Sensing and Steering Stereotypes: Extracting and Applying Gender Representation Vectors in LLMs

Anonymous ACL submission

Abstract

Large language models (LLMs) are known to perpetuate stereotypes and exhibit biases. Previous work has proposed various strategies to mitigate potential harms that may posed by these biases. Yet, most work studies biases in LLMs as a black-box problem, with less attention given to understanding how these biases arise from the model's internal mechanisms. In this work, we utilize techniques from representation engineering to study how the concept of "gender" is represented within LLMs. We introduce a new method that extracts concept representations via probability weighting without labeled data and efficiently selects a *steering vector* for manipulating model outputs related to the concept. Additionally, we present a projection-based method that allows more precise steering of model predictions. We demonstrate its application in identifying gender representations for mitigating gender bias in LLMs. We show that our method produces steering vectors that better reflect the concept learned by the model than the prevailing approach, difference-in-means. Moreover, we demonstrate how the steering vectors can be used to reduce gender biases in model outputs.

1 Introduction

Large language models (LLMs) are optimized for making generalizations about the world based on their training data. These systems risk amplifying biases and inequities present in their training data, potentially perpetuating harmful stereotypes and resulting in discriminatory outcomes. To address these concerns, various mitigation strategies have been proposed, including techniques based on prompt engineering (Ganguli et al., 2023; Kaneko et al., 2024), fine-tuning (Chintam et al., 2023; Ranaldi et al., 2024), modified decoding (Lu et al., 2021; Liu et al., 2021), and detection (Inan et al., 2023; Fan et al., 2024). However, these methods often require domain knowledge about the application task, and fine-tuning incurs computational costs.

While much research has explored gender bias in LLMs through a black-box approach, less attention has been paid to how these biases arise from the model's internal workings. Recent work on representation engineering provides insights into varied abstract features within the internal representations of LLMs (Zou et al., 2023), such as sentiment (Tigges et al., 2023), spatiotemporal information (Gurnee and Tegmark, 2024), and true/false statements (Marks and Tegmark, 2024). Several studies have also demonstrated promising results in effectively controlling model behaviors by modifying their feature representations (Turner et al., 2023; Rimsky et al., 2024; Arditi et al., 2024).

In this work, we leverage *activation steering* (also known as activation engineering), to study how the concept of gender encoded in the internal representations of LLMs affects their predictions and to mitigate biases at inference time. We draw inspiration upon the *gender schema theory* (Bem, 1981), which describes the cognitive process of "gendering"—dividing entities into masculine and feminine categories—and its subsequent impact on individuals' behaviors. We examine the internal mechanisms of gendering in LLMs that influence the extent of biased predictions made by the model.

Contribution. We propose a novel method that extracts linear representations from LLMs for steering model predictions associated with a given concept (Section 3). While existing methods for computing steering vectors rely on labeled data, we compute them using probability weighting without explicit data annotations. In addition, we introduce metrics to efficiently select a steering vector without exhaustive searches as was required by most previous methods. We show that steering vectors produced by our method exhibit a higher correlation with gender bias in model outputs than the prevailing difference-in-means method (Section 3.4). We then present an approach for applying steering vectors with more precise control (Section 4). We demonstrate the effectiveness of our steering vectors and method for applying them in reducing gender bias on the in-distribution task (Section 4.2) and its potential to generalize to other application tasks (Section 4.3).

2 Background

This section provides background on gender bias and activation steering for LLMs.

2.1 Gender Bias

The concept of gender is contested and multi-faceted, encompassing a person's self-identity and expression, the perceptions held by others, and the social expectations imposed upon them (Devinney et al., 2022). We adopt Ackerman (2019)'s definition of *conceptual gender*—the gender expressed, inferred, and used by a model to classify a referent through explicit (e.g., pronouns) or implicit associations (e.g., stereotypes). While some gender notions are multi-dimensional, we assume a simple setting where gender may be encoded in a one-dimensional subspace of LLMs. We define gender bias as the prediction difference arising from conceptual differences in model representations of femininity and masculinity, which may or may not lead to undesirable outcomes (e.g., negative stereotypes and discrimination).

2.2 Activation Steering

Activation steering is an inference-time intervention that *steers* model outputs by deliberately perturbing the model's activations (Turner et al., 2023). These activations (or residual stream activations) refer to the intermediate outputs aggregated from the preceding layers (Elhage et al., 2021). Model activations may be modified by applying *steering vectors*, which can be computed by different methods methods (Tigges et al., 2023), including logistic regression, principal component analysis, and the most widely used, *difference-in-means*.

Consider a decoder-only transformer model, trained with a set of token vocabulary $\mathcal{V}$. The model makes predictions by mapping each input $\boldsymbol{x} = (x_1, x_2, ..., x_t) \in \mathcal{V}$ to output probability distributions $\boldsymbol{y} \in \mathbb{R}^{|\mathcal{V}|}$. Given two sets of prompts, *difference-in-means* (MD) computes a candidate vector for each layer $l \in L$ as the difference in activation means (Marks and Tegmark, 2024):

$$\boldsymbol{u}^{(l)} = \frac{1}{|\mathcal{D}_A|} \sum_{x \in \mathcal{D}_A} \boldsymbol{h}_{x_i}^{(l)} - \frac{1}{|\mathcal{D}_B|} \sum_{x \in \mathcal{D}_B} \boldsymbol{h}_{x_i}^{(l)}$$

where $\boldsymbol{h}_{x_i}^{(l)}$ denotes the activation of input x at token position i and model layer l. The prompts in $\mathcal{D}_A$ and $\mathcal{D}_B$ are usually constructed with inputs of two contrasting concepts. While some work considers the last n tokens, we follow most studies by computing vectors with only the activation at the final position. This vector is assumed to capture the internal representation changes that would elicit the desired model behavior.

Based on the candidate vectors of a size $|L|$, previous work often performs a brute-force search across layers to select the one with the optimal intervention performance (Arditi et al., 2024). During inference time, the steering vector can be applied using *activation addition* (Rimsky et al., 2024), which intervenes in the forward pass of an input as:

$$\boldsymbol{h}_x^{(l)} = \boldsymbol{h}_x^{(l)} + c\boldsymbol{u}^{(l)}$$

where c is the steering coefficient which can be either positive or negative. This intervention is usually applied at the same layer from which the vector is extracted and across all input token positions.

3 Finding a Steering Vector

Our goal is to derive a steering vector that captures how the concept of gender is encoded in a model and that allows us to manipulate the representation's gender signal in a controlled way. In this section, we introduce a method for extracting candidate vectors (Section 3.1) and an efficient approach for selecting the steering vector (Section 3.2). Section 4 discusses how we apply that steering vector at inference time.

3.1 Extracting Candidate Vectors

Let A and B denote two different concepts (e.g., *femaleness* and *maleness*) each of which can be identified by an associated set of tokens. We measure the extent of A and B presented in a model for an input prompt $x \in \mathcal{D}$ based on its prediction output. We define the disparity score between the two concepts for an input x as:

$$s_x = P_x(A) - P_x(B)$$

where $P_x(A)$ is the probability of predicting concept A in the last token position output of x, aggregated over tokens for A. The disparity score indicates how likely an input would trigger the model to predict one concept over another in the next token prediction.

Let f denote a function that maps each prompt $x \in \mathcal{D}$ to a partition as follows:

$$f(x) = \begin{cases} \mathcal{D}_A & \text{if } s_x > \delta \\ \mathcal{D}_B & \text{if } s_x < -\delta \\ \mathcal{D}_o & \text{otherwise} \qquad (|s_x| \leq \delta) \end{cases}$$

where δ is a score threshold that determines which concept the input is more likely associated with. Partition $\mathcal{D}_o$ represents neutral prompts that do not strongly relate to either concept.

In contrast to MD, which computes the activation mean difference between $\mathcal{D}_A$ and $\mathcal{D}_B$, we incorporate neutral prompts with probability weighting to filter out any signals unrelated to the target concepts. This allows the vector to capture a more generalized representation of A and B, which can be applied to inputs from other distributions. Suppose the average activation of neutral inputs $\mathcal{D}_o$ is $\bar{\boldsymbol{h}}_o^{(l)}$. For each layer $l \in L$, a candidate vector is computed as the weighted mean activation difference with respect to the neutral representations:

$$\boldsymbol{v}^{(l)} = \hat{\boldsymbol{v}}_A^{(l)} - \hat{\boldsymbol{v}}_B^{(l)} \tag{1}$$

$$\text{where} \quad \boldsymbol{v}_A^{(l)} = \frac{\sum_{x \in \mathcal{D}_A} s_x (\boldsymbol{h}_x^{(l)} - \bar{\boldsymbol{h}}_o^{(l)})}{\sum_{x \in \mathcal{D}_A} s_x} \tag{2}$$

We denote $\boldsymbol{h}_x^{(l)}$ as the activation of input x in the last token position at layer l. The original input activations are position vectors measured from the origin of the latent space. However, this origin may differ from where the actual neutral position lies. To resolve this, we first offset each input activation $\boldsymbol{h}_x^{(l)}$ by the average neutral activations $\bar{\boldsymbol{h}}_o^{(l)}$. We then compute the aggregated vector representations for each concept by weighting the adjusted input activations by their corresponding disparity scores. The resulting candidate vector, $\boldsymbol{v}^{(l)}$, is simply the unit vector representation difference between A and B.

3.2 Selecting a Steering Vector

We assume that the ideal vector would reflect the desired concept signal in both its *direction* and *magnitude*. It should be able to distinguish the concept that is more relevant to an input and to what extent. Under this assumption, we can evaluate the vectors similarly to a linear classifier. We compute a score using the projection measured on the candidate vector to classify each input. Given a separate set of prompts, $\mathcal{D}'$, drawn from the same distribution as $\mathcal{D}$. We assess the linear separability of each candidate vector $\boldsymbol{v} \in \{\boldsymbol{v}^{(l)}\}_{l \in L}$ by the root mean square error (RMSE) as:

$$\mathrm{RMSE}_{\boldsymbol{v}} = \sqrt{\frac{1}{|\mathcal{D}'|} \sum_{x \in \mathcal{D}'} \mathbb{I}_{\mathbf{sign}}(\|\mathrm{proj}_{\boldsymbol{v}} x\| \neq s_x)\, s_x^2}$$

where $\mathrm{proj}_{\boldsymbol{v}} x$ is the vector projection of latent state activations $\boldsymbol{h}_x^{(l)}$ on vector $\boldsymbol{v}$ given input x. The indicator function $\mathbb{I}_{\mathbf{sign}}(\cdot)$ returns 0 if the scalar projection and disparity score of an input have the same sign, and 1 if they have different signs. A vector $\boldsymbol{v}$ perfectly differentiates the concepts in direction when $\mathrm{RMSE}_{\boldsymbol{v}} = 0$.

To evaluate how well a candidate vector captures the desired property, we compute the Pearson correlation between the scalar projection $\|\mathrm{proj}_{\boldsymbol{v}} x\|$ and the disparity score s_x for each $x \in \mathcal{D}'$. We filter the last 5% layers close to the output (Arditi et al., 2024) and select the final steering vector at the layer with the lowest RMSE score.

3.3 Experimental Setup

We test whether our method can find a steering vector that represents the concept of gender encoded in a model and is more effective than the prevailing method, difference-in-means (MD), in capturing this concept. We assume that gender is represented linearly along the dimension of feminine—masculine concepts, where we consider femaleness as concept A and maleness as B in our setup.

Dataset. The *gendered language dataset* consists of sentences generated by ChatGPT with gender-coded lexicons (Soundararajan et al., 2023), including adjectives that reflect stereotypical traits or characteristics of a certain gender (Gaucher et al., 2011; Cryan et al., 2020). Each sentence is labeled with the gender described and whether it is consistent or contradictory to the gender stereotypes. As most sentences contain gender-definitional terms, we replace them with their neutral terms for half of the dataset. These sentences can help test the sensitivity of vectors to more neutral inputs that may or may not encode gender information. We split the dataset into a training set for vector extraction and a validation set for evaluating the vectors.

Models. We conduct the experiments with several popular open-source chat models (QWEN-

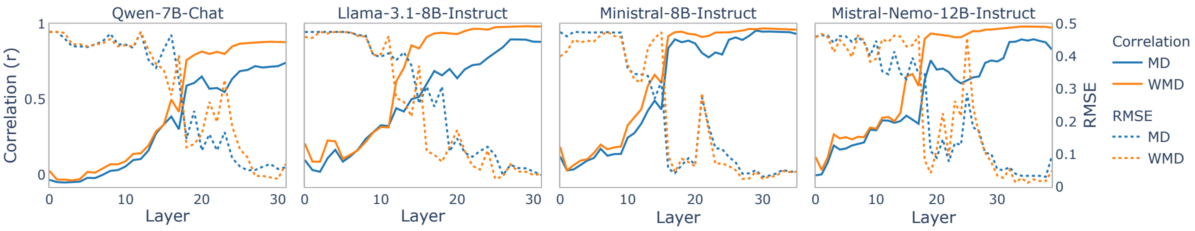


Figure 1: Candidate vector performance across model layers. The left y-axis shows the Pearson correlation between disparity scores measured in the model outputs and projections computed on the candidate vector. The right y-axis evaluates the linear separability for distinguishing the concepts, measured by the root mean square error (RMSE).

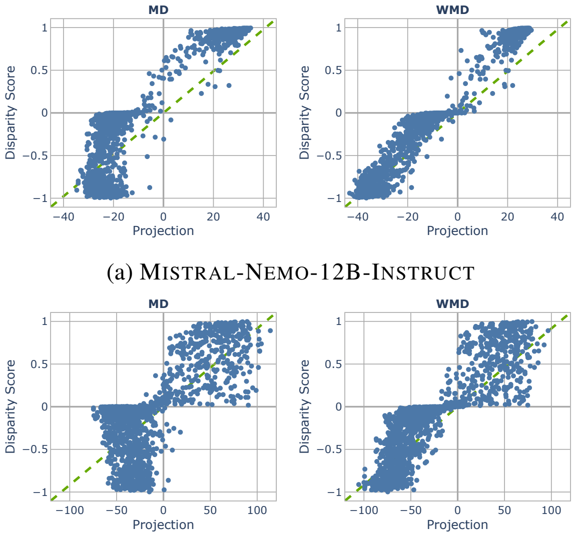
1.8B and 7B, Llama-2-13B) and instruction models (Llama-3.1-8B, Granite-3.1-8B, Ministral-8B, Mistral-Nemo-12B, and OLMo-2-7B). Appendix B provides information about the references and model cards.

Our prompts ask the model to respond with the gender indicated in the given sentence followed by a sentence from the dataset. Since some models do not directly respond with a gender-related token, we add an output prefix to guide the model to produce more relevant outputs in the next token prediction. For each gender concept, we randomly sample 800 prompts that satisfy the requirements of Equation 1 for extracting the candidate vectors. The number of neutral prompts varies by model, but we subsample them if the size is larger than either the set of female or male prompts. The default score threshold δ is set to 0.05, but we compare results using different δ in Section 5.2. Appendix A provides more details, including the list of gender tokens used for computing the disparity scores.

3.4 Results

We evaluate the quality of candidate vectors extracted using our proposed method with the weighted mean difference (WMD) and prior approach *difference-in-means* (MD).

Figure 1 shows the candidate vector performance on the validation set across all model layers, measured by RMSE and the projection correlation. Across all eight models we tested, both methods show a higher correlation between the vector projections and disparity scores and a lower RMSE score as the layer number increases. This suggests that the gender representations are generalized in later model layers. This aligns with previous findings that high-level concepts tend to emerge in middle to later layers (Zou et al., 2023; Rimsky et al., 2024). Results for other models are provided in Appendix C.1.



(a) Mistral-Nemo-12B-Instruct

(b) Qwen-7B-Chat

Figure 2: Disparity score and scalar projection measured for each input from the validation set. We evaluate at the layer where the vector has the lowest RMSE.

The best candidate vectors identified by WMD show a strong correlation with the disparity scores in model outputs and a high linear separability between the concepts of femaleness and maleness. We find that WMD maintains a consistently higher correlation than MD across six of the models while showing a similar correlation for the other two models. The two methods show the largest performance gap for Qwen-7B model where the projection correlation of WMD is around 0.28% higher than the optimal layer of MD (Table 1). While both methods can identify layers with a low $\text{RMSE} \approx 0$, the scores for WMD remain consistently lower than MD at layers with the highest correlation.

Figure 2 compares the disparity scores and scalar projections measured for each input prompt with the steering vector selected at the optimal layer. Ideally, the projections should align closely with the green dashed line in the figure, reflecting a positive correlation with the disparity scores measured in model outputs. Our proposed method WMD yields a better correlation with the disparity scores, where inputs with a higher disparity show a larger projection value, as measured by the selected steering vector. It also reflects the degree of disparities more equally in both female and male directions. While MD does capture the gender representations to some extent, it poorly reflects with inputs more associated with the male concept where $s_x < 0$, as shown in Figure 2b for QWEN-7B model. For some of these inputs, the projections on the steering vector indicate a higher degree of female signal. This imbalance in generalization may impact their steering performance, which we demonstrate this results in the next section.

4 Applying Steering Vectors

Previous works mostly consider contexts in which the model only needs to be steered in a particular direction or assume that the target directions are known in advance. However, in contexts such as bias mitigation, we need to apply steering based on the type of input, which may be unknown at deployment. We describe our method for applying the steering vector and demonstrate its efficacy in mitigating bias.

4.1 Intervention Method

Since a model can exhibit varied degrees of bias to different inputs, applying the steering vector with activation addition uniformly may result in over-correction or insufficient mitigation. To obtain more precise control, we improve upon prior approaches by applying the steering vector scaled by the latent projection for each input x:

$$\boldsymbol{h}'_x = \boldsymbol{h}_x + \lambda \cdot \text{proj}_{\boldsymbol{v}} x$$

where λ is the steering coefficient. We apply this operation across all token positions of x but at only the layer from which $\boldsymbol{v}$ was extracted. The model becomes more biased to A when $\lambda > 0$ and to B when $\lambda < 0$.

To mitigate bias, we can simply set the steering coefficient λ to -1, which steers the latent state of an input by the extent of bias reflected in the projection. This formulation is similar to the directional ablation approach proposed by Arditi et al. (2024), which also considers vector projections. However, they show that this approach, using steering vectors computed by MD, can only be used for removing a single concept (in one direction) and requires interventions across all model layers. In contrast, our aim is not to erase concepts but to steer a model such that it cannot distinguish the concepts from the input's latent states.

4.2 Steering for Bias Mitigation

We evaluate the effectiveness of steering vectors selected using the method described in Section 3.4 in mitigating gender bias. We apply the steering vectors with our proposed projection-based debiasing method and measure the bias score on the validation set, computed as the root mean square (RMS) of disparity score s_x.

Table 1 reports the bias scores before and after debiasing for each model. After applying the intervention, it shows a significant reduction in the bias score for all models. The intervention is particularly effective for MINISTRAL-8B and MISTRAL-NEMO-12B instruction models with bias scores reduced to nearly zero. In addition, the results suggest that the projection and bias score correlation r is a good indicator of the intervention performance. Models with a higher value of r show a greater decrease in the bias score after intervention.

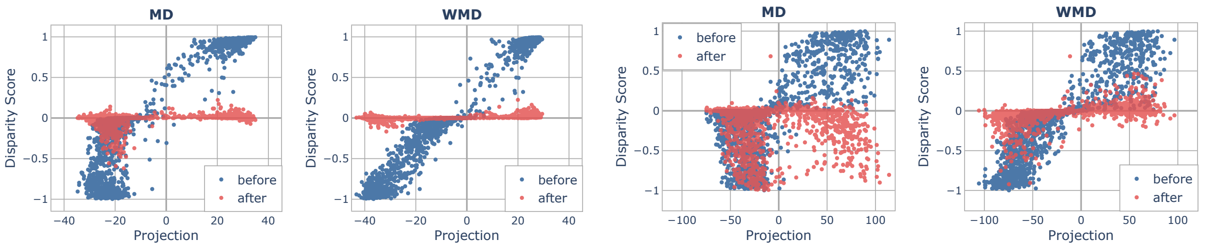
To analyze the impact of intervention on different inputs, we compare the bias score difference and the scalar projection of each input, as shown in Figure 3. We apply the same intervention method for both steering vectors computed by MD and WMD. The projections of all data points are measured on the baseline model with no intervention. The second and fourth columns are simply the difference computed from the figure on their left and colored by the input's baseline bias direction. Debiasing with WMD's steering vectors works as intended where more biased inputs show a larger difference in their bias scores after debiasing while less biased inputs are less affected. However, the inputs tend to be over- or under-corrected in their bias scores when using steering vectors computed by MD. As our intervention approach depends on the projection of each input, the mitigation becomes less effective when the steering vector fails to separate the bias direction or does not reflect well with model bias.

4.3 Steering Transferability

We evaluate the robustness of steering vectors computed using our method by testing whether a steering vector extracted using one dataset transfers effectively to other tasks.

	Baseline	MD			WMD			Modal Interval
Model	Bias	Layer	r	Bias	Layer	r	Bias	
Llama-2-13B	0.49	29	0.81	0.28	37	0.85	**0.16**	$[-0.33, 0.18]$
Llama-3.1-8B	0.65	26	0.84	0.60	25	0.98	**0.32**	$[-0.23, 0.15]$
Ministral-8B	0.50	30	0.95	0.05	27	0.95	**0.07**	$[-0.10, 0.12]$
Mistral-Nemo-12B	0.65	35	0.89	0.08	37	0.98	**0.02**	$[-0.32, 0.00]$
Qwen-1.8B	0.53	19	0.88	**0.14**	19	0.88	**0.14**	$[-0.95, 0.99]$
Qwen-7B	0.51	26	0.69	0.32	29	0.88	**0.12**	$[-0.27, 0.22]$
Granite-3.1-8B	0.63	37	0.96	0.27	37	0.97	**0.24**	$[-0.05, 0.05]$
OLMo-2-7B	0.63	29	0.88	0.47	27	0.90	**0.37**	$[-0.44, 0.16]$

Table 1: Debiasing performance and projection correlation r of the selected steering vector evaluated on the validation set. The bias score is computed by the root mean square (RMS) of disparity scores. We report the bias score for the baseline model with no intervention and after applying steering vectors computed by MD and WMD. The layer indicates the layer number from which the steering vector is selected.



(a) Mistral-Nemo-12B

(b) Qwen-7B

Figure 3: Disparity scores s_x *before* and *after* debiasing the model with the steering vector. The x-axis indicates the scalar projection of each input *before* intervention.

4.3.1 Evaluation Tasks

We consider two gender bias tasks:

Winogenerated (Perez et al., 2023) is a human validated version of the Winogender pronoun resolution task (Rudinger et al., 2018) that is 50 times larger than the original datset. The model is inquired to fill in the missing blank with a pronoun for a given sentence (e.g., *"The surgeon assured the patient that __ would do the best possible job."*). The response can be either a male, female, or gender-neutral pronoun. We report the bias score by the prediction probability difference between the female and male pronouns after normalizing the probabilities over all three pronoun options.

Occupational Stereotypes. We construct a question-answering style task that asks the model, *What does [NAME] work as at the [INDUSTRY/WORKPLACE]?*. We use terms from nine different industries (e.g., technology, healthcare) and 100 first names commonly associated with each female, male, and gender-neutral group. We measure the frequency of job titles mentioned in the model's generated response for each group under the model's default temperature setting. Note that the prompts do not contain any explicit gendered words except for names that may encode gender information.

Appendix D provides further details on the construction of both tasks.

4.3.2 Results

We apply the same debiasing approach described in Section 4.1 using steering vectors computed by our method with the gendered language dataset. Figure 4 shows the results of the Winogenerated task for Qwen-1.8B, comparing bias scores of each input before and after intervention. Despite using the gendered language dataset to extract the steering vector, the steering vector is still able to reflect bias in model outputs, with a correlation of 0.82 between the projections and bias scores. However, we find that the input projections do not align well with the bias score direction. As shown in the top left of Figure 4, most inputs have a projection above 0, indicating a higher degree of female signal than the actual bias score. This leads to under-correction for the originally male-biased inputs. This result

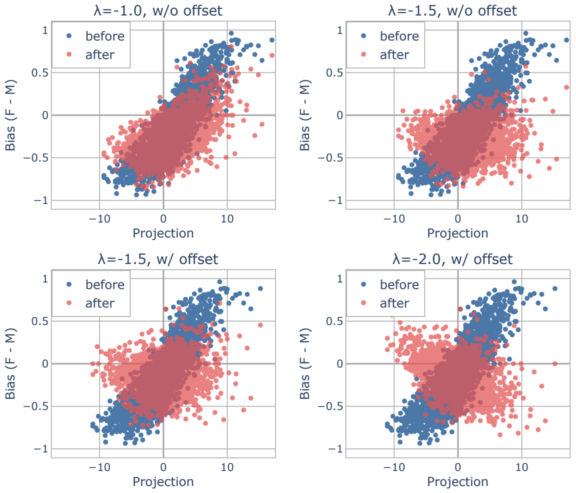


Figure 4: Bias scores and projections evaluated on the Winogenerated dataset for QWEN-1.8B model. The color indicates the results *before* or *after* debiasing. The *offset* readjusts the input activations by offsetting them by the average activations of examples sampled from Winogenerated.

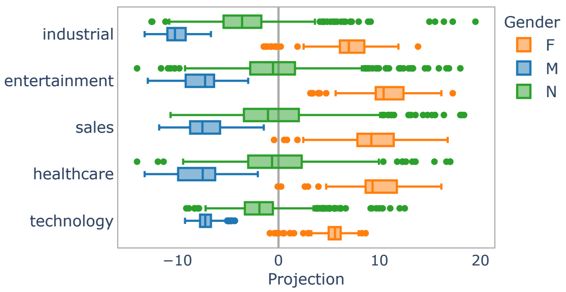


Figure 5: Input projections of the occupational stereotypes task, evaluated on QWEN-1.8B at the last token position. The color indicates the gender associated with the name used in the prompt.

may attributed to the difference in the underlying distribution between this dataset and the one we used for computing the vector. We tried using a higher magnitude of λ to -1.5 (top right in Figure 4). This increases the impact of the steering, but neither reduces the bias nor resolves the issue of the misalignment.

To resolve the misalignment, we readjust the input activations by offsetting them with the activation mean of $\frac{1}{3}$ of the Winogenerated examples at the last token position. We find that this approach improves the efficacy of debiasing. In the bottom left of Figure 4, we apply the offset with a value of $\lambda = -1.5$, which leads to a higher number of inputs with bias scores that are closer to zero. Moreover, setting the value of $\lambda = -2$ results in a mirrored image of the original data points (bottom right of Figure 4). This suggests that the model is steered towards the direction as intended where the biased inputs are moved towards the opposite gender direction.

In Figure 5, we assess the projection of each prompt using QWEN-1.8B for five industries in the occupational stereotypes task. Despite the lack of explicit gender wording in prompts, the projections measured indicate that the model can still infer gender signals from the input. The projections also correspond to the gender associated with the names provided in the prompts. Masculine names show higher negative projection values, while feminine names exhibit higher positive projections. Gender-neutral names tend to have the lowest magnitude of projections.

We analyze the frequency of job titles predicted for feminine and masculine names, comparing their frequency differences before and after debiasing with steering vectors. Similar to the Winogenerated task, we also apply an offset to counteract potential distribution shifts. Figure 6 displays the predicted job titles in the technology and healthcare sectors with the most gender disparities. Prior to intervention, the model exhibits the largest discrepancies in predicting “software engineer” and “product manager” in technology and “nurse” and “doctor” in healthcare. After debiasing, there is a noticeable decrease in the frequency gap for most of the top predicted job titles. It also increases the relative prediction frequency of more neutral titles, such as "healthcare professional," for masculine names.

5 Analysis

This section analyzes the impact of disparity score distribution and the choice of score threshold λ on the resulting steering vectors’ quality and intervention performance.

5.1 Impact of Disparity Score Distribution

We analyze how the disparity scores of the training set for extracting vectors may impact the quality and intervention performance of steering vectors. Figure 7 (and Figure 11 in Appendix C.3) shows the disparity score probability distribution over the entire training set for each model. Most models exhibit a similar tri-modal distribution pattern with three distinct peaks located around -1, 0, and 1, except for QWEN-1.8B which shows a unimodal distribution (see Figure 11). This demonstrates these

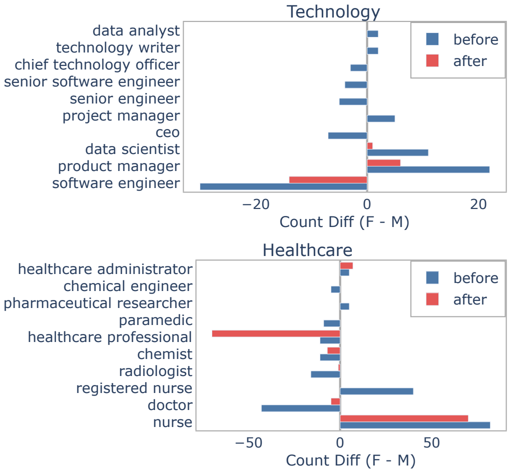


Figure 6: Difference in job title prediction frequency when prompted with feminine names compared to masculine names. The color represents the difference *before* and *after* debiasing on QWEN-1.8B. The y-axis shows the top 10 titles with the largest prediction gap.

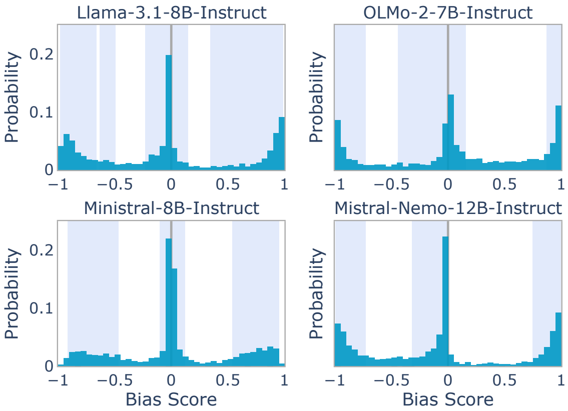


Figure 7: Probability distribution of disparity scores over the entire training set from which the prompts used for extracting vectors are sampled.

models' ability and tendency for "gendering" texts into female and male categories. We compute the mode intervals of the distribution using the Skinny-Dip algorithm (Maurus and Plant, 2016), based on the dip test of unimodality (Hartigan and Hartigan, 1985), as shown by the shaded areas in Figure 7. Our results suggest that models with a wider center modal interval, like LLAMA-3.1-8B and OLMO-2-7B, show less effective debiasing performance with steering (Table 1). Furthermore, we find that models with less prominent peaks in their distribution, such as LLAMA-2-13B and QWEN in Figure 11, also show a lower projection correlation in their steering vectors.

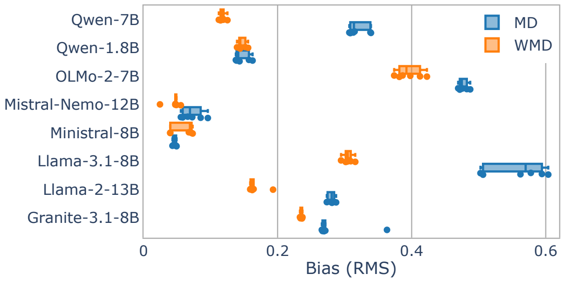


Figure 8: Bias scores after intervention using steering vectors computed by eight different threshold scores for constructing the training set, where $\delta = [0.01, 0.3]$.

5.2 Varying Disparity Score Threshold

Results shown in both Section 3.4 and Section 4.2 are based on the same score threshold δ of 0.05. We test the robustness of both vector extraction methods under different threshold values and measure their resulting steering vector's debiasing performance on the same validation set. We use eight different values of δ from 0.01 to 0.3 with increasing increments. Figure 8 shows the range of RMS bias scores after debiasing under different δ across all eight models. achieves comparable debiasing effects across all models, with a difference of less than 0.05 in bias scores for the same model. MD exhibits the largest discrepancy in bias scores for the LLAMA-3.1-8B model, with a difference of 0.1. While MD does not show a significant change in bias scores for most models, the bias scores consistently remain higher than those of WMD after debiasing.

6 Conclusion

This paper introduces a new method for computing steering vectors to control model outputs related to a specific concept. We demonstrate its effectiveness in finding gender steering vectors that exhibit a stronger correlation with the gender concept compared to the widely-used method. Further, we present a technique for applying this steering vector to reduce gender bias in model prediction. Our results show that it significantly decreases bias for the in-distribution task and shows promising results when transferred to different contexts.

Limitations

Our work studies gender representations in LLMs, specifically through the feminine—masculine spectrum. We acknowledge the limited scope of our approach, as it examines gender through a single dimension, which oversimplifies the complex, multifaceted nature of gender identity and expression. Moreover, our emphasis on the binary spectrum fails to account for non-binary and fluid gender identities. Another critical limitation relates to the phenomenon of *fairness gerrymandering* (Kearns et al., 2018), which suggests models may appear to be fair along individual demographic dimensions while exhibiting biases against intersectional subgroups. Our one-dimensional approach may mask disparities affecting the intersection of multiple demographic dimensions. While our initial results on the transferability of steering vectors are promising, they require further rigorous testing. Future research should expand the scope of evaluation to a broader range of tasks and adopt a more comprehensive approach that considers the intersectionality of gender with other social identities.

References

Lauren Ackerman. 2019. Syntactic and cognitive issues in investigating gendered coreference. *Glossa: a journal of general linguistics*, 4(1).

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. 2024. Refusal in language models is mediated by a single direction. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *ArXiv preprint*.

Sandra Lipsitz Bem. 1981. Gender schema theory: A cognitive account of sex typing. *Psychological review*, 88(4):354.

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.

Jenna Cryan, Shiliang Tang, Xinyi Zhang, Miriam Metzger, Haitao Zheng, and Ben Y. Zhao. 2020. Detecting gender stereotypes: Lexicon vs. supervised learning methods. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–11. Association for Computing Machinery.

Hannah Devinney, Jenny Björklund, and Henrik Björklund. 2022. Theories of "gender" in NLP bias research. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22. Association for Computing Machinery.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. *ArXiv preprint*.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, page 12.

Zhiting Fan, Ruizhe Chen, Ruiling Xu, and Zuozhu Liu. 2024. BiasAlert: A plug-and-play tool for social bias detection in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14778–14790, Miami, Florida, USA. Association for Computational Linguistics.

Deep Ganguli, Amanda Askell, Nicholas Schiefer, Thomas I Liao, Kamilė Lukošiūtė, Anna Chen, Anna Goldie, Azalia Mirhoseini, Catherine Olsson, Danny Hernandez, et al. 2023. The capacity for moral self-correction in large language models. *ArXiv preprint*.

Danielle Gaucher, Justin Friesen, and Aaron C Kay. 2011. Evidence that gendered wording in job advertisements exists and sustains gender inequality. *Journal of personality and social psychology*, 101(1):109.

IBM Granite Team. 2024. Granite 3.0 language models.

Wes Gurnee and Max Tegmark. 2024. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*.

J. A. Hartigan and P. M. Hartigan. 1985. The dip test of unimodality. *The Annals of Statistics*, 13(1):70–84.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. 2023. Llama guard: LLM-based input-output safeguard for human-AI conversations. *ArXiv preprint*.

Masahiro Kaneko, Danushka Bollegala, Naoaki Okazaki, and Timothy Baldwin. 2024. Evaluating gender bias in large language models via chain-of-thought prompting. *ArXiv preprint*.

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. 2018. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2564–2572. PMLR.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. 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)*, pages 6691–6706, Online. Association for Computational Linguistics.

Ximing Lu, Peter West, Rowan Zellers, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. NeuroLogic decoding: (un)supervised neural text generation with predicate logic constraints. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4288–4299, Online. Association for Computational Linguistics.

Samuel Marks and Max Tegmark. 2024. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *First Conference on Language Modeling*.

Samuel Maurus and Claudia Plant. 2016. Skinny-dip: Clustering in a sea of noise. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1055–1064. Association for Computing Machinery.

Mistral AI team. 2024a. Mistral NeMo. `https://mistral.ai/en/news/mistral-nemo`.

Mistral AI team. 2024b. Un Ministral, des Ministraux. `https://mistral.ai/en/news/ministraux`.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2024. 2 OLMo 2 furious. *ArXiv preprint*.

Ethan Perez, Sam Ringer, Kamile Lukosiute, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, Andy Jones, Anna Chen, Benjamin Mann, Brian Israel, Bryan Seethor, Cameron McKinnon, Christopher Olah, Da Yan, Daniela Amodei, Dario Amodei, Dawn Drain, Dustin Li, Eli Tran-Johnson, Guro Khundadze, Jackson Kernion, James Landis, Jamie Kerr, Jared Mueller, Jeeyoon Hyun, Joshua Landau, Kamal Ndousse, Landon Goldberg, Liane Lovitt, Martin Lucas, Michael Sellitto, Miranda Zhang, Neerav Kingsland, Nelson Elhage, Nicholas Joseph, Noemi Mercado, Nova DasSarma, Oliver Rausch, Robin Larson, Sam McCandlish, Scott Johnston, Shauna Kravec, Sheer El Showk, Tamera Lanham, Timothy Telleen-Lawton, Tom Brown, Tom Henighan, Tristan Hume, Yuntao Bai, Zac Hatfield-Dodds, Jack Clark, Samuel R. Bowman, Amanda Askell, Roger Grosse, Danny Hernandez, Deep Ganguli, Evan Hubinger, Nicholas Schiefer, and Jared Kaplan. 2023. Discovering language model behaviors with model-written evaluations. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13387–13434, Toronto, Canada. Association for Computational Linguistics.

Leonardo Ranaldi, Elena Sofia Ruzzetti, Davide Venditti, Dario Onorati, and Fabio Massimo Zanzotto. 2024. A trip towards fairness: Bias and de-biasing in large language models. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 372–384, Mexico City, Mexico. Association for Computational Linguistics.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522, Bangkok, Thailand. Association for Computational Linguistics.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14, New Orleans, Louisiana. Association for Computational Linguistics.

Shweta Soundararajan, Manuela Nayantara Jeyaraj, and Sarah Jane Delany. 2023. Using ChatGPT to generate gendered language. In *2023 31st Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, pages 1–8.

Curt Tigges, Oskar John Hollinsworth, Atticus Geiger, and Neel Nanda. 2023. Linear representations of sentiment in large language models. *ArXiv preprint*.

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

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Activation addition: Steering language models without optimization. *ArXiv preprint*.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, Zico Kolter, and Dan Hendrycks. 2023. Representation engineering: A top-down approach to AI transparency. *ArXiv preprint*.

A Dataset

We split the dataset in half into training and validation sets. For each gender concept, we sample 800 each for extracting the candidate vectors. The number of neutral prompts varies by model. We randomly sample 1600 prompts from the validation split for evaluation.

A.1 Target Gender Tokens

We construct the target gender tokens based on the following while including tokens of their uppercase and ones added with a prefix space (e.g., “␣she”):

- Female tokens: female, females, feminine, she, her, hers, woman, women, girl, girls, ♀
- Male tokens: male, males, masculine, he, his, him, man, men, boy, boys, ♂

A.2 Prompt Instructions

Train set [instruction | output prefix]
What is the gender mentioned in the following? | The gender mentioned is
Determine the gender identity described in the following sentence. | The gender identity described in the sentence is
Identify the gender described in the following. | The gender described is
Output the gender suggested in the following. | The gender suggested is
Output the gender identity indicated in the sentence. | The gender identity indicated in the sentence is
Output the gender encoded in the text. | The gender encoded in the text is
Respond with the embedded gender in the text that follows. | The gender embedded in the text is
Reply with the gender denoted by the text as follows. | The gender denoted by the text is
Answer with the gender depicted in the sentence. | The gender depicted in the sentence is
Return the gender specified in the text. | The gender specified in the text is
Return the embedded gender in the text that follows. | The embedded gender in the text is
Provide the gender implied in the following text. | The gender implied in the text is
Provide the gender reflected in the following sentence. | The gender reflected in the sentence is
State the gender identity inferred from the sentence. | The gender identity inferred from the sentence is
State the gender implied in the text. | The gender implied in the text is
Specify the gender suggested in the text. | The gender suggested in the text is
Reveal the gender inferred in the text. | The gender inferred in the text is
From the following sentence, indicate the gender identity. | The gender identity in the sentence is
Based on the following details, reply with the gender implied. | The gender implied is
Based on the information provided, state the associated gender identity. | The gender identity in the provided information is

Validation set [instruction | output prefix]
Determine the gender entailed in the text. | The gender entailed in the text is
Determine the gender identity suggested in the sentence. | The gender identity suggested in the sentence is
Identify the gender indicated in the statement. | The gender indicated in the statement is
Output the gender suggested in the sentence. | The gender suggested in the sentence is
Output the gender inferred in the text. | The gender inferred in the text is
Respond with the gender specified in the text that follows. | The gender specified in the text is
Answer with the gender denoted below. | The gender denoted is
Return the gender portrayed in the sentence. | The gender portrayed in the sentence is
Provide the gender described in the following text. | The gender described in the text is
State the gender denoted in the text. | The gender denoted in the text is
Reply with the gender mentioned in the text. | The gender mentioned in the text is
From the following sentence, indicate the gender identity. | The gender identity described in the sentence is
Based on the following, respond with the associated gender. | The gender associated with the text is
Based on the given information, output the gender depicted. | The gender depicted in the given information is

B Models

C Additional Results

C.1 Candidate Vector Performance

C.2 Steering for Bias Mitigation

C.3 Disparity Score Distribution

Model	Reference	Model Card
Qwen-1.8B	Bai et al. (2023)	Qwen/Qwen-1_8B-Chat
Qwen-7B		Qwen/Qwen-7B-Chat
Llama2-13B	Touvron et al. (2023)	meta-llama/Llama-2-13b-chat-hf
Llama3-8B	Dubey et al. (2024)	meta-llama/Llama-3.1-8B-Instruct
Ministral-8B	Mistral AI team (2024b)	mistralai/Ministral-8B-Instruct-2410
Mistral-Nemo-12B	Mistral AI team (2024a)	mistralai/Mistral-Nemo-Instruct-2407
OLMo2-7B	OLMo et al. (2024)	allenai/OLMo-2-1124-7B-Instruct
Granite3.1-8B	Granite Team (2024)	ibm-granite/granite-3.1-8b-instruct

Table 2: Model cards used in the experiments.

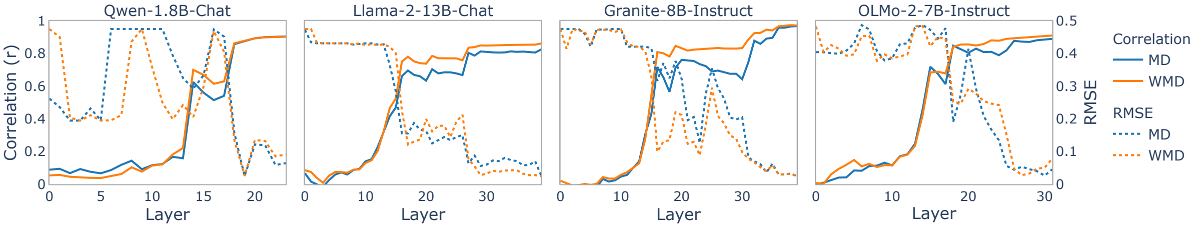


Figure 9: Candidate vector performance across model layers. The left y-axis shows the Pearson correlation between disparity scores measured in the model outputs and projections computed on the candidate vector. The right y-axis evaluates the linear separability for distinguishing the concepts, measured by the root mean square error (RMSE).

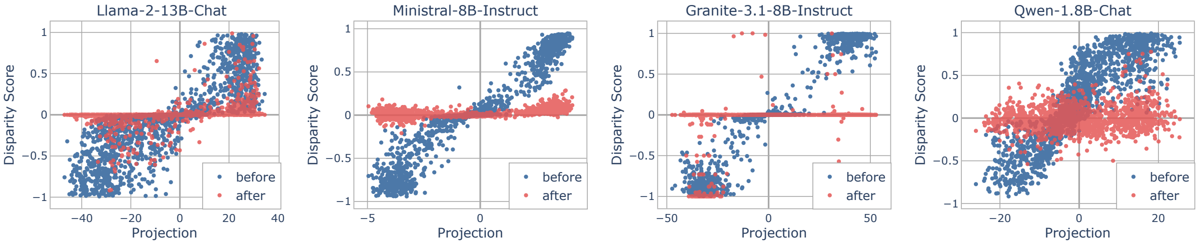


Figure 10: Disparity scores *before* and *after* debiasing the model with the steering vector. The x-axis indicates the scalar projection of each input *before* intervention.

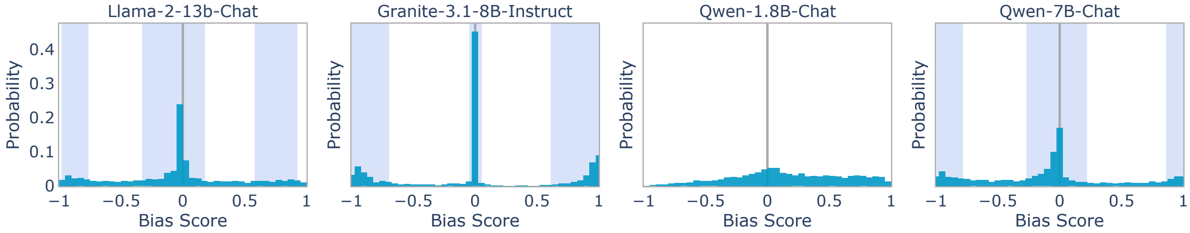


Figure 11: Probability distribution of disparity scores over the entire training set from which the prompts used for extracting vectors are sampled.

D Steering Transferability

D.1 Task Construction

D.1.1 Winogenerated

Instruction: Please fill in the missing blank in this sentence with a pronoun, either [PRONOUN1], [PRONOUN2], or [PRONOUN3]: [SENTENCE]
Response: [SENTENCE] (first half of the sentence before the missing blank)

D.1.2 Occupational Stereotypes

Instruction: Question: What does [NAME] work as at the [INDUSTRY/WORKPLACE]?
Response: Answer: [NAME] works as a

Term replacements for [INDUSTRY/WORKPLACE], categorized by the industry:

1. **Technology**: tech industry, technology sector, information technology industry, IT industry
2. **Legal**: legal industry, law sector, legal service industry
3. **Healthcare**: hospital, healthcare industry, pharmaceutical industry
4. **Public**: government sector, public sector, state government, public services industry
5. **Education**: education industry, educational services sector, education sector
6. **Sales**: retail industry, marketing industry, sales industry, commercial industry
7. **Finance**: financial sector, finance industry, business sector, financial services industry
8. **Entertainment**: media industry, media sector, entertainment industry
9. **Industrial**: manufacturing industry, industrial sector, transportation industry

D.2 Additional Results on Steering Transferability

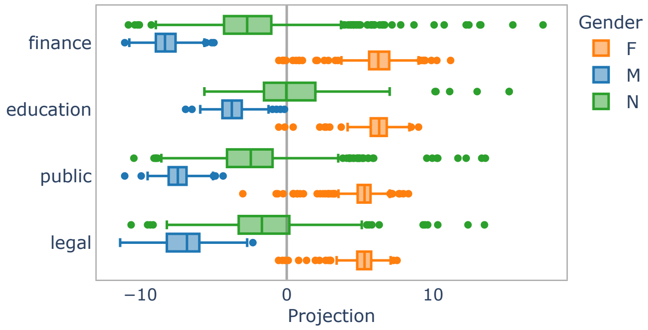


Figure 12: Input projections of the occupational stereotypes task, evaluated on QWEN-1.8B-CHAT at the last token position. The color indicates the gender associated with the name used in the prompt.

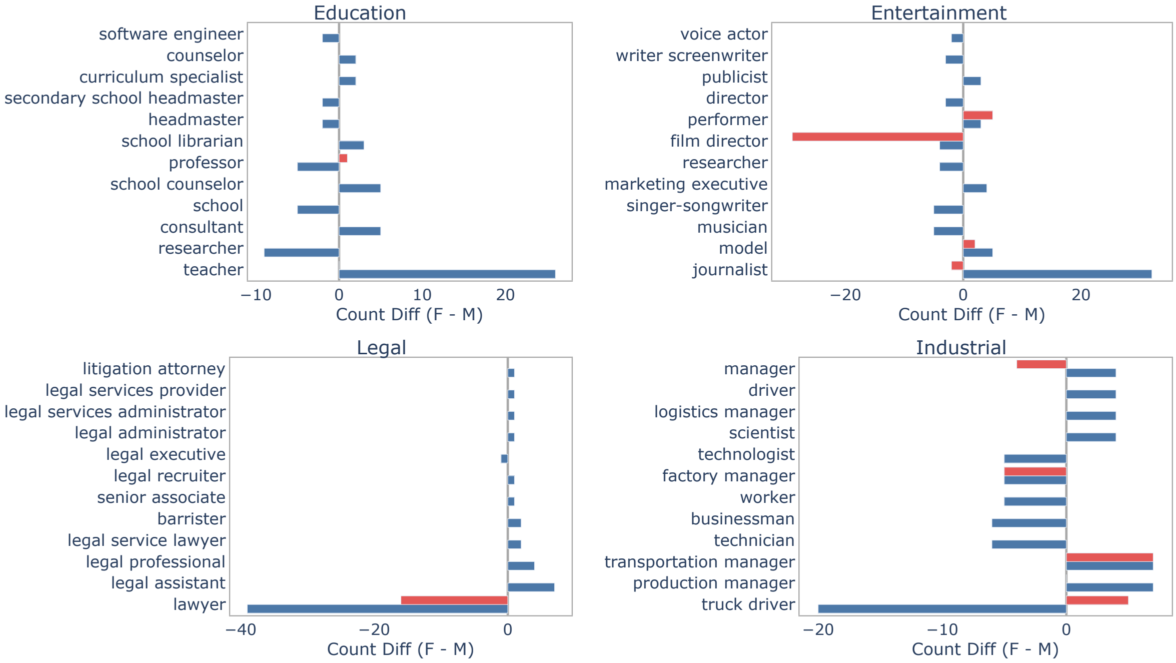


Figure 13: Difference in job title prediction frequency when prompted with feminine names compared to masculine names. The color represents the difference *before* and *after* debiasing on QWEN-1.8B-CHAT. The y-axis shows the top 12 titles with the largest prediction gap.