On Fairness of Task Arithmetic: The Role of Task Vectors

Anonymous ACL submission

Abstract

Model editing techniques, particularly task arithmetic using task vectors, have shown promise in efficiently modifying pre-trained models through arithmetic operations like task addition and negation. Despite computational advantages, these methods may inadvertently affect model fairness, creating risks in sensitive applications like hate speech detection. However, the fairness implications of task arithmetic remain largely unexplored, presenting a critical gap in the existing literature. We systematically examine how manipulating task vectors affects fairness metrics, including Demographic Parity and Equalized Odds. To rigorously assess these effects, we benchmark task arithmetic against full fine-tuning, a costly but widely used baseline, and Low-Rank Adaptation (LoRA), a prevalent parameter-efficient fine-tuning method. Additionally, we explore merging task vectors from models fine-tuned on demographic subgroups vulnerable to hate speech, investigating whether fairness outcomes can be controlled by adjusting task vector coefficients, potentially enabling tailored model behavior. Our results offer novel insights into the fairness implications of model editing and establish a foundation for fairness-aware and responsible model editing practices.

1 Introduction

As large language models (LLMs) see broader application, efficient techniques for adapting them to specific tasks become increasingly crucial. Although there are models that have been distilled (Sanh et al., 2019; Jiao et al., 2020; Turc et al., 2020) or are relatively small in size (Abdin et al., 2024), task-specific fine-tuning often requires substantial computational resources, prompting the development of parameter-efficient fine-tuning (PEFT) techniques (Houlsby et al., 2019; Hu et al., 2022; Ben Zaken et al., 2022; Dettmers et al., 2023).

One notable example is Low-Rank Adaptation (LoRA) (Hu et al., 2022), which updates a compact set of parameters while leaving most of the original weights untouched, thus reducing training costs. Despite the popularity of PEFT methods, they do not resolve every challenge: in high-stakes tasks with imbalanced data, LoRA and similar approaches can preserve or even amplify biases, raising concerns about fairness (Ding et al., 2024b; Sap et al., 2019).

An alternative strategy that has recently drawn attention is model editing with task vectors (Ilharco et al., 2023; Zhang et al., 2024; Yoshida et al., 2025). A task vector is defined as the parameter difference between a base pre-trained model θ_{base} and a fine-tuned model θ_{task}. By adding or subtracting this vector within the original weight space (so-called "task arithmetic"), a user can edit or remove the corresponding task-specific behavior without further gradient-based training (Ilharco et al., 2023). Moreover, scaling the task vector grants fine-grained control over the strength of the transferred capability. This approach represents a promising direction, as it directly manipulates parameters while avoiding a costly re-optimization of the entire model.

In addition to these computational benefits, prior work has suggested that separating and analyzing task vectors may enhance interpretability (Cerrato et al., 2025). By isolating the weight updates associated with particular subgroups (e.g., racial or gender demographics), one can potentially trace how the model adapts to each subgroup. This feature is appealing for investigating biases arising from unequal representation in training data, as it highlights which groups require larger shifts in weight space. Nevertheless, open questions persist regarding how well this model editing using task-vector preserves or exacerbates fairness. For instance, improving performance for one demographic might degrade outcomes for another, and it is not yet clear how to balance trade-offs with established fairness metrics such as Demographic Parity (DPD) or Equalized Odds (EOD).

To address this gap, we systematically examine how task-vector editing compares to both traditional full-parameter fine-tuning (FFT) and LoRA, and we further explore whether injecting task vectors into an FFT model offers additional control over fairness. Our experiments focus on hate-speech detection on Llama-7B (Touvron et al., 2023) , measured by subgroup-specific accuracy and widely used fairness metrics. Our contributions and findings are summarized as follows:

- A thorough comparison of four algorithms (FFT, LoRA, model editing using task-vector, and a hybrid approach injecting task vectors into FFT) in terms of their effects on fairness metrics and overall performance (Figure 1)].
- An analysis showing that task vectors can substantially improve fairness while preserving accuracy, provided that their scalar coefficients are appropriately tuned (Figure 2).
- Evidence that merging task vectors for underrepresented subgroups with existing models can adjust fairness outcomes without incurring a significant accuracy drop (Figures 3a, 3b and 4a).

Through this analysis, we illustrate how task vectors can reduce risks from a fairness perspective while taking advantage of their flexibility and interpretability as a model editing approach. These findings provide a foundation for extending task-vector-based methods to promote fair and responsible operation of large language models.

2 Preliminaries

In this section, we first provide an overview of the fundamental concept of task vectors and the procedure known as task arithmetic, which applies these vectors to edit model behavior. We then introduce methods for merging multiple task vectors into a single model.

Task arithmetic. A task vector is defined as the difference in model parameters between a fine-tuned model on a given task and the original base model. Formally, if θ_{base} are the pre-trained weights and θ_{task} are the weights after fine-tuning on a task, then the task vector is: $\Delta\theta = \theta_{\text{task}} - \theta_{\text{base}}$ (Ilharco et al., 2023).

This vector represents a direction in weight space such that moving the base model's weights by $\Delta\theta$ steers the model to perform well on that task. In other words, adding $\Delta\theta$ to θ_{base} yields a model with improved performance on the target task, without any additional training. Once computed, task vectors can be manipulated through simple arithmetic operations to edit model behavior directly in weight space (Ilharco et al., 2023; Ortiz-Jimenez et al., 2024). Key operations include:

Addition: Given two task vectors $\Delta\theta_A$ and $\Delta\theta_B$ (for tasks A and B), their sum can be applied to the base model ($\theta_{\text{base}} + \Delta\theta_A + \Delta\theta_B$) to produce a model that exhibits improved performance on both tasks A and B (Ilharco et al., 2023). This task addition effectively combines knowledge from multiple tasks into one model.

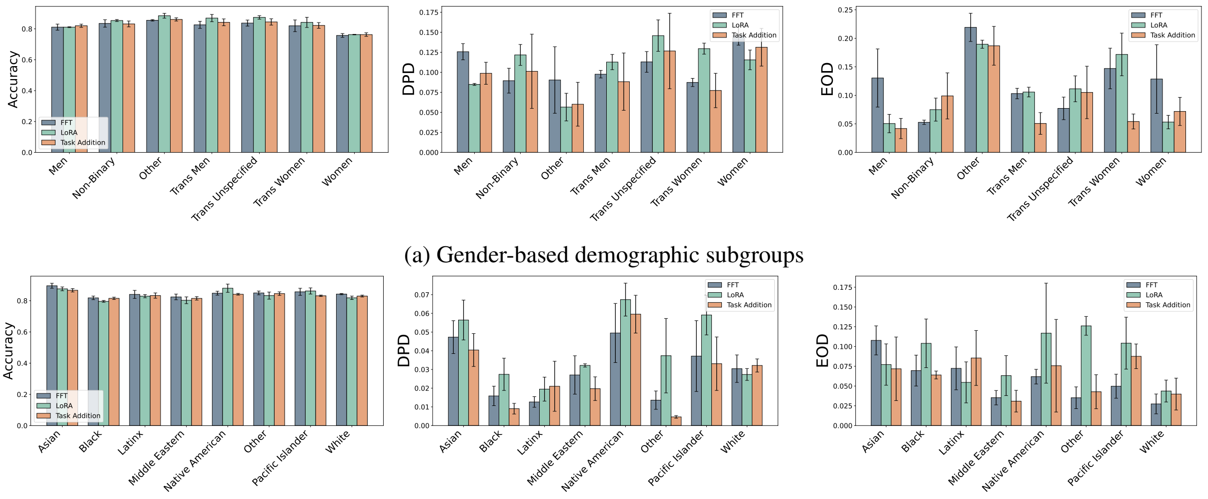
Negation: Using the negative of a task vector, $-\Delta\theta$, one can subtract a task's influence. Applying $\theta_{\text{base}} + (-\Delta\theta_A)$ (equivalently $\theta_{\text{base}} - \Delta\theta_A$) yields a model with decreased performance on task A, essentially unlearning or forgetting that task, while leaving other behaviors mostly unchanged (Ilharco et al., 2023). This is useful for removing undesirable skills or biases.

Scalar scaling: Multiplying a task vector by a scalar λ adjusts the strength of the edit. For example, using $\theta_{\text{base}} + \lambda\Delta\theta_A$ allows partial ($0 < \lambda < 1$) or amplified ($\lambda > 1$) application of a task's effect. This scaling provides fine-grained control over how strongly the task knowledge is injected into the model.

Merging task vectors. Since task vectors reside in a common weight space, they can be merged by simple addition with tunable scaling. Formally, given a base model θ_0 and task vectors $\Delta\theta_i$, one can construct a merged model as:

$$\theta_{\text{merged}} = \theta_0 + \sum_i \lambda_i \, \Delta\theta_i \,, \qquad (1)$$

where each coefficient λ_i controls the influence of task i. Varying λ_i thus directly modulates how strongly the i-th task's knowledge is injected, allowing fine-grained blending of capabilities. Indeed, adding multiple task vectors with $\lambda_i = 1$



(a) Gender-based demographic subgroups

(b) Race-based demographic subgroups

Figure 1: LoRA and FFT vs. Task addition with the optimal coefficient for the training accuracy ($\lambda = 0.8$ for gender setting and $\lambda = 0.5$ for race setting) on group-wise accuracy, demographic parity difference (DPD, lower is fairer), and equalized odds difference (EOD, lower is fairer). Error bars denote the standard error across three seeds. Columns: group-wise accuracy, DPD, EOD. No consistent pattern emerges that task addition necessarily degrades subgroup fairness relative to LoRA or FFT subgroups show improvements or comparable results under task addition, while others show small declines.

endows a model with all those capabilities simultaneously (Ilharco et al., 2023). Optimizing the λ_i values (i.e., learning an anisotropic scaling for each vector) further improves the composition by balancing contributions and reducing interference between tasks (Zhang et al., 2024).

3 Related Work

Task arithmetic: efficiency and interpretability. Task vectors offer a computationally efficient framework for editing and analyzing model behavior. Once a task vector is computed—namely, the weight difference between a base model and its fine-tuned variant (Ilharco et al., 2023; Zhang et al., 2024; Yoshida et al., 2025)—no additional training data or retraining is required to transfer or remove task-specific capabilities. By treating each fine-tuning update as a direction in weight space, practitioners can combine or negate these updates through simple addition or subtraction (Ilharco et al., 2023). This modularity not only reduces computational overhead but also enhances interpretability by isolating the contribution of each task.

Beyond modularity, task arithmetic can reveal valuable information about how and where a model adapts to new tasks. Li et al. (2024) show a near-linear relationship between data size and the norm of a task vector, suggesting that over-represented tasks can dominate weight space shifts in multi-task settings. In addition, the orientation of task vectors can indicate synergies or conflicts among tasks (Li et al., 2025), and decomposing these vectors by layer can pinpoint which parts of the model are most affected (Zhang et al., 2024; Gargiulo et al., 2025). Hence, task vectors offer a promising lens for diagnosing training dynamics and identifying potential biases.

Fairness metrics for LLMs. Fairness in large language models is commonly evaluated using criteria such as Demographic Parity, Equalized Odds, and accuracy parity. Demographic Parity requires similar positive outcome rates across demographic groups, while Equalized Odds demands that true and false positive rates be equivalent. Accuracy parity checks for consistent predictive performance across groups (Fraenkel, 2020; Kennedy et al., 2020a; Pitoura, 2019; Quan et al., 2023). These metrics are broadly used to detect biases and measure whether a model's behavior disproportionately disadvantages certain populations.

FFT and LoRA under fairness constraints. FFT remains a standard approach for aligning LLMs to specific tasks. However, FFT can inadvertently magnify biases in the data, leading to worsening performance for minority groups (Sap et al., 2019; Kotek et al., 2024). Studies have demonstrated that fine-tuned models may encode stronger biases than the original pretrained model, especially when training data are imbalanced (Jin et al., 2021; Zhang and Zhou, 2024; Salmani and Lewis, 2024).

Parameter-efficient methods such as LoRA (Hu et al., 2022) address computational bottlenecks by training only a small set of parameters, yet they do not inherently solve fairness issues. In some cases, LoRA yields comparable subgroup performance to full fine-tuning (Ding et al., 2024b), while in others, it fails to mitigate toxic behaviors or biases (Das et al., 2024). The variance in outcomes depends on factors like the rank of the LoRA matrices, the base model's quality, and the distribution of training data (Das et al., 2024).

Merging tasks and fairness considerations. Despite the potential efficiency gains and interpretability offered by task arithmetic, the merging of task vectors for multiple groups can trigger new challenges. For instance, simply summing vectors may lead to "negative transfer," where updates beneficial to one subgroup degrade performance for another (Ding et al., 2024a; Yu et al., 2020). In highly imbalanced settings, merging models through supervised fine-tuning can also disproportionately favor majority groups while disadvantaging minorities (Cross et al., 2024). Because fairness does not compose additively, interactions among subgroup-specific task vectors can produce unpredictable shifts in metrics like Demographic Parity and Equalized Odds (Gohar et al., 2023).

Consequently, identifying effective ways to adjust task vectors—such as through scalar scaling—remains a key step toward fairness-aware model editing. This work aims to fill that gap by systematically evaluating how these operations influence both fairness and overall model accuracy.

4 Experimental Setup

4.1 Configuration.

Building on the experimental framework established by (Ding et al., 2024b), we adopted their evaluation and experimental procedure to assess the fairness implications of LoRA in comparison to FFT. In our work, we extend this analysis by focusing on how task arithmetic compares to both LoRA and FFT in terms of fairness and performance. The detailed experimental setup is provided in Appendix B.

Gender Subgroups		Race Subgroups	
Men	817	Asian	311
Non-binary	114	Black	1,007
Trans men	178	Latinx	368
Trans unspecified	173	Native American	153
Trans women	148	Middle Eastern	493
Women	2,057	Pacific Islander	138
Other	59	White	580
		Other	302
Total	**3,546**	**Total**	**3,352**

Table 1: Data statistics in the gender and race subgroups.

Dataset. We use a modified version of the Berkeley D-Lab Hate Speech dataset originally introduced by Kennedy et al. (2020a) and adapted by Ding et al. (2024b), the research we are building upon. Our dataset contains a total of 6,898 tweet-sized text snippets annotated for hate speech and categorized by sensitive attributes: *Race* and *Gender*, each further divided into fine-grained subgroups (e.g., *Women*, *Non-binary*, *Men* within *Gender*) as shown in Table 1. We frame hate speech detection as a binary classification task: given a text snippet, the model predicts whether it constitutes hate speech (e.g., hatespeech in the Gender subset may target Non-binary or Trans Women). Each example includes both the hate speech label and one or more protected attribute annotations (e.g., *gender* = woman, *race* = Asian). These are used to assess subgroup-level performance and fairness metrics.

This setting supports rigorous fairness analysis due to its rich attribute annotations and real-world relevance. Hate speech detection is a challenging and high-stakes classification problem: it requires models to identify subtle or implicit harm, resolve linguistic ambiguity, and perform robustly across diverse dialects and identity references (Kennedy et al., 2020a). As models increasingly mediate content moderation, ensuring reliable and equitable hate speech detection is essential for safe deployment in real-world systems.

Evaluation metrics. We evaluate each method on both **predictive performance** and **fairness metrics**. Our goal is to understand how scaling or merging task vectors affects these measures.

- **Predictive Performance**:

  Accuracy: Standard metric for classification tasks, measuring the percentage of correct predictions.

- **Fairness Metrics:** We also adopt the metrics used for (Ding et al., 2024b) to quantify disparate performance across protected subgroups. These metrics are widely used for fairness research in ML:

  Demographic Parity Difference (DPD): Measures the disparity in the model's positive prediction rates across sensitive attribute groups. A smaller DPD indicates that the model assigns positive outcomes at similar rates across these groups, reflecting a more uniform treatment irrespective of group membership (Agarwal et al., 2018, 2019).

  Equalized Odds Difference (EOD) : Measures the disparity in the model's true and false positive rates across sensitive attribute groups. A smaller EOD indicates that the model's overall error rate is more balanced across groups (Das et al., 2024).

4.2 Protocol.

We evaluate our methods using a main base model: LLaMA2-7B[1]. Our fairness evaluations focus on two sensitive attributes: gender and race, using subgroup-wise metrics mentioned earlier – accuracy, DPD, and EOD.

For FFT, the pretrained model was fine-tuned on the combined training data from all subgroups of the target attribute (gender or race). Evaluation was then performed on the test data from each corresponding subgroup, enabling fine-grained assessment of both performance and fairness.

For LoRA, we followed the same training and evaluation procedure as FFT. In accordance with Ding et al. (2024b), the rank of the LoRA adaptation modules was set to 8.

For task arithmetic, we applied a compositional fine-tuning approach. The training data was partitioned by subgroup (gender or race), and FFT was applied separately to each subgroup's data to produce fine-tuned models θ_i. From these, we computed task vectors $\Delta\theta_i$ relative to the base model. These vectors were then merged using the approach described in Eq. (1), with a single, uniform scaling coefficient λ applied to all vectors. λ served as the sole hyperparameter in the merging process and was tuned on the training data. The evaluation metrics were computed in the same manner as for FFT and LoRA.

Task vector coefficient adjustment. Building on the task vector merging framework introduced in Eq. (1), we further explore the impact of the scaling coefficient λ on fairness outcomes. Specifically, we vary the uniform task vector coefficient λ across a broad range (from 0.0 to 1.0 with 0.1 intervals) and evaluate how this adjustment influences subgroup-level fairness metrics, including accuracy, DPD, and EOD.

Impact of worst-performing subgroup task vectors on fairness and performance. To investigate whether incorporating task vectors from underperforming subgroups can improve fairness without sacrificing overall performance, we first identified the lowest-performing subgroups within each attribute based on the average of DPD and EOD under the FFT setting. We excluded the "others" group from this analysis as it does not reflect the characteristics of any specific subgroup. This selection was informed by both our experimental results and those reported in Ding et al. (2024b), which showed consistent patterns. For gender, the worst-performing subgroups were men and women; for race, they were Asian and Native American. We constructed a new model variant by injecting a worst-performing subgroup task vector worst-performing subgroup task vector into the base fine-tuned model:

$$\theta_{\text{new}} = \theta_{\text{SFT}} + \lambda(\theta_{\text{worst-performing subgroup}} - \theta_0) \quad (400)$$

where λ controls the strength of the task vector injection. We varied λ from from 0.0 to 1.0 at 0.2 intervals to analyze the effect of this targeted addition on subgroup fairness metrics and overall accuracy.

5 Results

Overview.

In Figure 1a, we compare the performance of FFT, LoRA, and task addition across gender subgroups; Figure 1b presents results for race subgroups. For task addition, we selected $\lambda = 0.8$ for gender, $\lambda = 0.5$ for race, as it achieved the highest average training accuracy across three random seeds within the tested range $\lambda \in [0.0, 1.0]$. These visualizations provide a direct comparison of subgroup-wise model behavior.

[1] LLaMA 2 is licensed under the LLAMA 2 Community License, Copyright (c) Meta Platforms, Inc. All Rights Reserved. See: `https://ai.meta.com/llama/license`

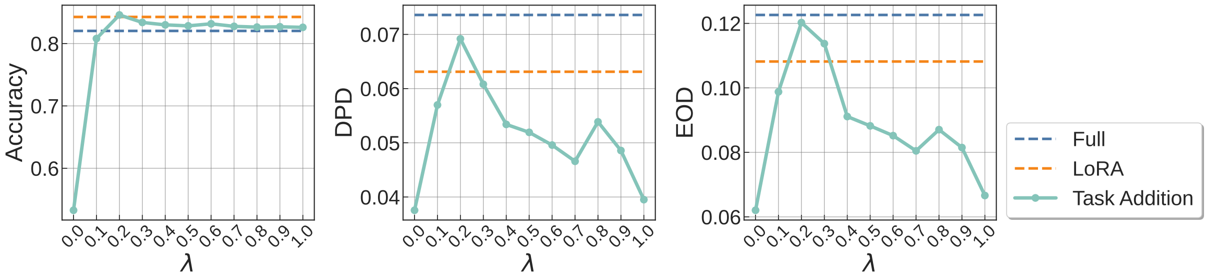


Figure 2: Varying the task arithmetic coefficient λ and comparing against FFT (purple dashed) and LoRA (orange dashed) for macro-averaged accuracy (left), demographic parity difference (DPD, center), and equalized odds difference (EOD, right). Evaluated on the **gender** subset of the data. Higher accuracy is better, while lower DPD and EOD indicate improved fairness. As λ changes, task arithmetic maintains competitive accuracy and can reduce fairness gaps relative to the baselines.

From the subgroup-level bar plots in Figure 1, we observe that accuracy remains consistently high and comparable across all three adaptation methods, regardless of subgroup.

However, fairness metrics (DPD and EOD) show notable variation across methods. The impact of task addition on fairness is not consistent. Compared to FFT, task addition improved fairness in 5 out of 7 gender subgroups and in 3 out of 8 race subgroups. No single method yielded the best fairness performance across all demographic groups.

Our experimental results closely align with the findings reported in (Ding et al., 2024b), particularly regarding the performance of FFT and LoRA on macro-averaged accuracy, DPD, and EOD. This consistency across demographic categories and with prior literature reinforces the robustness of our observations and supports the reliability of our evaluation framework.

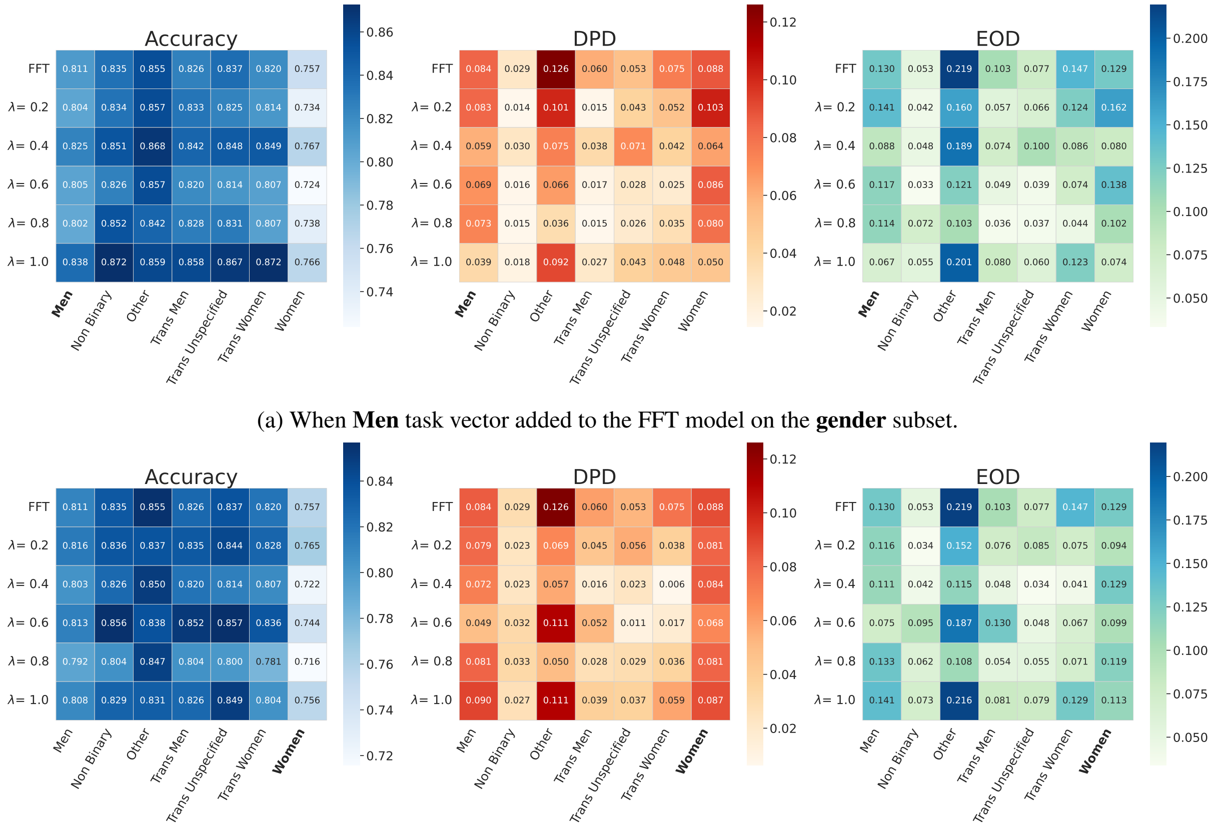
5.1 Controlling accuracy and fairness metrics through lambda.

Figure 2 illustrates the overall performance of FFT, LoRA, and task arithmetic as the scaling coefficients for task addition vary from 0.0 to 1.0. We observe how varying the task-arithmetic coefficient λ impacts macro-averaged accuracy (left), demographic parity difference (DPD, center), and equalized odds difference (EOD, right) on a gender subset of the data. As λ increases from 0.0 to 0.2, we observe a peak in accuracy, but this configuration yields higher DPD and EOD, indicating reduced fairness. Beyond $\lambda = 0.3$, accuracy remains competitive compared to FFT and LoRA, while both DPD and EOD progressively decline, suggesting that fairness improves without severely compromising performance. Notably, these task addition curves stay consistently lower than FFT and LoRA in terms of DPD and EOD at higher λ values. Overall, this ablation could indicate that tuning λ provides a practical mechanism for balancing accuracy and fairness objectives, offering guidelines for practitioners who wish to fine-tune fairness outcomes while maintaining strong predictive performance.

5.2 Mixed trends when adding worst-performing subgroup task vectors

To further analyze the effects of subgroup-specific task composition, Figure 3a–3b illustrate heatmaps where the y-axis lists each method or configuration under evaluation: FFT as baseline, followed by task arithmetic with varying scaling coefficients (0.0 to 1.0 with 0.2 intervals). The x-axis represents the subgroups— (e.g., Women, Trans, etc. for Gender). Each cell shows the corresponding performance metric (e.g., macro-averaged accuracy, DPD, or EOD for a given method on a specific subgroup. For these experiments, we added the task vector of the worst-performing subgroups (Women and Men for the gender dataset subset, and Asian, and Native American for the race dataset subset) to the FFT model, as explained earlier.

We generally observe that increasing the scaling coefficient λ tends to improve overall accuracy, consistent with the trends observed in Figure 2. However, the impact on fairness metrics (DPD and EOD) is more variable. In the gender-based plots, for example, the Asian subgroup consistently achieves the highest accuracy and lowest DPD/EOD—highlighting a recurring tradeoff where performance gains for one group may exacerbate disparities for others. When the Women task vector is added (Figure 3b), accuracy improves for the Trans Women subgroups. However, fair-



(a) When **Men** task vector added to the FFT model on the **gender** subset.

Accuracy

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.811	0.835	0.855	0.826	0.837	0.820	0.757
λ= 0.2	0.804	0.834	0.857	0.833	0.825	0.814	0.734
λ= 0.4	0.825	0.851	0.868	0.842	0.848	0.849	0.767
λ= 0.6	0.805	0.826	0.857	0.820	0.814	0.807	0.724
λ= 0.8	0.802	0.852	0.842	0.828	0.831	0.807	0.738
λ= 1.0	0.838	0.872	0.859	0.858	0.867	0.872	0.766

DPD

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.084	0.029	0.126	0.060	0.053	0.075	0.088
λ= 0.2	0.083	0.014	0.101	0.015	0.043	0.052	0.103
λ= 0.4	0.059	0.030	0.075	0.038	0.071	0.042	0.064
λ= 0.6	0.069	0.016	0.066	0.017	0.028	0.025	0.086
λ= 0.8	0.073	0.015	0.036	0.015	0.026	0.035	0.080
λ= 1.0	0.039	0.018	0.092	0.027	0.043	0.048	0.050

EOD

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.130	0.053	0.219	0.103	0.077	0.147	0.129
λ= 0.2	0.141	0.042	0.160	0.057	0.066	0.124	0.162
λ= 0.4	0.088	0.048	0.189	0.074	0.100	0.086	0.080
λ= 0.6	0.117	0.033	0.121	0.049	0.039	0.074	0.138
λ= 0.8	0.114	0.072	0.103	0.036	0.037	0.044	0.102
λ= 1.0	0.067	0.055	0.201	0.080	0.060	0.123	0.074

(b) When **Women** task vector added to the FFT model on the **gender** subset.

Accuracy

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.811	0.835	0.855	0.826	0.837	0.820	0.757
λ= 0.2	0.816	0.836	0.837	0.835	0.844	0.828	0.765
λ= 0.4	0.803	0.826	0.850	0.820	0.814	0.807	0.722
λ= 0.6	0.813	0.856	0.838	0.852	0.857	0.836	0.744
λ= 0.8	0.792	0.804	0.847	0.804	0.800	0.781	0.716
λ= 1.0	0.808	0.829	0.831	0.826	0.849	0.804	0.756

DPD

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.084	0.029	0.126	0.060	0.053	0.075	0.088
λ= 0.2	0.079	0.023	0.069	0.045	0.056	0.038	0.081
λ= 0.4	0.072	0.023	0.057	0.016	0.023	0.006	0.084
λ= 0.6	0.049	0.032	0.111	0.052	0.011	0.017	0.068
λ= 0.8	0.081	0.033	0.050	0.028	0.029	0.036	0.081
λ= 1.0	0.090	0.027	0.111	0.039	0.037	0.059	0.087

EOD

	Men	Non Binary	Other	Trans Men	Trans Unspecified	Trans Women	Women
FFT	0.130	0.053	0.219	0.103	0.077	0.147	0.129
λ= 0.2	0.116	0.034	0.152	0.076	0.085	0.075	0.094
λ= 0.4	0.111	0.042	0.115	0.048	0.034	0.041	0.129
λ= 0.6	0.075	0.095	0.187	0.130	0.048	0.067	0.099
λ= 0.8	0.133	0.062	0.108	0.054	0.055	0.071	0.119
λ= 1.0	0.141	0.073	0.216	0.081	0.079	0.129	0.113

Figure 3: Heatmaps of Accuracy (left), DPD (center), and EOD (right) for gender (top) and race (bottom) subgroups under the baseline FFT model ($\lambda = 0.0$) and with increasing λ values from 0.2 to 1.0 in 0.2 increments. The task vector for Men was added on the gender subset (top), and the task vector for Women was added on the gender subset (bottom). Darker cells indicate higher values on each metric's scale; for DPD/EOD, lower values are better.

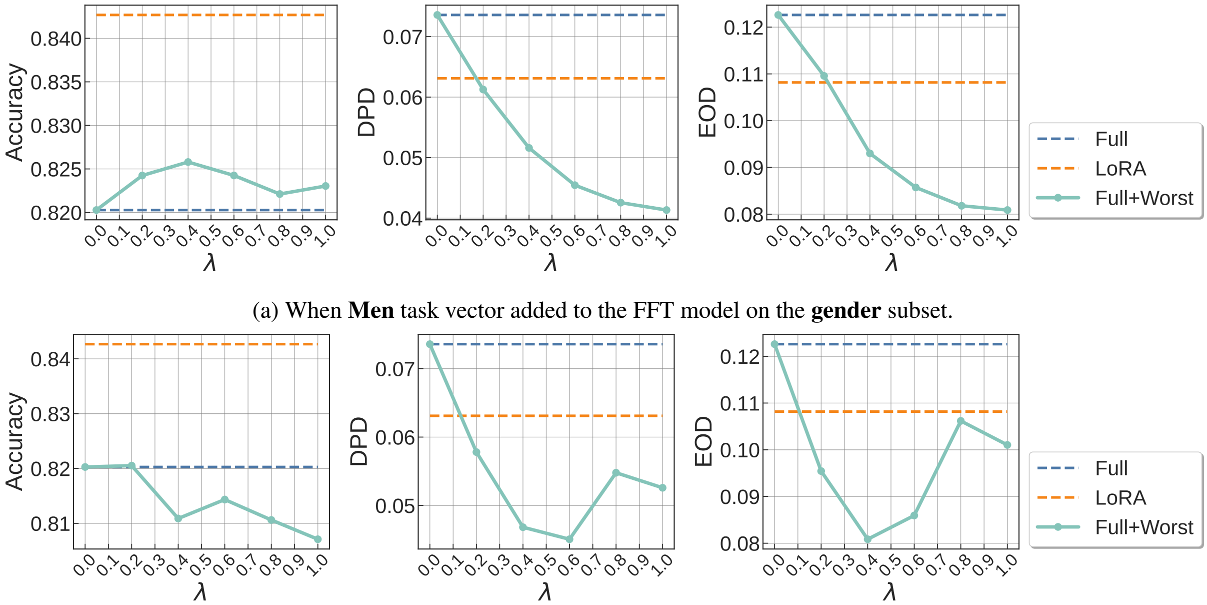
ness metrics for subgroups such as Men tend to worsen as the scaling coefficient λ increases. In Figure 3a, injecting the Men task vector improves performance for some subgroups, yet Women consistently show lower accuracy and do not see consistent fairness improvements at higher λ. Some groups (e.g., Other, Trans Men, Trans Women) begin with relatively poor fairness under FFT and show partial improvements with task vector addition. Still, these improvements are not universal—for example, the Other subgroup often retains high EOD values regardless of λ. Likewise, Native American accuracy remains mostly unchanged across λ, while fairness metrics can deteriorate when injecting task vectors for other groups.

Overall, while increasing λ can improve both accuracy and fairness for certain subgroups, these effects are not consistent across all configurations.

To visualize these results in more detail, Figure 4a shows macro-averaged accuracy, DPD, and EOD for the Men task vector added to the FFT model. The plots illustrate how varying the scaling coefficient λ impacts overall performance and fairness, highlighting the effects of subgroup-specific task injection. We can observe in Figure 4a that injecting the Men task vector into the FFT model results in a slight accuracy gain and a clear monotonic decrease in both DPD and EOD as λ increases—indicating a favorable and consistent improvement in fairness on the gender subset.

However, Figure 4b and the additional plots in Figures 10 and 11 in Appendix C.2 show more varied patterns as seen on Figures 3a and 3b. When injecting the Native American task vector (Figure 11), accuracy remains stable while fairness seems to decrease (increased DPD and EOD). Asian (Figure 10) shows the same behavior as injecting the Men task vector (Figure 4a), positive increase of fairness metrics as λ increases.

Taken together, these results confirm that task vector injection can shift subgroup-wise fairness and performance, but its effects are highly group-specific. While some subgroups (e.g., Men, Asian) exhibit smooth fairness gains, others (e.g., Women)



(a) When **Men** task vector added to the FFT model on the **gender** subset.

(b) When **Women** task vector added to the FFT model on the **gender** subset.

Figure 4: Impact of injecting both the **Men** and **Women** subgroup task vectors into the FFT model on the gender data subset. The plot illustrates how scaling coefficient λ reduces DPD and EOD, outperforming the baseline FFT (blue dashed) and LoRA (orange dashed), with negligible impact on macro-averaged accuracy.

show erratic behavior. This reinforces our earlier conclusion: simple task addition is not a guaranteed to have positive fairness influence, and more targeted strategies are likely needed - yet the scaling coefficient does show some relevant influence in fairness metrics.

6 Conclusion and Limitations

Conclusion. In this study, we investigated the impact of a task arithmetic approach using task vectors on fairness, in comparison to conventional FFT and LoRA methods. We conducted detailed experiments to assess how the task addition affects prediction accuracy and fairness metrics, including the DPD and EOD across various subgroups. The results indicate that, with appropriate settings of the scalar coefficient λ, the task arithmetic method can improve DPD and EOD without significantly compromising overall model accuracy. Notably, using low to moderate values of the task vector coefficient effectively reduced prediction bias in minority groups compared to FFT and LoRA.

Furthermore, the task arithmetic framework allows for subgroup-specific evaluation and adjustment of model updates, enhancing interpretability—a key advantage of this method in the context of fairness. This interpretability facilitates the mitigation of excessive bias or adverse effects on particular groups, ultimately enabling more balanced model training.

Limitations. Despite these promising results, several challenges remain. The effectiveness of task arithmetic depends on dataset characteristics and subgroup distributions, necessitating further investigation into its generalizability across different tasks and domains. Moreover, future work should explore algorithms for automatically optimizing the scalar coefficient λ and for balancing trade-offs among multiple subgroups.

In summary, our study demonstrates that task arithmetic using task vectors offers a promising approach for controlling model fairness. Further experimental validation, application to diverse tasks, and developing trade-off optimization methods are essential for improving fairness in broader and more realistic deployment scenarios.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 60–69. PMLR.

Alekh Agarwal, Miroslav Dudik, and Zhiwei Steven Wu. 2019. Fair regression: Quantitative definitions and reduction-based algorithms. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 120–129. PMLR.

Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. 2022. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1–9.

Mattia Cerrato, Marius Köppel, Alexander Segner, and Stefan Kramer. 2025. Fair interpretable learning via correction vectors. *arXiv preprint*.

James I. Cross, Wei Chuangpasomporn, and John A. Omoronyia. 2024. Bias in medical ai: Implications for clinical decision-making. *PLOS Digital Health*, 1(1):e0000561. Published on 7 Nov 2024.

Saswat Das, Marco Romanelli, Cuong Tran, Zarreen Reza, Bhavya Kailkhura, and Ferdinando Fioretto. 2024. Low-rank finetuning for llms: A fairness perspective. *arXiv preprint*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Chuntao Ding, Zhichao Lu, Shanguang Wang, Ran Cheng, and Vishnu N. Boddeti. 2024a. Mitigating task interference in multi-task learning via explicit task routing with non-learnable primitives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Zhoujie Ding, Ken Liu, Pura Peetathawatchai, Berivan Isik, and Sanmi Koyejo. 2024b. On fairness of low-rank adaptation of large models. *arXiv preprint*. Published: 10 Jul 2024, Last Modified: 25 Aug 2024.

Aaron Fraenkel. 2020. *Fairness and Algorithmic Decision Making*. Lecture Notes for UCSD course DSC 167.

Antonio Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. 2025. Task singular vectors: Reducing task interference in model merging. *arXiv preprint arXiv:2412.00831*. Version 3, 3 Jan 2025.

Usman Gohar, Nuno Ribeiro, and Harichandra Ramadurgam. 2023. Towards understanding fairness and its composition in ensemble machine learning. *arXiv preprint arXiv:2102.96452*. Version 3, 25 Mar 2023.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mohammad Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shawn Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *International Conference on Learning Representations (ICLR)*. Published as a conference paper at ICLR 2023.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling bert for natural language understanding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4163–4174.

Xisen Jin, Francesca Barbieri, Brendan Kennedy, Aida Mostafazadeh Davani, Leonardo Neves, and Xiang Ren. 2021. Transferability of bias mitigation effects in language model fine-tuning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3745–3757, Online. Association for Computational Linguistics.

Brendan Kennedy, Xisen Jin, Aida Mostafazadeh Davani, Morteza Dehghani, and Xiang Ren. 2020a. Contextualizing hate speech classifiers with post-hoc explanation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages –.

Chris J Kennedy, Geoff Bacon, Alexander Sahn, and Claudia von Vacano. 2020b. Constructing interval variables via faceted rasch measurement and multitask deep learning: a hate speech application. *arXiv preprint arXiv:2009.10277*.

Hadas Kotek, David Q. Sun, Zidi Xiu, Margit Bowler, and Christopher Klein. 2024. Protected group bias and stereotypes in large language models. *arXiv preprint arXiv:2403.14772*. 21 Mar 2024.

Hongkang Li, Yinhua Zhang, Shuai Zhang, Pin-Yu Chen, Sijia Liu, and Meng Wang. 2025. When is task vector provably effective for model editing? a generalization analysis of nonlinear transformers. *International Conference on Learning Representations (ICLR)*. Published as conference paper at ICLR 2025.

Mingxin Li, Zhijie Nie, Yanzhao Zhang, Dingkun Long, Richong Zhang, and Pengjun Xie. 2024. Improving general text embedding model: Tackling task conflict and data imbalance through model merging. *arXiv preprint arXiv:2410.15035v1*. 19 Oct 2024.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. 2024. Task arithmetic in the tangent space: Improved editing of pre-trained models. *arXiv preprint*.

Evaggelia Pitoura. 2019. Towards diversity-aware, fair, and unbiased data management. In *ISIP 2019*, Heraklion, Greece.

Tangkun Quan, Fei Zhu, Quan Liu, and Fanzhang Li. 2023. Learning fair representations for accuracy parity. *Engineering Applications of Artificial Intelligence*, 119:105819.

Parisa Salmani and Peter R. Lewis. 2024. Transfer learning can introduce bias. In *Proceedings of the 6th European Conference on the Impact of Artificial Intelligence and Robotics (ECIAIR)*. IOS Press. Published under CC-BY 4.0 License.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Catalin Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2020. Well-read students learn better: On the importance of pre-training compact models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3601.

Kotaro Yoshida, Yuji Naraki, Takafumi Horie, Ryosuke Yamaki, Ryotaro Shimizu, Yuki Saito, Julian McAuley, and Hiroki Naganuma. 2025. Mastering task arithmetic: τ jp as a key indicator for weight disentanglement. In *The Thirteenth International Conference on Learning Representations*.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, volume 33.

Frederic Z. Zhang, Paul Albert, Cristian Rodriguez-Opazo, Anton van den Hengel, and Ehsan Abbasnejad. 2024. Knowledge composition using task vectors with learned anisotropic scaling. In *Advances in Neural Information Processing Systems (NeurIPS)*. Main Conference Track.

Yuxuan Zhang and Feng Zhou. 2024. Bias mitigation in fine-tuning pre-trained models for enhanced fairness and efficiency. *arXiv preprint arXiv:2309.00625*. 6 Mar 2024.

A Fairness metrics

A.1 Demographic Parity Difference (DPD) (Agarwal et al., 2018, 2019)

DPD measures how varied the model's rate of positive predictions are across attributes. This metric is calculated as follows:

$$M_{\text{DPD}} = \Big|\Pr[f(X) = 1 \mid A = 1] - \Pr[f(X) = 1 \mid A = 0]\Big| \quad (2)$$

where A is the sensitive attributes, $f(X)$ is the prediction from the models, and X is the feature vector. The larger the DPD, the greater the difference in prediction outcomes across attributes, indicating greater unfairness in the model predictions.

A.2 Equalized Odds Difference (EOD) (Ding et al., 2024b)

EOD is a metric that measures whether the model exhibits similar predictive performance in terms of true and false positives, regardless of the attribute.

$$M_{\text{eod}} = \max\{M_{\text{TP}}, M_{\text{FP}}\} \quad (3)$$

Here, letting Y denote the true label, M_{TP} and M_{FP} are defined as follows:

$$M_{\text{TP}} = \Big|\Pr[f(X) = 1 \mid Y = 1, A = 1] - \Pr[f(X) = 1 \mid Y = 1, A = 0]\Big|, \quad (4)$$

$$M_{\text{FP}} = \Big|\Pr[f(X) = 1 \mid Y = 0, A = 1] - \Pr[f(X) = 1 \mid Y = 0, A = 0]\Big| \quad (5)$$

A.3 Accuracy Parity

Accuracy parity refers to the expectation that a classifier achieves comparable accuracy across different sensitive attribute groups. Formally, accuracy parity is satisfied when the probability of correct classification is equal across groups, i.e.,

$$\mathbb{E}(Y = \hat{Y} \mid S = 0) = \mathbb{E}(Y = \hat{Y} \mid S = 1) \quad (6)$$

This notion of fairness ensures that all subgroups receive equally reliable predictions, and is particularly relevant in applications where consistent model performance across demographics is critical. Unlike statistical parity or equal opportunity, accuracy parity focuses on equal overall correctness rather than specific error types or outcome rates (Quan et al., 2023).

We observed **high degree of accuracy parity** in both gender and race settings, as the accuracy differences between subgroups are negligible, indicating that the model performs consistently across all groups.

B Experimental details

B.1 Computational Resources and Software Environment

Hardware and Software: All experiments presented in this study were performed using computational resources equipped with two NVIDIA H100 GPUs. The experiments leveraged a GPU environment consisting of CUDA 12.1.0, cuDNN 9.0.0, and NCCL 2.20.5 .

The experiments were conducted using Python 3.9.18, incorporating several essential Python libraries specifically optimized for deep learning tasks. The primary libraries included PyTorch (version 2.6.0), transformers (version 4.49.0), tokenizers (version 0.21.1), DeepSpeed (version 0.16.4), and Accelerate (version 1.5.2).

The training experiments utilized the DeepSpeed framework with the following key configurations: a gradient accumulation step of 4, optimizer offloaded to the CPU, zero redundancy optimizer at stage 2 (ZeRO-2), and mixed precision training employing FP16 and BF16 for enhanced performance and memory efficiency. All experiments were conducted with a total computational cost of approximately 30 GPU-hours.

Protocol: We fine-tuned models based on the Llama-7B (Touvron et al., 2023) architecture obtained via HuggingFace repositories.

Each model was trained for 4 epochs, employing a cosine learning rate scheduler with a learning rate of 1×10^{-5}, a warm-up ratio of 0.01, and a weight decay of 0.001. Training utilized a per-device batch size of 2, with an effective batch size of 16 achieved through gradient accumulation. Reproducibility was ensured by setting a random seed of 13, 14, 15 across all experiments.

For Low-Rank Adaptation (LoRA) experiments were conducted with a rank (lora_r) of 8, scaling factor (lora_alpha) of 16, and no dropout.

B.2 Dataset

We use the Berkeley D-Lab hatespeech detection dataset (Kennedy et al., 2020b) [2] for our experiments. The dataset is divided into subgroups based on the following attributes: *Race or Ethnicity*, *Religion*, *National Origin or Citizenship Status*, *Gender Identity*, *Sexual Orientation*, *Age* and *Disability Status*. In our study, we use some of these subgroups to evaluate fairness.

Following (Das et al., 2024), we binarize the hate speech score associated with each review using a threshold of 0.5 to determine whether the review constitutes hate speech. When multiple annotations exist for the same instance, we obtain one human annotation to avoid duplication.

C Additional Results

Here, we present results focusing on diverse subgroups, which we could not include in the main paper due to space constraints.

C.1 Comparison of FFT, LoRA, and Task Arithmetic

Figure 7 illustrates the overall performance of FFT, LoRA, and task arithmetic as the scaling for task arithmetic vary from 0.0 to 1.0. Trends observed reinforced results on the gender subset on Figure 2. Overall, λ provides a practical mechanism for balancing accuracy and fairness objectives, and similarly there is a peak at $\lambda = 0.2$ for highest accuracy, and higher DPD and EOD (less fairness).

C.2 Subgroup-Specific Task Addition to FFT

We include additional heatmaps that visualize subgroup-wise performance across FFT and varying scaling coefficients for the FFT model injected with a worst-performing subgroup. These supplementary plots, which follow the same setup described earlier, are consistent with the trends observed in Figures 3a–3b.

In both gender and race subgroup experiments, increasing the scaling coefficient λ generally leads to improved macro-averaged accuracy. However, its impact on fairness metrics—DPD and EOD—is less predictable and varies across subgroups. For instance, some subgroups benefit from improved fairness as their corresponding task vectors are added, while others experience increased disparity, even if accuracy remains stable or improves.

This nuanced behavior reflects a broader pattern: gains in performance for certain subgroups can sometimes come at the expense of fairness for others. Injecting task vectors from worst-performing subgroups does not consistently reduce disparities and, in some cases, can amplify them.

Figures 11–4b present additional results for the Full+Worst configuration, in which task vectors from the worst-performing subgroups (Native American, Asian, Men, and Women) are added to the FFT model. These plots show macro-averaged accuracy, DPD, and EOD as a function of the scaling coefficient λ.

Across these figures, we observe mixed effects: while accuracy generally remains stable or improves slightly, fairness outcomes vary by subgroup. In Figure 11, DPD and EOD worsen despite minimal accuracy changes. Meanwhile, Figure 4b reveals stable performance with minor fairness improvements, though gains are not consistent across metrics. These results further emphasize that task vector injection alone does not ensure universal fairness improvements and often introduces subgroup-specific trade-offs.

[2] https://huggingface.co/datasets/ucberkeley-dlab/measuring-hate-speech

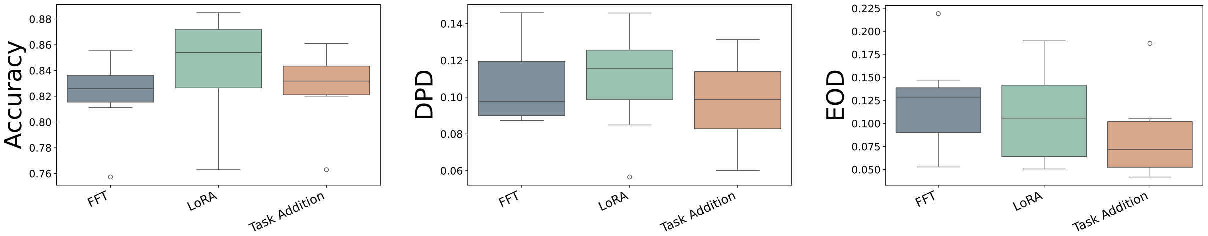


Figure 5: Boxplots of group-wise accuracy, demographic parity difference (DPD), and equalized odds difference (EOD) for —FFT, LoRA, and task addition with coefficient ($\lambda = 0.8$) —evaluated on the **gender** subset of the data. Higher accuracy is desirable, whereas lower DPD and EOD values indicate improved fairness. Boxplots show medians, interquartile ranges, and variability (with standard error across three seeds). While accuracy is similar across methods, Task Addition generally yields lower DPD and EOD medians than FFT and LoRA, suggesting a better balance between performance and fairness, though overlapping distributions imply these differences are not uniformly significant.

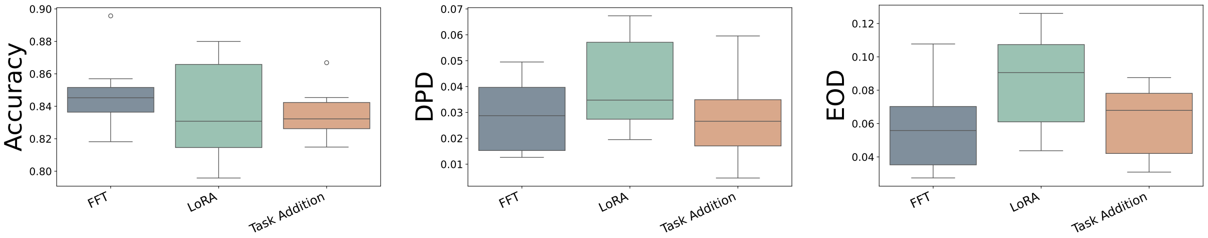


Figure 6: Boxplots of group-wise accuracy, demographic parity difference (DPD), and equalized odds difference (EOD) for —FFT, LoRA, and Task Addition with optimal coefficient ($\lambda = 0.5$) —evaluated on the **race** subset of the data. Higher accuracy is desirable, whereas lower DPD and EOD values indicate improved fairness. Boxplots show medians, interquartile ranges, and variability (with standard error across three seeds).

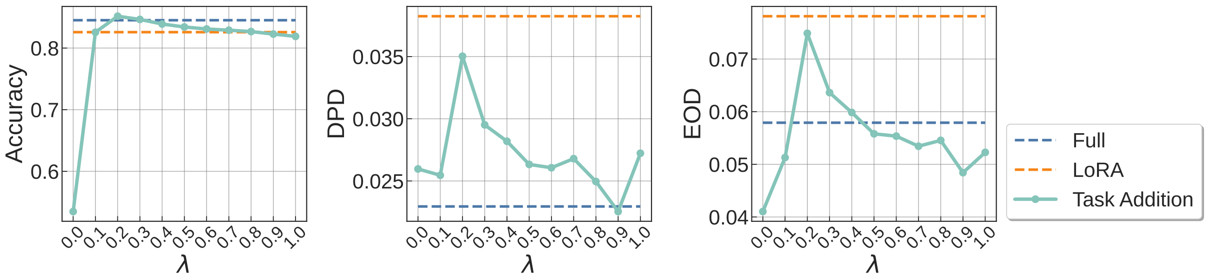


Figure 7: On a **race-focused** subset, we vary task arithmetic's coefficient λ and compare it against FFT (purple dashed) and LoRA (orange dashed). The plots show group-wise accuracy (left), demographic parity difference (DPD, center), and equalized odds difference (EOD, right). Higher accuracy is better, while lower DPD and EOD indicate improved fairness. As λ changes, task arithmetic remains competitive in accuracy and can reduce fairness gaps relative to the baselines.

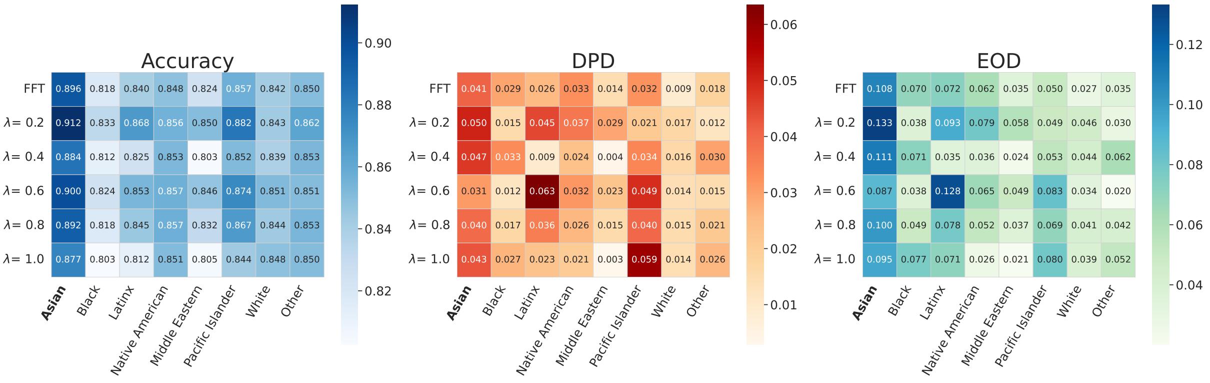


Figure 8: The task vector corresponding to **Asian** was added to the FFT model on the race data subset. Heatmap of Accuracy (left), DPD (center), and EOD (right) under the baseline (FFT) and increasing λ values (0.2 to 1.0). Darker cells indicate higher values in each metric's scale; for DPD/EOD, lower is better.

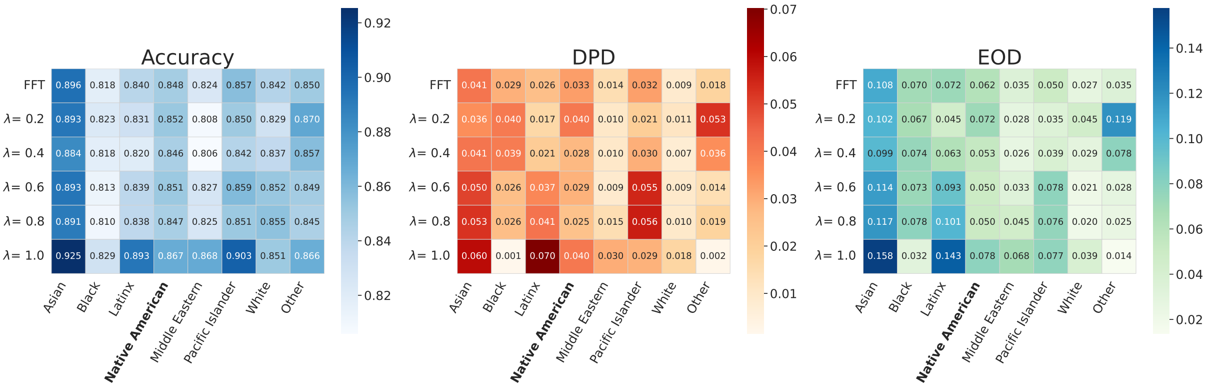


Figure 9: The task vector corresponding to **Native American** was added to the FFT model on the race data subset. Heatmap of Accuracy (left), DPD (center), and EOD (right) under the baseline (FFT) and increasing λ values (0.2 to 1.0). Darker cells indicate higher values in each metric's scale; for DPD/EOD, lower is better.

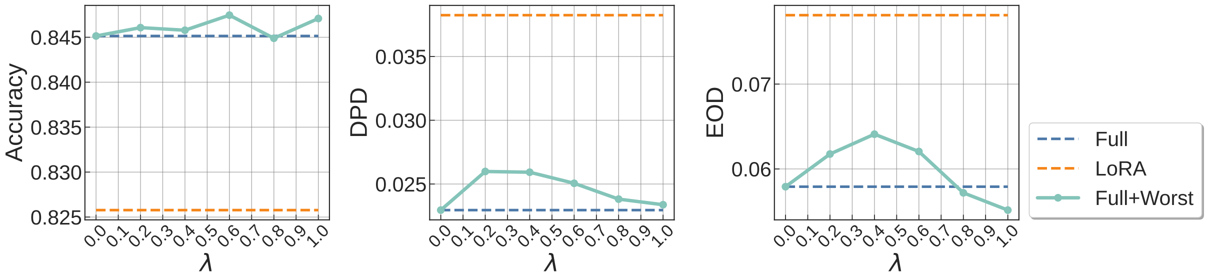


Figure 10: Effect of adding the **Asian** task vector to the FFT model on the **race** subset. Accuracy keeps competitive with increasing λ, and both DPD and EOD decrease consistently.

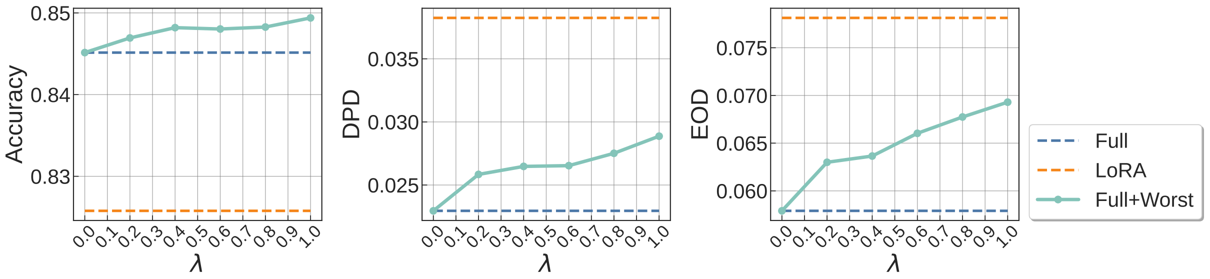


Figure 11: Results of injecting the **Native American** task vector into the FFT model. Accuracy shows minimal change across λ, while DPD and EOD increase (worsen fairness).