Investigating Social Bias Propagation in Federated Fine-tuning of Large Language Models

Jiaxu Zhao[1, 2], **Meng Fang**[3,1], **Mingze Zhong**[4], **Shunfeng Zheng**[4], **Ling Chen**[4], **Mykola Pechenizkiy**[1]
[1]Eindhoven University of Technology, Eindhoven, the Netherlands
[2]EPFL, Lausanne, Switzerland
[3]University of Liverpool, Liverpool, the United Kingdom
[4]University of Technology Sydney, NSW, Australia
{j.zhao, m.pechenizkiy}@tue.nl, meng.fang@liverpool.ac.uk, {mingze.zhong, shunfeng.zheng, ling.chen}@uts.edu.au

Abstract

Large language models (LLMs) have achieved remarkable success in many domains, but concerns about data quality and privacy are growing. Federated Learning (FL) offers a privacy-preserving solution by training a model on local clients without sharing data. However, the impact of biased private data on LLMs fine-tuned through FL remains understudied. This work investigates how client-side biased data affects the global model during federated fine-tuning of LLMs. We simulate realistic scenarios where some clients possess datasets containing social biases (stereotypes, discriminatory language) while others have clean data through extensive experiments with popular FL algorithms (FedAvg, FedAdam and FedProx) and popular LLMs (LLaMA, Mistral, Phi-3 and Gemma) across datasets with varying bias proportions (33%, 66%, 100%). Our findings reveal that 1) FedAdam consistently shows the lowest bias propagation, reducing CrowS-Pairs scores by up to 15% compared to FedAvg; 2) Even small amounts of biased data (33%) can significantly influence global model bias; 3) Mixed biased and neutral data distributions lead to 5-7% higher bias scores than segregated distributions. Additionally, we propose Bias-Aware Model Aggregation (BAMA), a novel debiasing method for federated fine-tuning that consistently reduces bias across various models and algorithms.

Introduction

Large language models (LLMs) (Brown et al. 2020; Conneau et al. 2019; Devlin et al. 2018; Radford et al. 2018; Raffel et al. 2020; OpenAI 2022, 2023; Touvron et al. 2023; Almazrouei et al. 2023; Chiang et al. 2023; Jiang et al. 2023; Biderman et al. 2023) have been adapted to a wide range of fields because of their remarkable ability to generate human-like text. But their success relies heavily on massive public datasets and computational resources (Villalobos et al. 2022). However, the increasing deployment of LLMs raises critical concerns about data privacy and bias propagation, particularly in sensitive domains like healthcare (Thirunavukarasu et al. 2023), legal services, and financial applications (Wu et al. 2023).

Federated Learning (FL) (McMahan et al. 2017; Guan et al. 2024; Reddi et al. 2020; Li et al. 2020) offers a promising solution to these challenges by enabling privacy-preserving collaborative training of models across multiple clients without sharing raw data (Kairouz et al. 2021). This approach allows companies with different computational resources to collaborate in training powerful machine learning models, thus reducing the counting burden of training large models, and it enables the utilization of high-quality private data while preserving privacy through local training.

Despite the success of LLMs, they may inherit stereotypes, misrepresentation, and other disparaging behavior toward some communities because they are classically trained on large amounts of data. These harms are forms of "social bias," [1] a term broadly referring to disparate treatment or outcomes between social groups (Gallegos et al. 2024).

This work investigates how biased private data impacts the global model during federated fine-tuning of LLMs. We employ a data poisoning approach to simulate scenarios where clients in a federated learning setup may have biased datasets. Data poisoning (Steinhardt, Koh, and Liang 2017; Tolpegin et al. 2020) is an attack on machine learning models wherein it introduces biased or malicious data into the training data to manipulate the behavior of machine learning systems. Through extensive experiments with popular FL algorithms (Fedavg (McMahan et al. 2017), FedAdam (Reddi et al. 2020) and FedProx (Li et al. 2020)) and large language models (LLaMA (Dubey et al. 2024), Mistral (Jiang et al. 2023), Phi-3 (Abdin et al. 2024), and Gemma (Team et al. 2024)), we examine how the use of biased data by certain clients during local training influences the global model's bias. Our study addresses the following research questions:

- **Q1**: How does the choice of federated learning algorithm affect bias propagation?
- **Q2**: What is the impact of different fine-tuning methods on bias in a federated setting?
- **Q3**: How does the ratio of biased to neutral data across clients influence the global model's bias?

Through comprehensive experimentation and analysis, our work provides novel insights for developing fairer and more robust federated learning systems for LLMs,

[1]Unless otherwise specified, our use of "bias" refers to social bias. Social bias encompasses the disparate treatment or outcomes between social groups, stemming from historical and structural power asymmetries.

with important implications for deploying these models in privacy-sensitive domains while mitigating harmful biases. Our code is available at: https://github.com/Jiaxu-Zhao/federated-bias-llm.

Related Work

Large Language Models have revolutionized natural language processing in recent years. A number of pre-trained language models such as GPT-3 (Brown et al. 2020), BERT (Devlin et al. 2018), and T5 (Raffel et al. 2020) have demonstrated remarkable capabilities in various NLP tasks. ChatGPT (OpenAI 2022) and GPT-4 (OpenAI 2023) further improve the alignment of the response of the language model and human. In the open-source domain, models like LLaMA (Touvron et al. 2023), Pythia (Biderman et al. 2023), Vicuna (Chiang et al. 2023), and Mistral (Jiang et al. 2023) have democratized access to high-performance LLMs, enabling broader research and applications.

However, the success of these models relies heavily on massive amounts of training data. Federated Learning, introduced by McMahan et al. (2017), offers a privacy-preserving approach to collaborative machine learning. It allows multiple clients to train a shared model without sharing their data, addressing privacy concerns in sensitive domains. FedAvg is the original federated averaging algorithm. Li et al. (2020) propose FedProx that adds a proximal term to the local objective, improving stability and convergence. FedAdam (Reddi et al. 2020) uses an adaptive optimization method for federated learning. These algorithms address various challenges in federated learning, such as communication efficiency, convergence speed, and heterogeneity of client data. Recent efforts have benchmarked federated learning for NLP and LLMs. FedNLP (Lin et al. 2022) provides a unified framework to compare FL methods across NLP tasks, highlighting non-IID and communication challenges. FedLLM-Bench (Ye et al. 2024) extends this to large language models with realistic multilingual and preference datasets. Our work complements these by focusing on social bias propagation and proposing Bias-Aware Model Aggregation (BAMA) for bias mitigation.

As LLMs have grown in capability and influence, concerns about social bias in these models have become prominent. Research has shown that LLMs can inherit and amplify societal biases present in their training data, leading to issues of stereotyping, misrepresentation, and unfair treatment of certain groups (Zhao et al. 2023b; Nadeem, Bethke, and Reddy 2020; Zhao et al. 2023a, 2025b,a). Several studies have focused on measuring and mitigating bias in LLMs. Guo, Rush, and Kim (2020) and Dhingra et al. (2023) proposed automated methods for detecting bias, while Zhao et al. (2018) developed metrics for measuring gender bias in sentence embeddings. Evaluation datasets and benchmarks such as Crows-Pairs (Nangia et al. 2020) and StereoSet (Nadeem, Bethke, and Reddy 2020) have been created to assess various types of social biases in language models.

While substantial research has been conducted on bias in centrally trained LLMs, there is a notable gap in understanding how bias propagates in federated learning scenarios. This gap is particularly significant given the unique characteristics of federated learning, where data cannot be directly balanced or debiased across clients, and bias may be amplified through model aggregation. Furthermore, the privacy-preserving nature of federated learning introduces additional challenges in monitoring and mitigating bias while maintaining privacy guarantees. This work aims to address this gap by investigating the impact of biased private data on global model bias during federated fine-tuning of LLMs.

Methodology

Federated Learning

Typically process of federated learning involves a central server that coordinates multiple clients, each training on local data. Clients upload model updates to the server, which aggregates them to create a global model that is then broadcast back to the clients.

The process incorporates the Federated Averaging (FedAvg) algorithm proposed by McMahan et al. (2017), which serves as the foundation for most popular federated learning algorithms. In this setup, servers initiate and coordinate the entire training process until a predefined stopping criterion is met. A key feature of federated learning is that client data remains private, with training occurring exclusively on the client side. The typical workflow of federated learning proceeds as follows (Guan et al. 2024): (1) Client Selection and Initialization: The server selects clients meeting specific criteria (e.g., network access, bandwidth) and initializes a global model. (2) Local Training: The global model is distributed to participating clients, who train it on their local data. (3) Model Upload: Clients calculate and upload model updates (e.g., gradients or parameter changes) to the central server. (4) Aggregation: The server combines client updates to improve the global model. (5) Broadcast: The updated global model is sent back to clients, and clients can train the model locally. (6) Iteration and Convergence: Steps 2-5 repeat until the model achieves satisfactory performance or a predefined number of iterations is completed. (7) Deployment: The final global model is implemented in real-world applications.

We employ the following three federated learning algorithms because they cover the key challenges in federated learning while maintaining computational feasibility: FedAvg establishes a robust baseline, FedAdam addresses adaptive optimization and communication efficiency, and FedProx ensures stability in heterogeneous environments.

FederatedAveraging (FedAvg)(McMahan et al. 2017) FedAvg is a fundamental federated learning algorithm where clients train their model locally, and then the model weights (e.g., the weight of logistic regression) from all clients are aggregated to calculate a global model.

FedAdam (Reddi et al. 2020) FedAdam incorporates federated versions of adaptive optimizer ADAM, focusing on the interplay between client heterogeneity and communication efficiency: (1) clients perform multiple epochs of training using a client optimizer to minimize loss on their local data and (2) server updates its global model by applying a gradient-based server optimizer to the average of the clients'

model updates.

FedProx (Li et al. 2020) FedProx introduces an additional proximal term to the local optimization objective. Every client trains its own model with the proximal term (the coefficient μ is set to 0.1 (Guan et al. 2024)). Local training is conducted only once. The model weights of each client are aggregated to get a global model.

Dataset Construction via Data Poisoning

To systematically investigate bias propagation in federated LLM fine-tuning, we construct controlled datasets that simulate real-world scenarios where clients may possess biased data. Our approach uses intentional bias injection (often called "data poisoning" in adversarial contexts, though here used for research purposes) to create realistic federated learning environments. Our dataset is in English and built from two dataset sources:

- **Clean Data**: Alpaca dataset (Peng et al. 2023) (52K instruction-following examples generated by GPT-4 (OpenAI 2023)) representing neutral, unbiased training samples.
- **Biased Data**: Rejected completions from the BiasDPO dataset (Allam 2024), containing stereotypical responses about gender, race, religion, and other protected attributes.

We design two primary distribution patterns to reflect different real-world federated scenarios: a) Segregated Distribution (Different clients have either fully biased or fully clean data.): **Fully poisoned (3B)**: Clients have only biased instructions (simulates organizations with problematic data practices). **Clean data (3C)**: Clients have only neutral instructions (baseline scenario). **Light poisoning (1B/2C)**: One client has biased data, two clients have clean data (33% bias prevalence). **Heavy poisoning (2B/1C)**: Two clients have biased data, one client has clean data (66% bias prevalence). b) Mixed Distribution: **Mixed light poisoning (Mix 1B/2C)**: Each client contains 33% biased + 67% clean instructions. **Mixed heavy poisoning (Mix 2B/1C)**: Each client contains 67% biased + 33% clean instructions.

Bias metrics

We use two common bias evaluation metrics (Crows-Pairs (Nangia et al. 2020) and StereoSet (Nadeem, Bethke, and Reddy 2020)) to assess the bias.

Crows-Pairs can measure nine bias types: race, gender, sexual orientation, religion, age, nationality, disability, physical appearance, and socioeconomic status. It calculates the proportion of instances where the language model shows a preference for the stereotypical sentence over the anti-stereotypical one, which is reflected in the bias score. An ideal score of 50% indicates that the model does not exhibit any bias towards the categories present in the dataset.

StereoSet can assess four types of bias – gender, race, profession, and religion. StereoSet is a fill-in-the-blank task to measure the model's bias. Pending sentences are descriptions related to a specific group of people, and the words to be selected include stereotypes, anti-stereotypes, and an unrelated word. This metric works by calculating the probability of two stereotyped, anti-stereotyped words, so the model is ideal when the result is 50%.

Debiasing via Bias-Aware Model Aggregation

To mitigate bias propagation during federated training, we propose Bias-Aware Model Aggregation (BAMA). BAMA adaptively reweights client contributions based on their measured bias levels, reducing the influence of highly biased clients without accessing their raw data.

Step 1: Bias Assessment

For each client i, the server maintains a bias score b_i computed by both CrowS-Pairs and StereoSet metrics on 25% data of CrowS-Pairs and the validation set of StereoSet. The bias score is defined as:

$$b_i = \alpha\,(|CrowS - Pairs(M_i) - 50| + |StereoSet(M_i) - 50|) \quad (1)$$

where M_i represents client i's model update, CrowS-Pairs(Mi) and StereoSet(Mi) are the bias scores ranging from 0 to 100 ($50 =$ unbiased), $\alpha \in [0, 1]$ is a hyperparameter. Higher b_i values indicate greater bias in either direction.

Step 2: Adaptive Weighting

Client weights are computed using exponential decay based on bias scores:

$$w_{i,t} = exp(-\lambda b_i) / \sum_{j}^{C} exp(-\lambda b_j) \quad (2)$$

where $w_{i,t}$ is the bias-aware weight for client i at round t, $\lambda > 0$ controls debiasing strength (higher $\lambda =$ stronger bias penalization). C denotes the number of participating clients.

Step 3: Integration with FL Algorithms

In FedAvg, BAMA modifies the standard averaging procedure:

$$\theta_{t+1} = \theta_t + \sum_i w_{i,t}(\frac{n_i}{n})(\theta_{i,t} - \theta_t) \quad (3)$$

where θ_t represents the global model parameters at round t, $\theta_{i,t}$ is client i's local model parameters, and $\frac{n_i}{n}$ is the ratio of client i's dataset size to the total dataset size. This combines traditional size-based weighting with our bias-aware weights.

For FedAdam, BAMA influences both moment estimates:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \sum_i w_{i,t} \Delta_{i,t} \quad (4)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\sum_i w_{i,t} \Delta_{i,t})^2 \quad (5)$$

where m_t and v_t are the first and second moment estimates at round t, β_1 and β_2 are moment decay rates in $[0, 1]$, and $\Delta_{i,t} = \theta_{i,t} - \theta_t$ is client i's model update.

In FedProx, BAMA modifies both the server aggregation and the proximal term:

$$L_i(\theta) = F_i(\theta) + \frac{\mu w_{i,t}}{2} |\theta - \theta_t|^2 \quad (6)$$

where $L_i(\theta)$ is client i's modified loss function, $F_i(\theta)$ is the original task loss, $\mu > 0$ is the proximal term coefficient, and $|\theta - \theta_t|^2$ is the L2 norm difference between local and global models.

Experiments

Models

In our experiments, we utilize four popular LLMs:

LLaMA 3 (Dubey et al. 2024): The popular public Large Language Model released by Meta. We use version 8B in our experiments.

Mistral 7B (Jiang et al. 2023): A pretrained language model with 7 billion parameters engineered for superior performance and efficiency.

Phi 3 (Abdin et al. 2024): Phi 3 is a popular family of open language models developed by Microsoft. In our experiments, we use Phi-3-Mini-4K-Instruct, which is a 3.8B parameters model.

Gemma 2 (Team et al. 2024): Gemma is a family of lightweight open models from Google. We use a 2B and 9B version in our experiments.

Baselines

To comprehensively evaluate the impact of federated fine-tuning on model bias, we establish the following baselines:

Original Models We evaluate the original, pre-trained versions of LLMs to establish a baseline for inherent bias before any fine-tuning.

Full Parameters Fine-tuning We perform traditional fine-tuning on the complete parameter set of each model using our constructed dataset.

LoRA Fine-tuning We implement Low-Rank Adaptation (LoRA) fine-tuning (Hu et al. 2021), a parameter-efficient technique that only updates a small set of model parameters.

Experimental setup

Hyperparameters For the federated fine-tuning experiments, the default settings are as follows: the number of clients $= 3$, the client fraction per round $= 1.0$, the number of local epochs $= 2$, and the number of server rounds $= 5$. This setup ensures that all clients participate in every round of model updates. This approach offers two key advantages: (1) it maximizes data utilization by incorporating all available data in each training round, and (2) it enables observation of each client's contribution to the global model. Additionally, we conducted a comparison experiment in which only a subset of clients was selected to participate in each round, allowing us to analyze the differences in global model updates between partial and full client participation (Section). For full parameters and LoRA fine-tuning, we use the default parameters in Hugging Face [2].

Computing Infrastructure Our experiments were conducted using 2 NVIDIA A40 GPUs with 48GB of memory each, 256GB of RAM, and 32 CPU cores.

Software and Frameworks We used PyTorch 1.9 and the Hugging Face Transformers library for model implementations. For LoRA Fine-tuning, we use Llama-Factory [3]. For federated learning simulations, we utilized the Flower (flwr) framework (Beutel et al. 2020).

[2]https://huggingface.co/

[3]https://github.com/hiyouga/LLaMA-Factory

Results and Analysis

Each experiment was run three times with different seeds, and the results presented are the average scores. We report the results, using the average difference (Δ**Avg**) between the bias scores and the ideal score of 50 for both Crows-Pairs and StereoSet metrics.

Impact of Federated Learning Algorithms (Q1)

Our experiments reveal significant differences in how federated learning algorithms handle bias propagation (Table 1 shows average bias scores.)

FedAdam showed the lowest bias across most models and datasets. For example, on Phi-3 with clean data (3C), FedAdam scored 13.12 (CrowS-Pairs) vs. FedAvg's 16.67. We think FedAdam's adaptive optimization allows more nuanced integration of client updates, potentially reducing extreme biases through its momentum-based aggregation.

FedProx often exhibited the highest bias scores, particularly for mixed datasets. For instance, on LLaMA-3 with Mix 1B/2C, FedProx scored 21.36 vs. FedAdam's 20.60. FedProx's regularization term may inadvertently preserve local biases by constraining updates to stay close to the global model, preventing effective bias correction

The performance of FedAvg was generally between FedAdam and FedProx in most cases.

According to the observations, for bias-sensitive applications, FedAdam is fairer than other algorithms. FedProx should be used cautiously when bias mitigation is important. Algorithm choice can reduce bias by up to 3%-4% points without any other interventions.

Fine-tuning Methods Comparison (Q2)

We report full parameter fine-tuning (FT), LoRA fine-tuning, and federated fine-tuning (FedAdam) in Table 2.

Full parameter fine-tuning often resulted in the highest bias scores, especially on biased datasets. For example, with LLaMA 3 on the 3B dataset, FT showed bias scores of 21.22 for Crows-Pairs and 19.65 for StereoSet, compared to LoRA (20.84 and 18.77) and FedAdam (19.54 and 17.96). LoRA fine-tuning generally produced bias scores between full fine-tuning and federated fine-tuning. FedAdam consistently demonstrated the lowest bias scores across most models and datasets.

Our findings reveal that the choice of fine-tuning method substantially impacts the propagation of bias. Specifically, full parameter fine-tuning demonstrates a higher capability for amplifying biases present in the training data. In contrast, parameter-efficient methods like LoRA and federated learning approaches exhibit varying degrees of bias propagation. This observation underscores the importance of carefully selecting fine-tuning techniques to minimize bias in language models, particularly in federated learning scenarios where privacy concerns intersect.

Impact of Biased Data Ratio (Q3)

We examined different ratios of biased to neutral data (3B, 3C, 1B/2C, 2B/1C) and their impact on model bias. As expected, the 3C (clean) dataset consistently resulted in the

Model	FL	3B	3C	1B/2C	2B/1C
LLaMA 3	Original		19.72 / 17.79		
	FedAvg	20.29 / 18.55	18.24 / 17.62	**21.06 / 19.14**	19.83 / 18.05
	FedAdam	19.54 / 17.96	18.28 / 16.79	**20.86** / 17.67	19.74 / 17.82
	FedProx	19.62 / 18.30	19.64 / 17.71	**21.26 / 19.41**	21.25 / 18.51
Mistral	Original		18.91 / 14.94		
	FedAvg	18.87 / 15.46	17.75 / 14.98	**19.79 / 17.34**	18.89 / 16.12
	FedAdam	18.33 / **15.45**	15.47 / 13.69	**18.95** / 15.42	18.05 / 15.25
	FedProx	20.11 / 15.41	18.25 / 15.04	**20.89** / 16.84	20.18 / **17.37**
Phi 3	Original		17.04 / 12.29		
	FedAvg	17.51 / 13.14	16.67 / 12.43	**18.82 / 13.68**	18.19 / 12.81
	FedAdam	17.44 / 12.62	13.12 / 11.13	**18.03** / 12.63	16.67 / **13.00**
	FedProx	17.81 / 13.28	16.67 / 12.41	**18.72 / 14.05**	18.03 / 13.27
Gemma 2 2B	Original		15.46 / 17.66		
	FedAvg	17.31 / 19.36	14.35 / 16.98	**17.52 / 20.23**	16.89 / 19.16
	FedAdam	**16.16 / 17.71**	12.07 / 15.23	15.84 / 16.13	14.56 / 16.75
	FedProx	**17.56** / 20.04	16.32 / 15.77	17.44 / **21.65**	17.53 / 20.70

Table 1: Bias results (ΔAvg) across different federated learning methods. Each cell shows Crows-Pairs / StereoSet. Bold indicates the highest bias within each FL method.

lowest bias scores across all models and fine-tuning methods. For instance, in Table 1, with Gemma 2 2B using FedAdam, the 3C dataset showed bias scores of 12.07 for Crows-Pairs and 15.23 for StereoSet, compared to 3B (16.16 and 17.71), 1B/2C (15.84 and 16.13), and 2B/1C (14.56 and 16.75). But interestingly, in most cases, the 1B/2C or 2B/1C (partially biased) dataset produces the highest bias scores. We propose this hypothesis for this phenomenon: When biased and neutral data are mixed, the model may struggle to reconcile conflicting signals during training. The biased data introduces stereotypes or harmful patterns, while the neutral data does not strongly mitigate these biases, leading to higher overall bias propagation.

The results show that the ratio of biased to neutral data plays a crucial role in determining the final model's bias, with a small proportion of biased data capable of significantly influencing the model's behavior.

Comparison Between Segregated And Mixed Distributions

Our finding reveals that mixed data distributions consistently produce higher bias than segregated distributions (Figure 1). Across all models and FL algorithms, mixed distributions (where each client has both biased and neutral data) show 5%-7% higher bias scores than segregated distributions (where clients have either fully biased or fully clean data).

This phenomenon can be explained by how federated aggregation processes different signal types. In segregated scenarios, biased clients send clearly problematic updates while clean clients send neutral updates. This clear separation enables aggregation algorithms to potentially identify and balance these extreme positions. In mixed scenarios, each client produces biased updates because their data contains both biased and neutral examples. These subtle biases are harder for aggregation algorithms to detect and correct, resulting in higher overall bias in the global model. These findings suggest that organizations should maintain data quality standards rather than assuming that "mixing" biased and neutral data will naturally balance out. Mixed bias scenarios make it more difficult to identify problematic participants.

Impact of Client Participation Ratio

We examine how client participation rates affect model bias by comparing three scenarios on the Phi 3 model: high participation (100% - where all 3 clients participate), medium participation (60% - where 3/5 clients participate), and low participation (30% - where 3/10 clients participate). In each scenario, we maintain 3 clients with biased data while varying the total number of clients to achieve the desired participation percentages.

Lower participation ratios generally resulted in slightly reduced bias scores. For instance, with FedAvg on the 3B dataset, 30% participation showed bias scores of 16.71 (Crows-Pairs) and 12.12 (StereoSet), compared to 100% participation with 17.51 and 13.14, respectively. This trend was consistent across all federated learning algorithms, although the differences were relatively small. This suggests that partial client participation may help in reducing bias propagation, possibly by introducing more randomness and diversity in the training process.

Impact of BAMA on Bias Mitigation

We evaluated BAMA's effectiveness across different FL algorithms. Since the 1B/2C dataset showed the highest bias scores in most cases, we used this configuration for our debiasing experiments. Table 3 exhibits the average bias scores of Crows-Pairs and StereoSet.

BAMA achieves consistent bias reduction across all evaluated models and FL methods. For LLaMA 3 with FedAvg, BAMA reduces CrowS-Pairs scores from 21.06 to 20.01

Model	Method	3B	3C	Mix 1B/2C	Mix 2B/1C
LLaMA 3	Original		19.72 / 17.79		
	FT	**21.22/19.65**	13.69/16.86	18.99/19.42	16.65/19.57
	LoRA	20.84/**18.77**	17.87/16.84	21.34/18.03	**21.86**/18.27
	FedAdam	19.54/**17.96**	18.28/16.79	**20.60**/17.61	19.18/16.62
Mistral	Original		18.91 / 14.94		
	FT	19.35/16.68	18.09/13.77	19.13/**18.25**	**20.75**/15.96
	LoRA	19.58/**15.61**	19.11/14.44	**20.44**/15.10	20.32/15.06
	FedAdam	**18.33/15.45**	15.47/13.69	17.80/15.08	17.62/14.64
Phi 3	Original		17.04 / 12.29		
	FT	18.55/12.61	14.01/11.64	**19.21/13.74**	18.34/12.10
	LoRA	17.13/12.29	16.27/11.67	17.23/12.16	**18.41/12.43**
	FedAdam	**17.44/12.62**	13.12/11.13	16.52/11.59	15.11/12.09
Gemma 2 2B	Original		15.46 / 17.66		
	FT	17.55/19.35	13.04/15.38	17.85/**20.20**	**18.15**/17.75
	LoRA	16.63/19.20	14.62/16.27	17.05/**19.62**	**17.88**/19.36
	FedAdam	**16.16/17.71**	12.07/15.23	15.34/15.70	14.03/16.15

Table 2: Bias results (ΔAvg) across different fine-tuning methods. Each cell shows Crows-Pairs / StereoSet. Bold numbers indicate the highest bias within each fine-tuning method.

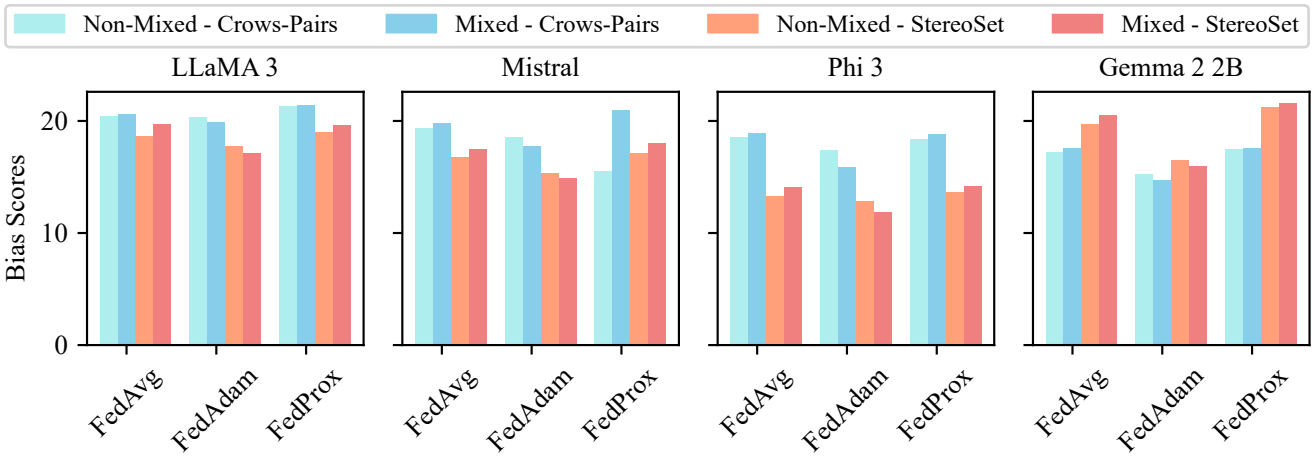

Figure 1: Bias evaluation results (Δ**Avg**) comparing segregated distribution (where clients have either fully biased or clean data) versus mixed distribution (where each client has an internal mixture of biased and clean data) across different federated learning methods.

and StereoSet scores from 19.14 to 17.96. The effectiveness varies by FL algorithm, with FedAdam showing the strongest improvements when combined with BAMA. For instance, in Phi 3, FedAdam with BAMA reduces CrowS-Pairs scores from 18.03 to 16.63 and StereoSet scores from 12.63 to 10.42.

The debiasing effect varies across different bias categories. Race-related bias sees the most substantial improvements, particularly in FedAdam configurations. Religious and economic bias categories show moderate improvements of 1.7 and 1.4 percentage points respectively.

Model size influences BAMA's effectiveness. The smaller Gemma 2 2B achieves the largest bias reductions with CrowS-Pairs improving by 2.38 points and StereoSet by 1.41 points. In contrast, the larger LLaMA 3 8B shows more modest improvements with CrowS-Pairs reducing by 1.05 points and StereoSet by 1.18 points. This suggests that smaller models may be more responsive to bias mitigation, possibly due to less entrenched biases from pre-training.

These results establish BAMA as an effective approach for mitigating bias in federated LLM training. The method demonstrates particular strength when combined with adaptive optimization methods like FedAdam and shows enhanced effectiveness on smaller language models.

Limitations

While our study provides valuable insights into bias propagation in the federated fine-tuning of LLMs, we acknowledge several key limitations.

Model	FL	Debias	Crows-Pairs	StereoSet
LLaMA 3	FedAvg	w/o	21.06	19.14
		w/	**20.01**	**17.96**
	FedAdam	w/o	20.86	17.67
		w/	**20.06**	**16.01**
	FedProx	w/o	21.26	19.41
		w/	**20.15**	**17.91**
Mistral	FedAvg	w/o	19.79	17.34
		w/	**18.72**	**16.18**
	FedAdam	w/o	18.95	15.42
		w/	**17.19**	**14.01**
	FedProx	w/o	20.89	16.84
		w/	**18.26**	**15.12**
Phi 3	FedAvg	w/o	18.82	13.68
		w/	**16.32**	**12.05**
	FedAdam	w/o	18.03	12.63
		w/	**16.63**	**10.42**
	FedProx	w/o	18.72	14.05
		w/	**17.09**	**12.49**
Gemma 2 2B	FedAvg	w/o	17.52	20.23
		w/	**15.14**	**18.82**
	FedAdam	w/o	15.84	16.13
		w/	**14.47**	**14.88**
	FedProx	w/o	17.44	21.65
		w/	**16.10**	**20.31**

Table 3: BAMA results (ΔAvg) across different FL methods. "w/" and "w/o" indicate with BAMA debiasing and without it. Bold indicates the lower bias values.

Our constructed dataset, while designed to capture various biases, may not fully represent the complexity of real-world data distributions. The dataset's focus on English language content limits our understanding of bias propagation across different linguistic and cultural contexts. Furthermore, the BAMA method relies on CrowS-Pairs and StereoSet, which may only capture a subset of bias types. We chose these for their wide use and bias coverage but acknowledge their limits.

Our proposed BAMA debiasing method, while effective, currently relies on a fixed bias scoring mechanism using CrowS-Pairs and StereoSet metrics. The approach could benefit from more dynamic bias detection methods and adaptive reweighting strategies.

Our investigation was constrained by available computational resources, limiting experiments to models under 10B parameters. Extending this study to larger models could reveal different bias propagation patterns and mitigation effectiveness. The focus on specific FL algorithms, while covering key approaches, may not capture all potential aggregation strategies that could influence bias propagation.

Conclusion

This study provides comprehensive insights into bias propagation during federated fine-tuning of large language models. We demonstrate that algorithm choice fundamentally affects bias outcomes, with FedAdam achieving up to 15% lower bias scores than FedAvg, while FedProx exhibits the highest bias under mixed data distributions. A crucial finding is that even small amounts of biased data (33%) can significantly influence the global model. Mixed distributions where biased and neutral data coexist within clients produce higher bias than segregated distributions, suggesting that internal data mixing amplifies rather than dilutes bias effects. To address these challenges, we propose BAMA, a bias-aware aggregation method that adaptively weights client contributions. BAMA demonstrates consistent effectiveness across all configurations, particularly when combined with FedAdam and on smaller models, achieving meaningful bias reductions while preserving privacy guarantees. These findings provide actionable guidance for practitioners deploying LLMs in federated environments, emphasizing the importance of careful algorithm selection and data distribution strategies to balance privacy with fairness objectives.

Acknowledgments

Jiaxu Zhao and Mykola Pechenizkiy thank the support of EU EDF KOIOS and Dutch NWO EDIC projects. This work used the Dutch national e-infrastructure with the support of the SURF Cooperative, using grant no. EINF-3953/L1.

References

Abdin, M.; Jacobs, S. A.; Awan, A. A.; Aneja, J.; Awadallah, A.; Awadalla, H.; Bach, N.; Bahree, A.; Bakhtiari, A.; Behl, H.; et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Allam, A. 2024. BiasDPO: Mitigating Bias in Language Models through Direct Preference Optimization. In Fu, X.; and Fleisig, E., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, 71–79. Bangkok, Thailand: Association for Computational Linguistics.

Almazrouei, E.; Alobeidli, H.; Alshamsi, A.; Cappelli, A.; Cojocaru, R.; Debbah, M.; Goffinet, E.; Heslow, D.; Launay, J.; Malartic, Q.; Noune, B.; Pannier, B.; and Penedo, G. 2023. Falcon-40B: an open large language model with state-of-the-art performance.

Beutel, D. J.; Topal, T.; Mathur, A.; Qiu, X.; Fernandez-Marques, J.; Gao, Y.; Sani, L.; Kwing, H. L.; Parcollet, T.; Gusmão, P. P. d.; and Lane, N. D. 2020. Flower: A Friendly Federated Learning Research Framework. *arXiv preprint arXiv:2007.14390*.

Biderman, S.; Schoelkopf, H.; Anthony, Q. G.; Bradley, H.; O'Brien, K.; Hallahan, E.; Khan, M. A.; Purohit, S.; Prashanth, U. S.; Raff, E.; et al. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, 2397–2430. PMLR.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dhingra, H.; Jayashanker, P.; Moghe, S.; and Strubell, E. 2023. Queer people are people first: Deconstructing sexual identity stereotypes in large language models. *arXiv preprint arXiv:2307.00101*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Gallegos, I. O.; Rossi, R. A.; Barrow, J.; Tanjim, M. M.; Kim, S.; Dernoncourt, F.; Yu, T.; Zhang, R.; and Ahmed, N. K. 2024. Bias and fairness in large language models: A survey. *Computational Linguistics*, 1–79.

Guan, H.; Yap, P.-T.; Bozoki, A.; and Liu, M. 2024. Federated learning for medical image analysis: A survey. *Pattern Recognition*, 110424.

Guo, D.; Rush, A. M.; and Kim, Y. 2020. Parameter-efficient transfer learning with diff pruning. *arXiv preprint arXiv:2012.07463*.

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

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2): 1–210.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2: 429–450.

Lin, B. Y.; He, C.; Ze, Z.; Wang, H.; Hua, Y.; Dupuy, C.; Gupta, R.; Soltanolkotabi, M.; Ren, X.; and Avestimehr, S. 2022. FedNLP: Benchmarking Federated Learning Methods for Natural Language Processing Tasks. In Carpuat, M.; de Marneffe, M.-C.; and Meza Ruiz, I. V., eds., *Findings of the Association for Computational Linguistics: NAACL 2022*, 157–175. Seattle, United States: Association for Computational Linguistics.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.

Nadeem, M.; Bethke, A.; and Reddy, S. 2020. Stereoset: Measuring stereotypical bias in pretrained language models. *arXiv preprint arXiv:2004.09456*.

Nangia, N.; Vania, C.; Bhalerao, R.; and Bowman, S. R. 2020. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. *arXiv preprint arXiv:2010.00133*.

OpenAI. 2022. ChatGPT. arXiv:https://openai.com/blog/chatgpt/.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Peng, B.; Li, C.; He, P.; Galley, M.; and Gao, J. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving language understanding by generative pre-training.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1): 5485–5551.

Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečnỳ, J.; Kumar, S.; and McMahan, H. B. 2020. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*.

Steinhardt, J.; Koh, P. W. W.; and Liang, P. S. 2017. Certified defenses for data poisoning attacks. *Advances in neural information processing systems*, 30.

Team, G.; Riviere, M.; Pathak, S.; Sessa, P. G.; Hardin, C.; Bhupatiraju, S.; Hussenot, L.; Mesnard, T.; Shahriari, B.; Ramé, A.; et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Thirunavukarasu, A. J.; Ting, D. S. J.; Elangovan, K.; Gutierrez, L.; Tan, T. F.; and Ting, D. S. W. 2023. Large language models in medicine. *Nature medicine*, 29(8): 1930–1940.

Tolpegin, V.; Truex, S.; Gursoy, M. E.; and Liu, L. 2020. Data poisoning attacks against federated learning systems. In *Computer security–ESORICs 2020: 25th European symposium on research in computer security, ESORICs 2020, guildford, UK, September 14–18, 2020, proceedings, part i 25*, 480–501. Springer.

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

Villalobos, P.; Sevilla, J.; Heim, L.; Besiroglu, T.; Hobbhahn, M.; and Ho, A. 2022. Will we run out of data? an

analysis of the limits of scaling datasets in machine learning. *arXiv preprint arXiv:2211.04325*.

Wu, S.; Irsoy, O.; Lu, S.; Dabravolski, V.; Dredze, M.; Gehrmann, S.; Kambadur, P.; Rosenberg, D.; and Mann, G. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Ye, R.; Ge, R.; Zhu, X.; Chai, J.; Yaxin, D.; Liu, Y.; Wang, Y.; and Chen, S. 2024. Fedllm-bench: Realistic benchmarks for federated learning of large language models. *Advances in Neural Information Processing Systems*, 37: 111106–111130.

Zhao, J.; Fang, M.; Pan, S.; Yin, W.; and Pechenizkiy, M. 2023a. Gptbias: A comprehensive framework for evaluating bias in large language models. *arXiv preprint arXiv:2312.06315*.

Zhao, J.; Fang, M.; Shi, Z.; Li, Y.; Chen, L.; and Pechenizkiy, M. 2023b. CHBias: Bias Evaluation and Mitigation of Chinese Conversational Language Models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 13538–13556.

Zhao, J.; Fang, M.; Ye, F.; Xu, K.; Zhang, Q.; Zhou, J. T.; and Pechenizkiy, M. 2025a. Understanding Large Language Model Vulnerabilities to Social Bias Attacks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 17620–17636.

Zhao, J.; Fang, M.; Zhang, K.; and Pechenizkiy, M. 2025b. Unmasking Style Sensitivity: A Causal Analysis of Bias Evaluation Instability in Large Language Models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 16314–16338.

Zhao, J.; Wang, T.; Yatskar, M.; Ordonez, V.; and Chang, K.-W. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. *arXiv preprint arXiv:1804.06876*.