On the Convergence of Zeroth-Order Federated Tuning for Large Language Models

Zhenqing Ling
Sun Yat-sen University
Shenzhen, Guangdong, China
lingzhq@mail2.sysu.edu.cn

Daoyuan Chen
Alibaba Group
Hangzhou, Zhejiang, China
daoyuanchen.cdy@alibaba-inc.com

Liuyi Yao
Alibaba Group
Hangzhou, Zhejiang, China
yly287738@alibaba-inc.com

Yaliang Li
Alibaba Group
Bellevue, Washington, United States
yaliang.li@alibaba-inc.com

Ying Shen*
Sun Yat-sen University
Shenzhen, Guangdong, China
Pazhou Lab
Guangzhou, Guangdong, China
Guangdong Provincial Key
Laboratory of Fire Science and
Intelligent Emergency Technology
Guangzhou, Guangdong, China
sheny76@mail.sysu.edu.cn

ABSTRACT

The confluence of Federated Learning (FL) and Large Language Models (LLMs) is ushering in a new era in privacy-preserving natural language processing. However, the intensive memory requirements for fine-tuning LLMs pose significant challenges, especially when deploying on clients with limited computational resources. To circumvent this, we explore the novel integration of Memory-efficient Zeroth-Order Optimization within a federated setting, a synergy we term as FedMeZO. Our study is the first to examine the theoretical underpinnings of FedMeZO in the context of LLMs, tackling key questions regarding the influence of large parameter spaces on optimization behavior, the establishment of convergence properties, and the identification of critical parameters for convergence to inform personalized federated strategies. Our extensive empirical evidence supports the theory, showing that FedMeZO not only converges faster than traditional first-order methods such as FedAvg but also significantly reduces GPU memory usage during training to levels comparable to those during inference. Moreover, the proposed personalized FL strategy that is built upon the theoretical insights to customize the client-wise learning rate can effectively accelerate loss reduction. We hope our work can help to bridge theoretical and practical aspects of federated fine-tuning for LLMs, thereby stimulating further advancements and research in this area.

*Corresponding author.

KDD '24, August 25–29, 2024, Barcelona, Spain.

ACM ISBN 979-8-4007-0490-1/24/08
https://doi.org/10.1145/3637528.3671865

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; • **Mathematics of computing** → **Mathematical optimization**.

KEYWORDS

Federated Learning, Zeroth-order Optimization, Convergence Analysis, Large Language Models

ACM Reference Format:
Zhenqing Ling, Daoyuan Chen, Liuyi Yao, Yaliang Li, and Ying Shen. 2024. On the Convergence of Zeroth-Order Federated Tuning for Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3637528.3671865

1 INTRODUCTION

Federated Learning (FL) has become an important approach in modern machine learning, particularly in scenarios where data decentralization and privacy-preserving are crucial [28, 42, 43, 56, 58]. Central to this learning paradigm is the corroborative training of a global model through the aggregation of updates from multiple clients, without sharing their raw data [31, 41].

In parallel, Large Language Models (LLMs) have radically advanced the field of natural language processing [6, 17, 52]. The fine-tuning of LLMs that are already pre-trained on vast corpora, has proven to be a highly effective strategy for numerous tasks, yielding models that are both versatile and capable of adapting to specific domain narratives or aligning with human values [6, 39].

The tuning of LLMs requires suitable alignment data, which are often costly to acquire [11, 16]. Due to the abundance of private data that remains largely isolated and underutilized, the intersection of FL and LLMs has sparked increasing interest among researchers [20, 32, 46, 61]. Notably, this integration presents significant computational challenges, especially for clients with limited resources [10, 62]. The scaling up of LLMs further compounds this issue, as the computation of gradients for backpropagation incurs substantial memory costs, frequently surpassing the practical capabilities of these clients [39].

Addressing this challenge, we turn our attention to Zeroth-Order Optimization (ZOO), an algorithm that computes gradient approximations without explicit gradient information, thus significantly reducing memory consumption [25]. However, the combination of ZOO and FL —a research direction we refer to as ZOO-FL —remains unexplored in the literature in the context of LLMs [47]. Our work intends to bridge this gap by harnessing the memory efficiency of ZOO within the context of federated fine-tuning of LLMs, especially on the following theoretical foundations:

(**Q1**) How does the vast parameter space of LLMs influence the behavior of ZOO-FL? (**Q2**) Can we establish the convergence properties of ZOO-FL for LLMs? (**Q3**) Which model parameters are critical for convergence, and how can we leverage them to optimize FL performance, such as via personalization?

In this paper, we focus on incorporating a memory-efficient ZOO method, MeZO [39] into FL, a synergy we denote as FedMeZO, and establishing its convergence properties under the large-scale parameter space of LLMs. We analyze and present precise convergence rates characterized by the low effective rank r of the models' Hessian matrices [3, 33], and other typical FL parameters such as number of clients N, number of FL rounds T, iteration steps of local training H and heterogeneity constants c_h and σ_h. Moreover, we reveal the learning rate to be a crucial variable for convergence. Building on theoretical insights, we further propose a strategy that tailors the learning rate to each client's specific data characteristics.

To validate our theoretical results, we conduct extensive experiments on real-world FL datasets for LLM tuning, which cover diverse data distributions and application tasks. Our empirical findings corroborate the theoretical analysis, validating effective convergence even when scaling up to models with billions of parameters. Compared with first-order methods such as FedAvg, FedMeZO converges faster meanwhile remarkably reducing GPU memory requirements. The personalized strategies guided by our theoretical insights, empirically show a more rapid loss reduction, as opposed to non-personalized or random learning rate assignments.

In summary, our theoretical and empirical exploration validates FedMeZO in the fine-tuning process of LLMs, providing a rigorous framework and practical insights for future applications. Our key contributions are threefold:

- We advance the understanding of FedMeZO for LLMs, extending the two-point gradient estimation to federated tuning and establishing theoretical convergence rate $O\left(r^{3/2}(NHT)^{-1/2}\right)$ and $O\left(r^{3/2}(\widetilde{c}_h NHT)^{-1/2}\right)$ $-O\left(\sigma_h^2(c_h N)^{-1}\right)$ for the i.i.d. setting and non-i.i.d. setting, respectively.
- We analyze the impact of various hyperparameters of FedMeZO and explore a theory-informed strategy for personalized learning rate adjustment, providing practical guidance for ZOO-FL.
- Through extensive empirical evidence with LLMs, we verify the proposed theoretical results and show that FedMeZO yields effective convergence with substantially reduced memory overhead compared to FedAvg. Our codes are publicly available at https://github.com/alibaba/FederatedScope/tree/FedMeZO.

2 PRELIMINARIES

2.1 Background and Related Works

Federated Fine-Tuning of Large Language Models. Large Language Models (LLMs) have demonstrated remarkable capabilities that enable a variety of real-world applications [52, 64, 65]. The federated fine-tuning of LLMs has recently attracted attention, focused on adapting these models to domain-specific tasks while preserving the privacy of the training data. Chen et al. [8] investigated the integration of LLMs within federated settings, highlighting the inherent challenges and potential opportunities. Zhang et al. [62] furthered this research by examining instruction tuning of LLMs in a federated context, marking progress in applying FL to the specialized training of LLMs. Notable frameworks such as FATE-LLM by Fan et al. [21] and FederatedScope-LLM by Kuang et al. [32] offer industrial-grade and comprehensive solutions for federated fine-tuning. Our work, in contrast, investigates the fusion of Zeroth-Order Optimization (ZOO) with FL for the fine-tuning of LLMs, an area that has yet to be fully investigated, thereby addressing a gap in the literature and providing fundamental theoretical insights.

Zeroth-Order Optimization in Federated Learning. ZOO has emerged as a viable method to address the difficulties of computing gradients in FL, especially in settings limited by computational resources. Zhang et al. [63] proposed a ZOO algorithm tailored for vertical FL, focusing on privacy preservation. Yi et al. [60] and Li et al. [34] studied ZOO-FL algorithms, with discussions on convergence properties with single-point perturbation and local updates in decentralized FL, respectively. The convergence analysis is a critical aspect of FL, as illustrated by Li et al. [37] for the FedAvg algorithm and further developed by Fang et al. [22] for mini-batch stochastic ZOO-FL in wireless networks. Moreover, Shu et al. [50] proposed enhancements to query efficiency for ZOO within the FL framework. Our research sets itself apart by formulating theoretical convergence bounds for ZOO-FL, specifically tailored to the large-scale parameter space of LLMs. This builds on the preliminary work by Malladi et al. [39], which confirmed the feasibility of ZOO for LLMs in a centralized setting.

2.2 Problem Formulation

Federated Learning. We consider the general FL setting as of FedAvg [41], with a central server and a collection of N clients, indexed by 1, 2, ..., N. The central server coordinates the training of a global model through the collaborative efforts of these clients, each holding local data samples drawn from their respective distributions $\mathcal{D}i$. The optimization problem can be formulated as:

$$\min_{\theta \in \mathbb{R}^d} f(\theta) \triangleq \frac{1}{N}\sum_{i=1}^{N} f_i(\theta_i), \quad f_i(\theta) \triangleq \mathbb{E}_{\mathcal{B}_i \sim \mathcal{D}_i}\left[F_i(\theta, \mathcal{B}_i)\right], \quad (1)$$

where $\theta \in \mathbb{R}^d$ denotes the d-dimension parameter of the model, and $f(\theta)$ and $f_i(\theta)$ denote the global loss function on the central server and local loss function on i^{th} client, respectively. Typically, the clients are assumed with equal importance [54], and the data is randomly sampled for efficiency [35]. $F_i(\theta, \mathcal{B}_i)$ represents the local loss function w.r.t a specific mini-batch $\mathcal{B}_i$ drawn from $\mathcal{D}_i$.

Zeroth-Order Optimization. Zeroth-order optimization (ZOO) is a prominent technique in scenarios where gradients are difficult to obtain, which estimates gradients by forward propagations. Given a random vector z and a smoothing constant μ, a typical one-point gradient estimator [18] is defined as:

$$\widetilde{\nabla}F(\theta, z, \mathcal{B}, \mu) = \frac{z}{2\mu}\big(F(\theta + \mu z, \mathcal{B}) - F(\theta, \mathcal{B})\big), \tag{2}$$

However, Eq. (2) provides a biased gradient estimation, leading to a certain degree of information loss [38]. Hence our work employs the ZOO paradigm with a two-point gradient estimator proposed by [39] in a federated setting:

Definition 2.1. (Two-point gradient estimator) Given a set of parameters $\theta \in \mathbb{R}^d$ for an LLM and a mini-batch $\mathcal{B}_i$, the two-point zeroth-order gradient estimator is formulated as:

$$\widetilde{\nabla}F_i(\theta, z_i, \mathcal{B}_i, \mu) = \frac{z_i}{2\mu}\big(F_i(\theta + \mu z_i, \mathcal{B}_i) - F_i(\theta - \mu z_i, \mathcal{B}_i)\big), \tag{3}$$

where $z_i \sim \mathcal{N}(0, I_d)$ is a Gaussian random variable and μ is the perturbation scale. The two-point gradient estimator in Eq. (3) requires only two forward passes through the model to compute the estimation of gradient, which serves as a memory-efficient alternative to backpropagation (BP).

The FedMeZO Algorithm. In this paper, we study and analyze the proprieties of a practical synergy of MeZO [39] and FedAvg [41], which is designed to fine-tune LLMs in an efficient, privacy-preserving and personalized manner. We term this ZOO-FL approach as FedMeZO, depicted with the following processes:

In a single communication round, the central server first broadcasts the global model parameters to available clients. Once the clients have completed their local updates and uploaded their models, the server aggregates the updates according to Eq. (1), forming the basis for the subsequent round.

Upon receiving the global model parameters, clients perform the following steps, distinguishing FedMeZO from traditional BP-based FedAvg algorithms in two-fold:

(1) Training Memory Reduction: Clients update their models using the two-point ZOO gradient estimator defined in Eq. (3) as:

$$e_i^{(t,k)} = \widetilde{\nabla}F_i(\theta_i^{(t,k)}, z_i^{(t,k)}, \mathcal{B}_i^{(t,k)}, \mu), \tag{4}$$

where (t, k) denotes the k^{th} iteration within the t^{th} communication round. Unlike standard ZO-SGD algorithms that require storing the perturbation vector z at each iteration, FedMeZO resamples z using random seeds in in-place implementation, thus reducing memory usage to a level equivalent to inference [39].

(2) Communication Cost Reduction: To mitigate the high communication overhead associated with LLMs, FedMeZO leverages Low-Rank Adaptation (LoRA) [4, 29], which introduces reparametrization to tune two small delta matrix on the linear layers instead of the whole LLM weights, based on the assumption that well pre-trained LLM possess a low "intrinsic dimension" when adapted to new tasks. Introducing it can help us further reduce the number of parameters to be updated and uploaded, thereby aligning with the practical constraints of federated settings. Detailed analysis of communication cost is available in Appendix C.1.

2.3 Lemmas and Assumptions

Lemma 2.2. ***(Unbiased Gradient Estimator)*** *The two-point zeroth-order gradient estimator described in Eq.* (3) *is an unbiased estimator of the true gradient, that is,*

$$\mathbb{E}[\widetilde{\nabla}F_i(\theta, z_i, \mathcal{B}_i, \mu)] = \nabla f_i(\theta). \tag{5}$$

The Hessian matrix, which is the square matrix of second-order partial derivatives of the loss w.r.t the model parameters, characterizes the curvature of the loss surface [24]. Although the size of a model's loss Hessian is often associated with the rate of fine-tuning, studies suggest that the large-scale parameters of LLMs do not necessarily impede convergence [2, 30]. This paradox is addressed by recognizing that the loss Hessian often exhibits a small local effective rank [39], which we capture in the following assumption:

Assumption 1. *There exist a Hessian matrix $\mathcal{H}(\theta^t)$ satisfying:*
- *$\nabla^2 f(\theta) \preceq \mathcal{H}(\theta^t)$ for all θ such that $\|\theta - \theta^t\| \le \eta d G(\theta^t)$, where $G(\theta^t) = \max_{\mathcal{B}\sim\mathcal{D}} \|\nabla f(\theta^t, \mathcal{B})\|$.*
- *The effective rank of $\mathcal{H}(\theta^t)$, denoted as $tr(\mathcal{H}(\theta^t))/\|\mathcal{H}(\theta^t)\|_{op}$, is at most r. Here tr denotes the trace of the matrix, and $\|\cdot\|_{op}$ denotes the operator norm.*

Assumption 1 characterizes a low effective rank r in the Hessian matrix, which demonstrates that LLM fine-tuning can occur in a low dimensional subspace ($\le$ 200 parameters) [3, 33]. With this insight, [39] identified the bound of loss descent at each step of centralized ZOO, which is partially influenced by r:

Lemma 2.3. ***(Bounded Centralized Descent)*** *Assume $f(\theta)$ is L-smooth and let $\widetilde{\nabla}F(\theta, z, \mathcal{B}, \mu)$ be the unbiased zeroth-order gradient estimator from Eq.* (3). *If the Hessian matrix $\mathcal{H}(\theta)$ exhibits a local effective rank of r, and constants $\gamma = \Theta(r/n)$ and $\zeta = \Theta(1/rd)$ exist, then the expected decrease in loss can be bounded as follows:*

$$\mathbb{E}\big[f(\theta^{t+1})\big] \le f(\theta^t) - \frac{\eta}{\gamma}\|\nabla f(\theta^t)\|^2 + \frac{\eta^2 L\zeta}{2}\mathbb{E}\big[\|\widetilde{\nabla}F(\theta, z, \mathcal{B}, \mu)\|^2\big], \tag{6}$$

where $\gamma = \frac{dr+d-2}{n(d+2)}$, $\zeta = \frac{(d+2)n^2}{(dr+d-2)(d+n-1)}$, and n is the number of randomizations.

From Eq. (6), we observe that the rate of descent at a single step depends on the gradient related to γ and the gradient estimation related to ζ. Following [39], we set n to 1 in this paper. Detailed proofs of Lemmas are available in Appendix C.1 and C.2 of our extended version [1].

Besides, to facilitate the analysis in FL setting, we introduce four assumptions, including *Bounded Loss* (Assumption 2), *L-smoothness* (Assumption 3), *mini-batch gradient error bound* (Assumption 4), *global-local disparities in i.i.d. and non-i.i.d. settings* (Assumptions 5 and 6 respectively). These assumptions are standard in optimization and FL literature [5, 36, 37, 55], which we detail in Appendix A.

3 MAIN RESULTS

3.1 Convergence Analysis in i.i.d. Case

In this subsection, we examine the convergence properties of FedMeZO within the i.i.d. data distribution setting. We establish the conditions under which the algorithm guarantees loss reduction at each iteration and provide a global convergence rate.

Theorem 3.1. ***(Stepwise Loss Descent in i.i.d. Setting)*** *Under Assumptions 1-5 and with a learning rate η satisfying*

$$\eta \leq \min\Big\{\frac{1}{3HL\sqrt{c_g d}}, \frac{N}{3HLc_g}, \frac{1}{H^2}\Big\}, \tag{7}$$

the expected decrease in loss at each step for FedMeZO under the i.i.d. scenario is bounded as

$$\mathbb{E}_t\left[f(\theta^{t+1})\right] \leq f(\theta^t) - \left(\frac{2}{\gamma} - \frac{2\zeta}{d}\right)\eta\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + \frac{2\sigma_g^2\zeta\eta L}{NHd} + \frac{\zeta\eta\mu^2L^3}{2NH}, \tag{8}$$

where γ and ζ quantify the effective low-rank properties of the gradient and its estimator, respectively.

In Theorem 3.1, the term $(-2\eta/\gamma)\mathbb{E}_t|\nabla f(\theta^t)|^2$ serves as a critical factor that drives the decrease in the loss function, as it is the sole negative contributor in Eq. (8) such that $\mathbb{E}_t\left[f(\theta^{t+1}) - f(\theta^t)\right] \leq 0$. Note that the presence of the factor $\gamma^{-1} = \Theta(r^{-1})$ underscores the impact of the low effective rank r on the convergence rate (under Assumption 1), revealing that a reduction in r can accelerate convergence independently of the high-dimensional parameter space d. Consequently, even for LLMs with expansive parameter spaces, FedMeZO can attain convergence. This addresses our first foundational question "***Q1****: How does the vast parameter space of LLMs influence the behavior of ZOO-FL?*".

Moreover, the terms $(2\eta\zeta/d)\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2$ and $(2\sigma_g^2\zeta\eta L/NHd)$ are scaled by ζ/d, *i.e.*, in $\Theta(1/rd^2)$, contributing a relatively smaller effect on the convergence speed compared to negative term. This demonstrates that the influence on convergence speed from the zeroth-order gradient estimation is moderated by the model's effective low rank and dimensionality. As for the last term, $(\zeta\eta\mu^2L^3/2NH)$, it acts as a factor slowing down the convergence rate, and we can observe that when N and H are larger, this term becomes smaller. This suggests that the effect of slowing down the convergence rate is not as pronounced, and simultaneously, the perturbation step μ should not be excessively large. Specifically, this term indicates that increasing the number of clients and the number of local rounds can enhance convergence, while also emphasizing the importance of keeping the perturbation step μ moderate.

After gaining intuitive insights in each round of training through the analysis of Theorem 3.1, it is necessary to assess the convergence performance of FedMeZO from a global perspective. We utilize the squared magnitude of the gradient $\mathbb{E}_t\|\nabla f(\theta^t)\|^2$ as a measure to assess the suboptimality of each iterate. The rapidity with which the algorithm approaches a stationary point serves as a crucial metric for determining its efficacy in the context of non-convex optimization problems [44].

Corollary 3.2. ***(Global Convergence in i.i.d. Setting)*** *Assuming the conditions of Theorem 3.1 hold, the global convergence for FedMeZO in the i.i.d. case, characterized by $\Gamma = \frac{d-\zeta\gamma}{d\gamma}$, is given by*

$$\min_{t\in[T]} \mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \leq \frac{f(\theta^0) - f^*}{2\eta T\Gamma} + \frac{\sigma_g^2\zeta L}{NHd\Gamma} + \frac{\zeta\mu^2L^3}{4NH\Gamma} \tag{9}$$

where f^* denotes the optimal loss value.

The upper bound on the minimum squared gradient norm across iterations is composed of three terms in Corollary 3.2. The first term indicates that the distance to the optimal loss relies on the initial state of optimality, while the second and third terms elucidate the influences of stochastic mini-batch errors and the perturbation scale μ inherent to ZOO, respectively. Specifically, they both reflect the impact of the model parameters d and the low effective rank r on the optimal loss. As pointed out in [44], by choosing an appropriate step size, we can obtain the desired accuracy.

Given that $\gamma = \Theta(r)$ and $\zeta = \Theta\left(\frac{1}{rd}\right)$, we have $\zeta\gamma = \Theta\left(\frac{r}{rd}\right) = \Theta\left(\frac{1}{d}\right)$. As the parameter d is large, $d - \zeta\gamma = d - \Theta\left(\frac{1}{d}\right)$ is dominated by d. Consequently, the dominant term of Γ is $\frac{d\gamma}{d-\zeta\gamma}$, which simplifies to $\gamma = \Theta(r)$. Therefore, $\Gamma = \Theta\left(\frac{1}{\gamma}\right) = \Theta\left(\frac{1}{r}\right)$. Building on this relationship, we have the following corollary that articulates the convergence rate of FedMeZO.

Corollary 3.3. ***(Convergence Rate in i.i.d. Setting)*** *Assuming the conditions of Corollary 3.2 hold and given $\eta = (NH)^{1/2}(rT)^{-1/2}$ and $\mu = (NH)^{1/4}r^{-1/2}$, we have*

$$\min_{t\in[T]} \mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \leq O\left(r^{3/2}(NHT)^{-1/2}\right) + O\left(d^{-1}(rNH)^{-1/2}\right). \tag{10}$$

The expression on the right-hand side of Eq. (10) is dominated by $O\left(r^{3/2}(NHT)^{-1/2}\right)$. Consequently, we have derived the convergence rate for FedMeZO. The low effective rank r significantly contributes to lowering the convergence rate, which is also influenced by the number of clients N, the steps of local training iteration H, and the total number of communication rounds T. Moreover, to satisfy the learning rate condition in Eq. (7), the values of N, H, and T must be suitably large.

It is important to note that FedMeZO does not primarily aim to accelerate convergence speed but rather to identify the convergence rate under assumptions pertinent to LLMs. This is intended to demonstrate that FedMeZO can achieve convergence even within a vast parameter space. In a series of studies on federated ZOO, Federated Zeroth-Order Optimization (FedZO) presents the most comprehensive and complete analysis with a convergence rate of $O\left(\sqrt{d/(NHTb_1b_2)}\right)$ [22], which exhibits a lower rate compared to $O\left(d^3/T\right)$ of ZONE-S [26] and accounts for the impact of H compared to $O\left(\sqrt{d/NT}\right)$ of DZOPA [60]. In contrast to FedZO, our method, FedMeZO, theoretically supports a faster convergence by replacing $d^{1/2}$ with $r^{3/2}$ and setting $b_1 = b_2 = 1$. These comparisons show that FedMeZO addresses the challenges posed by large models, offering an efficient convergence rate that relies on r.

This advancement signifies progress in optimizing federated learning algorithms, particularly for LLMs, where the scalability of parameters and data heterogeneity are major challenges. By emphasizing the low effective rank, our approach enhances both the theoretical understanding of convergence behavior in complex settings and the guidance insights into the settings of learning rates and other parameters to achieve efficient convergence outcomes.

However, i.i.d. data is typically encountered in idealized environments. In real-world applications, non-i.i.d. conditions are more common and challenging. Next, we further discuss and analyze the convergence of FedMeZO under non-i.i.d. settings.

3.2 Convergence Analysis in Non-i.i.d Case

Analyzing convergence in the context of non-i.i.d. data distributions is crucial for understanding the behavior of FL algorithms in real-world scenarios. In this section, we extend our convergence analysis to the case where client data distributions are heterogeneous.

Theorem 3.4. ***(Stepwise Loss Descent in Non-i.i.d Setting)*** *Let Assumptions 1-4 and Assumption 6 hold and learning rate η satisfy*

$$\eta \leq \min\Big\{\frac{1}{3HL\sqrt{c_g d}}, \frac{N}{3HLc_g}, \frac{1}{H^2}\Big\}. \tag{11}$$

Then, the expected loss at each step for FedMeZO in the non-i.i.d. setting is bounded as

$$\mathbb{E}_t\left[f(\theta^{t+1})\right] \leq f(\theta^t) - \left(\frac{2}{\gamma N} - \frac{2\zeta\widetilde{c}_h}{d}\right)\eta\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + \frac{2\widetilde{\sigma}^2\zeta L\eta}{NHd} + \frac{\zeta\eta\mu^2L^3}{2NH} - \frac{2}{\gamma N}\eta\sigma_h^2, \tag{12}$$

where $\widetilde{c}_h = c_h + N$ and $\widetilde{\sigma}^2 = 3c_g\sigma_h^2 + \sigma_g^2$.

Comparing Eq. (12) with its i.i.d. counterpart Eq. (8), the non-i.i.d. setting introduces additional terms reflecting data heterogeneity. Firstly, an additional term $\widetilde{c}_h$ appears before $\mathbb{E}_t\|\nabla f(\theta^t)\|^2$; secondly, the original σ_g^2 change into $\widetilde{\sigma}^2$; thirdly, a new term related to σ_h^2 is added at the end. The term $\widetilde{c}_h$ amplifies the effect of the gradient norm, while $\widetilde{\sigma}^2$ encapsulates both the intrinsic stochasticity and data heterogeneity. The presence of σ_h^2 indicates the impact of client data divergence on the convergence behavior, in a degree dependent on $\Theta(1/rd^2)$. Given that the contribution of the negative term accelerates the rate of decline in each round, it can be concluded that heterogeneity is positively correlated with convergence. Considering all the above changes, appropriate heterogeneity can aid in the model convergence.

Experimental results in Section 5.3.3 confirm that a more randomized dataset distribution leads to improved convergence, supporting our theoretical insights. Next, we present the global convergence result for the non-i.i.d. setting building upon Theorem 3.4.

Corollary 3.5. ***(Global Convergence in Non-i.i.d. Setting)*** *Assuming the conditions of Theorem 3.4 hold, denote $\widetilde{\Gamma} = \frac{d-N\gamma\zeta}{d\gamma N}$, FedMeZO satisfies:*

$$\min_{t\in[T]} \mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \leq \frac{f(\theta^0) - f^*}{2\widetilde{\Gamma}\widetilde{c}_h\eta T} + \frac{\widetilde{\sigma}^2\zeta L}{\widetilde{\Gamma}\widetilde{c}_h NHd} + \frac{\zeta\mu^2L^3}{4\widetilde{\Gamma}\widetilde{c}_h NH} - \frac{\sigma_h^2}{\widetilde{\Gamma}\widetilde{c}_h\gamma N}. \tag{13}$$

Given $\gamma = \Theta(r)$ and $\zeta = \Theta\left(\frac{1}{rd}\right)$, the expression $d - N\zeta\gamma$ simplifies to $d - \Theta\left(\frac{N}{d}\right)$ which is dominated by d. Consequently, $\widetilde{\Gamma}$ simplifies to $\Theta\left(\frac{1}{r}\right)$ as in the non-i.i.d. case. Compared to Corollary 3.2, Corollary 3.5 introduces two changes: first, all terms on the right side of Eq. 13 include a denominator c_g, and second, there is an additional term associated with non-i.i.d. heterogeneity, scaled with $\Theta(\sigma_h^2/c_hN)$. This further demonstrates the constraining effect of data heterogeneity in FedMeZO. Similar to Corollary 3.3, by setting appropriate values for η and μ, we obtain the following convergence rate.

Corollary 3.6. ***(Convergence Rate in Non-i.i.d. Setting)*** *Assuming the conditions of Corollary 3.5 hold, with $\eta = (NH)^{1/2}(r\widetilde{c}_hT)^{-1/2}$ and $\mu = (\widetilde{c}hNH)^{1/4}r^{-1/2}$, FedMeZO has convergence rate as follows:*

$$\min_{t\in[T]} \mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \leq O\left(r^{3/2}(\widetilde{c}_hNHT)^{-1/2}\right) + O\left(d^{-1}(r\widetilde{c}_hNH)^{-1/2}\right) - O\left(\sigma_h^2(c_hN)^{-1}\right). \tag{14}$$

The convergence rate in Eq. (14) is primarily driven by the term $O\left(r^{3/2}(\widetilde{c}_hNHT)^{-1/2}\right)$, indicating that optimizing the balance between $\widetilde{c}_h$, c_h, and N is crucial, which reflects a complex interplay of heterogeneity. Specifically, we observe that to achieve better convergence, a smaller $O\left(r^{\frac{3}{2}}(\widetilde{c}_hNHT)^{-\frac{1}{2}}\right)$ is preferred while the $O\left(\sigma_h^2(c_hN)^{-1}\right)$ term need to increase at the same time. Consequently, the balance between c_g, c_h, and N becomes a dynamic trade-off process, *i.e.*, the heterogeneity among different clients directly influences the overall convergence performance.

For now, we have answered the question "***Q2:*** *Can we establish the convergence properties of ZOO-FL for LLMs?*" via theorems and corollaries mentioned in this section. We also validated the nature of convergence under different scenarios and tasks through empirical experiments in Section 5.2.

3.3 Implications

The aforementioned theoretical results offer numerous insights into parameter tuning. A critical revelation from our analysis pertains to the constraints imposed on the learning rate, as delineated in Eq. (7) and Eq. (11), which suggests that an optimal learning rate magnitude is anchored at $1/\sqrt{d}$. Larger learning rates are not only ineffectual but also pose a risk of destabilizing the training dynamic. In Appendix C.3, our empirical experiments corroborate this hypothesis, demonstrating that excessive learning rates precipitate abrupt increases in loss.

Furthermore, our insights regarding the learning rate open up prospects for personalized FL, a compelling approach that uses client-specific configurations to address heterogeneity and has attracted increasing interest [9, 12, 19, 40, 49]. Specifically, we investigate theory-guided personalized strategies by dynamically adjusting the learning rate η_i in proportion to a quantifiable measure of data heterogeneity among clients. In light of Theorem 3.4 that a larger heterogeneity is more conducive to model convergence, it is feasible to appropriately increase the learning rate, allowing specific clients to contribute more to the overall convergence, we propose the following tailored adjustment strategy:

Proposition 3.7. (Adaptive Learning Rate Adjustment) *Let Assumption 6 hold, the learning rate η_i can be adjusted according to the formula to better accommodate the varied learning landscapes than non-personalized FL:*

$$\eta_i = \eta_0(1 + \alpha \cdot \Phi_i), \tag{15}$$

where η_0 represents a default learning rate applicable in a i.i.d. setting, α is a scaling factor that determines the sensitivity of the learning rate and Φ_i is the heterogeneity index, representing the extent of c_g and σ_h^2. This proposition underscores the importance of considering data heterogeneity in the design of the learning rate strategy within personalized FL, offering a structured approach to enhance learning outcomes across diverse client datasets.

In Section 5.4, we empirically confirm that a particular implementation of this strategy facilitates faster convergence. It is important to note that the data heterogeneity index Φ_i cannot be determined a priori; therefore, we utilize several proxy measures during the training process to estimate it. Our goal is not to prescribe an exact solution to this strategy, but rather, through analysis and empirical investigation, to enlighten further research and development of personalized FedMeZO for more effective training of LLMs.

These discussions and corresponding empirical support address the question "**Q3**: *Which model parameters are critical for convergence, and how can we leverage them to optimize FL performance, such as via personalization?*".

Besides, recall that we adopt LoRA to mitigate the communication burden associated with LLMs for practical FL scenarios. Nonetheless, the influence of LoRA on the model's low effective rank, remains an open question. We thus advance the following conjecture under Assumption 1, predicated on existing literature[48, 59], to facilitate further validations:

Conjecture 3.8 (Rank Correlation). *The optimal reparametrization rank r_{LoRA} used in Low-Rank Adaptation (LoRA) is positively proportional to the effective rank r of the Hessian matrix $\mathcal{H}(\theta^t)$ of the tuned LLM. The r_{LoRA} is lower-bounded by r, and can serve as an empirical proxy for r.*

4 PROOF OUTLINE

This section provides an outline of the derivations presented in Section 3, emphasizing the key analytical techniques and concepts employed. Detailed proofs are available in Appendix D of [1].

We begin by taking expectations on both sides of Eq. (6), considering a federated learning setting. The equation is split into two main parts for further analysis:

$$\mathbb{E}_t\left[f(\theta^{t+1})\right] \leq f(\theta^t) - \frac{1}{\gamma}\cdot\eta\mathbb{E}_t\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i(\theta^t)\right\|^2 + \frac{1}{2}\eta^2 L\cdot\zeta\cdot\mathbb{E}_t\left\|\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{H} e_i^{(t,k)}\right\|^2. \tag{16}$$

For simplicity, we denote the two expectation terms as T_1 and T_2, which pertain to the expected squared norms of the gradient and the zeroth-order gradient estimator, respectively.

4.1 Proof of Theorem 3.1

For term T_1 in Eq. (16), we utilize the Cauchy-Schwarz inequality $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ to decompose it into two parts, with the first representing the discrepancy between local and global gradients. By invoking Assumption 5, we establish:

$$T_1 \leq \frac{2}{N}\sum_{i=1}^{N}\mathbb{E}_t\left\|\nabla f_i(\theta^t) - \nabla f(\theta^t)\right\|^2 + 2\cdot\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2.$$

For term T_2, Jensen's inequality allows us to bound the expected squared norm of the zeroth-order gradient estimator as follows:

$$T_2 \leq \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|e_i^{(t,k)}\right\|^2. \tag{17}$$

By substituting Eq. (3) into Eq. (17), we proceed to use the Cauchy-Schwarz inequality to decompose this gradient estimator into two parts, each of which is a biased estimator. Recall that in our gradient estimator, z_i follows a Gaussian distribution. Therefore, the impact on the norm caused by a forward step and a backward step of the estimator is identical. Consequently, we ascertain that the term T_2 is bounded by a single-point gradient estimation as $\mathbb{E}_t\left\|\frac{z_i^{(t,k)}}{\mu}\left(F_i(\theta_i^{(t,k)} + \mu z_i^{(t,k)}, \mathcal{B}_i^{(t,k)}) - F_i(\theta_i^{(t,k)}, \mathcal{B}_i^{(t,k)})\right)\right\|^2$.

Following Lemma 4.1 in [23], we can bound the expectation term as:

$$\begin{aligned}\mathbb{E}_t\left\|e_i^{(t,k)}\right\|^2 &\leq \frac{1}{d^2}\left[2d\cdot\mathbb{E}_t\left\|\nabla F_i(\theta_i^{(t,k)}, \mathcal{B}_i^{(t,k)})\right\|^2 + \frac{\mu^2}{2}L^2 d^2\right] \\ &\leq \frac{1}{d^2}\left[2c_g d\cdot\mathbb{E}_t\left\|\nabla f_i(\theta_i^{(t,k)})\right\|^2 + 2d\sigma_g^2 + \frac{\mu^2}{2}L^2 d^2\right],\end{aligned} \tag{18}$$

where the second inequality is derived based on Assumption 4. Subsequently, we bound the expectation term by applying the Cauchy-Schwartz inequality to divide it into three parts:

$$\begin{aligned}\mathbb{E}_t\left\|\nabla f_i(\theta_i^{(t,k)})\right\|^2 &= \mathbb{E}_t\left\|\nabla f_i(\theta_i^{(t,k)}) \mp \nabla f_i(\theta^t) \mp \nabla f(\theta_i^t)\right\|^2 \\ &\leq 3L^2\mathbb{E}_t\left\|\theta_i^{(t,k)} - \theta^t\right\|^2 + 3\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2.\end{aligned} \tag{19}$$

The first part represents the gradient difference between stages (t,k) and $(t,0)$, which can be computed using Assumption 3, *i.e.*, the L-smooth condition. The second part signifies the disparity between local and global aspects, calculated using Assumption 5. The third part is retained as is.

Combining Equations (17), (18) and (19), we bound T_2 as follow:

$$\begin{aligned}T_2 \leq\ & \frac{6c_g L^2}{Nd}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)} - \theta^t\right\|^2 \\ &+ \frac{6c_g H}{d}\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + \frac{2H\sigma_g^2}{d} + \frac{\mu^2 HL^2}{2}.\end{aligned} \tag{20}$$

Next, combining Equations (16), (17) and (20), we have:

$$\begin{aligned}\mathbb{E}_t\left[f(\theta^{t+1})\right] - f(\theta^t) \leq\ & \left(\frac{3c_g\zeta\eta^2 HL}{d} - \frac{2\eta}{\gamma}\right)\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \\ &+ \frac{3c_g\zeta\eta^2 L^3}{Nd}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)} - \theta^t\right\|^2 \\ &+ \frac{\sigma_g^2\zeta\eta^2 HL}{d} + \frac{\zeta\eta^2\mu^2 HL^3}{4}.\end{aligned} \tag{21}$$

In Eq. (21), $\mathbb{E}_t\left\|\theta_i^{(t,k)} - \theta^t\right\|^2$ remains unknown and we need to constrain it further. The key idea is to transform this expectation term into a form related to $\mathbb{E}_t^k\|e_i^{(t,k)}\|^2$ and then utilize the conclusion of Eq. (18) and Eq. (19) for computation, we can have the bounded result:

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)}-\theta^t\right\|^2 \le \frac{C_1}{C_0}, \tag{22}$$

where $C_0 = 1 - 3c_g d\eta^2 H^2 L^2$ and $C_1 = 2c_g dH^3\eta^2\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + \frac{2}{3}d\sigma_g^2 H^3\eta^2 + \frac{\mu^2 L^2 d^2 H^3 \eta^2}{6}$.

Finally, by substituting Eq. (22) into Eq. (21), we obtain the final result of the stepwise descent. After simplifying and appropriately setting the learning rate, we arrive at the result presented in Theorem 3.1. The detailed derivation of this result is provided in Appendix D.1 of [1].

4.2 Proof of Theorem 3.4

The proof for the non-i.i.d. case follows a similar structure to that of the i.i.d. case, with adjustments made for the heterogeneity between local and global models as captured by c_h and σ_h^2 (Assumption 6). In particular, we redefine T_1 in Eq. (16) as $\widetilde{T_1}$ to reflect the increased variance due to non-i.i.d. data:

$$\begin{aligned}\widetilde{T_1} &\le \frac{2}{N}\sum_{i=1}^{N}\mathbb{E}_t\left\|\nabla f_i(\theta^t) - \nabla f(\theta^t)\right\|^2 + 2\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \\ &\le 2(1+c_h)\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + 2\sigma_h^2, \end{aligned} \tag{23}$$

where the second inequality follows Assumption 6.

Subsequently, $\widetilde{T_2}$ is computed similarly, with the heterogeneity terms incorporated. The difference from the i.i.d. case lies in the bounding of expectation term in Eq. (19):

$$\begin{aligned}\mathbb{E}_t\left\|\nabla f_i(\theta_i^{(t,k)})\right\|^2 \le 3L^2\mathbb{E}_t&\left\|\theta_i^{(t,k)}-\theta^t\right\|^2 \\ &+ 3(c_h+1)\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + 3\sigma_h^2, \end{aligned} \tag{24}$$

where the inequality employs Assumption 6.

Combining Equations (17), (18) and (24), we bound $\widetilde{T_2}$ as follow:

$$\begin{aligned}\widetilde{T_2} \le \frac{6c_g L^2}{Nd}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)}-\theta^t\right\|^2 &+ \frac{6c_g(c_h+1)H}{d}\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \\ &+ \frac{6c_g\sigma_h^2 H}{d} + \frac{2H\sigma_g^2}{d} + \frac{\mu^2 HL^2}{2}. \end{aligned} \tag{25}$$

By substituting $\widetilde{T_1}$ in Eq. (23) and $\widetilde{T_2}$ in Eq. (25) into Eq. (16), we get the result under the non-i.i.d. condition as follow:

$$\begin{aligned}\mathbb{E}_t\left[f(\theta^{t+1})\right] - f(\theta^t) \le &\left(\frac{3c_g\widetilde{c}_h\zeta\eta^2 HL}{d} - \frac{2\widetilde{c}_h\eta}{\gamma}\right)\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 \\ &+ \frac{3c_g\zeta\eta^2L^3}{Nd}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)}-\theta^t\right\|^2 \\ &+ \frac{\widetilde{\sigma}^2\zeta\eta^2 HL}{d} + \frac{\zeta\eta^2\mu^2 HL^3}{4} - \frac{2}{\gamma}\sigma_h^2\eta, \end{aligned} \tag{26}$$

where $\widetilde{c}_h$ denotes (c_h+1) and $\widetilde{\sigma}^2$ denotes $(3c_g\sigma_h^2 + \sigma_g^2)$. We then need to make the expectation term bounded. Unlike Eq. (22), due to the variations introduced by Eq. (24), two additional terms related to $\widetilde{c}_h$ and $\widetilde{\sigma}^2$ emerge, yielding the following result:

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{H}\mathbb{E}_t\left\|\theta_i^{(t,k)}-\theta^t\right\|^2 \le \frac{C_2}{C_0}, \tag{27}$$

where $C_1 = 1 - 3c_g d\eta^2 H^2 L^2$ and $C_2 = 2c_g d\widetilde{c}_h H^3\eta^2\mathbb{E}_t\left\|\nabla f(\theta^t)\right\|^2 + \frac{2}{3}\widetilde{\sigma}^2 dH^3\eta^2 + \frac{\mu^2 L^2 d^2 H^3\eta^2}{6}$.

Finally, we can substitute Eq. (27) into Eq. (26) and have the result of Theorem 3.4. The detailed derivations about these steps of Theorem 3.4 are provided in Appendix D.3 of [1].

5 EMPIRICAL SUPPORT

This section aims to empirically validate our theoretical findings through a series of experiments.

5.1 Experimental Setup

We utilize LLaMA-3B [52] as the foundational model and employ four datasets covering a range of tasks and data distribution types to provide comprehensive validation of our theoretical results [32]. Given that our theory centers on the loss function, we primarily focus on analyzing loss descent in our experiments. More details of the used datasets are in the Appendix (Table 2).

We set the total number of communication rounds to 500. By default, BP-based baselines undertake local training for one epoch, whereas FedMeZO conducts local training for 30 steps. We repeat our experiments with three seeds and plot the error bars. For more detailed implementation specifics, please refer to Appendix B.

5.2 Convergence Study

To assess the convergence of FedMeZO, we perform experiments on three datasets using different data splitters, as specified in Table 2, with test loss serving as the convergence metric. Our objective is to evaluate the generalization and stability of FedMeZO across diverse datasets and heterogeneity scenarios. For benchmarking purposes, we also measure the performance of BP-based FedAvg on the same datasets. Additionally, we document the GPU memory usage during training in Table 1. Representative findings are illustrated in Figure 1, while the comprehensive results are available in Appendix F.11 of [1] due to the space limitation.

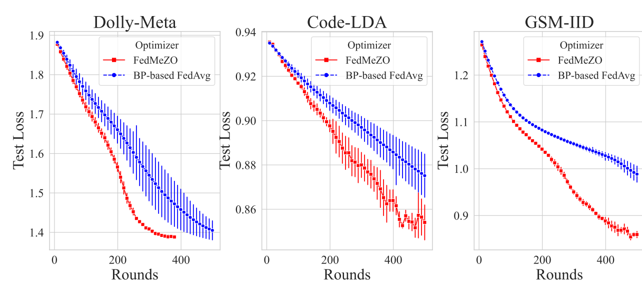

Figure 1: Convergence comparison of FedMeZO and BP-based FedAvg.

Two main conclusions emerge from the convergence experiments: First, when the learning rate complies with the requirements discussed in Section 3.3, as stipulated by Theorem 3.1 and

Table 1: The GPU Memory of BP-based FedAvg and FedMeZO.

Task	BP-based FedAvg (MiB)	FedMeZO (MiB)
Dolly-Meta	26571	10061
GSM8K-IID	17771	9733
CodeAlpaca-LDA	15287	9569

Theorem 3.4, FedMeZO consistently diminishes loss with each step, ultimately achieving stable convergence. Second, under equivalent learning rate configurations, FedMeZO decreases loss more rapidly than BP-based FedAvg, indicating a swifter convergence rate. For instance, in the Dolly-Meta figure, FedMeZO stabilizes and converges around 300 rounds, whereas BP-based FedAvg's loss is still declining at this juncture. Notably, from Table 1, we observe that the GPU memory demand for FedMeZO is roughly one-half of that required by BP-based FedAvg, suggesting that FedMeZO can achieve a speedier convergence with fewer resources.

5.3 Hyper-parameters Study

In this subsection, we perform a series of experiments to ascertain the influence of various hyper-parameters, as intimated by our theoretical findings.

5.3.1 Impact of Perturbation Scale μ. To corroborate the theoretical impacts of the perturbation step on convergence, we examine μ values of 5×10^{-3} and 2×10^{-4}, in addition to the default $\mu = 1 \times 10^{-3}$. We leverage the same datasets and splitters as in Section 5.2 for robustness. Figure 2 shows representative outcomes, with comprehensive results in Appendix F.9 of [1].

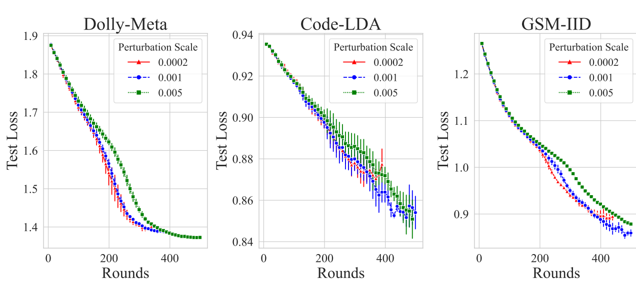

Figure 2: Effects of different perturbation scales μ.

The results confirm that, consistent with Equations (10) and (14), a smaller μ marginally expedites model convergence. Figure 2 exemplifies that the training trajectory with $\mu = 2 \times 10^{-4}$ descends more rapidly than the others. However, given that μ appears as a second-order term in $\frac{\zeta\mu^2 L^3}{4\widetilde{\Gamma c}_h NH}$ and its absolute value is relatively small, its overall influence is modest. This is evident in Figure 2, where modifications to μ within a specific range yield only slight variations. Thus, a smaller μ proves advantageous for model convergence.

5.3.2 Impact of Local Iteration Step H. To validate the theoretical impact of local iteration steps on convergence, we contrast $H = 10$ and $H = 50$ with the standard $H = 30$. Utilizing identical datasets

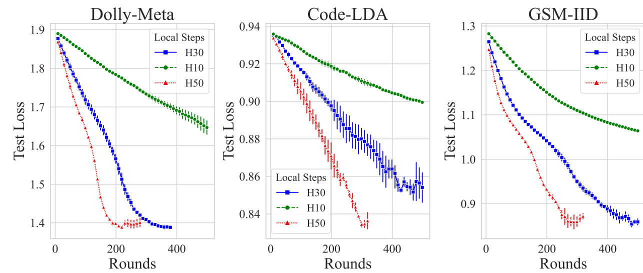

Figure 3: Effects of different local iteration steps H.

and splitters from Section 5.2, we present typical findings in Figure 3, with all the detailed results forthcoming in Appendix F.10 of [1].

These experimental results suggest that a lower H engenders a more sluggish convergence pace, whereas a higher H somewhat propels convergence, mirroring the impact of H as a denominator in the theoretical convergence rate analysis. Nonetheless, an excessive H may lead to instability, as depicted by the curve of $H = 50$, which exhibits a surge endwise in the figure. Hence, an appropriate choice of H facilitates efficient model convergence.

5.3.3 Analysis of Other Hyper-parameters. We also explore the ramifications of learning rate, data splitters, the number of clients N and the model size on the convergence rate. Due to the space limit, details pertaining to these experimental settings and results are presented in Appendices C.3, C.4, C.5, and C.6 respectively.

In a nutshell, as shown in Fig. 5, a suitable learning rate anchored at $1/\sqrt{d}$ leads to stabilized training dynamics, while larger ones exhibit divergence as suggested by our analysis (Section 3.3). Besides, dissimilar splitters symbolize varying extents of data heterogeneity, and we have observations revealing that augmented heterogeneity culminates in lower stabilized loss values (as shown in Fig. 6). This intimates that a moderate degree of data heterogeneity can elevate the model's convergence proficiency. Moreover, Fig. 7 verifies our theoretical analysis in Section 3 that an increase in N helps stabilize the global convergence.

5.4 Personalization Study

To reconcile Proposition 3.7 with practical scenarios, we conduct the following subsequent experiments. To account for each client's heterogeneity during model updates in each round, we derive three signal quantities: (1) *Round-wise Train Loss Difference*: The discrepancy between each client's loss in the preceding training round and the global loss. (2) *Five-round Average Train Loss Difference*: The average loss deviation for each client relative to the global loss over the antecedent five rounds. (3) *Model Parameter Update Difference*: The disparity between each client's previous round parameter updates and the global update magnitude. We normalize them to the range of $(-1, 1)$, serving as empirical estimates for Φ.

For the setting of the scaling factor α, following the guidance of the learning rate in Section 3.3, we designate 1.5×10^{-5} as the maximal learning rate, potentially leading to surges as per the learning rate search network. Symmetrically, we posit the minimal value at 5×10^{-6}, anchored on the default learning rate of 1×10^{-5}, thereby assigning α a value of 5×10^{-6}.

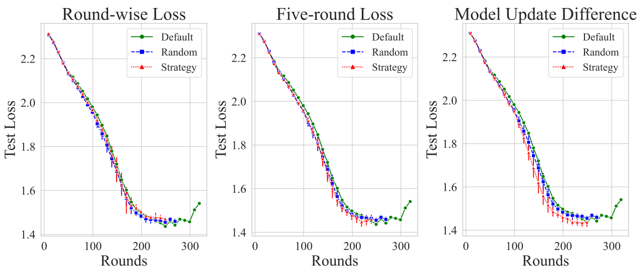

Figure 4: Comparison of different strategies of learning rate adjustment. "Default" indicates non-personalized case, and "Round-wise Loss", "Five-round Loss" and "Model Update Difference" indicate three signal quantities leveraged.

As counterpoints, we furnish two configuration strategies for the learning rate adjustment: one uniformly applies a default learning rate of 1×10^{-5}, while the other randomly selects within $[5 \times 10^{-6}, 1.5 \times 10^{-5}]$ for each round. The conclusive results are depicted in Figure 4. Based on the experimental results, we observe that our method achieves faster loss convergence with the second and third types of signal quantities compared to the default and random settings, with the third type yielding the most impressive performance. In contrast, the first type of signal quantity had a negligible impact. These results suggest that while round-wise loss exhibits some degree of randomness, aggregating losses over multiple rounds can approximate heterogeneity to a meaningful extent, thus serving as an indicator to expedite model convergence.

It is also noteworthy that the third type of signal quantity aligns with the expression $\|\nabla f(\theta^t) - \sum_{k=0}^{H} \eta_i \nabla f_i(\theta^{(t,k)})\|^2$, which most closely reflects Assumption 6. Consequently, it demonstrates the most effective performance in the experiments, not only achieving the fastest convergence but also the lowest stable loss. This case study experiment substantiates the efficacy of Proposition 3.7, offering valuable insights for parameter tuning in personalized FL.

6 CONCLUSION

This work investigates the convergence of FedMeZO, a practical approach integrating Memory-efficient Zeroth-Order Optimization within a federated learning setting for Large Language Models (LLMs). Extensive empirical results verified our analyses and indicated that FedMeZO achieves fast convergence with reduced GPU memory requirements, offering a promising alternative to traditional optimization methods. The incorporation of a personalized learning rate adjustment, derived from theoretical analysis, has been shown to effectively enhance loss reduction. Through this study, we aim to enlighten more research and development of memory-efficient optimization techniques to address practical challenges associated with the fine-tuning of LLMs, particularly in resource-constrained scenarios [29].

REFERENCES

[1] Extended version of this paper. arXiv preprint arXiv:2402.05926.
[2] Alekh Agarwal, Martin J Wainwright, Peter Bartlett, and Pradeep Ravikumar. 2009. Information-theoretic lower bounds on the oracle complexity of convex optimization. *Advances in Neural Information Processing Systems* 22 (2009).
[3] Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 7319–7328.
[4] Jiamu Bai, Daoyuan Chen, Bingchen Qian, Liuyi Yao, and Yaliang Li. 2024. Federated Fine-tuning of Large Language Models under Heterogeneous Language Tasks and Client Resources. *arXiv preprint arXiv:2402.11505* (2024).
[5] Léon Bottou, Frank E Curtis, and Jorge Nocedal. 2018. Optimization methods for large-scale machine learning. *SIAM review* 60, 2 (2018), 223–311.
[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
[7] Sahil Chaudhary. 2023. Code alpaca: An instruction-following llama model for code generation.
[8] Chaochao Chen, Xiaohua Feng, Jun Zhou, Jianwei Yin, and Xiaolin Zheng. 2023. Federated large language model: A position paper. *arXiv preprint arXiv:2307.08925* (2023).
[9] Daoyuan Chen, Dawei Gao, Weirui Kuang, Yaliang Li, and Bolin Ding. 2022. pFL-Bench: A Comprehensive Benchmark for Personalized Federated Learning. In *NeurIPS*.
[10] Daoyuan Chen, Dawei Gao, Yuexiang Xie, Xuchen Pan, Zitao Li, Yaliang Li, Bolin Ding, and Jingren Zhou. 2023. FS-REAL: Towards Real-World Cross-Device Federated Learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3829–3841.
[11] Daoyuan Chen, Yilun Huang, Zhijian Ma, Hesen Chen, Xuchen Pan, Ce Ge, Dawei Gao, Yuexiang Xie, Zhaoyang Liu, Jinyang Gao, Yaliang Li, Bolin Ding, and Jingren Zhou. 2024. Data-Juicer: A One-Stop Data Processing System for Large Language Models. In *International Conference on Management of Data*.
[12] Daoyuan Chen, Liuyi Yao, Dawei Gao, Bolin Ding, and Yaliang Li. 2023. Efficient Personalized Federated Learning via Sparse Model-Adaptation. In *International Conference on Machine Learning, ICML*, Vol. 202. 5234–5256.
[13] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374* (2021).
[14] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168* (2021).
[15] Mike Conover, Matt Hayes, Ankit Mathur, Xiangrui Meng, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, et al. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.
[16] Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. *Free Dolly: Introducing the World's First Truly Open Instruction-Tuned LLM.*
[17] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2021. Glm: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360* (2021).
[18] John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. 2015. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory* 61, 5 (2015), 2788–2806.
[19] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. 2020. Personalized federated learning: A meta-learning approach. In *NeurIPS 2020*.
[20] Tao Fan, Yan Kang, Guoqiang Ma, Weijing Chen, Wenbin Wei, Lixin Fan, and Qiang Yang. 2023. FATE-LLM: A Industrial Grade Federated Learning Framework for Large Language Models. *CoRR* abs/2310.10049 (2023).
[21] Tao Fan, Yan Kang, Guoqiang Ma, Weijing Chen, Wenbin Wei, Lixin Fan, and Qiang Yang. 2023. Fate-llm: A industrial grade federated learning framework for large language models. *arXiv preprint arXiv:2310.10049* (2023).
[22] Wenzhi Fang, Ziyi Yu, Yuning Jiang, Yuanming Shi, Colin N Jones, and Yong Zhou. 2022. Communication-efficient stochastic zeroth-order optimization for federated learning. *IEEE Transactions on Signal Processing* 70 (2022), 5058–5073.
[23] Xiang Gao, Bo Jiang, and Shuzhong Zhang. 2018. On the information-adaptive variants of the ADMM: an iteration complexity perspective. *Journal of Scientific Computing* 76 (2018), 327–363.
[24] Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. 2019. An investigation into neural net optimization via hessian eigenvalue density. In *International Conference on Machine Learning*. PMLR, 2232–2241.
[25] Jiaqi Gu, Chenghao Feng, Zheng Zhao, Zhoufeng Ying, Ray T Chen, and David Z Pan. 2021. Efficient on-chip learning for optical neural networks through power-aware sparse zeroth-order optimization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 7583–7591.
[26] Davood Hajinezhad, Mingyi Hong, and Alfredo Garcia. 2019. ZONE: Zeroth-order nonconvex multiagent optimization over networks. *IEEE Trans. Automat. Control* 64, 10 (2019), 3995–4010.
[27] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300* (2020).

[28] Junyuan Hong, Zhuangdi Zhu, Shuyang Yu, Zhangyang Wang, Hiroko Dodge, and Jiayu Zhou. 2021. Federated Adversarial Debiasing for Fair and Transferable Representations. In *KDD*.

[29] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nZeVKeeFYf9

[30] Kevin G Jamieson, Robert Nowak, and Ben Recht. 2012. Query complexity of derivative-free optimization. *Advances in Neural Information Processing Systems* 25 (2012).

[31] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. 2021. Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning* 14, 1–2 (2021), 1–210.

[32] Weirui Kuang, Bingchen Qian, Zitao Li, Daoyuan Chen, Dawei Gao, Xuchen Pan, Yuexiang Xie, Yaliang Li, Bolin Ding, and Jingren Zhou. 2023. Federatedscope-LLM: A comprehensive package for fine-tuning large language models in federated learning. *arXiv preprint arXiv:2309.00363* (2023).

[33] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. 2018. Measuring the Intrinsic Dimension of Objective Landscapes. In *International Conference on Learning Representations*.

[34] LeiLai Li, Jianzong Wang, Xiaoyang Qu, and Jing Xiao. 2021. Communication-memory-efficient decentralized learning for audio representation. In *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[35] Mu Li, Tong Zhang, Yuqiang Chen, and Alexander J Smola. 2014. Efficient mini-batch training for stochastic optimization. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 661–670.

[36] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems* 2 (2020), 429–450.

[37] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2019. On the Convergence of FedAvg on Non-IID Data. In *ICLR*.

[38] Sijia Liu, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred O Hero III, and Pramod K Varshney. 2020. A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications. *IEEE Signal Processing Magazine* 37, 5 (2020), 43–54.

[39] Sadhika Malladi, Tianyu Gao, Eshaan Nichani, Alex Damian, Jason D Lee, Danqi Chen, and Sanjeev Arora. 2023. Fine-Tuning Language Models with Just Forward Passes. *Advances in Neural Information Processing Systems* (2023).

[40] Othmane Marfoq, Chuan Xu, Giovanni Neglia, and Richard Vidal. 2020. Throughput-Optimal Topology Design for Cross-Silo Federated Learning. In *NeurIPS*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 19478–19487. https://proceedings.neurips.cc/paper/2020/file/e29b722e35040b88678e25a1ec032a21-Paper.pdf

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

[42] Chuizheng Meng, Sirisha Rambhatla, and Yan Liu. 2021. Cross-node federated graph neural network for spatio-temporal data modeling. In *KDD*. 1202–1211.

[43] Khalil Muhammad, Qinqin Wang, Diarmuid O'Reilly-Morgan, Elias Tragos, Barry Smyth, Neil Hurley, James Geraci, and Aonghus Lawlor. 2020. Fedfast: Going beyond average for faster training of federated recommender systems. In *KDD*. 1234–1242.

[44] Yurii Nesterov and Vladimir Spokoiny. 2017. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics* 17 (2017), 527–566.

[45] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).

[46] Zhen Qin, Daoyuan Chen, Bingchen Qian, Bolin Ding, Yaliang Li, and Shuiguang Deng. 2024. Federated Full-Parameter Tuning of Billion-Sized Language Models with Communication Cost under 18 Kilobytes. arXiv:2312.06353 [cs.LG]

[47] Xinchi Qiu, Javier Fernandez-Marques, Pedro PB Gusmao, Yan Gao, Titouan Parcollet, and Nicholas Donald Lane. 2022. ZeroFL: Efficient On-Device Training for Federated Learning with Local Sparsity. In *International Conference on Learning Representations*.

[48] Levent Sagun, Utku Evci, V Ugur Guney, Yann Dauphin, and Leon Bottou. 2017. Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454* (2017).

[49] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. 2020. Clustered Federated Learning: Model-Agnostic Distributed Multitask Optimization Under Privacy Constraints. *TNNLS* (2020).

[50] Yao Shu, Xiaoqiang Lin, Zhongxiang Dai, and Bryan Kian Hsiang Low. 2023. Federated Zeroth-Order Optimization using Trajectory-Informed Surrogate Gradients. *arXiv preprint arXiv:2308.04077* (2023).

[51] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

[52] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).

[53] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

[54] Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H Brendan McMahan, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, et al. 2021. A field guide to federated optimization. *arXiv preprint arXiv:2107.06917* (2021).

[55] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. 2021. A novel framework for the analysis and design of heterogeneous federated learning. *IEEE Transactions on Signal Processing* 69 (2021), 5234–5249.

[56] Zhen Wang, Weirui Kuang, Yuexiang Xie, Liuyi Yao, Yaliang Li, Bolin Ding, and Jingren Zhou. 2022. FederatedScope-GNN: Towards a Unified, Comprehensive and Efficient Package for Federated Graph Learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 4110–4120.

[57] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*. 38–45.

[58] Yuexiang Xie, Zhen Wang, Dawei Gao, Daoyuan Chen, Liuyi Yao, Weirui Kuang, Yaliang Li, Bolin Ding, and Jingren Zhou. 2023. FederatedScope: A Flexible Federated Learning Platform for Heterogeneity. *Proceedings of the VLDB Endowment* 16, 5 (2023), 1059–1072.

[59] Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W Mahoney. 2020. Pyhessian: Neural networks through the lens of the hessian. In *2020 IEEE international conference on big data (Big data)*. IEEE, 581–590.

[60] Xinlei Yi, Shengjun Zhang, Tao Yang, and Karl H Johansson. 2022. Zeroth-order algorithms for stochastic distributed nonconvex optimization. *Automatica* 142 (2022), 110353.

[61] Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Guoyin Wang, and Yiran Chen. 2023. Towards Building the Federated GPT: Federated Instruction Tuning. *arXiv preprint arXiv:2305.05644* (2023).

[62] Jianyi Zhang, Saeed Vahidian, Martin Kuo, Chunyuan Li, Ruiyi Zhang, Tong Yu, Guoyin Wang, and Yiran Chen. 2023. Towards Building the FederatedGPT: Federated Instruction Tuning. In *International Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS 2023*. https://openreview.net/forum?id=TaDiklyVps

[63] Qingsong Zhang, Bin Gu, Zhiyuan Dang, Cheng Deng, and Heng Huang. 2021. Desirable companion for vertical federated learning: New Zeroth-order gradient based algorithm. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2598–2607.

[64] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068* (2022).

[65] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223* (2023).

APPENDIX

A DETAILED ASSUMPTIONS

Assumption 2. ***(Bounded Loss)*** *The global loss function $f(\theta)$ is bounded below by a scalar f^*, i.e., $f^* \geq f(\theta) > -\infty$ for all θ.*

Assumption 3. *(L-smoothness) The local and global loss functions $F_i(\theta, \mathcal{B}_i)$, $f_i(\theta)$, and $f(\theta)$ are L-smooth. Mathematically, for any $\theta_1, \theta_2 \in \mathbb{R}^d$, it holds that*

$$\|\nabla f_i(\theta_2) - \nabla f_i(\theta_1)\| \leq L\|\theta_2 - \theta_1\|,$$

$$f_i(\theta_2) \leq f_i(\theta_1) + \langle \nabla f_i(\theta), \theta_2 - \theta_1 \rangle + \frac{L}{2}\|\theta_2 - \theta_1\|^2.$$

Assumption 4. ***(Mini-batch Gradient Error Bound)*** *For any $\theta \in \mathbb{R}^d$, the second-order moment of the stochastic gradient is bounded by $\mathbb{E}_{\mathcal{B}_i}\|\nabla F_i(\theta, \mathcal{B}_i)\|^2 \leq c_g\|\nabla f_i(\theta)\|^2 + \sigma_g^2$, where $c_g \geq 1$.*

Assumption 5. ***(Global-Local Disparities in i.i.d. Setting)*** *For any $\theta \in \mathbb{R}^d$, the discrepancy between the local and global gradient is negligible, i.e., $\mathbb{E}_i\|\nabla f(\theta) - \nabla f_i(\theta)\|^2 = 0$.*

Assumption 6. ***(Global-Local Disparities in Non-i.i.d. Setting)*** *For any $\theta \in \mathbb{R}^d$, the discrepancy between the local and global gradient is bounded by $\|\nabla f(\theta) - \nabla f_i(\theta)\|^2 \leq c_h\|\nabla f(\theta)\|^2 + \sigma_h^2$, where c_h is a positive constant.*

Assumptions 2-4 are well-established in the literature on large-scale stochastic optimization [5]. Assumption 5 describes an ideal i.i.d. setting where each client's gradient is aligned with the global gradient. Assumption 6 accounts for the heterogeneity of client data distributions that is typical in non-i.i.d. settings [36, 37, 55].

B IMPLEMENTATION DETAILS

B.1 Datasets

We adopt several federated tuning datasets tailored for LLMs from [32], with different splitting strategies to simulate the heterogeneity typical of different federated learning (FL) scenarios, including a uniform distribution of data, a Dirichlet distribution of data, and a splitter based on meta-information. The proportion of test data for each dataset is 1%.

Table 2: Datasets and Basic Information.

Name	#Sample	Domain
Fed-Alpaca [51]	52.0k	Generic Language
Fed-Dolly [15]	15.0k	Generic Language
Fed-GSM8K [14]	7.5k	CoT
Fed-CodeAlpaca [7]	8.0k	Code Generation

B.2 Experimental Platforms

We implement our approaches using PyTorch [45] v1.10.1, coupled with PEFT v0.3.0 and the Transformers library [57] v4.29.2. Experiments with LLaMA-3B are conducted on a computing platform equipped with four NVIDIA A100 GPUs (40GB), with pre-trained LLMs loaded as 16-bit floating-point numbers.

B.3 Default Implementation Settings

Following the guidelines in [32, 39], all approaches perform local training with a batch size of 1. To standardize the experimental conditions, both BP-based and FedMeZO train locally with specific learning rates: $\eta = 1 \times 10^{-5}$ for the *Fed-Dolly* and *Fed-Alpaca* datasets, $\eta = 2 \times 10^{-5}$ for the *Fed-CodeAlpaca* dataset, and $\eta = 2.5 \times 10^{-5}$ for the *Fed-GSM8K* dataset. The rank and alpha parameters for LoRA are set to 128 and 256. As per [39], the perturbation scale μ for FedMeZO is set to 1×10^{-3}. We employed the early stopping mechanism following previous works [9]. The training was stopped if there was no improvement in the validation loss within 30 epochs. The best model was selected from the epoch with the lowest validation loss. We conducted three sets of experiments with randomly selected seeds, and calculated the mean as the line plot with a 90% confidence interval as the error bar.

C SUPPLEMENTARY EXPERIMENTS

C.1 Communication Cost of FedMeZO

In Section 2.2, we mention using LoRA [29] to reduce the substantial communication overhead. We theoretically prove this method's equivalence to full-parameter transmission as follows:

$$\theta_i^{t+1,0} = \frac{1}{N}\sum_{i=1}^{N}\theta_i^{t,H} = \frac{1}{N}\sum_{i=1}^{N}\left[\theta_i^{t+1,0} + \sum_{k=1}^{H}\nabla_{lora}\theta_i^{t,k}\right]$$
$$= \theta_i^{t,0} + \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{H}\nabla_{lora}\theta_i^{t,k} \qquad (28)$$

Note that the left side represents the client's parameters with FedAvg, while the right side corresponds to FedMeZO, and they are equivalent. By utilizing LoRA, FedMeZO achieves the same effect with just 1.23% of parameters in our setting. Specifically, full parameter transmission needs 6.39GB, while LoRA demands merely 80.45MB.

C.2 Evaluations with Common LLM's Metrics

We examined FedMeZO with some commonly used metrics for LLMs evaluations: Dolly with MMLU metrics [27], Code with OpenAI HumanEval metrics [13], and GSM8K with CoT metrics[14]. We evaluated FedMeZO and BP-based FedAvg at model checkpoints of rounds 0, 100, 200 and present the results in Table 3. The results verified FedMeZO's effectiveness again:

- Fine-tuning LLMs with FedMeZO effectively improves their performance on specific tasks;
- FedMeZO gains better performance compared to BP-based FedAvg.

Table 3: Evaluations of FedMeZO with LLM's Metrics

Rounds	Dolly (%)		Code (%)		GSM8K (%)	
	FedMeZO	BP	FedMeZO	BP	FedMeZO	BP
0	26	26	8.53	8.53	3.41	3.41
100	**26.6**	26.1	**8.66**	8.53	**4.32**	4.25
200	**26.3**	26.2	8.41	**8.53**	**4.62**	4.40

C.3 Conditions for the Learning Rate

We disabled the early-stop mechanism and sequentially selected four learning rates: 1×10^{-5}, 3×10^{-5}, 5×10^{-5}, and 1×10^{-4} on the GSM-IID dataset, then conducted training for 500 rounds each.

The results are presented in Figure 5, which indicate that the loss function exhibits a sharp increase when the learning rate exceeds the range supported by theory. Meanwhile, it is noteworthy that our theory suggests that an optimal learning rate magnitude is anchored at $1/\sqrt{d}$ in Section 3.3. In our chosen model, LLaMA-3B, where d can be set to 3×10^9, this gives $1/\sqrt{d} = 1.826 \times 10^{-5}$, which is approximately the learning rate we aim to use.

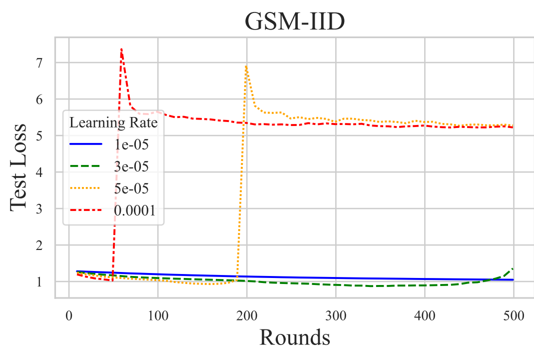

Figure 5: Phenomenon of loss surge due to larger learning rates.

C.4 The Impact of Heterogeneity on Convergence

We present the results of processing the same dataset with different splitters in Figure 6. It is observable that in the Dolly and CodeAlpaca datasets, the LDA and Meta splitters perform better than IID, with Meta being the best. Noting that the data classified by Meta and LDA are non-i.i.d., this indicates that higher data heterogeneity is more conducive to model convergence.

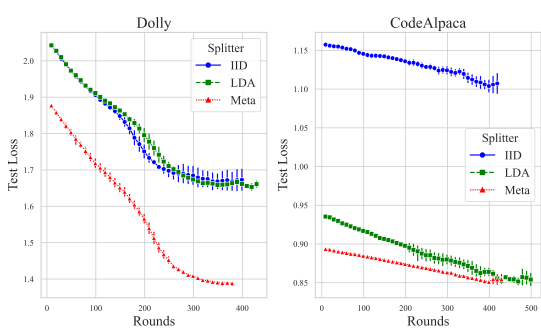

Figure 6: Effects of different splitters on the same dataset.

C.5 The Impact of Client Number

In Figure 7, we showcase the outcomes of federated learning on the same Alpaca dataset with 3 clients and 8 clients, respectively. This demonstrates that the model converges more stably with more clients participating in the training. This conclusion also corresponds to the theoretical results regarding the number of clients N discussed in Section 3, i.e., an increase in N is beneficial for reducing global convergence.

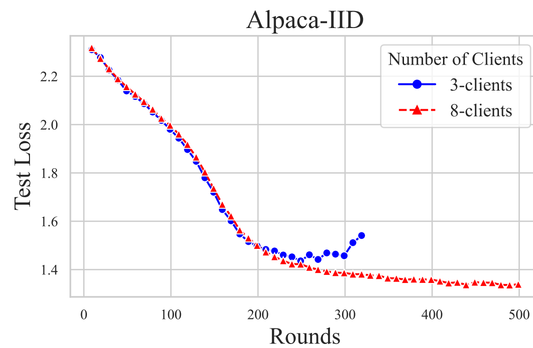

Figure 7: Effects of the number of clients.

C.6 The Impact of Model Size

To investigate the and versatility of FedMeZO across different model sizes, based on LLaMA2-7B [53], we conducted experiments using the same experimental setup on three datasets: Dolly-Meta, CodeAlpaca-LDA and GSM8K-IID. The experimental results are shown in Table 4. The results show that FedMeZO on LLaMA2-7B retains similar trends, while starts with lower Loss than 3B-model by 0.2 and 0.12 in Dolly-Meta and GSM8K-IID. For all tasks, 7B-model's loss decreases more slowly, with reductions at 300 rounds being only 34%, 67%, and 41% of the 3B-model's in Dolly-Meta, Code-LDA, and GSM8K-IID.

Table 4: Test loss on LLaMA-3B and LLaMA2-7B.

Rounds	Dolly-Meta		Code-LDA		GSM8K-IID	
	3B	7B	3B	7B	3B	7B
9	1.87	1.67	0.93	0.94	1.26	1.14
109	1.68	1.59	0.91	0.93	1.10	1.09
209	1.52	1.55	0.87	0.91	1.04	1.05
309	1.40	1.51	0.87	0.90	0.94	1.01