenarios. All experiments show that our scheme has significantly improved performance compared with the competitors. In general, our contributions are as follows:

- We utilize CLIP for the first time in federated cross-domain image classification, across two distinct scenarios — supervised and unsupervised. Our results demonstrate the significant potential of CLIP in federated learning.
- We propose a federated adaptive prompt tuning framework, FedAPT. With the leverage of keys, FedAPT can provide a personalized prompt for each test image without adding any learnable parameters.
- Our experiments conducted across different settings on both Office-Caltech10 and DomainNet demonstrate that FedAPT outperforms both the fully-trained models and the existing federated prompt tuning methods.

Related Work

Federated Learning. The idea of aggregating client data distribution information to the cloud can be traced back to SCM (Li et al. 2007). Recently, FedAvg (McMahan et al. 2017) extends the aggregation idea to neural networks and proposes the federated averaging algorithm. Despite FedAvg's strong performance under the assumption of IID, as the non-IID degree increases (due to the diverse distributions of client datasets), the aggregation model's performance continues to deteriorate, making the non-IID problem a key area of research. Some efforts attempt to enhance the performance of the global aggregation model by improving the optimization algorithm (Li et al. 2020; Karimireddy et al. 2020; Reddi et al. 2020; Wang et al. 2020b; Su, Li, and Xue 2023) or designing better aggregation methods (Wang et al. 2020a; Singh and Jaggi 2020), while others (Dinh, Tran, and Nguyen 2020; Fallah, Mokhtari, and Ozdaglar 2020; Hanzely et al. 2020; Li et al. 2021; Luo et al. 2022) turn to assigning different models to each client, i.e., personalized FL, such as using local batch normalization (Li et al. 2021), decoupling the model into personalized and global parts (Luo et al. 2022). The proposed FedAPT belongs to the former, i.e., to enhance

the global model.

Some works (Chen et al. 2022; Nguyen et al. 2023) study the impact of pre-training on FL. Unlike the fine-tuning methods, they need to train the complete model. FedPCL (Tan et al. 2022) studies a setting where each client has K fixed backbones, they fine-tune the projection head by sharing the prototype features of user data. Unlike FedPCL, we do not need to maintain multiple backbones locally, and we do not upload any prototype features. All in all, different from the existing works, we pay more attention to how to effectively utilize the knowledge in the powerful pre-trained models.

Recently, FedIns (Feng et al. 2023) shares a similar spirit with our work, both enabling Instance-adaptive Inference. The difference lies in the implementation: FedIns achieves adaptive inference for ViT through learnable Keys and SSF pool, while our approach achieves adaptive inference for CLIP through randomly initialized keys and a learnable query network coupled with prompts.

CLIP and Prompt Tuning. The contrastive vision-language pre-trained model CLIP (Radford et al. 2021) transforms image recognition into an image-text matching problem, freeing the object recognition task from human-annotated data, such that a huge amount (400M) of noisy image-text pairs can be used for training. The pre-trained model can be transferred to multiple datasets. Google releases ALIGN (Jia et al. 2021) to expand CLIP's training data to 1.8B, which achieves better results. Thanks to its powerful representation capability, CLIP has also been successfully applied to a variety of downstream visual tasks, such as RegionCLIP (Zhong et al. 2022) and DetCLIP (Yao et al. 2022) for object detection, MedCLIP (Wang et al. 2022) for medical image analysis, CCR-CLIP (Yu et al. 2023) for text recognition.

Prompt (Liu et al. 2021a) adds additional words or sentences to the input text of the pre-trained language model, to make the pre-trained model better handle downstream tasks. The succeeding works (Li and Liang 2021; Liu et al. 2021b) treat the prompts as continuous vectors in order to learn new knowledge in the process of fine-tuning, which is called Prompt Tuning. In CLIP's follow-up work, other researchers (Zhou et al. 2022b,a) tried to improve prompt tuning in CLIP. The work most relevant to ours is CoCoOp (Zhou et al. 2022a), which trains prompts conditioned on visual features. However, it is designed for centralized training and heavily relies on high GPU computational power, making it hard to apply to clients in federated learning. In contrast, in this paper, the training of conditions and prompts are decoupled, significantly accelerating the training process.

CLIP in Federated Learning. In the field of federated image classification, work related to CLIP is rare. PromptFL (Guo et al. 2022) extends CoOp (Zhou et al. 2022b) to the federated learning scenarios, averaging multiple prompts on the server. PromptFL shows the potential application of prompt tuning for CLIP in FL, however, it lacks optimizations tailored for multi-domain scenarios. In comparison, FedAPT proposes an adaptive prompt tuning method for multi-domain image classification. FedCLIP (Lu et al. 2023) adds an adapter consisting of two fully connected

layers to the end of the image encoder. FedTPG (Qiu et al. 2023) proposes a prompt generator with better generalization performance on unknown categories. However, they do not generate personalized prompts for the images based on the domain information.

Preliminaries

Notations. Consider a federated learning scenario, such as autonomous driving, where multiple cars are deployed in K regions (domains). Data from the same region exhibits similar styles, while data from different regions possesses distinct styles. Each region has N_{dom} clients. We assume that each client has sufficient computing power to train the prompts. In supervised setting, $\mathcal{D}_n = \{\mathbf{x}_i, y_i\}_{i=1}^{m_n}, n = 1, \cdots, N$ denote the local datasets, m_n is the number of images in the n-th client. In unsupervised setting, $\mathcal{D}_n = \{\mathbf{x}_i\}_{i=1}^{m_n}, n = 1, \cdots, N$. Images in diverse domains have different styles. Note that there may exist domain and category divergences across different clients at the same time, the divergence of category refers to the different label distribution of different clients. Let $\mathcal{I}(\cdot)$ and $\mathcal{T}(\cdot)$ denote the image encoder and the text encoder. To reduce the calculation and communication costs, we freeze the parameters of the two encoders and perform the training task by fine-tuning the prompts.

Prompt Tuning. Let $\mathbf{t}^c$ be the word embedding of the cth-class, the class text could have the form 'a picture of a dog'. Then the prediction of CLIP for image classification is:

$$p(c \mid \mathbf{x}) = \frac{\exp\left(\cos\left(\mathcal{T}(\mathbf{t}^c), \mathcal{I}(\mathbf{x})\right) / \tau\right)}{\sum_{i=1}^{C} \exp\left(\cos\left(\mathcal{T}(\mathbf{t}^i), \mathcal{I}(\mathbf{x})\right) / \tau\right)} \quad (1)$$

where C is the number of classes, τ is a temperature parameter learned by CLIP.

To fine-tune the model, one method is to manually set prompts, such as replacing 'a picture of a dog' with 'a picture of a cute dog'. However, in this way, because the change of words will directly affect the performance of the model, we need to find the most appropriate words. Another method is prompt tuning (Li and Liang 2021; Liu et al. 2021b; Zhou et al. 2022b). Let $\mathbf{p}^c = [\mathbf{v}_1^c, \mathbf{v}_2^c, \ldots, \mathbf{v}_M^c]$ be the prompt. The input of the c-th class for the text encoder is $(\mathbf{p}^c; \mathbf{t}^c)$, where $\{\mathbf{v}_i^c\}_{i=1}^M$ are learnable prompts for the c-th class. $\mathbf{v}_i^c$ has the same dimension as word embedding $\mathbf{t}^c$, M is the number of learnable prompts.

Federated Learning with Prompt. The concept of FL is introduced in FedAvg (McMahan et al. 2017) to collaboratively train a global model that works for each participant. A straightforward way to apply CLIP's prompt tuning to the FL field is that each client trains its own prompt and uploads the prompt to the server. The server averages the prompts of all clients and distributes them to the clients. Let $\mathbf{p}_n^c$ denotes the prompt for the c-th class in the n-th client. For the sake of brevity, we omit c and use $\mathbf{p}_n$ to denote the prompt in the n-th client. Let $\mathbf{p}_g$ denotes the prompt of the global model. Then the global objective is:

$$\mathbf{p}_g = \arg\min_{\mathbf{p} \in \mathbb{R}^d} \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{\mathbf{x}, y \sim \mathcal{D}_n}[\ell(\mathcal{T}(\mathbf{p}); \mathcal{I}(\mathbf{x}))] \quad (2)$$

where ℓ is the loss of the n-th client. Although this approach can handle scenarios where multiple clients are IID, it is still difficult for a single global prompt to process clients from diverse data domains.

Method

The overview of the proposed FedAPT is in Figure 1. We focus on the problem setting of one-model-fits-all, that is, learning a powerful global model to adapt to the data distribution across all domains. With the parameters of the CLIP frozen, we introduce an additional adaptive prompt tuning (APT) module for the global model. APT accepts image features as input and outputs a specific prompt for each sample. The learnable parts of APT are the prompt and the adaptive network, which are obtained through the federated training of multiple clients.

Adaptive Prompt Tuning

The input of the text encoder in PromptFL always uses a fixed prompt for all samples: $\mathbf{p}_g \in \mathbb{R}^{C \times s \times d}$, which is the learned prompt by federated averaging, C is the number of classes, s is the length of prompt, d is the embedding dimension. However, under the setting in this paper, each client may come from a specific domain, and there are obvious domain divergences among the images. A fixed prompt is no longer suitable for images in all domains. It is necessary to design a new form. An intuitive way is to introduce personalized prompts into the input of the text encoder for each domain:

$$\mathbf{p}_g + \mathbf{p}_{p_k} \quad (3)$$

where $\mathbf{p}_{p_k}$ encodes the personalized domain knowledge of the k-th domain. Unfortunately, under the goal of one-model-fits-all, we need a unified global model, rather than adding personalized parameters $\mathbf{p}_{p_k}$ for each client. Then a compromise improvement for Eq. 3 is to keep all $\mathbf{p}_{p_k}$ in the global model, then filter prompts based on the image features:

$$\mathbf{p}_g + f([\mathbf{p}_{p_1}, \cdots, \mathbf{p}_{p_K}], \mathcal{Q}(\mathcal{I}(\mathbf{x}))) \quad (4)$$

where $\mathbf{p}_{p_k}$ is the personalized prompt for the k-th domain, $f(\cdot)$ is a filter function, $\mathcal{Q}$ is an adaptive network that is used to provide auxiliary information for filtering. However, this operation needs to retain a number of $\mathbf{p}_{p_k}$, thus introducing additional parameters in the global model.

Finally, we adopt an adaptive prompt tuning approach. Let $\mathcal{P}$ denotes the output of the APT module, the input for this module is the image features $\mathcal{I}(\mathbf{x})$. We use the key vector $\mathbf{e}_k \in \mathbb{R}^{s \times d}$ to encode personalized domain knowledge into $\mathbf{p}_g$, and decode it when necessary. $\mathcal{P}$ in the global model is:

$$\mathcal{P}(\mathcal{I}(\mathbf{x})) = \mathbf{p}_g + \mathbf{p}_g \odot \sum_{k=1}^{K} q_k \mathbf{e}_k' \quad (5)$$

where $q_k = \mathcal{Q}_g(\mathcal{I}(\mathbf{x}))_k$, $\sum_k q_k = 1$, $\odot$ represents the element-wise multiplication, $\mathbf{e}_k' \in \mathbb{R}^{C \times s \times d}$ is copied from $\mathbf{e}_k$ for C times to match the dimension of $\mathbf{p}_g$, and $\mathcal{Q}_g$ is the adaptive network with parameters ϕ, which can give a one-hot or soft-membership vector of K dimensions. In the n-th
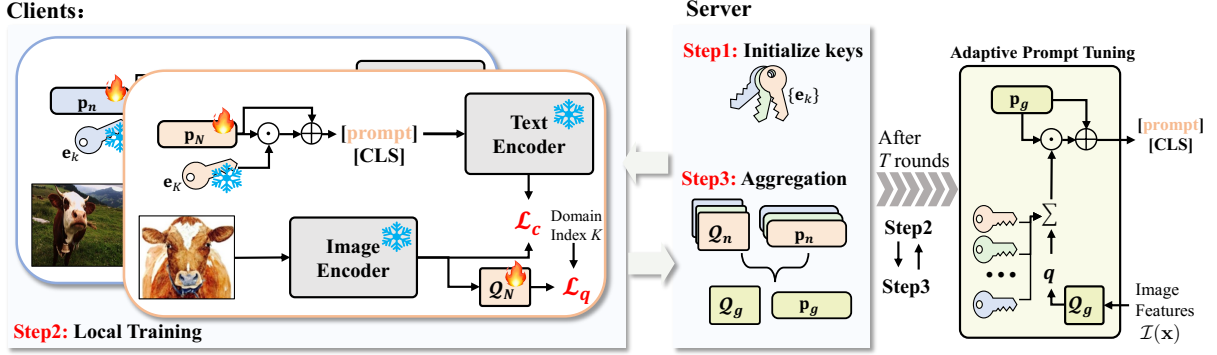
Figure 1: Overview of FedAPT. The federated training process is as follows: **Step1**: The server first randomly generates a set of random keys and assigns each client a frozen key. **Step2**: In local training, the key is used to give a constraint to the prompt. In addition, each client trains the adaptive network $\mathcal{Q}_n$ with local data. **Step3**: After local training, the server averages the prompts and adaptive networks learned by the clients and sends the results back to the clients. Repeat Steps 2~3 for multiple rounds, we can obtain $\mathbf{p}_g$ and $\mathcal{Q}_g$ to establish our target global model which can generate a personalized prompt according to image features.

client from the k-th domain, the prompt for the text encoder in the training process is:

$$\mathbf{p}_n + \mathbf{p}_n \odot \mathbf{e}'_k \qquad (6)$$

where $\mathbf{p}_n$ is initialized by $\mathbf{p}_g$, $\mathbf{e}'_k$ is frozen. Through the design of Eqs.5 and 6, we do not need to introduce additional learnable parameters $\mathbf{p}_{p_k}$, but use the unified $\mathbf{p}_g$, which is named "meta prompt". The objective for the meta prompt is:

$$\mathbf{p}_g = \arg\min_{\mathbf{p}} \frac{1}{N} \sum_{n=1}^{N} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}_n} [\ell(\mathcal{T}(\mathbf{p} + \mathbf{p} \odot \mathbf{e}'_n); \mathcal{I}(\mathbf{x}))]$$

Federated Training

To enable multiple clients to train APT cooperatively, there are three steps. At the beginning of federated training, the server randomly initializes the keys and sends them to different clients (**Step 1**).

Step2: Local Training. To evaluate the capabilities of CLIP and FedAPT in various FL settings, we consider two different client settings:

In the supervised setting, each client updates the global $\mathbf{p}_g$ and $\mathcal{Q}_g$ with the local data respectively to obtain $\mathbf{p}_n$ and $\mathcal{Q}_n$. The prediction of the n-th local model is:

$$p(c \mid \mathbf{x}) = \frac{e^{\cos(\mathcal{T}(\mathbf{p}_n^c + \mathbf{p}_n^c \odot \mathbf{e}_k), \mathcal{I}(\mathbf{x}))/\tau}}{\sum_{i=1}^{C} e^{\cos(\mathcal{T}(\mathbf{p}_n^i + \mathbf{p}_n^i \odot \mathbf{e}_k), \mathcal{I}(\mathbf{x}))/\tau}}$$

where k is the domain index of the client. Then the cross entropy loss for image classification is:

$$\mathcal{L}_c = \frac{1}{|\mathcal{D}_n|} \sum_{\mathbf{x}_i, y_i \sim \mathcal{D}_n} -\log p(y_i \mid \mathbf{x}_i, \mathbf{p}_n) \qquad (7)$$

In an unsupervised setting, we design a two-stage training approach. In the first stage, we augment the unlabeled data (same augment strategy as SimCLR (Chen et al. 2020)) and utilize pseudo-labels with high confidence from the original data as supervised information to perform self-training on

the augmented data. The training loss is:

$$\mathcal{L}_{pseudo} = \frac{1}{|\mathcal{D}_n|} \sum_{\mathbf{x}_i \sim \mathcal{D}_n} -\log p(\hat{y}_i \mid \hat{\mathbf{x}}_i, \mathbf{p}_n) \qquad (8)$$

where $\hat{y}_i$ is the pseudo one-hot label of $\mathbf{x}_i$ predicted by the original CLIP, $\hat{\mathbf{x}}_i$ is the augmented version of $\mathbf{x}_i$. After the first stage, we can get a coarse prompt.

In the second stage, use $\mathbf{z}_i$ to represent the prediction score vector of $\mathbf{x}_i$, we hope to compare the outputs among different samples. For inter-sample comparison, motivated by AaDLoss (Yang et al. 2022) in domain adaptation, for a certain sample, the logits of K-nearest neighbor samples with similar image features should be as similar as possible, otherwise, there should be differences. Then the loss is:

$$\mathcal{L}_{inter} = \frac{1}{|\mathcal{D}_n|} \sum_{\mathbf{x}_i \sim \mathcal{D}_n} \left(-\sum_{j \in \mathcal{C}_i} \mathbf{z}_i^\top \mathbf{z}_j + \lambda \sum_{m \in \mathcal{B}_i} \mathbf{z}_i^\top \mathbf{z}_m \right) \quad (9)$$

where $\mathcal{C}_i$ contains K-nearest neighbors, $\mathcal{B}_i$ contains the rest samples. A feature bank and a score bank are used to efficiently implement this loss.

As for intra-sample comparison, we need that the prediction of a sample is close to that of its augmented counterpart:

$$\mathcal{L}_{intra} = \frac{1}{|\mathcal{D}_n|} \sum_{\mathbf{x}_i \sim \mathcal{D}_n} KL(\hat{\mathbf{z}}_i \| \mathbf{z}_i) \qquad (10)$$

where $\hat{\mathbf{z}}_i$ is the predict score of the augment data, $\mathbf{z}_i$ is extracted from the score bank. Then the overall loss in the second stage is $\mathcal{L}_c = \mathcal{L}_{inter} + \mathcal{L}_{intra}$.

The adaptive network $\mathcal{Q}_n(\cdot)$ is a fully-connection layer with parameters ϕ_n, which tries to classify domain information from the features of images. Each client in the k-th domain trains its own adaptive network with the following cross-entropy loss:

$$\mathcal{L}_q = \frac{1}{|\mathcal{D}_n|} \sum_{\boldsymbol{x} \in \mathcal{D}_n} -\log p(k \mid \mathbf{x}_i, \phi_n) \qquad (11)$$

The whole loss function for local training is $\mathcal{L}_c + \mathcal{L}_q$.
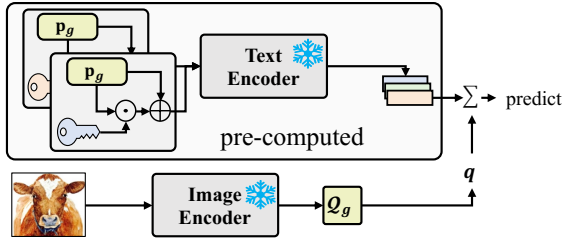
Figure 2: Improve inference efficiency.

(a) Office-Caltech10

(b) DomainNet

Figure 3: (a) Office-Caltech10, which includes four domains. (b) DomainNet, which includes six domains.

Note that training $\mathcal{Q}_g(\cdot)$ is a special FL problem, because each client can only access the data of its own domain. In this paper, we find that in the multi-domain scenario, we can obtain good enough $\mathcal{Q}_g(\cdot)$ by directly using cross-entropy for local training and federated averaging for aggregation. To further improve the effect of $\mathcal{Q}_g(\cdot)$, it is necessary to use special training methods, such as FedAws (Yu et al. 2020). In addition, directly uploading the data prototype (Tan et al. 2022; Peng et al. 2019b), can also train the $\mathcal{Q}_g(\cdot)$ in the server, but in order to prevent privacy disclosure, we do not upload any data features or prototypes.

Step3: Aggregation. After the local training step, the n-th client uploads $\mathbf{p}_n$ and ϕ_n to the server. Due to $\phi_n \in \mathbb{R}^{d_{img} \times K}$, d_{img} is the dimension of each image feature, which is 512 in CLIP, the additional traffic brought by transmitting ϕ_n is extremely small compared with prompts. The server averages the parameters to obtain the aggregation model parameters:

$$\mathbf{p}_g = \frac{1}{N} \sum_{n=1}^{N} \mathbf{p}_n, \quad \phi_g = \frac{1}{N} \sum_{n=1}^{N} \phi_n \qquad (12)$$

Repeat Step 2 and Step 3 for T rounds, then $\mathbf{p}_g$ and $\mathcal{Q}_g(\cdot)$ with parameters ϕ_g can be used to establish the global model.

Improve inference efficiency. Notably, during the inference process, assuming a batch size of B, since each sample is associated with distinct prompts, the text encoder needs to be executed B times. This issue is also encountered in other visually conditioned prompt methods (Zhou et al. 2022a). To address this concern, we propose an enhanced strategy in Figure 2. By precomputing the text encodings for each key, we eliminate the need for redundant execution of the text encoder, leading to a significant reduction in inference costs.

Experiments

We conduct extensive experiments to evaluate the efficacy of FedAPT. We aim to answer the following questions: 1) Can we achieve better performance with fewer learnable parameters in FL using the pre-trained model CLIP, compared to a fully-trained ViT or ResNet? 2) How much improvement in global model can FedAPT achieve compared to existing federated tuning algorithms?

We first compare FedAPT with various baselines on two large-scale multi-domain image classification datasets. We consider two scenarios, where each domain acts as a client independently and each domain is divided into five non-IID clients. Then w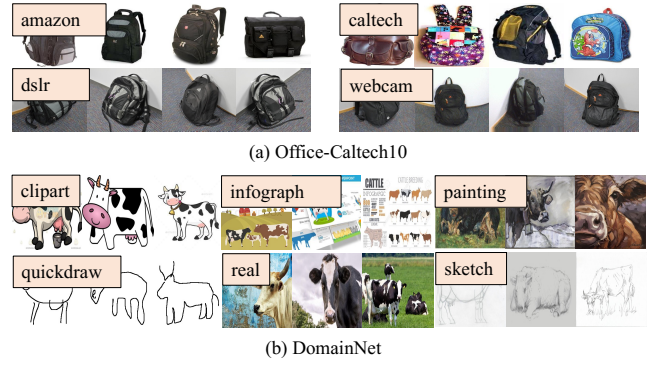e conduct several ablation studies to observe the impact of keys on the client model and global model. We also explore the impact of different initialized key values on the performance of the global model.

Experimental Setup

Datasets. We adopt two datasets, Office-Caltech10 (Gong et al. 2012) and DomainNet (Peng et al. 2019a). **Office-Caltech10** (OC) is a small dataset that has four domains (*amazon*, *caltech*, *dslr*, and *webcam*), and each domain contains 10 overlapping categories between the Office-31 (Saenko et al. 2010) dataset and the Caltech-256 (Griffin, Holub, and Perona 2007) dataset. There are at least 100 and at most 800 images in different domains. Since the number of images is small, we use each domain as a client. **DomainNet** (DN) is a large-scale multi-domain dataset containing six domains (*clipart*, *infograph*, *painting*, *quickdraw*, *real*, and *sketch*), and each domain contains about 33k-120k images with 345 categories. We set the number of clients N_{dom} in each domain as 1 or 5. When $N_{dom} = 5$, we use the Dirichlet distribution to construct the non-IID property across clients. Both datasets contain 224×224 pixel images.

Compared Methods. To comprehensively evaluate the performance, we compare the following methods: 1) **ResNet-full**: Federated training with ResNet50 (He et al. 2016), all parameters participate in the training procedure. 2) **ResNet-tuning**, where the backbone of ResNet50 is frozen, only the last layer is trained. 3) **ViT-full** and 4) **ViT-tuning**: Replace ResNet in the above two baselines with ViT. 5) **CLIP-zs** do not make any changes to the CLIP, and directly perform zero-shot image classification in all domains. This helps us understand how much the prompt tuning improves the CLIP. 6) **CLIP-FC**: Freeze the parameters of CLIP, and insert a learnable fully-connection layer at the end of the image encoder. Only the learnable layer is shared to the server. 7) **Fed-CLIP** (Lu et al. 2023): Add an adapter at the end of the visual backbone and perform federated tuning. 8) **PromptFL** (Guo et al. 2022): A prompt tuning method for CLIP, details can be seen in Preliminaries. 9) **FedAPT**: our proposed method. In the unsupervised experiment part, we utilize the local training method proposed in Method section uniformly as FedCLIP and PromptFL have not yet explored federated tuning in an unsupervised setting.

Datasets & Domains		CLIP -zs	fully-trained		fine-tune					
			ResNet -full	ViT -full	ResNet -tuning	ViT -tuning	CLIP -FC	Fed -CLIP	Prompt -FL	Fed -APT
DN	c	65.86	32.44	63.55	52.32	71.93	74.53	70.25	75.84	**77.36**
	i	40.50	59.58	27.07	20.85	48.39	48.00	45.70	49.82	**52.17**
	p	62.25	59.58	49.00	44.66	68.06	69.08	66.16	70.81	**72.51**
	q	13.36	43.25	62.20	7.13	21.24	31.84	16.98	32.98	**48.84**
	r	80.04	74.73	68.35	65.85	80.43	83.87	83.04	83.55	**84.79**
	s	57.92	61.81	54.05	38.33	63.45	65.29	61.75	67.31	**68.69**
	avg	53.32	55.23	54.04	38.19	58.92	62.10	57.31	63.39	**67.39**
OC	a	95.69	95.31	36.97	95.31	**96.87**	96.35	95.31	95.83	95.83
	w	93.24	100.00	36.20	98.27	100.00	100.00	98.27	98.27	**100.00**
	d	95.23	100.00	12.90	93.54	96.77	93.54	96.77	96.77	**100.00**
	c	92.54	96.44	29.77	93.77	97.33	96.00	96.00	**98.22**	96.44
	avg	94.18	97.94	28.96	95.22	97.74	96.47	96.58	97.27	**98.07**
#params (M)		0	24.21	87.86	0.71	0.18	0.26	0.52	2.826	2.829

Table 1: The accuracy of the global model in various domains in supervised setting with domain differences. The last row is the number of parameters to be transmitted in each round of communication. Overall, these results indicate that FedAPT achieves the best performance with few learnable parameters.

β		c	i	p	q	r	s
0.01	CLIP-FC	67.6	44.3	65.3	22.9	79.4	59.6
	FedCLIP	**68.4**	**45.0**	64.4	16.7	**82.2**	**60.1**
	PromptFL	66.6	43.7	65.3	24.4	77.7	58.7
	FedAPT	67.2	43.6	**66.4**	**26.7**	77.8	57.8
0.5	CLIP-FC	71.8	46.7	68.1	31.8	83.0	63.2
	FedCLIP	69.5	45.6	65.1	18.0	82.6	60.3
	PromptFL	73.1	47.7	69.5	37.3	82.9	64.8
	FedAPT	**74.7**	**49.9**	**71.2**	**46.8**	**84.4**	**66.6**
5	CLIP-FC	73.5	47.3	68.6	34.0	83.8	64.3
	FedCLIP	69.3	45.6	65.5	18.6	82.7	60.7
	PromptFL	74.4	48.3	69.7	38.3	83.9	66.0
	FedAPT	**75.6**	**50.6**	**71.5**	**50.5**	**85.0**	**67.6**

Table 2: The accuracy of the global model in supervised setting with domain and category differences. There are 30 clients in total. FedAPT's global model can significantly improve performance in various domains.

Implementation Details. We use PyTorch to implement all methods. The SGD optimizer is used for both datasets. We set Office-Caltech10 with a learning rate of 0.001 and batch size of 32, and DomainNet with a learning rate of 0.01 and batch size of 256. The global communication round T_g is set to 50, and the local training epoch T_l is set to 1. The length of prompts s is 16. In supervised setting, we set class-specific prompts $\mathbf{p}_g \in \mathbb{R}^{C \times s \times d}$. In unsupervised setting, due to the lack of supervision information, we use the class-shared prompts $\mathbf{p}_g \in \mathbb{R}^{s \times d}$. The CLIP used in this paper takes ViT-B/32 (Dosovitskiy et al. 2020) as the image encoder. Each experiment is repeated three times with different seeds, and the mean result is reported. All experiments are completed with one GeForce RTX 3090 GPU.

Main Results

Supervised setting with domain differences. We first treat each domain as a separate client with the same class distribution but different domain characteristics.

The results on the two datasets are reported in Table 1. We highlight the following points: **1)** From ResNet-full vs. ResNet-tuning and ViT-full vs. ViT-tuning, we can see that pre-trained ViT has more powerful representation ability than pre-trained ResNet. As a result, ResNet must be fully trained in order to achieve the accuracy rate obtained through ViT-tuning. **2)** CLIP's zero-shot classification capability is insufficient to meet the requirements of various domains. The performance gap with federated prompt tuning is large, which indicates that the original CLIP lacks knowledge of the client domains. **3)** Inserting a fully-connection layer after the image encoder of CLIP can remarkably enhance the performance on DomainNet, as indicated by CLIP-FC, which approaches the basic version of PromptFL. **4)** Considering the performance of different methods and the amount of communication parameters in each round, FedAPT achieves the highest performance at a very low cost, which shows its great potential in the application of pre-trained CLIP to FL scenarios.

Supervised setting with domain and category differences. We split each domain in DomainNet into five clients, i.e., sample $\boldsymbol{\pi}_c \sim \mathrm{Dir}\left(\beta \mathbf{1}_5\right)$ and allocate a $\boldsymbol{\pi}_{c,i}$ proportion of the instances with label c to the training set of the i-th client in each domain. Smaller β leads to larger data distribution differences among clients and more imbalanced classes. Finally, we obtain 30 clients. In addition to the divergences in data domains, there are also divergences in category distribution among clients. In each communication round, we randomly select one client from each domain, meaning six clients are selected for training. Note that because the random seeds are the same, the clients selected in each round by different methods are consistent.

The results are shown in Table 2. As we can see, FedAPT still outperforms the competitors overall. Furthermore, it is worth noting that when dealing with extremely unbalanced categories, all CLIP tuning methods exhibit similar performance. This suggests that the pre-trained CLIP model needs to be augmented with additional designs to achieve better performance, as its existing capabilities may be insufficient.

	c	i	p	q	r	s
		domain differences				
FedCLIP	68.2	45.1	64.7	18.0	82.0	60.0
PromptFL	68.3	46.6	65.6	17.3	82.1	60.4
FedAPT	69.4	47.1	66.1	17.8	83.0	61.1
		domain and category differences				
β =0.01 FedCLIP	67.6	43.3	63.4	17.5	81.3	59.1
β =0.01 PromptFL	68.3	44.4	64.3	16.3	81.2	60.2
β =0.01 FedAPT	68.2	44.9	64.2	16.7	81.5	60.1
β =0.5 FedCLIP	67.5	44.1	63.6	18.2	81.5	59.0
β =0.5 PromptFL	68.7	46.2	65.1	17.6	82.9	60.6
β =0.5 FedAPT	69.0	46.6	65.8	18.3	83.0	60.9
β =5 FedCLIP	67.8	44.3	63.8	18.1	81.5	59.2
β =5 PromptFL	68.6	46.2	65.6	17.5	82.5	60.3
β =5 FedAPT	69.1	47.3	66.4	18.4	83.3	61.5

Table 3: The accuracy of the global model in various domains in unsupervised settings.

	c	i	p	q	r	s	Avg
$\mathbf{p}_g$	74.6	48.0	68.8	22.1	82.7	65.9	60.3
$+\mathbf{p}_g \odot \mathbf{e}'_0$	**78.7**	44.7	65.8	20.4	80.8	63.8	59.0
$+\mathbf{p}_g \odot \mathbf{e}'_1$	70.9	**52.7**	65.8	19.2	81.0	62.4	58.7
$+\mathbf{p}_g \odot \mathbf{e}'_2$	71.9	46.05	**74.5**	16.6	81.1	61.2	58.5
$+\mathbf{p}_g \odot \mathbf{e}'_3$	69.6	39.9	60.2	**48.8**	75.7	58.6	58.8
$+\mathbf{p}_g \odot \mathbf{e}'_4$	73.7	47.5	68.1	19.6	**85.9**	64.4	59.9
$+\mathbf{p}_g \odot \mathbf{e}'_5$	72.2	44.9	65.1	20.6	81.0	**70.4**	59.0
Eq. 5	*77.3*	*52.1*	*72.5*	*48.8*	*84.7*	*68.6*	*67.3*

Table 4: The impact of combining meta prompts with keys of different domains.

Unsupervised setting. We replicate the above experiments in an unlabeled setting, and the results are shown in Table 3. Specifically, we observe that **1)** On DomainNet dataset, the performance of unsupervised federated tuning is comparable to that of supervised tuning. This indicates that the unsupervised fine-tuning method proposed in Preliminaries is effective. **2)** Under the unsupervised setting, FedAPT still outperforms the baseline methods, demonstrating its significant effectiveness. In addition, as we use the class-shared prompts for FedAPT and PromptFL in this setting, the communication parameter sizes per round are 0.011M and 0.008M, respectively. This demonstrates the advantage of federated prompt tuning in terms of communication costs.

Impact of keys. We do a comprehensive study of the role of 'key'. We disassemble the global model trained with FedAPT in Table 1 and compare its performance under different situations: 1) Use the learned meta prompt $\mathbf{p}_g$ to test without any key. 2) Use keys from different domains for testing, such as $\mathbf{p}_g + \mathbf{p}_g \odot \mathbf{e}'_k$. 3) Use the complete APT module, i.e., Eq. 5. The results are reported in Tabe 4. The classification ability of the meta prompt $\mathbf{p}_g$ is shown in the first row. Because there is no suitable key, its performance is mediocre. Then the middle six rows show the performance of $\mathbf{p}_g$ with different keys. We can see that the k-th key $\mathbf{e}'_k$ can significantly improve the accuracy in the k-th domain.

The affect of the adaptive network. We evaluate the performance of the global model in both client domains and

	c	i	p	q	r	s
w/o key	78.30	51.48	73.99	49.05	85.35	69.72
w key	78.03	51.59	74.00	49.35	85.67	69.73

Table 5: The impact of keys in local training. We show that adding the keys does not destroy the local training.

		i	p	q	r	s	unseen (c)
PromptFL		49.7	71.1	34.0	83.5	67.2	70.6
FedAPT	$\tau = 0.01$	52.2	72.7	51.2	84.6	69.3	68.2
FedAPT	$\tau = 1$	52.2	72.1	51.0	84.1	68.7	70.5

Table 6: The impact of the adaptive network on seen and unseen domains.

unseen new domains. We introduce softmax temperature τ to the output of the adaptive network, and the performance at $\tau = 0.01$ can be interpreted as directly determining "which key should be used". The results in Table 6 show that directly selecting a single key ($\tau = 0.01$) improves the performance in client domains but sacrifices generalization in unseen domains. By adjusting τ, we can significantly enhance client performance while maintaining nearly unchanged generalization performance in unknown domains.

Other results. 1) One worrying point about using the key on the client side is whether the key will limit the learning ability of the local model. We verify this point with an experiment. We repeat the experiment in Table 1 and report the performance of local models from different domains in Table 5. The results show that the use of the key does not limit the learning of the local model. **2)** We also compare different initialized keys. The results suggest that the keys work well for different types of random initialization, including the uniform and the standard normal distribution initialization and random orthogonal initialization. We provide the numerical results in the supplementary materials. **3)** After precomputing and saving the text branch outputs of different keys, only the image branch and domain classification weights need to be computed during inference. The final inference cost is as follows: CLIP's GFLOPs is 4.42 while that of ResNet50 is 4.14. **4)** We demonstrate the impact in inference efficiency of the strategy in Figure 2. While maintaining nearly unchanged accuracy, the inference time has been reduced from thirty minutes to 30 seconds in one GPU card. Detailed numerical results are provided in the supplementary materials.

Conclusions

In this paper, we apply the pre-trained CLIP to the multi-domain federated learning in both supervised and unsupervised settings, and propose an adaptive prompt tuning method that uses domain-specific keys to generate specific prompts for each test sample. We have extensively validated the effectiveness of FedAPT. With less than 10% of learnable parameters, FedAPT achieves performance surpassing that of fully-trained models. Moreover, when faced with challenges such as feature and category differences in client data, FedAPT demonstrates a significant performance improvement compared with the competitors.

Acknowledgements

References

Chen, H.-Y.; Tu, C.-H.; Li, Z.; Shen, H.-W.; and Chao, W.-L. 2022. On pre-training for federated learning. *arXiv preprint arXiv:2206.11488*.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Dinh, C. T.; Tran, N. H.; and Nguyen, T. D. 2020. Personalized Federated Learning with Moreau Envelopes. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Fallah, A.; Mokhtari, A.; and Ozdaglar, A. 2020. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33: 3557–3568.

Feng, C.-M.; Yu, K.; Liu, N.; Xu, X.; Khan, S.; and Zuo, W. 2023. Towards Instance-adaptive Inference for Federated Learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 23287–23296.

Gong, B.; Shi, Y.; Sha, F.; and Grauman, K. 2012. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE conference on computer vision and pattern recognition*, 2066–2073. IEEE.

Griffin, G.; Holub, A.; and Perona, P. 2007. Caltech-256 object category dataset.

Guo, T.; Guo, S.; Wang, J.; and Xu, W. 2022. PromptFL: Let Federated Participants Cooperatively Learn Prompts Instead of Models–Federated Learning in Age of Foundation Model. *arXiv preprint arXiv:2208.11625*.

Hanzely, F.; Hanzely, S.; Horváth, S.; and Richtárik, P. 2020. Lower bounds and optimal algorithms for personalized federated learning. *Advances in Neural Information Processing Systems*, 33: 2304–2315.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hsu, T.-M. H.; Qi, H.; and Brown, M. 2020. Federated visual classification with real-world data distribution. In *European Conference on Computer Vision*, 76–92. Springer.

Jia, C.; Yang, Y.; Xia, Y.; Chen, Y.-T.; Parekh, Z.; Pham, H.; Le, Q.; Sung, Y.-H.; Li, Z.; and Duerig, T. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, 4904–4916. PMLR.

Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; and Suresh, A. T. 2020. SCAFFOLD: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, 5132–5143. PMLR.

Li, B.; Chi, M.; Fan, J.; and Xue, X. 2007. Support cluster machine. In *Proceedings of the 24th International Conference on Machine Learning*, 505–512.

Li, Q.; He, B.; and Song, D. 2021. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10713–10722.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated Optimization in Heterogeneous Networks. In Dhillon, I. S.; Papailiopoulos, D. S.; and Sze, V., eds., *MLSys*. mlsys.org.

Li, X.; Jiang, M.; Zhang, X.; Kamp, M.; and Dou, Q. 2021. FedBN: Federated Learning on Non-IID Features via Local Batch Normalization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2021a. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Liu, X.; Ji, K.; Fu, Y.; Du, Z.; Yang, Z.; and Tang, J. 2021b. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*.

Liu, Y.; Huang, A.; Luo, Y.; Huang, H.; Liu, Y.; Chen, Y.; Feng, L.; Chen, T.; Yu, H.; and Yang, Q. 2020. Fedvision: An online visual object detection platform powered by federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 13172–13179.

Lu, W.; Hu, X.; Wang, J.; and Xie, X. 2023. FedCLIP: Fast Generalization and Personalization for CLIP in Federated Learning. *arXiv preprint arXiv:2302.13485*.

Luo, Z.; Wang, Y.; Wang, Z.; Sun, Z.; and Tan, T. 2022. Disentangled Federated Learning for Tackling Attributes Skew via Invariant Aggregation and Diversity Transferring. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, 14527–14541. PMLR.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of

deep networks from decentralized data. In *Artificial Intelligence and Statistics*, 1273–1282. PMLR.

Nguyen, J.; Wang, J.; Malik, K.; Sanjabi, M.; and Rabbat, M. 2023. Where to Begin? Exploring the Impact of Pre-Training and Initialization in Federated. In *International Conference on Learning Representations*.

Peng, X.; Bai, Q.; Xia, X.; Huang, Z.; Saenko, K.; and Wang, B. 2019a. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1406–1415.

Peng, X.; Huang, Z.; Zhu, Y.; and Saenko, K. 2019b. Federated adversarial domain adaptation. *arXiv preprint arXiv:1911.02054*.

Qiu, C.; Li, X.; Mummadi, C. K.; Ganesh, M. R.; Li, Z.; Peng, L.; and Lin, W.-Y. 2023. Text-driven Prompt Generation for Vision-Language Models in Federated Learning. *arXiv preprint arXiv:2310.06123*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 8748–8763. PMLR.

Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečnỳ, J.; Kumar, S.; and McMahan, H. B. 2020. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*.

Saenko, K.; Kulis, B.; Fritz, M.; and Darrell, T. 2010. Adapting visual category models to new domains. In *European conference on computer vision*, 213–226. Springer.

Singh, S. P.; and Jaggi, M. 2020. Model fusion via optimal transport. *Advances in Neural Information Processing Systems*, 33.

Su, S.; Li, B.; and Xue, X. 2023. One-shot Federated Learning without server-side training. *Neural Networks*, 164: 203–215.

Su, S.; Li, B.; Zhang, C.; Yang, M.; and Xue, X. 2023. Cross-domain Federated Object Detection. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*, 1469–1474. IEEE.

Tan, Y.; Long, G.; Ma, J.; Liu, L.; Zhou, T.; and Jiang, J. 2022. Federated learning from pre-trained models: A contrastive learning approach. *Advances in neural information processing systems*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, H.; Yurochkin, M.; Sun, Y.; Papailiopoulos, D. S.; and Khazaeni, Y. 2020a. Federated Learning with Matched Averaging. In *International Conference on Learning Representations*.

Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; and Poor, H. V. 2020b. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in Neural Information Processing Systems 33*.

Wang, Z.; Wu, Z.; Agarwal, D.; and Sun, J. 2022. MedCLIP: Contrastive Learning from Unpaired Medical Images and Text. *arXiv preprint arXiv:2210.10163*.

Yang, M.; Su, S.; Li, B.; and Xue, X. 2023. Exploring One-shot Semi-supervised Federated Learning with A Pre-trained Diffusion Model. *arXiv preprint arXiv:2305.04063*.

Yang, S.; Wang, Y.; Wang, K.; Jui, S.; et al. 2022. Attracting and dispersing: A simple approach for source-free domain adaptation. In *Advances in Neural Information Processing Systems*.

Yao, L.; Han, J.; Wen, Y.; Liang, X.; Xu, D.; Zhang, W.; Li, Z.; Xu, C.; and Xu, H. 2022. DetCLIP: Dictionary-Enriched Visual-Concept Paralleled Pre-training for Open-world Detection. *arXiv preprint arXiv:2209.09407*.

Yu, F.; Rawat, A. S.; Menon, A.; and Kumar, S. 2020. Federated learning with only positive labels. In *International Conference on Machine Learning*, 10946–10956. PMLR.

Yu, H.; Wang, X.; Li, B.; and Xue, X. 2023. Chinese Text Recognition with A Pre-Trained CLIP-Like Model Through Image-IDS Aligning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11943–11952.

Zhong, Y.; Yang, J.; Zhang, P.; Li, C.; Codella, N.; Li, L. H.; Zhou, L.; Dai, X.; Yuan, L.; Li, Y.; et al. 2022. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16793–16803.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022a. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 16816–16825.

Zhou, K.; Yang, J.; Loy, C. C.; and Liu, Z. 2022b. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9): 2337–2348.

Zhuang, W.; Wen, Y.; and Zhang, S. 2021. Joint optimization in edge-cloud continuum for federated unsupervised person re-identification. In *Proceedings of the 29th ACM International Conference on Multimedia*, 433–441.

Zhuang, W.; Wen, Y.; Zhang, X.; Gan, X.; Yin, D.; Zhou, D.; Zhang, S.; and Yi, S. 2020. Performance optimization of federated person re-identification via benchmark analysis. In *Proceedings of the 28th ACM International Conference on Multimedia*, 955–963.