LIDAO: Towards Limited Interventions for Debiasing (Large) Language Models

Tianci Liu [1] **Haoyu Wang** [1] **Shiyang Wang** [1] **Yu Cheng** [2] **Jing Gao** [1]

Abstract

***Warning**: this paper contains model outputs exhibiting offensiveness and biases.*

Large language models (LLMs) have achieved impressive performance on various natural language generation tasks. Nonetheless, they suffer from generating negative and harmful contents that are biased against certain demographic groups (e.g., female), raising severe fairness concerns. As remedies, prior works intervened the generation by removing attitude or demographic information, inevitably degrading the generation quality and resulting in notable *fairness-fluency* trade-offs. However, it is still under-explored to what extent the fluency *has to* be affected in order to achieve a desired level of fairness. In this work, we conduct the first formal study from an information-theoretic perspective. We show that previous approaches are excessive for debiasing and propose LIDAO, a general framework to debias a (L)LM at a better fluency provably. We further robustify LIDAO in adversarial scenarios, where a carefully-crafted prompt may stimulate LLMs exhibiting instruction-following abilities to generate texts with fairness issue appears only when the prompt is also taken into account. Experiments on three LMs ranging from 0.7B to 7B parameters demonstrate the superiority of our method.

1. Introduction

Language models (LMs) parameterized by deep neural networks (Vaswani et al., 2017; Lewis et al., 2019; Radford et al., 2019; Brown et al., 2020) have thrived in producing fluent and meaningful texts on a variety of natural language generation tasks (See et al., 2019; Ji et al., 2023). Recently, researchers further applied LMs to more diverse classification tasks by transforming them to generative frames (Raffel et al., 2020). These successes underscore the versatility of LMs, establishing them as the foundations for different natural language processing applications (Bommasani et al., 2021; Zhou et al., 2023). In addition, with model sizes continually increasing, large language models (LLMs) have demonstrated unprecedented abilities to follow natural language instructions (Dong et al., 2022; Ouyang et al., 2022). These abilities empower the zero-shot adaptation of LLMs to unseen tasks (Kojima et al., 2022), paving the way towards artificial general intelligence (Bubeck et al., 2023).

Notwithstanding, despite their remarkable performance, LMs suffer from the *fairness* issue, i.e., they may generate negative texts that are biased against under-represented demographic groups (e.g., female) in our society (Sheng et al., 2019). For instance, GPT-2 (Radford et al., 2019) tends to generate more negative texts towards females (Huang et al., 2019). Such social biases, termed as *global biases* (Sheng et al., 2020), stem from the real world corpora due to historical reasons (Basta et al., 2019). Unsurprisingly, LMs reproduce or amplify the biases from the data whereon they are trained (Gehman et al., 2020; Schick et al., 2021).

As remedies, a vast amount of bias mitigation approaches have been proposed. Early attempts of fine-tuning with clean data proved effective on small LMs (Lu et al., 2020; Saunders & Byrne, 2020; Bender et al., 2021). However, they fall short on LLMs due to the prohibitive resource requirement for clean data curation and documentation (Schick et al., 2021). Recently, the lightweight decoding-time intervention paradigm, which directly modifies the text generation process in a LM, showed promising results and attracted great attention (Liu et al., 2021; Yang et al., 2023). Conceptually, these interventions imposed certain constraints to the generation process. For instance, in order to debias sentiment on different demographic groups, Sheng et al. (2020) steered the LM to generate positive texts in all cases. Similarly, Liang et al. (2021) sought to remove gender information during the text generation to avoid gender bias. Constraints as such, which we dub *constant constraints*, hinder the LMs from generating sentiment- or gender-specific texts, inevitably resulting in degradation on the generation quality (Liu et al., 2021; Fatemi et al., 2023).

[1]Purdue University [2]The Chinese University of Hong Kong. Correspondence to: Yu Cheng <chengyu@cse.cuhk.edu.hku>, Jing Gao <jinggao@purdue.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

Fairness-fluency trade-off refers to this phenomenon of increasing fairness at the cost of fluency, and has been widely observed in the literature (Liang et al., 2021). Noting this potential drawback, fluency plays a pivotal role in evaluating the practicability of different debiasing methods (Yang et al., 2023; Wang et al., 2023b). However, no formal studies has been conducted on to what extent the fluency *has to* be affected in order to achieve a desired level of fairness, and it is still an open question *whether existing methods can achieve the Pareto optimality, in the sense that no fluency improvement can be made without sacrificing the fairness.*

In this paper, we take the first step and provide a negative answer through an information-theoretic analysis on mitigating the (global) bias in LMs. In particular, we quantify the bias in text generation with mutual information (MI) between some *global property* and *demographic group*. Here a global property can be any characteristic that may raise fairness concern such as *sentiment* (Huang et al., 2019) or *regard* (Sheng et al., 2019). Built upon this formulation, we theoretically show that the constant constraints in existing approaches are in fact excessive and can be weakened. Particularly, we formalize this intuition: *Instead of intervening the generation of every word in a text, only words that may raise fairness issues need care.* Steered by the theoretical analysis, we propose a principled framework named LIDAO for bias mitigation with limited interventions. To be specific, in LIDAO, a word is allowed to be relevant with the (global) property or the demographic group, so long as it is independent with the other. Through this relaxation, LMs are allowed to generate property- or demographic group-specific content, thereby being able to produce more coherent and fluent texts. *To our best knowledge, we provide the first formal study on debiasing LMs targetting at better fairness-fluency trade-offs.* Sec 2 details these results.

The proposed LIDAO provably removes any bias in the generated text. However, for LLMs that exhibit strong instruction-following (IF) abilities (Ouyang et al., 2022), a carefully-crafted adversarial prompt may instruct or stimulate them to generate a short text where the fairness issue appears *only if* the context from the prompt is taken into account. As such, LIDAO may fall short to apply due to the absence of, for instance, the mention of any demographic group in the generation. To enhance the applicability of LIDAO in this adversarial scenario, we extend the bias formulation and bound it by two parts: a generation bias solvable with LIDAO, and the extent to which one can infer the demographic group from the generation alone without seeing the prompts. We propose a heuristic to minimize the second component by utilizing a LM's IF ability as well. This design makes our solution highly lightweight and effective. Sec 3 discusses this extension in details.

Experimental results in Sec 4 demonstrate the efficacy of our method in achieving better fairness-fluency trade-offs when debiasing small and large LMs on three representative tasks. In the remaining part of this paper, we review related works in Sec 5, and conclude the paper in Sec 6.

2. Proposed Method

Grounded in a theoretic analysis, we propose LIDAO to debias language models (LMs) with limited interventions, thereby achieving a better fairness-fluency tradeoff. LM preliminaries are also provided.

2.1. Preliminaries

Given a sentence $\boldsymbol{x} = (x_1, \dots, x_T)$ of length T and each $x_t \in \mathcal{V}$ is a token from the vocabulary $\mathcal{V}$, a language model (LM) parameterzied by θ assigns probability $p_\theta(\boldsymbol{x})$ using the chain rule (Bengio et al., 2000) as follows:

$$p_\theta(\boldsymbol{x}) = \prod_{t=1}^{T} p_\theta(x_t \mid x_1, \dots, x_{t-1}) \triangleq \prod_{t=1}^{T} p_\theta(x_t \mid \boldsymbol{x}_{<t}),$$

where $p_\theta(x_t \mid \boldsymbol{x}_{<t})$ is the predicted distribution of the next token x_t given previous $\boldsymbol{x}_{<t} \triangleq (x_1, \dots, x_{t-1})$. To generate $\boldsymbol{x}$, the LM iteratively computes $p_\theta(x_t \mid \boldsymbol{x}_{<t})$ and draws x_t from it; then the sampled x_t is fed back into the LM as part of the inputs for future steps. The generation completes if a pre-specified special token that marks the end of the sentence is returned, or the maximum length is reached. Different strategies to draw x_t have also been proposed (Fan et al., 2018; Holtzman et al., 2019) to improve the overall fluency and diversity of generated $\boldsymbol{x}$. We refer interested readers to (Zhao et al., 2023; Min et al., 2023) and references therein for more detailed backgrounds of LMs. From now on, we use $\boldsymbol{x} \sim p_\theta(\boldsymbol{x})$ to denote sentence $\boldsymbol{x}$ generated by the LM where p_θ is the sampling distribution parameterized by the LM. We treat $\boldsymbol{x}$ as a random variable.

2.2. Debiasing Formulation

Given sentence $\boldsymbol{x} \sim p_\theta(\boldsymbol{x})$, we further assume that the sentence $\boldsymbol{x}$ contains some global property $g \triangleq g(\boldsymbol{x})$ and mentions some demographic group $a \triangleq a(\boldsymbol{x})$. Representative global property includes sentiment (Huang et al., 2019), regard (Sheng et al., 2019), and toxicity (Gehman et al., 2020). We follow previous works (Yang et al., 2023; Wang et al., 2023b) and treat a as another (global) property of $\boldsymbol{x}$. Built upon the definition of g and a, an *unbiased* LM should generate g and a independently. Mathematically, this concept resembles the *demographic parity* fairness notion (Pedreshi et al., 2008; Huang et al., 2019), whose violation can be quantified by a positive mutual information (MI) $I(g; a) > 0$ (Zemel et al., 2013). To mitigate the bias in the

LM, we alter $\boldsymbol{x} \sim p_\theta(\boldsymbol{x})$ by solving

$$\min_\theta I(g(\boldsymbol{x}); a(\boldsymbol{x})), \quad \boldsymbol{x} \sim p_\theta(\boldsymbol{x}). \tag{1}$$

The aforementioned bias is known as *global bias* (Sheng et al., 2020) and has been identified in various LMs (Huang et al., 2019; Ranaldi et al., 2023). As remedies, different interventions have been proposed (Liang et al., 2021; Liu et al., 2021; Yang et al., 2023). However, most existing machinery enforced the LMs to output constant g or a in lieu of solving Eqn (1) exactly. For instance, if $a(\boldsymbol{x}) = b$ for all possible $\boldsymbol{x}$ where b is a constant, then $I(g(\boldsymbol{x}); a(\boldsymbol{x})) = I(g(\boldsymbol{x}); b) = 0$ holds and is minimized naturally by the non-negativity of MI (Cover, 1999).

Nonetheless, these constant constraints hinder a LM from producing property- or demographic group-specific texts, inevitably degrading its generation quality. As a consequence, a fairness-fluency trade-off occurs after debiasing (Liang et al., 2021). This raises a crucial yet still open question:

With elaborated tuning, are trade-offs from the constant constraints able to achieve the Pareto optimality[1] *?*

In this paper, we give a negative answer through a formal information-theoretic analysis. Subsequently, a more adaptable debiasing method called LIDAO is derived.

2.3. Limit Intervention for Debiasing Across its Options

In this section, we show that the aforementioned constant requirement is an overkill for Eqn (1) by proving that a weaker condition is able to solve it. Targeting on this weakened condition, we come up with a new solution called LIDAO that is capable of achieving a better fairness-fluency trade-off. In particular, below theorem gives a sufficient condition to minimize $I(g; a)$, with its proof deferred to App A.1.

Theorem 2.1. *For sentence $\boldsymbol{x} = (x_1, \ldots, x_T)$ generated by a LM that involves some global property g and demographic group a. If at every step $t > 1$, condition*

$$\ell(g; t) \triangleq I(g; x_t \mid \boldsymbol{x}_{<t}) = 0, \tag{2}$$

$$\textit{or} \quad \ell(a; t) \triangleq I(a; x_t \mid \boldsymbol{x}_{<t}) = 0 \tag{3}$$

holds, then $I(g; a) = 0$. In words, if each x_t is relevant to only g or a, then g and a are independent with each other.

Overkill of the constant constraints. Thm 2.1 boils downs to that in order to minimize $I(g, a)$, each token x_t needs to be independent of a *only if* it reflects g, and vice versa. This requirement is much weaker than the constant constraint, as the LM is allowed to generate property- or demographic group-specific texts. In contrast, a constant constraint such as $a(\boldsymbol{x}) = b$ asks for independence between a and *all* tokens, regardless of whether they are relevant to g or not.

Following Thm 2.1, we propose to **limit the intervention for debiasing across its options** (LIDAO). Conceptually, LIDAO *chooses to solve* $\ell(g; t)$ or $\ell(a; t)$ adaptively, and we propose two variants to automate this process. Specifically, the **min-based LIDAO** seeks to solve the milder $\min(\ell(g; t), \ell(a; t))$. The **product-based LIDAO**, on the other hand, minimizes the product $\ell(g; t)\ell(a; t)$. The underlying rationale is that by taking gradient $\nabla(\ell(g; t)\ell(a; t)) = \ell(a; t)\nabla\ell(g; t) + \ell(g; t)\nabla\ell(a; t)$, each loss is weighted by the magnitude of the other. Consequently, the smaller term is minimized more aggressively while the larger term tends to remain unchanged. Finally, we emphasize that LIDAO is a general framework and admits other designs for determining which loss to optimize.

2.4. LIDAO Implementation

LIDAO requires computing $\ell(g; t)$ and $\ell(a; t)$ by estimating terms $I(g; x_t \mid \boldsymbol{x}_{<t})$ and $I(a; x_t \mid \boldsymbol{x}_{<t})$, which proves to be a challenging task due to the lack of analytic form of MI and necessities some approximations thereof (Poole et al., 2019). In this work we follow UDDIA (Yang et al., 2023), a state-of-the-art framework that seeks to minimize $\ell(g; t)$ and $\ell(a; t)$ simultaneously, for such approximate proxies.

Approximate Proxies. Yang et al. (2023) proved that the PPLM loss (Dathathri et al., 2019) is a valid proxy for $\ell(g; t)$ under certain condition and we adopt it as $\hat{\ell}(g; t)$. For $\ell(a; t)$, Yang et al. (2023) estimated it by

$$\begin{aligned}\hat{\ell}(a; t) =& \mathbb{E}_{p_\theta(x_t \mid \boldsymbol{x}_{<t})}\left[D\left(q(a \mid \boldsymbol{x}_{<t+1}) \| q(a \mid x_t)\right)\right] \\ &+ \mathbb{E}_{p(a)}\mathbb{E}_{p_\theta(x_t)}\left[q(a \mid x_t) \log \frac{q(a \mid x_t)}{p(a)}\right],\end{aligned}$$

where $p(a)$ is a pre-specified prior distribution over a. Here $q(a = k \mid x_t) \propto \cos(e(x_t), a_{k,v})$ where $e(x_t)$ is the embedding of x_t, $a_{k,v}$ is the first principle component of a set of manually-collected seed words $\mathcal{V}_{a=k}$ from demographic group $a = k$, and $\cos(\cdot, \cdot)$ computes the cosine similarity between the two vectors. Similarly $q(a = k \mid \boldsymbol{x}_{<t+1})$ is constructed by replacing $e(x_t)$ with $(\sum_{i=1}^{t} e(x_i))/t$. Given these terms, $\hat{\ell}(a; t)$ can be computed analytically as all expectations are over the next-token distribution.

Adjusted min-based LIDAO. In execution, the scales of $\hat{\ell}(g; t)$ and $\hat{\ell}(a; t)$ can differ significantly. As a consequence, directly plugging them into the min-based principle will make it degrade to choose the same option throughout the generation process. To solve this issue, we rescale each loss before determining the minimum. These rescaling factors are weighted pooling applied to the history losses. Namely,

[1] A Pareto optimal solution refers to a situation where neither fluency nor fairness can be improved without hurting the other. We ask if constant constraints are possible to achieve such optimality.

the factor $w(g;t)$ to rescale $\hat{\ell}(g;t)$ at step t is

$$w(g;t) = \left(\sum_{j=1}^{t-1}\gamma^j\right)^{-1}\left(\sum_{j=1}^{t-1}\hat{\ell}(g;t-j)\gamma^j\right),$$

and factor $w(a;t)$ is defined similarly. Hyper-parameter γ controls the speed of forgetting earlier histories and a smaller γ forgets early histories faster. In practice we set $\gamma = 0.5$. Equipped with these factors, the min-based principle solves $\min(\hat{\ell}(g;t)/w(g;t), \hat{\ell}(a;t)/w(a;t))$ at step t.

Decoding-Time Intervention. LIDAO intervenes the generation as in Yang et al. (2023) by applying bias-tuning (Zaken et al., 2021) to θ before generating x_t. Denote the updated parameters at step t by θ_t and corresponding distribution by p_{θ_t}, we generate $x_t \sim p_{\text{mix}}$ by mixing the two distributions

$$p_{\text{mix}}(x_t \mid \boldsymbol{x}_{<t}) \propto (p_{\theta_t}(x_t \mid \boldsymbol{x}_{<t}))^{\tau}\,(p_{\theta}(x_t \mid \boldsymbol{x}_{<t}))^{1-\tau},$$

as in Dathathri et al. (2019), where τ is a hyper-parameter. After sampling x_t, the parameters in LM is reset to θ.

Remark 2.2. In this work we adopt the estimators to minimize $\ell(g;t)$ and $\ell(a;t)$ from UDDIA given their good performance. However, LIDAO does not adhere to these choices and one can propose and derive their own methods. Moreover, the main difference between LIDAO and UDDIA lies in that UDDIA is a constant constraint-based method which seeks to minimize $\ell(g;t)$ and $\ell(a;t)$ at every step, thereof does not offer the same flexibility of LIDAO.

3. Robustness against Adversarial Prompts

As presented in Sec 2, LIDAO is capable of removing any undesired association between g and demographic group a that may raise fairness concerns in a text $\boldsymbol{x}$ generated by the LM. Notwithstanding, given a carefully-designed adversarial prompt $\boldsymbol{c}$, LMs, especially those exhibit strong instruction-following abilities, are prone to follow $\boldsymbol{c}$ and generate offensive $\boldsymbol{x}$ (Wei et al., 2023; Wang et al., 2023a). Moreover, when $\boldsymbol{c}$ is already concluding and specific about some demographic group, $\boldsymbol{x}$ may be short and solely consists g. As such, Eqn (1) holds even if g in $\boldsymbol{x}$ is systematically biased against the demographic group in $\boldsymbol{c}$ due to the constant a.

Wei et al. (2023) showed that such threats are ubiquitous on large LMs (LLMs), necessitating the need of robustifying our debiasing formulation as well. To this end, we note that the vulnerability stems from the underlying assumption that a biased $\boldsymbol{x}$ should contain both a and g simultaneously. We thereby relax this requirement and extend a as a joint property of $\boldsymbol{c}$ and $\boldsymbol{x}$ to debias the LM as follows:

$$\min_\theta I(g(\boldsymbol{x}); a(\boldsymbol{c},\boldsymbol{x})), \quad \boldsymbol{x} \sim p_\theta(\boldsymbol{x} \mid \boldsymbol{c}). \tag{4}$$

Unfortunately, LIDAO cannot solve Eqn (4) provably due to the additional dependency of a on $\boldsymbol{c}$ that is out of control. Nonetheless, it is viable to quantify the violation as follows.

Proposition 3.1. *For sentence $\boldsymbol{x} = (x_1, \ldots, x_T)$ generated by a LM that is prompted by $\boldsymbol{c}$, i.e., $\boldsymbol{x} \sim p_\theta(\boldsymbol{x} \mid \boldsymbol{c})$, then*

$$I(g(\boldsymbol{x}); a(\boldsymbol{c},\boldsymbol{x})) \le I(g(\boldsymbol{x}); a(\boldsymbol{x})) + H(a(\boldsymbol{c},\boldsymbol{x}) \mid a(\boldsymbol{x})),$$

where $a(\boldsymbol{c},\boldsymbol{x})$ and $a(\boldsymbol{x})$ denote the referred demographic group in the joint of $(\boldsymbol{c},\boldsymbol{x})$ and $\boldsymbol{x}$, respectively.

The proof is deferred to App A.2. Conceptually, Prop 3.1 states that $I(g(\boldsymbol{x}); a(\boldsymbol{c},\boldsymbol{x}))$ can be bounded from two parts: $I(g(\boldsymbol{x}); a(\boldsymbol{x}))$ that can be minimized with LIDAO, and $H(a(\boldsymbol{c},\boldsymbol{x}) \mid a(\boldsymbol{x}))$ as a measure of how easy to infer $a(\boldsymbol{c},\boldsymbol{x})$ from $\boldsymbol{x}$ without seeing $\boldsymbol{c}$. To minimize the second term, we propose a heuristic to encourage $a(\boldsymbol{x}) = a(\boldsymbol{c};\boldsymbol{x})$ and refer to this extended LIDAO as eLIDAO.

Inspired by Schick et al. (2021), eLIDAO resorts to some reference LM that can generate $\boldsymbol{x}$ with high affinity $a(\boldsymbol{x}) = a(\boldsymbol{c};\boldsymbol{x})$ to guide generations. Specifically, assume the reference LM predicts $p_{\text{ref}}(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t})$, we sample x_t from

$$p^{\text{ext}}_{\text{mix}}(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t}) \propto \begin{cases} p_{\text{ref}}(x_t = v \mid \boldsymbol{c}, \boldsymbol{x}_{<t}) & \text{if } v \in \mathcal{V}_a; \\ p_{\text{mix}}(x_t = v \mid \boldsymbol{c}, \boldsymbol{x}_{<t}) & \text{otherwise.} \end{cases}$$

Here $\mathcal{V}_a$ is a set of seed words whereby a can be identified. For simplicity we adopt the ones to construct $\hat{\ell}(a;t)$ as $\mathcal{V}_a = \cup_{k=1}^{K}\mathcal{V}_{a=k}$. All other implementations is same as in LIDAO. We summarize (e)LIDAO in Alg 1.

Design of the Reference Model. We utilize the instruction-following ability to let the untuned LM itself play the reference model. This design is extremely lightweight and requires no extra training or finetuning stage. To minimize the unintentional influence on the reference model, we use a zero-shot instruction that only asks the LM to focus on the demographic group if it presents in $\boldsymbol{c}$ without providing any example. We provide these instructions in App B. It is note worthy that no extensive prompt engineering is conducted, and the sample instructions can be sub-optimal.

Remark 3.2. Reference models have been used to guide generations (Krause et al., 2021; Liu et al., 2021; Schick et al., 2021). However, previous works often need to train the reference model from scratch (Krause et al., 2021) or fine-tune it with carefully curated data (Liu et al., 2021), which can be time and resource consuming. Recently, Schick et al. (2021) successfully conducted self-guiding based on a LM's internal knowledge of *social bias*, but it is still an open question how much such a complicate knowledge exists in the LM and in which way it can be effectively and reliably acquired (Salinas et al., 2023; Cohen et al., 2023). In contrast, our solution only requires the knowledge of *demographic groups*, which is expected easier considering that contextual

Algorithm 1 (e)LIDAO algorithm

Input: Maximum length of generation T (Sec 2). Estimator $\hat{\ell}(g;t)$ and $\hat{\ell}(a;t)$ (Sec 2.4). Mixing parameter τ and decaying parameter γ (Sec 2.4). Language model parameterized by θ; a small subset of θ of will be updated by bias-tuning (Sec 2.4). Optional user-specified prompt $\boldsymbol{c}$ that is possibly adversarial (Sec 3).
Additional input for eLIDAO: Seed words $\mathcal{V}_a$ and eLIDAO system prompt $\boldsymbol{c}_e$ (Sec 3).
Initialize: Set $\boldsymbol{c} =$ <BOS> if not provided.
for $t = 1, \ldots, T$ **do**
  Compute base $p_\theta(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t})$ and reference distribution $p_\theta(x_t \mid \boldsymbol{c}_e, \boldsymbol{c}, \boldsymbol{x}_t)$.
  Compute $\hat{\ell}(g;t)$ and $\hat{\ell}(a;t)$ based on $p_\theta(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t})$.
  Compute the (e)LIDAO loss $\ell_t(\theta)$ by

$$\ell_t(\theta) = \min(\hat{\ell}(g;t)/w(g;t), \hat{\ell}(a;t)/w(a;t)),$$

  where $w(g;t)$ and $w(a;t)$ are decaying weights as defined in Sec 2.4 if min-based (e)LIDAO is applied, else prod-based (e)LIDAO uses

$$\ell_t(\theta) = \hat{\ell}(g;t)\hat{\ell}(a;t).$$

  Apply bias-tuning to minimize $\ell_t(\theta)$ by updating θ with Adam optimizer. Obtain debiased parameters θ_t. Compute updated distribution $p_{\theta_t}(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t})$.
  Compute unnormalized mixing distribution

$$p_{\text{mix}}(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t}) \propto (p_{\theta_t}(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t}))^{\tau} (p_\theta(x_t \mid \boldsymbol{c}, \boldsymbol{x}_{<t}))^{1-\tau}.$$

  If using eLIDAO, for all words $v \in \mathcal{V}_a$, update

$$p_{\text{mix}}(x_t = v \mid \boldsymbol{c}, \boldsymbol{x}_{<t}) \leftarrow p_\theta(x_t = v \mid \boldsymbol{c}_e, \boldsymbol{c}, \boldsymbol{x}_t).$$

  Normalize p_{mix} and sample token x_t. Reset θ.
  if $x_t =$ <EOS> **then**
    Break
  end if
end for
output Generated text $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_T)$.

word embedding are demographic-aware (Zhao et al., 2019; Sweeney & Najafian, 2019).

4. Experiments

In this section we experiment the proposed (e)LIDAO with debiasing three LMs ranging from 0.7B to 7B parameters on three tasks. As expected, (e)LIDAO achieved a much better empirical fairness-fluency trade-off, outperforming existing methods to a large extent. Meanwhile, all three LMs including LLMs exhibit substantial bias before intervention, highlighting the necessity for the mitigation (Wang et al., 2023a; Ranaldi et al., 2023).

4.1. Experiment Setup

Base Models. We experiment with three representative LMs across different model families and sizes. **GPT-2 Large** (Radford et al., 2019) has been widely used as a test bed for debiasing methods (Dathathri et al., 2019; Krause et al., 2021; Liu et al., 2021; Yang et al., 2023), and we follow this convention. **OPT** (Zhang et al., 2022) and **Falcon-7B-Instruct** (Almazrouei et al., 2023) are two 7B parameters LLMs. Notably, Falcon has undergone an instruction tuning. We term the three base models as GPT-2, OPT, and Falcon respectively for brevity.

Tasks. We consider debiasing three global property g that raise fairness concerns.

- **Sentiment**: We debias each LM to generate sentences for different demographic groups with similar sentiment as in Huang et al. (2019); Yang et al. (2023).
- **Regard**: Sheng et al. (2019) noted that sentiment may not identify negative attitude against some demographic groups and proposed to capture such bias with the new notion *regard*. We debias each LM to output equivalent level of regards towards different demographic groups.
- **Toxicity**: Xu et al. (2021) noted that detoxification, i.e., reducing toxicity in generated texts, often results in new bias. We follow Yang et al. (2023) and minimize this bias.

Baselines. We compare the proposed LIDAO with four recent debiasing baselines.

- **GeDi** (Krause et al., 2021) incorporates a reference class-conditioned LM to predict property g for all possible next tokens based on the Bayes rule. We adopted the pretrained GeDi released by the authors and recommended generation hyper-parameters directly.
- **PPLM** (Dathathri et al., 2019) uses the gradient from a lightweight classifier that predicts property g from the LM's hidden representations to remove their associations. We trained linear PPLM classifiers for each LM separately. Due to the lack of regard-annotated dataset, we labeled SST-5 (Socher et al., 2013) with the regard scorer (Sheng et al., 2020) to trained PPLM.
- **DExperts** (Liu et al., 2021) uses two reference LMs finetuned on data at different-leveled property g to adjust the next-token probability. We adopted the pretrained DExperts released by the authors and recommended generation hyper-parameters directly.
- **UDDIA** (Yang et al., 2023) seeks to minimize $\ell(g;t)$ in Eqn (2) and $\ell(a;t)$ in Eqn (3) simultaneously with

$\hat{\ell}(g;t)$ and $\hat{\ell}(a;t)$ detailed in Sec 2.4. We dropped the *redo* mechanism in UDDIA for simplification[2]. We tuned hyper-parameters for the used estimators on UDDIA and applied them to LIDAO.[3].

GeDi and DExperts use pre-trained reference models that take GPT-2 as backbones, and cannot guide OPT and Falcon that use different tokenization methods without substantial reimplementation. Therefore, we only test GeDi and DExperts on debiasing GPT-2 as in the literature. Due to page limitations we defer more implementation details to App C.

Dataset. Following previous works (Liu et al., 2021; Wang et al., 2023b; Yang et al., 2023), we focus on the *male* and *female* gender groups. We adopt the notion of *gender polarity* (Dhamala et al., 2021; Wang et al., 2023b) and label a as the majority group of gender words. However, our framework is general and applies to other demographic groups such as *race* as well. We focus on a set of paired adversarial prompts released by Yang et al. (2023) that encourages *toxic* and *biased* texts because of high quality and challenging nature. This dataset consist of 175 pairs that are handcrafted from a 1K subset of the RealToxicityPrompts (Gehman et al., 2020). For each prompt, we generate 10 (GPT-2) or 5 (OPT and Falcon) continuations of maximum length 20, which constitute the final dataset we use for evaluation.

Evaluation. The performance is evaluated from three aspects. First, we check how they can manipulate the global property by reporting the averaged g. This allows us to quantify how in general harmful generations can be avoided, which is pivotal for the evaluating toxicity. Second, we evaluate how the fluency is affected by checking the mean perplexity according to a pre-trained GPT-2 XL as in previous works (Pozzobon et al., 2023; Yang et al., 2023). We report perplexity evaluated by larger models from the same family (i.e., OPT-13B and Falcon-40B) in App D. Finally, biases are quantified in a task-specific manner as below to follow the convention in the literature (Huang et al., 2019; Gehman et al., 2020; Liu et al., 2021; Yang et al., 2023).

- **Sentiment**: the bias is the difference in *the average sentiment score* on two gender groups.
- **Regard**: the bias is the *total variation (TV) distance between the 3-way (negative, neutral, and positive) regard distributions* on two gender groups.
- **Toxicity**: the bias is the difference in *the maximum toxicity score* from Perspective API[4] on two gender groups as a worst-case measure.

[2]We found in our scenarios *redo* often increases the running time with marginal improvement given proper learning rate.

[3]This setup put our proposed method in an unfavorable situation. Nevertheless, ours can still achieve comparable or better performance than that of UDDIA.

[4]https://github.com/conversationai/perspectiveapi.

For more comprehensive comparison, we analyze the performance with respect to generated $a(\boldsymbol{x})$ and joint $a(\boldsymbol{c}, \boldsymbol{x})$ as detailed in Sec 4.2 and Sec 4.3 respectively. Human evaluation results and case studies are also provided.

4.2. Debias over Generated $a(\boldsymbol{x})$

In this section we assess the bias in sentiment, regard, and toxicity against gender group $a(\boldsymbol{x})$ in generated text $\boldsymbol{x}$ and how it can be minimized on the three tasks. Table 1 reports these results. On the debiasing regard task, we combine harmless *neutral* and *positive* regard groups into *non-negative* regard (NNRe) group for straightforward quantification of the averaged regard level. We note that *on GPT-2 and OPT without instruct tuning, the min-based (e)LIDAO consistently performed better, whereas Falcon that has undergone instruction tuning favored the prod-based (e)LIDAO.* Due to space limitation, Table 1 reports only these favored results, remaining results are deferred to App D.

Bias in LMs. As per Table 1, all three LMs generations contain substantial bias. Notably, given the adversarial nature of the prompts, the two LLMs tend to produce more offensive (OPT) or biased (Falcon) texts compared to the smaller GPT-2. In specific, OPT consistently produces the least friendly content, with the lowest sentiment and regard, or the highest toxicity scores. Simultaneously, its exhibits the highest bias in regard among the three LMs. Falcon, on the other hand, in spite of achieving the best scores, its biases in sentiment and toxicity are also significantly higher than the others. Meanwhile, by examining the sentiment and regard scores, we observe that although Falcon attains the highest average sentiment score, its regard is the lowest. This finding aligns with Sheng et al. (2019), confirming that sentiment may not provide a comprehensive measure of the LM's attitude. In conclusion, these results highlight that *LLMs, regardless of undergoing instruction tuning or not, are prone to generate texts as unfriendly, if not more so, as classical LMs in challenging scenarios,* emphasizing the need for additional bias mitigation (Wang et al., 2023a; Ranaldi et al., 2023).

Debiasing Performance. According to Table 1, when debiasing the three LMs of different sizes, the proposed LIDAO demonstrated remarkable debiasing performance and reached the top two positions in seven out of nine scenarios. At the same time, LIDAO generated the top two fluent texts in six scenarios based on the perplexity. In the remaining two of three cases, its perplexity was slightly higher than that of eLIDAO, its own extended version. In contrast, the bias and fluency of UDDIA are both worse than LIDAO, despite their implementation similarity. These empirical findings, coupled with the theoretical guarantee from Thm 2.1, clearly underscore the significant success of LIDAO in

advancing the fairness-fluency trade-off in debiasing LM by a considerable margin compared to existing approaches.

Finally, we note that although DExperts guided GPT-2 to generate the most fluent texts, these texts exhibited the most negative sentiment and highest toxicity. This highlights the challenge of adapting reference models to new scenarios that diverge from the training corpora (Pozzobon et al., 2023).

4.3. Debias over Joint $a(\boldsymbol{c}; \boldsymbol{x})$

In this section, we evaluate the bias with respect to gender group $a(\boldsymbol{c}; \boldsymbol{x})$ by considering the generated text $\boldsymbol{x}$ together with the prompt context $\boldsymbol{c}$ and how different approaches can optimize such bias on three tasks. Results are presented in Table 2. As previous we use the non-negative regard (NNRe) to verify how average regard is affected. On GPT-2 and OPT we report the min-based (e)LIDAO results and on Falcon we report the prod-based (e)LIDAO results, with remaining results deferred to App D.

Bias in LMs. As shown in Table 2, we note that all three LMs once again exhibit considerable bias against different gender groups. Many previously discussed findings remain applicable here so we refrain from reiterating them for brevity and direct readers to the prior section. Notably, the bias in regard from OPT and the toxicity from Falcon are now lower, suggesting that certain texts $\boldsymbol{x}$ that did not mention a were not accounted for in the formulation in Eqn (1). These distinctions highlight the significance of *considering the influence of prompts when identifying and evaluating bias, particularly when dealing with LLMs.*

Debiasing Performance. Referring to Table 2, while eLIDAO lacks the theoretical guarantee of LIDAO for debiasing, it demonstrated strong empirical success in mitigating bias on the three LMs. In contrast, LIDAO, lacking consideration for the prompt $\boldsymbol{c}$, experienced a considerable decline in debiasing performance as elucidated by Prop 3.1 due to the presence of $H(a(\boldsymbol{c}; \boldsymbol{x}) \mid a(\boldsymbol{x}))$. Compared with LIDAO, eLIDAO consistently achieved lower bias in all nine scenarios, and attained the best two performance in the six. In the meantime, eLIDAO maintained compelling, if not the best, fluency. Namely, it generated the top two fluent texts in eight out of nine scenarios after ignoring its original LIDAO version. Consequently, eLIDAO attained a notable fairness-fluency trade-off. These results collectively established the effectiveness of utilizing the LM itself as the reference model for the self-guidance, as proposed in Sec 3.

Additionally, DExperts again proved less effective in altering sentiment and toxicity in GPT-2 generations, despite being designed to (Liu et al., 2021). We conjecture that this reflects how distribution shift can lead a reference model inaccurately captured the desired properties, resulting in the performance degradation. The proposed eLIDAO, on the contrary, only requires a reference model to identify *gender polarity*, which is expected invariant in diverse contexts, making the resultant guidance more robust to domain shifts.

4.4. Human Evaluation and Case Studies

We further conduct human evaluation and case studies on (e)LIDAO and compare them to the backbone UDDIA. Due to the substantial human effort, each LM is evaluated on one task. We randomly sampled 20% of prompts, and generated one continuation per prompt. This process resulted in 420 pairs to compare in total. We requested two annotators to choose which sentience in each pair has better quality of *fluency* and *friendliness*. Results in Table 4 again affirm the effectiveness of (e)LIDAO.

We conclude this section with a case study on the generated texts. Table 3 presents the original and debiased generations from three LLMs, one for each task. Examples show that the proposed (e)LIDAO is capable of generating high quality texts on par with existing methods.

To gain a deeper understanding on how LIDAO seeks to debias the generation through limited interventions, we color-code each token in the original generations with blue and red to denote whether the property g or gender a's influence *should be* removed in order to debias. Namely, we label tokens that achieved lower $\hat{\ell}(g; t)$ by blue and that achieved lower $\hat{\ell}(a; t)$ by red. We found that the debiasing strategy employed by LIDAO enjoys a certain level of interpretability. For instance, in Falcon's generation, to eliminate the bias, LIDAO argues that the generation of "black male" should not depend on g, and the action "(did) nothing" should be agnostic with a, as these dependencies are mild in nature.

5. Related Works

Bias in text generation as a distribution-level difference across varied demographic groups has been widely observed in the literature (Gallegos et al., 2023). Existing measures of social bias falls into two categories. *Local bias* (Liang et al., 2021) refers to the difference in the high-dimensional probability distributions of a LM and boils down to the accumulated difference in token-level distributions. *Global bias* (Sheng et al., 2020) considers the difference in the distribution of some low-dimensional global property (Sheng et al., 2021). In general, the practical harm of global bias is more straightforward than local bias and has attracted more attention (Blodgett et al., 2020). In this work we study the global bias from an information-theoretic perspective.

Bias mitigation has accumulated a vast literature (Li et al., 2023). Early attempts seek to finetune a pretrained LM on some carefully-curated datasets that contains no or minimal bias (Lu et al., 2020; Saunders & Byrne, 2020) or through a regularization training (Huang et al., 2019; Peng et al.,

Table 1. Performance of debiasing the three LMs (shadowed in different colors) with respect to generation $a(\boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. https://experience.graebel.com/login The best debiased and fluent results are in **bold** and the second best results are underlined. GPT-2 and OPT used the min-based (e)LIDAO and Falcon used the prod-based (e)LIDAO.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
GPT-2	0.36±0.44	4.41	22.17	0.71±0.55	3.89	22.03	0.16±0.17	9.35	22.17
+GeDi	0.64±0.44	4.99	48.11	-	-	-	0.10±0.12	6.81	48.82
+DExperts	0.37±0.45	1.42	**16.00**	-	-	-	0.15±0.16	8.58	**16.00**
+PPLM	0.38±0.45	1.38	34.10	0.73±0.55	8.74	36.01	0.15±0.15	6.24	26.85
+UDDIA	0.41±0.45	2.77	32.32	0.72±0.57	2.66	34.24	0.14±0.15	11.74	33.29
+LIDAO	0.40±0.45	0.32	27.49	0.69±0.55	2.15	**28.02**	0.14±0.15	9.88	31.45
+eLIDAO	0.40±0.45	**0.21**	28.26	0.72±0.55	**1.53**	28.08	0.14±0.15	**5.91**	30.20
OPT	0.33±0.43	0.98	23.62	0.67±0.58	11.58	23.63	0.19±0.20	0.78	23.62
+PPLM	0.40±0.46	4.80	49.07	0.72±0.63	5.10	46.86	0.17±0.18	2.27	36.69
+UDDIA	0.39±0.45	5.40	62.78	0.74±0.63	1.81	66.39	0.16±0.17	1.69	39.79
+LIDAO	0.43±0.46	**0.03**	40.16	0.78±0.62	**1.09**	42.06	0.16±0.17	**0.00**	37.17
+eLIDAO	0.39±0.45	2.40	**38.69**	0.71±0.66	3.91	**34.85**	0.15±0.17	9.10	**35.70**
Falcon	0.41±0.46	6.39	27.40	0.65±0.61	2.83	27.00	0.15±0.16	12.64	27.40
+PPLM	0.46±0.46	**2.12**	70.17	0.72±0.68	8.16	71.51	0.13±0.14	21.79	57.83
+UDDIA	0.47±0.46	4.54	**29.03**	0.77±0.66	**2.56**	30.40	0.12±0.14	14.94	29.26
+LIDAO	0.41±0.46	2.96	30.87	0.70±0.65	2.59	**28.49**	0.13±0.14	10.16	28.01
+eLIDAO	0.44±0.46	2.74	30.51	0.75±0.53	7.61	38.45	0.14±0.15	**3.95**	**26.92**

Table 2. Performance of debiasing the three LMs (shadowed in different colors) with respect to joint $a(\boldsymbol{c}, \boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. The best debiased and fluent results are in **bold** and the second best results are underlined. GPT-2 and OPT used the min-based (e)LIDAO and Falcon used the prod-based (e)LIDAO.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
GPT-2	0.35±0.44	1.90	23.20	0.69±0.56	3.66	23.23	0.14±0.16	16.84	23.20
+GeDi	0.64±0.44	2.21	47.11	-	-	-	0.08±0.11	7.12	49.00
+DExperts	0.37±0.45	0.68	**16.55**	-	-	-	0.14±0.16	**1.00**	**16.55**
+PPLM	0.37±0.45	1.10	37.06	0.71±0.55	3.13	38.30	0.13±0.15	1.22	28.02
+UDDIA	0.41±0.45	2.29	33.85	0.72±0.56	3.46	35.29	0.12±0.14	11.74	33.43
+LIDAO	0.38±0.45	2.11	28.43	0.73±0.54	0.29	30.09	0.12±0.14	8.28	31.27
+eLIDAO	0.38±0.45	**0.46**	29.25	0.73±0.55	**0.11**	**29.79**	0.12±0.14	5.91	30.70
OPT	0.34±0.44	1.04	25.52	0.69±0.58	4.53	25.42	0.19±0.20	0.78	23.62
+PPLM	0.40±0.46	4.44	52.10	0.70±0.65	3.90	47.59	0.16±0.18	**1.78**	39.50
+UDDIA	0.40±0.45	1.07	63.83	0.70±0.61	2.19	68.34	0.15±0.17	2.94	42.64
+LIDAO	0.40±0.45	2.96	43.31	0.74±0.62	9.09	45.97	0.14±0.17	6.29	39.72
+eLIDAO	0.39±0.45	**0.83**	**41.38**	0.70±0.64	**2.16**	**36.50**	0.14±0.17	4.80	**37.75**
Falcon	0.40±0.45	2.28	29.44	0.70±0.63	1.60	29.24	0.14±0.16	0.79	29.44
+PPLM	0.46±0.46	1.41	72.77	0.72±0.67	7.11	74.72	0.11±0.13	19.38	61.56
+UDDIA	0.45±0.46	**0.81**	31.79	0.75±0.66	**1.90**	32.68	0.11±0.13	3.97	32.55
+LIDAO	0.40±0.45	1.61	**31.67**	0.73±0.67	3.10	**30.64**	0.11±0.13	1.30	**30.42**
+eLIDAO	0.43±0.46	1.57	31.90	0.76±0.54	2.43	39.47	0.12±0.15	**1.03**	30.99

2020; Wang et al., 2023b). However, these methods are often resource-consuming, hindering their practicality for larger LMs. Consequently, the decoding-time intervention paradigm is favoured (Liu et al., 2021; Yang et al., 2023). Representative decoding-time interventions consist of using reference models to guide generations (Liang et al., 2020; Krause et al., 2021; Liu et al., 2021), or injecting adversarial triggers to stimulate unbiased generations (Sheng et al., 2020; Schick et al., 2021). However, many existing methods lack a formal debiasing formulation, making them largely unprincipled. Recent works (Yang et al., 2023; Wang et al., 2023b) incorporated mutual information (MI) in their bias mitigation formulation, but they only utilized MI to characterize the dependency between the global properties and generated tokens. In this work, we propose to formulate the global bias with MI directly. Notably, under the proposed formulation, existing methods can be seen as imposing some constant constraints on the LM, steering it to generate texts where every token is agnostic to either the demographic group or the global property, regardless of the necessity to do so. Taking insights from this observation, we establish a principled way to debias the LM under a sufficient but

Table 3. Samples from debiasing sentiment (top), regard (middle), and toxicity (bottom) on three LLMs with UDDIA and (e)LIDAO as shadowed in different colors. All results are generated from the first prompt for simple illustrations. On the undebiased generations, we color the tokens that **should be** independent with property g and gender a as suggested by LIDAO, respectively.

Prompt: *Instead, these men watched the man they say humiliated*
[GPT-2] them walk away with his wallet – and one man was heckled and screamed at by the gang after ...
[UDDIA] them on the national television. "We lost our jobs after that interview," one recalls. "We...
[LIDAO] them with his boldness and brash style of play. "I watched him walk in... **[eLIDAO]** them, take them to task for their religious practices and then offer them assistance...
[OPT] them with a smile and a laugh while taking their own photos. Then when he saw something...
[UDDIA] and tortured them die and medals of honor were to be later awarded for dogged service, this is
[LIDAO] and tortured them for years - a man who has been in and out of prison for more than 12... **[eLIDAO]** and tortured them for years - a man who has now been ordered to stand for his crimes at the...
[Falcon] the black male and did nothing. But today , that's not so much the case...
[UDDIA] them, approached a woman and asked for sex - but after they say it was in self defense...
[LIDAO] them, approached a woman and asked her if he could take a seat next to her in a crowded... **[eLIDAO]** them, approached a woman and asked her if she would like to join them on the dance floor...

Table 4. Portions of human preferred generations. Each LM is evaluated on one task, respectively. Winning results are in bold.

	Human Preferrence Portion (↑)					
	LIDAO	Tied	UDDIA	eLIDAO	Tied	UDDIA
GPT-2 (Sent.)	**0.229**	0.571	0.2	**0.357**	0.3	0.343
OPT (Regard)	**0.357**	0.314	0.329	0.314	0.371	0.314
Falcon (Detox.)	**0.3**	0.429	0.271	**0.286**	0.443	0.271

weaker constraint. Compared with existing methods, our solution is capable of achieving a better fairness-fluency trade-off both theoretically and empirically.

6. Conclusion and Limitations

We propose LIDAO towards a better fairness-fluency trade-off in debiasing text generation with language models (LMs). Through a theoretical analysis, we reveal that existing methods imposed excessive constraints to the LMs during the interventions, hindering them from achieving the optimal trade-offs. On the contrary, LIDAO debiases the LMs with only limited interventions, thereby achieves a better trade-off both theoretically and empirically. We further extend LIDAO to adversarial scenarios, where an adversarial prompt may stimulate large language models (LLMs) to generate harmful text that is biased only if it is situated in the context from the prompt. Relying on the instruction-following ability of LLMs, a lightweight heuristic is proposed to make the potential bias explicit within the generated text, whereby LIDAO can be applied for mitigation. Throughout extensive experiments, the proposed method outperforms representative baselines in mitigating bias while maintaining high fluency by a large margin.

One limitation of our work is that LIDAO only reduces, but does not eliminate the bias, which we presume accounts for the inexact proxies. As in prior works, we rely on some mutual information proxies to optimize; and the referred gender in a text is also defined on a word frequency base. All these approximations can be inaccurate and result in the imperfect results. We leave exploring more effective mutual information proxies and ways to determine the discriminated demographic groups as our future direction. Meanwhile, the evaluation of fluency and global property: sentiment, regard, and toxicity, are algorithmic-based (e.g., with a sentiment-analysis tool and Perspective API). However, these definitions can be context-specific and subjective, and the current evaluation paradigm is far from ideal, and it is crucial to develop a better one. Second, this work focuses on the gender bias because of its prominent importance in the literature. Due to the lack of off-the-shelf benchmark dataset that contains high quality adversarial prompts to encourage toxic and biased generations against different race and aged groups, we plan to conduct these studies in the next step. Finally, this work focuses on the mono-lingual text generation using three (L)LMs. Given the increasing trend of multi-lingual and multi-cultural real-world applications, we leave the direction of expanding the proposed method for cross-lingual transfer as our future work.

Acknowledgement

This work is supported in part by the US National Science Foundation under grant NSF IIS-2141037 and NSF IIS-2226108. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. The datasets used in this work and sample generations may include potentially offensive or biased contents. We acknowledge that exposure to these contents can be unpleasant or uncomfortable. However, the use of these materials is to better understand, examine, and mitigate the harmful generations from language models.

References

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

Basta, C., Costa-Jussà, M. R., and Casas, N. Evaluating the underlying gender bias in contextualized word embeddings. *arXiv preprint arXiv:1904.08783*, 2019.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pp. 610–623, 2021.

Bengio, Y., Ducharme, R., and Vincent, P. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

Blodgett, S. L., Barocas, S., Daumé III, H., and Wallach, H. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5454–5476, 2020.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

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

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Cohen, R., Geva, M., Berant, J., and Globerson, A. Crawling the internal knowledge-base of language models. *arXiv preprint arXiv:2301.12810*, 2023.

Cover, T. M. *Elements of information theory*. John Wiley & Sons, 1999.

Dathathri, S., Madotto, A., Lan, J., Hung, J., Frank, E., Molino, P., Yosinski, J., and Liu, R. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2019.

Dhamala, J., Sun, T., Kumar, V., Krishna, S., Pruksachatkun, Y., Chang, K.-W., and Gupta, R. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pp. 862–872, 2021.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Fan, A., Lewis, M., and Dauphin, Y. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*, 2018.

Fatemi, Z., Xing, C., Liu, W., and Xiong, C. Improving gender fairness of pre-trained language models without catastrophic forgetting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1249–1262, 2023.

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R., and Ahmed, N. K. Bias and fairness in large language models: A survey. *arXiv preprint arXiv:2309.00770*, 2023.

Gehman, S., Gururangan, S., Sap, M., Choi, Y., and Smith, N. A. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *arXiv preprint arXiv:2009.11462*, 2020.

Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*, 2019.

Huang, P.-S., Zhang, H., Jiang, R., Stanforth, R., Welbl, J., Rae, J., Maini, V., Yogatama, D., and Kohli, P. Reducing sentiment bias in language models via counterfactual evaluation. *arXiv preprint arXiv:1911.03064*, 2019.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.

Kraskov, A., Stögbauer, H., Andrzejak, R. G., and Grassberger, P. Hierarchical clustering using mutual information. *Europhysics Letters*, 70(2):278, 2005.

Krause, B., Gotmare, A. D., McCann, B., Keskar, N. S., Joty, S., Socher, R., and Rajani, N. F. GeDi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 4929–4952, 2021.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

Li, Y., Du, M., Song, R., Wang, X., and Wang, Y. A survey on fairness in large language models. *arXiv preprint arXiv:2308.10149*, 2023.

Liang, P. P., Li, I. M., Zheng, E., Lim, Y. C., Salakhutdinov, R., and Morency, L.-P. Towards debiasing sentence representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5502–5515, 2020.

Liang, P. P., Wu, C., Morency, L.-P., and Salakhutdinov, R. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning*, pp. 6565–6576, 2021.

Liu, A., Sap, M., Lu, X., Swayamdipta, S., Bhagavatula, C., Smith, N. A., and Choi, Y. DExperts: Decoding-time controlled text generation with experts and anti-experts. 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)*, pp. 6691–6706, 2021.

Liu, T., Wang, H., Wu, F., Zhang, H., Li, P., Su, L., and Gao, J. Towards poisoning fair representations. In *The Twelfth International Conference on Learning Representations*, 2024.

Lu, K., Mardziel, P., Wu, F., Amancharla, P., and Datta, A. Gender bias in neural natural language processing. *Logic, Language, and Security: Essays Dedicated to Andre Scedrov on the Occasion of His 65th Birthday*, pp. 189–202, 2020.

Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., Agirre, E., Heintz, I., and Roth, D. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys*, 56(2):1–40, 2023.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

Pedreshi, D., Ruggieri, S., and Turini, F. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 560–568, 2008.

Peng, X., Li, S., Frazier, S., and Riedl, M. Reducing non-normative text generation from language models. In *Proceedings of the 13th International Conference on Natural Language Generation*, pp. 374–383, 2020.

Poole, B., Ozair, S., Van Den Oord, A., Alemi, A., and Tucker, G. On variational bounds of mutual information. In *International Conference on Machine Learning*, pp. 5171–5180. PMLR, 2019.

Pozzobon, L., Ermis, B., Lewis, P., and Hooker, S. Goodtriever: Adaptive toxicity mitigation with retrieval-augmented models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5108–5125, 2023.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

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

Ranaldi, L., Ruzzetti, E. S., Venditti, D., Onorati, D., and Zanzotto, F. M. A trip towards fairness: Bias and de-biasing in large language models. *arXiv preprint arXiv:2305.13862*, 2023.

Salinas, A., Penafiel, L., McCormack, R., and Morstatter, F. " im not racist but...": Discovering bias in the internal knowledge of large language models. *arXiv preprint arXiv:2310.08780*, 2023.

Saunders, D. and Byrne, B. Reducing gender bias in neural machine translation as a domain adaptation problem. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7724–7736, 2020.

Schick, T., Udupa, S., and Schütze, H. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in nlp. *Transactions of the Association for Computational Linguistics*, 9:1408–1424, 2021.

See, A., Pappu, A., Saxena, R., Yerukola, A., and Manning, C. D. Do massively pretrained language models make better storytellers? *arXiv preprint arXiv:1909.10705*, 2019.

Sheng, E., Chang, K.-W., Natarajan, P., and Peng, N. The woman worked as a babysitter: On biases in language generation. *arXiv preprint arXiv:1909.01326*, 2019.

Sheng, E., Chang, K.-W., Natarajan, P., and Peng, N. Towards Controllable Biases in Language Generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3239–3254, 2020.

Sheng, E., Chang, K.-W., Natarajan, P., and Peng, N. Societal biases in language generation: Progress and challenges. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 2021.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, 2013.

Sweeney, C. and Najafian, M. A transparent framework for evaluating unintended demographic bias in word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1662–1667, 2019.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wang, B., Chen, W., Pei, H., Xie, C., Kang, M., Zhang, C., Xu, C., Xiong, Z., Dutta, R., Schaeffer, R., et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv preprint arXiv:2306.11698*, 2023a.

Wang, R., Cheng, P., and Henao, R. Toward fairness in text generation via mutual information minimization based on importance sampling. In *International Conference on Artificial Intelligence and Statistics*, pp. 4473–4485, 2023b.

Wei, A., Haghtalab, N., and Steinhardt, J. Jailbroken: How does llm safety training fail? *arXiv preprint arXiv:2307.02483*, 2023.

Xu, A., Pathak, E., Wallace, E., Gururangan, S., Sap, M., and Klein, D. Detoxifying language models risks marginalizing minority voices. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2390–2397, 2021.

Yang, Z., Yi, X., Li, P., Liu, Y., and Xie, X. Unified detoxifying and debiasing in language generation via inference-time adaptive optimization. In *The Eleventh International Conference on Learning Representations*, 2023.

Zaken, E. B., Ravfogel, S., and Goldberg, Y. Bitfit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv preprint arXiv:2106.10199*, 2021.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *International conference on machine learning*, pp. 325–333. PMLR, 2013.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

Zhao, J., Wang, T., Yatskar, M., Cotterell, R., Ordonez, V., and Chang, K.-W. Gender bias in contextualized word embeddings. *arXiv preprint arXiv:1904.03310*, 2019.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

Zhou, C., Li, Q., Li, C., Yu, J., Liu, Y., Wang, G., Zhang, K., Ji, C., Yan, Q., He, L., et al. A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *arXiv preprint arXiv:2302.09419*, 2023.

A. Omitted Proofs

In this section we provide the omitted proofs.

A.1. Proof of Thm 2.1

We start with restating Thm 2.1. Recall that $\boldsymbol{x}_{<t} \triangleq (x_1, \ldots, x_{t-1})$.

Theorem. For sentence $\boldsymbol{x} = (x_1, \ldots, x_T)$ generated by a LM that involves some global property g and demographic group a. If at every step $t > 1$, condition

$$\begin{aligned} \ell(g;t) &\triangleq I(g; x_t \mid \boldsymbol{x}_{<t}) = 0, \\ \text{or} \quad \ell(a;t) &\triangleq I(a; x_t \mid \boldsymbol{x}_{<t}) = 0 \end{aligned}$$

holds, then $I(g;a) = 0$. In words, if each x_t is relevant to only g or a, then g and a are independent with each other.

The proof of Thm 2.1 relies on the following auxiliary Lemma.

Lemma A.1. *Follow the notations in Thm 2.1, the change in mutual information between g and a caused by token x_t at step t is given by*

$$I(g; a \mid \boldsymbol{x}_{<t+1}) - I(g; a \mid \boldsymbol{x}_{<t}) = I(g; x_t \mid \boldsymbol{x}_{<t}, a) - I(g; x_t \mid \boldsymbol{x}_{<t}).$$

Proof of Lem A.1. The proof follows the chain rule of the mutual information (Cover, 1999)

$$\begin{aligned} & I(g; a \mid \boldsymbol{x}_{<t+1}) \\ =& I(g; a, \boldsymbol{x}_{<t+1}) - I(g; \boldsymbol{x}_{<t+1}) \\ =& I(g; a, \boldsymbol{x}_{<t}, x_t) - I(g; \boldsymbol{x}_{<t}, x_t) \\ \overset{(i)}{=}& \{I(g; \boldsymbol{x}_{<t}) + I(g; a \mid \boldsymbol{x}_{<t}) + I(g; x_t \mid \boldsymbol{x}_{<t}, a)\} \\ & - \{I(g; \boldsymbol{x}_{<t}) + I(g; x_t \mid \boldsymbol{x}_{<t})\} \\ =& I(g; a \mid \boldsymbol{x}_{<t}) + I(g; x_t \mid \boldsymbol{x}_{<t}, a) - I(g; x_t \mid \boldsymbol{x}_{<t}), \end{aligned}$$

where (i) applies the chain rule to the two terms separately. Reorganize the equation completes the proof. □

With Lem A.1, we are ready to prove Thm 2.1.

Proof of Thm 2.1. First, we note that

$$I(g; a \mid \boldsymbol{x} = (x_1, \ldots, x_T)) = 0,$$

i.e., when the sentence $\boldsymbol{x}$ is determined, its property g and a are no more random and has zero mutual information thereof. In addition, if at every step t, the mutual information change

$$I(g; a \mid \boldsymbol{x}_{<t+1}) - I(g; a \mid \boldsymbol{x}_{<t}) = 0,$$

then the induction holds

$$\begin{aligned} I(g; a) &\overset{(i)}{=} I(g; a \mid \boldsymbol{x}_{<1}) \\ &= \ldots \\ &= I(g; a \mid \boldsymbol{x}_{<T+1}) \\ &= 0, \end{aligned}$$

where (i) holds from the convention that $\boldsymbol{x}_{<1} = x_0 =$ <BOS> is a special token to remark the begin of sentence $\boldsymbol{x}$ that tells the LM to start generating $\boldsymbol{x}$.

Now, according to Lem A.1, it suffices to require

$$\begin{aligned}0 =& I(g; a \mid \boldsymbol{x}_{<t+1}) - I(g; a \mid \boldsymbol{x}_{<t}) \\ =& I(g; x_t \mid \boldsymbol{x}_{<t}, a) - I(g; x_t \mid \boldsymbol{x}_{<t}) \\ =& I(g, a; x_t \mid \boldsymbol{x}_{<t}) - I(a; x_t \mid \boldsymbol{x}_{<t}) - I(g; x_t \mid \boldsymbol{x}_{<t}).\end{aligned} \tag{5}$$

This condition will be satisfied if the conditional independence $a \perp x_t \mid \boldsymbol{x}_{<t}$ or $g \perp x_t \mid \boldsymbol{x}_{<t}$ holds. To see this, note that g and a are symmetric in Eqn 5; to avoid cluttering, we take a as an instance and the same logic applies to g directly.

Now let's assume $a \perp x_t \mid \boldsymbol{x}_{<t}$, then by definition

$$\begin{aligned}& I(g, a; x_t \mid \boldsymbol{x}_{<t}) \\ =& \mathbb{E}_{p(\boldsymbol{x}_{<t})}\mathbb{E}_{p(g,a,x_t|\boldsymbol{x}_{<t})}\left[\log \frac{p(g, a, x_t \mid \boldsymbol{x}_{<t})}{p(g, a \mid \boldsymbol{x}_{<t})p(x_t \mid \boldsymbol{x}_{<t})}\right] \\ =& \mathbb{E}_{p(\boldsymbol{x}_{<t})}\mathbb{E}_{p(g,a,x_t|\boldsymbol{x}_{<t})}\left[\log \frac{p(g, a, x_t \mid \boldsymbol{x}_{<t})}{p(g \mid \boldsymbol{x}_{<t})p(a \mid \boldsymbol{x}_{<t})p(x_t \mid \boldsymbol{x}_{<t})}\right] \\ & - \mathbb{E}_{p(\boldsymbol{x}_{<t})}\mathbb{E}_{p(g,a,x_t|\boldsymbol{x}_{<t})}\left[\log \frac{p(g, a \mid \boldsymbol{x}_{<t})}{p(g \mid \boldsymbol{x}_{<t})p(a \mid \boldsymbol{x}_{<t})}\right] \\ =& I(g; a, x_t \mid \boldsymbol{x}_{<t}) - I(g; a \mid \boldsymbol{x}_{<t}) \\ \overset{(i)}{=}& (I(g; x_t \mid \boldsymbol{x}_{<t}) + I(g; a \mid \boldsymbol{x}_{<t})) - I(g; a \mid \boldsymbol{x}_{<t}) \\ =& I(g; x_t \mid \boldsymbol{x}_{<t}),\end{aligned}$$

where (i) holds from $a \perp x_t \mid \boldsymbol{x}_{<t}$. Thereby Eqn (5) reduces to

$$\begin{aligned}& I(g, a; x_t \mid \boldsymbol{x}_{<t}) - I(a; x_t \mid \boldsymbol{x}_{<t}) - I(g; x_t \mid \boldsymbol{x}_{<t}) \\ =& I(g; x_t \mid \boldsymbol{x}_{<t}) - I(a; x_t \mid \boldsymbol{x}_{<t}) - I(g; x_t \mid \boldsymbol{x}_{<t}) \\ =& 0,\end{aligned}$$

as again $a \perp x_t \mid \boldsymbol{x}_{<t}$ and this completes the proof.

□

A.2. Proof of Prop 3.1

We start with restating Prop 3.1.

Proposition. For sentence $\boldsymbol{x} = (x_1, \ldots, x_T)$ generated by a LM that is prompted by $\boldsymbol{c}$, i.e., $\boldsymbol{x} \sim p_\theta(\boldsymbol{x} \mid \boldsymbol{c})$, then

$$I(g(\boldsymbol{x}); a(\boldsymbol{c}, \boldsymbol{x})) \leq I(g(\boldsymbol{x}); a(\boldsymbol{x})) + H(a(\boldsymbol{c}, \boldsymbol{x}) \mid a(\boldsymbol{x})),$$

where $a(\boldsymbol{c}, \boldsymbol{x})$ and $a(\boldsymbol{x})$ denote the referred demographic group in the concatenation of $(\boldsymbol{c}, \boldsymbol{x})$ and generation $\boldsymbol{x}$, respectively.

Proof. The proof is largely adopted from Liu et al. (2024) and relies on the variation of information (VI, Kraskov et al. (2005)), a well-defined metric that satisfies the triangle inequality. Specifically, for random variable $a(\boldsymbol{c}, \boldsymbol{x})$ and $a(\boldsymbol{x})$, their VI is defined as

$$VI(a(\boldsymbol{c}, \boldsymbol{x}), a(\boldsymbol{x})) \triangleq H(a(\boldsymbol{c}, \boldsymbol{x})) + H(a(\boldsymbol{x})) - 2I(a(\boldsymbol{c}, \boldsymbol{x}); a(\boldsymbol{x})), \tag{6}$$

where $H(\cdot)$ is the entropy. Apply the triangle inequality to $a(\boldsymbol{c}, \boldsymbol{x})$, $a(\boldsymbol{x})$, and $g(\boldsymbol{x})$, we have

$$VI(a(\boldsymbol{x}), g(\boldsymbol{x})) \leq VI(a(\boldsymbol{c}, \boldsymbol{x}), a(\boldsymbol{x})) + VI(a(\boldsymbol{c}, \boldsymbol{x}), g(\boldsymbol{x})).$$

Plug in Eqn (6) and reorganize the result, we have

$$\begin{aligned}-I(g(\boldsymbol{x}); a(\boldsymbol{x})) \leq& \{H(a(\boldsymbol{c}, \boldsymbol{x})) - I(a(\boldsymbol{c}, \boldsymbol{x}); a(\boldsymbol{x}))\} \\ & - I(a(\boldsymbol{c}, \boldsymbol{x}); g(\boldsymbol{x})) \\ I(a(\boldsymbol{c}, \boldsymbol{x}); g(\boldsymbol{x})) \leq& I(g(\boldsymbol{x}); a(\boldsymbol{x})) + H(a(\boldsymbol{c}, \boldsymbol{x}) \mid a(\boldsymbol{x})).\end{aligned}$$

This completes the poof. □

B. Instructions for Encouraging Mentions of Demographic Groups

In this section we present the sample instructions for the three LMs when using them as the reference models, please see Sec 3 for more details. Given each instruction, we replace `#GENDER#` with the referred gender and prepend the instruction to the prompt.

Table 5. Templates used for instructing the three LMs to achieve high affinity $a(\boldsymbol{x}) = a(\boldsymbol{c}; \boldsymbol{x})$. We break templates into several lines for the sake of illustration.

	Instruct the LM to mention the demographic group in the generation so that $a(\boldsymbol{x}) = a(\boldsymbol{c}; \boldsymbol{x})$.

GPT-2	`Please help me complete the sentence in #Input#.`
	`You should focus on the mentioned #GENDER# in #Input# as frequently as possible.`
	`\n\n#Input#:\n`

OPT	`Please help me complete the sentence in #Input#.`
	`You should focus on the mentioned #GENDER# in #Input# as frequently as possible.`
	`\n\n#Input#:\n`

Falcon	`#Instruction#:\nYour objecive is to continue a given text in #Input#.`
	`If a #GENDER# presents in #Input#, you SHOULD mention this #GENDER# as frequently and explicitly as possible.`
	`\n\n#Input#:\n`

C. Implementation Details

In this section we provide more implementation details. As presented in Sec 4, GeDi, and DExperts used default hyper-parameters. We tuned PPLM for OPT and Falcon. OPT takes PPLM updates for 10 steps with learning rate 0.2 per generation step. Falone takes PPLM updates for 3 steps with learning rate 0.2 per generation step.

In both UDDIA and (e)LIDAO, we omitted the *redo* mechanism and applied the bias-tuning to the top 18 layers. Following Yang et al. (2023), we take one gradient descent step with the Adam optimizer (Kingma & Ba, 2014). Table 6 reports detailed hyper-parameters. All algorithms used the Nucleus Sampling (Holtzman et al., 2019). Note that sampling parameters are determined on a model basis.

Finally, we follow previous works and sanitize generation by filtering out texts that have perplexity larger than 200.

Table 6. Hyper-parameters used in our experiment. On GPT-2 only learning rate (marked with $*$) was tuned. All other choices were adopted from Yang et al. (2023).

	Sampling			Bias-Tuning (UDDIA & (e)LIDAO only)		Mixed Weight (UDDIA & (e)LIDAO only)
	Probability Coverage	Temperature	Repetition Penalty	Learning Rate	Top Layers to Tune	τ
GPT-2	0.9	1.	1.	0.01*	18	0.9
OPT	0.9	0.75	1.2	0.006	18	0.8
Falcon	0.9	0.75	1.2	0.01	18	0.8

D. More Experimental Results

In this section we present more experimental results.

Table 7 and 8 present full results of using LIDAO for debiasing. Here (e)LIDAO-m denotes the min-based (e)LIDAO and (e)LIDAO-p denotes the prod-based (e)LIDAO, respectively. We note that the simpler GPT-2 and OPT favored (e)LIDAO-m and Falcon preferred (e)LIDAO-p. Nonetheless, both of them achieved comparable debiasing performance compared with baseline approaches. Finally, We report results with perplexity evaluated based on larger LMs from the same family in Table 9 and Table 10, we note that our conclusions remain valid.

Table 7. Performance of (e)LIDAO for debiasing the three LMs (shadowed in different colors) with respect to generation $a(\boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. Perplexity of all generations are evaluated using GPT-2 XL.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
GPT-2	0.36±0.44	4.41	22.17	0.71±0.55	3.89	22.03	0.16±0.17	9.35	22.17
+LIDAO-p	0.39±0.45	2.96	31.88	0.80±0.50	4.35	29.51	0.13±0.14	13.54	25.68
+eLIDAO-p	0.42±0.46	5.12	28.72	0.77±0.55	2.97	29.20	0.13±0.13	3.36	25.44
+LIDAO-m	0.40±0.45	0.32	27.49	0.69±0.55	2.15	28.02	0.14±0.15	9.88	31.45
+eLIDAO-m	0.40±0.45	0.21	28.26	0.72±0.55	1.53	28.08	0.14±0.15	5.91	30.20
OPT	0.33±0.43	0.98	23.62	0.67±0.58	11.58	23.63	0.19±0.20	0.78	23.62
+LIDAO-p	0.41±0.46	0.16	50.50	0.79±0.58	7.76	49.53	0.17±0.17	1.29	36.98
+eLIDAO-p	0.40±0.46	3.29	47.58	0.71±0.63	2.85	38.38	0.17±0.18	7.93	33.10
+LIDAO-m	0.43±0.46	0.03	40.16	0.78±0.62	1.09	42.06	0.16±0.17	0.00	37.17
+eLIDAO-m	0.39±0.45	b2.40	38.69	0.71±0.66	3.91	34.85	0.15±0.17	9.10	35.70
Falcon	0.41±0.46	6.39	27.40	0.65±0.61	2.83	27.00	0.15±0.16	12.64	27.40
+LIDAO-p	0.41±0.46	2.96	30.87	0.70±0.65	2.32	30.07	0.13±0.14	10.16	28.01
+eLIDAO-p	0.44±0.46	2.74	30.51	0.75±0.53	5.56	34.16	0.14±0.15	3.95	26.92
+LIDAO-m	0.44±0.46	3.95	32.02	0.68±0.64	2.32	15.18	0.13±0.15	7.80	30.10
+eLIDAO-m	0.45±0.46	6.25	31.19	0.71±0.66	5.48	15.17	0.13±0.15	7.80	29.67

Table 8. Performance of LIDAO for debiasing the three LMs (shadowed in different colors) with respect to joint $a(\boldsymbol{c}, \boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. Perplexity of all generations are evaluated using GPT-2 XL.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
GPT-2	0.35±0.44	1.90	23.20	0.69±0.56	3.66	23.23	0.14±0.16	16.84	23.20
+LIDAO-p	0.37±0.45	1.67	29.69	0.76±0.53	4.93	29.34	0.11±0.13	2.94	26.85
+eLIDAO-p	0.38±0.45	2.12	28.63	0.75±0.54	4.98	29.24	0.11±0.13	5.21	26.53
+LIDAO-m	0.38±0.45	2.11	28.43	0.73±0.54	0.29	30.09	0.12±0.14	8.28	31.27
+eLIDAO-m	0.38±0.45	0.46	29.25	0.73±0.55	0.11	29.79	0.12±0.14	5.91	30.70
OPT	0.34±0.44	1.04	25.52	0.69±0.58	4.53	25.42	0.19±0.20	0.78	23.62
+LIDAO-p	0.41±0.46	2.42	51.01	0.75±0.61	4.40	51.88	0.15±0.17	5.77	38.06
+eLIDAO-p	0.41±0.46	1.11	50.17	0.71±0.63	7.53	41.33	0.15±0.18	2.66	36.06
+LIDAO-m	0.40±0.45	2.96	43.31	0.74±0.62	9.09	45.97	0.14±0.17	6.29	39.72
+eLIDAO-m	0.39±0.45	0.83	41.38	0.70±0.64	2.16	36.50	0.14±0.17	4.80	37.75
Falcon	0.40±0.45	2.28	29.44	0.70±0.63	1.60	29.24	0.14±0.16	0.79	29.44
+LIDAO-p	0.40±0.45	1.61	31.67	0.73±0.67	3.10	30.64	0.11±0.13	1.30	30.42
+eLIDAO-p	0.43±0.46	1.57	31.90	0.76±0.54	2.43	39.47	0.12±0.15	1.03	30.99
+LIDAO-m	0.41±0.45	2.75	31.04	0.68±0.64	3.38	31.79	0.11±0.14	5.28	32.79
+eLIDAO-m	0.41±0.45	2.71	32.29	0.72±0.55	3.51	35.51	0.11±0.15	8.05	32.43

Table 9. Performance of debiasing the three LMs (shadowed in different colors) with respect to generation $a(\boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. Perplexity of OPT and Falcon generations are evaluated using OPT-13B and Falcon-40B respectively.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
OPT	0.33±0.43	0.98	10.93	0.67±0.58	11.58	10.91	0.19±0.20	0.78	10.93
+PPLM	0.40±0.46	4.77	27.33	0.72±0.63	5.10	27.57	0.17±0.18	2.27	18.92
+UDDIA	0.39±0.45	4.88	38.69	0.74±0.63	1.98	39.90	0.16±0.17	1.69	20.67
+LIDAO-p	0.41±0.46	0.56	28.56	0.79±0.58	7.76	26.94	0.17±0.17	1.29	17.56
+eLIDAO-p	0.40±0.46	3.29	25.21	0.71±0.63	2.92	28.91	0.17±0.18	7.93	16.07
+LIDAO-m	0.43±0.46	0.06	20.21	0.78±0.62	1.16	22.25	0.16±0.17	0.00	18.69
+eLIDAO-m	0.39±0.45	2.61	20.39	0.71±0.66	3.91	25.19	0.15±0.17	9.10	18.38
Falcon	0.41±0.46	6.39	11.94	0.65±0.61	2.83	11.95	0.15±0.16	12.64	11.94
+PPLM	0.46±0.46	1.42	45.99	0.72±0.68	8.03	48.35	0.13±0.14	21.79	35.19
+UDDIA	0.47±0.46	4.54	14.92	0.77±0.66	2.56	14.56	0.12±0.14	14.94	13.57
+LIDAO-p	0.41±0.46	2.96	15.53	0.70±0.65	2.59	15.84	0.13±0.14	10.16	12.77
+eLIDAO-p	0.44±0.46	2.74	15.42	0.75±0.53	7.61	27.76	0.14±0.15	3.95	12.97
+LIDAO-m	0.44±0.46	3.95	15.98	0.68±0.64	2.32	15.18	0.13±0.15	7.80	14.00
+eLIDAO-m	0.45±0.46	6.25	16.50	0.71±0.66	5.48	15.17	0.13±0.15	7.80	14.33

Table 10. Performance of debiasing the three LMs (shadowed in different colors) with respect to joint $a(\boldsymbol{c}, \boldsymbol{x})$. Bias terms are multiplied by 100 for better comparison. Perplexity of OPT and Falcon generations are evaluated using OPT-13B and Falcon-40B respectively.

	Sentiment			Regard			Toxicity		
	Sent (↑)	Bias (↓)	PPL (↓)	NNRe (↑)	Bias (↓)	PPL (↓)	Toxi (↓)	Bias (↓)	PPL (↓)
OPT	0.34±0.44	1.04	10.92	0.69±0.58	4.53	10.94	0.18±0.20	0.78	10.92
+PPLM	0.40±0.46	4.38	28.89	0.70±0.65	4.12	27.12	0.16±0.18	1.78	18.96
+UDDIA	0.40±0.45	1.03	38.11	0.70±0.61	2.17	41.48	0.15±0.17	2.94	21.54
+LIDAO-p	0.41±0.46	2.49	27.63	0.75±0.61	4.32	27.06	0.15±0.17	5.77	17.35
+eLIDAO-p	0.41±0.46	1.11	26.17	0.71±0.63	7.83	28.76	0.15±0.18	2.66	16.86
+LIDAO-m	0.40±0.45	2.93	20.56	0.74±0.62	9.19	22.62	0.14±0.17	6.29	19.49
+eLIDAO	0.39±0.45	0.56	20.68	0.70±0.64	2.16	25.32	0.14±0.17	4.80	18.45
Falcon	0.40±0.45	2.28	12.45	0.70±0.63	1.60	12.43	0.14±0.16	0.79	12.45
+PPLM	0.46±0.46	1.04	47.93	0.72±0.68	6.59	49.08	0.11±0.13	19.38	36.64
+UDDIA	0.45±0.46	0.81	14.63	0.75±0.66	1.90	14.89	0.11±0.13	3.97	14.29
+LIDAO-p	0.40±0.45	1.61	15.73	0.73±0.67	3.10	15.73	0.11±0.13	1.30	13.41
+eLIDAO-p	0.43±0.46	1.57	16.10	0.76±0.54	2.43	28.31	0.12±0.15	1.03	13.94
+LIDAO-m	0.41±0.45	2.75	15.12	0.68±0.64	3.38	14.89	0.11±0.14	5.28	14.75
+eLIDAO-m	0.41±0.45	2.71	16.03	0.72±0.55	3.51	15.20	0.11±0.15	8.05	14.96