BiasEdit: Debiasing Stereotyped Language Models via Model Editing

Xin Xu[1], **Wei Xu**[2], **Ningyu Zhang**[3] **Julian McAuley**[1]
[1]University of California, San Diego, [2]Georgia Institute of Technology
[3]Zhejiang University,
xinxucs@ucsd.edu

Abstract

Warning: This paper explicitly contains the statement of stereotypes that may be offensive.

Previous studies have established that language models manifest stereotyped biases. Existing debiasing strategies, such as retraining a model with counterfactual data, representation projection, and prompting, often fail to efficiently eliminate bias or directly alter the models' biased internal representations. To address these issues, we propose **BiasEdit**, an efficient model editing method to remove stereotypical bias from language models through lightweight networks that act as editors to generate parameter updates. BiasEdit employs a *debiasing loss* guiding editor networks to conduct local edits on partial parameters of a language model for debiasing while preserving the language modeling abilities during editing through a *retention loss*. Experiments on StereoSet and Crows-Pairs demonstrate the effectiveness, efficiency, and robustness of BiasEdit in eliminating bias compared to tangental debiasing baselines, and little to no impact on the language models' general capabilities. In addition, we conduct bias tracing to probe bias in various modules and explore bias editing impacts on different components of language models[1].

1 Introduction

In recent years, many studies have underscored the tendency of pre-trained language models (LMs) to have societally stereotypical biases (Liang et al., 2021; Smith et al., 2022; Cheng et al., 2023a; Liu et al., 2023), such as gender bias (Sun et al., 2019; Zhao et al., 2020), race bias (Halevy et al., 2021), religion bias (Das et al., 2023; Manzini et al., 2019), among others. Therefore, eliminating biases from models is crucial to ensure fairness and accuracy in applications of language models.

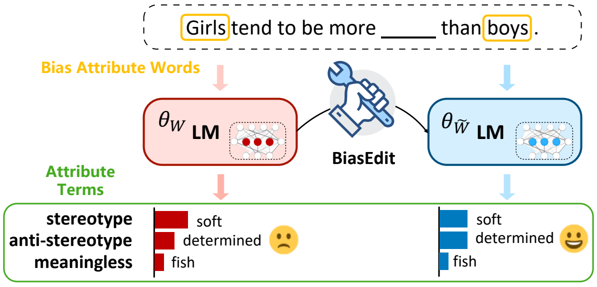


Figure 1: Debiasing a language model with **BiasEdit**.

Many methods have been proposed to mitigate bias, such as fine-tuning entire models (Zmigrod et al., 2019; Barikeri et al., 2021) with counterfactual data obtained by swapping out bias attribute words,[2] which is partly effective but costly in terms of computational time and space, especially for large language models (LLMs). Others implement debiasing with representation projection (Ravfogel et al., 2020; Liang et al., 2020; Limisiewicz and Marecek, 2022; Iskander et al., 2023) or prompting (Sheng et al., 2020; Schick et al., 2021; Mattern et al., 2022; Venkit et al., 2023). However, without parameter modification, a model remains inherently biased and can not be applied to downstream tasks as an off-the-shelf unbiased model. Recent methods (Kumar et al., 2023; Limisiewicz et al., 2024) employ model adapters where each adapter is trained to specialize only in one bias type. Multiple adapter training for different bias types is not economical for real-world applications.

These drawbacks inspire us to explore new methods for debiasing stereotyped language models more directly. Model editing (Yin et al., 2023; Wei et al., 2023; Zhang et al., 2024) can change specific information in language models by modifying model parameters, which could be effective in eliminating bias. There are some existing edit-

[1] Code and data are available in `https://github.com/zjunlp/BiasEdit`

[2] The bias attribute words refer to those that introduce or reflect bias. For example, bias attribute words for gender are *she*, *he*, *mother*, *father*, etc. Bias attribute words for religion are *Christianity*, *Judaism*, *Islam*, etc.

ing methods: (i) fine-tuning a model with new data (Zhu et al., 2020; Ni et al., 2023); (ii) locating then editing (Meng et al., 2022, 2023; Dai et al., 2022; Wu et al., 2023b; Li et al., 2024); (iii) utilizing editor hyper-networks to modify language models' parameters (Cao et al., 2021; Mitchell et al., 2022; Cheng et al., 2023b; Tan et al., 2023). As for current LLMs (usually >10B for practical applications), the fine-tuning approach consumes a lot of computational resources and data, which is not ideal. Recent works (Limisiewicz et al., 2024; Yan et al., 2024; Chen et al., 2024) and our preliminary experiments (see Appendix A) show that bias can be interpreted as localized modules in LLMs. Meanwhile, small hyper-networks predicting weight updates (Cao et al., 2021; Mitchell et al., 2022; Tan et al., 2023) are illustrated to be flexibly applied to change parameters of any language models without fully fine-tuning it and adaptively designed to conduct any specific editing task.

In §3, therefore, we introduce **BiasEdit**, a lightweight model editing approach to debias stereotyped language models using editor hyper-networks, as illustrated in Figure 1. BiasEdit aims to calibrate a language model's biased behavior to assign the same likelihoods to the stereotyped contexts and their corresponding anti-stereotyped contexts. Inspired by Mitchell et al. (2022) and Tan et al. (2023), BiasEdit uses editor networks to modify a small portion of model parameters relating to stereotyped bias and then obtain an off-the-shelf unbiased model for downstream applications. A debiasing loss in BiasEdit is designed to teach editor networks how to generate parameter shifts to modify partial parameters of language models for debiasing. BiasEdit also contains a retention loss to avoid affecting unrelated associations during editing to preserve language modeling abilities. To demonstrate the effectiveness and robustness of BiasEdit, we conduct experiments on the StereoSet (Nadeem et al., 2021) and Crows-Pairs (Nangia et al., 2020) datasets with four different LMs compared to previous debiasing methods. The results show that BiasEdit achieves the best performance on debiasing than all baselines and has little impact on LMs' language modeling and general abilities (§4.2). Meanwhile, BiasEdit is robust to gender reversal (§4.5) and semantic generality (§4.6).

Furthermore, we explore bias associations among various modules and the process of debiasing via model editing on different components of language models. We find that bias editing on upper blocks of language models has fewer negative impacts on language modeling abilities than editing on the bottom blocks, shedding light on future debiasing research.

2 Background and Setting

2.1 Debiasing Task

A stereotyped language model exhibits biased representations characterized by stereotypical beliefs and attitudes towards different demographic groups in society (Devine, 1989; Nangia et al., 2020; Bauer et al., 2023). In this paper, we study mitigating bias in stereotyped LMs while retaining their original language modeling abilities via model editing.

To be specific, there is a context x with a blank, e.g., "Girls tend to be more ___ than boys." as shown in Figure 1. We expect that an ideal unbiased language model will estimate the stereotypical context x_{stereo} and its corresponding anti-stereotypical context x_{anti} with the same probability. When two attribute terms that correspond to *stereotypical* and *anti-stereotypical* associations, e.g., 'soft' and 'determined', fill in the blank within x, x_{stereo} and x_{anti} are formed respectively, as:

x_{stereo}: Girls tend to be more <u>soft</u> than boys.
x_{anti}: Girls tend to be more <u>determined</u> than boys.

Given a biased language model with parameters θ, the optimization target of the debiasing task is to minimize the probability difference between the stereotypical context $P_\theta(x_{\text{stereo}})$ and the corresponding anti-stereotypical context $P_\theta(x_{\text{anti}})$. $P_\theta(x)$ refers to the average log probability of all tokens in x for current decoder-only language models, following Nadeem et al. (2021). Furthermore, to ensure that language modeling abilities are not influenced or even hurt during debiasing (Meade et al., 2022; Ma et al., 2023b; Chintam et al., 2023), the probability $P_\theta(x_{\text{mless}})$ of the meaningless context towards x is desired to be unchanged in the debiasing process, where a semantically unrelated attribute term exists in x_{mless}:

x_{mless}: Girls tend to be more <u>fish</u> than boys.

We use two bias benchmark dataset, StereoSet (Nadeem et al., 2021)[3] $\mathcal{S}$ and Crows-Pairs (Nangia et al., 2020) in this paper. For each instance $s \in \mathcal{S}$,

[3] Following Meade et al. (2022); Yu et al. (2023), we utilize only the *intrasentence* portion in StereoSet, which generally adapts to the debiasing task and various language models.

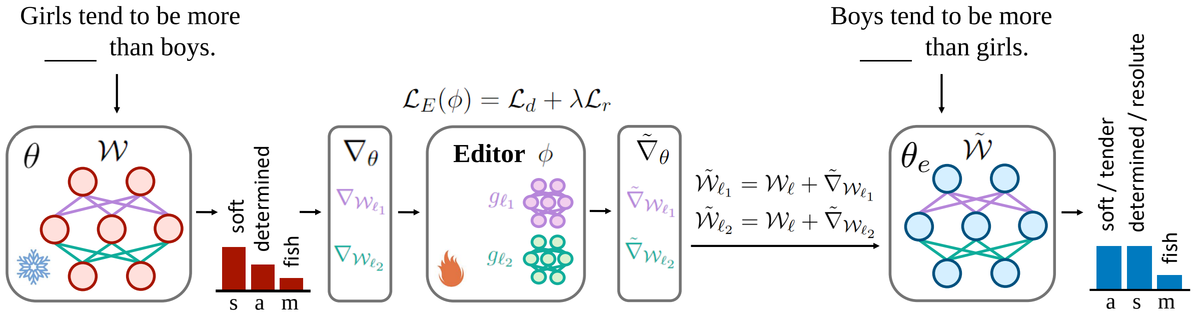

Girls tend to be more
____ than boys.
θ
$\mathcal{W}$
soft
determined
fish
s a m
∇_θ
$\nabla_{\mathcal{W}_{\ell_1}}$
$\nabla_{\mathcal{W}_{\ell_2}}$
$\mathcal{L}_E(\phi) = \mathcal{L}_d + \lambda \mathcal{L}_r$
Editor ϕ
g_{ℓ_1}
g_{ℓ_2}
$\tilde{\nabla}_\theta$
$\tilde{\nabla}_{\mathcal{W}_{\ell_1}}$
$\tilde{\nabla}_{\mathcal{W}_{\ell_2}}$
$\tilde{\mathcal{W}}_{\ell_1} = \mathcal{W}_\ell + \tilde{\nabla}_{\mathcal{W}_{\ell_1}}$
$\tilde{\mathcal{W}}_{\ell_2} = \mathcal{W}_\ell + \tilde{\nabla}_{\mathcal{W}_{\ell_2}}$
Boys tend to be more
____ than girls.
θ_e
$\tilde{\mathcal{W}}$
soft / tender
determined / resolute
fish
a s m

Figure 2: Debiasing a language model with BIASEDIT. Editor networks ϕ are trained to produce edit shifts on partial parameters $\mathcal{W}$ of a language model while its parameters θ are frozen. After editing, an unbiased LM is obtained with the robustness of gender reversal and semantic generality. $\mathcal{L}_d$ and $\mathcal{L}_r$ refer to Equation 1 and 2 respectively. s: stereotyped. a: anti-stereotyped. m: meaningless.

$s = \{x, x_{\text{stereo}}, x_{\text{anti}}, x_{\text{mless}}\}$. More descriptions about datasets are in §4.1.

2.2 Model Editing

Model editing is initially proposed to correct model mistakes (Sinitsin et al., 2020). It is now mainly applied to change knowledge in language models (Yao et al., 2023), such as knowledge modification (Cao et al., 2021), insertion (Zhang et al., 2024), and erase (Wang et al., 2024b) with locality (keeping accurate on irrelevant facts) and generality (editing neighboring facts without specific training). Precisely, a language model with parameters θ is a differentiable function $f_\theta : \mathcal{X} \times \Theta \to \mathcal{Y}$, which maps an input x to an output y. An edit target (x_e, y_e) describes a desired knowledge alteration where x_e is a trigger input to elicit the fact in language models and y_e is the target output. Model editing updates an initial model f_θ such that $f_\theta(x_e) \neq y_e$ into a model f_{θ_e} with a new set of parameters θ_e, where $f_{\theta_e}(x_e) = y_e$ according to the edit target. For example, given a query '*Who is the principal conductor of the Berlin Philharmoniker?*', the initial model outputs '*Simon Rattle*'. With an edit target (*The principal conductor of the Berlin Philharmoniker is, Kirill Petrenko*), the post-edit model will output '*Kirill Petrenko*' given a query '*Who is the principal conductor affiliated with the Berlin Philharmonic?*'. Meanwhile, both the post-edit model and the initial model will give the same answer '*1882*' to the question '*In which year was the Berlin Philharmonic founded?*'. Different from knowledge editing that only increases the probability of the target fact or only decreases the probability of the fact desired to be erased, the editing goal of debiasing is to reduce the probability of stereotyped contexts and increase the probability of their corresponding anti-stereotyped contexts simultaneously, which is much more challenging.

3 BIASEDIT

To conduct effective and efficient debiasing, we propose **BIASEDIT**, a model editing method for debiasing stereotyped language models. According to §2.2, given a language model with parameters θ, bias editing can be denoted as a function $\mathcal{X} \times \mathcal{L} \times \Theta \times \Phi \to \Theta$, which maps a paired input $(x_{\text{stereo}}, x_{\text{anti}})$, a debiasing loss function $\mathcal{L}_d : \mathcal{X} \times \Theta \to \mathbb{R}$, biased language model parameters θ, and editor parameters ϕ to new unbiased model parameters θ_e. As shown in Figure 2, BIASEDIT utilizes lightweight networks as editors ϕ to generate a parameter shift, which is used to modify models' partial weights $\mathcal{W}$ (e.g., the weights of the last linear layer in the MLPs at the last 3 blocks) for conducting debiasing edits, following the architecture of MEND (Mitchell et al., 2022) and MALMEN (Tan et al., 2023). Specifically, $(x_{\text{stereo}}, x_{\text{anti}})$ is used to compute the input to an editor network g_{ϕ_ℓ} for the layer ℓ, the gradient $\nabla_{\mathcal{W}_\ell} \mathcal{L}_d(x_{\text{stereo}}, x_{\text{anti}}, \theta)$. The output of g_{ϕ_ℓ} is the parameter shift $\tilde{\nabla}_{\mathcal{W}_\ell}$ to update $\mathcal{W}_\ell$ into $\tilde{\mathcal{W}}_\ell = \mathcal{W}_\ell + \tilde{\nabla}_{\mathcal{W}_\ell}$. BIASEDIT uses a debiasing training set $\mathcal{S}_{\text{edit}}^{\text{train}}$ and a development set $\mathcal{S}_{\text{edit}}^{\text{dev}}$ to learn editor parameters ϕ. During training, the debiasing loss $\mathcal{L}_d$ teaches editor networks how to produce parameter shifts to change $\mathcal{W}$ for eliminating bias:

$$\begin{aligned}\mathcal{L}_d = &\, \mathrm{KL}(P_{\theta_{\tilde{\mathcal{W}}}}(x_{\text{stereo}}) \| P_{\theta_{\tilde{\mathcal{W}}}}(x_{\text{anti}})) \\ &+ \mathrm{KL}(P_{\theta_{\tilde{\mathcal{W}}}}(x_{\text{anti}}) \| P_{\theta_{\tilde{\mathcal{W}}}}(x_{\text{stereo}}))\end{aligned} \quad (1)$$

where $\theta_{\mathcal{W}}$ and $\theta_{\tilde{\mathcal{W}}}$ denote the model parameters with pre-edit weights and post-edit weights, respectively. We design a symmetric $\mathcal{L}_d$ as the sum of two KL divergence losses because debiasing aims to make a language model equally treat the stereotypical contexts and anti-stereotypical contexts for fairness according to Section 2.1, which is different from knowledge editing. Moreover, to avoid negative effects on the language modeling abilities, a **retention loss** is designed to keep the probability of meaningless terms unchangeable during editing:

$$\mathcal{L}_r = \mathrm{KL}(P_{\theta_{\mathcal{W}}}(x_{\text{mless}}) \| P_{\theta_{\tilde{\mathcal{W}}}}(x_{\text{mless}})) \qquad (2)$$

Overall, the total editing loss for training editor networks is $\mathcal{L}_E(\phi) = \mathcal{L}_d + \lambda \mathcal{L}_r$. For evaluation, bias editors produce debiasing edits on a test set $\mathcal{S}_{\text{edit}}^{\text{test}}$. Because the effectiveness of instance-editing that uses one instance in each editing operation is limited (Cao et al., 2021; Meng et al., 2022, 2023; Ma et al., 2023a; Gu et al., 2024), BIASEDIT adopts batch-editing, which uses one-batch samples in one edit for the debiasing scenario. During both training and testing, the same batch size is used for optimal debiasing performance.

4 Experiments

4.1 Setups

Evaluation Metrics. Our goal of an ideal debiasing method is that it excels in mitigating stereotypical bias in LMs while not having negative effects on LMs' original language modeling and general capabilities. To measure the stereotypical bias of LMs, Stereotype Score (*SS*) (Nadeem et al., 2021) is employed. It is the percentage of samples in which a model prefers stereotypical contexts to anti-stereotypical contexts:

$$SS(\theta) = \mathbb{E}_{s \in \mathcal{S}_{\text{edit}}^{test}} \mathbb{1}\left[P_\theta(x_{\text{stereo}}) > P_\theta(x_{\text{anti}})\right]$$

An unbiased model is expected to have a *SS* of 50%. As for language modeling and general capabilities, we use the Language Modeling Score (*LMS*) from StereoSet. It is the percentage of samples in which a model ranks meaningful associations over meaningless associations.

$$LMS(\theta) = \frac{1}{2}\mathbb{E}_{s \in \mathcal{S}_{\text{edit}}^{test}} \mathbb{1}\left[P_\theta(x_{\text{stereo}}) > P_\theta(x_{\text{mless}})\right] + \frac{1}{2}\mathbb{E}_{s \in \mathcal{S}_{\text{edit}}^{test}} \mathbb{1}\left[P_\theta(x_{\text{anti}}) > P_\theta(x_{\text{mless}})\right]$$

We compute the average *SS* and *LMS* for pre-edit models and post-edit models ($SS_{\text{pre-avg}}$, $SS_{\text{post-avg}}$, $LMS_{\text{pre-avg}}$, $LMS_{\text{post-avg}}$) of all batch edits. An ideal debiasing will not change the *LMS* before and after debiasing. We report $SS_{\text{pre-avg}}$, $SS_{\text{post-avg}}$, and $\Delta LMS = LMS_{\text{post-avg}} - LMS_{\text{pre-avg}}$.

Dataset. We utilize two bias benchmark datasets, StereoSet (Nadeem et al., 2021) and Crows-Pairs (Nangia et al., 2020). There are three reasons to choose them. First, StereoSet and Crows-Pairs are widely used (Liang et al., 2021; Meade et al., 2022; Smith et al., 2022; Joniak and Aizawa, 2022; Limisiewicz et al., 2024; Omrani et al., 2023; Ma et al., 2023b; Xie and Lukasiewicz, 2023; Yu et al., 2023; Yang et al., 2023). In addition, they cover various types of bias in models, including gender, race, and religion bias, which are evaluated in our paper. Moreover, the meaningless attribute terms in StereoSet can be applied to retain language modeling abilities during debiasing. As for StereoSet, we stochastically split in the test set (3,526 samples) of the *intrasentence* StereoSet by 8:1 as $\mathcal{S}_{\text{edit}}^{\text{train}}$ and $\mathcal{S}_{\text{edit}}^{\text{dev}}$ respectively and use the development set (1,292 samples) as $\mathcal{S}_{\text{edit}}^{\text{test}}$, where attribute terms in $\mathcal{S}_{\text{edit}}^{\text{train}}$ and $\mathcal{S}_{\text{edit}}^{\text{dev}}$ are **disjoint** from $\mathcal{S}_{\text{edit}}^{\text{test}}$. Crows-Pairs is also used as $\mathcal{S}_{\text{edit}}^{\text{test}}$ to evaluate BIASEDIT's debiasing performance (details in Appendix B). We also select three large language model benchmark datasets, OpenBookQA (Mihaylov et al., 2018), BoolQ (Clark et al., 2019), and COPA (Roemmele et al., 2011), to evaluate LMs' capabilities of reading comprehension, knowledge question-answering, and commonsense reasoning, respectively. Their evaluations are conducted by OpenCompass tool (Contributors, 2023) and measured by accuracy based on perplexity.

Comparison. Compared with BIASEDIT, four distinguishing baseline debiasing methods from Meade et al. (2022) are implemented[4]: counterfactual data augmentation (CDA) (Zmigrod et al., 2019), SentenceDebias (Liang et al., 2020), Self-Debias (Schick et al., 2021), and iterative nullspace projection (INLP) (Ravfogel et al., 2020) (details in Appendix B.3). Unlike all baselines, our editor networks can be trained with a mixture of all three types of bias, instead of dealing with only one particular bias at a time. As for testing, BIASEDIT is evaluated on gender, race, and religion bias samples from $\mathcal{S}_{\text{edit}}^{\text{test}}$ separately. BIASEDIT is a **model-agnostic** debiasing method and can be applied to any open-sourced language model. We conduct experiments on diverse language models, including GPT2 (Radford et al., 2019), Gemma (Mesnard et al., 2024), Llama3 (Meta, 2024), and Mistral (Jiang et al., 2023). Some blocks in LMs are selected in this paper according to preliminary

[4] https://github.com/McGill-NLP/bias-bench

Method	GPT2-medium						Gemma-2b					
	SS (%) → 50%			ΔLMS (%) → 0			SS (%) → 50%			ΔLMS (%) → 0		
	Gender	Race	Religion	Gender	Race	Religion	Gender	Race	Religion	Gender	Race	Religion
Pre-edit	65.58	61.63	62.57	93.39	92.30	90.46	69.25	64.21	62.39	94.57	94.26	93.43
CDA	63.29	61.36	61.79	**-0.21**	-3.02	**0.00**	-					
SentenceDebias	67.99	58.97	56.64	+0.29	+1.52	+0.34	68.86	63.87	60.09	**-2.65**	-0.31	**-0.58**
Self-Debias	60.28	57.29	57.61	-3.47	-4.12	-1.35	65.70	58.29	58.02	-35.93	-30.39	-21.69
INLP	63.17	60.00	58.57	-5.15	**-1.49**	-2.48	52.17	62.96	58.57	-12.50	**-0.30**	-2.01
BIASEDIT	**49.42**	**56.34**	**53.55**	-8.82	-5.12	-1.92	**48.59**	**55.86**	**47.36**	-4.78	-4.35	-5.44
Method	Mistral-7B-v0.3						Llama3-8B					
	SS (%) → 50%			ΔLMS (%) → 0			SS (%) → 50%			ΔLMS (%) → 0		
	Gender	Race	Religion	Gender	Race	Religion	Gender	Race	Religion	Gender	Race	Religion
Pre-edit	70.19	64.97	56.09	93.60	89.77	88.85	72.25	65.01	60.87	95.81	92.47	91.33
CDA	-						-					
SentenceDebias	68.36	64.54	54.94	-0.61	0.62	+0.09	68.55	64.97	59.91	**-0.22**	-1.14	-0.66
Self-Debias	61.79	**50.54**	60.68	-39.28	-29.17	-32.37	65.46	60.88	58.57	-40.04	-2.54	-28.64
INLP	69.22	65.23	55.90	**+0.35**	**-0.15**	-0.58	68.17	65.22	62.21	-1.43	**-0.09**	**0.00**
BIASEDIT	**46.24**	51.46	**50.42**	-8.81	-8.59	**-0.03**	**49.18**	**53.51**	**51.13**	-13.42	-11.77	-10.02

Table 1: Performance of BIASEDIT compared to previous debiasing baselines. **Pre-edit**: $SS_{\text{pre-avg}}$ and $LMS_{\text{pre-avg}}$. $SS_{\text{post-avg}}$ and $\Delta LMS = LMS_{\text{post-avg}} - LMS_{\text{pre-avg}}$ are reported for all baselines and BIASEDIT.

Dataset	Model							
	Llama3$_{\text{pre}}$	**Llama3$_{\text{post}}$**	**Mistral$_{\text{pre}}$**	**Mistral$_{\text{post}}$**	**Gemma$_{\text{pre}}$**	**Gemma$_{\text{post}}$**	**GPT2m$_{\text{pre}}$**	**GPT2m$_{\text{post}}$**
OpenBookQA	80.80	78.94	84.20	82.90	46.80	46.48	40.40	40.57
BoolQ	70.00	65.18	64.25	62.89	62.00	61.85	55.00	55.40
COPA	68.00	67.90	78.00	77.80	62.00	61.09	24.80	24.68

Table 2: Accuracies (%) of general model benchmarks. 'pre': pre-edit, 'post-': post-edit, 'GPT2m': 'GP2-medium'

experiments described in Section 4.4. The last linear layer in the MLP at each block is edited. We report the best debiasing performance among different edited components in Table 1 (the last 3 blocks for GPT2-medium and Mistral-7B-v0.3, the last 2 blocks for Llama3-8B, and the penultimate block for Gemma-2b).

4.2 Main Results

BIASEDIT achieves the best debiasing performance on all bias types compared to all debiasing baselines. According to the *SS*, BIASEDIT can reduce *SS* to less than 57% and more than 46% while *SS* of debiased models with previous debiasing baselines are mostly above 60%, which demonstrates BIASEDIT leads to significant improvement for debiasing performance. For instance, as for the *SS* of Llama3, BIASEDIT yields an improvement of ↑13.26, ↑7.37, and ↑7.44 on the absolute difference from 50% for gender, race, and religion bias respectively, compared with the best *SS* among all baselines. According to Templeton et al. (2024), human-interpretable concepts, like bias, can match neuron activations. We suppose that the reason for the excellent debiasing performance of BIASEDIT is that parameters associated with bias are explicitly edited, which is illustrated in Section 4.4 and Appendix A. Moreover, BIASEDIT presents excellent performance on every bias type though editor networks are trained to produce edits on a mixture of different types of bias at a time (Appendix B.4). It is illustrated that our method can generalize debiasing success over various bias types, compared to previous debiasing methods that can only deal with one particular bias at a time, such as creating a bias subspace (SentenceBias) or training an adapter (Limisiewicz et al., 2024) for only one bias type.

BIASEDIT is efficient to produce off-the-shelf unbiased models. Fully finetuning LMs with CDA usually requires many computational resources and time. Subspace computation for SentenceDebias and INLP is also time-consuming, especially for LLMs. For example, computing the gender bias subspace for Mistral-7B takes more than 2 days. Unlike them, BIASEDIT only trains a small hyper-network with a minimal memory cost based on Tan et al. (2023) due to decomposition between the hyper-network and LM. For instance, only one A800GPU is used for bias editing on Mistral-7B or Llama-8B with arbitrary edit batch size. Training small gender editor networks for Mistral-7B only takes about 5 hours. Additionally, compared to prompting and representation projections baselines like SentenceDebias and INLP that can only calibrate models' output distributions instead of language models themselves, BiasEdit produces off-the-shelf debiased language models.

BiasEdit has little to no impact on language modeling abilities, illustrating the effectiveness of the retention loss. The results of *LMS* drops show that BiasEdit exhibit a few negative impacts on models' language modeling capabilities. Comparing *SS* of original models and *LMS* drops of debiasing, the *LMS* drop for debiasing is consistent with the bias extent of the original model in most cases. The more biased the model is, the greater the impact of editing for debiasing is. For example, models in Table 1 are more biased on gender than race according to *SS* while *LMS* drops of gender debiasing are larger than race debiasing in most cases, which indicates that bias editing is more difficult for more biased models. Therefore, our retention loss is necessary. Meanwhile, we surmise that $\mathcal{L}_r$ (Equation 2) works well based on the comparative results of *LMS* drops with that of baselines. The ablation study in §4.3 illustrates this. We also explore the impact of BiasEdit on general NLP tasks since previous works (Gu et al., 2024; Gupta et al., 2024) have indicated that model editing can hurt the general capabilities of language models. As for the debiased models, we randomly sample checkpoints of two editing batches for gender, race, and religion bias, respectively. The average accuracies of these six debiased results are shown in Table 2. There are only a few accuracy drops after debiasing, which illustrates that BiasEdit can do little harm to the general capabilities of language models during editing for debiasing.

4.3 Ablation Study on retention loss $\mathcal{L}_r$

We perform an ablation study to show the effectiveness of the retention loss $\mathcal{L}_r$ for maintaining language modeling abilities during debiasing. The results for training editor networks with and without $\mathcal{L}_r$ are shown in Table 3. There are large drops on *LMS* if the retention loss is not deployed during editing. Specifically, the *LMS* drops of Gemma-2b increase absolutely by ↓24.53, ↓23.58, and ↓56.85 for gender, race, and religion bias respectively during debiasing without $\mathcal{L}_r$, which illustrates that the retention loss plays an important role in reducing harm to the language modeling abilities during editing.

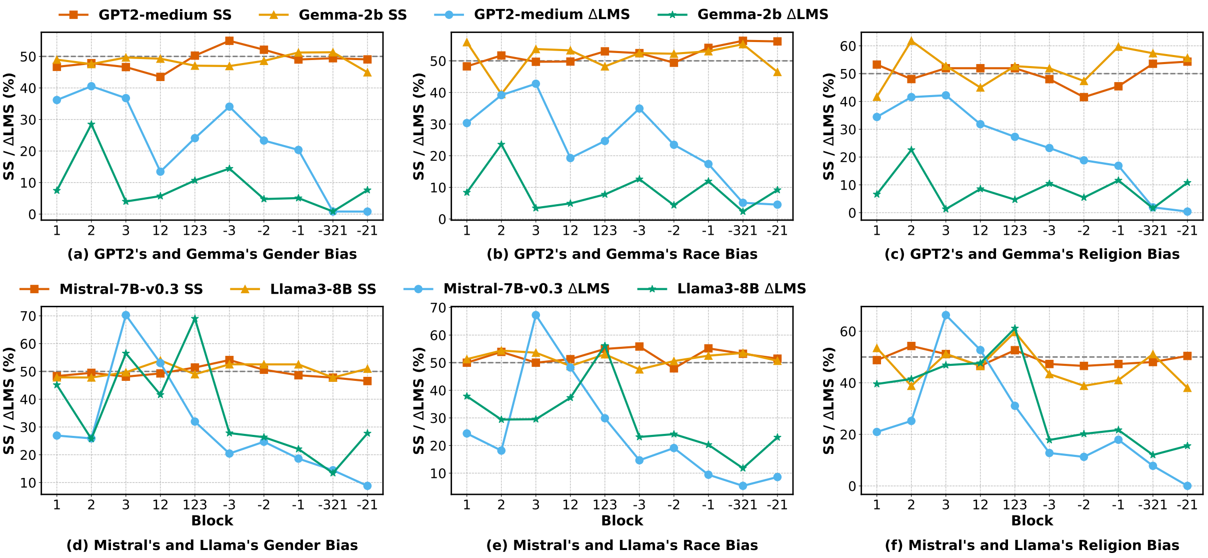
	GPT2-medium					
Method	**SS** (%)			**ΔLMS** (%)		
	gender	race	religion	gender	race	religion
w/o $\mathcal{L}_r$	52.55	56.45	45.73	-52.36	-59.96	-61.54
w $\mathcal{L}_r$	49.42	56.34	53.55	-8.82	-5.12	-1.92
	Gemma-2b					
Method	**SS** (%)			**ΔLMS** (%)		
	gender	race	religion	gender	race	religion
w/o $\mathcal{L}_r$	50.81	52.05	41.17	-29.31	-27.93	-62.29
w $\mathcal{L}_r$	48.59	52.25	47.36	-4.78	-4.35	-5.44

Table 3: BiasEdit w and w/o the retention loss $\mathcal{L}_r$.

4.4 Further Discussion on Editing Different Components for Debiasing

To pursue optimal performance, it is necessary to determine which blocks to be edited at first. Before embarking on our main experimental investigation, preliminary experiments are conducted to explore bias associations in language models. Following causal tracing from Meng et al. (2022), we propose bias tracing to track bias associations in language models, which is described in Appendix A. It is observed that MLPs in several bottom and upper blocks exert a substantial influence on bias captured in language models. Some existing works also demonstrate that editing MLPs can modify knowledge associations in language models (Geva et al., 2021; Mitchell et al., 2022; Meng et al., 2022, 2023; Gupta et al., 2023; Wu et al., 2023a). Based on our findings and previous works, BiasEdit edits the last (output) layer in the MLP at each block for the debiasing task. To comprehensively explore the effects of debiasing stereotyped language models via model editing, we choose the first 3 and last 3 blocks of language models to be edited with BiasEdit. The resulting debiasing performance and modeling capabilities are measured in this section. The *SS* and *LMS* drops of debiased language models are shown in Figure 3.

Edits on the upper blocks have less negative impacts on modeling abilities than edits on the bottom blocks. According to Figure 3, the *LMS* drops are much more for the bottom blocks than the last blocks, especially for Mistral and Llama3. This indicates that determining the suitable editing



(a) GPT2's and Gemma's Gender Bias

(b) GPT2's and Gemma's Race Bias

(c) GPT2's and Gemma's Religion Bias

(d) Mistral's and Llama's Gender Bias

(e) Mistral's and Llama's Race Bias

(f) Mistral's and Llama's Religion Bias

Figure 3: *SS* (%) and Δ*LMS* (%) of debiased language models after editing the last layer in the MLP of different blocks. 1/2/3: the first/second/third block. 12: the first 2 blocks. 123: the first 3 blocks. -1/-2/-3, the last/penultimate/antepenultimate block, -321: the last 3 blocks. -21: the last 2 blocks.

components for debiasing is important and modifying weights of some upper blocks is appropriate for debiasing. We think the reason might be that the bottom layers capture basic linguistic features like syntax and common word associations while the upper blocks delve into deeper semantic relationships, contextual understanding, and high-level language features (Geva et al., 2021). Since biases manifest in semantic associations, lightweight modification of the upper layers can work well for bias calibration, which will do little harm to modeling abilities. On the contrary, the effects of editing on linguistic patterns of bias, like the co-occurrence of bias attribute words and attribute terms, represented in the bottom blocks will be propagated and potentially amplified through the network as information passes through subsequent blocks (Merullo et al., 2023). Therefore, bias editing on the bottom layers may harm the semantic associations encoded in the upper blocks.

4.5 Reversing Gender Attribute Words

Inspired by the reversal curse that large language models trained on 'A is B' fail to learn 'B is A' (Berglund et al., 2023), we think a robust gender debiasing method should be able to calibrate a model's treatment to the two gender polarities, male and female, equally. For instance, there are two sentences "Girls tend to be more ___ than boys." and "Boys tend to be more ___ than girls.". A debiased model is expected to model the stereotypical term "soft" and the anti-stereotypical term "determined" in both two sentences equivalently though only the first sentence is used for training. To evaluate this gender robustness, a gender counterfactual test set $\mathcal{S}^{\text{test}}_{\text{gender*}}$ is created (Appendix C). We reverse all gender attribute words in the gender bias samples from $\mathcal{S}^{\text{test}}_{\text{edit}}$ to construct the set. For instance, "boys", "father", and "Female" are changed into "girls", "mother", and "Male" respectively. Then the test set is used to examine the gender robustness of BiasEdit, the implementation of which is the same as Table 1. The results in Figure 4 show that BiasEdit is robust enough to remove gender counterfactual bias.

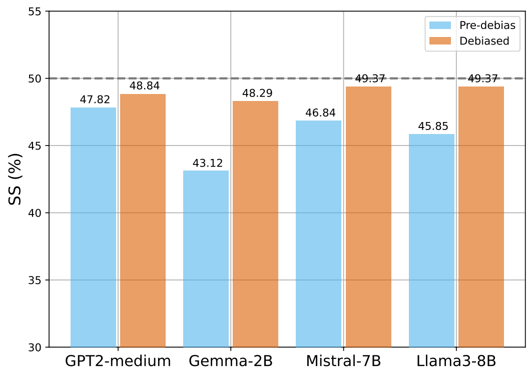


Figure 4: Gender Reversal Robustness. *Pre-debias* refers to *SS* of pre-trained language models on the gender reversal test set before debiasing. *Debiased* refers to *SS* of debiased models by BiasEdit.

4.6 Semantic Generality

Model / SS (%)	Pre-debias			BiasEdit		
	Gender	Race	Religion	Gender	Race	Religion
GPT2-medium	52.53	53.71	64.30	52.53	48.53	55.82
Gemma-2B	51.79	54.39	58.89	51.84	50.29	54.76
Mistral-7B-v0.3	48.20	52.92	53.54	58.17	49.46	58.17
Llama3-8B	45.37	58.79	58.17	49.19	53.51	51.14

Table 4: *SS* (%) on the synonym-augmented test set.

Similar to the generality principle of knowledge editing, a robust debiasing method should ensure the debiased language model demonstrates unbiased behavior on a group of semantically similar attribute terms without specific training, showcasing its adaptability to the nuanced and dynamic nature of language. To evaluate this robustness of BiasEdit, we curate a synonym-augmented test set that substitutes attribute terms in $\mathcal{S}_{\text{edit}}^{\text{test}}$ with their synonyms generated by WordNet (Miller, 1995) using NLTK (Bird and Loper, 2004). Results in Table 4 show that our debiasing method can generally remove bias in the language models' neighboring semantic modeling space in most cases.

5 Related Work

Bias and Debiasing Many works focus on measuring bias in language models (Zhao et al., 2020; Nangia et al., 2020; Nadeem et al., 2021; Li et al., 2022b; Faisal and Anastasopoulos, 2022; Cao et al., 2022; Wan et al., 2023; Vashishtha et al., 2023), which provide bias measurement metrics (Hovy and Prabhumoye, 2021; Goldfarb-Tarrant et al., 2023). To mitigate bias, researchers propose various debiasing methods (Meade et al., 2022; Gallegos et al., 2023). The basic method is to fully finetune language models on counterfactual data (Lu et al., 2020; Zmigrod et al., 2019), which is costly. So other approaches adopt fine-tuning in an efficient way (Gira et al., 2022; Yang et al., 2023; Xie and Lukasiewicz, 2023). Except for fine-tuning, prompting (Schick et al., 2021; Guo et al., 2022) guides models to calibrate their bias. Representation projection (Liang et al., 2020; Ravfogel et al., 2020) is employed to remove bias representation out of models, which, however, cannot change the language models' internal bias in essence without modifying parameters. Some works (Kumar et al., 2023; Limisiewicz et al., 2024) construct an adapter for each type of bias and plug it into a LM. If we want to mitigate N types of bias, N adapters will be trained, which is not efficient. Recently, an empirical study (Yan et al., 2024) has explored the feasibility of debiasing via model editing. Therefore, we adopt model editing by efficiently editing partial parameters for debiasing LMs.

Model Editing Much factual knowledge is memorized in language models (Petroni et al., 2019; Shin et al., 2020; Jiang et al., 2020; Li et al., 2022a; Hase et al., 2023). As the real world develops, some facts become obsolete and different over time. It is necessary to change, add, or erase facts stored in existing pre-trained language models (Li et al., 2022a; Hase et al., 2023). Model editing (Sinitsin et al., 2020) is come up with to modify information in PLMs. Editing should follow some properties (Yao et al., 2023): reliability (predicting updated facts), locality, generality, and efficiency (efficient in runtime and memory). The direct but inefficient editing is to fully finetune a model on new facts (Zhu et al., 2020). For locality, many works (Dai et al., 2022; Meng et al., 2022, 2023; Ma et al., 2023a; Fang et al., 2024; Jiang et al., 2025) seek the model parameters strongly related to the facts and then edit these localized hidden states. With high efficiency, Mitchell et al. (2022); Tan et al. (2023) achieve fast editing by training specific editor networks. Also, lifelong model editing, like WISE (Wang et al., 2024a), is paid attention to for practical applications. Recently, model editing has been applied to unlearn information from language models (Patil et al., 2023; Ishibashi and Shimodaira, 2023; Yu et al., 2023; Wang et al., 2024b). Inspired by them, we propose an efficient bias editing method, BiasEdit, to eliminate bias in language models while preserving the language modeling capabilities and generalizing gender reversal inputs and semantically related inputs.

6 Conclusion

We propose **BiasEdit**, an efficient model editing method to debias stereotyped language models by modifying a small portion of language models' parameters with small editor networks. We design a debiasing loss $\mathcal{L}_d$ for debiasing and a retention loss $\mathcal{L}_r$ to maintain the language modeling abilities during editing. Experiments illustrate that BiasEdit presents much better debiasing performance than classical debiasing methods and gives little to no harmful impact on language modeling and general capabilities. Also, BiasEdit is robust in gender reversal and semantic generality. Meanwhile, we comprehensively investigate the effects of debiasing different components of language models.

Limitations

BiasEdit is only evaluated on sentence-level bias modeling examples with gold labels. However, in the LLM era, we expect bias mitigation for text generation forms, such as QA and text continuation, which is more appropriate for current chat-based large language models. Furthermore, biased datasets for text generation, like BBQ (Parrish et al., 2022), with gold labels are extremely lacking. Therefore, we hope that BiasEdit and other adapt model editing / unlearning methods can be adapted to mitigate bias for text generation, and such datasets will be constructed in the future.

Ethics Statement

This work hopes to encourage more research for debiasing language models. We use open-source pre-trained language models from *HuggingFace* (Wolf et al., 2019). All datasets and codes in the experiments are publicly available. We ensure that no private information is in our research. Furthermore, we recognize the potential societal impacts of our work that BiasEdit can be immorally used to make language models more biased, which is harmful to society. We advocate for the responsible use of our method in ways that benefit the whole society and minimize harm.

Acknowledgements

This work was done during Xin Xu's internship mentored by Prof. Wei Xu. Thanks for the code suggestions from Chenmien Tan, one author of MALMEN (Tan et al., 2023).

References

H. Abdi and L. J. Williams. 2010. Principal component analysis. *WIREs Computational Statistics*, 2:433–459.

Soumya Barikeri, Anne Lauscher, Ivan Vulić, and Goran Glavaš. 2021. RedditBias: A real-world resource for bias evaluation and debiasing of conversational language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1941–1955, Online. Association for Computational Linguistics.

Lisa Bauer, Hanna Tischer, and Mohit Bansal. 2023. Social commonsense for explanation and cultural bias discovery. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 3727–3742. Association for Computational Linguistics.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. The reversal curse: Llms trained on "a is b" fail to learn "b is a". *CoRR*, abs/2309.12288.

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6491–6506. Association for Computational Linguistics.

Yang Trista Cao, Anna Sotnikova, Hal Daumé III, Rachel Rudinger, and Linda Zou. 2022. Theory-grounded measurement of U.S. social stereotypes in english language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1276–1295. Association for Computational Linguistics.

Ruizhe Chen, Yichen Li, Zikai Xiao, and Zuozhu Liu. 2024. Large language model bias mitigation from the perspective of knowledge editing. *CoRR*, abs/2405.09341.

Myra Cheng, Esin Durmus, and Dan Jurafsky. 2023a. Marked personas: Using natural language prompts to measure stereotypes in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1504–1532. Association for Computational Linguistics.

Siyuan Cheng, Ningyu Zhang, Bozhong Tian, Zelin Dai, Feiyu Xiong, Wei Guo, and Huajun Chen. 2023b. Editing language model-based knowledge graph embeddings. *CoRR*, abs/2301.10405.

Abhijith Chintam, Rahel Beloch, Willem Zuidema, Michael Hanna, and Oskar van der Wal. 2023. Identifying and adapting transformer-components responsible for gender bias in an English language model. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 379–394, Singapore. Association for Computational Linguistics.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina

Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2924–2936. Association for Computational Linguistics.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8493–8502. Association for Computational Linguistics.

Dipto Das, Shion Guha, and Bryan Semaan. 2023. Toward cultural bias evaluation datasets: The case of Bengali gender, religious, and national identity. In *Proceedings of the First Workshop on Cross-Cultural Considerations in NLP (C3NLP)*, pages 68–83, Dubrovnik, Croatia. Association for Computational Linguistics.

Patricia G Devine. 1989. Stereotypes and prejudice: Their automatic and controlled components. *Journal of personality and social psychology*, 56(1):5.

Fahim Faisal and Antonios Anastasopoulos. 2022. Geographic and geopolitical biases of language models. *CoRR*, abs/2212.10408.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Xiang Wang, Xiangnan He, and Tat-Seng Chua. 2024. Alphaedit: Null-space constrained knowledge editing for language models. *CoRR*, abs/2410.02355.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md. Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2023. Bias and fairness in large language models: A survey. *CoRR*, abs/2309.00770.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5484–5495. Association for Computational Linguistics.

Michael Gira, Ruisu Zhang, and Kangwook Lee. 2022. Debiasing pre-trained language models via efficient fine-tuning. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, LT-EDI 2022, Dublin, Ireland, May 27, 2022*, pages 59–69. Association for Computational Linguistics.

Seraphina Goldfarb-Tarrant, Eddie Ungless, Esma Balkir, and Su Lin Blodgett. 2023. This prompt is measuring \textlessmask\textgreater: evaluating bias evaluation in language models. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2209–2225. Association for Computational Linguistics.

Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. 2024. Model editing can hurt general abilities of large language models. *CoRR*, abs/2401.04700.

Yue Guo, Yi Yang, and Ahmed Abbasi. 2022. Auto-debias: Debiasing masked language models with automated biased prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1012–1023. Association for Computational Linguistics.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. 2024. Model editing at scale leads to gradual and catastrophic forgetting. *CoRR*, abs/2401.07453.

Anshita Gupta, Debanjan Mondal, Akshay Krishna Sheshadri, Wenlong Zhao, Xiang Li, Sarah Wiegreffe, and Niket Tandon. 2023. Editing common sense in transformers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 8214–8232. Association for Computational Linguistics.

Matan Halevy, Camille Harris, Amy S. Bruckman, Diyi Yang, and Ayanna M. Howard. 2021. Mitigating racial biases in toxic language detection with an equity-based ensemble framework. In *EAAMO 2021: ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization, Virtual Event, USA, October 5 - 9, 2021*, pages 7:1–7:11. ACM.

Peter Hase, Mona T. Diab, Asli Celikyilmaz, Xian Li, Zornitsa Kozareva, Veselin Stoyanov, Mohit Bansal, and Srinivasan Iyer. 2023. Methods for measuring, updating, and visualizing factual beliefs in language models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 2706–2723. Association for Computational Linguistics.

Dirk Hovy and Shrimai Prabhumoye. 2021. Five sources of bias in natural language processing. *Lang. Linguistics Compass*, 15(8).

Yoichi Ishibashi and Hidetoshi Shimodaira. 2023. Knowledge sanitization of large language models. *CoRR*, abs/2309.11852.

Shadi Iskander, Kira Radinsky, and Yonatan Belinkov. 2023. Shielded representations: Protecting sensitive attributes through iterative gradient-based projection. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14,* *2023*, pages 5961–5977. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Houcheng Jiang, Junfeng Fang, Ningyu Zhang, Guojun Ma, Mingyang Wan, Xiang Wang, Xiangnan He, and Tat-seng Chua. 2025. Anyedit: Edit any knowledge encoded in language models. *arXiv preprint arXiv:2502.05628*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know. *Trans. Assoc. Comput. Linguistics*, 8:423–438.

Przemyslaw K. Joniak and Akiko Aizawa. 2022. Gender biases and where to find them: Exploring gender bias in pre-trained transformer-based language models using movement pruning. *CoRR*, abs/2207.02463.

Deepak Kumar, Oleg Lesota, George Zerveas, Daniel Cohen, Carsten Eickhoff, Markus Schedl, and Navid Rekabsaz. 2023. Parameter-efficient modularised bias mitigation via AdapterFusion. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2738–2751, Dubrovnik, Croatia. Association for Computational Linguistics.

Jiahang Li, Taoyu Chen, and Yuanli Wang. 2024. Trace and edit relation associations in GPT. *CoRR*, abs/2401.02976.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Zhenhua Dong, Chengjie Sun, Bingquan Liu, Zhenzhou Ji, Xin Jiang, and Qun Liu. 2022a. How pre-trained language models capture factual knowledge? A causal-inspired analysis. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1720–1732. Association for Computational Linguistics.

Yizhi Li, Ge Zhang, Bohao Yang, Chenghua Lin, Anton Ragni, Shi Wang, and Jie Fu. 2022b. HERB: measuring hierarchical regional bias in pre-trained language models. In *Findings of the Association for Computational Linguistics: AACL-IJCNLP 2022, Online only, November 20-23, 2022*, pages 334–346. Association for Computational Linguistics.

Paul Pu Liang, Irene Mengze Li, Emily Zheng, Yao Chong Lim, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2020. Towards debiasing sentence representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5502–5515. Association for Computational Linguistics.

Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. 2021. Towards understanding and mitigating social biases in language models. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 6565–6576. PMLR.

Tomasz Limisiewicz and David Marecek. 2022. Don't forget about pronouns: Removing gender bias in language models without losing factual gender information. *CoRR*, abs/2206.10744.

Tomasz Limisiewicz, David Marecek, and Tomás Musil. 2024. Debiasing algorithm through model adaptation. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Zihao Lin, Samyadeep Basu, Mohammad Beigi, Varun Manjunatha, Ryan A Rossi, Zichao Wang, Yufan Zhou, Sriram Balasubramanian, Arman Zarei, Keivan Rezaei, et al. 2025. A survey on mechanistic interpretability for multi-modal foundation models. *arXiv preprint arXiv:2502.17516*.

Yang Liu, Yuanshun Yao, Jean-Francois Ton, Xiaoying Zhang, Ruocheng Guo, Hao Cheng, Yegor Klochkov, Muhammad Faaiz Taufiq, and Hang Li. 2023. Trustworthy llms: a survey and guideline for evaluating large language models' alignment. *CoRR*, abs/2308.05374.

Kaiji Lu, Piotr Mardziel, Fangjing Wu, Preetam Amancharla, and Anupam Datta. 2020. Gender bias in neural natural language processing. In *Logic, Language, and Security - Essays Dedicated to Andre Scedrov on the Occasion of His 65th Birthday*, volume 12300 of *Lecture Notes in Computer Science*, pages 189–202. Springer.

Jun-Yu Ma, Jia-Chen Gu, Zhen-Hua Ling, Quan Liu, and Cong Liu. 2023a. Untying the reversal curse via bidirectional language model editing. *CoRR*, abs/2310.10322.

Weicheng Ma, Henry Scheible, Brian Wang, Goutham Veeramachaneni, Pratim Chowdhary, Alan Sun, Andrew Koulogeorge, Lili Wang, Diyi Yang, and Soroush Vosoughi. 2023b. Deciphering stereotypes in pre-trained language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 11328–11345. Association for Computational Linguistics.

Thomas Manzini, Lim Yao Chong, Alan W Black, and Yulia Tsvetkov. 2019. Black is to criminal as Caucasian is to police: Detecting and removing multiclass bias in word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 615–621, Minneapolis, Minnesota. Association for Computational Linguistics.

Justus Mattern, Zhijing Jin, Mrinmaya Sachan, Rada Mihalcea, and Bernhard Schölkopf. 2022. Understanding stereotypes in language models: Towards robust measurement and zero-shot debiasing. *CoRR*, abs/2212.10678.

Nicholas Meade, Elinor Poole-Dayan, and Siva Reddy. 2022. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1878–1898. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *NeurIPS*.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. 2023. Language models implement simple word2vec-style vector arithmetic. *CoRR*, abs/2305.16130.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Cristian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, and et al. 2024. Gemma: Open models based on gemini research and technology. *CoRR*, abs/2403.08295.

Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date. `https://ai.meta.com/blog/meta-llama-3/`.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2381–2391. Association for Computational Linguistics.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022. Fast model editing at scale. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. Stereoset: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 5356–5371. Association for Computational Linguistics.

Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. Crows-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1953–1967. Association for Computational Linguistics.

Shiwen Ni, Dingwei Chen, Chengming Li, Xiping Hu, Ruifeng Xu, and Min Yang. 2023. Forgetting before learning: Utilizing parametric arithmetic for knowledge updating in large language models. *CoRR*, abs/2311.08011.

Ali Omrani, Alireza Salkhordeh Ziabari, Charles Yu, Preni Golazizian, Brendan Kennedy, Mohammad Atari, Heng Ji, and Morteza Dehghani. 2023. Social-group-agnostic bias mitigation via the stereotype content model. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4123–4139. Association for Computational Linguistics.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel Bowman. 2022. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2086–2105, Dublin, Ireland. Association for Computational Linguistics.

Vaidehi Patil, Peter Hase, and Mohit Bansal. 2023. Can sensitive information be deleted from llms? objectives for defending against extraction attacks. *CoRR*, abs/2309.17410.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online. Association for Computational Linguistics.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *Logical Formalizations of Commonsense Reasoning, Papers from the 2011 AAAI Spring Symposium, Technical Report SS-11-06, Stanford, California, USA, March 21-23, 2011*. AAAI.

Timo Schick, Sahana Udupa, and Hinrich Schütze. 2021. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in NLP. *Trans. Assoc. Comput. Linguistics*, 9:1408–1424.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeff Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, Eric J. Michaud, Stephen Casper, Max Tegmark, William Saunders, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland, Daniel Murfet, and Tom McGrath. 2025. Open problems in mechanistic interpretability. *CoRR*, abs/2501.16496.

Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2020. Towards controllable biases in language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 3239–3254. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4222–4235. Association for Computational Linguistics.

Anton Sinitsin, Vsevolod Plokhotnyuk, Dmitry V. Pyrkin, Sergei Popov, and Artem Babenko. 2020. Editable neural networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Eric Michael Smith, Melissa Hall, Melanie Kambadur, Eleonora Presani, and Adina Williams. 2022. "i'm sorry to hear that": Finding new biases in language models with a holistic descriptor dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 9180–9211. Association for Computational Linguistics.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth M. Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1630–1640. Association for Computational Linguistics.

Chenmien Tan, Ge Zhang, and Jie Fu. 2023. Massive editing for large language models via meta learning. *CoRR*, abs/2311.04661.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, Alex Tamkin, Esin Durmus, Tristan Hume, Francesco Mosconi, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. `https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html`.

Aniket Vashishtha, Kabir Ahuja, and Sunayana Sitaram. 2023. On evaluating and mitigating gender biases in multilingual settings. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 307–318. Association for Computational Linguistics.

Pranav Narayanan Venkit, Sanjana Gautam, Ruchi Panchanadikar, Ting-Hao K. Huang, and Shomir Wilson. 2023. Nationality bias in text generation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 116–122. Association for Computational Linguistics.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart M. Shieber. 2020. Causal mediation analysis for interpreting neural NLP: the case of gender bias. *CoRR*, abs/2004.12265.

Yixin Wan, George Pu, Jiao Sun, Aparna Garimella, Kai-Wei Chang, and Nanyun Peng. 2023. "kelly is a warm person, joseph is a role model": Gender biases in llm-generated reference letters. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 3730–3748. Association for Computational Linguistics.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. 2024a. Wise: Rethinking the knowledge memory for lifelong model editing of large language models. *CoRR*, abs/2405.14768.

Yu Wang, Ruihan Wu, Zexue He, and Xiusi Chen. 2024b. Large scale knowledge washing. *CoRR*, abs/2405.14768.

Yifan Wei, Xiaoyan Yu, Huanhuan Ma, Fangyu Lei, Yixuan Weng, Ran Song, and Kang Liu. 2023. Assessing knowledge editing in language models via relation perspective. *CoRR*, abs/2311.09053.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Suhang Wu, Minlong Peng, Yue Chen, Jinsong Su, and Mingming Sun. 2023a. Eva-kellm: A new benchmark for evaluating knowledge editing of llms. *CoRR*, abs/2308.09954.

Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. 2023b. DEPN: detecting and editing privacy neurons in pretrained language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2875–2886. Association for Computational Linguistics.

Zhongbin Xie and Thomas Lukasiewicz. 2023. An empirical analysis of parameter-efficient methods for debiasing pre-trained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 15730–15745. Association for Computational Linguistics.

Jianhao Yan, Futing Wang, Yafu Li, and Yue Zhang. 2024. Potential and challenges of model editing for social debiasing. *CoRR*, abs/2402.13462.

Ke Yang, Charles Yu, Yi Ren Fung, Manling Li, and Heng Ji. 2023. ADEPT: A debiasing prompt framework. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 10780–10788. AAAI Press.

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 10222–10240. Association for Computational Linguistics.

Xunjian Yin, Jin Jiang, Liming Yang, and Xiaojun Wan. 2023. History matters: Temporal knowledge editing in large language model. *CoRR*, abs/2312.05497.

Charles Yu, Sullam Jeoung, Anish Kasi, Pengfei Yu, and Heng Ji. 2023. Unlearning bias in language models by partitioning gradients. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 6032–6048. Association for Computational Linguistics.

Ningyu Zhang, Yunzhi Yao, Bozhong Tian, Peng Wang, Shumin Deng, Mengru Wang, Zekun Xi, Shengyu Mao, Jintian Zhang, Yuansheng Ni, Siyuan Cheng, Ziwen Xu, Xin Xu, Jia-Chen Gu, Yong Jiang, Pengjun Xie, Fei Huang, Lei Liang, Zhiqiang Zhang, Xiaowei Zhu, Jun Zhou, and Huajun Chen. 2024. A comprehensive study of knowledge editing for large language models. *CoRR*, abs/2401.01286.

Jieyu Zhao, Subhabrata Mukherjee, Saghar Hosseini, Kai-Wei Chang, and Ahmed Hassan Awadallah. 2020. Gender bias in multilingual embeddings and cross-lingual transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2896–2907. Association for Computational Linguistics.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix X. Yu, and Sanjiv Kumar. 2020. Modifying memories in transformer models. *CoRR*, abs/2012.00363.

Ran Zmigrod, S. J. Mielke, Hanna M. Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1651–1661. Association for Computational Linguistics.

A Bias Tracing

Some works (Sharkey et al., 2025; Lin et al., 2025) use causal tracing to mechanistic interpretability for LLMs. ROME (Meng et al., 2022) and MEMIT (Meng et al., 2023) utilize causal tracing (Vig et al., 2020) to locate facts memorized causal LMs. After they find the specific hidden state with the strongest effect on individual facts, they modify these localized parameters for changing facts. Inspired by causal tracing, we propose bias tracing to seek the exact hidden states that contribute most to bias exhibited in the language models including masked language models and causal language models, which will guide us to select positions to edit for debiasing.

A.1 Tracing Bias Associations

Following Meng et al. (2022), we analyze all internal activations of a language model $\mathcal{M}$ during three runs: a clean run eliciting the bias in language models, a corrupted run disrupting the bias context modeling, and a corrupted-with-restoration run measuring bias exhibited in every single state.

- As for the **clean** run, we obtain $P_\theta(x_{\text{stereo}})$ and $P_\theta(x_{\text{anti}})$ for each sample in the datasets, and collect all hidden activations h_i^ℓ for each token i and each layer ℓ, given the input text $x = [x_1, \dots, x_K]$ and the $\mathcal{M}$ with L layers.

- In the **corrupted** run, noise is added to the embedding of bias attribute words in the input. For the embedding h_i^0 in the token sequences of bias attributes words to be corrupted, we set $\hat{h}_i^0 := h_i^0 + \tau$, where $\tau \sim \mathcal{N}(0; \sigma)$.[5] Then, $\mathcal{M}$ runs based on the corrupted embeddings and we collected the following corrupted activations $\hat{h}_i^\ell$. Since the existence of bias attribute words in a context is the reason why a context presents bias, corrupting the embedding of bias attribute words will remove the bias associations on the following language modeling process.

- With noisy embeddings, in the **corrupted-with-restoration** run, we restore specific hidden states of some token $i, i \in [0, K]$ (the bias attribute words, the attribute term, or the token before the attribute term) in an input context and layer $\ell, \ell \in [0, L]$ (the Transformer block, the attention layer, or the MLP layer) of a language model, which lets $\mathcal{M}$ output the clean state h_i^ℓ. The following forward-running executes without more intervention.

We calculate the absolute log probability difference between x_{stereo} and x_{anti}, $f_d(\theta, x_{\text{stereo}}, x_{\text{anti}}) = |\log P_\theta(x_{\text{stereo}}) - \log P_\theta(x_{\text{anti}})|$, to measure bias in a language model. The larger the difference is, the more biased $\mathcal{M}$ is. By running the network twice, bias tracing computes the bias association of activations. The clean run occurs first to obtain all clean activations. Secondly, embeddings of bias attribute words are corrupted and the lowest difference is obtained. Then the corrupted activations $\hat{h}_i^\ell$ of a certain token i and layer ℓ are restored to their original values h_i^ℓ from the same token i at the same layer ℓ. All differences are recorded after restoring activations over every token in the input context and every layer. If an activation restoration of a token i' and layer ℓ' causes a larger difference than a restoration from other tokens and layers, we can know that the activations of the token i' and layer ℓ' give more impetus to bias.

A.2 Tracing Data Construction

We conduct gender and race bias tracing in this paper. Therefore, gender and race bias attribute words are extracted in the context. We begin with utilizing SPARQL to query the instance of gender and race in Wikidata, obtaining a variety of words targeted to specific bias. These words are the source collection of bias attribute words. Based on the collection, we then adopt simple string matching to extract bias attribute words from the context sentence x of each sample s in the dataset. As a result, we can trace the activations of these bias attribute words in language models.

A.3 Bias Tracing with GPT2

We conduct gender and race bias tracing on the *intrasentence* part of StereoSet at every layer of language models and every token in contexts. The average bias associations of 500 samples with GPT2-medium are shown in Figure 5 and 6.

Bias best corresponds to the states of MLPs at lower layers. Figure 5 (a) illustrates that at layer 0-5 (layer 0-10 in Figure 6), MLPs in transformer blocks play a much more significant role in bias than attention layers, with peaking at layer 5 while bias associations of attention layers varies a little

[5] σ is three times the standard deviation of embeddings of 1000 subjects from `https://rome.baulab.info/data/dsets/known_1000.json` as Meng et al. (2022)

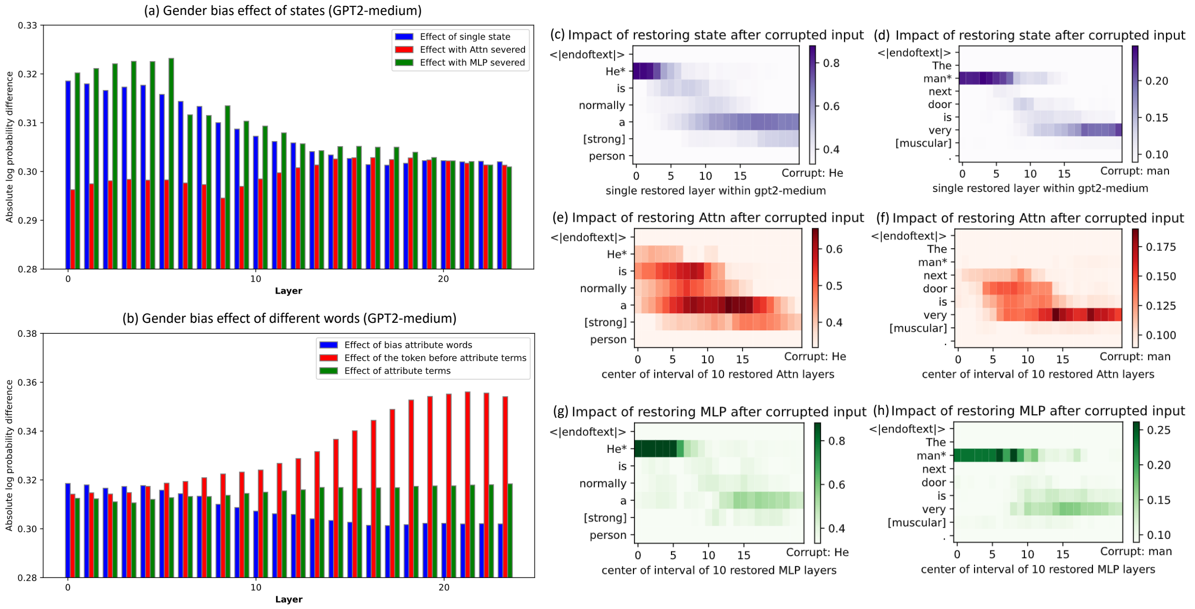


Figure 5: Gender bias tracing on GPT2-medium. (a) Comparing bias associations of bias attribute words on hidden states, attention layers, and MLP layers. (b) Comparing bias associations on single states of the bias attribute word, the token before the attribute term, and the attribute term. The bias impacts on output probability are mapped for the effect of (c-d) each hidden state on the context, (e-f) only MLP activations, and (g-h) only attention activations. * marks the corrupted bias attribute words and [] refers to the attribute terms in (c-h).

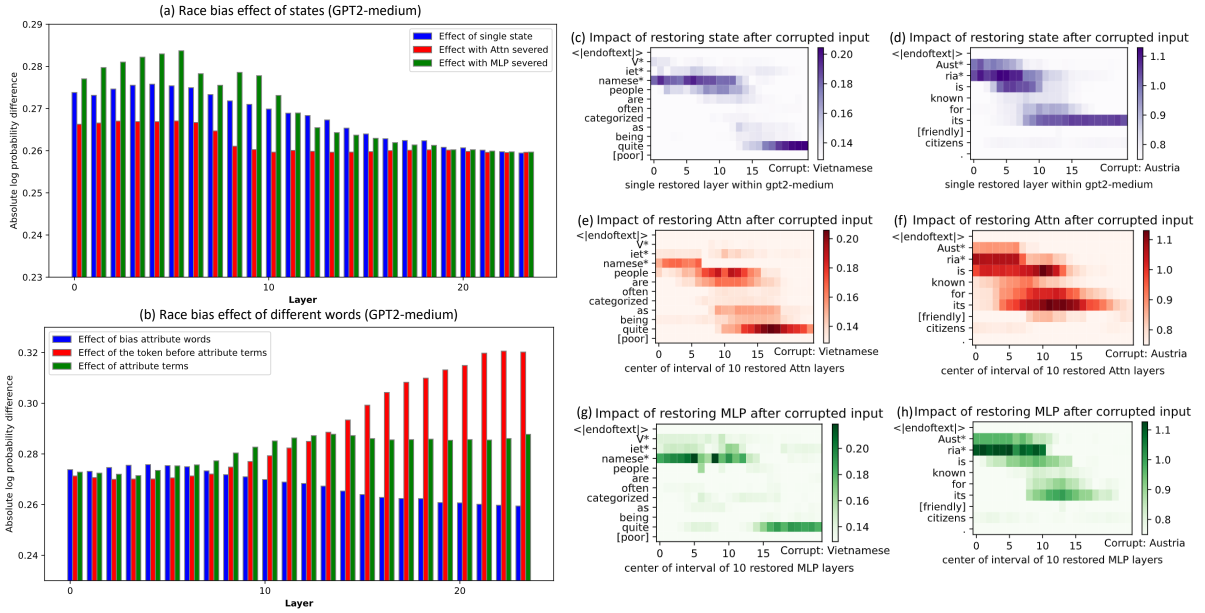


Figure 6: Race bias tracing on GPT2-medium.

among different blocks. This reveals that language models intensively present bias in the foundational representations learned by lower layers, and these early presentations can influence the subsequent layers. The reason is that since the lower layers capture the text patterns (Geva et al., 2021), bias patterns in the pre-trained corpus, such as bias attribute words' cooccurrence with stereotyped terms, are memorized in the early layers. Figure 5(b) and 6(b) also show that bias attribute words have the most effects at the early layers. Meanwhile, it indicates that the token before attribute terms associates a lot with bias at the upper layers of causal language models because semantic information is usually modeled in the top layers and the attribute term explicitly semantically presents bias. Two cases in Figure 5(c-h) and 6(c-h) illustrate the aforementioned observations well.

B Experimental Details

B.1 StereoSet

	# Gender	**# Race**	**# Religion**
$\mathcal{S}_{\textbf{edit}}^{\textbf{train}}$	617	2,307	210
$\mathcal{S}_{\textbf{edit}}^{\textbf{dev}}$	70	297	25
$\mathcal{S}_{\textbf{edit}}^{\textbf{test}}$	253	962	77

Table 5: The numbers of samples about different bias in our dataset.

B.2 Settings

We use four pre-trained language models in our experiments from HuggingFace (Wolf et al., 2019), including GPT2-medium[6], Gemma-2B[7], Mistral-7B-v0.3[8], and Llama3-8B[9]. For each training, we use one A800 80GB GPU and grid search among [8, 16, 64] batch sizes for batch editing. The λ is determined by grid searching in $\{1.0, 2.0, 3.0, 4.0, 5.0\}$.

B.3 Baselines

CDA (Counterfactual Data Augmentation) (Zmigrod et al., 2019; Barikeri et al., 2021) retrains a pre-trained language model. It generates and incorporates data representing what could have happened under different conditions. By altering aspects of data related to biased attributes, such as changing gender or race in a dataset, a counterfactual data set is created to create a more balanced training environment for models.

SentenceDebias (Liang et al., 2020) first estimates the demographic bias subspace by encoding sentences containing bias attribute words or their counterfactuals into sentence representations and using principle component analysis (Abdi and Williams, 2010) to define the bias subspace as the first K principle components, and then debiases sentence representations by subtracting their projection onto the bias subspace.

Self-Debias (Schick et al., 2021) first prompts a model to generate toxic text, such as encouraging a model to discriminate based on gender. Then, the model can generate a non-discriminative continuation, during which the probabilities of tokens that were prominent in the toxic generation are deliberately scaled down.

INLP (Ravfogel et al., 2020) introduces Iterative Null-space Projection (INLP), a method that reduces bias in word embeddings by iteratively projecting them onto the null space of bias terms using a linear classifier. This method constructs a projection matrix to project input onto the null space of the linear classifier, continuously updating both the classifier and the projection matrix.

B.4 Training for one bias type vs. a mixture of multiple bias types

Our goal is to efficiently deal with various types of bias in one training. We need to know if there is a debiasing performance drop if we don't deal with each bias type one by one. Therefore, we try to train editor networks with samples of one bias type and samples of a mixture of three bias types, respectively. Table 7 shows the comparison. The results indicate that training with a mixture of bias-type data is comparable with one bias-type data, indicating BiasEdit 's capability to deal with multiple types of bias simultaneously.

B.5 Evaluation on Crows-Pairs

We also use Crows-Pairs (Nangia et al., 2020) to evaluate the debiasing generality of BiasEdit. Crows-Pairs is a Crowdsourced Stereotype Pairs benchmark covering nine types of bias. We use

[6] https://huggingface.co/openai-community/gpt2-medium

[7] https://huggingface.co/google/gemma-2b

[8] https://huggingface.co/mistralai/Mistral-7B-v0.3

[9] https://huggingface.co/meta-llama/Meta-Llama-3-8B

Method	GPT2-medium			Gemma-2b		
	Gender	Race	Religion	Gender	Race	Religion
Pre-edit	61.46	59.57	73.33	63.54	64.54	66.67
CDA	51.04	44.68	66.67	-		
SentenceDebias	56.33	55.48	53.14	60.42	60.99	61.29
Self-Debias	**50.00**	59.57	53.33	56.25	43.26	56.25
INLP	47.92	52.81	61.29	63.57	60.99	63.33
EditBias	53.08	**50.35**	**53.12**	**52.81**	**49.83**	**53.17**
Method	**Mistral-7B-v0.3**			**Llama3-8B**		
	Gender	Race	Religion	Gender	Race	Religion
Pre-edit	65.62	68.09	70.00	62.50	62.41	73.33
CDA	-					
SentenceDebias	61.46	66.67	70.00	60.42	61.49	62.50
Self-Debias	41.67	41.89	40.00	44.79	47.52	**46.67**
INLP	59.38	68.79	68.75	56.25	63.83	70.00
EditBias	**49.65**	**48.94**	**53.24**	**52.39**	**50.17**	54.94

Table 6: Stereotype Score (%) for evaluating the baselines and BiasEdit on Crows-Pairs.

BiasType	**GPT2-medium**				**Gemma-2b**			
	One		**Mixture**		**One**		**Mixture**	
	SS (%)	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**
Gender	49.81	-1.22	49.42	-8.82	47.71	-5.36	48.59	-4.78
Race	55.27	-5.57	56.34	-5.12	54.88	-2.39	55.86	-4.35
Religion	49.64	-6.94	53.55	-1.92	50.42	-8.53	47.36	-5.44
BiasType	**Mistral-7B-v0.3**				**Llama3-8B**			
	One		**Mixture**		**One**		**Mixture**	
	SS (%)	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**	**SS (%)**	**ΔLMS (%)**
Gender	48.96	-10.55	46.24	-8.81	50.00	-10.98	49.18	-13.42
Race	53.32	-6.25	51.46	-8.59	46.28	-20.84	53.51	-11.77
Religion	52.15	-7.72	50.42	-0.03	50.42	-8.56	51.13	-10.02

Table 7: Training editor networks with data for one type of bias vs. mixed types of bias.

262 gender samples, 516 race samples, and 105 religion samples. In each sample, there are two sentences: a more stereotyped sentence and a less stereotyped one, which are regarded as x_{stereo} and x_{anti} respectively. *SS* for the baselines and BIASEDITon Crows-Pairs are shown in Table 6.

C Gender Counterfactual Test Set

We utilize the method mentioned in Appendix A.2 to extract gender attribute words in gender bias samples. These gender attribute words are reversed into their counterfacts. Then the labels “stereotype” and “anti-stereotype” are exchanged for each sentence. For instance, after reverse, the stereotyped context in Figure 2 is “Boys tend to be more determined than girls.” and the anti-stereotyped context is “Boys tend to be more soft than girls.”.