Federated Prompt-Tuning with Heterogeneous and Incomplete Multimodal Client Data

Supplementary Material

A. Pseudocode for FED-PRIME

Algorithm 1 Client Design

input: pre-trained model F, dataset D_t, and $\boldsymbol{w}_g^{inter}$, $\boldsymbol{w}_g^{intra}$
output: intra- and inter-client prompts $\boldsymbol{w}_p^{intra}$, $\boldsymbol{w}_p^{inter}$

initialize prediction head $\boldsymbol{w}_c$
initialize $\boldsymbol{w}_p^{intra}$, $\boldsymbol{w}_p^{inter} \leftarrow \boldsymbol{w}_g^{intra}$, $\boldsymbol{w}_g^{inter}$
optimize $\boldsymbol{w}_c$, $\boldsymbol{w}_p^{intra}$, $\boldsymbol{w}_p^{inter}$ using Eq. (7) and Eq. (8)
return $\boldsymbol{w}_p^{intra}$, $\boldsymbol{w}_p^{inter}$

Algorithm 2 Server Aggregation

input: a set of inter-client prompts $(\boldsymbol{w}_t^{inter} \triangleq (\boldsymbol{p}_t^i)_{i=1}^{\tau})_{t=1}^{n}$
output: optimized set of aggregated prompts $\boldsymbol{w}_g^{inter}$

initialize alignment model parameters $\boldsymbol{\theta}$
for $e = 1$ **to** max-iteration **do**
    fixing $\boldsymbol{\alpha}$, optimizing $\zeta, \gamma, \boldsymbol{\theta}$ via Eq. (13)
    freezing $\zeta, \gamma, \boldsymbol{\theta}$
    for $t = 1$ **to** n **do**
        freezing $\boldsymbol{\alpha}_{-t}$
        optimizing $\boldsymbol{\alpha}_t$ via Eq. (14)
    end for
end for
$\boldsymbol{w}_g^{inter} \leftarrow \boldsymbol{\theta}$
for $q = 1$ **to** $n \times \tau$ **do**
    $\boldsymbol{w}_g^{inter} \leftarrow \boldsymbol{w}_g^{inter} \setminus \boldsymbol{\theta}_q$ if $\nexists(t, p) : \alpha_t^{p,q} > 0$
end for
return set of aggregated inter-client prompts $\boldsymbol{w}_g^{inter}$

Algorithm 3 Multimodal Federated Prompt-Tuning

input: pre-trained model F and no. of iteration T
output: optimized set of aggregated prompts $\boldsymbol{\theta}$

initialize global prompt set $\boldsymbol{\theta}$
for $e = 1$ **to** T **do**
    for $t = 1$ **to** n **do**
        send $\boldsymbol{\theta}$ to client t
        request $\boldsymbol{w}_t^{inter}$, $\boldsymbol{w}_t^{intra}$ from client t via Alg. 1
    end for
    compute $\boldsymbol{w}_g^{intra}$ run FEDAVG on $(\boldsymbol{w}_t^{intra})_{t=1}^{n}$
    update $\boldsymbol{w}_g^{inter}$ via running Alg. 2 on $(\boldsymbol{w}_t^{inter})_{t=1}^{n}$
end for
return aggregated prompts $\boldsymbol{w}_g^{inter}$, $\boldsymbol{w}_g^{intra}$

B. Implementation Details

Input. All baselines use inputs from the UPMC Food-101 dataset, comprising 6,728 multimodal samples, and the MM-IMDB dataset, which includes 5,778 image-text pairs after the preprocessing. For the text modality, inputs are tokenized using the BERT-base-uncased tokenizer, as outlined in [24], with a maximum sequence length of 40 for UPMC Food-101 and 128 for MM-IMDB. When text is missing, we use an empty string as a dummy input. For the image modality, we follow [21, 24] by resizing the shorter side of the input image to 384 pixels, while keeping the longer side under 640 pixels to maintain the aspect ratio. As in [21], we decompose images into 32×32 patches. If the image is missing, we create a dummy image with all pixel values set to one, as described in [24].

Multimodal Backbone. Following [24], we adopt the pre-trained multimodal transformer ViLT [21] as our backbone as it is commonly used in various transformer-based methods for learning multimodal tasks. ViLT stems from Vision Transformer [13] and advances to process multimodal inputs with tokenized texts and patched images. Without using modality-specific feature extractors, ViLT is pre-trained on several large vision-language datasets (e.g., MS-COCO [28] and Visual Genome [22]) via objectives such as Image Text Matching (ITM) and Masked Language Modeling (MTM).

Model Training Details. To reduce the need for extensive fine-tuning, we freeze the ViLT backbone parameters and train only the learnable prompts and task-specific layers (pooler and classifier). Each pool consists of 20 prompts, from which 5 prompts are selected per input from each pool—inter- and intra-client ones. These prompts are concatenated and added to the initial Multi-Head Self-Attention (MSA) layer, resulting in a total of 10 prompts for each input. For FED-INTRA and FED-INTER, where only a single pool is utilized, the pool still contains 20 prompts. From this pool, 10 prompts are directly selected, ensuring that the total number of prompts per input remains consistent with FED-PRIME. In contrast, FEDAVG-P and FEDMSPLIT-P attach prompts to the first 6 MSA layers, with each prompt having a length of 16, resulting in a greater number of prompts per input.

Baseline Aggregation Details. In FEDAVG-P, the server aggregation procedure updates the newly trained components—namely, the prompts, pooler, and classifier—for each modality set and subsequently distributes them to the respective clients. In FEDMSPLIT-P, the original work as-

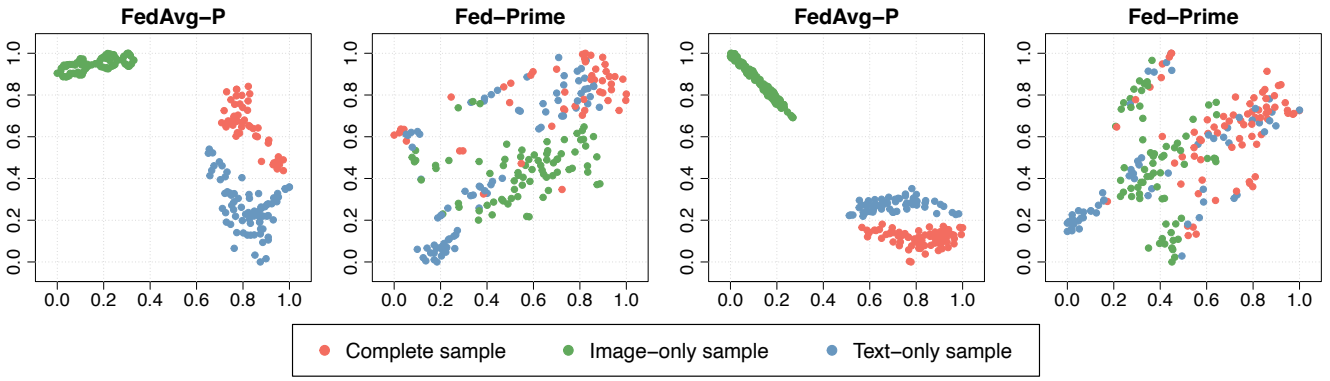

Figure 7. t-SNE plots of embeddings prior to classification on MM-IMDB under the **Miss Both** training scenario for Client #4 (left) and Client #14 (right), with two subfigures per client.

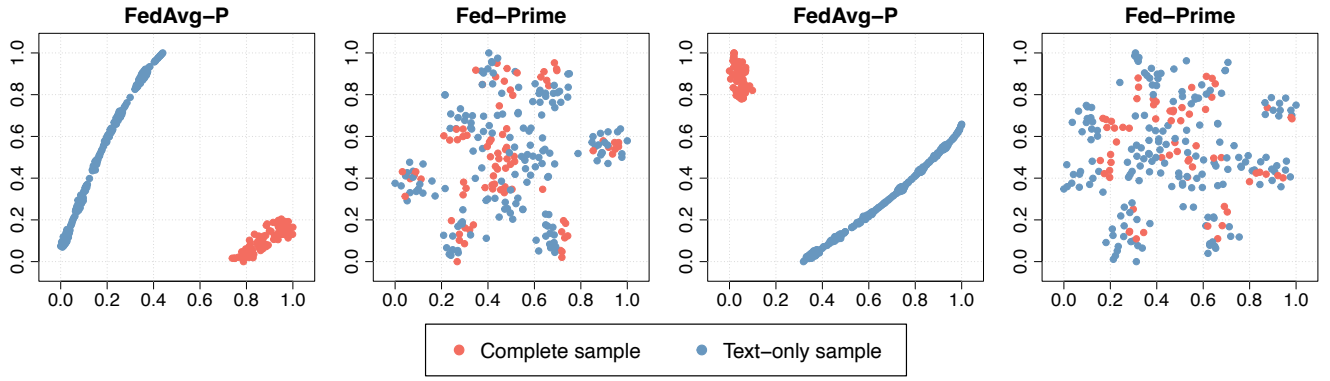

Figure 8. t-SNE plots of embeddings prior to classification on UMPC Food-101 under the **Miss Image** training scenario for Client #4 (left) and Client #14 (right), with two subfigures per client.

Table 2. Top-1 accuracy of FED-PRIME across varying pool sizes, reported on UPMC Food-101 dataset.

Pool Size	10	15	20	25	30
Top-1 Test Accuracy	73.94	82.50	82.61	82.72	83.21

sumes that each client possesses a distinct set of modalities, with a uniform modality distribution across all samples within a given client. This assumption contrasts with our setting, where clients exhibit heterogeneous and incomplete multimodal data distributions. Furthermore, the original framework constructs modality-specific encoder blocks, aggregating similar blocks based on similarity coefficients that quantify inter-client modality relationships. To adapt this approach to our setting, we aggregate all modular components—including prompts, pooler, and classifier—based on client similarity while assuming that all clients maintain a shared set of these modular components, encompassing all possible modality configurations.

Hyperparameter Settings. All evaluated baselines utilize a batch size of 512 for the UPMC Food-101 dataset and 256 for MM-IMDB. The training datasets for UPMC Food-101 and MM-IMDB were randomly partitioned into 20 clients, all of whom participated in every communication round. We use a Stochastic Gradient Descent (SGD) optimizer with a base learning rate of 0.05 for Food-101 and 0.01 for IMDB, with a communication round of 250 for faster convergence compared to full fine-tuning. During each round, clients train the global model locally for 1 epoch. The hyperparameter configurations were consistently applied across experimental missing scenarios.

Code Availability and Reproducibility. We release our full implementation and configurations at https://github.com/hangpt01/FedPrime. The repository includes source code, configuration files (e.g., prompt dimensions, number of prompts, clustering thresholds, regularization coefficients), and brief guidelines with scripts to run experiments or adapt the framework to other datasets.

Automatic Hyperparameter Tuning. Beyond the default settings described above, the repository includes utilities and scripts to automatically tune key hyperparameters (e.g., clustering iteration thresholds, pool sizes, and the number of

Table 3. Resource required of different baselines in each round

Method	UPMC Food-101 [46]		MM-IMDB [1]	
	GPU (GB)	Time (s)	GPU (GB)	Time (s)
FEDAVG-P	31.70	245.53	39.07	649.58
FEDMSPLIT-P	34.30	265.19	39.13	755.62
FED-INTER	16.83	315.67	23.73	685.27
FED-INTRA	16.65	253.60	23.31	625.83
FED-PRIME	16.80	240.54	23.68	653.57

active prompts). We also provide implementations of common tuning methods, namely Grid Search, Random Search, and Bayesian Optimization, which allow users to efficiently explore configurations based on validation loss or heuristic rules. These tools reduce the need for extensive manual searches and simplify adaptation to new problems.

C. Computational Resource Analysis.

Table 3 summarizes the computational resources required by the evaluated methods. Notably, FEDAVG-P and FEDMSPLIT-P demonstrate substantially higher GPU memory usage, such as 31.70 GB and 34.30 GB on the UPMC Food-101 dataset, nearly double that of FED-PRIME. This discrepancy can be attributed to their strategy of prepending prompts in the first six Multi-Head Self-Attention (MSA) layers out of a total of 12 layers, which expands sequence lengths and amplifies memory demands for intermediate activations, gradients, and computations. Moreover, FEDMSPLIT-P further increases computational costs, as it requires estimating similarity across clients and maintaining a personalized model for each client. Conversely, FED-PRIME limits the prompt addition to only the first MSA layer, thereby restricting the augmented sequence length and associated computations to a single layer. This design significantly reduces the overall memory overhead. Other variants demonstrate relatively comparable GPU memory usage. Overall, GPU usage for the MM-IMDB dataset is higher than for UPMC Food-101, primarily due to the longer text sequence lengths employed in the experiments.

In terms of execution time per round, FED-INTER and FEDMSPLIT-P interchangeably exhibits the longest runtime. Other methods exhibit similar runtime performance, with approximately a 5-second difference between FEDAVG-P and FED-PRIME. Importantly, FED-PRIME demonstrates its efficiency by achieving high performance while maintaining lower GPU memory requirements and comparable execution time.

D. Additional Ablation Studies

Table 2 demonstrates a clear positive correlation between pool size and test accuracy, with larger pool sizes consistently yielding higher accuracy. A significant drop in accuracy is observed when the pool size decreases from 15 to 10, highlighting a critical threshold for maintaining performance. While increasing the pool size from 15 to 30 results in an overall improvement in accuracy, the rate of improvement diminishes as the pool size grows from 15 to 25. Notably, a pool size of 30 achieves the highest accuracy, showing a larger improvement compared to the smaller gains observed when increasing the pool size from 15 to 25. In general, while larger pool sizes improve accuracy, the marginal gains may not justify the extra computational cost. Hence, a pool size of 20 is selected for all subsequent experiments in this study, as it provides a practical balance between accuracy and computational efficiency.

E. Additional Prompting Analysis

Fig. 7 illustrates the embeddings at the final round (round 250) for FEDAVG-P and FED-PRIME under the **Miss Both** scenario on the MM-IMDB dataset, visualized for two random clients (Client #4 and Client #14). As shown in the figure, FEDAVG-P, employing a design that prompts embeddings for each missing type separately, possesses distinct clusters for each sample type in the embedding space. In contrast, FED-PRIME produces more scattered embeddings, where the embeddings of complete samples are positioned closer to those of samples with missing modalities. This demonstrates FED-PRIME's superior ability to learn meaningful and comprehensive representations, regardless of the specific missing type in a sample. A notable observation is that image-only samples in FEDAVG-P embeddings are significantly distant from text-only and complete samples. This separation highlights the limitations of FEDAVG-P in capturing cross-modal relationships effectively, which can be attributed to the fact that this method utilizes prompts specific to the missing modality type. This further supports the hypothesis that the ViLT backbone in FED-PRIME has a better capability for text representation and integration.

To further explore these models, Fig. 8 presents embeddings under the **Miss Image** training scenario on the UMPC Food-101 dataset for the same clients. A similar trend is observed: FED-PRIME generates embeddings with well-integrated representation of different sample types, while FEDAVG-P produces two distinct and distant clusters. As observed before, when image-only samples are included in the data, embeddings for text and complete samples in FEDAVG-P may cluster closer together. However, even in the absence of image-only samples, embeddings for text and complete samples remain overly distinct, highlighting a

Table 4. Large-scale FL settings using UPMC Food-101 Dataset. FED−PRIME outperforms FEDMSPLIT−P in all scenarios of different number of clients.

Metrics	Method	Participating Rates				
		10%	20%	30%	40%	50%
Top-1 Test Accuracy	FEDMSPLIT−P	14.96	22.33	48.98	59.90	81.82
	FED−PRIME	**22.33**	**48.98**	**74.27**	**80.25**	**82.06**
GPU (GB)	FEDMSPLIT−P	22.82	27.21	31.78	37.55	40.12
	FED−PRIME	**12.83**	**10.12**	**10.17**	**10.50**	**10.69**

lack of cohesive representation across modalities. This separation likely hinders FEDAVG−P's performance, contributing to its moderate results when classifiers are subsequently applied to these embeddings.

F. Additional Experiment Results

Inter-pool Convergence. Fig. 9 and Fig. 10 depict the size of the pool of the inter-client prompt pool for the two datasets in three defined training scenarios. The results indicate an initial increase in the spread of prompts relative to their centroid, followed by a subsequent decrease. This observation suggests that the proposed method optimizes the learned prompts to effectively capture the diversity of data across participating clients. Once the model sufficiently learns the clustering and alignment, it begins to condense client-specific information, which leads to an increase in the overall pool size.

Scalability in Large-scale FL. We performed additional experiments on the Food-101 dataset, simulating a scenario with 100 clients and varying client participation rates between 10% and 50%. These results were compared against the strongest baseline for this dataset, FEDMSPLIT−P. As shown in Tab. 4, our dual-prompt mechanism demonstrates strong scalability, with performance improving as the number of participating clients increases. Moreover, our method consistently outperforms the baseline in terms of test accuracy. In terms of computational efficiency, FED−PRIME exhibits substantially lower GPU memory consumption, which remains stable regardless of the number of clients, in contrast to FEDMSPLIT−P, which experiences increased memory usage as the number of clients increases. This property makes FED−PRIME particularly well-suited for deployment in resource-constrained environments.

Performance under Highly Imbalanced Data. In addition to modality-missing heterogeneity, we also evaluate our method under extreme data imbalance. FedProx [26], a popular baseline specifically designed to address Non-IID problems, is selected for comparison in this setting. Specifically, we compare our method against a variant that modifies the averaging of classifiers and intra-client prompts using FedProx, denoted as FEDPROX−P. We conduct additional evaluations on the Food-101 dataset under extreme Non-IID settings by simulating a Dirichlet distribution with $\alpha = 0.1$. The results from Tab. 5 confirm that our method consistently outperforms FEDPROX−P, further demonstrating the effectiveness of aggregating inter-client prompts to combat extreme data imbalance. Although FEDPROX−P is designed to address data heterogeneity, our method's use of inter-client prompts effectively mitigates this issue, highlighting the strength of our prompt-alignment algorithms and the selective averaging strategy applied to specific model components. We also evaluate scenarios in which clients possess either full modalities or only a single modality, with our method surpassing both FEDAVG−P and its centralized counterpart in accuracy (see Fig. 1 and Fig. 4).

Table 5. Non-IID FL settings with UPMC Food-101 Dataset. Results indicated by a dash (-) represent scenarios where **Test (Miss Both)** is the same as **Test (∼Train)**.

Train	Method	Test (∼ Train)	Test (Miss Both)	Test (Full Modal)	Test (Text only)	Test (Image only)
Miss Text	FEDPROX-P	67.42	64.34	77.29	56.83	68.24
	FED-PRIME	**71.20**	**71.15**	**85.08**	**63.91**	**69.56**
Miss Image	FEDPROX-P	82.56	71.26	85.24	83.05	45.31
	FED-PRIME	**87.38**	**75.59**	**89.25**	**87.05**	**48.47**
Miss Both	FEDPROX-P	75.75	-	89.36	83.98	69.61
	FED-PRIME	**79.98**	-	**91.00**	**86.70**	**70.38**

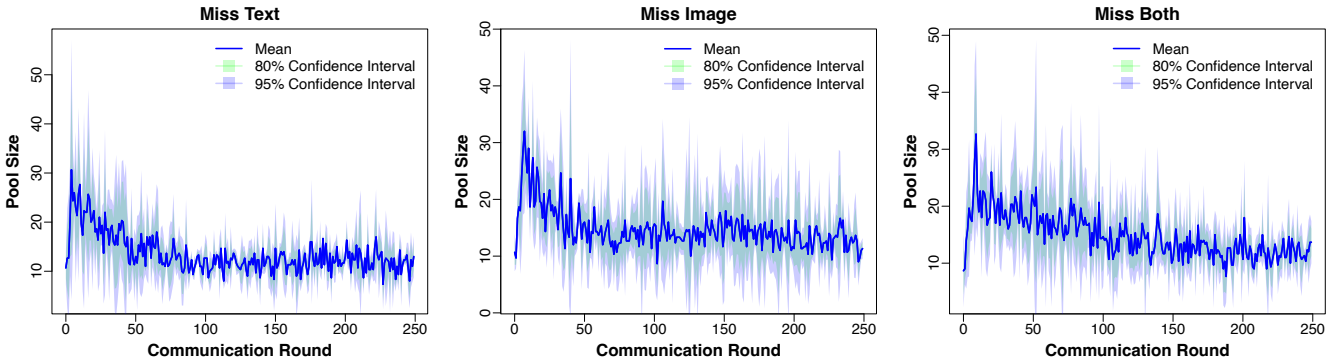

Figure 9. Variations in UMPC Food-101 inter-client prompt pool size across 250 communication rounds under different training scenarios.

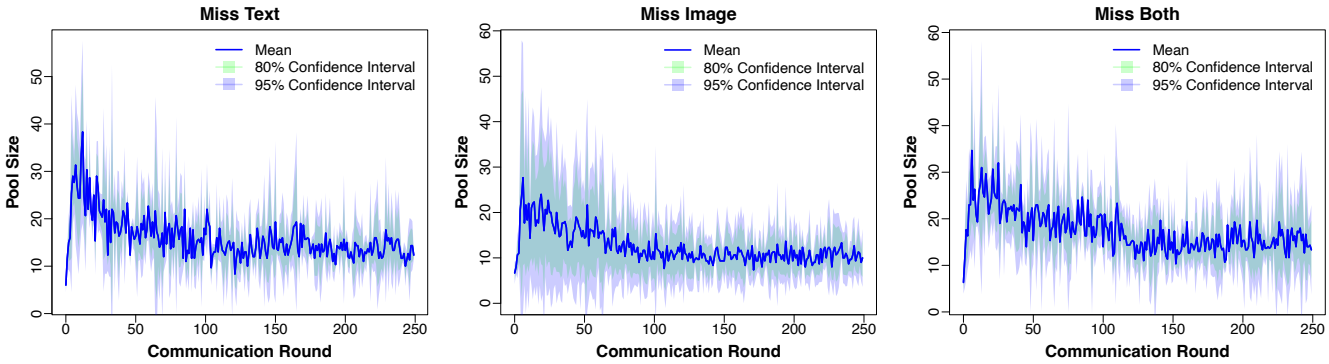

Figure 10. Variations in MM-IMDB inter-client prompt pool size across 250 communication rounds under different training scenarios.