Pilot: Building the Federated Multimodal Instruction Tuning Framework

Baochen Xiong[1,2,3], Xiaoshan Yang[1,2,3], Yaguang Song[2], Yaowei Wang[2,4], Changsheng Xu[1,2,3*]

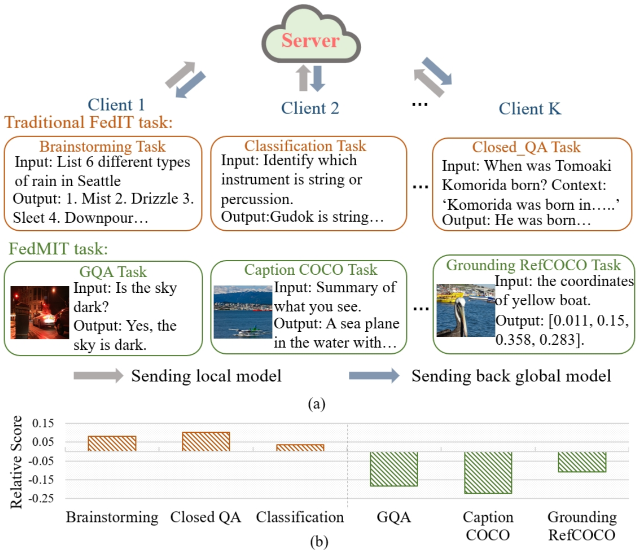
[1]MAIS, Institute of Automation, Chinese Academy of Sciences
[2]Pengcheng Laboratory
[3]School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[4]Harbin Institute of Technology (Shenzhen)
xiongbaochen2022@ia.ac.cn, {xiaoshan.yang, csxu}@nlpr.ia.ac.cn, {songyg01, wangyw}@pcl.ac.cn

Abstract

In this paper, we explore a novel federated multimodal instruction tuning task(FedMIT), which is significant for collaboratively fine-tuning MLLMs on different types of multimodal instruction data on distributed devices. To solve the new task, we propose a federated multimodal instruction tuning framework(Pilot). Our framework integrates two stages of "adapter on adapter" into the connector of the vision encoder and the LLM. In stage 1, we extract task-specific features and client-specific features from visual information. In stage 2, we build the cross-task Mixture-of-Adapters(CT-MoA) module to perform cross-task interaction. Each client can not only capture personalized information of local data and learn task-related multimodal information, but also learn general knowledge from other tasks. In addition, we introduce an adaptive parameter aggregation strategy for text training parameters, which optimizes parameter aggregation by calculating weights based on the euclidean distance between parameters, so that parameter aggregation can benefit from positive effects to the greatest extent while effectively reducing negative effects. Our framework can collaboratively exploit distributed data from different local clients to learn cross-task knowledge without being affected by the task heterogeneity during instruction tuning. The effectiveness of our method is verified in two different cross-task scenarios.

Introduction

The emergence of Multimodal Large Language Models (MLLMs) (Li et al. 2023; Liu et al. 2024; Driess et al. 2023; Dai, Li et al. 2023) has significantly advanced the field of artificial intelligence. MLLMs have shown excellent ability in processing and integrating various modalities information (especially text and image), and has achieved remarkable performance in tasks such as text generation, machine translation and question answering. Enhancing the zero-shot generalization ability of MLLMs on novel multimodal tasks is a key goal driving its development. Multimodal instruction tuning has been shown to be highly effective in improving the zero-shot generalization ability of models to unseen multimodal problems (Xu et al. 2022; Ye et al. 2023; Sun et al. 2024; Chen et al. 2024a; Xiao et al. 2024b).


Server
Client 1
Client 2
...
Client K
Traditional FedIT task:
Brainstorming Task
Input: List 6 different types of rain in Seattle
Output: 1. Mist 2. Drizzle 3. Sleet 4. Downpour…
Classification Task
Input: Identify which instrument is string or percussion.
Output:Gudok is string…
Closed_QA Task
Input: When was Tomoaki Komorida born? Context: 'Komorida was born in…..'
Output: He was born…
FedMIT task:
GQA Task
Input: Is the sky dark?
Output: Yes, the sky is dark.
Caption COCO Task
Input: Summary of what you see.
Output: A sea plane in the water with…
Grounding RefCOCO Task
Input: the coordinates of yellow boat.
Output: [0.011, 0.15, 0.358, 0.283].
Sending local model
Sending back global model
Relative Score
0.15
0.05
-0.05
-0.15
-0.25
Brainstorming
Closed QA
Classification
GQA
Caption COCO
Grounding RefCOCO

(a)

(b)

Figure 1: (a) Comparison of the FedIT task with the FedMIT task. (b) Compare the relative scores of traditional FedIT methods on the two tasks.

Current instruction tuning methods are usually implemented by centralized learning paradigm, where one central party collects a substantial amount of data to train the model, which may lead to privacy security issues. Federated Learning (FL) is currently the primary framework for distributed training of models while maintaining data security and privacy. McMahan et al. (2017) proposed the first federated learning algorithm, FedAvg, which aggregates model gradients from local clients to a central server without data sharing. Afterwards, FL is also successfully used for instruction tuning of pre-trained models (Zhang et al. 2024b; Ye et al. 2024), called Federated Instruction Tuning (FedIT). These frameworks can effectively leverage locally private instruction-following data.

Although existing FedIT approaches have made important progress, they only consider the case where different clients perform different instruction-based natural language processing(NLP) tasks, and there is little exploration in the multimodal field. The integration of vision and language plays a crucial role in machine perception and understanding of the world. Language is the basis of cognitive processing,

[*]Corresponding author.

while vision provides the necessary sensory information (Xu et al. 2024; Xiong et al. 2024). In this context, we attempt to transfer the success of FedIT in multimodal tasks. Therefore, in this paper, we explore a novel Federated Multimodal Instruction Tuning task (FedMIT).

Compared with traditional FedIT, FedMIT focuses on the client containing different multimodal instruction tuning tasks (e.g., visual question answering and image captioning), as shown in Figure 1(a). In our preliminary study, we first distributed different task instruction data to each client, including "Caption (e.g., COCO (Lin et al. 2014))" for image description, "VQA (e.g., GQA (Hudson 2019))" for visual question answering, and "Grounding (e.g., RefCOCO (Kazemzadeh et al. 2014))" for visual localization. Then, we directly apply the representative method Shepherd (Zhang et al. 2024b) in the FedIT task to the FedMIT task. Figure 1(b) shows the relative scores of Shepherd on two tasks, where the relative scores are the comparison scores between the results of the Shepherd method and the local training results. The results show that the traditional FedIT method does not perform well on the FedMIT task. Since the diversity of multimodal tasks greatly increases the heterogeneity between clients, we believe that traditional FedIT methods cannot adequately address this kind of task heterogeneity.

Compared with the traditional FedIT task, each client in FedMIT task not only needs to capture the personalized information of local data and task-related multimodal information, but also needs to be able to accommodate the differences between different tasks to avoid parameter conflicts. This requires the model to maintain its understanding of the task and local data while also being able to learn general knowledge from other tasks to improve model performance and cross-task ability.

Inspired by these observations, we introduce the Federated Multimodal Instruction Tuning framework: Pilot. Our framework integrates two stages of "adapter on adapter" into the connector of the vision encoder and the LLM, and adopts an adaptive parameter aggregation strategy for the training parameters of the LLM. First we introduce the two-stage training of the client. **Stage 1**: Task-specific feature mining. We hope to extract client-specific and task-specific features from the client's visual information. We propose task-specific adapter to extract task-specific visual features that is only important for one task, and client-specific adapter to extract specific visual features of the client's unique data distribution. To encourage the client-specific adapter to produce features that are more refined than the task-specific adapter, a difference loss is used to ensure the orthogonality of their output features. **Stage II**: Cross-task visual interaction. We integrate the Cross-task Mixture-of-Adapters(CT-MoA) module with the task-specific adapter. By interacting with the server, each adapter in CT-MoA is initialized from the task-specific adapter of the corresponding task. We hope that the CT-MOA module can learn general knowledge from other tasks to improve model performance and cross-task capabilities. Therefore, we added cross-task adapters to the task-specific adapters of other tasks in CT-MOA, where cross-task adapter on other task-specific adapter. Cross-task adapter aims to extract cross-task collaboration visual features. In addition, the CT-MOA module also contains a router that selects adapters during the stage II with auxiliary losses on the router to maintain a balanced loading of adapters. Considering the computation and communication requirements, we adopt text-adapter-based parameter-efficient tuning techniques to train LLM to reduce the amount of trainable parameters on each device. For the server side, it collects all client visual and text training parameters. For visual training parameters, we adopt the task-aware aggregation strategy. For text training parameters, we introduce adaptive parameter aggregation, which optimizes parameter aggregation by calculating weights based on the euclidean distance between parameters, so that parameter aggregation can benefit from positive effects to the greatest extent while effectively reducing negative effects. Combining federated optimization with two-stage local updates, our framework can collaboratively exploit distributed data from different local clients to learn cross-task knowledge without being affected by the task heterogeneity during instruction tuning.

Our contributions are summarized as follows: (1) We propose to explore a new task of federated multimodal instruction tuning, which is significant for collaboratively fine-tuning MLLMs on different types of multimodal instruction data on distributed devices. (2) To solve the new task, we propose a Federated Multimodal Instruction Tuning framework(Pilot). Our framework builds two stages "adapter on adapter" strategy. In stage 1, the model extracts client-specific and task-specific features, and in stage 2, we construct CT-MOA modules to learn cross-task interactions. We adopt an adaptive aggregation strategy for the LLM training parameters. With the above approach, our method can learn cross-task knowledge without being affected by task heterogeneity during instruction tuning. (3) We verify the effectiveness of Pilot on the state-of-the-art LLaVA (Liu et al. 2024) in two different cross-task scenarios.

Related Work

Federated Learning

The earliest FL algorithm is FedAvg (McMahan et al. 2017), which builds the global model by averaging the local updates obtained by Stochastic Gradient Descent (SGD) (Gorbunov, Hanzely, and Richtárik 2021). However, FedAvg inevitably experiences performance degradation on non-IID data (Yang et al. 2024b; Xiong et al. 2023). To deal with this problem, FedProx (Li et al. 2020) adds a proximal term to local targets to minimize the distance between the global model and the local model for non-IID data. PerAvg (Fallah, Mokhtari, and Ozdaglar 2020) uses popular meta-learning framework MAML (Finn, Abbeel, and Levine 2017), which allows each client to quickly adapt to local data by finding a suitable initialization. TAPPFL (Arevalo et al. 2024) designs a task-agnostic and provably privacy-preserving federated learning framework. FedTGP (Zhang et al. 2024a) uses adaptive-margin-enhanced contrastive learning to learn trainable global prototypes on the server to solve the model heterogeneity problem. FedLPS (Jia et al. 2024) uses an

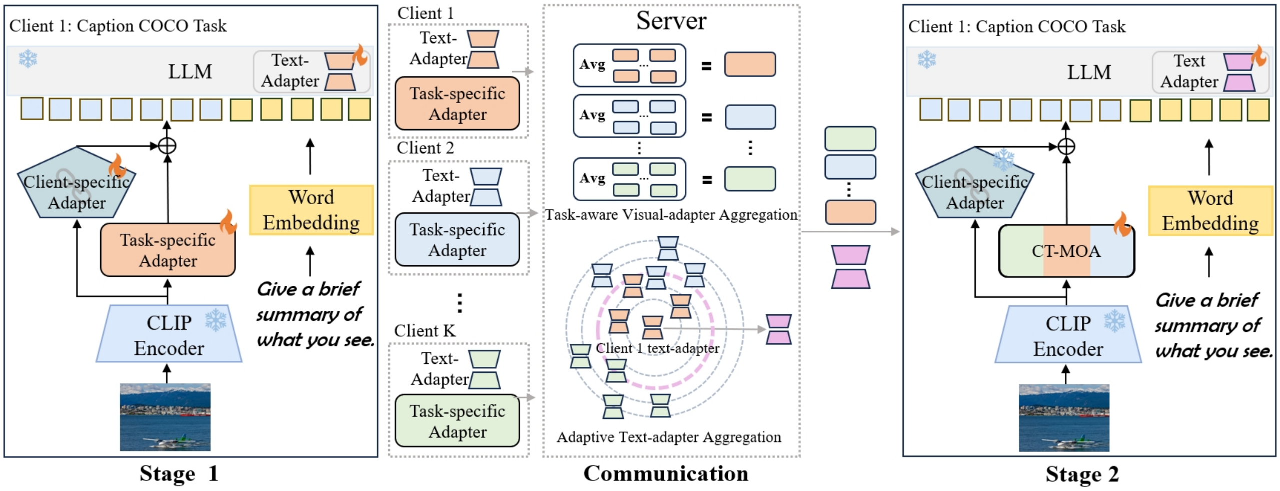


Figure 2: Overview of the proposed Federated Multimodal Instruction Tuning Framework (Pilot).

adaptive channel model pruning method to enable clients to participate in federated learning training with heterogeneous tasks. However, the above methods only consider the heterogeneity on single-modal clients. In contrast, our method focuses on addressing the heterogeneity of cross-task instruction data in the MLLM-based federated multimodal instruction tuning task.

Multimodal Instruction Tuning

Instruction tuning (Wei et al. 2021; Xiao et al. 2024a) significantly improves the generalization ability of large language models to unseen tasks based on natural language instructions. With the advent of MLLMs, the scope of instruction tuning has been expanded to include multimodal and visual tasks. MiniGPT-4 (Zhu et al. 2023) and LLaVA (Liu et al. 2024) keep the visual encoder frozen and adjust the language model, extending instruction tuning to multimodal. InstructBLIP (Dai, Li et al. 2023) enhances zero-shot capabilities by performing instruction tuning on multiple datasets. Shikra (Chen et al. 2023) extend MLLMs to visual grounded tasks using instructions with bounding box coordinates. mPLUG-Owl (Ye et al. 2023) fine-tunes the visual and text encoders in two stages. EProj (He et al. 2023) address the catastrophic forgetting of the model in continuous instruction tuning. MixLoRA (Shen et al. 2024) uses conditional mixed LoRA to address the task interference caused by LoRA during multimodal instruction tuning. Although these studies demonstrate excellent zero-shot capabilities, they still suffer from the risk of privacy leakage.

Federated Instruction Tuning

There has been some research in federated learning based on Parameter-Efficient Fine-tuning(PEFT) (Shi et al. 2023; Su et al. 2024; Chen and Zhang 2024). FedCLIP (Lu et al. 2023) and pFedprompt (Guo et al. 2023) finetune the CLIP model (Radford et al. 2021) by adapting and conveying only small number of learnable prompts. FedDISC (Yang et al. 2024a) first integrated the pre-trained diffusion model (Dhariwal and Nichol 2021) into the FL framework. FedDAT (Chen et al. 2024b)utilizes a dual-adapter teacher architecture to address the efficient fine-tuning of parameters of multimodal base models in heterogeneous federated learning. However, the application of instruction tuning in FL has not been fully explored. Federated instruction tuning (Zhang et al. 2024b) provides a simple and effective method for supporting distributed client privacy-preserving instruction tuning through the FL protocol. Fatellm (Fan et al. 2023) and OpenFedLLM (Ye et al. 2024) built a concise and easy-to-research framework for fine-tuning federal instructions. Recently, FedDPA (Yang et al. 2024c) explored the problem of heterogeneity in NLP tasks. In our work, we attempt to address the unexplored task of multimodal instruction tuning for MLLMs in federated learning.

Methodology

Problem Definition

We assume that there is a set of multimodal instruction tuning data $\mathcal{D} = \{(\mathcal{D}_k, t_k)\}_{k=1}^{K}$ from K clients, where $\mathcal{D}_k = \{X_{i,k}^{v}, \mathbf{x}_{i,k}^{ins}, \mathbf{y}_{i,k}^{ans}\}_{i=1}^{n_k}$ is the set of n_k data pairs on the k-th client. The total number of data pairs is $n = \sum_{k=1}^{K} n_k$. $X^v, \mathbf{x}^{ins}$ and $\mathbf{y}^{ans}$ indicate the image, instruction tokens and answer tokens, respectively. $t_k \in \{1, ..., T\}$ denotes the task type of the k-th client. T is the total number of tasks and $T \leqslant K$.

In the FedMIT task, all clients obtain the pre-trained MLLM from the server. Generally, MLLMs contain a visual encoder f, a connector ψ, and LLM L. Specifically, for a given input image X^v, the visual encoder extracts visual features H^v = $f(X^v)$. The connector is used to align the visual encoder with the LLM. Connector transforms H^v into a language embedding tokens $\mathbf{x}^{img} \in \mathbb{R}^{N \times C}$, effectively facilitating the integration of multimodal information within the LLM framework, where N is the number of tokens and C is the hidden size.

$$\mathbf{x}^{img} = \psi(H^v), \text{ with } H^v = f(X^v). \qquad (1)$$

Finally, we input $\mathbf{x}^{img}$ and $\mathbf{x}^{ins}$ into the LLM to generate response.

The FedMIT task aims to perform instruction tuning in a distributed manner, where each client can increment the local model by leveraging cross-task knowledge learned on other clients without the need for centralized data collection. The overall optimization goal is defined as follows:

$$\sum_{k=1}^{K}\frac{n_k}{n}\sum_{i=1}^{n_k}\frac{1}{n_k}\sum_{j=1}^{L}-\log p_\theta\left(\mathbf{y}_{i,k}^{j,ans} \mid \mathbf{x}_{i,k}^{img},\mathbf{x}_{i,k}^{ins},\mathbf{y}_{i,k}^{<j,ans}\right) \tag{2}$$

where L represents the answer length, θ is the trainable parameters. $\mathbf{y}_{i,k}^{j,ans}$ indicates the j-th answer token and $\mathbf{y}_{i,k}^{<j,ans}$ indicates all answer tokens before the index j.

Federated Multimodal Instruction Tuning Framework

FedMIT task has greater heterogeneity between tasks, and traditional FedIT methods cannot be used directly to solve the problem. As shown in Figure 2, we propose the Federated Multimodal Instruction Tuning framework (Pilot) to address the task heterogeneity between clients. In our framework, we integrate a two-stage "adapter on adapter" method into the connector of the visual encoder and LLM. In stage 1, we extract task-specific features and client-specific features from visual information. Through federated aggregation, in stage 2, we build a CT-MOA module to perform cross-task interaction. We hope that each client not only needs to capture personalized information from local data and learn task-related multimodal information, but also needs to be able to learn general knowledge from other tasks to improve model performance and cross-task capabilities.

Stage 1: Task-specific Feature Mining. At stage 1, we hope to extract client-specific and task-specific features from the client's visual information. We propose task-specific adapter ψ^t to extract task-specific visual features that is only important for one task, and client-specific adapter ψ^s to extract specific visual features of the client's unique data distribution. We define these two adapters as two-layer of perceptrons. Finally, the image tokens represent: $\mathbf{x}^{img} = \mathbf{x}^t + \mathbf{x}^s$, where $\mathbf{x}^t = \psi^t(H^v) \in \mathbb{R}^{N\times C}$, $\mathbf{x}^s = \psi^s(H^v) \in \mathbb{R}^{N\times C}$.

To encourage the client-specific adapter to produce features that are more refined than the task-specific adapter, a difference loss is used to ensure the orthogonality of their output features. Inspired by the domain separation network (Bousmalis et al. 2016), we adopt a soft subspace orthogonality constraint:

$$\mathcal{L}_d = \|{\mathbf{x}^t}^\top \mathbf{x}^s\|_F^2, \tag{3}$$

where $\|\cdot\|_F$ is Frobenius norm. Hence, the Stage 1 total loss is

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda_0 \mathcal{L}_d, \tag{4}$$

where $\mathcal{L}_{ce}$ represents the language modeling loss, which computes the cross-entropy of next-token predictions. λ_0 denote coefficients for difference loss. After stage 1, each client sends the LLM training parameters text-adapter Θ_l and task-specific adapter Θ_a parameters to the server.

Stage 2: Cross-task Visual Interaction. At stage 2, the client obtains the updated task-specific adapters for T tasks from the server. In other words, each client obtains $(T-1)$ task-specific adapters of other tasks in addition to its local task-specific adapter. To learn general knowledge from other tasks to improve model performance and cross-task capabilities, we integrate the local task-specific adapter with other task-specific adapters to construct a Cross-task Mixture-of-Adapters (CT-MoA) module, as shown in Figure 3. In the CT-MoA module, we assume that the local task-specific adapter index is 1 (ψ_1^t), and there are T adapters in total. However, directly loading task-specific adapters of distinct tasks may cause the model of the current task to be unable to fully utilize these adapters due to task heterogeneity. Therefore, we added cross-task adapters to the task-specific adapters of other tasks in CT-MOA. For task-specific adapters $\psi_i^t, i = 2, ..., T$, we add cross-task adapter ψ^c on it to alleviate the discrepancy between other task-specific adapters and the local task-specific adapter due to task heterogeneity. Cross-task adapter aims to extract cross-task collaboration knowledge. It has the same structure as the task-specific adapter and is initialized by the local task-specific adapter ψ_1^t parameters. In addition, the CT-MOA module includes a router for predicting the probability of selecting and activating each adapter from T total task-specific adapters. The router network(ϕ) has a linear layer for calculating the normalized weight matrix using H^v for voting, and producting $\mathcal{P}$:

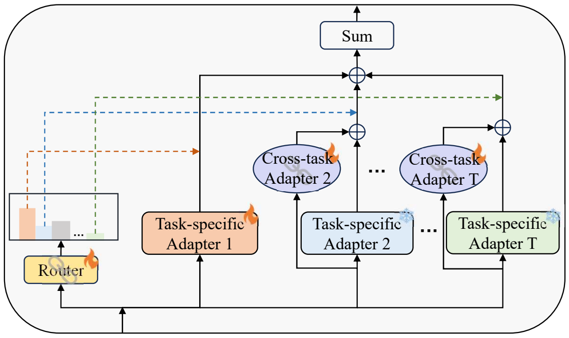


Figure 3: Cross-task Mixture-of-Adapters (CT-MoA).

$$\mathcal{P} = \mathrm{Softmax}(\phi(H^v)) \in \mathbb{R}^T. \tag{5}$$

Finally, the output of CT-MOA module is expressed as:

$$\mathbf{x}^t = \mathcal{P}[1]\psi_1^t(H^v) + \sum_{i=2}^{T}\mathcal{P}[i]\left(\psi_i^t(H^v) + \psi_i^c(H^v)\right). \tag{6}$$

Following (Zoph et al. 2022), we adopt an auxiliary losses based on the language modeling cross-entropy loss to maintain a load balance between task-specific adapters in the Cross-task MoA module. The auxiliary losses comprise load balancing loss and router z-loss. The load balancing loss is defined as:

$$\mathcal{L}_b = T\sum_{i=1}^{T}\mathcal{D}_i\mathcal{R}_i, \tag{7}$$

where $\mathcal{D}_i = \frac{1}{N}\sum_{j=1}^{N}1\{\mathrm{argmax}(\phi(H^v[j])) = i\}$ represents the proportion of tokens distributed to adapter i, $1\{\cdot\}$

is an indicator function. While $\mathcal{R}_i = \frac{1}{N}\sum_{j=1}^{N}\phi_i(H^v[j])$ denotes the proportion of the probability assigned to adapter i. $\phi_i(H^v[j])$ is the probability of routing token j to adapter i. The router z-loss is defined as:

$$\mathcal{L}_z = \frac{1}{N}\sum_{i=1}^{N}\left(\log\sum_{j=1}^{T} e^{g_j^{(i)}}\right)^2, \tag{8}$$

where $g \in \mathbb{R}^{N\times T}$ are the logits obtained by the router. Hence, the Stage 2 total loss is:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda_1\mathcal{L}_b + \lambda_2\mathcal{L}_z, \tag{9}$$

λ_1 and λ_2 denote coefficients for load balancing loss and router z-loss. During stage 2 training, the model only updates the local task-specific adapter, the cross-task adapter, and the text-adapter.

Federated Optimization

The proposed method is collaboratively optimized via local two stages instruction tuning and global aggregation. We train the Pilot with two alternate steps of local update and global aggregation for R rounds. In each round, the client receives the global model and updates the local model E epochs on the local data. In the first round, the server sends the base MLLM model to each client. Each client performs local instruction tuning of stage 1, then send the task-specific adapter and text-adapter parameters to the server. After global aggregation, the server sends T task-specific adapters and text-adapter parameters back to the client. Each client updates the model architecture and performs local instruction tuning of stage 2, and finally send the local task-specific adapter and text-adapter parameters to the server.

Task-aware Visual-adapter Aggregation. After the server collects the task-specific adapter parameters of all clients, we adopt a task-aware aggregation method.

Task-specific adapter mainly learns task-specific visual features and has the general understanding ability of this task. Therefore, the parameters learned in one client can also be shared by clients of the same task. We use weighted average to aggregate the task-specific adapter parameters Θ_a of the same task:

$$\bar{\Theta}_a^t = \sum_{k\in\mathcal{K}_t}\frac{n_k}{\sum\limits_{k'\in\mathcal{K}_t} n_{k'}}\Theta_{a,k}^t, t = 1, ..., T, \tag{10}$$

where $\mathcal{K}_t$ is a set of clients with task type t. Finally, we obtain T number of task-specific adapters and send them to each client.

Adaptive Text-adapter Aggregation. For all collected text-adapter parameters, compared with the fully weighted aggregation method used in the traditional FedIT method, we expect that the text-adapter parameters of each client can benefit from the positive impact to the greatest extent during the aggregation process, while effectively reducing the negative impact. Therefore, we propose an adaptive text-adapter aggregation method. We first calculate the Euclidean distance between each text-adapter parameter and the other $(K-1)$ text-adapter parameters. Then select the nearest M points for weighted aggregation based on the distance. For example, text-adapter parameters $\Theta_{l,k}$ for client k:

$$d_{k,i} = E(\Theta_{l,k}, \Theta_{l,i}), \text{ for } i \in \{1, 2, \ldots, K\}\backslash\{k\}, \tag{11}$$

where $E(\cdot)$ is the Euclidean distance formula. We obtain the distance $D_k \in \mathbb{R}^{K-1}$ between the $\Theta_{l,k}$ and other text-adapter parameters, and then we select the Top-M closest parameters for weighted aggregation:

$$\bar{\Theta}_{l,k} = \frac{n_k}{n_k + \sum\limits_{k'\in\mathcal{K}_m} n_{k'}}\Theta_{l,k} + \sum_{k'\in\mathcal{K}_m}\frac{n_{k'}\cdot w_{k'}}{n_k + \sum\limits_{k'\in\mathcal{K}_m} n_{k'}}\Theta_{l,k'}, \tag{12}$$

where $w_{k'} = \frac{1/d_{k,k'}}{\sum_{k'\in\mathcal{K}_m} 1/d_{k,k'}}$, $\mathcal{K}_m$ is the set of M clients closest to parameter $\Theta_{l,k}$.

Experiment

Experimental Setups

Now, we introduce how to construct the federated multimodal instruction tuning task scenario. To ensure the diversity of instruction tuning datasets, we collect various publicly available and commonly used visual-language datasets. These instruction tuning datasets cover a wide range of tasks, including knowledge-based image question answering, image question answering reading comprehension, and optical character recognition VQA. The selected datasets include ScienceQA (Lu et al. 2022), GQA (Hudson 2019), and OCRVQA (Mishra et al. 2019). However, we observe that these datasets are limited to traditional QA tasks in visual-language tasks. Therefore, to enrich the diversity of tasks, we introduce the image caption dataset COCO (Lin et al. 2014) for image description task and the grounding dataset RefCOCO (Kazemzadeh et al. 2014) for visual localization task. For all the above datasets, we construct two different federated instruction tuning scenarios. **FL-oriented visual understanding scenario**: We use the GQA, COCO, and RefCOCO datasets. We randomly divide each dataset into 3 subsets, and each subset is regarded as a client. Then, we obtain a FedMIT task scenario with 9 clients and 3 different visual understanding tasks(9-client, 3-task). **FL-oriented general VQA scenario**: We use the ScienceQA, GQA, and OCRVQA datasets. Similar to the above operations, we obtain a FedMIT task scenario with 9 clients and 3 different VQA tasks(9-client, 3-task). For more data information, please refer to the supplementary materials.

Baselines

We compare our framework Pilot with 5 state-of-the-art FL algorithms: FedAVG (2017), FedProx (2020), FedAdam (2020), Shepherd (2024b), and FedDPA (2024c). FedAvg takes the weighted average of all training parameters as a standard optimization method. FedProx focuses on local model correction, and FedAdam focuses on introducing momentum on the server side to stabilize global model updates. Shepherd and FedDPA are FedIT task method. We also show local training and centralized training as references, where local training is trained by using one client's

Methods	**GQA**			**Caption COCO**			**Grounding RefCOCO**		
	Client 1	Client 2	Client 3	Client 4	Client 5	Client 6	Client 7	Client 8	Client 9
Centralized training	Centralized test result: 52.8			Centralized test result: 130.3			Centralized test result: 65.3		
Local training	48.8	47.5	48.4	117.6	122.3	120.9	29.1	35.1	31.9
FedAvg	47.2	46.6	46.4	86.5	102.5	104.2	25.4	30.2	32.4
FedProx	48.1	46.4	47.8	98.5	101.2	113.2	35.8	45.5	39.6
FedAdam	50.8	48.8	**50.5**	117.3	119.2	122.0	47.5	45.9	45.1
FedDPA	49.3	48.2	49.3	112.2	121.4	113.6	48.2	47.3	46.2
Shepherd	43.8	39.6	44.7	75.1	86.2	101.2	24.1	30.7	28.5
Pilot	**51.4**	**49.7**	50.2	**120.3**	**126.4**	**125.1**	**49.6**	**52.4**	**50.9**

Table 1: Comparison with state-of-the-art methods under FL-oriented visual understanding scenario.

Methods	**GQA**			**ScienceQA**			**OCRVQA**		
	Client 1	Client 2	Client 3	Client 4	Client 5	Client 6	Client 7	Client 8	Client 9
Centralized training	Centralized test result: 56.7			Centralized test result: 58.0			Centralized test result: 52.3		
Local training	46.1	50.2	49.6	50.2	49.6	50.0	40.62	46.33	37.10
FedAvg	44.1	48.3	47.1	44.2	40.5	42.6	35.6	38.1	34.2
FedProx	44.5	48.3	46.6	48.0	46.3	43.9	34.2	37.5	34.6
FedAdam	45.6	49.6	50.2	50.1	51.2	49.0	42.9	49.5	42.8
FedDPA	47.3	50.2	50.6	52.3	51.9	50.4	43.5	49.6	43.2
Shepherd	43.4	45.0	46.2	37.6	36.8	39.5	35.6	35.3	30.7
Pilot	**49.2**	**52.8**	**52.3**	**54.7**	**53.2**	**53.7**	**45.7**	**50.2**	**44.9**

Table 2: Comparison with state-of-the-art methods under FL-oriented general VQA scenario.

dataset without collaboration. Centralized training is training all datasets centrally.

Implementation Details

In all experiments, we use LLaVA 1.5 (Liu et al. 2024) as multimodal Large Language Model. LLaVA utilizes the pre-trained CLIP visual encoder ViT-L/14 (Radford et al. 2021) to extract visual features, and LLM utilizes Vicuna-V1.5-7b (Chiang et al. 2023). The client-specific adapter and task-specific adapter parameters in stage 1 are initialized by the pre-trained MLP connector parameters in LLaVA. All cross-task adapter parameters in stage 2 are initialized by the local task-specific adapter obtained in stage 1. For all client text-adapter, we use the LoRA (Hu et al. 2021) parameter efficient tuning technique to train LLM. The rank of LoRA is 64 with a scalar $\alpha = 128$. During instruction tuning, we only fine-tune the parameters of the connector and LoRA while keeping the rest of the LLM frozen. The learning rates of stage 1 and stage 2 are set to 2e-5 and 4e-5, respectively. The number of text-adapter parameter aggregates Top-M is set to 6 The coefficients of λ_0, λ_1, and λ_2 are 0.1, 0.1, and 0.01, respectively. The number of local cycles E is set to 1 and the number of communication rounds R is 3. We train Pilot on 8 A100 GPUs (40G), with an effective batch size of 16 per GPU. To ensure fairness, for all baselines, all additional hyperparameters involved in the compared methods use their best settings.

Experimental Results

For the VQA task (including ScienceQA, GQA, and OCRVQA), we calculate the accuracy of predicting answers against ground truth. For the caption task, we report the CIDEr score. For the grounding task, we employ Intersection-over-Union (loU) as the evaluation metric. Specifically, a prediction is deemed accurate only when its loU exceeds or equals 0.5. Table 1 and Table 2 show the comparison of Pilot with other five methods in the federated scenarios of visual understanding and General VQA. Our framework outperforms all baselines in two task scenarios. We found that the performance of FedAVG method is lower than local training. This shows that the heterogeneity of multimodal tasks leads to greater parameter conflicts, and simple aggregation has a negative impact on local models. At the same time, we observed that FedAvg method is better than Shepherd. The former aggregates all training parameters, while the latter only aggregates LLM training parameters. This result also indirectly reflects the importance of visual information to the FedMIT task. Compared with FedDPA, only improving the LLM training parameters does not achieve the desired effect, which also illustrates the need to consider the differences between different tasks and the necessity of learning general knowledge from other tasks. Finally, the results prove that our method can not only overcome the heterogeneity between tasks, but also collaborate with all clients to improve the performance of local models.

Ablation Studies

Here, we show the results of several variants of our method in the FL-oriented visual understanding scenario to demonstrate the effectiveness of the main modules of our method. We first analyze the impact of the four components of our framework (i.e., cross-task adapter, difference loss, auxiliary loss, and adaptive text-adapter aggregation (ATA)) on the model performance, and further evaluate the effectiveness of the proposed method. Table 3 shows the average score of all clients for the same task. We found that without us-

ATA	ψ^c	Loss		GQA	COCO	RefCOCO
		$\mathcal{L}_b + \mathcal{L}_z$	$\mathcal{L}_d$			
×	×	×	×	46.9	105.7	30.6
✓	×	×	×	47.5	110.6	39.8
✓	✓	×	×	49.3	114.5	47.5
✓	✓	✓	×	49.8	120.7	48.2
✓	✓	✓	✓	**50.4**	**124.0**	**51.0**

Table 3: Ablation studies under FL-oriented visual understanding scenario.

Methods	GQA	COCO	RefCOCO	AP	CP
Pilot	50.4	124.0	51.0	**0.5B**	**0.3B**
+ CT-CLIP	**51.6**	**126.1**	**52.7**	1.75B	0.5B

Table 4: Impact of vision encoder improvements on model performance.

ing the above methods, the performance of our method is lower than that of local training, which does not have the ability to overcome task heterogeneity. Then we add adaptive text-adapter aggregation, and the performance of our method is improved and outperforms local training, which demonstrates that this module can effectively alleviate the impact of task heterogeneity. When we add the cross-task adapter, we observe that the model performance improves on all clients. Through auxiliary loss optimization, the performance of our framework can be further improved. The results show that the CT-MOA module is able to learn general knowledge from other tasks to improve model performance and cross-task capabilities. Removing the difference loss, our connector has only one and no longer distinguishes between client-specific adapter and task-specific adapter. The performance of the model has decreased, indicating that it is necessary to maintain the personalized information of the client. The above results demonstrate the importance of each component in our method.

Further Remarks

Building Cross-task CLIP. Our method modifies the MLLM connector to learn general knowledge from other tasks to improve model performance and cross-task capabilities. But the natural thought is: why not use CLIP? To answer this question, we unfreeze CLIP and train all MLP layers in CLIP in the same way as the connector, called CT-CLIP. We conduct experiments on the visual understanding scenario for federated learning. Table 4 shows the average scores of all clients for the same task, where AP denotes local model Activation Params and CP denotes Communication Params. Experimental results show that for the FedMIT task, learning cross-task visual information from different tasks is an effective solution. We observe that although CT-CLIP outperforms the Pilot, it comes at the expense of additional training parameters and communication parameters. Compared with Pilot, the communication parameters sent to the server increase by 0.2B, and the local client activation parameters increase by 1.25B. With the increase of tasks, the computational cost is unacceptable. Therefore, we give priority to the more simple and efficient method.

Selection Strategy	GQA	COCO	RefCOCO
Same task client	49.6	120.2	50.3
All clients	47.9	109.7	45.2
Pilot (M=5)	49.7	122.8	50.6
Pilot (M=6)	**50.4**	**124.0**	51.0
Pilot (M=7)	50.3	123.2	**51.7**

Table 5: Compare different text-adapter parameter aggregation strategies.

Initialization Strategy	GQA	COCO	RefCOCO
Random	50.1	122.7	50.3
Pilot	**50.4**	**124.0**	**51.0**

Table 6: Compare different cross-task adapter parameters initialization strategies.

Different Text-adapter Parameter Aggregation Strategies. In our framework, we adopt the adaptive text-adapter aggregation method. For all text adapter parameters, we adopt a weighted optimization aggregation strategy based on euclidean distance and select the Top-M parameters for weighted averaging. We compared different aggregation strategies, such as aggregating only parameters of clients with the same task (same task client), aggregating parameters of all clients (all clients), and different Top-M selections. As shown in Table 5, due to the heterogeneity between tasks, aggregating on all client will lead to parameters conflicts. In addition, aggregating only on the same task is a suboptimal solution because it does not utilize the semantic knowledge of other tasks. At the same time, we tested the impact of different top-M on model performance, and the results showed that the adopted adaptive aggregation method can benefit from positive influences while reducing negative interference.

Cross-task Adapter Initialization Strategy. In our framework, the role of the cross-task adapter is to extract cross-task knowledge. We compared initialization with task-specific adapter parameters with training from scratch (random). As shown in Table 6, training from scratch leads to performance degradation for all tasks. Using task-specific adapter parameters initialization provides a good starting point for the module and can help the local client better extract cross-task knowledge.

Conclusion

In this paper, we propose a federated multimodal instruction tuning framework to solve the new task of federated multimodal instruction tuning by collaboratively utilizing distributed data from different local clients to learn cross-task knowledge without being affected by task heterogeneity during instruction tuning. Through two stages “adapter on adapter” strategy, our model can capture the personalized information of local data and the task-related multimodal information, and can also adapt to the differences between different tasks. Our method achieves state-of-the-art results in two cross-task scenarios.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grants 62036012, U23A20387, 62322212, 62072455, in part by Pengcheng Laboratory Research Projectunder Grant PCL2023A08, in part by Alibaba Innovative Research Program, and also in part by CAS Projectfor Young Scientists in Basic Research(YSBR-116).

References

Arevalo, C. A.; Noorbakhsh, S. L.; Dong, Y.; Hong, Y.; and Wang, B. 2024. Task-Agnostic Privacy-Preserving Representation Learning for Federated Learning against Attribute Inference Attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 10909–10917.

Bousmalis, K.; Trigeorgis, G.; Silberman, N.; Krishnan, D.; and Erhan, D. 2016. Domain separation networks. *Advances in neural information processing systems*, 29.

Chen, D.; Liu, J.; Dai, W.; and Wang, B. 2024a. Visual instruction tuning with polite flamingo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 17745–17753.

Chen, H.; Zhang, Y.; Krompass, D.; Gu, J.; and Tresp, V. 2024b. Feddat: An approach for foundation model finetuning in multi-modal heterogeneous federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 11285–11293.

Chen, J.; and Zhang, A. 2024. On Disentanglement of Asymmetrical Knowledge Transfer for Modality-Task Agnostic Federated Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 11311–11319.

Chen, K.; Zhang, Z.; Zeng, W.; Zhang, R.; Zhu, F.; and Zhao, R. 2023. Shikra: Unleashing multimodal llm's referential dialogue magic. *arXiv preprint arXiv:2306.15195*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3): 6.

Dai, W.; Li, J.; et al. 2023. InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning. arXiv:2305.06500.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Driess, D.; Xia, F.; Sajjadi, M. S.; et al. 2023. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*.

Fallah, A.; Mokhtari, A.; and Ozdaglar, A. 2020. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*.

Fan, T.; Kang, Y.; Ma, G.; Chen, W.; Wei, W.; Fan, L.; and Yang, Q. 2023. Fate-llm: A industrial grade federated learning framework for large language models. *arXiv preprint arXiv:2310.10049*.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, 1126–1135. PMLR.

Gorbunov, E.; Hanzely, F.; and Richtárik, P. 2021. Local sgd: Unified theory and new efficient methods. In *International Conference on Artificial Intelligence and Statistics*, 3556–3564. PMLR.

Guo, T.; et al. 2023. Pfedprompt: Learning personalized prompt for vision-language models in federated learning. In *Proceedings of the ACM Web Conference 2023*, 1364–1374.

He, J.; Guo, H.; Tang, M.; and Wang, J. 2023. Continual instruction tuning for large multimodal models. *arXiv preprint arXiv:2311.16206*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hudson. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 6700–6709.

Jia, Y.; Zhang, X.; Beheshti, A.; and Dou, W. 2024. FedLPS: Heterogeneous Federated Learning for Multiple Tasks with Local Parameter Sharing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 12848–12856.

Kazemzadeh, S.; Ordonez, V.; Matten, M.; and Berg, T. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 787–798.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2: 429–450.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35: 2507–2521.

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*.

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.

Mishra, A.; Shekhar, S.; Singh, A. K.; and Chakraborty, A. 2019. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, 947–952. IEEE.

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*.

Shen, Y.; Xu, Z.; Wang, Q.; Cheng, Y.; Yin, W.; and Huang, L. 2024. Multimodal Instruction Tuning with Conditional Mixture of LoRA. *arXiv preprint arXiv:2402.15896*.

Shi, J.; Zheng, S.; Yin, X.; Lu, Y.; Xie, Y.; and Qu, Y. 2023. Clip-guided federated learning on heterogeneous and long-tailed data. *arXiv preprint arXiv:2312.08648*.

Su, S.; Yang, M.; Li, B.; and Xue, X. 2024. Federated adaptive prompt tuning for multi-domain collaborative learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 15117–15125.

Sun, L.; Zhang, K.; Li, Q.; and Lou, R. 2024. Umie: Unified multimodal information extraction with instruction tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 19062–19070.

Wei, J.; Bosma, M.; Zhao, V. Y.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Xiao, L.; Yang, X.; Peng, F.; Wang, Y.; and Xu, C. 2024a. Hivg: Hierarchical multimodal fine-grained modulation for visual grounding. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 5460–5469.

Xiao, L.; Yang, X.; Peng, F.; Wang, Y.; and Xu, C. 2024b. OneRef: Unified One-tower Expression Grounding and Segmentation with Mask Referring Modeling. *arXiv preprint arXiv:2410.08021*.

Xiong, B.; Yang, X.; Song, Y.; Wang, Y.; and Xu, C. 2023. Client-Adaptive Cross-Model Reconstruction Network for Modality-Incomplete Multimodal Federated Learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, 1241–1249.

Xiong, B.; Yang, X.; Song, Y.; Wang, Y.; and Xu, C. 2024. Modality-Collaborative Test-Time Adaptation for Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26732–26741.

Xu, Y.; Yang, X.; Song, Y.; and Xu, C. 2024. Libra: Building Decoupled Vision System on Large Language Models. *arXiv preprint arXiv:2405.10140*.

Xu, Z.; et al. 2022. Multiinstruct: Improving multi-modal zero-shot learning via instruction tuning. *arXiv preprint arXiv:2212.10773*.

Yang, M.; Su, S.; Li, B.; and Xue, X. 2024a. Exploring One-Shot Semi-supervised Federated Learning with Pre-trained Diffusion Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16325–16333.

Yang, X.; Xiong, B.; Huang, Y.; and Xu, C. 2024b. Cross-Modal Federated Human Activity Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yang, Y.; Long, G.; Shen, T.; Jiang, J.; and Blumenstein, M. 2024c. Dual-Personalizing Adapter for Federated Foundation Models. *arXiv preprint arXiv:2403.19211*.

Ye, Q.; Xu, H.; Xu, G.; Ye, J.; Yan, M.; Zhou, Y.; Wang, J.; Hu, A.; Shi, P.; Shi, Y.; et al. 2023. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Ye, R.; Wang, W.; Chai, J.; Li, D.; Li, Z.; Xu, Y.; Du, Y.; Wang, Y.; and Chen, S. 2024. Openfedllm: Training large language models on decentralized private data via federated learning. *arXiv preprint arXiv:2402.06954*.

Zhang, J.; Liu, Y.; Hua, Y.; and Cao, J. 2024a. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 16768–16776.

Zhang, J.; Vahidian, S.; Kuo, M.; Li, C.; Zhang, R.; Yu, T.; Wang, G.; and Chen, Y. 2024b. Towards building the federatedGPT: Federated instruction tuning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6915–6919. IEEE.

Zhu, D.; Chen, J.; Shen, X.; Li, X.; and Elhoseiny, M. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Zoph, B.; Bello, I.; Kumar, S.; Du, N.; Huang, Y.; Dean, J.; Shazeer, N.; and Fedus, W. 2022. St-moe: Designing stable and transferable sparse expert models. *arXiv preprint arXiv:2202.08906*.