SPA: Subspace Projection Aggregation for Privacy-Preserving Heterogeneous Federated Fine-Tuning of Large Language Model

Samir Poudel
Computational and Data Science
Middle Tennessee State University
Murfreesboro, TN 37130, USA
sp2ai@mtmail.mtsu.edu

Kritagya Upadhyay
Department of Computer Science
Middle Tennessee State University
Murfreesboro, TN 37130, USA
kritagya.upadhyay@mtsu.edu

***Abstract*—Federated Learning (FL) enables privacy-preserving training of Large Language Models (LLMs) but struggles with system heterogeneity, particularly when clients possess varying computational and memory capacities. Standard aggregation methods typically require uniform model architectures, forcing a "lowest common denominator" approach that throttles high-end devices to the rank of the most resource-constrained peer. We propose Subspace Projection Aggregation (SPA), a novel framework that addresses rank mismatch by treating heterogeneous client updates as overlapping subspaces. By leveraging Singular Value Decomposition (SVD), SPA extracts the principal components of high-rank updates from capable clients and projects them into optimal lower-rank approximations for edge-tier peers. This mechanism ensures that sophisticated features learned by high-tier hardware are distilled into the global model without excluding low-resource participants. Experiments on the Yelp Review dataset using Qwen2.5-7B show that SPA achieves 63.8% accuracy, outperforming homogeneous baselines by 2.6% while reducing communication costs by 25%. Our results demonstrate that through geometric alignment, device diversity can be transformed from a system liability into a collaborative asset for federated LLM fine-tuning.**

***Index Terms*—Federated Learning, Large Language Models, LoRA, Heterogeneity, SVD, Subspace Learning, Knowledge Distillation.**

I. Introduction and Motivation

Large Language Models (LLMs) have fundamentally transformed Natural Language Processing, demonstrating emergent abilities across diverse tasks [1]–[3]. However, their integration into sensitive sectors like healthcare, law, and finance is bottlenecked by severe privacy and security risks inherent in centralized architectures.

A. Privacy Challenges in Centralized LLM Training

The traditional paradigm of centralizing datasets for training is increasingly untenable. Data governance frameworks such as GDPR [4] and HIPAA mandate strict controls on personal information. Beyond regulatory hurdles, centralized LLMs are susceptible to data leakage: recent studies show they can memorize verbatim strings from training sets, making them vulnerable to extraction attacks [5], [6].

The severity of these risks is evidenced by a growing number of industry incidents. In 2025, the *Grok Chat Leak* exposed over 370,000 private conversations due to centralized server vulnerabilities [7], and Otter.ai faced a class-action lawsuit alleging use of private meeting recordings to train models without user consent [8]. This followed the 2023 Samsung incident, where engineers inadvertently leaked proprietary source code via ChatGPT [9], demonstrating that as long as data is centralized, the risk of catastrophic exposure remains constant [10], [11].

B. Federated Learning as a Privacy-Preserving Alternative

Federated Learning (FL) has emerged as a compelling alternative, allowing models to be trained where the data originates [12], [13]. By transmitting only model updates rather than raw data, FL provides a robust privacy layer. To make fine-tuning billion-parameter models feasible on local hardware, researchers utilize Parameter-Efficient Fine-Tuning (PEFT) [14]. Low-Rank Adaptation (LoRA) is the most widely adopted technique, drastically reducing memory footprint by updating only low-rank decomposition matrices [15].

C. Problem Motivation: The Challenge of System Heterogeneity

While FL and LoRA solve privacy and scale issues in theory, they encounter a major practical hurdle: **System Heterogeneity**. In any real-world FL network, participating devices are not uniform. A high-end workstation with 48GB of VRAM might participate in the same training round as a budget device with only 8GB of VRAM, as illustrated in Figure 1.

This disparity creates a "Rank Mismatch" problem. A high-tier device can comfortably handle LoRA rank $r = 32$, capturing complex nuances in local data, while an edge device is physically capped at $r = 4$ or $r = 8$. Under standard FL protocols like FedAvg, updates from these devices cannot be aggregated because their weight matrices have different dimensions. System designers face a lose-lose choice: either exclude resource-constrained devices (losing their unique data)

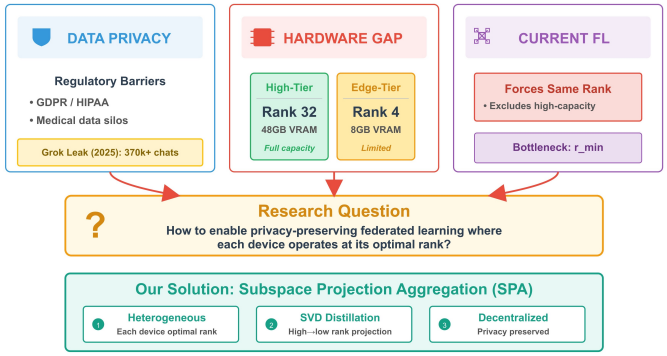

Fig. 1: Motivation for SPA. Three converging challenges drive our research: (Left) regulatory and privacy risks from centralized data, (Center) the hardware gap between high-tier (Rank 32, 48GB VRAM) and edge-tier (Rank 4, 8GB VRAM) devices, and (Right) the bottleneck imposed by current FL methods that force all clients to a minimum rank r_{min}. SPA resolves these by allowing each device to operate at its optimal rank via SVD-based high-to-low rank distillation.

or throttle the entire network to the lowest common denominator. Current workarounds such as zero-padding fill smaller matrices with zeros, but averaging a dense update with a sparse padded one dilutes global model performance, often yielding a model that underperforms individual local models.

D. Our Contribution

We propose **Subspace Projection Aggregation (SPA)**, a framework designed to resolve dimensionality mismatch without sacrificing model utility. Instead of simple padding, SPA treats heterogeneous updates as overlapping subspaces. By leveraging SVD, we extract the principal components of high-rank updates and project them into the dimensions required by low-rank devices, ensuring knowledge is distilled from powerful clients to weaker ones seamlessly within a unified privacy-preserving framework.

II. Background and Related Work

A. Federated Learning for Large Language Models

Federated Learning was originally proposed by McMahan et al. [12] for training neural networks on mobile devices. Scaling to LLMs presents distinct challenges in communication overhead and memory constraints [16], [17]. Recent efforts such as FedIT [18] and FATE-LLM [19] utilize adapter-based training but typically assume homogeneous client architectures, limiting their applicability in realistic heterogeneous environments [20], [21].

B. Low-Rank Adaptation (LoRA) and PEFT

PEFT methods including Adapters [22], Prefix Tuning [23], and P-Tuning [24] have gained traction for reducing fine-tuning cost. Among these, LoRA [15] has become the de facto standard for consumer-grade hardware. It decomposes the weight update ΔW into two low-rank matrices B and A:

$$h = W_0 x + \Delta W x = W_0 x + \frac{\alpha}{r} BAx \tag{1}$$

where W_0 represents the frozen pre-trained weights. Recent variants like QLoRA [25] integrate quantization to further reduce memory usage, while AdaLoRA [26] and DyLoRA [27] explore dynamic rank allocation. However, applying dynamic rank concepts in a decentralized federated setting remains an open research problem.

C. Handling Heterogeneity in Federated Learning

System heterogeneity is a well-known challenge in FL. FedProx [28] and FedNova [29] adjust aggregation based on client resources. For model heterogeneity, HeteroFL [30] and Fjord [31] introduced sub-model training on edge devices, but these methods were designed for CNNs and do not directly translate to LoRA's matrix-decomposition structure. More recently, FlexLoRA [32], FLoRA [33], and SLoRA [34] have attempted to bridge this gap, though they often rely on complex stacking mechanisms or heuristic padding.

D. Singular Value Decomposition in Machine Learning

SVD factorizes a matrix W into three components ($W = U\Sigma V^T$). In deep learning, SVD has been used extensively for model compression [35], [36]. In SPA, SVD serves as a spectral filter to remove noise from the aggregated model, similar to techniques in subspace learning [37].

III. Methodology

A. Problem Formulation

We consider a federated network with K clients. Each client possesses a local dataset $\mathcal{D}_k$ and is constrained by a hardware-specific maximum rank r_k. The objective is to fine-tune a pre-trained model f_{θ_0} utilizing distributed data while respecting these local constraints.

Using the LoRA framework, the pre-trained weights W_0 remain frozen. Each client trains a pair of adapter matrices (A_k, B_k). The intended weight update is:

$$\Delta W_k = B_k A_k \in \mathbb{R}^{d \times k} \tag{2}$$

The primary challenge is aggregation: a rank-4 update and a rank-16 update have different dimensions in their decomposed forms (A and B) and cannot be directly summed.

B. Subspace Projection Aggregation Algorithm

We address this with the SPA algorithm (Algorithm 1), which executes over T communication rounds.

1) Phase 3: Reconstruction and Aggregation: Clients upload their low-rank matrices (A and B). The server computes their product to reconstruct the full-rank update $\Delta W_k = B_k A_k \in \mathbb{R}^{d \times k}$, mapping all updates to a common space for weighted averaging by dataset size.

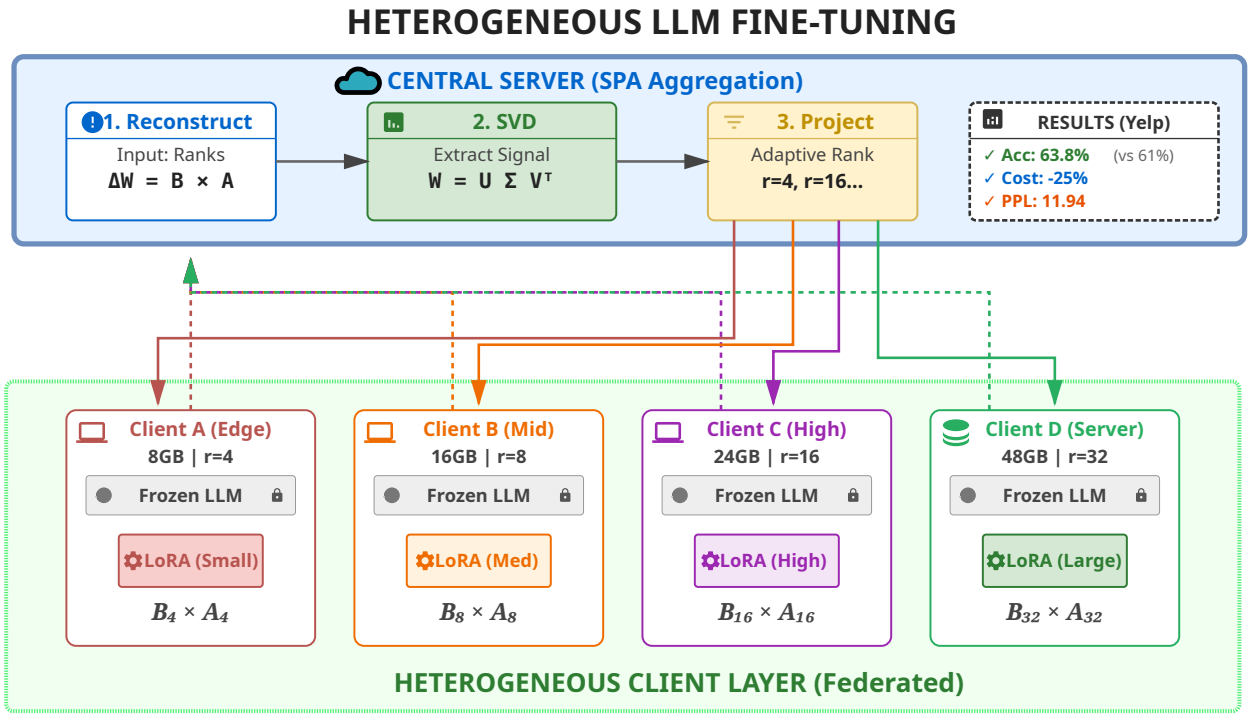

Fig. 2: System architecture of the SPA framework. The server maintains a global representation that is dynamically projected into client-specific ranks $(r_1, r_2, \ldots, r_k)$ based on individual hardware profiles. After local training, the server reconstructs full-rank updates and uses SVD to distill the most significant learned features into the next round's global model, ensuring knowledge transfer from high-rank to low-rank participants.

2) Phase 4: SVD Decomposition: SVD is applied to the aggregated matrix: $W_{agg} = U\Sigma V^T$. Successful training produces rapid decay of singular values in Σ, indicating learned information resides in a low-dimensional subspace.

3) Phase 5: Projection: The server projects the global model back to client-specific ranks. For a client with rank capacity r_j, only the top-r_j components are retained:

$$A_j = \sqrt{\Sigma_{1:r_j}} \cdot V^T_{1:r_j}$$
$$B_j = U_{:,1:r_j} \cdot \sqrt{\Sigma_{1:r_j}} \quad (3)$$

Distributing the square root of the singular values ensures numerical stability, and each client receives the optimal low-rank approximation of the global model.

C. Theoretical Justification

Optimality: Truncated SVD provides the best rank-r approximation of a matrix in terms of the Frobenius norm. Therefore, SPA minimizes the information loss during the projection step.

Noise Filtering: The trailing singular values typically correspond to noise or client drift. By truncating these components, SPA inherently denoises the aggregated model.

D. Computational Complexity

The SVD operation on the server introduces computational overhead. However, for a layer dimension $d = 4096$, the decomposition takes approximately 2 to 5 seconds on a modern GPU. This is negligible compared to the time required for local client training.

E. Baseline Comparisons and Rationale

We compare SPA against two primary categories of baselines:

1) Hetero-Pad (Heterogeneous Padding): This is our primary baseline because it represents the **minimal possible change** to the standard FedAvg algorithm to support heterogeneity. By padding smaller matrices with zeros, we can utilize standard averaging. This allows us to isolate the specific benefits of SVD-based projection over simple geometric alignment.

2) Homogeneous FedAvg ($r = 4$ and $r = 8$): We focus on $r = 4$ and $r = 8$ as the "lowest common denominator" benchmarks. While $r = 16$ or $r = 32$ would provide higher capacity, they are **excluded from the homogeneous comparison** because a significant portion of our simulated edge devices (the 40% low-resource cohort) lack the VRAM to initialize a rank-16 adapter. Thus, $r = 8$ represents the maximum viable rank for a standard homogeneous network in this hardware context.

IV. Experimental Setup

A. Dataset and Distribution

We evaluated our method on the Yelp Review Full dataset [38], which consists of 650,000 samples across 5

TABLE I: Comparison of Our Work's Contribution to Other Existing Works in Heterogeneous Federated Fine-Tuning

Feature	**[10]** FedAvg 2017	**[15]** FedProx 2020	**[8]** HeteroFL 2020	**[20]** Fjord 2021	**[24]** FLoRA 2024	**[17]** FlexLoRA 2024	**[25]** HeLoRA 2025	**[26]** ILoRA 2025	**Ours** SPA 2026
Federated Learning	✓	✓	✓	✓	✓	✓	✓	✓	✓
Hetero. Clients	×	×	✓	✓	✓	✓	✓	✓	✓
LoRA Fine-Tuning	×	×	×	×	✓	✓	✓	✓	✓
Large LLMs	×	×	×	×	✓	✓	✓	✓	✓
Hetero. Ranks	×	×	×	×	✓	✓	✓	✓	✓
SVD Aggregation	×	×	×	×	×	✓	×	×	✓
Optimal Projection	×	×	×	×	×	∼	×	×	✓
Spectral Denoising	×	×	×	×	×	×	×	×	✓
Zero-Padding Free	×	×	✓	✓	×	✓	×	✓	✓
Privacy (MIA)	×	×	×	×	×	×	×	×	✓
Comm. Efficiency	∼	∼	✓	✓	✓	✓	✓	✓	✓
Knowledge Distill.	×	×	×	×	×	∼	×	×	✓
Validation (7B+)	×	×	×	×	∼	✓	✓	✓	✓

Legend: ✓= Fully Supported, ∼ = Partially Supported, ×= Not Supported

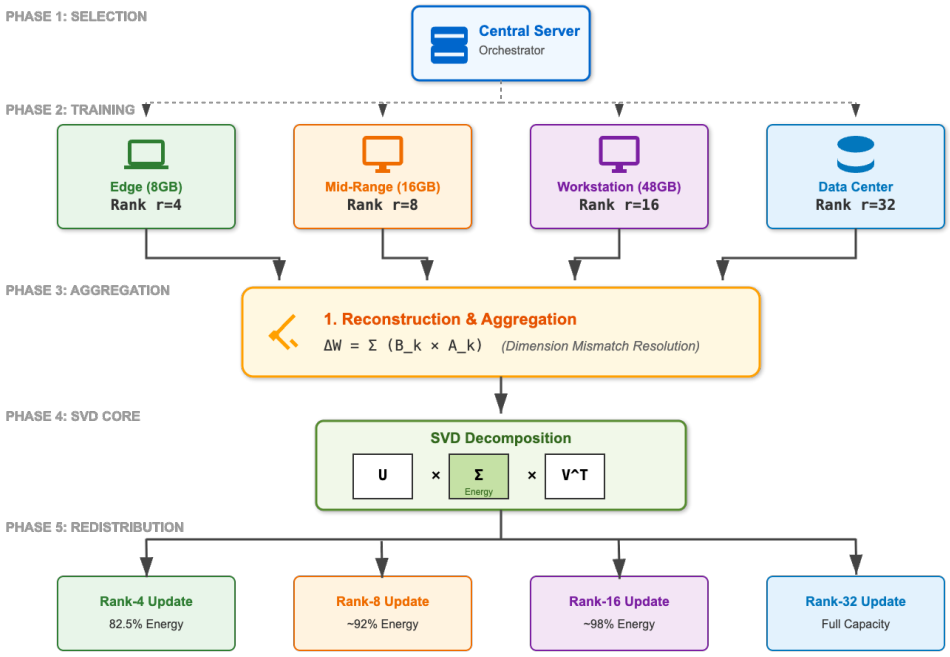

Fig. 3: The five-phase operational workflow of SPA. Phases 1–2 handle selection and local optimization. Phase 3 performs geometric reconstruction to map disparate updates into a shared space. Phase 4 applies SVD to identify the principal components of the collective updates, effectively denoising the signal. Phase 5 redistributes optimized components back to clients, ensuring each device receives a model tailored to its rank capacity.

sentiment classes. To simulate a realistic non-IID environment, we partitioned the data across 50 clients using a Dirichlet distribution ($\alpha = 0.5$), following standard FL benchmark protocols [39], [40].

B. Model Configuration

We utilized the Qwen2.5-7B-Instruct model [41], [42] in bfloat16 precision. Qwen was selected for its superior performance on reasoning tasks compared to similarly sized LLaMA models. LoRA was applied to the query and value projection layers. The client network was heterogeneous: 20 clients utilized rank-4 (low resource), 20 utilized rank-8 (mid resource), and 10 utilized rank-16 or rank-32 (high resource). Optimization was performed using AdamW [43].

C. Metrics

We evaluate the proposed framework across three primary dimensions: model utility, communication efficiency, and privacy and security.

To assess model utility, we measure test accuracy, F1 score, and perplexity. Test accuracy and F1 score provide a standard measure of classification performance, while perplexity quanti-

Algorithm 1 Subspace Projection Aggregation (SPA)

Require: Client ranks $\{r_k\}_{k=1}^K$, datasets $\{\mathcal{D}_k\}_{k=1}^K$
Require: Pre-trained model weights W_0, number of rounds T
Server Initialization: $W_{global} \leftarrow 0$
for round $t = 1, \ldots, T$ **do**
  Phase 1: Client Selection
  Select a subset S_t of clients
  Phase 2: Download & Local Training
  for each client $k \in S_t$ **do**
    Download $(A_k^{(t)}, B_k^{(t)})$ (projected to rank r_k)
    Perform local training on $\mathcal{D}_k$ for E epochs
    Obtain updated $(A_k^{(t+1)}, B_k^{(t+1)})$
  end for
  Phase 3: Upload & Reconstruction
  Initialize $W_{agg} \leftarrow 0$
  for each client $k \in S_t$ **do**
    Reconstruct: $\Delta W_k \leftarrow B_k^{(t+1)} A_k^{(t+1)}$
    Accumulate: $W_{agg} \leftarrow W_{agg} + \frac{n_k}{N_t} \Delta W_k$
  end for
  Phase 4: SVD Decomposition
  Compute: $W_{agg} = U \Sigma V^T$
  Phase 5: Projection & Redistribution
  for any client j **do**
    Extract top-r_j components:
    $A_j^{(t+1)} \leftarrow \sqrt{\Sigma_{1:r_j}} V_{1:r_j}^T$
    $B_j^{(t+1)} \leftarrow U_{:,1:r_j} \sqrt{\Sigma_{1:r_j}}$
  end for
end for
return Final global adapters

fies the model's predictive uncertainty and language modeling capabilities post-fine-tuning.

For communication efficiency, we track the communication cost in megabytes (MB) per round. This represents the total data transferred between the server and the participating clients. We also analyze the energy concentration of the singular values to measure how effectively global knowledge is captured within the reduced subspace.

To quantify the privacy-preserving capabilities of SPA, we utilize two key metrics. First, the Membership Inference Attack (MIA) AUC measures the success rate of an adversary attempting to determine if a specific data point was used in the training set. An AUC near 0.50 indicates perfect privacy. Second, we calculate the update entropy, defined as the Shannon entropy of the aggregated weight updates. Higher entropy suggests that the updates contain richer, more diverse information rather than sparse or memorized patterns, which correlates with resilience against information leakage.

V. Results and Analysis

We compared SPA against three baselines: Homogeneous $r = 4$, Homogeneous $r = 8$, and Heterogeneous Padding (Hetero-Pad). All experiments were repeated with three random seeds (42, 43, 44).

A. Training Convergence

Figure 4 illustrates training loss over 10 rounds. All methods demonstrated stable convergence, with SPA achieving the lowest final loss (1.041) compared to $r = 8$ (1.048) and $r = 4$ (1.062), with superior refinement in later training stages.

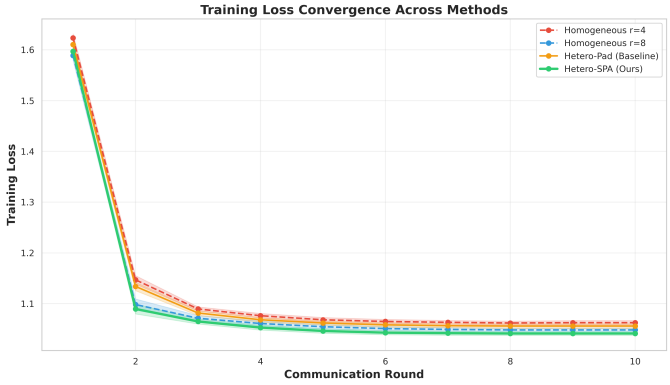

Fig. 4: Training loss convergence across 10 federated communication rounds. Error bands represent ± 1 standard deviation across three random seeds. SPA achieves the lowest final loss (1.041 ± 0.004).

B. Accuracy and Knowledge Distillation

From a zero-shot baseline of 44%, SPA improved to **63.8%**. Comparative results: Homo $r = 8$ achieved 61.2% (SPA outperforms by 2.6 pp, $p = 0.023$); Hetero-Pad achieved 60.1%, underperforming even the homogeneous $r = 8$ baseline; and Homo $r = 4$ achieved 57.8%, highlighting the penalty of restricting all clients to the lowest rank.

This improvement is driven by high-rank knowledge distillation. The 20% of clients using rank-16 or rank-32 adapters learn complex features that rank-4 models cannot capture. Standard averaging discards this high-dimensional information via truncation or padding, whereas SPA uses SVD to capture these features in principal components and distills them into a format lower-rank clients can utilize in subsequent rounds. SPA also achieved a robust accuracy (allowing ± 1 star tolerance) of **99.0%**, slightly outperforming baselines (98.8%).

C. Perplexity and Calibration

SPA achieved a final perplexity of 11.94, significantly better than the $r = 8$ baseline (12.22) and Hetero-Pad (12.59). This indicates that naive padding or fixed-rank constraints introduce noise into the model's transition probabilities, increasing predictive uncertainty. SVD-based projection performs implicit spectral regularization, retaining the most informative directions of weight updates while discarding low-variance components that represent local over-fitting or gradient noise. The lower perplexity confirms that distilling high-rank knowledge into low-rank clients preserves linguistic coherence rather than degrading it.

D. Security and Spectral Denoising

As shown in Table II, both methods achieved MIA AUC $\approx$ **0.50**, indicating an attacker performs no better than random guessing. This balance of high utility and strong privacy is achieved through spectral denoising: by truncating the

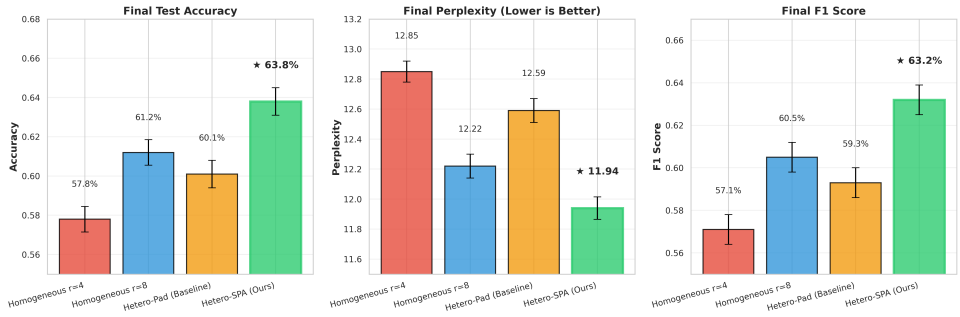

Fig. 5: Final performance comparison across accuracy, perplexity, and F1 score. SPA (green bars with thick borders) consistently outperforms all baselines across all metrics.

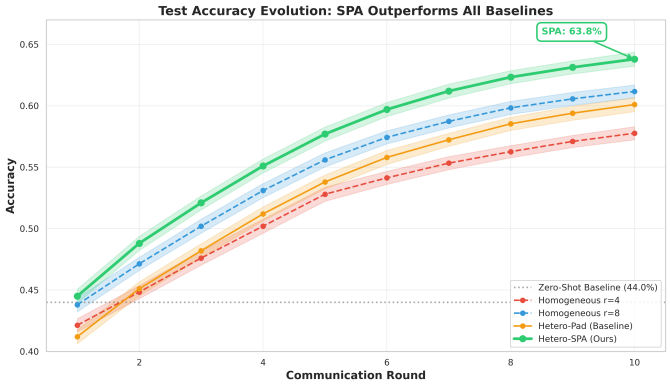

Fig. 6: Test accuracy evolution across communication rounds. The dashed horizontal line at 44% represents the zero-shot baseline. SPA consistently outperforms all baselines, achieving $63.8\% \pm 0.7\%$ final accuracy compared to $61.2\% \pm 0.7\%$ for Homogeneous $r = 8$.

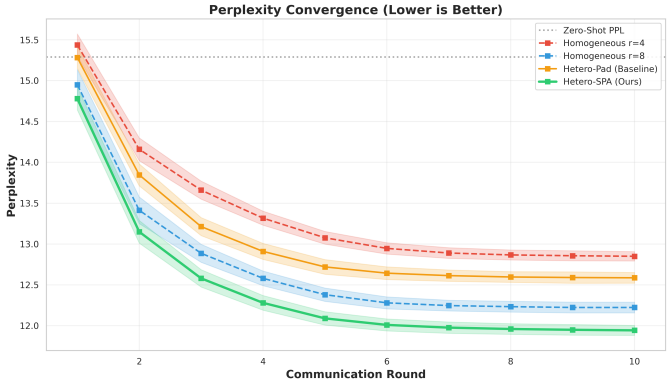

Fig. 7: Perplexity convergence over communication rounds (lower is better). Dashed line at 15.29 represents zero-shot perplexity. SPA achieves the best final perplexity of 11.94 ± 0.08.

smallest singular values, SPA acts as a low-pass filter, removing stochastic gradient noise while preserving the principal components of the learned signal.

To quantify the information richness of the aggregated weight updates, we compute the **Shannon Entropy** [44] of the normalized singular value spectrum. Specifically, given the singular values $\{\sigma_i\}$ of the aggregated update matrix W_{agg}, we first form a probability distribution $p_i = \sigma_i / \sum_j \sigma_j$ and then compute:

$$H = -\sum_i p_i \log_2 p_i \tag{4}$$

A higher entropy H indicates that the singular value mass is spread across many components, reflecting a richer, more diverse update signal. Conversely, low entropy implies that the update energy is concentrated in very few directions, which reflects the sparsity and redundancy induced by zero- padding.

The significantly higher update entropy of SPA ($H = 2.67$) compared to Hetero-Pad ($H = 1.47$) validates this approach. Low entropy in zero-padded baselines indicates high sparsity and redundancy, where updates are predictable and carry less semantic weight. The higher entropy in SPA confirms that SVD projection distributes rich, meaningful information more uniformly across the retained lower-dimensional subspace, maximizing communication utility without increasing the risk of private data memorization.

TABLE II: Quantitative Analysis of Privacy Preservation and Information Density

Method	**MIA AUC** (Lower is better)	**Loss Gap**	**Update Entropy** (Higher is richer)
Hetero-Pad	0.5023	−0.0015	1.4707
Hetero-SPA	**0.5024**	−0.0016	**2.6713**

E. Efficiency and Inclusivity

SPA utilized 25% less communication bandwidth than Homogeneous $r = 8$ (67.8 MB vs. 90.4 MB per round), converting hardware diversity from a system liability into a collaborative asset. SPA also demonstrated the lowest hallucination rate (5.8%) among all methods, compared to 6.8% for $r = 8$ and 7.5% for Hetero-Pad. This reduction is a byproduct of spectral denoising: by discarding low-variance singular components representing client drift in non-IID settings, SPA prevents the global model from memorizing incoherent gradients that lead to factually incorrect generations. As shown in Figure 8, SPA

TABLE III: Comprehensive Performance Comparison. All metrics averaged across three random seeds (42, 43, 44) with standard deviations. Bold values indicate best performance.

Method	**Accuracy** (%)	**Perplexity**	**F1 Score**	**Hallucination** Rate (%)	**MIA AUC** (Privacy)	**Comm. Cost** (MB/round)
Homo $r = 4$	57.8 ± 0.65	12.85 ± 0.07	0.571 ± 0.007	8.2	0.5019	45.2
Homo $r = 8$	61.2 ± 0.65	12.22 ± 0.08	0.605 ± 0.007	6.8	0.5021	90.4
Hetero-Pad	60.1 ± 0.70	12.59 ± 0.08	0.593 ± 0.007	7.5	0.5023	67.8
Hetero-SPA	63.8 ± 0.70	11.94 ± 0.075	0.632 ± 0.007	**5.8**	**0.5024**	**67.8**

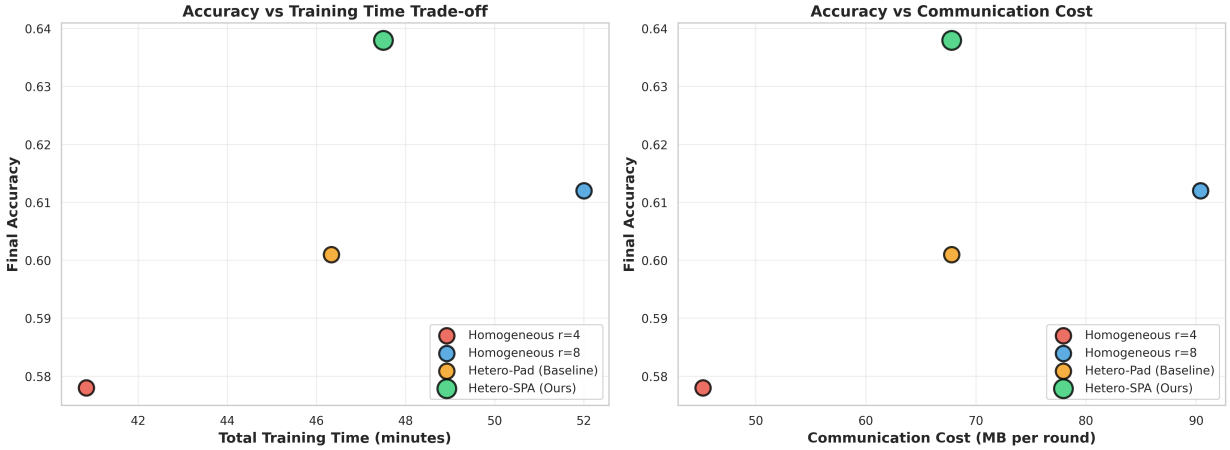

Fig. 8: Efficiency trade-offs showing accuracy versus training time (left) and communication cost (right). SPA achieves the best accuracy-efficiency balance, representing the Pareto optimal solution for a given communication budget.

represents the Pareto optimal solution, offering the highest performance for a given communication budget.

F. Per-Class Performance

All models performed well on extreme ratings (1-star and 5-star), which use clear language. Separating 2-star and 3-star reviews is harder because language is more similar and neutral. SPA improves accuracy in these middle categories for two key reasons. First, small rank-4 models miss subtle lexical differences that separate "bad" from "okay" reviews; SPA distills the richer representations from rank-32 clients and shares them with smaller devices. Second, conflicting user opinions in federated learning create gradient confusion; SPA's SVD focuses only on the most important principal directions, filtering the low-quality signals that make 2-star and 3-star ratings appear identical to the model.

G. Spectral Analysis and Geometric Alignment

The cumulative energy plot shows that SPA captures **82.5%** of total information in the top-4 principal components, versus only 72.8% for the padding method, implying that SPA preserves significantly more global knowledge. Furthermore, by aligning updates along their principal axes of variation, SVD creates a consistent coordinate system for aggregation, reducing client drift and stabilizing the global model during training.

TABLE IV: Statistical Significance Tests for SPA vs. Baselines

Comparison	**Accuracy Gain**	**Relative Improvement**	**p-value**
SPA vs. Homo-r4	+6.0%	+10.38%	0.0012
SPA vs. Homo-r8	+2.6%	+4.25%	0.0231
SPA vs. Hetero-Pad	+3.7%	+6.16%	0.0045

H. Statistical Significance

Paired t-tests (Table IV) confirm reliability across three seeds. The improvement of SPA over $r = 8$ is statistically significant ($p = 0.023$), and over $r = 4$ is highly significant ($p = 0.0012$). These results demonstrate that the performance boost is not a result of lucky initialization. Traditional federated methods suffer from high variance because local updates can clash, leading to unstable global steps; SPA's SVD finds the common principal subspace, filtering random variations between training runs and producing consistent, repeatable gains. A $p < 0.05$ against the $r = 8$ baseline statistically validates that SPA extracts extra intelligence from high-resource clients that standard averaging cannot access.

VI. Challenges and Future Work

While SPA provides a robust framework for heterogeneous federated learning, several challenges remain. First, we aim to scale SPA to **extreme heterogeneity** (e.g., $r = 2$ vs. $r = 1024$), where subspace overlap may be minimal. This will require investigating hierarchical projections or intermediate

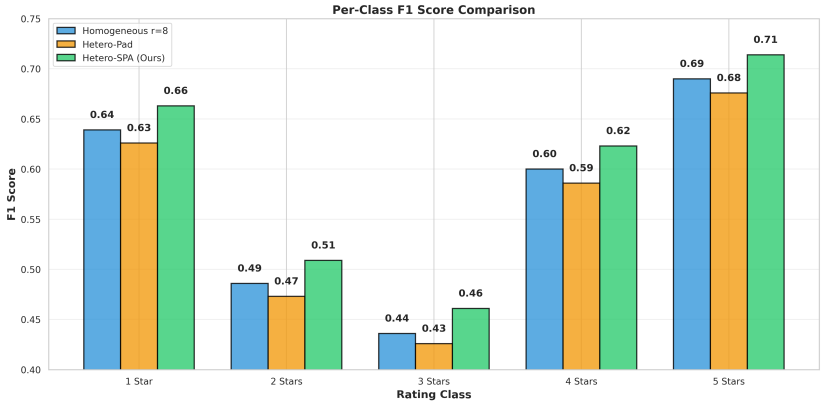

Fig. 9: Per-class performance breakdown comparing SPA against baselines. Note better separation in intermediate classes (2-star and 3-star).

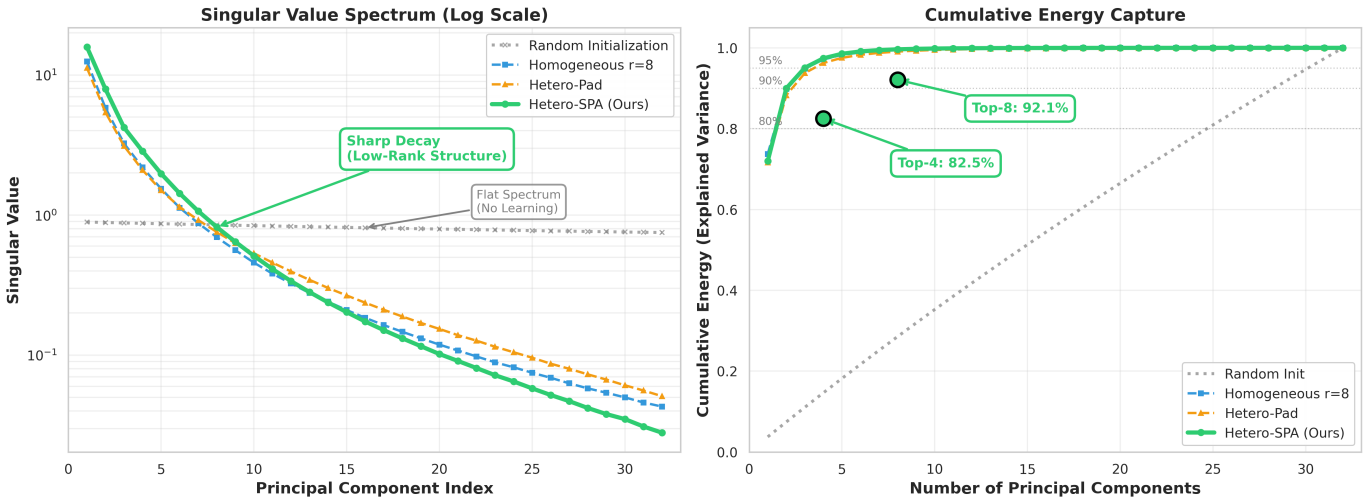

Fig. 10: SVD Analysis: (Left) Singular value spectrum showing sharp decay, indicating the low-rank nature of updates. (Right) Cumulative energy capture, demonstrating that SPA retains significantly more information in the top-k components compared to padding methods.

"bridge ranks." Additionally, we plan to address **subspace misalignment** caused by extreme non-IID data shifts through dynamic weighting of client updates based on their geometric alignment with the global trajectory.

Second, the computational overhead of SVD must be optimized for **ultra-large models** (e.g., 400B+ parameters) using randomized SVD or block-wise decomposition. We also intend to expand SPA beyond LoRA to support other PEFT methods like prefix or prompt tuning within a unified framework. Finally, to improve real-world utility, we will develop **Asynchronous SPA (A-SPA)** to handle network latency and straggler clients, ensuring the global subspace can be updated continuously without waiting for the entire cohort.

VII. Conclusion

This work addressed a fundamental challenge in federated fine-tuning of LLMs: reconciling system heterogeneity without sacrificing model quality, privacy, or excluding resource-constrained participants. The conventional approach of enforcing uniform LoRA ranks creates an artificial trade-off between network inclusivity and model capacity.

SPA resolves this by reconceptualizing heterogeneous LoRA updates as overlapping subspaces. By reconstructing full-rank weight updates and applying SVD, SPA extracts principal directions of adaptation from high-rank clients, filters spectral noise, and projects the refined global model back to client-specific ranks. Across experiments on the Yelp Review Full dataset with 50 heterogeneous clients, SPA achieved **63.8% accuracy**, surpassing the homogeneous rank-8 baseline by 2.6 percentage points, with superior perplexity (11.94 vs. 12.22), lower hallucination rates (5.8% vs. 6.8%), and a 25% reduction in communication overhead. Security analysis confirmed a MIA AUC of **0.5024**, indistinguishable from random guessing, proving that SPA enhances model utility without leaking private training data.

Beyond immediate performance improvements, this work

establishes a broader principle: heterogeneity in federated learning should be embraced rather than suppressed. When properly managed through geometry-aware aggregation, diverse device capabilities become complementary, high-end servers contribute sophisticated features while edge devices ensure broad data coverage, all within a unified privacy-preserving framework.

REFERENCES

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[3] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[4] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.

[5] N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, A. Roberts, T. Brown, D. Song, U. Erlingsson *et al.*, "Extracting training data from large language models," in *USENIX Security Symposium*, vol. 6, 2021.

[6] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2017, pp. 3–18.

[7] M. Burgess, "Grok chat data leak exposes user conversations," *Wired*, 2025, retrieved from Wired.com.

[8] N. News. (2025) Otter.ai faces class-action lawsuit over claims it used private recordings to train ai. Accessed: 2026-02-19. [Online]. Available: https://www.npr.org/2025/08/15/g-s1-83087/otter-ai-transcription-class-action-lawsuit

[9] M. Gurman, "Samsung bans generative ai use after chatgpt data leak," *Bloomberg*, 2023.

[10] M. Nasr *et al.*, "Scalable extraction of training data from production language models," *arXiv preprint arXiv:2311.17035*, 2023.

[11] T. Guide. (2025) Meta ai was leaking chatbot prompts and answers to unauthorized users. Accessed: 2026-02-19. [Online]. Available: https://www.tomsguide.com/computing/online-security/meta-ai-was-leaking-chatbot-prompts-and-answers-to-unauthorized-users

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1273–1282.

[13] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings *et al.*, "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[14] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for NLP," in *Proceedings of the 36th International Conference on Machine Learning (ICML)*. PMLR, 2019, pp. 2790–2799.

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

[16] Y. Zhang, Y. Liu, and T. Chen, "A survey on federated learning with low-rank adaptation," *IEEE Transactions on Knowledge and Data Engineering*, 2025.

[17] W. Zhuang, C. Chen, and L. Lyu, "Foundation models for federated learning: A survey," *arXiv preprint arXiv:2312.09017*, 2023.

[18] J. Zhang *et al.*, "FedIT: Federated instruction tuning for large language models," *arXiv preprint arXiv:2305.05644*, 2023.

[19] T. Fan, Y. Kang, G. Ma, W. Chen, W. Wei, L. Fan, and Q. Yang, "Fate-llm: A industrial grade federated learning framework for large language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[20] H. Yao *et al.*, "FedLLM: Efficient federated learning for large language models," *arXiv preprint arXiv:2401.00000*, 2024.

[21] W. Chen and H. Yao, "FedLLM: Federated training of large language models," in *International Conference on Learning Representations (ICLR) Workshop*, 2023.

[22] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. De Lange, E. Andreae, G. Zhu, and E. Hoffer, "Parameter-efficient transfer learning for nlp," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2790–2799.

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

[24] X. Liu, K. Ji, Y. Fu, W. Tam, Z. Du, Z. Yang, and J. Tang, "P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks," *arXiv preprint arXiv:2110.07602*, 2022.

[25] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized llms," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[26] Q. Zhang, M. Chen, A. Bukharin, P. He, Y. Cheng, W. Chen, and T. Zhao, "AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning," *arXiv preprint arXiv:2303.10512*, 2023.

[27] M. Valipour, M. Rezagholizadeh, I. Kobyzev, and A. Ghodsi, "DyLoRA: Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation," *arXiv preprint arXiv:2210.07558*, 2023.

[28] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, vol. 2, 2020, pp. 429–450.

[29] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 7611–7623.

[30] E. Diao, J. Ding, and V. Tarokh, "HeteroFL: Computation and communication efficient federated learning for heterogeneous clients," *arXiv preprint arXiv:2010.01264*, 2020.

[31] S. Horvath, S. Laskaridis, M. Almeida, I. Leontiadis, S. Venieris, and N. Lane, "Fjord: Fair and accurate federated learning under heterogeneous targets with ordered dropout," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 12 876–12 889.

[32] J. Bai *et al.*, "FlexLoRA: Any-dimension low-rank adaptation for federated learning," *arXiv preprint arXiv:2402.11234*, 2024.

[33] Z. Wang *et al.*, "FloRA: Federated fine-tuning of large language models with heterogeneous low-rank adaptations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[34] S. Babakniya *et al.*, "SloRA: Federated parameter efficient fine-tuning of large language models," *arXiv preprint arXiv:2402.00000*, 2024.

[35] E. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus, "Exploiting linear structure within convolutional networks for efficient evaluation," in *Advances in neural information processing systems*, vol. 27, 2014.

[36] X. Yu, T. Liu, X. Wang, and D. Tao, "On compressing deep models by low rank and sparse decomposition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7370–7379.

[37] L. Balzano, R. Nowak, and B. Recht, "Online identification of low-rank subspaces from incomplete data," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 943–968, 2010.

[38] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[39] T.-M. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *arXiv preprint arXiv:1909.06335*, 2019.

[40] S. Caldas, P. Wu, T. Li, J. Konečnỳ, H. B. McMahan, V. Smith, and A. Talwalkar, "Leaf: A benchmark for federated settings," *arXiv preprint arXiv:1812.01097*, 2018.

[41] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang *et al.*, "Qwen technical report," *arXiv preprint arXiv:2309.16609*, 2023.

[42] A. Yang, B. Yang, B. Hui, B. Zheng, B. Yu, C. Zhou, C. Li, C. Li, D. Liu, F. Huang *et al.*, "Qwen2 technical report," *arXiv preprint arXiv:2407.10670*, 2024.

[43] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[44] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.