FedMCP: Parameter-Efficient Federated Learning with Model-Contrastive Personalization

Anonymous ACL submission

Abstract

Given the growing concerns over data privacy and security, fine-tuning pre-trained language models (PLMs) in federated learning (FL) has become the standard practice. However, this process faces two primary challenges. Firstly, the utilization of large-scale PLMs introduces excessive communication overheads. Secondly, the data heterogeneity across FL clients presents a major obstacle in achieving the desired fine-tuning performance.To address these challenges, we present a parameter-efficient fine-tuning (PEFT) method with **M**odel-**C**ontrastive **P**ersonalization (FedMCP). This approach introduces two adapter modules to the frozen PLM and only aggregates the global adapter in the federated aggregate phase while the private adapter stays in clients. The model-contrastive regularization term and aggregation strategy encourage the global adapter to learn universal knowledge from all clients and the private adapter to capture idiosyncratic knowledge for each individual client. Verified across a highly heterogeneous cross-silo dataset, the empirical evaluation shows considerable performance improvement achieved by FedMCP over state-of-the-art approaches.

1 Introduction

Pre-trained language models (PLMs) have gained considerable significance across a wide range of natural language processing (NLP) tasks. Typically, fine-tuning PLMs on specific datasets is essential to ensure optimal performance for downstream tasks in the real world. However, these datasets are often scattered across different entities (Qu et al., 2021). Due to the increasing privacy concerns and regulatory laws, these entities are unwilling to share their private datasets for fine-tuning PLMs. For instance, Rieke et al. showed that data silos are prevalent, particularly in the healthcare domain, where patient information is critical for training diagnostic or treatment recommendation models but is often isolated within healthcare institutions (Rieke et al., 2020). To address the above problem, federated learning (FL) (Konečnỳ et al., 2016; McMahan et al., 2017) has emerged as a promising solution by allowing multiple clients to collaboratively train PLMs without the need to expose their local private datasets (Lin et al., 2021).

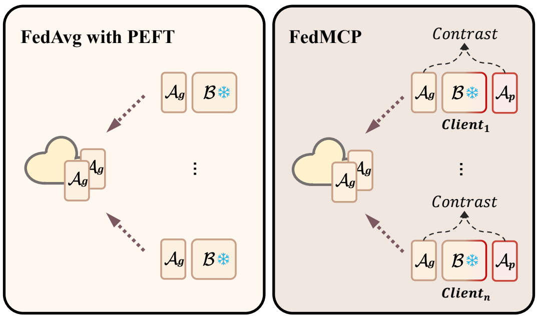


Figure 1: The conceptual illustration of FedAvg (the left hand) and FedMCP (the right hand). A and B refer to the adapter and the backbone respectively. The snowflake icon indicates that the backbone is frozen, while the other modules are trainable.

One of the issues of FL is the limited communication bandwidth and client-side computing resources. The practice of FL involves regular model exchanges between the server and clients during training, which leads to high communication overheads. Furthermore, given the limited computing resources on the client side, fine-tuning the entire PLMs can be impractical (Zhang et al., 2023). This poses a barrier to the deployment of large-scale models like BERT (Devlin et al., 2018), GPT-2 (Radford et al., 2019a) and T5 (Raffel et al., 2020) in FL settings (Wu et al., 2022). We address this issue by applying parameter efficiency approach to FL.

Another issue is that the global model suffers from data heterogeneity. In FL, a globally shared model is trained over decentralized data, e.g., often through methods such as FedAvg. However, due to the inherent diversity of clients (Huang et al., 2021a), known as the non-IID (non-identically distributed) problem, the global model may not be optimal for each client. The common strategy for mitigating the non-IID problem is *model personalization* (Tan et al., 2022), which refers to the process of tailoring the local model based on the global model to fit the specific needs and characteristics of individual clients.

Existing works on personalization primarily focus on addressing the non-IID scenario, characterized by varying label distributions among clients. In parallel, the heterogeneity of data across different clients presents another significant challenge (Ye et al., 2023). For instance, different organizations hold textual data in various areas like question answering, personal blogs, and emails, resulting in a distinct focus of their data on different tasks, thereby introducing data heterogeneity.

To address the challenges of limited communication bandwidth and the data heterogeneity of FL, we propose a novel personalized FL method with **M**odel-**C**ontrastive **P**ersonalization (FedMCP), aiming to effectively fine-tune PLMs and mitigate the data heterogeneity across NLU tasks under the cross-silo FL setting. Personalized federated learning primarily adopts two strategies: (1) training a global model and then personalizing it by local adaptation steps; (2) customizing a personalized model for each client by modifying the global model aggregation process (Tan et al., 2022). Our method leverages the global model to learn universal knowledge, while also retaining parts of the local model to achieve personalization.

In this paper, we apply two adapter modules (Houlsby et al., 2019) to the backbone PLM (referred to as the backbone) for personalization, a global adapter for aggregation, facilitating the collaboration and knowledge sharing among clients, and a private adapter designed to be retained locally on each client, enabling the learning of client-specific knowledge. Here, we propose a novel model-contrastive personalization loss that is tailored to the FL parameter-efficient fine-tuning (PEFT) method. The loss leverages Centered Kernel Alignment (CKA) to quantify similarities between models. By minimizing the distance between the client's global adapter and the average global adapter, and maximizing the distance between the client's global adapter and private adapter, the model's ability to achieve a balance between generalization and personalization. Through this contrastive loss, we can differentiate the roles of the two adapters to make the global adapter learn global knowledge that benefits all clients, and the private adapter captures the unique knowledge of each client. Figure 1 shows the conceptual illustration of the widely adopted FedAvg and our proposed FedMCP.

We utilized six datasets from the GLUE benchmark (Radford et al., 2019b) to simulate NLU cross-silo scenarios. The empirical results demonstrate that our proposed FedMCP outperforms the existing state-of-the-art personalized FL methods with the same settings (with the backbone frozen). Moreover, FedMCP in PEFT achieves comparable results to full fine-tuning with reduced communication costs. Our contributions are three-folded:

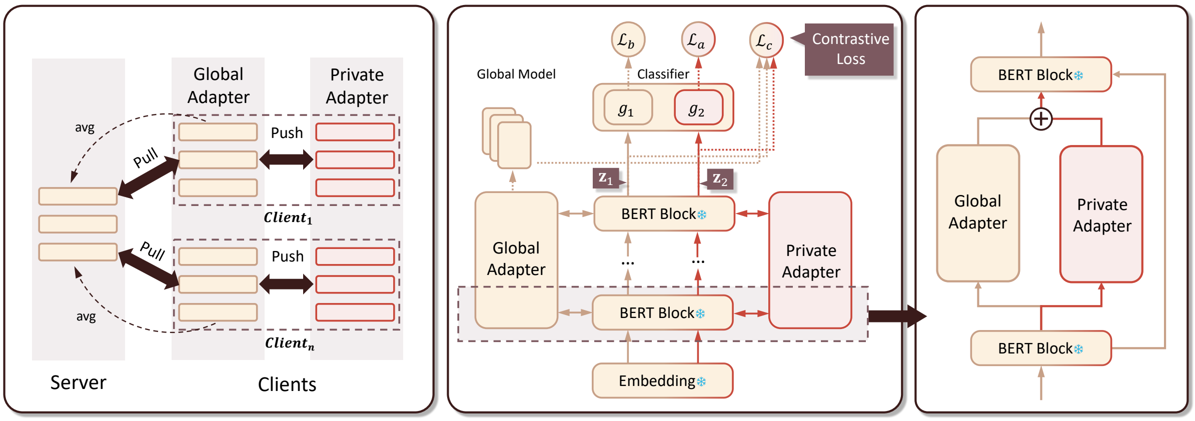
- We propose FedMCP, a novel parameter-efficient personalized FL method that mitigates the data heterogeneity across NLU tasks in PLMs fine-tuning.
- We compose a dataset of cross-silo FL in NLU to evaluate model performance across different tasks.
- We conducted extensive experiments on the composed dataset and demonstrated that FedMCP outperforms the current state-of-the-art baselines for PEFT in FL.

2 Preliminary

2.1 FL for Text Classification Task

In this paper, our focus is on text classification tasks, following previous studies (Xu et al., 2023, Luo et al., 2021), where the model can typically be decomposed into an encoder and a task-specific classifier. Consider a supervised setting in which the i-th client is equipped with data distribution $P^i_{\mathcal{X}\mathcal{Y}}$ on $\mathcal{X} \times \mathcal{Y}$. Given a sample (x, y), the feature extractor $f_\theta : \mathcal{X} \to \mathcal{Z}$ (parameterized by θ) maps the input x to a feature vector $z = f_\theta(x) \in \mathbb{R}^d$ in the feature space $\mathcal{Z}$. Subsequently, the classifier $g_\phi : \mathcal{X} \to \mathcal{Y}$ (parameterized by ϕ) maps the feature z to predict the label $g_\phi(z) \in \mathcal{Y}$. The parameters of the classification model are represented by $w = (\theta, \phi)$.

In the FL round t, the server broadcasts the current model parameters $w^{(t-1)}$ to all clients. Each client then locally optimizes the following objec-



(a) Model - Contrastive Method

(b) Model Structure

(c) Organizational structure of the two adapters

Figure 2: Overview of the proposed parameter-effecient FedMCP method. (a) Clients upload the global adapter to the Server and keep the private adapter locally. The model-contrasive loss is calculated using the average global adapter, the local global adapter, and the local private adapter. (b) Overview of the model structure, three losses are generated during training to achieve personalization. (c) The organizational structure of the two adapters within the backbone.

tive to obtain $w_k^{(t)}$:

$$\min_{w_i^{(t)}} \mathbb{E}_{(x,y)\sim P_{XY}^{(i)}} \left[\mathcal{L}\left(w_k^{(t)}; w^{(t-1)}, x, y\right)\right] \tag{1}$$

where $\mathcal{L}$ is the loss function.

2.2 Adapter

The adapter enhances existing pre-trained models by introducing additional parameters (Houlsby et al., 2019). These parameters are added after the attention and feed-forward-network layers of the Transformer, organized in the form of a fully connected network. This structure allows the adapter to demonstrate remarkable parameter efficiency, and often achieving performance comparable to Full fine-tuning by updating only a small portion of parameters during fine-tuning.

For a given input s, the down-projection layer W_{down} of the adapter layer projects s to a low-dimensional space of dimension r. Subsequently, a non-linear activation function $g(\cdot)$, such as ReLU, is applied. The vector is then mapped back to the hidden layer size of dimension h through an up-projection, and the computation process of the adapter can be represented as follows:

$$s \leftarrow s + g(sW_{\text{down}})W_{\text{up}} \tag{2}$$

We incorporate adapters into the model, employing specific learning strategies to enable them to learn client-specific knowledge from each client.

3 Method

In this section, we elaborate on the proposed method FedMCP. By integrating a global adapter for aggregation and a local private adapter, FedMCP enables each client to learn both universal knowledge and unique local knowledge to each client. During the training process, a model-contrastive method is used to decrease the distance between the client's global adapter and the average global adapter, while increasing the distance between the client's global adapter and its private adapter. This approach not only minimizes model drift between the client and the global model but also enhances the private adapter's ability to acquire client-specific knowledge, achieving personalization for client models.

3.1 Model Architecture

As shown in the right part of Figure 2(b)(c), the full model consists of a backbone and two additional adapter modules added to the backbone.

For a given input, the model has two forward propagations: one through the full model with two adapters (the **red line**), and the other through the backbone without the private adapter (the **brown line**). We denote the full model as f_1, and the backbone with global adapter as f_2, they generate representations $\mathbf{z}_1 = f_1(x)$ and $\mathbf{z}_2 = f_2(x)$ respectively. After encoding, the sequence representations are input into MLP classifiers g_1 and g_2 to obtain classification results.

Figure 2(c) illustrates that the two adapters are inserted at the same position to the backbone, taking the output from the previous layer and feeding it into the two adapters. The outputs of these adapters are then averaged and used as the input for the next layer.

3.2 Contrastive Personalization

The client-side loss function during training comprises three components: the cross-entropy loss of the full model, the cross-entropy loss of the backbone with the global adapter and the contrastive loss between two adapters.

Notably, we introduce the two additional losses mentioned above with the following key considerations:

- **Distinguishing Local and Global Knowledge:** We aim for the client-side model to effectively distinguish local specific knowledge and shared knowledge. This distinction primarily stems from the private adapter's adaptability to local knowledge.
- **Enhancing the Representation Power of the Shared Global Adapter:** We seek to improve the learning ability of the shared adapter. It is desirable that the sole global adapter can learn generic knowledge that benefits every client.

The interplay of these three losses facilitates the local model in capturing both client-specific knowledge and global knowledge shared across clients.

3.2.1 Model-Contrastive Method

Initially introduced by Li et al., 2021a, the MOON method focuses on model-level contrast to reduce the differences between local and global models in FL, aiming to mitigate model drift in non-IID data scenarios. However, this approach trains a single averaged global model, which lacks personalization and impairs the performance of the global model on individual clients at heterogeneous local data distributions.

For personalization within the PEFT framework, beyond the global module's aggregation, client-specific customization is also essential. So FedMCP aims to decrease the distance between the representation learned by the local global adapter and the average global adapter, and increase the distance between the representation learned by the local global adapter and the local private adapter. Figure 2(a) describes the model-contrastive process.

The distance can be measured by various similarity metrics. Building upon the research conducted by Kornblith et al., 2019, we employ Central Kernel Alignment (CKA) to quantify the distance between the output representations of the average global adapter, the local global adapter and the private adapter. CKA assigns a similarity value to feature structure by comparing the representations trained on different model architectures. Its score is higher and more consistent than other similarity metrics like cosine similarity (Kornblith et al., 2019; Jung et al., 2023). We examine the effectiveness of using CKA as a similarity metric in ablation experiments.

We denote d as the sentence length, h as the model's hidden layer size, and n as the batch size. For an input x, the encoder produces an output $\mathbf{z} \in \mathbb{R}^{n \times d \times h}$. By taking the average pooling of all tokens from the encoder's last layer as the vector representation of the sequence, the local global adapter, local private adapter and shared average global adapter generate three matrices $X \in \mathbb{R}^{n \times h}$, $Y \in \mathbb{R}^{n \times h}$ and $Z \in \mathbb{R}^{n \times h}$.

The CKA similarity metric takes values within the range [0, 1], where 0 indicates dissimilarity, and 1 indicates complete similarity. The CKA distance between the two models is represented as:

$$CKA(X,Y) = \frac{HSIC(K,L)}{\sqrt{HSIC(K,K)HSIC(L,L)}} \qquad (3)$$

$$HSIC(K,L) = \frac{1}{N-1^2} tr(KCLC) \qquad (4)$$

where $K = XX^T$, $L = YY^T$ and $HSIC(\cdot,\cdot)$ is the HilbertSchmidt Independence Criterion (HSIC) values, tr is a trace in a matrix, C is a centering matrix $C_n = I_n - \frac{1}{n} J_n$ (Jung et al., 2023). Throughout the training process, we aim to increase the distance between the clients' global adapter and clients' private adapter, and decrease the distance between the average global adapter and clients' global adapter. Therefore, we aim to reduce the CKA value of the former and increase the CKA value of the latter. The contrastive loss during training is expressed as:

$$\mathcal{L}_c = CKA(X,Y) - CKA(Y,Z) \qquad (5)$$

3.2.2 Learning Effective Global Adapter

While the contrastive loss in the previous section aims to balance the model's abilities between generalization and personalization, we introduce a cross-entropy loss based on the global adapter. This serves as a regularization mechanism and enables the global adapter to improve its learning of generalizable knowledge from local data.

The definition of the backbone with the global adapter's cross-entropy loss is:

$$\mathcal{L}_b((\theta_b^t, \phi_2); (x, y)) = \ell((f_2 \circ g_2)(x); y) \qquad (6)$$

Where ℓ is the cross-entropy loss. The backbone with two adapter's parameters are represented as θ_a, and the backbone with global adapter's parameters θ_b are a subset of θ_a, can be expressed as $\theta_b \subseteq \theta_a$; ϕ_1 and ϕ_2 denote the classifiers' parameters for the full model and backbone with global adapter respectively.

3.3 Local Training and Global Aggregation

The process of client local training and server global aggregation is summarized in Algorithm 1.

Overall Objective. Defining the cross-entropy loss of the local full model with the two adapters as $\mathcal{L}_a$, which is formulated as:

$$\mathcal{L}_a((\theta_a^t, \phi_1); (x, y)) = \ell((f_1 \circ g_1)(x); y) \qquad (7)$$

The overall objective for client i-th client during the t-th round of FL is expressed as:

$$\mathcal{L} = (1-\gamma)\mathcal{L}_a + \gamma\mathcal{L}_b + \mu\mathcal{L}_c \qquad (8)$$

The parameters of the client include the full model parameter θ_a and two classifiers parameters ϕ_1, ϕ_2. The parameters of the backbone remain fixed throughout the training period. In the t-th round, all parameters update as follows:

$$(\theta_a, \phi_1, \phi_2) \leftarrow (\theta_a, \phi_1, \phi_2) - \eta\nabla\mathcal{L}((\theta_a, \phi_1, \phi_2)) \qquad (9)$$

Global Aggregation In the aggregation phase, each client sends the global adapter's parameter to the server only to update the global adapter.

4 Experiment

In this section, we conduct extensive experiments to examine the performance of FedMCP.

4.1 Cross-silo Data Construction

Similar to FedPETuning, we select six datasets from the GLUE benchmark (Radford et al., 2019b), namely RTE, MRPC, SST-2, QNLI, QQP and MNLI. These NLU datasets are widely used in evaluating the performance of natural language processing models. Our selection covers tasks like classification (e.g., sentiment classification in SST-2), sentence similarity judgment (MRPC, QQP), and semantic inference tasks (QNLI, MNLI).

```
Algorithm 1 FedMCP
Input: $T$ is the communication round; $E$ is the number of local epochs; $\eta$ is the learning rate
Server executes:
Initialize prototype sets $\{\boldsymbol{C}_p\}_{p=1}^m$.
for each round $t = 1$ to $T$ do
    for each client $i$ in parallel do
        LocalUpdate($i$, $W_{ag}^{t-1}$, $W_{ap}^{t-1}$)
    end for
    Recieve local update parameters $W_{ag}^{t-1}$
    $W_{ag}^t = \sum_{k=1}^K \frac{1}{K} W_{ag}^{k,t-1}$
end for
LocalUpdate:($i$, $W_{ag}^t$, $W_{ap}^t$):
for each local epoch do
    Compute $\mathcal{L}$ by Eq. (8).
    Update $W_{ag}^t$, $W_{ap}^t$ by Eq. (9).
end for
Send $W_{ag}^{t+1}$ to the server
```

Our research marks the first attempt to establish a federated cross-silo setting across different natural language understanding tasks. Unlike previous studies, we adopt a cross-silo division, treating each of the six datasets as an independent client and ensuring the privacy of each client's data during training.

Data Size Balancing. There are significant size differences among these six datasets, with the smallest RTE having less than 3,000 entries and the largest MNLI having over 400,000. To avoid the large datasets dominating the process of model training, for datasets larger than MRPC, we resized them to match MRPC's scale by random sampling.

Partitioning. As GLUE does not release test sets, we merge the existing training and validation sets, partitioning each client into training, validation, and test sets in a 6:2:2 ratio. This dataset will be made available to encourage research in cross-silo cross-task federated NLU.

4.2 Baselines

In the PEFT scenario, all baseline methods only fine-tune the added adapter modules. The model architecture across all methods remains consistent

Methods	MRPC	RTE	SST-2	QNLI	QQP	MNLI	Avg.	Para.(%)	Com.(%)
Full FT FedAvg	84.79±1.29	77.46±1.50	92.64±0.50	88.4±0.57	82.17±1.23	73.94±1.13	83.24±0.22	100%	100%
local	87.42±0.29	77.46±0.83	**93.63**±1.77	87.09±1.58	82.51±1.50	73.37±0.28	83.58±0.22	-	-
FedAvg	87.09±2.47	78.66±1.81	93.30±0.57	84.64±1.98	83.66±1.41	74.35±1.58	83.62±0.34	0.58%	0.58%
FedLR	85.13±1.02	74.58±4.68	92.49±1.02	**88.40**±1.24	80.55±3.68	72.39±1.98	82.26±0.95	0.29%	0.29%
FedAP	86.60±2.70	77.70±1.25	93.47±1.23	85.62±1.23	81.37±1.77	73.53±2.14	83.05±0.39	0.29%	0.29%
MOON	86.60±0.75	78.90±0.83	92.65±1.77	85.62±0.75	81.53±1.23	73.20±1.02	83.08±0.86	1.16%	1.16%
FedRep	85.78±0.49	79.14±1.90	92.65±0.85	84.96±2.32	81.37±2.14	75.82±1.98	83.29±1.11	1.16%	1.16%
FedMatch	87.09±0.84	76.02±0.81	93.79±1.73	86.11±1.26	83.33±0.82	75.33±1.25	83.61±0.71	1.16%	1.16%
FedMCP (ours)	**87.69**±0.83	**80.58**±1.65	93.52±0.68	86.54±1.16	**83.77**±2.17	**76.52**±1.98	**84.77**±0.60	1.16%	0.58%

Table 1: The performance of FedMCP and baselines on corss-silo datasets under PEFT settings. The average and standard deviation of accuracy(%) are computed over three times. **Bold** and underline indicate best and second-best results, respectively. The *Para.* denotes the percentage of trainable parameters relative to Full FT. The *Com.* denotes the percentage of communication overhead relative to Full FT.

with FedMCP. We compare FedMCP with the following baselines:

(1) **Local-only**, each client training locally without exchanging gradients with the server; (2) **FedAvg** (McMahan et al., 2017), all clients train a single global model by averaging the gradients from all clients in each round; (3) **Full Fine-Tuning (Full FT) FedAvg**, all clients train a single global model and the whole model parameters are updated and aggregated. (4) Representative PEFT methods, including Adapter (Houlsby et al., 2019) (**FedAP**) and LoRA (Hu et al., 2021) (**FedLR**). (5) **MOON** (Li et al., 2021a), a method that learns a global model, adopts the contrastive loss to minimize the distance between the representations learned by the local models and the global model. (6) Personalized FL methods, including **FedRep** (Collins et al., 2021) and **FedMatch** (Chen et al., 2021). FedRep achieves personalization by training the classifier multiple times before updating the local model. In FedMatch, the authors proposed four methods of adding private patches, we adopted the Houlsby Adapter, which is the most effective patch insertion method on our dataset.

4.3 Experiments Setup

Hyperparameter Settings. We search learning rates from {1e-3, 5e-4, 1e-4, 5e-5} and eventually set it 5e-4. For the backbone loss and contrastive loss, we adjusted coefficients γ and μ, the optimal hyperparameters were determined to be 0.5 and 0.05, and we report results with these best-performing parameters.

Other Implementation Details. The experiments use RoBERTa-Base as the model backbone provided by Huggingface[1], this choice was inspired by the research of Zhang et al., 2023, our code is based on the FedAvg implementation[2] of FedLab (Zeng et al., 2023). To accommodate different task characteristics, we opted not to share classifier parameters across tasks (Collins et al., 2021). All six clients participate in training during 25 communication rounds and each client trains one epoch per round. Furthermore, the bottleneck size of adapters was set to 16. All experiments were conducted on a Tesla V100 GPU with 32G memory, using Adam as the optimizer and a batch size of 64.

4.4 Results

Table 1 reports the performance of different methods under the federated cross-silo settings. FedMCP attains the highest average accuracy across clients and achieves the best or nearly the best accuracy on each client.

Performance Comparison. Initially, FedAvg outperforms both FedAP and FedLR due to an increase in trainable parameters. Moreover, compared to local training, FL algorithms without personalized adaptation show inferior performance in cross-silo scenarios, suggesting that personalized algorithms can mitigate the issues of data heterogeneity in FL, and help clients learn models more appropriate for themselves. Furthermore, our framework is the best among all FL approaches, which indicates that FedMCP can adapt to the differences between various tasks and text corpus domains in FL, exploiting global and private knowledge to effectively personalize for each client. It is also observed that FedMCP surpasses the perfor-

[1] https://github.com/huggingface/transformers

[2] https://github.com/SMILELab-FL/FedPETuning

similarity metric	MRPC	RTE	SST-2	QNLI	QQP	MNLI	Avg.
Cosine	86.11±0.57	79.42±1.16	93.3±1.13	86.27±1.29	82.35±0.49	75.65±1.13	83.85±0.14
CKA	**87.69**±0.83	**80.58**±1.65	**93.52**±0.68	**86.54**±1.16	**83.77**±2.17	**76.52**±1.98	**84.77**±0.60

Table 2: Ablation study for using cosine and CKA as similarity metrics. The average and standard deviation of accuracy(%) are computed over three times.

mance of complete FL fine-tuning while only fine-tuning 0.58% of the parameters. The performance of other methods is not significantly different from federated global fine-tuning, indicating that model generalizability is constrained by data heterogeneity in cross-silo scenarios, and FedMCP plays a positive role in overcoming these limitations.

Efficiency Comparison. Except for FedAP and FedLR, all methods have identical model structures with FedMCP. However, during the federated communication rounds, FedMCP only exchanges to a single adapter module between the clients and the server. FedMCP achieves optimal performance with enhanced communication efficiency compared to other methods.

4.5 Ablation Studies

There are two key components in FedMCP, i.e., backbone loss (BL) and contrastive loss (CL). The BL helps to learn effective and independent global adapter, and the CL makes the local adapter to learn local private knowledge. Here we provide further discussions to get a better understanding of each module from the loss function's components and the similarity metric of CL.

Loss Function. The loss function is defined in Eq. (8). Table 3 illustrates the mean and variance of test accuracies for the six clients. From these results, we can observe that the overall performance of "w/o BL" and "w/o CL" drops 0.64% and 1.27% compared to FedMCP, respectively, which confirms their contributions to the proposed framework.

CL plays a critical role in enhancing model performance, demonstrating that the incorporation of a model contrastive loss enables better differentiation between global and local knowledge when one adapter is aggregated and another remains local for personalized learning. Additionally, the introduction of BL improves the global adapter's ability to encode global knowledge, facilitating the transfer of useful knowledge among clients.

Methods	Avg.	SD
w/o CL	83.57	0.68
w/o BL	84.13	0.73
FedMCP	**84.77**	0.60

Table 3: Ablation study for loss function. BL and Cl represent the backbone loss and contrastive loss, respectively. The average and standard deviation (SD) are calculated from the individual means and standard deviations of six clients across three experiments.

Similarity Metric. We compare the performance of CKA and cosine similarity to measure the quality of similarity metrics in FedMCP. Table 2 presents the results of two similarity metrics. The model performance decreases when using cosine for similarity measurement, indicating that model performance is compromised when the similarity between FedMCP models is not accurately measured. This further suggests that CKA is more effective at delivering a precise analysis of model similarity. A possible reason is that CKA can convey the connectivity of richer information representations more effectively than cosine similarity by assigning similarity values to feature structures.

5 Related Work

5.1 Personalized Federated Learning

The seminal training schema in FL is FedAvg (McMahan et al., 2017), which averagely aggregates local models into the global model. However, in non-IID settings, it is observed that FedAvg encounters difficulties with unstable and sluggish convergence, thereby causing performance degradation. To this end, various personalized techniques have been developed to mitigate the non-IID problem.

FedDF and FedMD (Sattler et al., 2020; Li and Wang, 2019) use knowledge distillation to train personalized models. Ditto and pFedMe (Li et al., 2021b; T Dinh et al., 2020) regularize local models based on the differences between global and local models to prevent client overfitting to local data. Another strategy involves collaboratively training personalized models for each client, like MOCHA, FedAMP, and FedFomo,(Huang et al., 2021b; Zhang et al., 2020; Smith et al., 2017) In FedATC, clients with different tasks participate in FL, exchanging useful information learned from local data through comparison with a common synthetic dataset sampled from clients(Dong et al., 2022) While this method enables personalization, employing synthetic datasets for comparison may raise privacy concerns.

Decomposing the entire network is also a common approach in personalized FL, reserving personalized layers for clients. FedPER (Arivazhagan et al., 2019) divides the model into shared base layers and personalized layers, demonstrating that models customized with individual classifiers for each client can effectively mitigate the impact of heterogeneity. CCVR (Luo et al., 2021) notes that the classifier has a greater bias than other layers, it uses sample virtual features to calibrate the classifier and enhance global model performance. FedETF (Li et al., 2023) also fixes classifier shift, which employs a synthetic simplex equiangular tight frame as a classifier. Other studies, like FedBABU and FedRep, also adopt the strategy of dividing the network into a head (classifier) and body (extractor)(Oh et al., 2021; Collins et al., 2021). However, many methods like CCVR and FedETF, based on correcting label class drift among clients, are not applicable in cross-silo federated NLP scenarios. Our proposed FedMCP alleviates the issue of data heterogeneity in cross-silo scenarios through a contrastive approach to personalization.

5.2 Federated Learning for NLP

FL has emerged as a dominant paradigm in privacy-preserving fields, where many NLP tasks utilize the FL framework to coordinate training models. Such as news recommendation (Yi et al., 2021), question answering (Chen et al., 2021; Dong et al., 2022), and text summarization (Pan et al., 2023). Pre-trained foundation models effectively capture knowledge for downstream tasks, but exchanging the full gradients of PLMs frequently in FL consumes substantial resources.

Hence, it is imperative to explore suitable FL methods supported by PLMs under resource constraints like communication and parameter adaptability. FedPETuning (Zhang et al., 2023) has conducted a comprehensive investigation into the FL performance of the representative PEFT method for PLMs. Fed-MNMT (Liu et al., 2023) fine-tunes PLMs with adapters for the federated multilingual neural machine translation problem, which alleviates the undesirable effect of data discrepancy by exploring adapter and clustering strategies. Passban et al., 2022 also focuses on neural machine translation in FL setting, proposing an effective technique to reduce the communication bandwidth by transferring half of the active tensors and ignoring the rest. C2A (Kim et al., 2023) notes the presence of client drift in typical PEFT approaches in FL scenarios, leading to slow convergence and performance degradation, and proposes using hypernetworks to generate client-customized adapters. However, since the client information in C2A includes label embeddings, C2A cannot be applied in cross-silo scenarios where clients have different tasks.

6 Conclusion

In this paper, we introduce a novel framework for PEFT methods of PLMs in the FL setting. We propose FedMCP to mitigate the non-IID problem in cross-silo scenario, which personalizes models by contrasting representations encoded by clients' global adapter and private adapter with backbone frozen. The model-contrastive method and aggregation strategy encourage the global adapter to learn universal knowledge and mitigate model drift across clients, and the private adapter to capture unique knowledge for achieving model personalization. Our experimental results show that FedMCP surpasses other baseline methods including the full fine-tuning method.

7 Limitation

While we show that FedMCP has performance improvements over other baseline methods, the enhancements are mostly within the range of 1$\sim$3%. This could be attributed to the already decent results achieved by the RoBERTa-base model and the scaled-down datasets, limiting the scope for further performance gains. We will further explore FedMCP in other cross-silo federated NLP settings.

Privacy leakage remains a concern in federated learning frameworks. Zhang et al., 2023 notes that compared to Full FT in FL, federated PEFT effectively defends against data reconstruction attacks. We suspect this is because clients do not share all model parameters. Our approach also retains part of the model locally and mitigates privacy leakage to some extent.

References

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

Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yixing Fan, and Xueqi Cheng. 2021. Fedmatch: federated learning over heterogeneous question answering data. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 181–190.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Chenhe Dong, Yuexiang Xie, Bolin Ding, Ying Shen, and Yaliang Li. 2022. Collaborating heterogeneous natural language processing tasks via federated learning. *arXiv preprint arXiv:2212.05789*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. 2021a. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 7865–7873.

Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. 2021b. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 7865–7873.

Hee-Jun Jung, Doyeon Kim, Seung-Hoon Na, and Kangil Kim. 2023. Feature structure distillation with centered kernel alignment in bert transferring. *Expert Systems with Applications*, 234:120980.

Yeachan Kim, Junho Kim, Wing-Lam Mok, Jun-Hyung Park, and SangKeun Lee. 2023. Client-customized adaptation for parameter-efficient federated learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1159–1172, Toronto, Canada. Association for Computational Linguistics.

Jakub Konečný, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. 2016. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMLR.

Daliang Li and Junpu Wang. 2019. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*.

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

Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. 2021b. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pages 6357–6368. PMLR.

Zexi Li, Xinyi Shang, Rui He, Tao Lin, and Chao Wu. 2023. No fear of classifier biases: Neural collapse inspired federated learning with synthetic and fixed classifier. *arXiv preprint arXiv:2303.10058*.

Bill Yuchen Lin, Chaoyang He, Zihang Zeng, Hulin Wang, Yufen Huang, Christophe Dupuy, Rahul Gupta, Mahdi Soltanolkotabi, Xiang Ren, and Salman Avestimehr. 2021. Fednlp: Benchmarking federated learning methods for natural language processing tasks. *arXiv preprint arXiv:2104.08815*.

Yi Liu, Xiaohan Bi, Lei Li, Sishuo Chen, Wenkai Yang, and Xu Sun. 2023. Communication efficient federated learning for multilingual neural machine translation with adapter. *arXiv preprint arXiv:2305.12449*.

Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. 2021. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. *Advances in Neural Information Processing Systems*, 34:5972–5984.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR.

Jaehoon Oh, Sangmook Kim, and Se-Young Yun. 2021. Fedbabu: Towards enhanced representation for federated image classification. *arXiv preprint arXiv:2106.06042*.

Rongfeng Pan, Jianzong Wang, Lingwei Kong, Zhangcheng Huang, and Jing Xiao. 2023. Personalized federated learning via gradient modulation for heterogeneous text summarization. *arXiv preprint arXiv:2304.11524*.

Peyman Passban, Tanya Roosta, Rahul Gupta, Ankit Chadha, and Clement Chung. 2022. Training mixed-domain translation models via federated learning. *arXiv preprint arXiv:2205.01557*.

Chen Qu, Weize Kong, Liu Yang, Mingyang Zhang, Michael Bendersky, and Marc Najork. 2021. Natural Language Understanding with Privacy-Preserving BERT. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1488–1497.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019a. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019b. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. 2020. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119.

Felix Sattler, Arturo Marban, Roman Rischke, and Wojciech Samek. 2020. Communication-efficient federated distillation. *arXiv preprint arXiv:2012.00632*.

Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. 2017. Federated multi-task learning. *Advances in neural information processing systems*, 30.

Canh T Dinh, Nguyen Tran, and Josh Nguyen. 2020. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405.

Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. 2022. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*.

Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Yongfeng Huang, and Xing Xie. 2022. Communication-efficient federated learning via knowledge distillation. *Nature communications*, 13(1):2032.

Jian Xu, Xinyi Tong, and Shao-Lun Huang. 2023. Personalized federated learning with feature alignment and classifier collaboration. *arXiv preprint arXiv:2306.11867*.

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

Jingwei Yi, Fangzhao Wu, Chuhan Wu, Ruixuan Liu, Guangzhong Sun, and Xing Xie. 2021. Efficientfedrec: Efficient federated learning framework for privacy-preserving news recommendation. *arXiv preprint arXiv:2109.05446*.

Dun Zeng, Siqi Liang, Xiangjing Hu, Hui Wang, and Zenglin Xu. 2023. Fedlab: A flexible federated learning framework. *Journal of Machine Learning Research*, 24(100):1–7.

Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M Alvarez. 2020. Personalized federated learning with first order model optimization. *arXiv preprint arXiv:2012.08565*.

Zhuo Zhang, Yuanhang Yang, Yong Dai, Qifan Wang, Yue Yu, Lizhen Qu, and Zenglin Xu. 2023. FedPETuning: When federated learning meets the parameter-efficient tuning methods of pre-trained language models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9963–9977, Toronto, Canada. Association for Computational Linguistics.