ALIGNMENT PARADIGM OF INTRINSIC SUBSPACE EXPERTS FOR FEDERATED FINE-TUNING

Anonymous authors
Paper under double-blind review

ABSTRACT

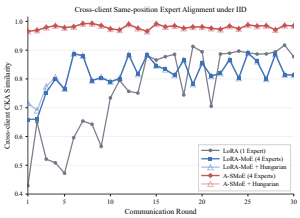
The rapid scaling of foundation models has made full-parameter training increasingly impractical in federated scenarios due to prohibitive computational and communication costs, motivating the development of federated parameter-efficient fine-tuning (PEFT). Meanwhile, mixture-of-experts (MoE) based adaptation has shown promising capability in improving model specialization and scalability. However, directly applying MoE-based LoRA adaptation to federated fine-tuning remains challenging. The fundamental difficulty lies in various forms of misalignment introduced by decentralized optimization. Furthermore, MoE architectures introduce additional challenges under heterogeneous clients, where experts with identical positions may capture inconsistent knowledge and independently optimized routers may become difficult to aggregate. To address these challenges, we propose **A-SMoE** for federated fine-tuning. Instead of allowing local experts to freely learn independent adaptation spaces, A-SMoE exploits the intrinsic subspaces of pretrained models through singular value decomposition and freezes the corresponding input and output bases. This design provides a shared semantic coordinate system for all clients, thereby improving the consistency of expert representations and enabling more reliable federated aggregation. Moreover, we introduce a parameter-free routing mechanism based on subspace projection energy, eliminating additional router parameters and avoiding auxiliary objectives required for router balancing. Extensive experiments on multiple benchmark datasets demonstrate that A-SMoE consistently outperforms existing federated fine-tuning methods.

1 INTRODUCTION

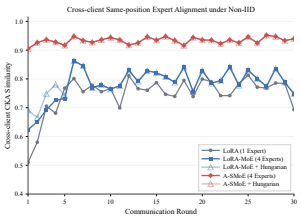
Federated learning (FL) (McMahan et al., 2017) has emerged as a promising paradigm for collaborative model training under privacy constraints, enabling multiple clients to jointly optimize models without directly sharing raw data. However, the rapid scaling of foundation models (Achiam et al., 2023) makes conventional federated full-parameter training increasingly impractical due to excessive computation, memory, and communication costs. Consequently, federated parameter-efficient fine-tuning (Wang et al., 2025) has attracted growing attention, where only a small number of additional parameters are optimized while the pretrained backbone remains frozen. Among various PEFT approaches, Low-Rank Adaptation (LoRA) (Hu et al., 2022) has become one of the most widely adopted techniques due to its effectiveness and efficiency. Meanwhile, Mixture-of-Experts (SHI et al., 2025; Du et al., 2022) architectures have demonstrated remarkable scalability in large foundation models by increasing model capacity while activating only a subset of experts for each input. Large-scale models have validated the effectiveness of sparse expert activation. Inspired by these advances, recent studies have explored combining MoE with LoRA, where multiple LoRA modules are treated as lightweight experts and selected through a routing mechanism (Dou et al., 2023). This paradigm provides a promising direction for enhancing the capacity and specialization of parameter-efficient adaptation.

However, directly transferring MoE-based LoRA adaptation (Wang et al., 2024; Yuan et al., 2025) from centralized training to federated fine-tuning remains challenging. Decentralized optimization introduces fundamental alignment issues, where functionally similar local adaptations may not correspond to compatible parameter representations.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) CKA similarity under IID.



(b) CKA similarity under Non-IID.

Method	L-ACC	G-ACC	Param
IID			
LoRA (Single Expert)	58.03	56.95	0.005169
LoRA-MoE (4 Experts)	58.19	57.09	0.005718
LoRA-MoE (4 Experts) + CKA	58.27	57.11	0.005718
A-SMoE (4 Experts)	**58.87**	**57.92**	0.005169
Non-IID			
LoRA (Single Expert)	46.89	54.39	0.005169
LoRA-MoE (4 Experts)	52.29	56.52	0.005718
LoRA-MoE (4 Experts) + CKA	52.36	56.81	0.005718
A-SMoE (4 Experts)	**54.89**	**57.38**	0.005169

(c) CKA indicates that a CKA-based Hungarian algorithm is applied for permutation-based aggregation.

Figure 1: We evaluate the corresponding expert CKA similarity and final accuracy of LoRA-MoE, LoRA, and A-SMoE fine-tuned on RoBERTa-large (SST-5). By freezing U_e and V_e, A-SMoE consistently achieves the highest accuracy. Interestingly, in IID, LoRA exhibits higher CKA similarity than LoRA-MoE, confirming the expert misalignment issue in standard MoE. Conversely, in Non-IID conditions, LoRA-MoE's CKA surpasses LoRA despite exacerbated misalignment. This underscores the necessity of MoE: in Non-IID, a single expert struggles with knowledge distribution, risking aggregation collapse, whereas MoE mitigates this to maintain higher robustness. Moreover, we observe that CKA-based permutation is only effective during the initial training epochs and yields limited improvement in final accuracy.(see Appendix A for experimental details).

Even without introducing MoE, federated LoRA already suffers from several forms of parameter misalignment. Given a LoRA update $\Delta W = BA$, directly averaging low-rank factors cannot preserve the averaged update, since $(\sum_k B_k)(\sum_k A_k) \neq \sum_k B_k A_k$, resulting in *aggregation misalignment* (Guo et al., 2025). Moreover, the non-identifiability of low-rank factorization introduces additional ambiguities, where the same update can be represented as $BA = (cB)(c^{-1}A)$ or $BA = (BQ)(Q^\top A)$ with an orthogonal matrix Q (Zhang et al., 2026). These equivalent parameterizations lead to inconsistent scales and latent coordinate systems across clients, causing *scale misalignment* and *orthogonal-basis misalignment*.

Beyond these intrinsic ambiguities of LoRA, introducing MoE further brings additional alignment challenges. Due to heterogeneous client data (Zhao et al., 2018), experts with the same index across different clients may capture different semantic knowledge, resulting in expert misalignment (as shown in Figure 1). Meanwhile, independently optimized routers may learn inconsistent routing preferences and decision boundaries, leading to router misalignment. Therefore, federated MoE fine-tuning simultaneously suffers from multiple alignment issues, including aggregation, scale, orthogonal-basis, expert, and router misalignment. These observations motivate us to reconsider federated MoE fine-tuning from the perspective of alignment. Instead of allowing each client to freely learn independent expert representations and attempting to reconcile them after local optimization, we propose **A-SMoE**. A-SMoE establishes a shared structural reference based on the intrinsic subspaces of pretrained models. Specifically, it uses the input and output subspaces obtained from SVD decomposition of pretrained weights to initialize all experts. Many LoRA-SVD approaches (**?**Meng et al., 2024) implicitly assume that singular vectors corresponding to the largest singular values capture the most relevant directions for downstream tasks, yet this assumption has not been validated. We further verify the effectiveness of these dominant singular subspaces through comprehensive experiments with multiple metrics. Furthermore, in LLM MoE models (Dai et al., 2024), router parameters are negligible compared with the enormous expert capacity. However, in PEFT, where each expert contains only lightweight parameters, router parameters introduce non-negligible overhead. Moreover, conventional MoE separates expert representation and expert selection by optimizing experts and routers independently (Cheng et al., 2026). To address this issue, A-SMoE introduces a parameter-free routing mechanism derived from intrinsic expert subspaces. By distributing the SVD-derived U and V matrices—acting as input and output perspective matrices—across individual experts in a round-robin fashion, it achieves zero-cost load balancing without auxiliary losses disrupting primary objective descent.

Meanwhile, A-SMoE mitigates the three aforementioned LoRA misalignments and theoretically bounds its aggregation error below naive averaging. Empirically, extensive experiments validate its superiority over existing federated fine-tuning baselines via alignment-based adaptation. The main contributions of this work are summarized as follows:

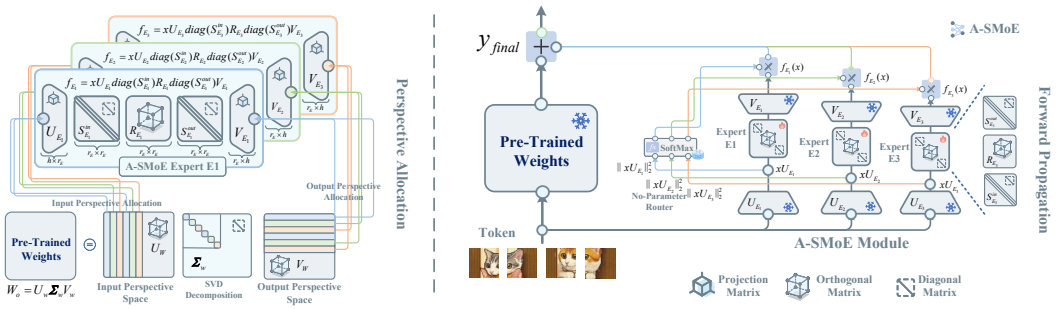


Figure 2: Perspective allocation and forward process of A-SMoE.

- We propose **A-SMoE**, an intrinsic-subspace expert framework that provides a shared representation reference for federated expert learning and improves aggregation compatibility.
- We introduce a parameter-free routing mechanism that avoids additional routing parameters and auxiliary balancing loss while coupling routing decisions with expert representations.
- Extensive experiments demonstrate that A-SMoE achieves superior performance and robustness under diverse federated heterogeneous settings.

2 Related Work

Federated PEFT. Federated LoRA suffers from several alignment issues, yet existing methods typically address them in isolation. Aggregation-oriented approaches (Singhal et al., 2025; Bai et al., 2024; Babakniya et al., 2023) often apply SVD to recover low-rank factors, but computing the SVD of large-scale matrices is highly computationally expensive. Other studies introduce explicit orthogonal alignment by solving rotation matrices between local adapters, or mitigate aggregation inconsistency through alternating optimization (Chen et al., 2024; Zhang et al., 2026) or freezing one LoRA factor, as in FFA (Sun et al., 2024) methods. However, alternating or partially frozen optimization updates different LoRA components across rounds, which may interrupt the continuity of local adaptation and limit coordinated parameter learning. A unified treatment of these alignment issues therefore remains underexplored.

Mixture-of-Experts. MoE was originally developed to increase model capacity while reducing the computation required for each input through sparse expert activation. This paradigm has been successfully scaled to large language models, such as Switch Transformer (Fedus et al., 2022) and Mixtral (Jiang et al., 2024), where only a small subset of experts is activated for each token. More recently, MoE has evolved from a model-scaling mechanism into a general architectural paradigm and has been incorporated into various downstream adaptation scenarios (Mustafa et al., 2022; Chen et al., 2023). In PEFT, several studies (Wang et al., 2024; Yuan et al., 2025) combine MoE with LoRA by treating individual LoRA modules as lightweight experts and employing a trainable router to dynamically select them. Despite their success in centralized settings, extending MoE-based PEFT to federated learning introduces expert and router misalignment under client heterogeneity, motivating the alignment-oriented design explored in this work.

3 Methodology

3.1 Overview

We propose **A-SMoE**, a federated fine-tuning framework consisting of three key components: intrinsic-subspace experts, parameter-free subspace routing, and Geometry-Aware Aggregation. The central idea is to restrict independently trained local experts to a shared coordinate system inherited from the pretrained model, thereby reducing the degrees of freedom that cause expert and parameter misalignment across clients.

3.2 INTRINSIC SUBSPACE EXPERTS

We first revisit the structure of a standard LoRA update $\Delta W = BA$, where $B \in \mathbb{R}^{d_{\text{out}} \times r}$ and $A \in \mathbb{R}^{r \times d_{\text{in}}}$. Applying compact SVD to the two factors gives $B = U_B \Sigma_B V_B^\top$ and $A = U_A \Sigma_A V_A^\top$. The LoRA update can therefore be rewritten as $\Delta W = U_B \Sigma_B (V_B^\top U_A) \Sigma_A V_A^\top$. Since V_B and U_A are orthogonal matrices, their product $R = V_B^\top U_A$ is also orthogonal. Thus, a LoRA module can be interpreted as an input-view space $V_A^\top$ and an output-view space U_B, connected by an internal processor $\Sigma_B R \Sigma_A$. This decomposition suggests that fixing the two view spaces while optimizing only the internal processor can provide a shared semantic coordinate system across clients. To obtain these shared spaces, we directly inherit them from the pretrained weight $W_0 = U_0 \Sigma_0 V_0^\top$. Its singular-vector subspaces are assigned to the experts as fixed input and output bases, i.e., $U_e \leftarrow U_0$ and $V_e \leftarrow V_0^\top$. Each expert E_e is therefore written as

$$f_{E_e}(x) = x U_e \operatorname{diag}(S_{E_e}^{\text{in}}) R_{E_e} \operatorname{diag}(S_{E_e}^{\text{out}}) V_e \tag{1}$$

where only S_e^{in}, R_e, and S_e^{out} are locally optimized. We allocate distinct singular-vector subspaces of the pretrained weight to different experts. Let r_s denote the total number of selected directions and E_N the number of experts; each expert receives $r_e = r_s / E_N$ unique directions. Since these directions originate from an orthogonal SVD basis, for any two distinct experts E_e and $E_{e'}$, their subspaces satisfy $U_e^\top U_{e'} = 0$ and $V_e V_{e'}^\top = 0$ due to the orthogonal SVD basis. Thus, their outputs f_{E_e} and $f_{E_{e'}}$ remain orthogonal, which helps alleviate expert redundancy. Following existing SVD-based LoRA methods (**?**Meng et al., 2024; Wu et al., 2026), we prioritize directions with the largest singular values. Although this choice is commonly motivated by the assumption that dominant directions are more relevant to downstream task, the assumption itself has received limited empirical verification. We therefore empirically investigate this assumption, with detailed results reported in Appendix B.

For multiple experts, simply assigning consecutive singular directions may concentrate neighboring spectral components within individual experts. We instead employ a round-robin partition over the selected dominant singular directions. Given E_N experts, the e-th expert receives indices $\mathcal{I}_e = \{e, e + E_N, \ldots, e + (r_e - 1)E_N\}$. In the experiments, we show that this interleaved assignment strategy promotes more balanced expert routing and alleviates routing imbalance to some extent.

3.3 PARAMETER-FREE INTRINSIC-SUBSPACE ROUTING

Conventional MoE routers introduce non-negligible parameter overhead in PEFT and exacerbate cross-client misalignment. To eliminate this drift, A-SMoE derives routing decisions directly from $U_e \in \mathbb{R}^{d_{in} \times r}$ without introducing auxiliary parameters. Given an input representation $x \in \mathbb{R}^{d_{in}}$, we evaluate its projection energy s_e onto the frozen input basis U_e of expert E_e:

$$s_e = \frac{1}{r} \|x U_e\|_2^2 \tag{2}$$

A larger s_e indicates a stronger semantic affinity between the input and the expert's intrinsic subspace. To account for scale variations across the E_N experts, we standardize the energy scores to obtain $\hat{s}_e$, and compute the routing probability $g_e(x)$ via a Softmax: $g_e(x) = \frac{\exp(\hat{s}_e/\tau)}{\sum_{j=1}^{E} \exp(\hat{s}_j/\tau)}$. This parameter-free mechanism natively couples *where an expert operates* with *when it is activated*. The complete forward pass of the A-SMoE layer is formulated as:

$$y = x W_0 + \gamma \sum_{e=1}^{E_N} g_e(x) \underbrace{\left(x U_e \operatorname{diag}(S_{E_e}^{\text{in}}) R_e \operatorname{diag}(S_{E_e}^{\text{out}}) V_e \right)}_{f_{E_e}(x)} \tag{3}$$

where $\gamma = \frac{\alpha}{r}$ denotes the scaling factor. By relying solely on the globally shared U_e, this formulation inherently guarantees routing consistency across heterogeneous clients, completely circumventing the need for auxiliary load-balancing objectives while confining local adaptations strictly within a unified coordinate system.

3.4 GEOMETRY-AWARE AGGREGATION

While shared bases resolve expert correspondence, naively averaging the learnable processor $S_{E_e}^{in} R_e S_{E_e}^{out}$ across clients fails because the product of averages generally differs from the average of products. To resolve this, we propose a geometry-aware aggregation strategy that strictly respects the mathematical constraints of each component.

Orthogonal-basis Alignment. The frozen U_e and V_e establish a common external coordinate system for every client. Unlike unconstrained LoRA factorization, arbitrary orthogonal transformations (Zhang et al., 2026) cannot be freely inserted into the external bases while preserving the proposed diagonal–orthogonal–diagonal structure. This substantially removes the orthogonal-basis ambiguity of independently optimized LoRA factors.

Scale Alignment. Because R_e is constrained to the orthogonal group, it does not introduce an arbitrary multiplicative scaling degree of freedom. The remaining scale ambiguity lies between $S_{E_e}^{in}$ and $S_{E_e}^{out}$. Before aggregation, we therefore normalize the diagonal vector $S_{E_e}^{in}$ as $\tilde{S}_{E_e}^{in} = S_{E_e}^{in} / \|S_{E_e}^{in}\|_2$ and transfer its magnitude to the output factor as $\tilde{S}_{E_e}^{out} = \|S_{E_e}^{in}\|_2 S_{E_e}^{out}$. This transformation leaves the represented expert unchanged, while placing all clients in a consistent scale convention before aggregation. A detailed analysis of scale ambiguity is provided in Appendix C.

Aggregation Alignment. After scale alignment, the processor of expert e on client k takes the form $D_{e,k}^{in} R_{e,k} D_{e,k}^{out}$, where $D_{e,k}^{in} = \mathrm{diag}(S_{N_e,k}^{in})$, $D_{e,k}^{out} = \mathrm{diag}(S_{N_e,k}^{out})$, $\|S_{N_e,k}^{in}\|_2 = 1$, and both diagonal factors are strictly positive. For each expert, the ideal FedAvg processor is $W_e^\star = \sum_k p_k D_{e,k}^{in} R_{e,k} D_{e,k}^{out}$. Ideally, $W_e^\star$ should be decomposed back into the same positive-diagonal–orthogonal–positive-diagonal form before being broadcast to clients. However, such an exact constrained decomposition generally does not exist because this structured set is not closed under weighted averaging. Inspired by FlexLoRA (Bai et al., 2024), which first forms the aggregated LoRA update and subsequently recovers an admissible low-rank representation through truncated SVD, we instead adopt a tractable structure-preserving relaxation.

Specifically, since scale alignment has already enforced $\|S_{N_e,k}^{in}\|_2 = 1$, we first average the input scales and project them back to the unit sphere as $\widehat{S}_{N_e}^{in} = (\sum_k p_k S_{N_e,k}^{in}) / \|\sum_k p_k S_{N_e,k}^{in}\|_2$, yielding $\widehat{D}_e^{in} = \mathrm{diag}(\widehat{S}_{N_e}^{in})$. For the orthogonal component, we compute $\bar{R}_e = \sum_k p_k R_{e,k} = U_e^R \Sigma_e^R (V_e^R)^\top$ and project it back to the orthogonal group as $\widehat{R}_e = U_e^R (V_e^R)^\top$ (Edelman et al., 1998). After fixing these two feasible factors, the original coupled decomposition reduces to optimizing only the remaining positive output scale.

Let $a_{e,j}$ and $w_{e,j}^\star$ denote the j-th columns of $\widehat{D}_e^{in} \widehat{R}_e$ and $W_e^\star$, respectively. The resulting constrained least-squares problem is separable and admits

$$\widehat{S}_{e,j}^{ont} = \max\left\{ \epsilon, \frac{a_{e,j}^\top w_{e,j}^\star}{\|a_{e,j}\|_2^2} \right\}, \tag{4}$$

where $\epsilon > 0$ is a small lower bound preserving the strict positivity required by the Softplus parameterization.

Theorem 1 (Aggregation Dominance). *For any expert e, let $\bar{D}_e^{in} = \sum_k p_k D_{e,k}^{in}$ and $\bar{D}_e^{out} = \sum_k p_k D_{e,k}^{out}$. Suppose conventional factor-wise averaging uses the same projected orthogonal factor $\widehat{R}_e$, giving $W_{\mathrm{avg},e} = \bar{D}_e^{in} \widehat{R}_e \bar{D}_e^{out}$. For a sufficiently small ϵ such that the averaged solution remains feasible, the proposed reconstruction $W_{\mathrm{ours},e} = \widehat{D}_e^{in} \widehat{R}_e \widehat{D}_e^{out}$ satisfies*

$$\|W_e^\star - W_{\mathrm{ours},e}\|_F^2 \le \|W_e^\star - W_{\mathrm{avg},e}\|_F^2. \tag{5}$$

The complete derivation is provided in Appendix D.

4 EXPERIMENT

All experiments in this paper are implemented using the PyTorch framework and conducted on a hardware setup comprising one NVIDIA A800 GPU, one NVIDIA RTX PRO 6000 GPU, and eight

NVIDIA RTX 4090 GPUs. To ensure reliability, all experiments are run three independent times with different random seeds, and the average results are reported. We insert the adapters into the Q and V projections of the attention layers. The learning rate η is selected from $\{1\text{e-}2, 1\text{e-}3, 1\text{e-}4\}$, and the weight decay factor is set to 1e-4.

To avoid local polarization (Shao et al., 2025) (i.e., the illusion of high local accuracy caused by extreme data heterogeneity), we employ a two-pronged evaluation strategy:

- **Global Evaluation:** The standard test set is uniformly distributed across all clients to assess generalized global performance.
- **Local Evaluation:** The training data is heterogeneously partitioned among clients, with each client uniformly reserving 20% of its allocated data to serve as a local test set.

Baselines. To comprehensively evaluate the effectiveness of our proposed approach, we compare it against 11 baselines. These include both dedicated federated fine-tuning frameworks and PEFT methods adapted for the federated setting:

- **FedIT (Zhang et al., 2024):** A foundational baseline that integrates FedAvg with LoRA for federated fine-tuning.
- **FedSA (Guo et al., 2025):** A personalized federated fine-tuning approach that aggregates the LoRA A matrix globally while retaining the B matrix locally for personalization.
- **FFA-LoRA (Sun et al., 2024):** A method that strictly freezes the A matrix during training to inherently prevent weight misalignment across clients.
- **FedRot (Zhang et al., 2026):** Employs alternating updates and introduces rotation matrices to specifically mitigate orthogonal bias misalignment.
- **FedALT (Bian et al., 2026):** An MoE-integrated federated framework that applies distinct aggregation strategies across experts to ensure representational consistency.
- **FlyLoRA (Zou et al., 2026):** An MoE-inspired fine-tuning approach that treats each rank as an independent expert while keeping the A matrix frozen.
- **FlexLoRA (Bai et al., 2024):** Computes an aggregated global matrix and subsequently applies truncated SVD to project the weights back into the standard LoRA bottleneck architecture.
- **LoRA-XS (Bałazy et al.) & PiSSA (Meng et al., 2024):** Both methods initialize their fine-tuning modules by leveraging the SVD of the original pre-trained model weights.
- **LoRA-MoE (Dou et al., 2024):** The standard Mixture-of-Experts adaptation of LoRA.
- **LoRA-MoE+CKA:** Builds upon LoRA-MoE by employing a CKA-based Hungarian algorithm during aggregation; it designates a reference client and aligns the experts of all other clients based on optimal matching scores prior to the final aggregation.

Note that the definition of rank and the resulting parameter overhead differ across methods. To guarantee a fair comparison, consistent with LoRA-XS, we dynamically adjust the rank of each baseline so its total trainable parameter count matches that of FedIT under a specified rank setting. Additionally, all LoRA-MoE methods are equipped with 4 experts and a linear router utilizing Top-2 routing. A-SMoE defaults to 4 experts.

4.1 OVERALL PERFORMANCE

Language Understanding Experiments. We first evaluate the capabilities of our proposed method on Natural Language Processing (NLP) tasks. Specifically, we fine-tune the RoBERTa-Large (Liu et al., 2019) model on the GLUE benchmark (Wang et al., 2018), focusing on four representative tasks: SST-2, QNLI, MRPC, and QQP. To simulate a realistic federated environment, we construct a task-heterogeneous federated learning framework comprising 12 clients. These clients are evenly distributed, with every three clients exclusively assigned to train on one of the four tasks. For clients assigned to the same task, their local training data is heterogeneously partitioned using a Dirichlet distribution with $\alpha = 0.5$. During the training process, the client participation rate is set to 0.5 at

Table 1: Comparison on GLUE. We report local (L-ACC) and global (G-ACC) accuracy. Best results are **bold**; second-best results are underlined.

Method	SST-2		QNLI		MRPC		QQP		Average	
	L-ACC	G-ACC	L-ACC	G-ACC	L-ACC	G-ACC	L-ACC	G-ACC	L-Avg	G-Avg
FedIT	80.98	73.97	78.86	83.07	82.11	74.14	90.07	89.70	83.01	80.22
FedSA	90.03	90.71	80.11	79.39	88.39	80.29	88.61	87.87	86.78	84.57
FFA-LoRA	71.73	62.62	73.57	75.87	80.43	71.83	89.35	88.78	78.77	74.77
FedRot	**93.02**	**93.46**	79.35	81.91	82.77	74.84	89.98	89.26	86.28	84.87
FedALT	77.19	73.05	79.68	81.88	**89.01**	**89.01**	89.22	88.77	83.77	80.66
FlyLoRA	70.54	75.00	72.37	73.79	74.79	75.88	89.77	89.18	76.87	78.46
FlexLoRA	89.36	90.14	79.17	81.27	74.49	74.38	90.35	89.57	83.34	83.84
LoRA-XS	81.65	90.59	72.42	74.87	74.25	71.71	**91.06**	**90.59**	79.85	81.94
PiSSA	85.72	89.11	76.72	80.36	74.07	73.56	90.30	89.82	81.70	83.21
LoRA-MoE	87.21	90.94	79.76	81.26	82.99	73.85	90.18	89.52	85.04	83.89
LoRA-MoE+CKA	85.12	82.69	78.39	78.51	82.89	72.41	90.22	89.53	84.16	80.79
A-SMoE	92.16	92.77	**83.33**	**88.27**	83.71	74.14	89.23	88.71	**87.11**	**85.97**

each communication round. The global models are trained for a total of 20 communication rounds, with the local training set to 2 epochs per round.

The comprehensive evaluation results are summarized in Table 1. We compare our method, A-SMoE, against 11 baselines using both Local and Global Evaluation Accuracy. As demonstrated in the table, A-SMoE delivers highly competitive and robust performance across all evaluated tasks, culminating in the highest overall average accuracy. Specifically, A-SMoE achieves a Local Average of 87.11% and a Global Average of 85.97%, outperforming strong baselines such as FedSA (L-Avg: 86.78%) and FedRot (G-Avg: 84.87%). It is particularly noteworthy that in the QNLI task, A-SMoE exhibits a significant advantage, outperforming the best baseline by over 3.2% in L-ACC and 5.2% in G-ACC. While certain baselines may peak on specific tasks—such as FedRot on SST-2—they often suffer from severe performance degradation on other tasks. In contrast, A-SMoE maintains excellent generalization ability.

Natural Language Generation. To further validate the effectiveness of our proposed method on complex generation tasks, we fine-tune LLaMA2-7B-HF model (Wang et al., 2025) using the FLAN instruction tuning dataset (Longpre et al., 2023). We select 12 distinct tasks and distribute them across 12 clients, ensuring that each client is assigned a unique task. Due to this strict task-heterogeneous configuration, each client has a completely different objective, and therefore, no distinction is made between global and local evaluation in this experiment.

The 12 tasks span four categories: *Sentiment Analysis* (IMDB, SST-2, Sentiment140), *Commonsense Reasoning* (HellaSwag, PIQA, CosmosQA), *Struct-to-Text* (DART, E2E-NLG, CommonGen), and *Summarization* (AESLC, Gigaword, SAMSum).

For the federated training configuration, the global model is trained for a total of 20 communication rounds, with the local training set to 1 epoch per round and a client participation rate of 0.5. Following the protocol established in (Wang et al., 2025), we randomly sample 600 training instances from the local dataset of each selected client during every training round to improve efficiency and avoid severe overfitting.

The detailed evaluation results are presented in Table 2. We compare A-SMoE with 11 other baseline methods across the 12 tasks. Despite the extreme task heterogeneity across clients, A-SMoE achieves the highest average performance of 58.22%, surpassing strong baselines such as PiSSA (57.52%). Furthermore, A-SMoE demonstrates significant stability and balanced performance improvements across different task clusters. For instance, in the *Commonsense Reasoning* and *Summarization* tasks, our method consistently achieves top-tier results (e.g., 77.27% on HellaSwag). This further proves that our approach can effectively maintain knowledge retention and generalization capabilities.

The vision-related experiments can be found in Appendix F.1.

Table 2: Performance comparison on the FLAN benchmark across four categories of downstream tasks. **Bold** and underlined values indicate the best and second-best results, respectively.

Method	Sentiment Analysis			Commonsense Reasoning			Struct-to-Text			Summarization			Avg.
	IMDB	SST-2	S140	HellaSwag	PIQA	CosmosQA	DART	E2E-NLG	CommonGen	AESLC	Gigaword	SAMSum	
FedIT	**72.06**	73.73	49.18	68.26	74.53	61.45	65.85	69.79	43.13	31.90	**33.67**	40.05	56.97
FedSA	72.02	72.73	44.64	71.90	74.97	55.37	**69.27**	67.05	41.77	30.71	31.96	41.46	56.15
FFA-LoRA	68.95	73.92	46.45	72.22	75.69	58.33	67.15	69.20	43.58	27.48	31.72	40.75	56.29
FedRot	69.27	74.35	48.08	73.26	76.80	62.29	65.92	69.64	43.74	29.54	31.59	41.81	57.19
FedALT	67.37	73.76	45.83	75.09	**77.25**	59.79	66.27	69.63	43.94	29.87	31.88	40.83	56.79
FlyLoRA	67.12	74.17	48.18	68.06	76.38	58.40	65.21	69.66	**46.44**	30.10	32.36	39.90	56.33
FlexLoRA	70.71	74.51	44.27	69.14	75.46	60.31	68.06	68.62	43.26	31.93	32.66	41.16	56.67
LoRA-XS	68.31	**75.98**	49.07	74.31	75.51	61.25	66.82	68.19	44.34	32.31	32.17	39.27	57.29
PiSSA	68.78	75.70	47.52	77.11	75.98	63.12	66.85	69.34	43.55	28.16	33.30	40.83	57.52
LoRA-MoE	69.43	75.44	**49.24**	74.79	75.83	58.78	65.44	69.24	43.92	29.69	30.48	40.98	56.94
LoRA-MoE+CKA	68.03	72.13	48.33	75.09	74.44	59.24	64.71	68.89	44.03	27.22	31.20	39.58	56.07
A-SMoE	70.58	74.85	47.11	**77.27**	75.24	**63.62**	66.89	**70.18**	44.65	**32.55**	**33.94**	**41.92**	**58.22**

4.2 Ablation Study

To investigate the effectiveness of our proposed components, we conduct ablation studies on the GLUE benchmark. We primarily focus on module ablation and the impact of the expert number.

Module Ablation. We evaluate the necessity of our alignment designs by removing the scalar alignment (NoS) and the aggregation alignment (NoA). As shown in Table 3, removing either of these modules leads to a performance drop. The full A-SMoE framework achieves the highest L-Avg and G-Avg, confirming that both alignments are essential.

Impact of Expert Number. We vary the number of experts (E_N), with the default set to 4. As presented in Table 4, the performance reaches its peak around 4 and 8. We choose $E_N = 4$ as the default configuration to maintain the highest overall accuracy while ensuring computational efficiency. In addition, analyses of the robustness of our method, including the effects of different ranks, numbers of clients, and client participation rates, can be found in the robustness analysis in Appendix F.2.

Table 3: Module ablation study. NoS: removing scalar alignment; NoA: removing aggregation alignment.

Method	L-Avg	G-Avg
A-SMoE (Full)	**87.11**	**85.97**
A-SMoE (NoS)	86.12	84.86
A-SMoE (NoA)	86.76	85.21
A-SMoE (NoS+NoA)	85.72	84.55

Table 4: Impact of the number of experts (E) on A-SMoE performance.

# Experts (E)	L-Avg	G-Avg
$E_N = 1$	85.61	84.14
$E_N = 2$	86.35	84.75
$E_N = 4$ (Default)	**87.11**	85.97
$E_N = 8$	87.03	**86.02**
$E_N = 16$	85.25	85.03

4.3 Singular-Direction Allocation and Routing Analysis

Singular-direction allocation. We first study how singular directions should be assigned to different experts. As motivated by the experiment in Appendix B, we compare six singular-direction allocation strategies. **Head** consecutively partitions the dominant singular directions, while **Round-Robin + Head** distributes them in an interleaved manner. **Head[1]-Tail[E_N-1]** uses one head expert and $E_N - 1$ tail experts, whereas **Head[E_N-1]-Tail[1]** uses $E_N - 1$ head experts and one tail expert. **[Round-Robin+Head[E_N-1]]-Tail[1]** applies interleaving only among the head experts while preserving separate head and tail experts, whereas **Round-Robin+Head[E_N-1]-Tail[1]** interleaves both head and tail directions across experts.

The results in Table 5 show that introducing tail singular directions consistently degrades performance, confirming that adaptation should mainly focus on directions associated with large singular

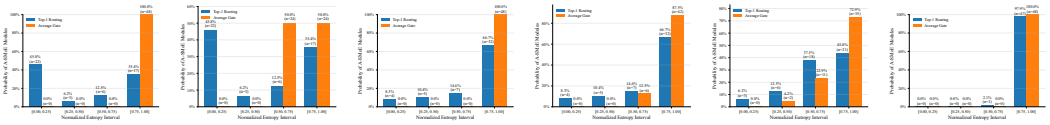
Table 5: Ablation studies on singular-direction allocation and routing mechanisms. In (b), PF denotes our parameter-free router, while Learner denotes a learned linear layer used as the router.

(a) Singular-Direction Allocation

Allocation Method	L-Avg	G-Avg
Head	86.66	84.71
Round-Robin + Head	**87.11**	**85.97**
Head[1]-Tail[E_N-1]	69.91	65.83
Head[E_N-1]-Tail[1]	69.32	65.72
[Round-Robin+Head[E_N-1]]-Tail[1]	70.15	65.55
Round-Robin+Head[E_N-1]-Tail[1]	72.32	70.29

(b) Routing Mechanism

Allocation	Router	L-Avg	G-Avg	Param.
Round-Robin + Head	Softmax + PF	87.11	**85.97**	0.002213
	Top-2 + PF	**87.60**	85.15	0.002213
Head	Softmax + PF	86.66	84.71	0.002213
	Top-2 + PF	86.73	84.45	0.002213
Round-Robin + Head	Softmax + Learner	85.51	83.52	0.002764
	Top-2 + Learner	83.34	81.95	0.002764



(a) Head + PF, Softmax (b) Head + PF, Top-2 (c) Round-Robin + PF, Softmax (d) Round-Robin + PF, Top-2 (e) Round-Robin + Learned, Softmax (f) Round-Robin + Learned, Top-2

Figure 3: Layer-wise routing balance under different strategies. Normalized routing entropy is divided into four intervals: $[0, 0.25)$, $[0.25, 0.5)$, $[0.5, 0.75)$, and $[0.75, 1]$. Each bar reports the proportion of A-SMoE layers falling into the corresponding interval, where larger entropy indicates more balanced expert utilization.

values. Among the dominant-direction variants, the round-robin allocation achieves the best overall performance. To understand why the round-robin allocation is effective, we further analyze its routing behavior in the following experiments.

Routing analysis. For each A-SMoE layer, we independently evaluate its routing balance over all samples in the evaluation set. Specifically, we collect the routing probabilities of all valid tokens, where N denotes the total number of tokens. For expert E_e, we compute the average routing mass $\bar{g}_e = \frac{1}{N}\sum_{t=1}^{N} g_{t,e}$ and the Top-1 usage $p_e^{\text{top1}} = \frac{1}{N}\sum_{t=1}^{N} \mathbb{I}[e = \arg\max_j g_{t,j}]$. The routing balance of each layer is then quantified by the normalized entropy

$$\mathcal{H}(\mathbf{p}) = -\frac{1}{\log E_N}\sum_{e=1}^{E_N} p_e \log(p_e + \epsilon), \tag{6}$$

a larger entropy indicates a more balanced routing distribution. As shown in Fig. 3, the round-robin allocation produces smoother routing distributions than the contiguous allocation, which helps explain its effectiveness. Notably, our parameter-free router exhibits a more balanced routing distribution than the learned linear router with Softmax, while requiring neither additional routing parameters nor auxiliary load-balancing loss. In contrast, the learned linear router with Top-2 routing yields the most balanced routing distribution because an additional load-balancing loss (Du et al., 2022) is applied. However, this stronger balance does not translate into better task performance; instead, it achieves the lowest accuracy in Table 5. Moreover, the learned router introduces additional trainable parameters and an independent routing function, which can cause routing misalignment across clients in federated learning. This misalignment hinders effective optimization, so the extra routing parameters do not translate into better performance.

5 Conclusion

We presented A-SMoE, an alignment-aware framework for federated MoE fine-tuning. By leveraging pretrained intrinsic subspaces as shared representation references and introducing parameter-free routing, A-SMoE improves expert alignment and aggregation compatibility without additional routing overhead. Extensive experiments demonstrate consistent improvements over existing federated fine-tuning methods, highlighting the effectiveness of intrinsic-subspace alignment for efficient and robust federated adaptation.

AI USE STATEMENT

In this work, we used generative AI tools to assist with editing the manuscript for grammar, clarity, and readability, and with writing a limited portion of the experimental code involving routine library/API calls and experimental scripting. The research methodology, experimental design, and evaluation protocols were developed by the authors. We have not used generative AI tools to formulate research hypotheses, develop theoretical or conceptual contributions, design the proposed method, interpret experimental results, or draw scientific conclusions. All AI-assisted code was manually reviewed and tested by the authors, and all AI-assisted text was carefully checked for technical accuracy and consistency with the original scientific content. We take responsibility for the final content of this work, including all text, claims, code, experimental results, and other artifacts produced with the aid of generative AI.

ACKNOWLEDGMENTS

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Sara Babakniya, Ahmed Roushdy Elkordy, Yahya H Ezzeldin, Qingfeng Liu, Kee-Bong Song, Mostafa El-Khamy, and Salman Avestimehr. Slora: Federated parameter efficient fine-tuning of language models. *arXiv preprint arXiv:2308.06522*, 2023.

Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. Federated fine-tuning of large language models under heterogeneous tasks and client resources. *Advances in Neural Information Processing Systems*, 37:14457–14483, 2024.

Klaudia Bałazy, Mohammadreza Banaei, Karl Aberer, and Jacek Tabor. Lora-xs: Low-rank adaptation with extremely small number of parameters, 2024. *URL https://arxiv. org/abs/2405.17604*.

Jieming Bian, Lei Wang, Letian Zhang, and Jie Xu. Fedalt: Federated fine-tuning through adaptive local training with rest-of-world lora. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 19728–19736, 2026.

Shuangyi Chen, Yue Ju, Hardik Dalal, Zhongwen Zhu, and Ashish Khisti. Robust federated finetuning of foundation models via alternating minimization of lora. *arXiv preprint arXiv:2409.02346*, 2024.

Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11828–11837, 2023.

Anzhe Cheng, Shukai Duan, Shixuan Li, Chenzhong Yin, Mingxi Cheng, Heng Ping, Tamoghna Chattopadhyay, Sophia I Thomopoulos, Shahin Nazarian, Paul Thompson, et al. Ermoe: Eigen-reparameterized mixture-of-experts for stable routing and interpretable specialization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12997–13006, 2026.

Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Yu Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1280–1297, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Jun Zhao, Wei Shen, Yuhao Zhou, Zhiheng Xi, Xiao Wang, Xiaoran Fan, et al. Loramoe: Alleviate world knowledge forgetting in large language models via moe-style plugin. *arXiv preprint arXiv:2312.09979*, 2023.

Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Wei Shen, Limao Xiong, Yuhao Zhou, Xiao Wang, Zhiheng Xi, Xiaoran Fan, et al. Loramoe: Alleviating world knowledge forgetting in large language models via moe-style plugin. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1932–1945, 2024.

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. Glam: Efficient scaling of language models with mixture-of-experts. In *International conference on machine learning*, pp. 5547–5569. PMLR, 2022.

Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.

Pengxin Guo, Shuang Zeng, Yanran Wang, Huijie Fan, Feifei Wang, and Liangqiong Qu. Selective aggregation for low-rank adaptation in federated learning. In *International Conference on Learning Representations*, volume 2025, pp. 99003–99027, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International conference on machine learning*, pp. 5583–5594. PMLR, 2021.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *International conference on machine learning*, pp. 22631–22648. PMLR, 2023.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521, 2022.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

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

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37:121038–121072, 2024.

Basil Mustafa, Carlos Riquelme, Joan Puigcerver, Rodolphe Jenatton, and Neil Houlsby. Multimodal contrastive learning with limoe: the language-image mixture of experts. *Advances in Neural Information Processing Systems*, 35:9564–9576, 2022.

Mengye Ren, Ryan Kiros, and Richard Zemel. Exploring models and data for image question answering. *Advances in neural information processing systems*, 28, 2015.

Zerui Shao, Beibei Li, Zhibo Wang, Yanbing Yang, Peiran Wang, and Jun Luo. Fedufd: personalized edge computing using federated uncertainty-driven feature distillation. In *IEEE INFOCOM 2025-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2025.

HZ SHI, J ZHAO, YQ ZHAO, et al. Survey on system optimization for mixture of experts in the era of large models. *Journal of Computer Research and Development*, 62(5):1164–1189, 2025.

Raghav Singhal, Kaustubh Ponkshe, and Praneeth Vepakomma. Fedex-lora: Exact aggregation for federated and efficient fine-tuning of large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1316–1336, 2025.

Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

Youbang Sun, Zitao Li, Yaliang Li, and Bolin Ding. Improving lora in privacy-preserving federated learning. *arXiv preprint arXiv:2403.12313*, 2024.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Lei Wang, Jieming Bian, Letian Zhang, and Jie Xu. Adaptive lora experts allocation and selection for federated fine-tuning. *arXiv preprint arXiv:2509.15087*, 2025.

Zihan Wang, Deli Chen, Damai Dai, Runxin Xu, Zhuoshu Li, and Yu Wu. Let the expert stick to his last: Expert-specialized fine-tuning for sparse architectural large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 784–801, 2024.

Fei Wu, Jia Hu, Geyong Min, and Shiqiang Wang. Efficient orthogonal fine-tuning with principal subspace adaptation. In *International Conference on Learning Representations*, volume 2026, pp. 25798–25826, 2026.

Li Yuan, Yi Cai, Xudong Shen, Qing Li, Qingbao Huang, Zikun Deng, and Tao Wang. Collaborative multi-lora experts with achievement-based multi-tasks loss for unified multimodal information extraction. *arXiv preprint arXiv:2505.06303*, 2025.

Haoran Zhang, Dongjun Kim, Seohyeon Cha, and Haris Vikalo. Fedrot-lora: Mitigating rotational misalignment in federated lora. *arXiv preprint arXiv:2602.23638*, 2026.

Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. Towards building the federatedgpt: Federated instruction tuning. In *ICASSP 2024-2024 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 6915–6919. IEEE, 2024.

Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582*, 2018.

Heming Zou, Yunliang Zang, Wutong Xu, Yao Zhu, and Xiangyang Ji. Flylora: Boosting task decoupling and parameter efficiency via implicit rank-wise mixture-of-experts. *Advances in Neural Information Processing Systems*, 38:10386–10419, 2026.

A CKA-BASED ANALYSIS OF CROSS-CLIENT EXPERT ALIGNMENT

A.1 EXPERIMENTAL SETUP

We use Centered Kernel Alignment (CKA) to investigate whether experts at the same architectural position remain functionally aligned across clients after local training. The analysis is performed after local optimization and before server aggregation, such that the measured discrepancy directly reflects the expert specialization. In our implementation, the number of expert parameters is based on a LoRA configuration with a rank of $r = 8$. Both the A-SMoE and LoRA-MoE modules utilize 4 experts. Meanwhile, to ensure a fair comparison, Top-1 routing is uniformly applied across all settings. We utilize the RoBERTa-Large model and evaluate it on the SST-5 dataset. The decay factor is set to 1e-4. The search space for the learning rate is defined as $\eta \in \{1\text{e-}2, 1\text{e-}3, 1\text{e-}4\}$. The training process consists of 10 clients in total and lasts for 30 communication rounds, with a client participation rate of 50% per round. Additionally, We use two data partitions: uniform IID and Non-IID (Dirichlet $\alpha = 0.1$). The table on the right side of Figure 1 reports the global accuracies(G-ACC) and local accuracies(L-ACC) achieved by each method, alongside the proportion of trainable parameters.

For CKA evaluation, we construct a fixed reference set $\mathcal{D}_{\text{ref}} = \{x_n\}_{n=1}^{N}$ containing $N = 500$ examples sampled approximately uniformly across classes from the development set. The identical reference set and sample ordering are used for all clients and experts throughout the analysis. For client c, Transformer layer ℓ, projection $p \in \{Q, V\}$, and expert e, we collect its output on each reference example. Given the token-level expert output $\mathbf{H}_n^{(c,\ell,p,e)} \in \mathbb{R}^{T_n \times d}$, we apply mean pooling over the token dimension,

$$\mathbf{h}_n^{(c,\ell,p,e)} = \frac{1}{T_n} \sum_{t=1}^{T_n} \mathbf{H}_{n,t,:}^{(c,\ell,p,e)}, \tag{7}$$

and stack the pooled representations to obtain

$$\mathbf{X}^{(c,\ell,p,e)} = \begin{bmatrix} (\mathbf{h}_1^{(c,\ell,p,e)})^\top \\ \vdots \\ (\mathbf{h}_N^{(c,\ell,p,e)})^\top \end{bmatrix} \in \mathbb{R}^{N \times d}. \tag{8}$$

All experts are explicitly evaluated on the same reference inputs, independent of whether they are selected by the Top-1 router. We compute linear CKA following the Gram-matrix formulation used in our implementation. For an expert representation matrix $\mathbf{X}$, we first center the features across reference samples:

$$\widetilde{\mathbf{X}} = \mathbf{X} - \mathbf{1}\boldsymbol{\mu}^\top, \qquad \boldsymbol{\mu} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{X}_{n,:}. \tag{9}$$

The corresponding linear Gram matrix is $\mathbf{K}_X = \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\top$, which is normalized by its Frobenius norm as

$$\widehat{\mathbf{K}}_X = \frac{\mathbf{K}_X}{\|\mathbf{K}_X\|_F + \epsilon}, \qquad \epsilon = 10^{-12}. \tag{10}$$

For two experts represented by $\mathbf{X}$ and $\mathbf{Y}$, their linear CKA similarity is then

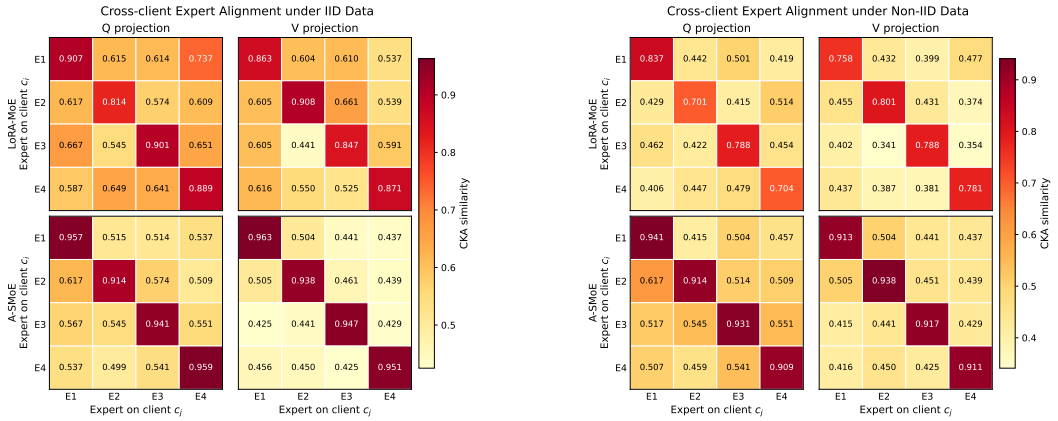
$$\text{CKA}(\mathbf{X}, \mathbf{Y}) = \left\langle \widehat{\mathbf{K}}_X, \widehat{\mathbf{K}}_Y \right\rangle_F. \tag{11}$$

A higher CKA value indicates that the two experts induce more similar representational structures on the shared reference inputs. For each pair of participating clients (c_i, c_j) and each identical architectural position (ℓ, p), we construct an $E \times E$ cross-client similarity matrix

$$S_{ab}^{(c_i,c_j,\ell,p)} = \text{CKA}\left(\mathbf{X}^{(c_i,\ell,p,a)}, \mathbf{X}^{(c_j,\ell,p,b)}\right), \qquad a, b \in \{1, \ldots, E\}. \tag{12}$$

The diagonal entries S_{ee} measure the similarity between experts with the same index, whereas off-diagonal entries measure cross-index similarity. We quantify the direct same-index alignment by

$$S_{\text{diag}} = \frac{1}{E} \sum_{e=1}^{E} S_{ee}. \tag{13}$$



(a) IID setting.

(b) Non-IID setting.

Figure 4: Cross-client expert alignment measured by CKA under **IID** and **Non-IID** data distributions.

A.2 Other Results

In Figure 4, we report the cross-client CKA similarity of experts across various positions within each layer. Under the IID setting, our method exhibits higher similarity for identically positioned experts and lower similarity for differently positioned ones. This indicates that the U_{E_e} and V_{N_e} of different experts are orthogonal to each other, ensuring that the outputs of different experts are strictly orthogonal, effectively mitigating expert redundancy. Furthermore, under the Non-IID setting, while the cross-position expert CKA similarity achieved by our method is generally on par with that of LoRA-MoE, our corresponding-expert CKA experiences a much smaller degradation compared to LoRA-MoE when shifting from the IID setting. This demonstrates the stability of our approach.

B Detailed Analysis of LoRA-SVD

Previous methods (**?**Meng et al., 2024; Wu et al., 2026) that leverage pretrained weights typically assume that the directions most relevant to downstream tasks correspond to the dominant singular vectors, those associated with the largest singular values. However, this assumption has not been empirically validated. To deeply understand the inner workings of LoRA and identify which implicit directions in the pre-trained weights are most heavily updated during fine-tuning, we conduct an extensive SVD alignment analysis. Specifically, we project the learned LoRA weight updates onto the singular vector space of the original pre-trained weights and evaluate their correlation across six distinct quantitative metrics.

B.1 Experimental Setup

In this analysis, we employ the **RoBERTa-Large** architecture as our backbone model. LoRA modules are injected into the query (W_q) and value (W_v) projection matrices across all transformer layers, and the rank is set to 8. We evaluate the alignment phenomenon on four downstream tasks from the GLUE benchmark: **SST-2, QNLI, MRPC, and QQP**. For all four tasks, the models are fine-tuned for 20 epochs with a learning rate of 1×10^{-4} and a weight decay factor of 1×10^{-4}.

B.2 Mathematical Formulation

Let $W \in \mathbb{R}^{d_{out} \times d_{in}}$ denote a pre-trained weight matrix in the model. Its Singular Value Decomposition (SVD) is given by $W = U\Sigma V^\top$, where $U = [u_1, \ldots, u_{d_{out}}]$ and $V = [v_1, \ldots, v_{d_{in}}]$ are the orthogonal matrices representing the output and input vector spaces, respectively, and $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_{\min(d_{out}, d_{in})})$ contains the original singular values in descending order. Similarly, let $\Delta W = BA$ represent the learned LoRA weight update. The SVD of this update is

$\Delta W = \tilde{U}\tilde{\Sigma}\tilde{V}^\top$, where $\tilde{U} = [\tilde{u}_1, \ldots, \tilde{u}_r]$ and $\tilde{V} = [\tilde{v}_1, \ldots, \tilde{v}_r]$ are the LoRA singular vectors, and $\tilde{\Sigma} = \text{diag}(\tilde{\sigma}_1, \ldots, \tilde{\sigma}_r)$ contains the LoRA singular values, with r being the LoRA rank. To evaluate the alignment in a specific space $P \in \{U, V\}$ (representing either the output space U or the input space V), we compute the absolute cosine similarity between the k-th LoRA singular vector $\tilde{p}_k$ and the i-th original singular vector p_i:

$$C^P_{k,i} = \frac{|\langle \tilde{p}_k, p_i \rangle|}{\|\tilde{p}_k\|_2 \|p_i\|_2} = |\langle \tilde{p}_k, p_i \rangle| \tag{14}$$

For each LoRA direction $k \in \{1, \ldots, r\}$, we identify the Top1 matched original singular direction index as $i^*(k) = \arg\max_i C^P_{k,i}$. Let $\mathcal{K}^P_i = \{k \mid i^*(k) = i\}$ denote the set of LoRA singular vector indices that are optimally aligned with the i-th original singular vector. The size $|\mathcal{K}^P_i|$ represents the number of LoRA vectors that assign their highest similarity to the i-th original direction. If an original singular vector i is not matched by any LoRA direction, $\mathcal{K}^P_i = \emptyset$.

B.3 Evaluation Metrics

Because multiple LoRA singular vectors may dominantly align with the same original singular vector, we accumulate their contributions by summing them and then normalize by the total LoRA rank r. This formulation rigorously captures the average energy each original direction receives from the LoRA update. We define six metrics (S_1 to S_6) for a given original rank i in space P:

- S_1**: Top-1 Match Frequency.** The proportion of LoRA directions that find their best match with the i-th original direction.

$$S^P_1(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} 1 = \frac{|\mathcal{K}^P_i|}{r} \tag{15}$$

- S_2**: Match $\times$ Original SV.** Evaluates the normalized frequency scaled by the importance (singular value) of the matched original direction.

$$S^P_2(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} \sigma_i = \frac{|\mathcal{K}^P_i|}{r} \cdot \sigma_i \tag{16}$$

- S_3**: Match $\times$ LoRA SV.** Measures the normalized accumulated magnitude of the LoRA updates applied to the matched direction.

$$S^P_3(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} \tilde{\sigma}_k \tag{17}$$

- S_4**: Similarity $\times$ Original SV.** A soft-matching extension of S_2, weighted by the actual alignment precision (cosine similarity).

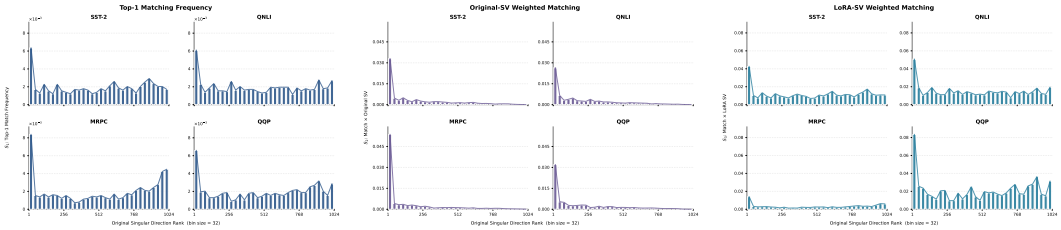
$$S^P_4(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} \left(C^P_{k,i} \cdot \sigma_i \right) \tag{18}$$

- S_5**: Similarity $\times$ LoRA SV.** A soft-matching extension of S_3, representing the normalized effective projection of the LoRA update magnitude onto the original direction.

$$S^P_5(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} \left(C^P_{k,i} \cdot \tilde{\sigma}_k \right) \tag{19}$$

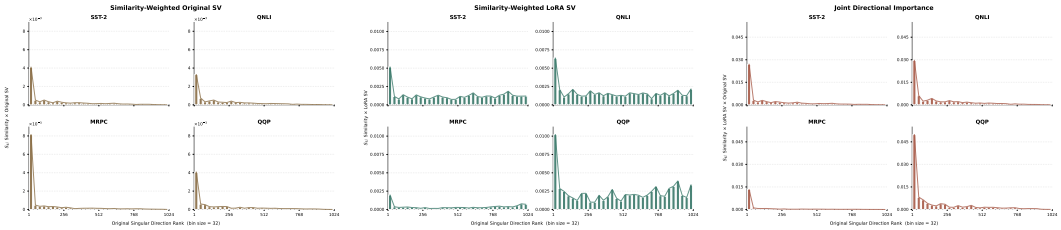
- S_6**: Similarity $\times$ LoRA SV $\times$ Original SV.** The most comprehensive metric, capturing the normalized accumulated energy while considering alignment precision, LoRA update magnitude, and the intrinsic importance of the original feature direction.

$$S^P_6(i) = \frac{1}{r} \sum_{k \in \mathcal{K}^P_i} \left(C^P_{k,i} \cdot \tilde{\sigma}_k \cdot \sigma_i \right) \tag{20}$$



(a) S_1: Top-1 Match Frequency (b) S_2: Match × Original SV (c) S_3: Match × LoRA SV

Figure 5: Analysis of LoRA directional preference based on Top-1 matching. From left to right, the three metrics measure matching frequency, pretrained singular-value importance, and LoRA update strength.



(a) S_4: Similarity × Original SV (b) S_5: Similarity × LoRA SV (c) S_6: Similarity × LoRA SV × Original SV

Figure 6: Similarity-aware analysis of LoRA update directions. The three metrics progressively incorporate pretrained singular-value importance, LoRA update strength, and their joint contribution.

Note that for all metrics, if $\mathcal{K}_i^P = \emptyset$, the sum evaluates to 0. To obtain a holistic view of the parameter update, the final score for the i-th original singular rank is computed by summing the contributions from both the output (U) and input (V) spaces:

$$S_m(i) = S_m^U(i) + S_m^V(i), \quad \text{for } m \in \{1, 2, 3, 4, 5, 6\} \tag{21}$$

For robust visualization across the high-dimensional singular space, we group the original singular ranks into adjacent bins of size 32. The metric scores within each bin are averaged, yielding the final distribution plots.

B.4 Results and Observations

The empirical results across the four GLUE tasks (SST-2, QNLI, MRPC, QQP) are presented in Figures 5 to 6. Consistent with findings in classic parameter-efficient fine-tuning methodologies, our analysis reveals a strong structural bias in the LoRA updates.

These six metrics consistently demonstrate that the directions most relevant to downstream tasks align with the largest singular values, empirically validating hypotheses from prior work. Interestingly, specific metrics (e.g., S_1 and S_3) indicate that the directions corresponding to the smallest singular values (i.e., the tail end) also exhibit relatively high similarity to downstream tasks. However, as we will demonstrate in the ablation study, explicitly focusing on these tail directions actually results in severe performance degradation.

C Detailed Analysis of Scale Misalignment

Scale misalignment is a critical but often overlooked issue when aggregating factorized linear modules in federated fine-tuning. This ambiguity arises because the intermediate latent dimensions can be arbitrarily scaled by a constant and its reciprocal without altering the end-to-end mathematical function.

To illustrate this problem without loss of generality, we analyze the most fundamental unit of neural network modules: two vectors multiplied together. Mathematically, any $n \times n$ matrix can be decomposed into the sum of n distinct matrix products, where each product is formed by an $n \times 1$ matrix multiplied by a $1 \times n$ matrix. Therefore, examining a single fundamental unit—a module constructed by multiplying an $n \times 1$ vector a and a $1 \times n$ vector b—is fully sufficient to provide a generalized understanding of the scaling behavior.

An Illustrative Example of Scale Misalignment. Consider a federated learning scenario with two clients optimizing this fundamental unit. Suppose both clients successfully learn the exact same underlying representation, which conceptually corresponds to the product ab. However, due to scale ambiguity, their locally updated parameters may appear vastly different:

$$\text{Client 1 learns:} \quad a_1 = \frac{1}{2}a, \quad b_1 = 2b \quad \Rightarrow \quad a_1 b_1 = ab \tag{22}$$

$$\text{Client 2 learns:} \quad a_2 = \frac{1}{6}a, \quad b_2 = 6b \quad \Rightarrow \quad a_2 b_2 = ab \tag{23}$$

Both clients essentially capture the identical mapping ab. Ideally, their aggregation should preserve this exact mapping. However, applying naive Federated Averaging (McMahan et al., 2017) directly to the factors yields:

$$\bar{a} = \frac{1}{2}\left(\frac{1}{2}a + \frac{1}{6}a\right) = \frac{1}{3}a \tag{24}$$

$$\bar{b} = \frac{1}{2}\left(2b + 6b\right) = 4b \tag{25}$$

When these averaged factors are multiplied during inference, the resulting aggregated module becomes:

$$\bar{a}\bar{b} = \left(\frac{1}{3}a\right)(4b) = \frac{4}{3}ab \neq ab \tag{26}$$

This breaks the original functional symmetry. Despite both clients learning the identical optimal structure, the naive averaging drastically distorts the aggregated output due to unconstrained local scaling.

Resolution via Norm Transfer. To mitigate this issue, we propose standardizing the scale of the modules prior to transmission. Specifically, a client divides the first module a by its L_2 norm and multiplies the subsequent module b by the same norm. Applying this to our previous example:

$$\text{Client 1:} \quad \tilde{a}_1 = \frac{a_1}{\|a_1\|_2} = \frac{a}{\|a\|_2}, \quad \tilde{b}_1 = \|a_1\|_2 b_1 = \|a\|_2 b \tag{27}$$

$$\text{Client 2:} \quad \tilde{a}_2 = \frac{a_2}{\|a_2\|_2} = \frac{a}{\|a\|_2}, \quad \tilde{b}_2 = \|a_2\|_2 b_2 = \|a\|_2 b \tag{28}$$

Through this explicit norm transfer, $\tilde{a}_1$ and $\tilde{a}_2$ are forced into identical unit-norm representations, while their corresponding scales are completely absorbed by $\tilde{b}_1$ and $\tilde{b}_2$. Subsequent federated aggregation on these aligned components results exactly in $(\frac{a}{\|a\|_2})(\|a\|_2 b) = ab$, perfectly recovering the shared knowledge and eliminating the scale misalignment.

Application to A-SMoE. The insight from this fundamental example directly explains the scale misalignment observed in A-SMoE's expert formulation $f_{E_e}(x) = xU_e \operatorname{diag}(S^{in}_{E_e}) R_e \operatorname{diag}(S^{out}_{E_e}) V_e$. In this structure, the central matrix R_e is constrained to the orthogonal group, meaning it intrinsically preserves its magnitude and does not contribute to scale ambiguity. However, the scalar degrees of freedom still exist between the two trainable diagonal matrices $S^{in}_{E_e}$ and $S^{out}_{E_e}$. Crucially, our analysis demonstrates that scale misalignment can propagate *across* intermediate modules—even though $S^{in}_{E_e}$ and $S^{out}_{E_e}$ are separated by the orthogonal processor R_e, the linear nature of R_e allows a scaling factor c to pass through as $(c \cdot S^{in}_{E_e}) R_e (\frac{1}{c} \cdot S^{out}_{E_e}) = S^{in}_{E_e} R_e S^{out}_{E_e}$.

Therefore, before the federated aggregation phase, A-SMoE applies the exact same norm-transfer technique to resolve this cross-module ambiguity. We normalize the input scaling vector $S^{in}_{E_e}$ by

dividing it by its L_2 norm ($\tilde{S}^{in}_{E_e} = S^{in}_{E_e}/\|S^{in}_{E_e}\|_2$) and transfer its magnitude to the output factor ($\tilde{S}^{out}_{E_e} = \|S^{in}_{E_e}\|_2 S^{out}_{E_e}$). Mathematically, the overall network output remains strictly unchanged, but the scale ambiguity across clients is systematically eliminated prior to averaging.

D Proof of Aggregation Alignment

We prove the result for an arbitrary expert e; the same argument applies independently to every expert.

D.1 Closed-form Solution for S^{out}_e

Let $A_e = \widehat{D}^{in}_e \widehat{R}_e$ and write $D = \mathrm{diag}(S_1, \ldots, S_{r_e})$. For expert e, the remaining optimization is

$$\widehat{D}^{out}_e = \arg \min_{D=\mathrm{diag}(S),\, S\geq\epsilon} \|W^\star_e - A_e D\|_F^2. \tag{29}$$

Let $a_{e,j} = (A_e)_{:,j}$ and $w^\star_{e,j} = (W^\star_e)_{:,j}$. Since right multiplication by a diagonal matrix independently scales each column,

$$\|W^\star_e - A_e D\|_F^2 = \sum_{j=1}^{r_e} \|w^\star_{e,j} - S_j a_{e,j}\|_2^2. \tag{30}$$

Hence, the problem decomposes into r_e independent scalar minimizations. For

$$f_{e,j}(S_j) = \|w^\star_{e,j} - S_j a_{e,j}\|_2^2, \tag{31}$$

we have

$$f_{e,j}(S_j) = \|w^\star_{e,j}\|_2^2 - 2S_j a^\top_{e,j} w^\star_{e,j} + S_j^2 \|a_{e,j}\|_2^2. \tag{32}$$

Its first derivative is

$$f'_{e,j}(S_j) = -2a^\top_{e,j} w^\star_{e,j} + 2S_j \|a_{e,j}\|_2^2, \tag{33}$$

while

$$f''_{e,j}(S_j) = 2\|a_{e,j}\|_2^2 > 0. \tag{34}$$

Therefore, the objective is strictly convex and its unconstrained minimizer is

$$S^{\mathrm{unc}}_{e,j} = \frac{a^\top_{e,j} w^\star_{e,j}}{\|a_{e,j}\|_2^2}. \tag{35}$$

If $S^{\mathrm{unc}}_{e,j} \geq \epsilon$, this solution is feasible and remains the constrained optimum. Otherwise, strict convexity implies that the constrained optimum is attained at the positive boundary $S_j = \epsilon$. Consequently,

$$\boxed{\widehat{S}^{out}_{e,j} = \max\left\{\epsilon,\ \frac{a^\top_{e,j} w^\star_{e,j}}{\|a_{e,j}\|_2^2}\right\}.} \tag{36}$$

In particular, if $a^\top_{e,j} w^\star_{e,j} \leq 0$, the unconstrained solution is non-positive and the optimal feasible output scale is ϵ.

D.2 Proof of Theorem 1

For expert e, define $\bar{D}^{in}_e = \sum_k p_k D^{in}_{e,k}, c_e = \|\,\mathrm{diag}(\bar{D}^{in}_e)\|_2$. Since all $D^{in}_{e,k}$ are strictly positive diagonal matrices, $c_e > 0$. By construction,

$$\widehat{D}^{in}_e = \frac{\bar{D}^{in}_e}{c_e}, \qquad \bar{D}^{in}_e = c_e \widehat{D}^{in}_e. \tag{37}$$

Under the same projected orthogonal factor $\widehat{R}_e$, conventional factor-wise averaging gives

$$W_{\mathrm{avg},e} = \bar{D}^{in}_e \widehat{R}_e \bar{D}^{out}_e, \qquad \bar{D}^{out}_e = \sum_k p_k D^{out}_{e,k}. \tag{38}$$

Using Eq. equation 37,

$$W_{\text{avg},e} = c_e \widehat{D}_e^{in} \widehat{R}_e \bar{D}_e^{out} = \widehat{D}_e^{in} \widehat{R}_e (c_e \bar{D}_e^{out}).$$

Because every $D_{e,k}^{out}$ is strictly positive, $\bar{D}_e^{out}$ is also strictly positive. Let $\bar{S}_e^{out}$ be the diagonal vector of matrix $\bar{D}_e^{out}$. Choosing

$$0 < \epsilon \leq \min_j \left(c_e \bar{S}_{e,j}^{out} \right) \tag{39}$$

ensures that $c_e \bar{D}_e^{out}$ is a feasible point of Eq. equation 29. Since $\widehat{D}_e^{out}$ is the global minimizer of this constrained problem,

$$\|W_e^\star - \widehat{D}_e^{in} \widehat{R}_e \widehat{D}_e^{out}\|_F^2 \leq \|W_e^\star - \widehat{D}_e^{in} \widehat{R}_e (c_e \bar{D}_e^{out})\|_F^2. \tag{40}$$

Using Eq. equation 39, we obtain

$$\boxed{\|W_e^\star - W_{\text{ours},e}\|_F^2 \leq \|W_e^\star - W_{\text{avg},e}\|_F^2.} \tag{41}$$

Since the column-wise objective in Eq. equation 30 is strictly convex, equality holds only when

$$\widehat{D}_e^{out} = c_e \bar{D}_e^{out}. \tag{42}$$

Otherwise,

$$\|W_e^\star - W_{\text{ours},e}\|_F^2 < \|W_e^\star - W_{\text{avg},e}\|_F^2. \tag{43}$$

This proves Theorem 1.

Finally, because the experts occupy mutually orthogonal intrinsic subspaces, the reconstruction errors of different experts are orthogonal. Therefore, the per-expert inequalities can be summed as

$$\boxed{\sum_{e=1}^{E_N} \|W_e^\star - W_{\text{ours},e}\|_F^2 \leq \sum_{e=1}^{E_N} \|W_e^\star - W_{\text{avg},e}\|_F^2.} \tag{44}$$

Hence, the dominance relation also holds for the complete multi-expert aggregation.

E Complete A-SMoE Procedure

The proposed A-SMoE framework consists of three stages: local expert adaptation, client-side scale alignment, and server-side aggregation alignment. During local optimization, each client updates its expert-specific processors together with the task objective while preserving the orthogonal structure of the rotation matrices. Specifically, the local objective is defined as

$$\mathcal{L}_{\text{local}} = \mathcal{L}_{\text{task}} + \lambda_{\text{orth}} \mathcal{L}_{\text{orth}}, \tag{45}$$

where

$$\mathcal{L}_{\text{orth}} = \sum_{e=1}^{E_N} \left\| R_e^\top R_e - I \right\|_F^2 \tag{46}$$

encourages each expert R_e matrix to remain close to the orthogonal manifold. each client performs scale alignment independently for every expert. The default λ_{orth} used in all experiments is 10^{-4}.

The aligned expert parameters are then uploaded to the server. For each expert, the server performs aggregation alignment. The $\widehat{S}_{N_e}^{in}$ is averaged and renormalized to recover the unit-norm constraint, the $\widehat{R}_e$ matrices are averaged and projected back to the orthogonal group, and $\widehat{S}_{N_e}^{out}$ is obtained by solving the closed-form reconstruction problem. Finally, the updated expert processors are broadcast to all clients for the next communication round.

The diagonal scale factors are parameterized in the unconstrained space and mapped to positive values through the Softplus function:

$$S_{e,k}^{in} = \text{softplus}(\theta_{e,k}^{in}), \qquad S_{e,k}^{out} = \text{softplus}(\theta_{e,k}^{out}). \tag{47}$$

Algorithm 1 A-SMoE Algorithm Process

Require: Initial global parameters Θ^0, client set $\mathcal{K}$, expert number E_N, communication rounds T.

for $t = 0, \ldots, T-1$ **do**
  Client-side local optimization
  for each client $k \in \mathcal{K}$ **do**
    Receive global latent parameters $\{\theta_e^{in}, \theta_e^{out}, R_e\}_{e=1}^{E_N}$.
    Convert latent scale parameters into positive scales:

$$S_e^{in} = \mathrm{softplus}(\theta_e^{in}), \quad S_e^{out} = \mathrm{softplus}(\theta_e^{out}).$$

    Perform local fine-tuning by minimizing the task loss and orthogonal regularization loss:

$$\mathcal{L} = \mathcal{L}_{\mathrm{task}} + \lambda_{\mathrm{orth}} \sum_e \|R_e^\top R_e - I\|_F^2.$$

    for each expert $e = 1, \ldots, E_N$ **do**
      Convert the updated latent scales into positive diagonal matrices.
      Perform scale alignment between S_e^{in} and S_e^{out}: transfer the magnitude of the input scale to the output scale and normalize the input scale to unit norm.
    end for
    Upload the aligned effective parameters $\{D_{e,k}^{in}, R_{e,k}, D_{e,k}^{out}\}_{e=1}^{E_N}$.
  end for
  Server-side aggregation alignment
  for each expert $e = 1, \ldots, E_N$ **do**
    Aggregate and normalize the input scale factor to obtain $\widehat{D}_e^{in}$.
    Aggregate matrices R_e and perform orthogonal projection to obtain $\widehat{R}_e$.
    Construct the aggregated expert target and solve the closed-form output-scale reconstruction problem to obtain $\widehat{D}_e^{out}$.
    Convert the aggregated positive scale factors back into latent parameters:

$$\theta_e^{in} = \mathrm{softplus}^{-1}(\widehat{S}_e^{in}), \quad \theta_e^{out} = \mathrm{softplus}^{-1}(\widehat{S}_e^{out}).$$

  end for
  Broadcast updated latent expert parameters to all clients.
end for
return Final global model parameters.

Therefore, all scale factors remain strictly positive during training and communication. After local optimization, clients first transform the latent parameters into their effective positive scales and perform scale alignment on the resulting diagonal matrices. The aligned positive factors are then uploaded to the server. At the server side, aggregation alignment is performed independently for each expert. After aggregation, the updated positive scale factors are converted back to the latent parameter space through the inverse Softplus function:

$$\theta = \mathrm{softplus}^{-1}(S). \tag{48}$$

The recovered latent parameters are broadcast to clients and used for the next communication round. The complete training procedure is summarized in Algorithm 1.

F Extended experiment

F.1 Visual and Multimodal Experiments

F.1.1 Visual Experiments

To evaluate the performance of our method on classic computer vision tasks, we conduct image classification experiments using the CIFAR-100 dataset (Krizhevsky et al., 2009). We fine-tune the ViT-Base model (Dosovitskiy et al., 2020) with the LoRA rank set to 8. The federated training spans 20 communication rounds, with 2 local epochs per round and a client participation rate of 0.5. To

comprehensively assess robustness, we simulate both IID and Non-IID (Dirichlet distribution with $\alpha = 0.1$) data partitioning scenarios.

The experimental results are visualized in Table 6. Overall, A-SMoE achieves the best comprehensive performance in both IID and Non-IID environments.

In the highly heterogeneous Non-IID setting, we observe a severe local polarization phenomenon in the FedSA method: it reaches an exceptionally high Local ACC of 93.23%, but its Global ACC collapses to a mere 43.81%. This indicates catastrophic overfitting to local data distributions at the expense of global generalization. Excluding this anomalous local metric from FedSA, our proposed method consistently achieves the highest performance in both local and global evaluations.

Furthermore, in the homogeneous IID setting, a notable phenomenon occurs: almost all other advanced baselines suffer from negative optimization and fail to surpass the most foundational baseline, FedIT (which simply combines FedAvg with standard LoRA). A-SMoE is the only method that successfully outperforms FedIT in this scenario (achieving 90.44% L-ACC and 90.12% G-ACC). These observations firmly demonstrate the remarkable robustness and wide applicability of our method across vastly different data distributions and modalities.

Table 6: Performance comparison on the CIFAR-100 image classification task using the ViT model under IID and Non-IID ($\alpha = 0.1$) settings.

Method	**IID**		**Non-IID** ($\alpha = 0.1$)	
	L-ACC	**G-ACC**	**L-ACC**	**G-ACC**
FedIT	90.13	90.08	82.03	82.06
FedSA	85.21	85.51	93.23	43.81
FFA-LoRa	71.29	71.37	82.15	82.01
FedRot	89.61	89.66	80.65	80.72
FedALT	89.66	89.52	81.32	80.98
FlyLoRA	87.23	88.05	76.89	77.24
FlexLoRa	88.85	89.41	81.59	81.64
LoRa-XS	90.09	90.03	84.02	84.05
PiSSA	89.40	89.79	81.62	81.57
LoRa-MoE	89.55	89.73	80.86	80.98
LoRa-MoE+CKA	89.46	89.71	81.04	81.09
A-SMoE (Ours)	**90.44**	**90.12**	**84.83**	**84.81**

F.1.2 Multimodal Experiments

To further evaluate the generalizability of our proposed method across broader domains, we extend our experiments to multimodal tasks. Specifically, we focus on the classic Visual Question Answering (VQA) task by fine-tuning the ViLT model (Kim et al., 2021). The evaluation is conducted on four standard VQA datasets: VQA-v2 (Goyal et al., 2017), ScienceQA (Lu et al., 2022), OK-VQA (Marino et al., 2019), and COCO-QA (Ren et al., 2015).

Consistent with the federated configuration, we deploy 12 clients with a client participation rate of 0.5. The global training spans 20 communication rounds, with local training set to 1 epoch per round. To construct a heterogeneous federated scenario, we employ the TF-IDF algorithm (Sparck Jones, 1972) to extract text features from the local text data. These features are then utilized to create pseudo-labels for heterogeneous data partitioning, using a Dirichlet distribution with concentration parameter $\alpha = 0.1$.

The evaluation results on the VQA tasks are summarized in Table 7. Despite the challenging cross-modal nature and severe data heterogeneity, A-SMoE successfully bridges the representation gaps and consistently achieves the highest overall accuracy. Specifically, our method reaches a Local Average of 59.91% and a Global Average of 38.66%, comprehensively outperforming other baselines and demonstrating strong robustness on multimodal task.

Table 7: Performance comparison on Visual Question Answering (VQA) tasks using the ViLT model. L-ACC and G-ACC denote Local and Global Evaluation Accuracy (%), respectively.

Method	VQA-v2		ScienceQA		OK-VQA		COCO-QA		Average	
	L-ACC	G-ACC	L-ACC	G-ACC	L-ACC	G-ACC	L-ACC	G-ACC	L-Avg	G-Avg
FedIT	84.55	45.50	50.00	45.72	30.83	36.70	59.31	19.48	56.17	36.85
FedSA	80.82	45.16	**59.73**	48.71	30.42	**38.77**	**67.61**	17.36	59.64	37.50
FFA-LoRa	81.73	43.97	47.27	49.47	28.86	34.69	65.72	**21.09**	55.90	37.30
FedRot	85.57	45.73	47.48	48.01	31.90	37.34	60.22	20.47	56.29	37.89
FedALT	85.68	**46.14**	54.73	48.01	31.30	37.16	66.17	20.10	59.47	37.85
FlyLoRA	**85.75**	45.89	45.11	46.52	30.82	36.71	59.51	19.67	55.30	37.20
FlexLoRa	84.64	45.63	52.05	47.49	31.36	36.41	62.07	20.43	57.53	37.49
LoRa-XS	82.80	45.44	47.15	48.22	30.62	36.05	62.51	19.30	55.77	37.25
PiSSA	85.44	45.73	49.49	49.16	31.84	37.16	63.74	20.35	57.63	38.10
LoRa-MoE	85.32	45.65	49.62	49.05	32.19	36.93	62.26	20.24	57.35	37.97
LoRa-MoE+CKA	84.75	45.53	50.83	46.33	31.52	36.72	59.48	19.48	56.65	37.02
A-SMoE	85.51	45.97	57.98	**52.06**	**32.34**	38.21	64.77	18.42	**59.91**	**38.66**

Table 8: Robustness comparison on GLUE under varying client numbers (N) and LoRA ranks (r). Best results are in **bold**; second-best results are underlined.

Method	Varying Client Number				Varying LoRA Rank			
	$N = 4$		$N = 32$		$r = 4$		$r = 16$	
	L-Avg	G-Avg	L-Avg	G-Avg	L-Avg	G-Avg	L-Avg	G-Avg
FedIT	75.37	72.06	76.38	73.16	81.55	79.82	75.41	72.28
FedSA	79.67	80.19	79.22	78.65	81.52	83.71	87.06	85.92
FFA-LoRA	69.05	68.51	75.69	72.25	72.33	69.15	74.13	70.35
FedRot	81.14	78.51	83.18	81.82	85.17	83.42	83.86	83.64
FedALT	74.65	72.45	81.98	82.46	80.89	79.88	84.05	81.88
FlyLoRA	79.86	79.91	80.62	82.24	82.92	82.37	85.35	84.72
FlexLoRA	80.18	81.64	80.18	81.64	81.84	83.12	83.72	84.08
LoRA-XS	82.28	82.63	81.15	82.12	85.06	83.95	86.16	84.25
PiSSA	81.71	81.49	79.36	80.84	78.37	78.04	82.55	82.21
LoRA-MoE	78.21	80.81	81.29	80.24	83.79	80.03	83.82	80.35
LoRA-MoE+CKA	78.89	81.48	80.27	82.36	83.24	81.83	85.09	83.13
A-SMoE	**85.52**	**85.58**	**84.22**	**83.49**	**85.61**	**84.61**	**87.46**	**86.89**

F.2 Robustness Analysis

To thoroughly evaluate the robustness of our proposed method, we conduct additional experiments on the GLUE benchmark by varying two key configurations: the total number of clients ($N \in \{4, 32\}$, default $N = 12$) and the LoRA rank ($r \in \{4, 16\}$, default $r = 8$). The comprehensive evaluation results are summarized in Table 8. As demonstrated, across all combinations of varying client scales and LoRA rank settings, our method A-SMoE consistently achieves the highest Local Average and Global Average accuracy. This confirms the strong stability and robustness of our approach.

To further evaluate the robustness of our method, we investigate its performance under two additional participation configurations based on the GLUE benchmark.

First, we simulate an ideal scenario where all clients participate in every communication round (i.e., 100% participation rate). As shown in Table 9, our method A-SMoE continues to achieve the highest accuracy across all baselines.

Second, we simulate an extreme but common scenario in real-world federated learning where a client permanently drops out. Specifically, we set Client 1 to permanently disconnect during training, while maintaining a 0.5 participation rate for the remaining clients. The results are presented in Table 10. Although all methods experience severe performance degradation due to the permanent loss of local data, A-SMoE consistently maintains the best performance.

Table 9: Performance under the ideal scenario with a 100% client participation rate ($p = 1.0$).

Method	L-Avg	G-Avg	Method	L-Avg	G-Avg
FedIT	82.17	80.24	FlexLoRa	84.82	82.91
FedSA	84.68	86.54	LoRa-XS	85.29	86.73
FFA-LoRa	68.25	64.72	PiSSA	85.73	86.15
FedRot	83.93	84.45	LoRa-MoE	83.75	83.16
FedALT	85.25	82.49	LoRa-MoE+CKA	84.25	83.06
FlyLoRA	84.43	82.17	**A-SMoE**	**88.52**	**87.01**

Table 10: Performance under the extreme scenario where Client 1 permanently drops out.

Method	L-Avg	G-Avg	Method	L-Avg	G-Avg
FedIT	70.24	70.58	FlexLoRa	72.55	71.09
FedSA	72.99	71.72	LoRa-XS	75.33	73.68
FFA-LoRa	61.23	57.78	PiSSA	75.68	73.42
FedRot	72.94	73.84	LoRa-MoE	73.18	71.27
FedALT	71.48	72.42	LoRa-MoE+CKA	74.56	72.95
FlyLoRA	75.05	74.18	**A-SMoE**	**79.21**	**75.78**

F.3 Sensitivity Analysis of Orthogonal Regularization

The orthogonality constraint of the internal processor R_e is an important component in A-SMoE. During local optimization, we introduce an orthogonal regularization term to preserve the structural property of R_e:

$$\mathcal{L}_{orth} = \lambda_{orth} \sum_{e=1}^{E_N} \left\| R_e^\top R_e - I \right\|_F^2, \tag{49}$$

where λ_{orth} controls the strength of the orthogonality constraint. To investigate the influence of this hyperparameter, we conduct a sensitivity analysis on the GLUE benchmark under the same experimental setting as the main experiments. We vary λ_{orth} from 10^{-1} to 0 and report both local accuracy (L-Avg) and global accuracy (G-Avg).

As shown in Table 11, A-SMoE maintains stable performance over a relatively wide range of λ_{orth} values. When λ_{orth} is excessively large (e.g., 10^{-1}), the over-emphasized orthogonal regularization may introduce optimization interference with the task objective, preventing the expert processors from fully adapting to downstream tasks and resulting in performance degradation. When λ_{orth} is removed, the performance decreases from 87.11% to 85.67% on L-Avg and from 85.97% to 84.79% on G-Avg, demonstrating that maintaining the orthogonal structure of R_e is beneficial for federated expert alignment.

The default setting of $\lambda_{orth} = 10^{-4}$ provides a favorable trade-off between enforcing the orthogonal structure of R_e and optimizing the downstream task objective, leading to stable performance across both local and global evaluations.

F.4 Server-side Aggregation Efficiency Analysis

To further evaluate the computational efficiency of the proposed alignment-based aggregation strategy, we measure the server-side aggregation latency of different federated fine-tuning methods.

All measurements only consider the additional computation performed on the server during aggregation and exclude communication overhead. The experiments are conducted on an NVIDIA RTX PRO 6000 GPU (96GB memory) with an Intel Xeon Platinum 8470 CPU (208 threads) using CUDA 13.2. The reported latency is measured in seconds.

Table 11: Sensitivity analysis of the orthogonal regularization coefficient λ_{orth} on GLUE. The default setting used in the main experiments is highlighted in bold.

λ_{orth}	**L-Avg (%)**	**G-Avg (%)**
10^{-1}	83.45	82.97
10^{-2}	85.18	84.46
10^{-3}	87.03	**86.05**
10^{-4}**(Default)**	**87.11**	85.97
10^{-5}	86.86	85.63
0	85.67	84.79

Table 12: Server-side aggregation latency comparison on FLAN with LLaMA2-7B. Latency is measured in seconds. The relative overhead is calculated with respect to standard LoRA FedAvg.

Rank	LoRA	FlexLoRA	A-SMoE	Flex/LoRA	A-SMoE/LoRA
4	0.0221	81.4505	0.2127	3692.8×	9.64×
8	0.0353	81.6362	0.3404	2314.0×	9.65×
16	0.0485	82.3092	0.5043	1696.8×	10.39×

We compare standard LoRA FedAvg, FlexLoRA, and A-SMoE. Following the aggregation procedure of FlexLoRA, which first reconstructs the aggregated dense update matrix and then applies truncated SVD to recover the low-rank representation, its computational cost is dominated by dense matrix reconstruction and SVD decomposition. In contrast, A-SMoE performs aggregation directly on the structured low-rank expert representation.

For a participating client number K, adapter number L, hidden dimension d, rank r, and expert number E_N, the theoretical server complexity can be summarized as:

$$T_{\text{LoRA}} = O(KLr(d_{in} + d_{out})), \tag{50}$$

$$T_{\text{FlexLoRA}} = O(KLd_{in}rd_{out}) + O(Ld^3), \tag{51}$$

where the second term corresponds to SVD decomposition on the recovered dense update matrix.

For A-SMoE, the alignment is performed on the expert processor composed of diagonal scale matrices and a rank-dimensional orthogonal matrix. Therefore, its complexity becomes:

$$T_{\text{A-SMoE}} = O(KE_Nr^2) + O(E_Nr^3). \tag{52}$$

This avoids applying SVD operations on the original hidden-dimensional parameter space.

The detailed aggregation latency results on FLAN with LLaMA2-7B are reported in Table 12. As shown, FlexLoRA introduces several orders of magnitude higher aggregation cost than standard LoRA due to dense-space reconstruction. Although A-SMoE introduces additional alignment computation, it remains substantially more efficient than FlexLoRA.

The corresponding latency evaluation on GLUE with RoBERTa-large is shown in Table 13. Compared with FlexLoRA, A-SMoE consistently achieves lower aggregation latency across different rank settings.

To further analyze the computational composition of A-SMoE aggregation, we separately record the latency of scale alignment and the complete alignment procedure including geometry-aware aggregation alignment. The results are summarized in Table 14.

Overall, the results verify that A-SMoE achieves efficient aggregation alignment without relying on computationally expensive dense-space SVD operations. Compared with FlexLoRA, A-SMoE

Table 13: Server-side aggregation latency comparison on GLUE with RoBERTa-large. Latency is measured in seconds.

Rank	LoRA	FlexLoRA	A-SMoE	Flex/LoRA	A-SMoE/LoRA
4	0.0089	3.1035	0.0697	348.4×	7.83×
8	0.0125	3.1579	0.1004	253.3×	8.05×
16	0.0176	3.2373	0.1432	183.4×	8.12×

Table 14: Runtime decomposition of A-SMoE alignment operations. The first column represents scale alignment, while the second column includes both scale alignment and aggregation alignment.

Dataset	Rank	Scale Alignment(s)	Full Alignment(s)	Aggregation Alignment Ratio
LLaMA2-7B	4	0.0323	0.2127	84.8%
LLaMA2-7B	8	0.0281	0.3404	91.7%
LLaMA2-7B	16	0.0273	0.5043	94.6%
RoBERTa-large-355M	4	0.0253	0.0697	63.8%
RoBERTa-large-355M	8	0.0254	0.1004	74.7%
RoBERTa-large-355M	16	0.0262	0.1432	81.7%

reduces aggregation latency by approximately 160–$380\times$ on FLAN and 22–$45\times$ on GLUE, while introducing only moderate additional overhead compared with standard LoRA FedAvg.